# ActiveMap Web admin manual 3.41.0 (5.42)

**Activemap Computer Systems Design**

**Oct 23, 2023**

# CONTENTS

**ANNOTATION**

This document is intended to provide an overview of the general information, structure, and configuration of the ActiveMap Web Program for individual conditions, requirements, and application areas of the software.

# GENERAL INFORMATION

## 1.1 About the Program

ActiveMap Web (administrator) is the administrative part of the ActiveMap Web program that allows you to add and structure a variety of geographic information, as well as configure access rights to user information and user interface settings.

ActiveMap Web is part of a multi-component web-based ActiveMap system for remote employee management.

ActiveMap is an online system for organizing the interaction between field workers and the dispatcher (task coordinator). The system helps to plan and manage the production work and to operationalize quality control of field services.

Capabilities of ActiveMap:

- Flexible customization to meet the needs of the company.

You can adapt ActiveMap to any business process. A list of work types, steps and deadlines can be set up for each organization cluster.

- Adding tasks and controlling their execution.

The system allows users to add operational and planned tasks, including scheduled tasks on a given template.

- Object inventory.

ActiveMap helps to carry out an inventory of objects: update information on the status of existing objects, identify nonexistent, and to create new ones.

- Control of field employees.

The system helps to control employees with real-time tracking of their location, viewing the history of their movement, and recording the execution of requests.

- Convenient and quick interaction between field employees and work coordinators.

ActiveMap speeds up the process of exchanging results between the field employee and the work coordinator. The coordinator can promptly update task information, which is immediately communicated to the field employee. The coordinator can also quickly return the task to the fieldworker for execution based on the results of the fieldwork.

- Using photo and video fixation materials and GPS data.

The system can verify that tasks were carried out using photos, video recordings, and location data. This avoids the necessity of field inspection of executed orders.

- User rights configuration.

The system enables the configuring of user rights. Each user is assigned a certain role. The role of the system user determines access to the list of tasks, rights to edit and manage these tasks. The roles vary from simple executors to the administrator of the entire system.

- Displaying service objects on a map.

ActiveMap allows users to create tasks based on service objects with the automatic filling out of coordinates and task fields.

- Creating electronic documents.

The system allows users to create reports on the work with tasks and user activity based on the document form of the organization, as well as invoices issued by field employees.

More information about the comprehensive capabilities of the ActiveMap system can be found on the website of the Activemap Computer Systems Design company https://activemap.me/.

## 1.2 System requirements

The Program is created using web technologies, allowing it to run from any personal computer with Internet access. To organize the dispatcher's workplace, a personal computer with technical specifications that meet the following minimum requirements is needed:

- Processor: Intel Core i3,

- Operating system: Windows 7,

- Internet access speed of at least 1 Mbps.

The Program does not require additional installation of third-party software on the workstation. The Program opens using internet browsers such as Internet Explorer, Mozilla Firefox, Opera, Google Chrome, or Microsoft Edge.

# WORKING IN THE PROGRAM

## 2.1 Running the Program

To run the Program, open an Internet browser (Internet Explorer, Mozilla Firefox, Opera, Google Chrome, Microsoft Edge) and enter the address of the web page of the Program in the address bar. The start window (Fig. 2.1) appears.



Fig. 2.1: Start window
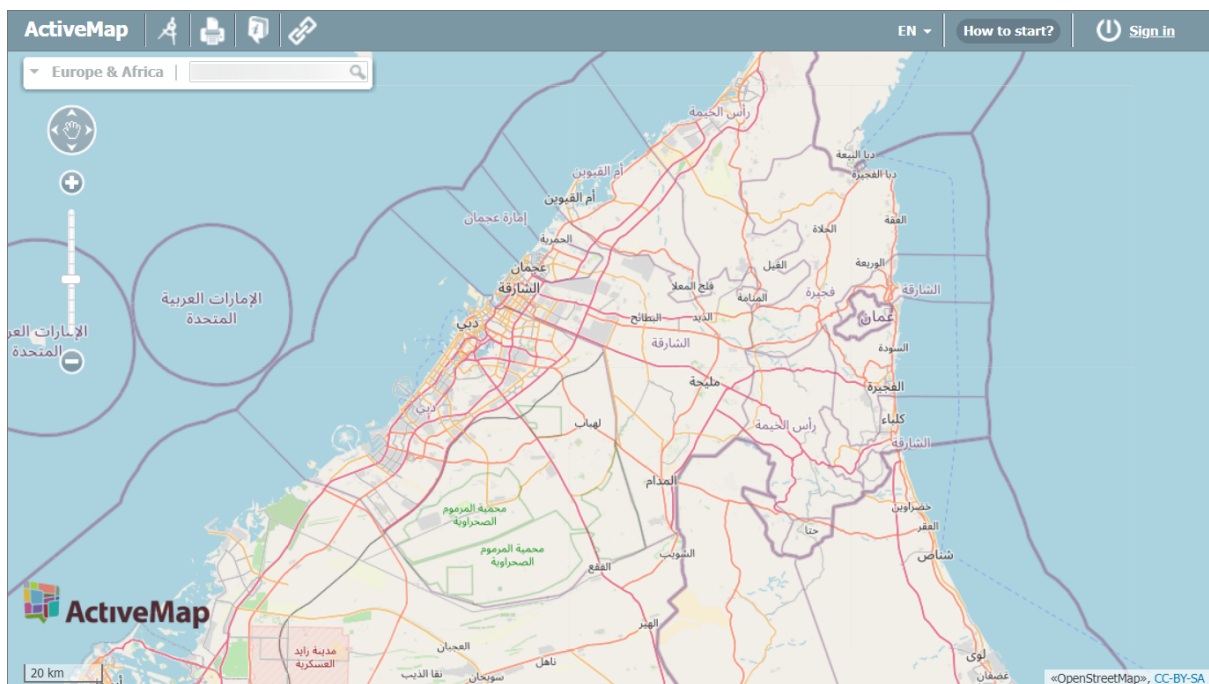
To log in you need to enter your credentials: login and password. Then the start web page is loaded (Fig. 2.2).
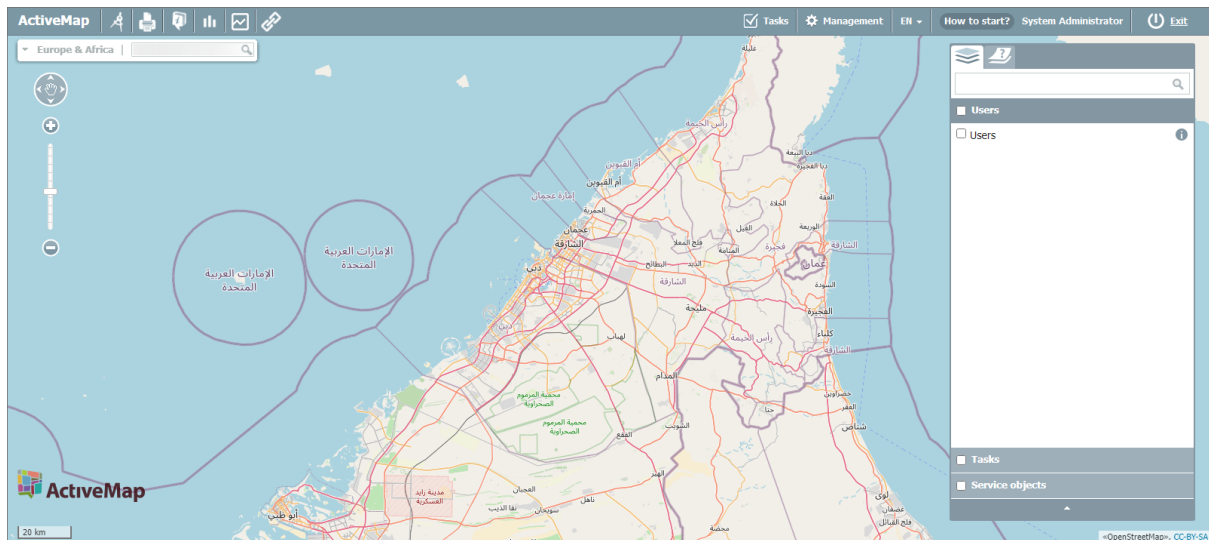
Fig. 2.2: Start page of the program

The main window of the Program contains (Fig. 2.3):

1. Map displaying area.

2. Toolbar.

3. Basemap controls with a search bar.

4. Control panel for thematic layers.

5. User panel.

6. Scale bar.

7. Scale ruler.

8. Map navigation panel.
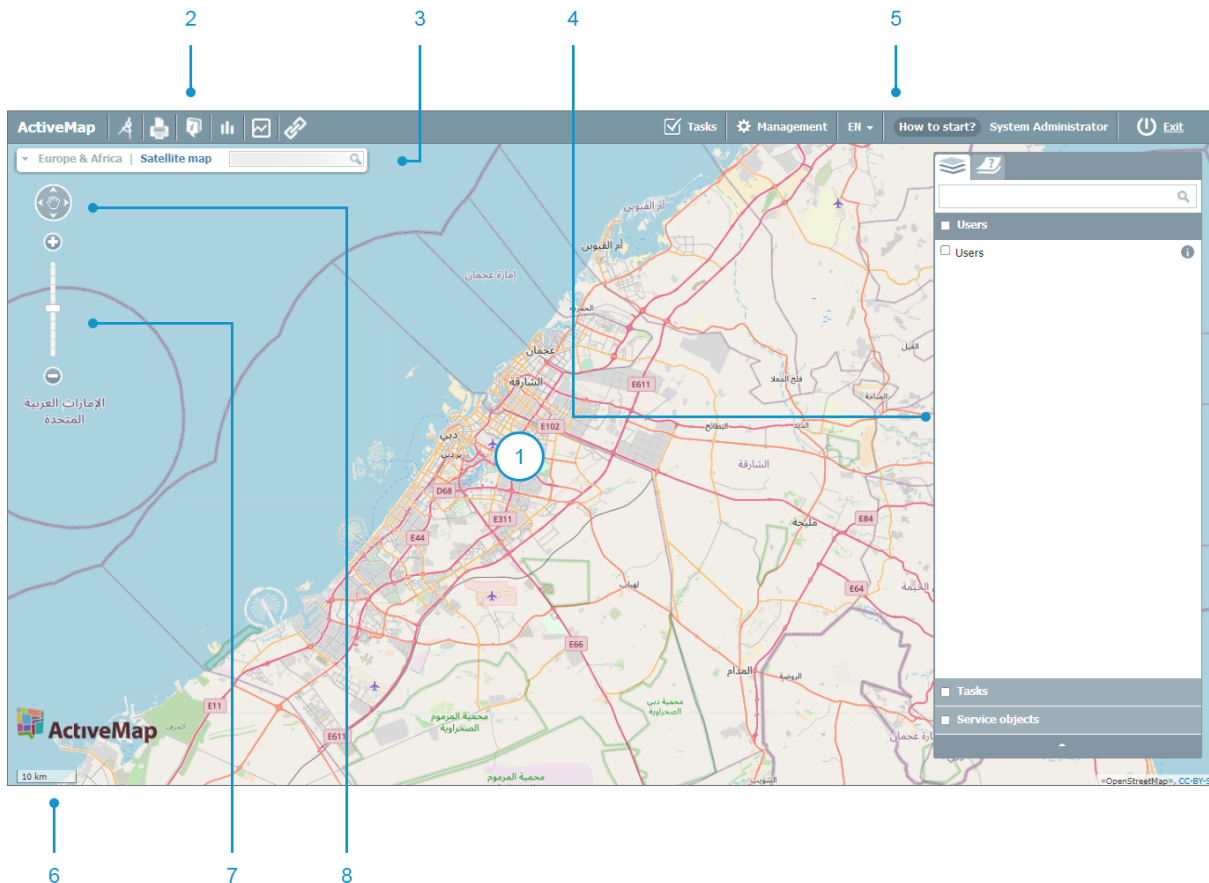
Fig. 2.3: Elements of the main window

Working with administration tools is available in the "Tasks" and "Management" modules. You can access them through the user panel at the top of the main program window (Fig. 2.4).
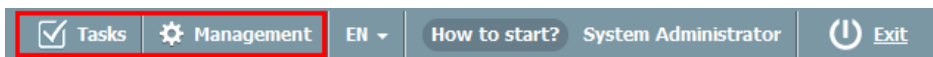


Fig. 2.4: Switching to the "Tasks" and "Management" modules

## 2.2 Tasks module

The task module allows users to work with contracts, create operational and planned tasks in the system.

Operational tasks are created to solve current issues. Planned (scheduled) tasks are created on a date and time specified in the schedule according to a given template.

You can create tasks within the contracts, concluded with an organization, and independently of them. The ability to view, create, and edit contracts and tasks depends on the user's role in the system. User roles are defined by the Organization Administrator or Cluster Administrator.

To go to the tasks module, click "Tasks" (Fig. 2.5) on the user panel of the main page of the geoportal:



Fig. 2.5: Accessing the task module

## 2.2.1 Contracts

### 2.2.1.1 Contracts in ActiveMap

A contract is an agreement for the provision of services made with an organization or client. Users with the System Administrator or Cluster Administrator permissions can create, edit, and delete contracts. Contracts operate within the cluster. The cluster is selected automatically. When creating a contract, the cluster of the customer organization is used. In another cluster, this contract is not available to users. The System Inspector, Cluster Inspector, Administrator and Inspector of the Assigned Organization have rights to view the contract. Users, who see the task created under the contract, also receive minimal information (id, title).

A contract may include a list of service objects and types of work. After the start date of the contract, it is impossible to change this date, customer or performer organizations, and the end date after the expiration of the contract.

You can create tasks only within one contract. You cannot add the same tasks to a different contract. However, you can attach multiple tasks and schedules to one contract. If necessary, you can delete a previously selected contract from the task and add a new one. If you delete a contract from the system, the ad hoc tasks created under it and the tasks created under the schedule are preserved (the contract name is displayed in the task), but the schedule itself is deleted.

### 2.2.1.2 List of contracts

To access the window with contracts in the "Tasks" module, click "Contracts" on the top panel of the page. In the opened window (Fig. 2.6), the System Administrator and System Inspector can see the entire list of contracts entered into the system. The Cluster Administrator and Cluster Inspector can view the cluster contract list. The Organization Administrator and Organization Inspector can see the organization contract list. All other users see the list of contracts for available tasks.

Fig. 2.6: List of contracts

The contracts window has a search by contract number, name, and ID. You can also set up filters by attributes:

- Cluster,

- Customer,

- Executor.

The list of contracts is presented in the form of a table, which includes all the basic information on the contract. For convenience, there is sorting in two directions, which works by clicking the attribute name.
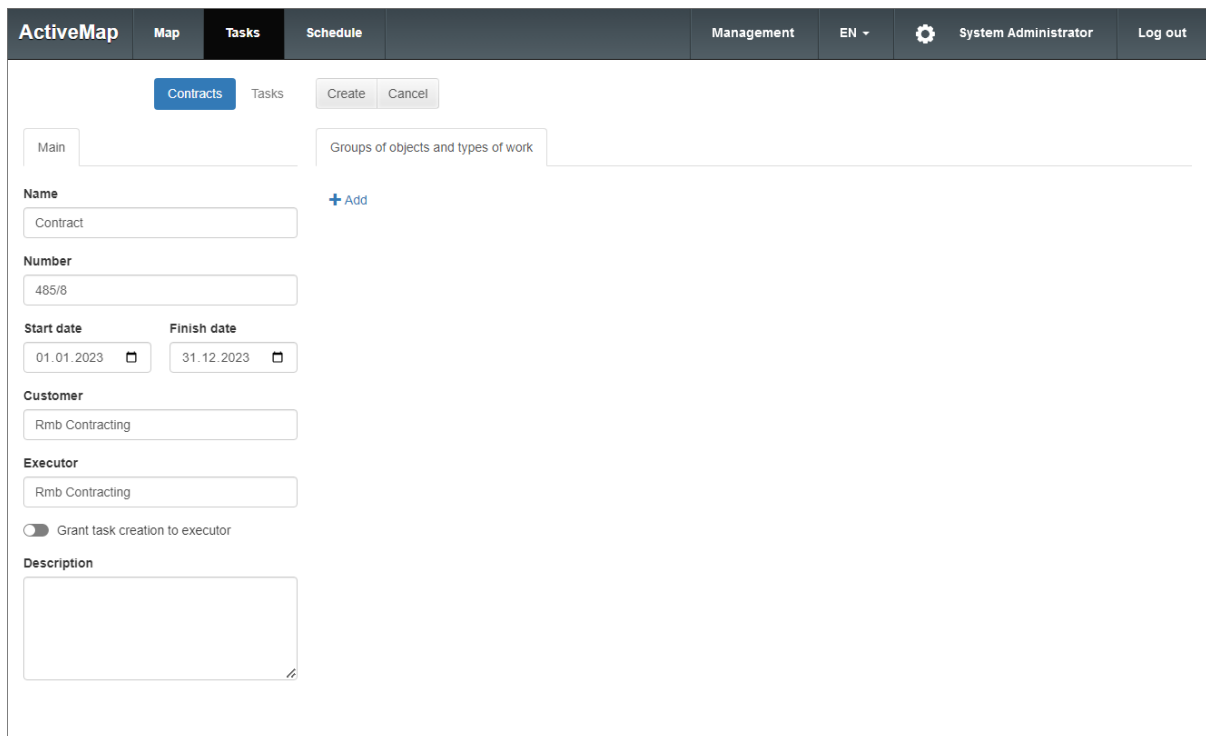
### 2.2.1.3 Creating a contract

Only users with System Administrator or Cluster Administrator permissions can create, edit, and delete contracts. To add a new contract, click "Create". A window opens (Fig. 2.7) with the following fields in the "Main" tab:

- "Name" (mandatory field) — name of the contract;

- "Number" — contract number;

- "Start date" and "Finish date" — the period of the contract, during which the work is performed and tasks are created (if these fields are not filled in, it is impossible to create an operational task or a task on a schedule);

- "Customer" (mandatory field) — the organization which cluster is assigned to the contract;

- "Executor" (mandatory field) — the organization, executing the contract, to which tasks are automatically assigned when this contract is selected;

- "Grant task creation to executor" toggle (disabled by default) – if it is disabled, users are not able to select this contract when creating a task;

- "Description" — a comment to the contract.



Fig. 2.7: Creating a contract

You can add types of work to the contract and specify the list of service objects for which to perform work. To do this, click "Add" in the "Groups of objects and types of work" tab. The block of selecting service objects and types of work is added. Click "Change" to add a value. When changing a service object, a window opens where you have to select the layer and objects. For convenience, the window has a search line and the ability to select all or individual objects (Fig. 2.8).

Fig. 2.8: Linking a contract to service objects

When you change the type of work, a window opens for selecting the required types of work. For convenience, there is a search line, the ability to select types of work, and a filter to display all types of work or only selected ones (Fig. 2.9).



Fig. 2.9: Linking a contract to work types

After saving the contract data, you can create tasks only for the selected types of work and service objects during the specified contract period. You can create several groups of objects and types of work within one contract. The Program also checks whether the selected types of work are available to the assigned organization. If the organization has no rights, the contract is not saved, as this can cause errors in the creation of tasks in the future. When selecting an object that has a connection to an organization/cluster, the task must be created in the specified organization. If this organization has no rights to the selected type of work, the contract is not saved.

After entering all the data for the contract, click "Create". The contract becomes available for attaching to tasks in the ActiveMap system software products: ActiveMap Web, ActiveMap Desktop, ActiveMap Mobile.

#### 2.2.1.4 Contract management

To edit a contract, select it and click "Edit". A window similar to the creation window opens, where you can make and save edits. To delete a contract, select it and click "Delete". When a contract is deleted, the operational tasks created under it and the tasks created according to the schedule are preserved (the name of the contract is displayed in the task), the schedule itself is deleted.

### 2.2.2 Tasks

To access the task list, click "Tasks" on the top panel of the page. In the window that opens, you can see all the tasks created so far (Fig. 2.10).



Fig. 2.10: List of tasks available to the user

The filter area located to the left of the task list allows searching in the general list by various parameters.

The following filtering parameters are used:

- "Search by Id, title or text",
- "Contract" (if you have access),
- "Organization",
- "Type of work",

- "Priority",

- "Status",

- "Step",

- "Assigned organization",

- "Executor".

The context search field allows you to find a task by number, title, or description. To search for a task, enter part of the number, name, or description. To filter tasks by status, type of work, step, priority, organization, performing organization, or performer, select values from the dropdown list. The values of these parameters are customized for the individual characteristics of the organization's activities. After entering and selecting all filtering parameters, the list displays tasks that meet the specified criteria.

### 2.2.2.1 Creating an operational task

To generate a new operational task, click the "Create" button. A window opens with "Main" and "Files" tabs (Fig. 2.11).

Fig. 2.11: New task creating window

The "Main" tab displays a map to specify the location of the object and fields to fill in the task characteristics.

To mark the location of the task object on the map, zoom in to the area of interest using

and ![icon] buttons and/or mouse and click on the map. You can also enter the address of the object in the search field. The search results are displayed as a list. When you hover over each of them, the Program moves to the specified address on the map (Fig. 2.12). If the task is not bound to a specific location, the geolocation can be omitted.
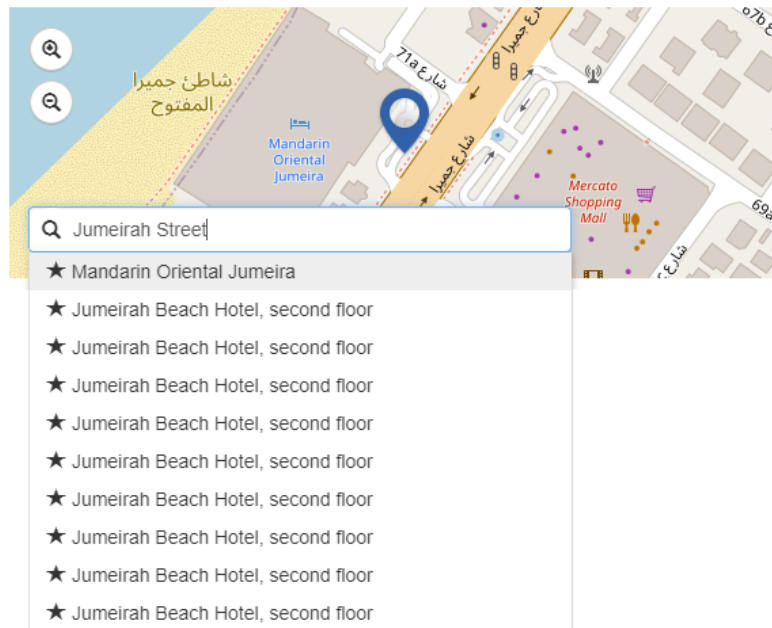
Fig. 2.12: Search for object address on the map

You can link a task to a service object. To do this, click "Select service object" above the map. Choose the service object layer in the left part of the opened window in the "Layers" tab. After that, objects of this layer are displayed in the "Objects" tab in the right part of the window. You can use the search line in each of the tabs. Select an object in the list and click "Select" (Fig. 2.13).

Fig. 2.13: Service object selection window

The selected service object is displayed on the map in the task window, the task fields are filled in according to the configured mapping. You can edit filled fields and enter values into empty fields (mandatory fields are marked with an asterisk) (Fig. 2.14):

- "Title" – a brief description of the problem/goal of the task (mandatory field);

- "Contract" – the contract under which the work is carried out;

- "Organization" – the organization on whose behalf the task is created (mandatory field, this field is available to the System Administrator, System Inspector, Cluster Administrator, and Cluster Inspector);

- "Type of work" – type of work for the task (mandatory field);

- "Priority" – a characteristic of the urgency of the task (mandatory field, you can choose from the list: planned, unplanned, additional, etc.);

- "Assigned organization" – the organization to which the task is assigned for execution (when selecting a contract, the value specified in the contract is automatically filled in);

- "Executor" – the user responsible for performing the task;

- "Description" – a detailed description of the task;

- "Until" – the date and time by which the task should be completed;

- Custom fields – additional fields of different formats previously created and linked to a specific type of work.

Fig. 2.14: Task creation window with the selected service object and filled fields

In the "Files" tab you can attach files of different formats (Fig. 2.15). These can be photos, videos, audio recordings, or text files.
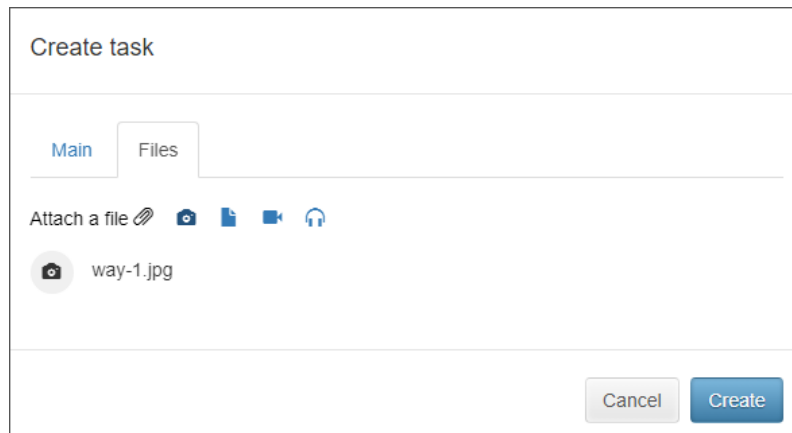
Fig. 2.15: Adding files to a new task

### 2.2.2.2 Editing a task

The ability to edit task fields depends on the role of the user. To edit a task, select the desired task and click "Edit" in the upper part of the window or double-click the task. The task opens in the edit mode. Here you can fill in/modify the available fields (Fig. 2.16).



Fig. 2.16: Task editing window

After making all the changes, click one of the buttons located at the top of the editing window:

- "OK" – save changes and go to the list of tasks;
- "Apply" – save changes without closing the editing window;

- "Cancel" – cancel all changes made before saving;

- "Close" – close the editing window (changes are not saved if they are not saved before closing).

If the task is not editable, only the "Close" button is active.

### 2.2.2.3 Deleting a task

To delete an existing task, select it and click "Delete" at the top of the window. Confirm the action in the pop-up information window (Fig. 2.17).
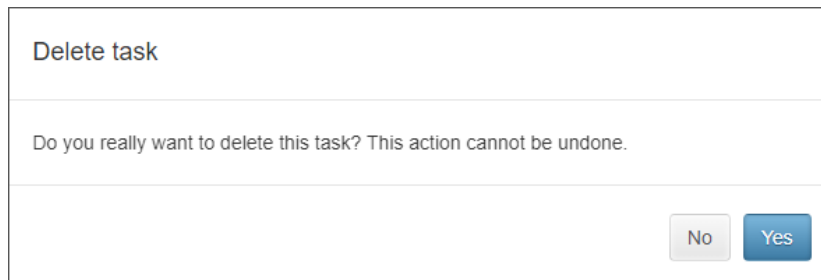


Fig. 2.17: Deleting a task

If the user does not have access to deleting tasks, a "No access rights" message appears.

## 2.2.3 Schedules

Schedules allow users to generate tasks automatically, based on templates (typical tasks), at a certain time with the required frequency. To go to the section with schedules and task templates, click "Schedule" on the top panel of the page. The schedule window with a calendar opens (Fig. 2.18).

Fig. 2.18: Schedule window with calendar

The window contains the following elements:

1. "Schedules" button – switching to the cards of all available schedules.

2. "Calendar" button – switching to the calendar with schedule launches (the calendar is opened by default when you go to the schedules section).

3. "Launches" button – switching to the cards of schedules with launches for the selected date.

4. "New schedule" button – switching to the schedule creation window.

5. Legend of the status of creating tasks with the ability to turn on/off.

6. Filter panel.

7. Calendar with marked task status.

When you click the "Schedules" button, you switch to the cards of all available schedules. The schedule card displays its name, organization, cluster, contract, launch (task creation) time, and a button taking to the list of templates attached to the schedule (Fig. 2.19).
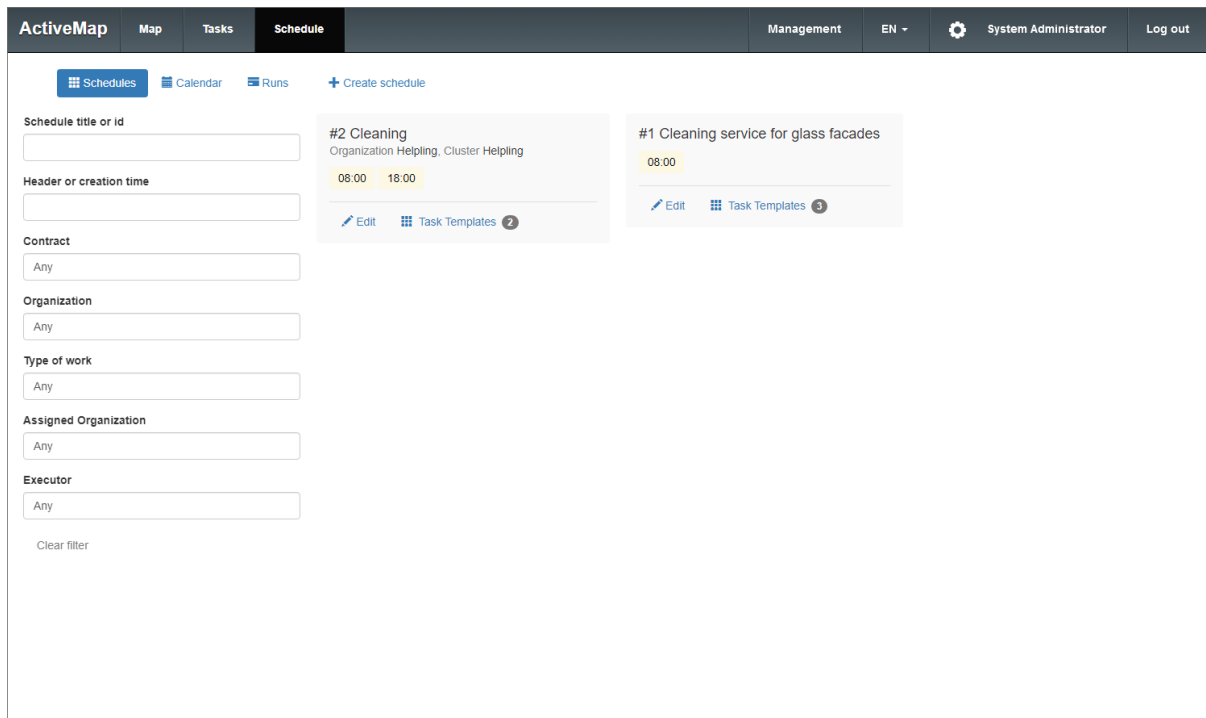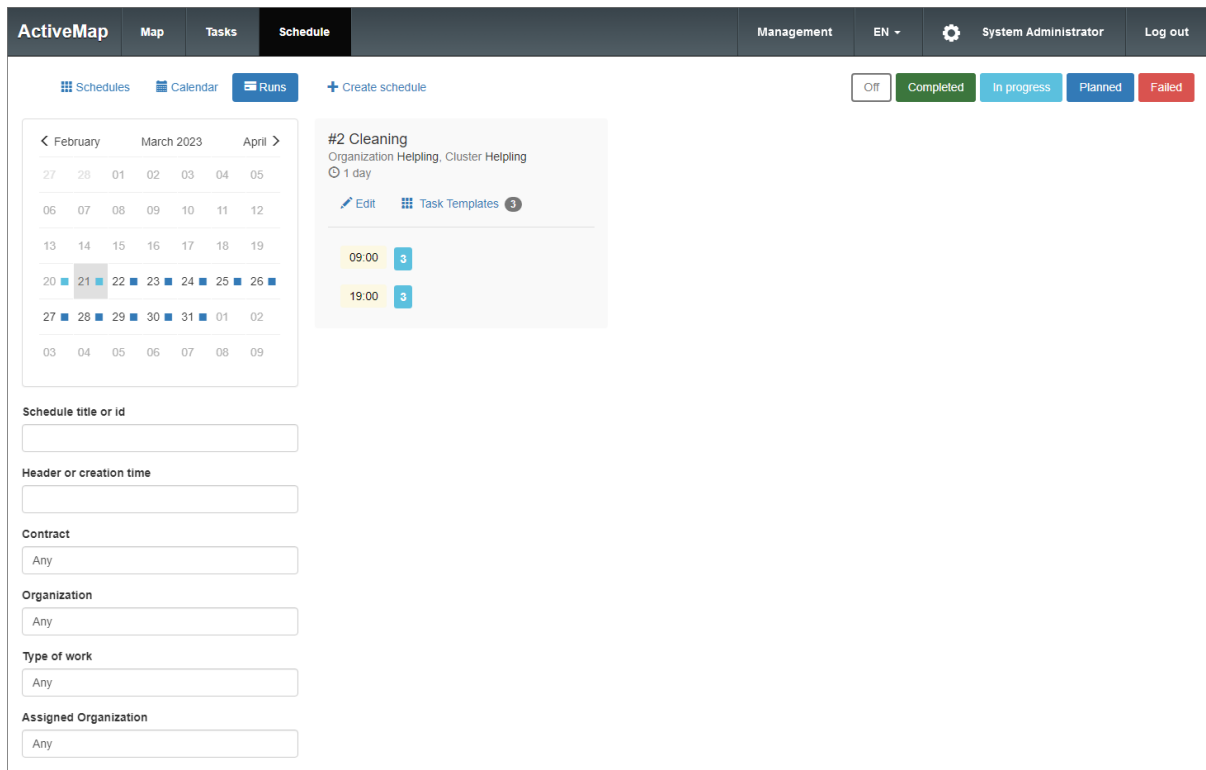
Fig. 2.19: Schedule cards

Clicking the "Calendar" button takes you to the calendar with schedule launches. The calendar with the current month is opened by default when switching to the schedules section (Fig. 2.18). The calendar cells display the number of tasks created by the schedule for the day. Tasks are grouped and color coded by status. The colors of the status are displayed in the legend in the upper right part of the window (enabled, completed, in progress, planned, not created). Click a status name to enable/disable displaying of the corresponding tasks in the calendar. Double-clicking a date in the calendar opens the launches window for that day.

You can go to the same window by clicking the "Launches" button. In the right part of the window, there is a smaller version of the calendar with task status marks (Fig. 2.20).

Fig. 2.20: Launches for the selected date

The schedule cards with launches for that day are placed to the right of the calendar. The schedule card displays its name, contract, a button taking to the list of templates attached to the schedule, the start time and the number of created tasks with status marking. Clicking on a start time in the card takes you to the task templates created at that time (Fig. 2.21). You can edit or delete these templates.

Fig. 2.21: Task templates

Creating and editing schedules and task templates are available for the following roles:

- System Administrator,

- Cluster Administrator,

- Organization Administrator.

### 2.2.3.1 Creating a schedule

To create a new schedule, click the "+ Create schedule" on the top panel of the window. In the opened form, enter the schedule name, task completion deadline, and the contract. Specify the launch time and select the checkbox to the right of the set time. Set the "Enabled" toggle switch, which is responsible for activating the schedule, to the desired position and check the dates in the calendar when tasks are created. Click "Create" to save (Fig. 2.22).

When selecting a contract in the task template, the "Assigned Organization" field is automatically filled in with the value specified in the contract. If the schedule was created under a contract that specifies service objects, you cannot create a task or template based on this contract without service objects.

Fig. 2.22: Schedule creation window

### 2.2.3.2 Schedule editing

To make changes, select the schedule card in the general list of schedules, in the calendar, or in the list of launches and click "Edit" (Fig. 2.23).



Fig. 2.23: Edit button in the schedule card

This opens the schedule editing window. Here you can change its name, due date, add a new start time, disable or delete the current start time (Fig. 2.24).



Fig. 2.24: Schedule editing window

You cannot delete the schedule from the system. To prevent the creation of scheduled tasks, turn off the "Enabled" toggle or remove the time for all launches and unmark the dates on the calendar through the edit form.

### 2.2.3.3 Creating a task template

Task templates are sample tasks that are created according to a schedule.

Clicking "Task Templates" in the schedule card opens a window with a list of templates attached to that schedule, if any (Fig. 2.25).

Fig. 2.25: Task templates

To create a task template, click the "New Template" button, fill in the form, and click "Create" (Fig. 2.26).



Fig. 2.26: Template creation window

#### 2.2.3.4 Editing a task template

To change the data of a template, click "Edit template" on the template card (Fig. 2.27).
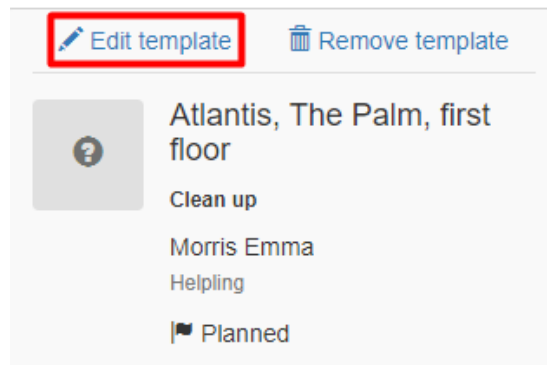


Fig. 2.27: Template editing

The task template card form opens where you can make changes and save them.

#### 2.2.3.5 Deleting a task template

To delete a task template, click "Remove template" on the task template card (Fig. 2.28).
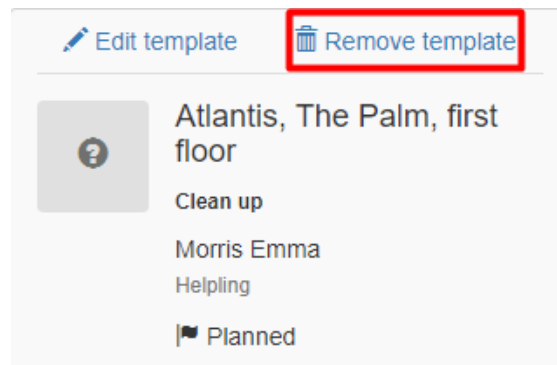


Fig. 2.28: Template deletion

#### 2.2.3.6 Restarting a schedule

If a task has not been created according to the schedule for some reason (for example, due to a mismatch of work type, organization, or executor), you can restart the template creation. To do this, click the "Runs" button, select a date and go to the schedule card. Make changes to the template and click "Restart" (Fig. 2.29). The ability to restart the schedule is determined by the role of the user.

However, changing the service object in this way is not possible (changes on the server are not saved). To change the service object, you have to create a new template.

Fig. 2.29: Restart from the schedule card

## 2.3 Management module

The management module is intended for configuring and managing data and system elements: layers, tasks, reports, users, organizations, etc. Only authorized users with certain access rights can work with it.

The main management tools include:

- exporting, loading, and indexing data;
- adding, deleting, and editing system elements (tasks, reports, users, and organizations);
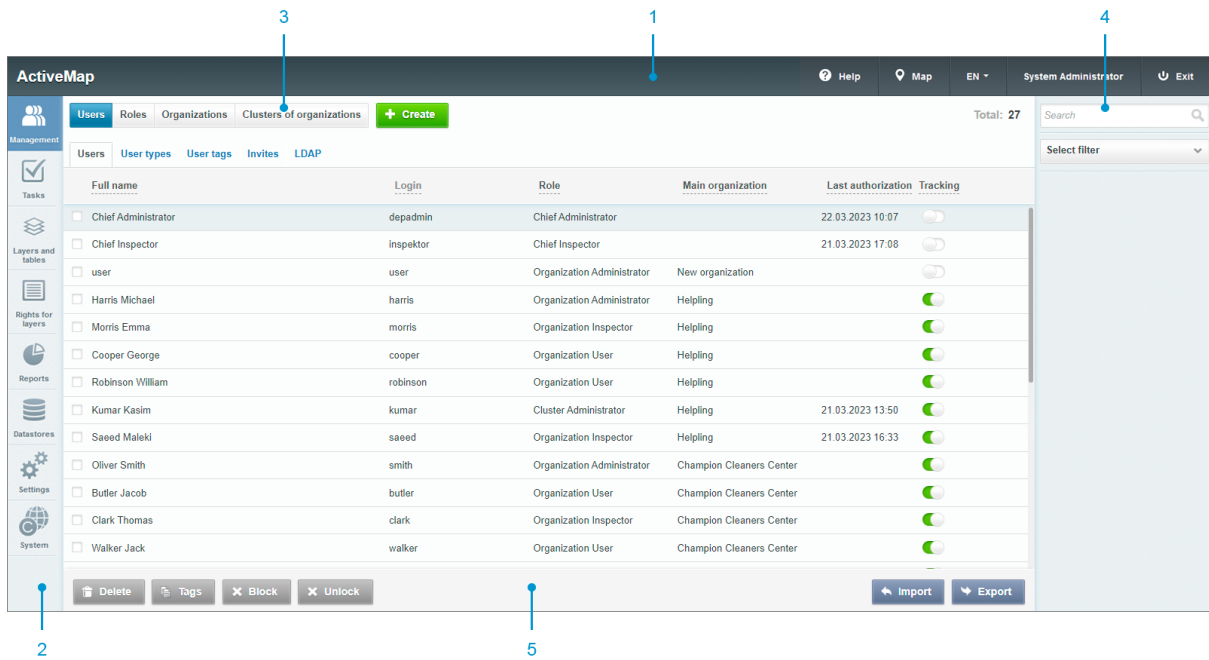- configuring access rights to the user information.

Fig. 2.30: Management module

The management module is divided into the following areas (Fig. 2.30):

1. User panel (contains buttons to access help and map, change program language, and exit the system).

2. Navigation panel of functional blocks (contains the main system management tools).

3. Tab bar (each tab contains information and a set of buttons to manage them).

4. Search string and filter (used to search for system elements).

5. Administration area (displays selected elements and their components).

### 2.3.1 Administration area

#### 2.3.1.1 Table sorting

All information about the main system elements (users, organizations, layers, etc.) is presented in the corresponding tabs in a tabular form. Using these tables, you can sort the available data.

To organize information about a particular element of the system, go to the tab with that element and click the title of the column by which the sorting should be done. When you click again on the column header, the sorting is done in reverse order.

### 2.3.1.2 Search string and filter

A search string [          🔍          ] is available on the right side of the page.

To search for an element, enter part of its name in the search string, after which all elements that correspond to the entered query are displayed in the administration area. The search is performed on all elements of the system.

In addition, you can use the filter located below the search string to select data according to specified search criteria. To get the most accurate search results, use the search string and filter at the same time (where both tools are available at once).

### 2.3.1.3 Adding a new element

To add a new element, go to the section of interest and click [ + Create ]. The appropriate rights are required to add new elements. When you click this button, a new window opens, where you have to fill in the required fields and click [ ✔ Save and exit ]. To cancel adding a new element, click [ ✖ Cancel ].

### 2.3.1.4 Editing an element

To change the data about a certain element, go to the section of interest. Then put the cursor on the element name and click the edit sign [✐] on the right side of the selected row. The edit sign [✐] appears only after hovering over the row with the element name.

Clicking it opens a form where you can enter new information about the selected element and click [ ✔ Save and exit ]. To cancel editing the element, click [ ✖ Cancel ]. In addition, you can open the editing form by double-clicking the row with the element name.

### 2.3.1.5 Deleting an element

To delete a specific element, go to the section of interest. Then hover over the row with the name of this element and click the delete sign [🗑] on the right side of the selected row. The delete sign [🗑] appears only after hovering over the row with the element name. Clicking it opens a dialog box confirming the deletion of the element. To delete the element from the system, click [ OK ]. To cancel the deletion, click the [ ✖ Cancel ] button.
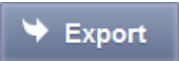
To delete multiple elements, place the mouse cursor on the rows with these elements and select the checkboxes next to their names. After that, the

[🗑 Delete] button becomes available in the lower part of the administration area. When you click it, a dialog box opens, confirming deleting the selected elements.

---

**Note:** When you try to delete your user or organization, the message appears in the administration area informing you that deletion is not possible.

---

### 2.3.1.6 Data export

The [➜ Export] button is required to export data as a separate Excel file (if you have the appropriate rights). If you use the search string and then click

[➜ Export], the downloaded Excel file contains information only about the found elements.

### 2.3.2 User panel

The user panel consists of the following elements (Fig. 2.31):

- "Help" – redirects from the main system page to the page with manuals.
- "Map" – navigates to the main system page.
- "Interface language" – switches the interface to one of the available languages.
- "User name" – opens a page for editing personal data.
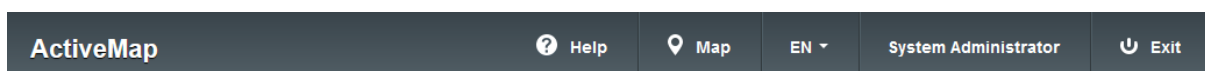- "Exit" – logs out of the user's account.



Fig. 2.31: User panel

### 2.3.3 Navigation panel of functional blocks

The navigation panel has the following functional blocks:

- "Management" – management of organizations, users, and their parameters (*"Management" block* (page 31));
- "Tasks" – management of task parameters (*"Tasks" block* (page 49));
- "Layers" – management of layers, layer groups, and their parameters (*"Layers and tables" block* (page 69));
- "Rights for layers" – management of rights for layers and groups of layers (*"Rights for layers" block* (page 100));

- "Reports" – viewing the list of reports and managing rights to reports (*"Reports" block* (page 104));

- "Datastores" – viewing and creating storages (*"Datastores" block* (page 109));

- "Settings" – viewing and creating system settings (*"Settings" block* (page 112));

- "System" – editing global and language settings of the system, activating licenses (*"System" block* (page 242)).

### 2.3.3.1 "Management" block

In the "Management" block, you can access detailed information about system users, roles, organizations, and clusters. When navigating to the tabs in the block, the following control elements become available: table sorting, search bar, filters, adding a new entry, editing an entry, deleting an entry, and exporting data.

#### 2.3.3.1.1 "Users" tab

The "Users" tab contains information about registered users in the system, roles, and organizations, and includes second level tabs (Fig. 2.32):

- "Users",

- "User types",

- "User tags",

- "Invites",

- "LDAP".

#### Users

Basic information about system users is presented in the form of a table with the following columns:

- "Full name" – full name of the user;

- "Login" – username used to log in to the system;

- "Role" – user's role in the system;

- "Main organization" – user's affiliation with the organization;

- "Last authorization" – user authorization time in the system applications;

- "Tracking" – management of the user's monitoring function (enabling the tracking of the user's movements when using ActiveMap Mobile).

Fig. 2.32: "Users" tab

If the license information for the number of users has been entered into the system, this tab also displays the information about the maximum number of users available according to the obtained license in the upper right corner. Deleted, blocked, and system users are not taken into account.

### Importing users

To upload new users to the system, click  in the lower right corner. A window opens with a choice of actions (Fig. 2.33):

- "Upload file" – import an Excel file containing information about users in the form of a table with the structure set in the template;

- "Download template" – export an Excel template with the structure required for correct import;

- "Cancel" – exit the import window.
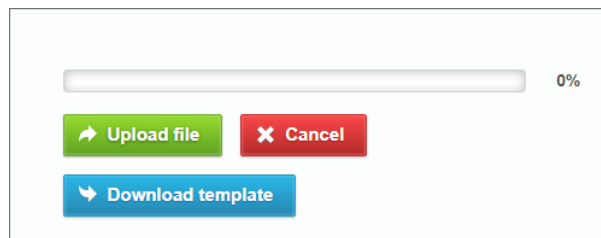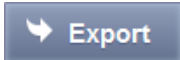


Fig. 2.33: Import users

If the file content is not suitable for import (required fields are not filled in, etc.), the system displays an error message. If you try to import

---

users that already exist in the system, you also receive an error message. After successful import, the new users specified in the file appear in the system.

To export information about users, click [Export] in the lower right corner. A window opens for selecting a directory to save the users.xlsx Excel file. This file contains information about users in the form of a table with the following fields: full name, login, role, user type, organization, additional organizations, tracking, user tags, e-mail, phone, address, and passport.

In addition, in the "Users" tab you can bulk add or remove tags to users. To do this, select users by ticking the appropriate checkboxes, then click [Tags] at the bottom of the screen. A window (Fig. 2.34) opens, where you can select tags to add/remove from the drop-down lists.
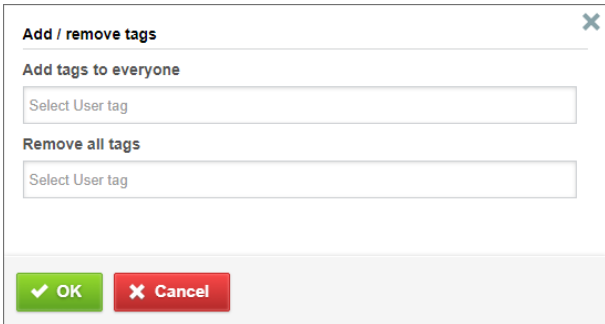


Fig. 2.34: Bulk adding and removing user tags

### Creating a new user

To add a new user to the system, click [Create]. A form (Fig. 2.35) opens in the administration area, where you have to:

- fill in the personal data of the user (name, login, password, user type, e-mail, phone number, address, and passport);

- define the user's role in the system (for more information about roles in the system, see *"Roles" tab* (page 42));

- enable/disable the geoposition monitoring;

- enable/disable the authorization via LDAP (single authorization system);

- enable/disable the integration with GIS editor;

- select the organization to which the user belongs (main and additional);

- add an avatar (file in *.jpg format);

- add user tags (for more information about user tags, see *User tags* (page 37)).



Fig. 2.35: User adding window

The "Full name" field is mandatory for all user roles. When creating a user under the System Administrator role, the "Main organization" field also becomes mandatory. Mandatory fields are marked with an asterisk.

If you leave the remaining fields empty, they are automatically filled with default values after clicking  (except for tags and personal data: email, phone numbers, address, and passport details). During the simplified creation of a user, an account with the "Executor" role, "Person" type is generated. The login is automatically filled in based on the entered full name. Monitoring is disabled by default. This user is granted access to layers that have the "default rights" setting activated, the layer cluster is not set or match the user's organization cluster. The main organization is automatically filled with the value of the creator's main organization (if the creator is not the System Administrator). Membership in main and additional organizations allows users of all roles to see the tasks and objects within the layers permitted to the specified organization. Organization administrators can manage

all the added organizations (both main and additional). Rights to layers are granted separately to each user of the organization.

When creating a user, you can leave the password field empty, and it is automatically generated. In this case, after creating the account, you should generate an invitation and send it to the user.

### Editing user information

To modify user information, click  or double-click the row with the name of the selected user. A form similar to the adding form opens in the administration area. Here you can fill in/modify the fields with the desired user information. The difference from the adding form is in the presence of the ❌ Block button. This button closes the user's access to the system.

### Deleting users

In the "Users" tab, you can delete one or several users at the same time.

To delete one user, click 🗑 on the right side of the user line. To delete several users at the same time, select the checkboxes near user names and click 🗑 Delete at the bottom of the screen. For more information about deleting system elements, see *Deleting an element* (page 29).

### Searching for users

You can use the search bar to find users by their name or login. To find a user by organization, cluster, role, type, tag, login, LDAP authentication status, and lock status, use the filter located below the search bar.

### User types

The tab contains a table with information about user types (Fig. 2.36).
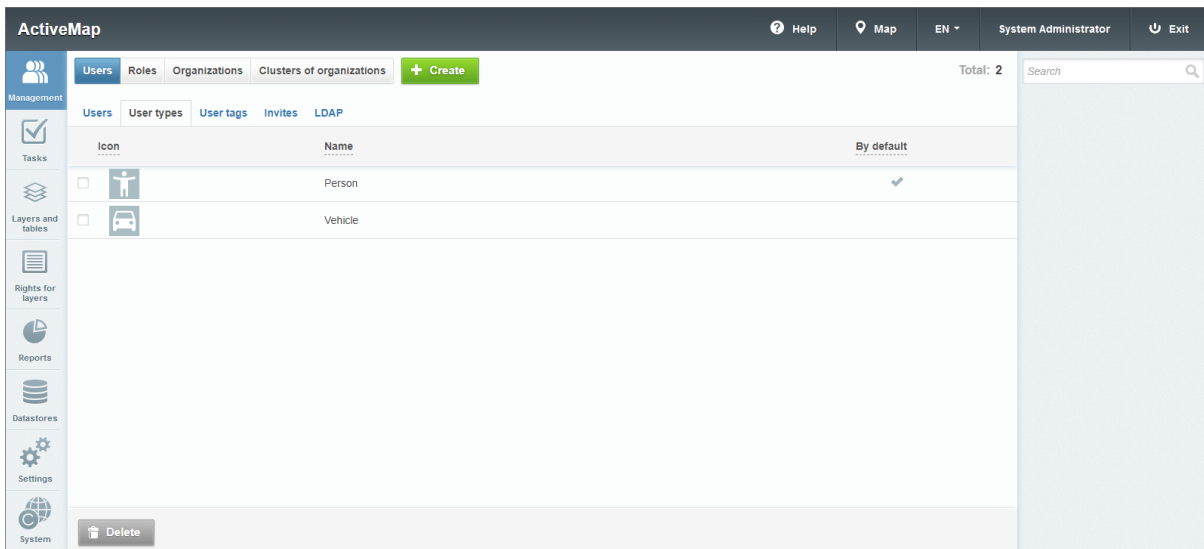
Fig. 2.36: "User Types" tab

There are two user types that are automatically available in the system: "Vehicle" and "Person". "Person" is always the default type and you cannot delete it. Any other user type cannot be the default.

You can also create other types. To do this, click ![Create] at the top of the screen. A window opens (Fig. 2.37), where you should set the type name, add an icon, and specify whether to display the avatar and initials of the user on the map.
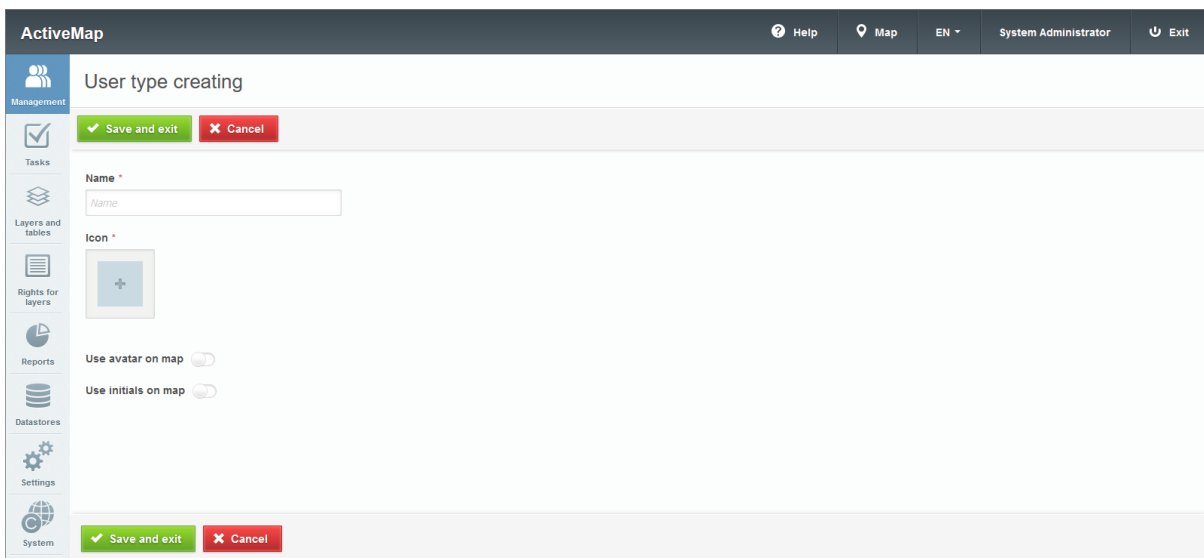


Fig. 2.37: Creating a user type

The name and the icon are mandatory. If both "Use avatar on map" and "Use initials on map" toggle switches are disabled, the icon is displayed on the map. If both toggle switches are enabled, then the avatar has priority (if any), then the initials, then the icon.

To edit the user type, hover over the type row and click . A window similar to the creation window opens, allowing you to enter or modify the information.

To delete one user type, click on the right side of the row. To delete multiple user types at once, check the corresponding rows and click

**Delete** at the bottom of the screen. See *Deleting an element* (page 29) for more information about deleting system elements.

### User tags

The tab contains a list of user tags with colors and names of tags (Fig. 2.38). Tags are used to display information about the user, in addition to the information provided by the system. For example, the user's phone model.
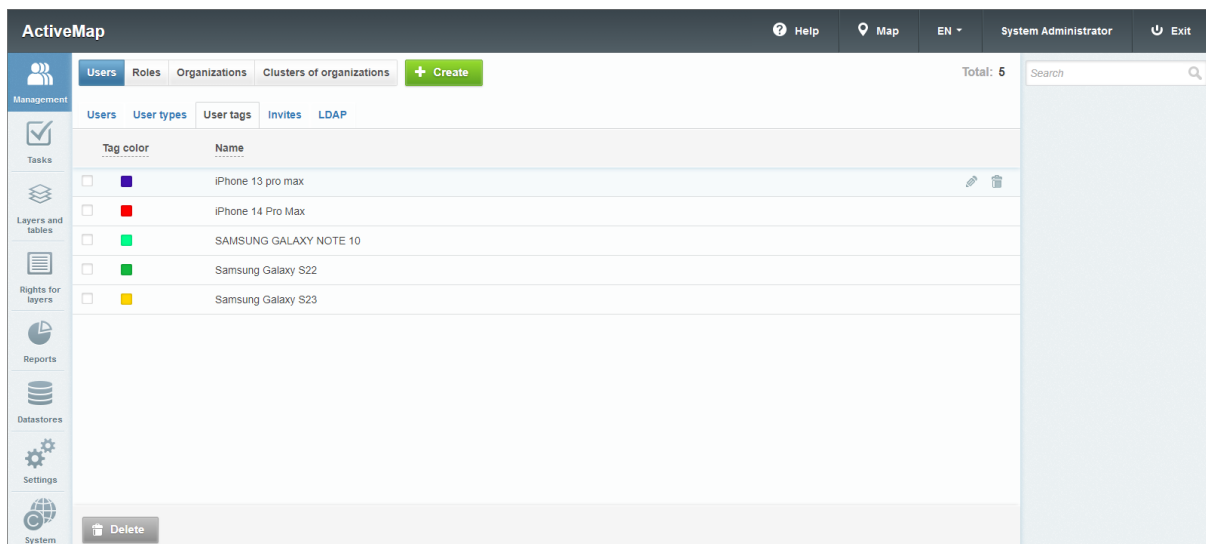


Fig. 2.38: "User tags" tab

To add a new tag, click **+ Create** and fill in the "Name" and "Tag color" fields. Both fields are required (Fig. 2.39).
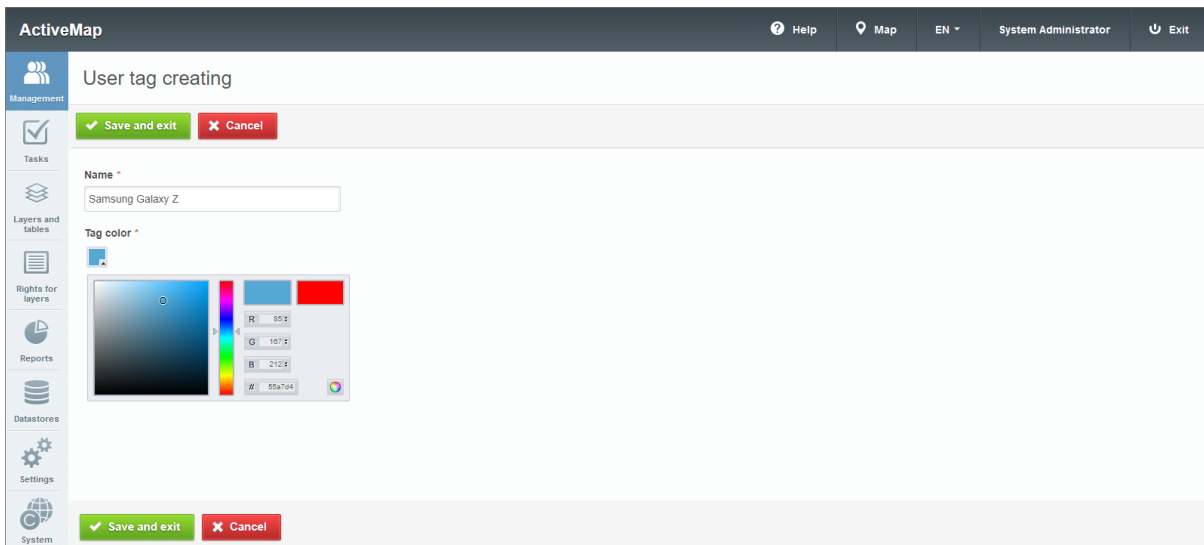
Fig. 2.39: Adding a user tag

After specifying the user's tag in the "Users" tab in the "User tags" field,
it appears in the user's card (Fig. 2.40)
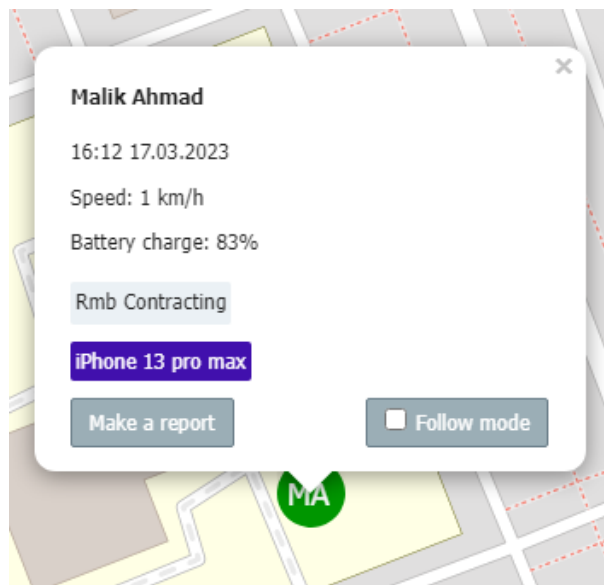


Fig. 2.40: Example of applying a user tag

To edit a user tag, click ![pencil icon] on the right side of the tag row. A window
similar to the creation window opens, allowing you to modify the tag
name and color.

To delete one tag, click ![trash icon] on the right side of the label row. To
delete multiple tags at once, select the corresponding rows and click

![Delete button] at the bottom of the screen. For more information about
deleting system elements, see *Deleting an element* (page 29).

### Invites

This tab is used to generate links that allow users to log in the ActiveMap Mobile application without entering the server address, login, and password. Information about created invitations is presented in a table. To sort the data, click the corresponding column header. To search for specific invitations, use the search bar or filter.

To create a new invitation, click ![Create]. A window opens (Fig. 2.41), where you have to fill in:

- Full name of the user to whom you want to send this link (you can select it from the drop-down list using the search string).

- Link expiration date (you can left it blank for an indefinite link).

- Maximum number of authentication attempts (after exceeding the specified number, the link becomes inactive).
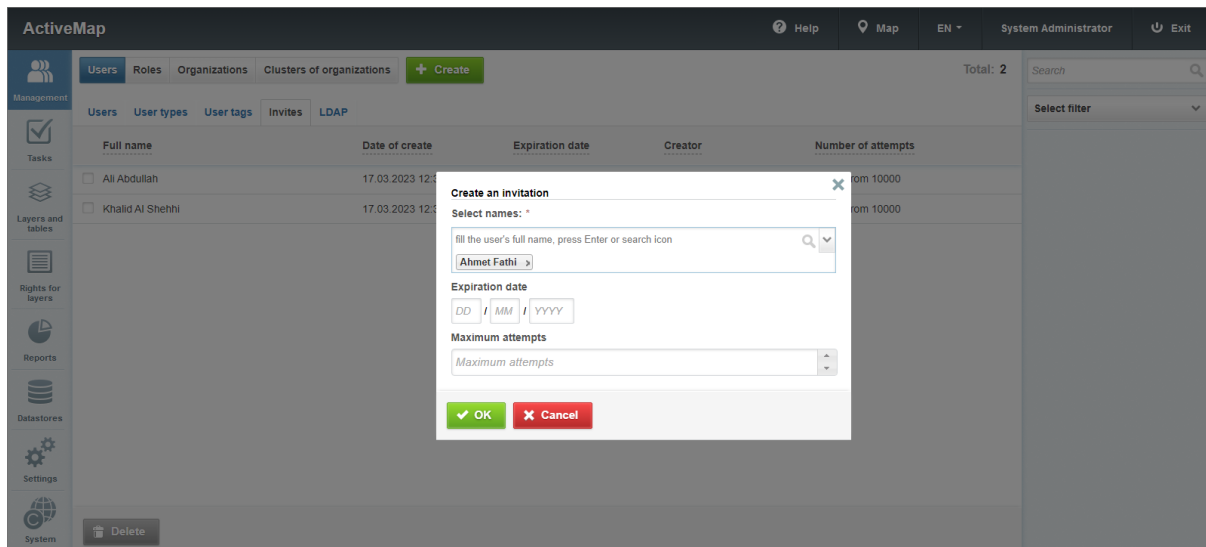


Fig. 2.41: Invitation creation window

Clicking ![pencil icon] shows you basic information about the generated invitation with an option to copy this link (Fig. 2.42).
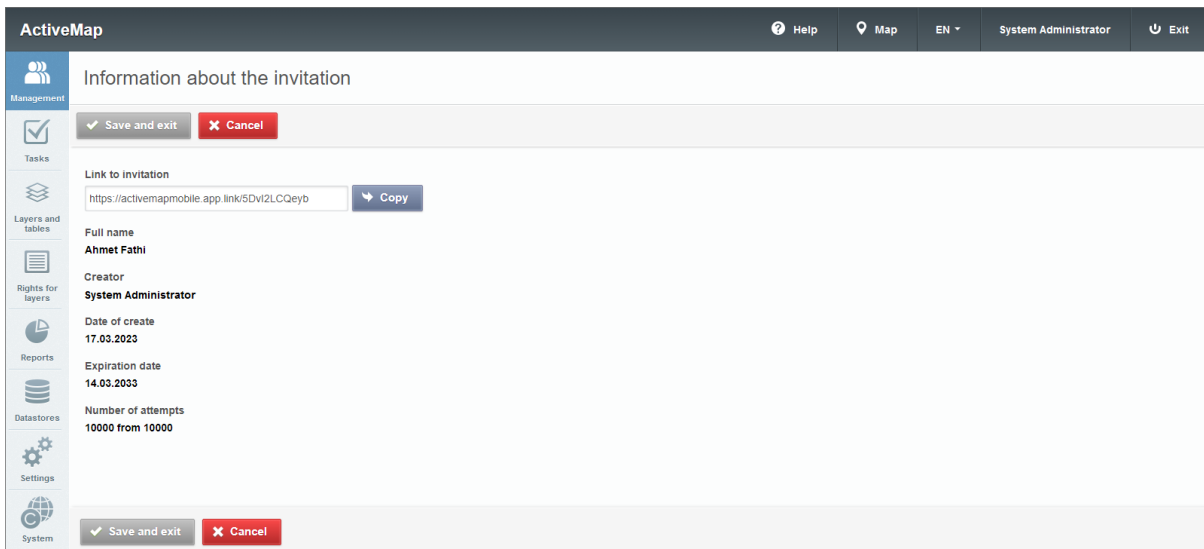
Fig. 2.42: Invitation information

Clicking  opens a window where you can also copy the link (Fig. 2.43).
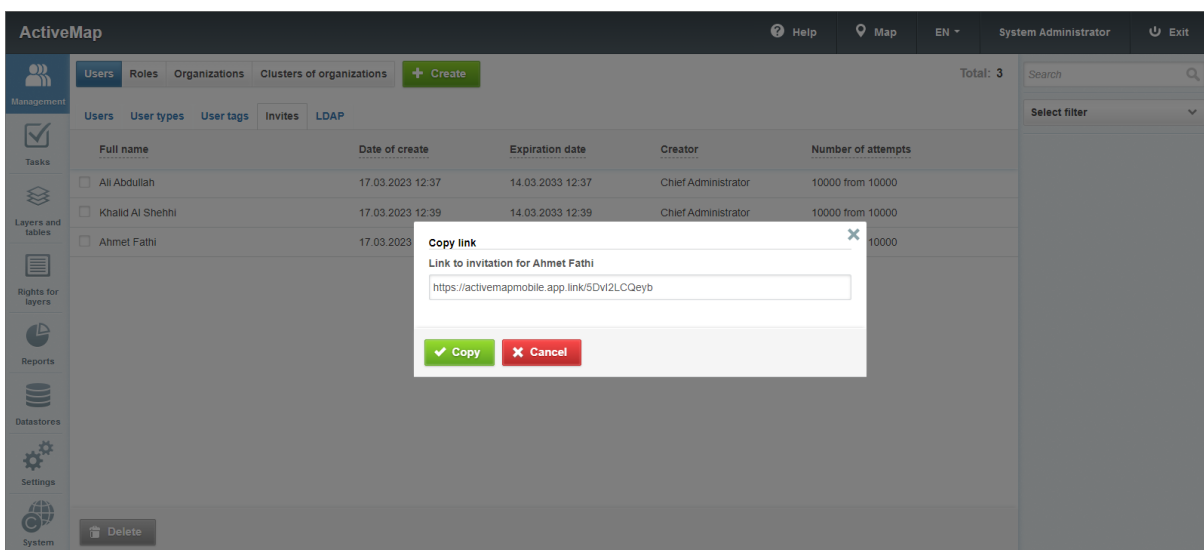


Fig. 2.43: Invitation link

You can send the link to the end user via familiar messenger.

Clicking  deletes the created invitation from the list. You can also delete an invitation by selecting the corresponding row and clicking the  button. If the invitation has been deleted and the user has not yet clicked on the link, you need to create a new invitation and re-send the link to the user. You can also create the invitation from a user's profile by clicking  and entering the necessary details in the opened window (Fig. 2.44). Copy and send the created invitation to

the end-user.
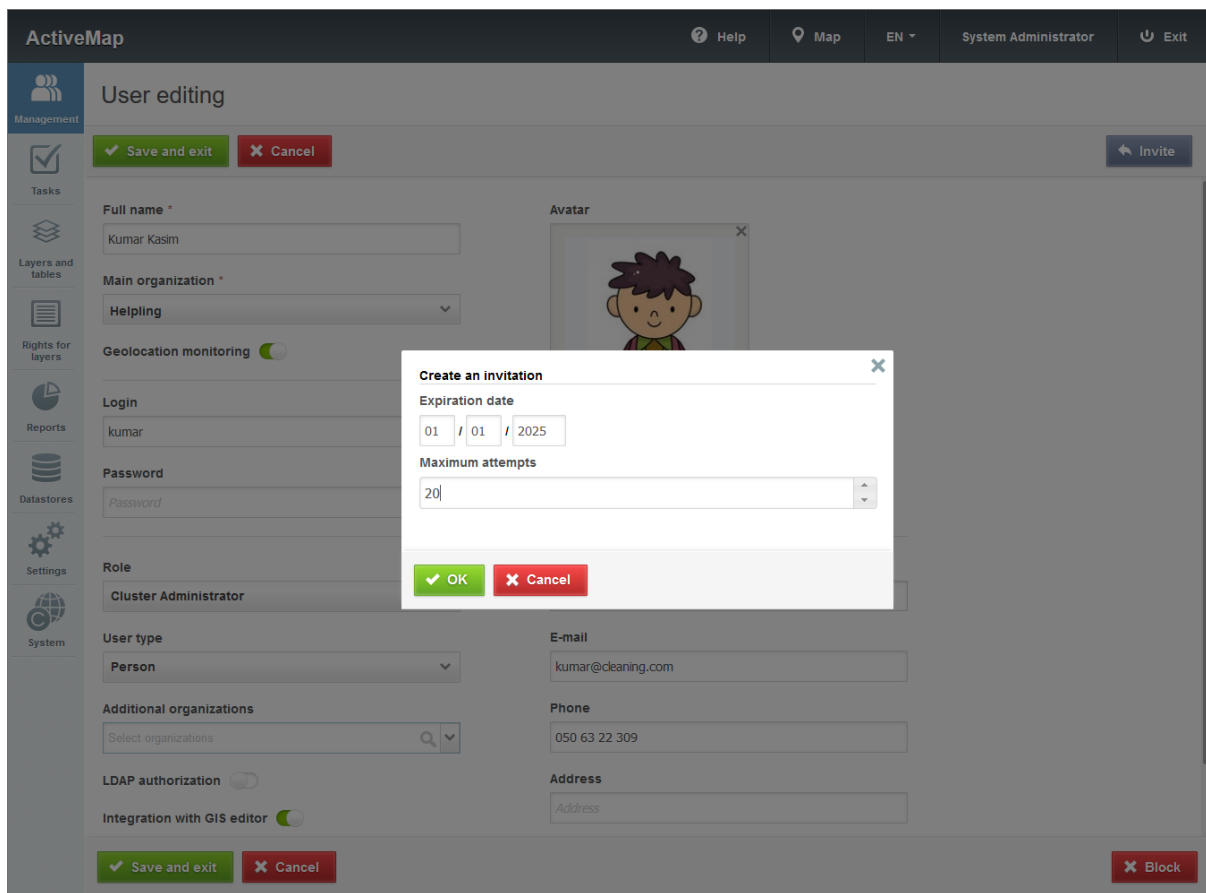


Fig. 2.44: Creating an invitation from a user profile

## LDAP

LDAP is an open protocol used to store and retrieve data from a hierarchically structured directory. It is typically used to store information about an organization, its assets, and users. It is a unified authorization system through which all software products work that are used in the organization. The ActiveMap system supports the LDAP protocol by enabling the "Authorize via LDAP" toggle switch in the user settings. The "LDAP" tab (Fig. 2.45) provides settings for integration of ActiveMap with LDAP. By default, the LDAP integration is disabled. If the integration with LDAP is required, fill the configuration fields with data provided by the organization.

Fig. 2.45: Default LDAP configuration

### 2.3.3.1.2 "Roles" tab

There are several user roles in the ActiveMap system. They are assigned by administrators when creating user accounts. Roles differ from each other in the set of actions that users can perform:

- The **"System Administrator"** is responsible for the system configuration, including the management of clusters, organizations, users of all roles, contracts, directories, and for the distribution of access rights to the different layers and reports.

- The **"System Inspector"** manages the tasks of all clusters.

- The **"Cluster Administrator"** is responsible for cluster administration, namely: managing organizations and users of his or her cluster, assigning access rights to layers and reports within the cluster, and for managing cluster tasks.

- The **"Cluster Inspector"** manages the tasks of the cluster.

- The **"Organization Administrator"** is responsible for administering the organization, namely: creating users, granting access rights to layers and reports within the organization, and managing tasks of the organization.

- The **"Organization Inspector"** manages the tasks of the organization.

- The **"Executor"** creates new tasks and executes the assigned tasks in the System.

The "Roles" tab displays a list of roles in the system (Fig. 2.46). To rename a role, hover over the role row and click  . After making changes, click the confirmation button  . Click the  button if you do not want to save the changes. To change the order in which roles are displayed, drag the role row up or down and click  .
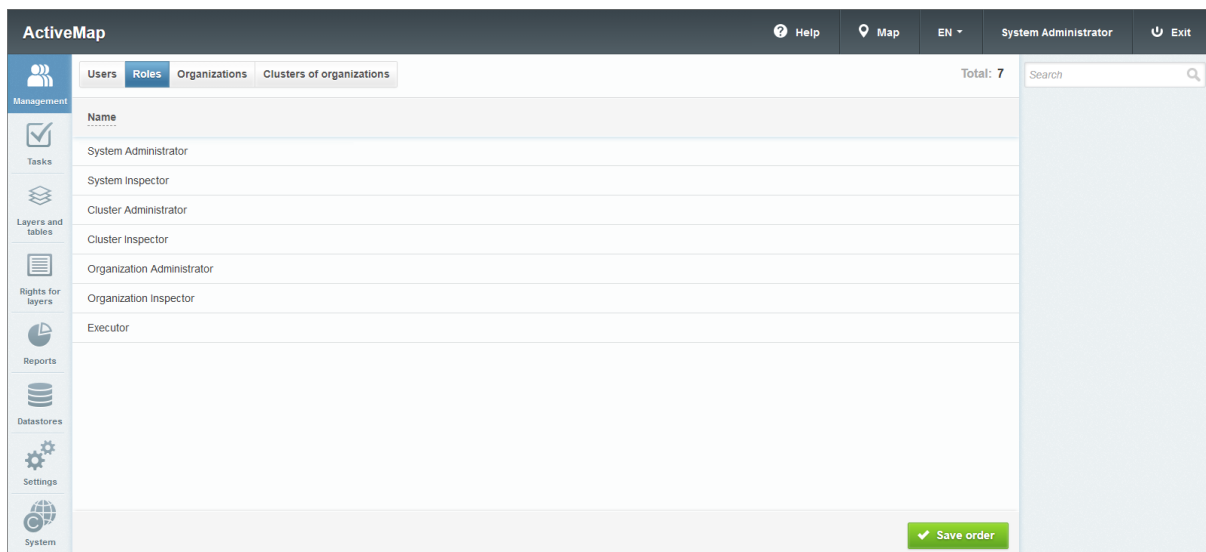


Fig. 2.46: "Roles" tab

### 2.3.3.1.3 "Organizations" tab

When switching to the "Organizations" tab, you can view a list of all existing organizations in the system and their parameters (Fig. 2.47):

- "Name" – name of the organization (mandatory field);

- "Cluster of the organization" – an association of several organizations for operational monitoring of the work of departments. For more information about the clusters, see *"Clusters of organizations" tab* (page 47).

- "Client organization" – an indication of whether the organization is a client. Client organization is an association of users who send their requests through a mobile application, monitor the status of their execution, have the ability to rate the work done, but have limited rights when working in the system.
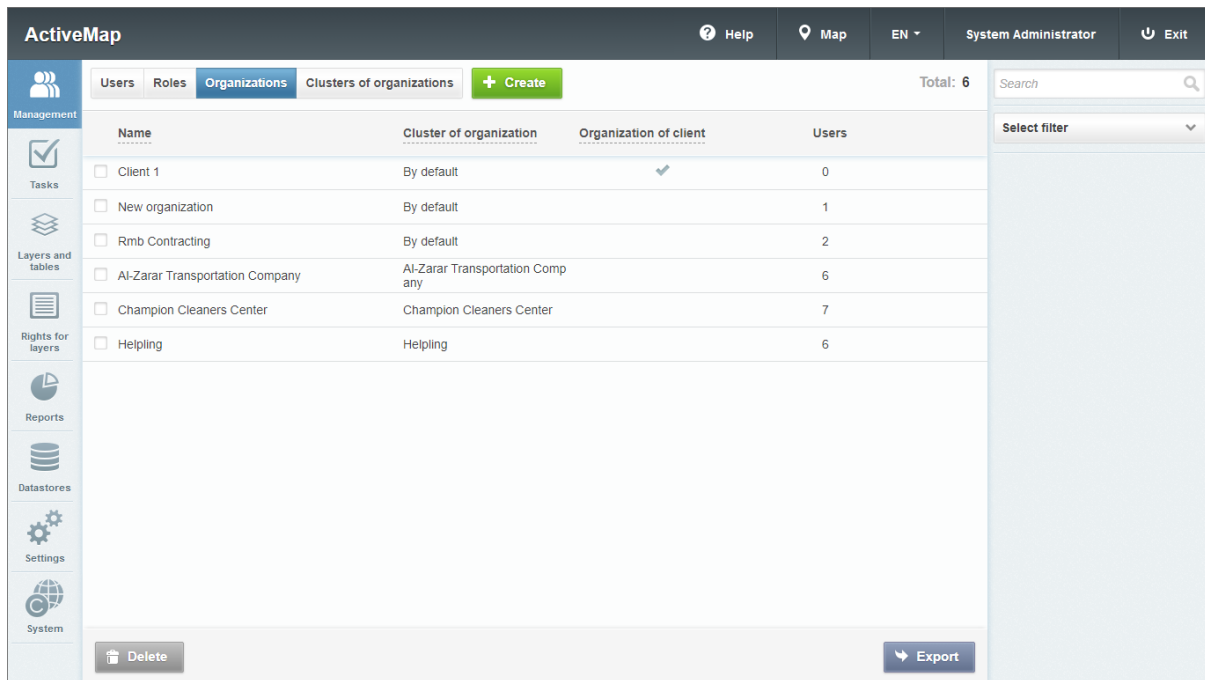
- "Users" – number of users in the organization.

Fig. 2.47: "Organizations" tab

### Adding a new organization

To add a new organization to the system, click **+ Create** at the top of the window. A window with fields to fill in ([Fig. 2.48](#)) opens in the administration area:

- "Name";

- "Cluster of organization";

- "Client Organization" toggle switch;

- "Default implementing organization" – automatically assigned organization (it must belong to the cluster of the creator's organization);

- "Integration with GIS editor by default" (available only for the administrative roles);

  If the System administrator enables this feature for the organization, then all its users created by the Cluster Administrator have integration with the GIS editor by default. If the System administrator adds users, he/she must enable this setting separately for each employee. When activating the integration, the user should enter the password to work in the desktop editor. If integration is not enabled, users are not created in the database and, accordingly, unable to work in the desktop editor. If integration is enabled for the organization, then all users are created for MapEditor and granted rights to layers and tables. If integration is disabled while editing the organization card, all users are removed from the database.

---

- "Maximum number of users";

- "E-mail";

- "Phone";

- "Legal name";

- Organization data (address, fax, TIN, VAT-ID, bank account, bank, correspondent account, SWIFT, BIC, name of the head, and name of the accountant);

- "About" (a field to enter information in any form);

- Logo, stamp, signature of the head, and signature of the accountant.

To add a logo, stamp, and signatures, click "+" and select the photo.



Fig. 2.48: Creating organization

To the right of the input fields there is a "Visible area" – the boundaries of the map visible to organization users when they log in. The administrator can use the scale bar to zoom in or out of the map, as well as move the map by pressing the left mouse button. After selecting the necessary view, click "Apply" to save the new map boundaries.
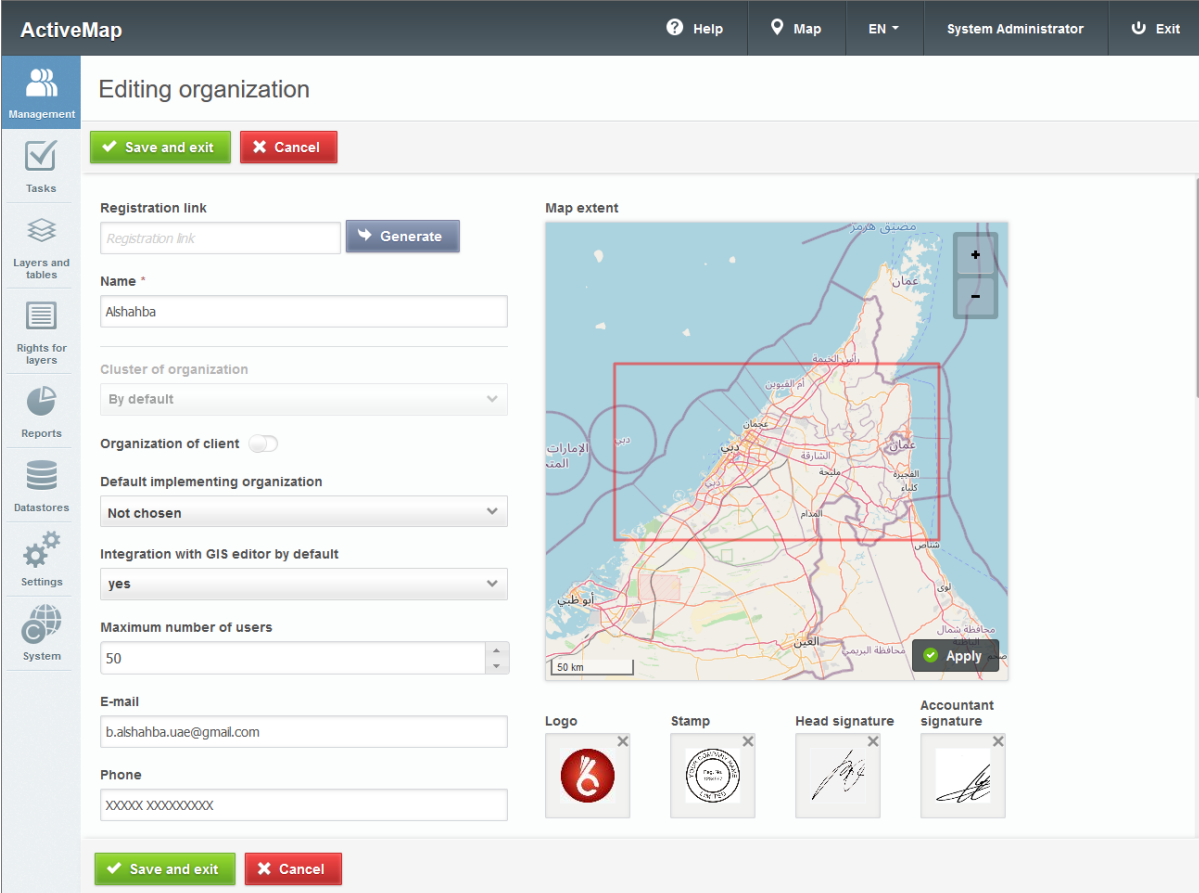
To add a new organization to the system, fill in its name and click


. If you leave the remaining fields empty, they

are automatically filled with default values (except for e-mail, phone number, legal name, and other organization data). You can manually change the automatically assigned data storage, marked as default in the system, in the database. The visible area and the cluster match the main organization of the Cluster Administrator, or the default value if the organization is created by the System Administrator. This organization is automatically granted access to work types (global and/or local) with activated "For all organizations" option.

### Editing an organization

Clicking [pencil icon] (or double-clicking the selected organization row) opens a form where you can fill in/change the fields with the information about the organization, get a link to register users in the mobile application ActiveMap Mobile or copy/delete it if it was generated earlier (Fig. 2.49). If you delete a link when editing an organization, it still works.



Fig. 2.49: Editing an organization

After receiving a link, the employee opens it on a mobile device and registers by the phone number. Thus, an account with the "Executor" role and "Person" type is created in the system, the login is automatically generated based on the entered phone number, monitoring is disabled by default. This user is granted access to the layers for which the

"default rights" setting is activated, the layer cluster is not specified, or matching the user's organization cluster.

**Deleting an organization**

To delete an organization, click <img> on the right side of the row. To delete several organizations at once, select the corresponding rows and click **🗑 Delete** at the bottom of the screen. For more information about deleting system items, see *Deleting an element* (page 29).

When you try to delete your own organization, a dialog box appears, informing that such deletion is not possible.

**Searching for an organization**

You can use the search bar to search by name in the "Organizations" tab.

**2.3.3.1.4 "Clusters of organizations" tab**

"Clusters of organizations" tab contains information about the grouping of organizations into clusters (Fig. 2.50). Before being set by the administrator, all organizations are located in the "Default" cluster.
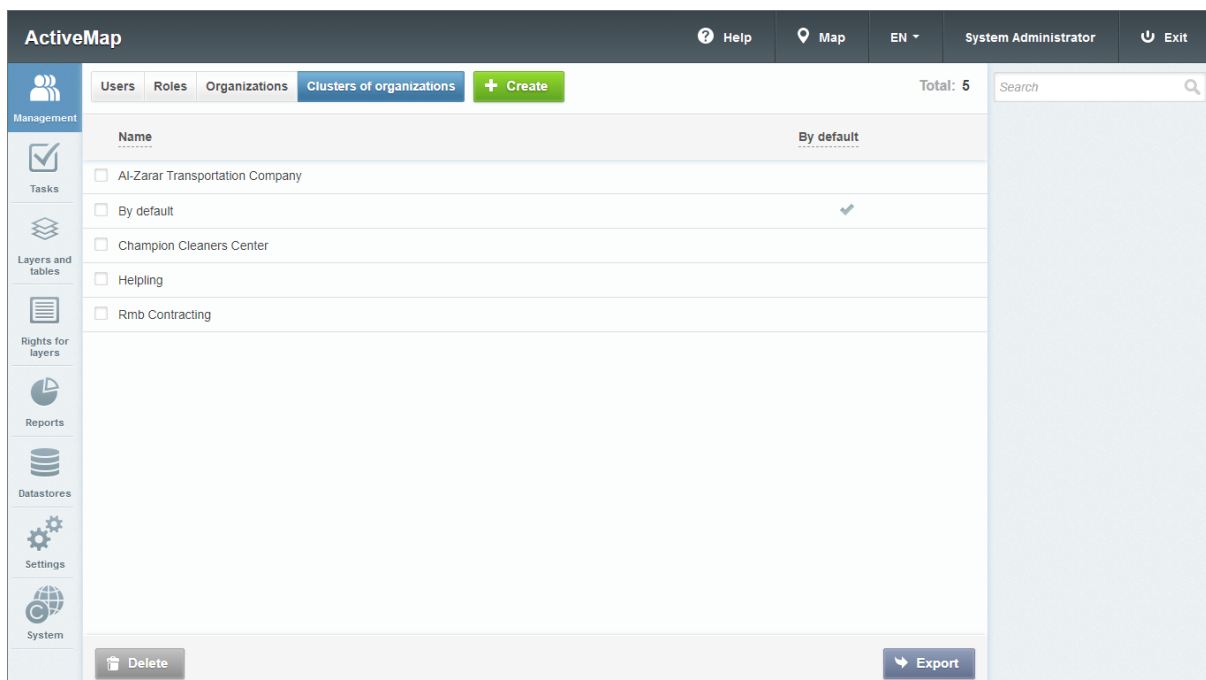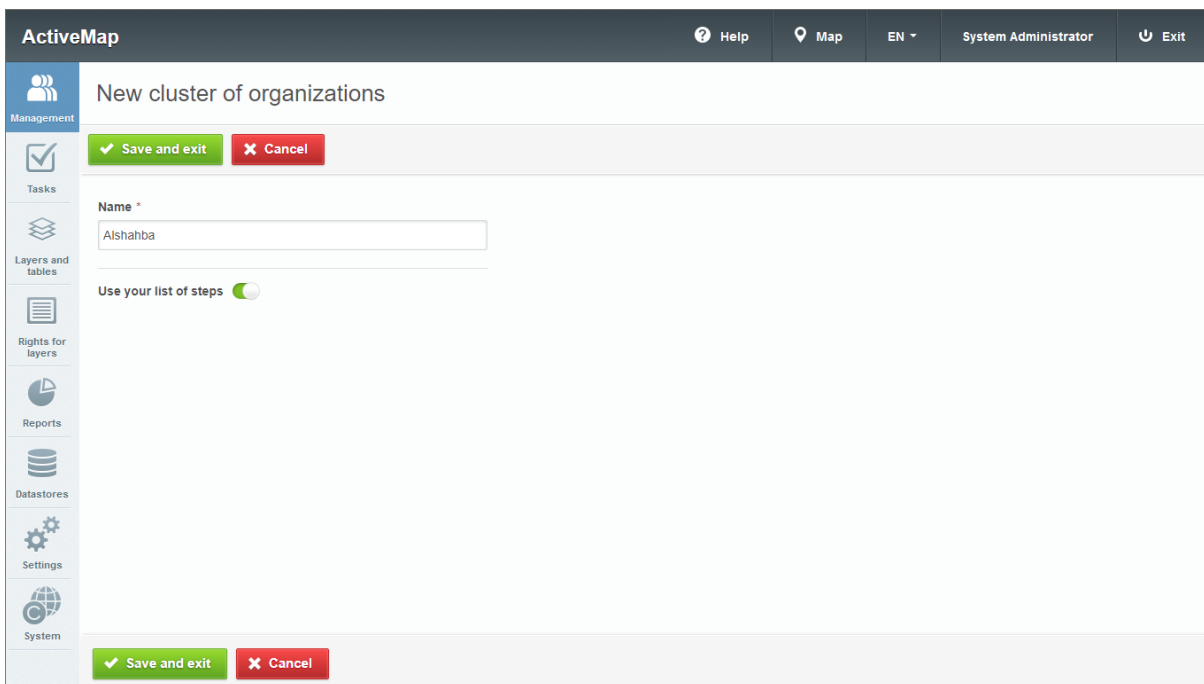


Fig. 2.50: "Clusters of organizations" tab

### Adding a new organization cluster

To add a new cluster of organization, click [**+ Create**]. A window opens in the administration area where you have to fill in the name of the cluster (Fig. 2.51). You can activate "Use your list of steps" setting to make the cluster isolated and create separate steps for it. The setting can only be activated when creating a cluster. You cannot add or disable it later. An isolated cluster does not have access to global steps, so it is necessary to create its own steps, otherwise tasks cannot be created. For more information about creating steps, see *"Steps" tab* (page 53).

After creating a cluster, it becomes available for selection in the organization creation/editing form.



Fig. 2.51: Adding an organization cluster

### Editing an organization cluster

In the "Clusters of organizations" tab, you can edit information about

the organizations of the system. Clicking  (or double-clicking on the cluster) opens the form of changing the name of the cluster of organizations with the list of organizations in the cluster (Fig. 2.52).
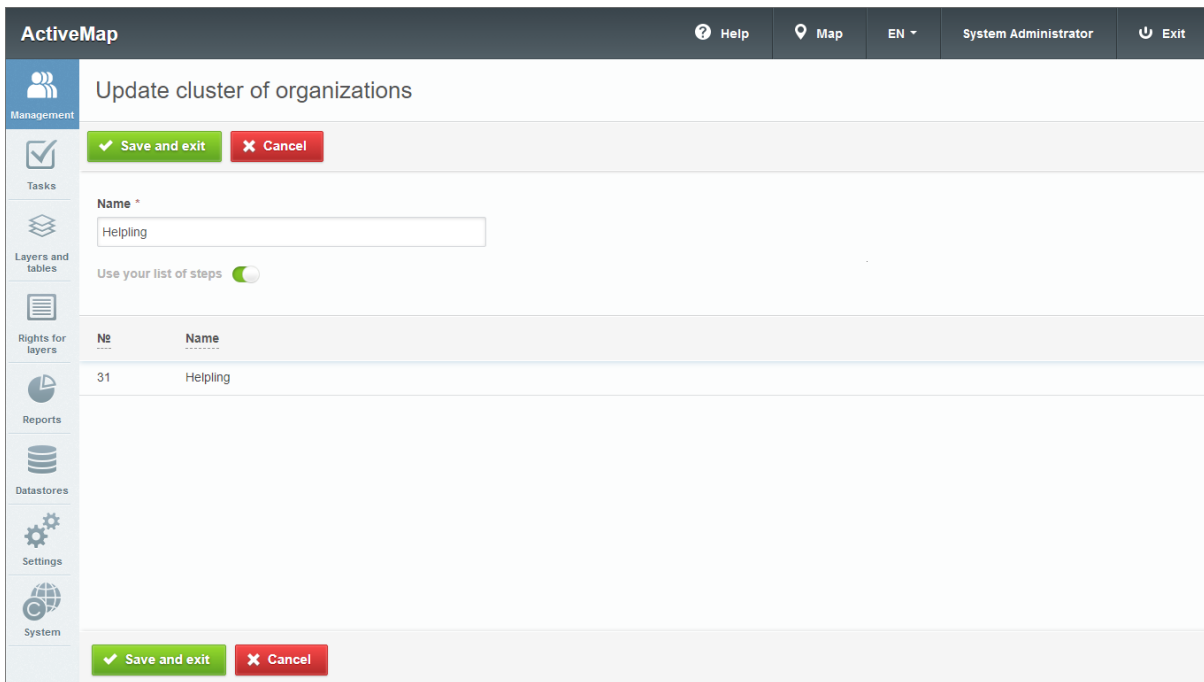
Fig. 2.52: Editing an organization cluster

### Deleting an organization cluster

To delete one cluster of organizations, click  on the right side of the row. To delete several organization clusters at once, check the corresponding rows and click the  button at the bottom of the screen. For more information on deleting system items, see *Deleting an element* (page 29).

When trying to delete the "Default" organization cluster, a dialog box appears, informing that such deletion is impossible.

### 2.3.3.2 "Tasks" block

You can manage task parameters in the "Tasks" block (Fig. 2.53).

Fig. 2.53: "Tasks" block

The following sections are located at the top of the window for management:

- "Types",
- "Steps",
- "Priorities",
- "Custom Fields",
- "File Labels",
- "Roles and Rights",
- "Plugins and Webhooks".

Types of work, their groups, steps, priorities, and plugins can be local (associated with a specific cluster) or global (without association). Only the System Administrator can manage global entities. The System Administrator and Cluster Administrator can manage local entities. Local entities are not visible to users outside the specified cluster.

By default, when you go to the "Tasks" block, the "Types" tab opens.

### 2.3.3.2.1 "Types" tab

There are two subsections in this tab: "Types" and "Work type groups".

Users outside the specified cluster cannot see local types of work. You can associate local types only with organizations within their cluster. If the "For all organizations" toggle is turned on, then they are available to all organizations in that cluster. You can include local types of work only in local groups.

You can associate global type of works with any organization. If the "For all organizations" toggle is turned on, then the types of work are available to all organizations in all clusters of the system. You can include global types of work only in global groups.

Only the System Administrator can manage global entities. The System Administrator and Cluster Administrator can manage local entities. The Cluster Administrator can edit a global types of work only when associating or disassociating organizations within the cluster (when the "for all" mode is not enabled).

### "Types" subsection

The "Types" subsection contains a list of existing types of work (Fig. 2.53). To add a new type of work, click "Create". A window opens with fields to fill in (Fig. 2.54):

"Main" tab:

- "Name" – name of the type of work;

- "Cluster" – organization cluster to which this type of work is linked (if the cluster is not specified, the type of work is available for all organizations);

- "Type group" – group of works to which this type belongs;

- "Deadline" – time required to complete the task (defined by the task description for this type of work);

- "Moment of setting the deadline" – time from which the deadline is counted: "when creating a task" – at the time the task is created in the system, or "when assigning a task" – when the task is assigned to a specific executor;

- "Icon" – task icon, which is displayed in the task list in the ActiveMap Mobile and ActiveMap Desktop applications.

"Organization binding" tab:

- "For all organizations" toggle switch – makes the type of work available for all organizations in the system (for global types of work) or within the cluster (for local types of work);

- toggle switches with the names of individual organizations – makes the type of work available for the selected organizations.

Fig. 2.54: Creating a type of work

To edit a type of work, select the row with the required type and click "Edit". A window similar to creating types of work opens, where you can change the field values.

To delete an existing type of work, select the required type and click "Delete". Confirm the deletion by clicking "Yes" or cancel it by clicking "No" in the dialog window.

### "Work type groups" subsection

The "Work type groups" subsection contains a list of existing groups (Fig. 2.55). Work type groups allow you to visually group objects in the ActiveMap Mobile and ActiveMap Desktop applications.

Fig. 2.55: Work type groups

To add a new work type group, click "Create". A window opens asking you to enter a name for the group. After that, the group becomes available in the corresponding field when creating the work type.

To edit a group of work types, select the group in the list and click "Edit". A window similar to the creation window opens, where you can change the name of the group.

To delete a group of work types, select it in the list and click "Delete".

### 2.3.3.2.2 "Steps" tab

This tab displays a list of existing work steps (Fig. 2.56).

Fig. 2.56: "Steps" tab

Local (cluster-specific) and global (unrestricted) steps of work are distinguished. Steps can be sorted within the group. The System Administrator can sort any group. The Cluster Administrator can sort only local steps in the cluster.

To add a new step, click "Create". A window opens, where you have to fill in the following fields (Fig. 2.57):

- "Name" – name of the work step;

- "Cluster" – belonging to a cluster of organizations (if no cluster is specified, the step is available for all organizations);

- "Color" – color of step displaying in the list of tasks (specified as RGB code);

- "Visible" toggle switch – visibility of the step being created;

- "Default Step" toggle switch – automatically setting at task creation;

- "Step of a closed task" toggle switch – passing to this step means that the task is closed.

Fig. 2.57: Creating a step

To edit an existing step, select it and click "Edit". A new window opens where you can edit the values in the fields mentioned above.

You can set the order of displaying steps in the system. To do this, select the appropriate cluster (if you plan to change the order within an isolated cluster) and click "Change Order". Drag the step to a new position in the list and save the changes.

To delete a step, select it and click "Delete". Confirm the deletion by clicking "Yes" or cancel it by clicking "No" in the dialog box.

Only the System Administrator can create, modify, and delete a global step. The System Administrator and Administrator of the corresponding cluster can create a local step. Administrators can only create a local step within an isolated cluster (when "Use your list of steps" setting is activated). To create a local step, select the corresponding cluster from the drop-down list. A list of local steps of the selected cluster is displayed. When creating a new step, name of the selected cluster is filled in automatically. Then you have to enter a new value and save the changes. The created step is available only within the selected cluster.

If the cluster does not have steps and the first one is created, it is automatically assigned the "default" value. You cannot delete the "default" step. If another step is marked as "default", the mark in the checkbox of the previous step is automatically removed. The first non-"default", non-closing step is considered as the "assigned" step. If there is no such step, the step does not change when you change the executor. If a task is created or edited within an isolated cluster, you can use only the local steps of that cluster. You can use only global steps within a regular cluster. Global steps are not available to an isolated cluster. When creating such a cluster, it is mandatory to create the steps for it, otherwise tasks cannot be created.

### 2.3.3.2.3 "Priorities" tab

This tab displays a list of existing work priorities (Fig. 2.58). Priorities allow you to group the types of work by importance.



Fig. 2.58: "Priorities" tab

To add a new priority, click "Create". Specify the name and, if necessary, the cluster of the new priority. Local priority is available only within the selected cluster. Id and serial number are assigned automatically (Fig. 2.59). You can also upload a custom icon or leave the default one.



Fig. 2.59: Creating a priority

To edit an existing priority, select it and click "Edit". This opens a window where you can change the name of the selected priority. To delete an existing priority, select it and click "Delete". A dialog box opens, where you should confirm the deletion by clicking "Yes" or cancel it by clicking "No".

#### 2.3.3.2.4 "Custom Fields" tab

This tab displays a list of existing custom fields (Fig. 2.60). Use custom fields to add user-defined fields to the task creation form. You can attach such fields to a certain type of work. For example, for types of work that involve the interaction with a client by a field specialist, you can create a field in the format of "Phone number" to enter the client's phone number.



Fig. 2.60: "Custom Fields" tab

Custom fields can be local (with binding to a particular cluster) and global (without binding). Only the System Administrator can manage global entities. The System Administrator and Cluster Administrator can manage local entities. There is an exception – the Cluster Administrator can edit a global custom field (if "for all" mode is not enabled), while only attaching or detaching local types of work of this cluster.

You can set the order of displaying custom fields in the system. Fields are sorted only within the group. The System Administrator can sort any groups. The Cluster Administrator can sort only local fields within the cluster. To do this, select a group of fields and click "Change Order" (Fig. 2.61).

Fig. 2.61: Selecting a group of custom fields to reorder

Use [arrow] and [arrow] to set the order of the custom fields in the task and click "Apply" (Fig. 2.62). The fields are sorted by the "Order" field within the group. If the task has both global and local fields, the global fields are displayed first, and then the local fields in the order specified in the system.



Fig. 2.62: Changing the order of displaying custom fields within a group

To add a new custom field, click "Create". A window opens with fields to fill in ([Fig. 2.63](#)):

"Main" tab:

- "Name" – name of the field.

- "Format" – data format of the field, one of the following values should be selected:

    - String – a short text;

    - Text – an extended text;

    - Integer number – an integer;

    - Real number – a real numeric value;

    - Date – date and time;

    - Logical value – a choice from true and false options;

    - Selection from the list – a format with the possibility of specifying a list of options;

    - Phone number – a format with the possibility of calling a specified number from the task window;

    - Barcode – a numeric decoding of barcode;

    - Geometry – a format that contains information about the type of geometry (point, line, polygon) and coordinates of one or several objects;

    - Data Objects – links to objects of layers, data tables or reference table (dictionaries).

- "Mandatory custom field" – toggle switch for mandatory filling of the created field.

- "Visible custom field" – toggle switch for visibility of the created field for users.

- "Default value" – automatically filled field value.

- "Fields group" – selection of a group of custom fields.

The list of fields may vary depending on the selected format. For fields with the "String", "Text", "Integer number", or "Real number" type you can additionally enter the "Regular expression". It is a pattern string that sets a template for a custom field. You should specify the minimum and maximum length of the field for text-format fields (string and text), possible values for the "Selection from the list".

"Binding to type of work" tab:

- "For all types of work" toggle switch – makes the field available for all types of work in the system (global custom fields) or within a cluster (local custom fields);

- toggle switches with names of individual organizations – makes the field available for selected types of work.

Fig. 2.63: Creating a custom field

To edit an existing custom field, select it and click "Edit". A window opens where you can change the values of the fields mentioned above.

To delete an existing custom field, select it and click "Delete". Confirm or cancel the deletion in the appeared dialog box.

### 2.3.3.2.5 "File Labels" tab

This tab displays a list of existing stickers – file labels (Fig. 2.64). They are used to mark photos when creating and editing tasks. For example, these could be "Before" and "After" labels to indicate photos showing progress in ongoing work.
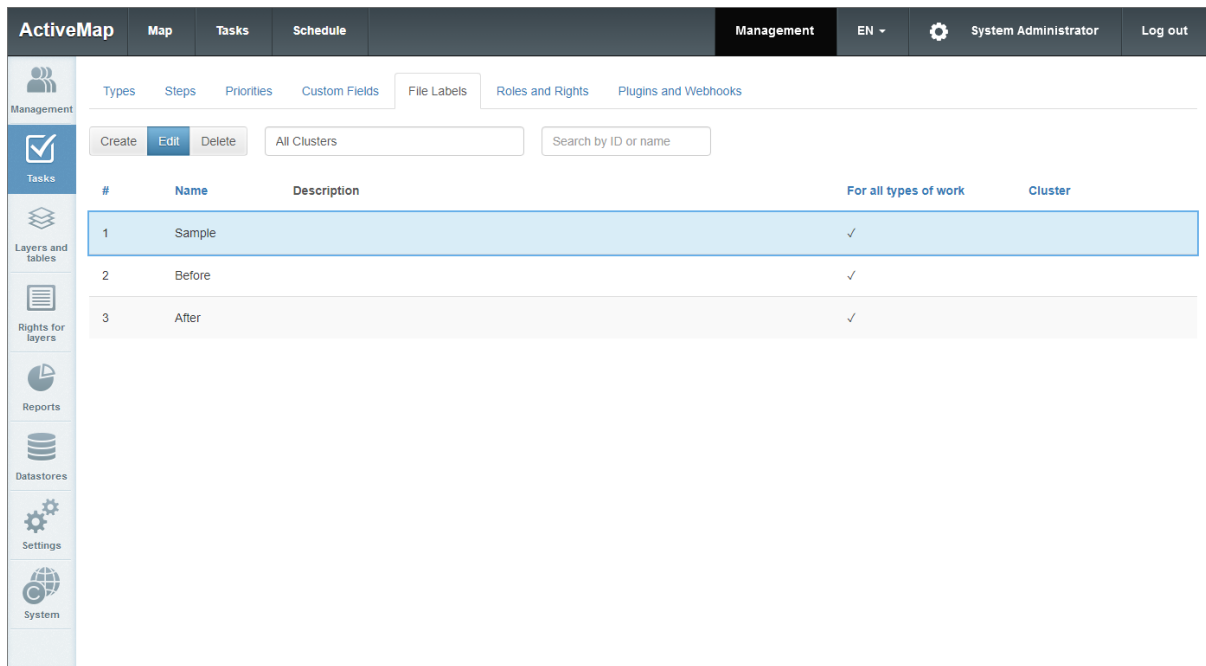
Fig. 2.64: "File Labels" tab

Stickers can be local (linked to a specific cluster) or global (not linked). Only the System Administrator can manage global entities. The System Administrator and Cluster Administrator can manage local entities. There is an exception – the Cluster Administrator can edit the global sticker (when "for all" mode is not enabled), while only attaching or detaching local types of work of the cluster.

To add a new label, click "Create". A window opens with fields to fill in (Fig. 2.65):

"Main" tab:

- "Name" – name of the label;
- "Description" – brief description of the label.

"Binding to types of work" tab:

- "For all types of work" toggle switch – makes sticker available for all types of work in the system (global sticker) or within the cluster (local sticker);
- toggle switches with the names of individual organizations – makes the sticker available for selected types of work.

Fig. 2.65: Creating a label for files

To edit an existing label, select it and click "Edit". A window opens where you can change the values of the fields mentioned above. To delete an existing label, select the row of the desired label and click "Delete". In the dialog box that appears, confirm deletion by clicking "Yes" or cancel it by clicking "No".

### 2.3.3.2.6 "Roles and Rights" tab

This section allows you to view the permissions of users with different roles (Fig. 2.66). For a detailed description of the possible roles, see *"Roles" tab* (page 42).

Fig. 2.66: "Roles and Rights" tab

To view the rights of a specific role, select it in the "Role" window. The list on the right side of the screen displays the permissions for that role. To view roles that have a specific permission, select them in the "Privilege" window. The result is displayed in the list on the right side of the screen. "Clear Filter" button resets the role and permission selection. As a result, all roles and all rights are displayed in the list.

### 2.3.3.2.7 "Plugins and Webhooks" tab

This tab displays existing plugins and webhook destination servers (Fig. 2.67). Plugins are scripts written in Python. They are used to automate actions with tasks, comments, and users and further adapt the system to operating peculiarities, e.g., for automatic:

- creation of new tasks after the occurrence of certain conditions in the performed task;

- transferring tasks to another status or step when attaching a certain number of files or filling in the specified fields;

- adding a comment on any change in the task;

- attaching files to the task and stickers to files;

- setting the main photo of the task;

- assigning a task when creating a new user.

Fig. 2.67: "Plugins and Webhooks" tab

Scripts and webhook destination servers can be local (bound to a specific cluster) and global (unbound). You can specify a cluster when creating a script/destination server but cannot change or delete it from the script/destination server. If the script is local, you can specify only a global server or a local server of the same cluster. If the script is global, you can specify any script. Priorities and types of work can be specified both local in relation to its cluster and global. Local scripts are applied to tasks and comments only in their own cluster. Only System Administrator can create, edit, and delete global and local scripts.

### "Plugins and Webhooks" subsection

To add a new plugin, go to the corresponding subsection and click "Create". The plugin creation window opens (Fig. 2.68).
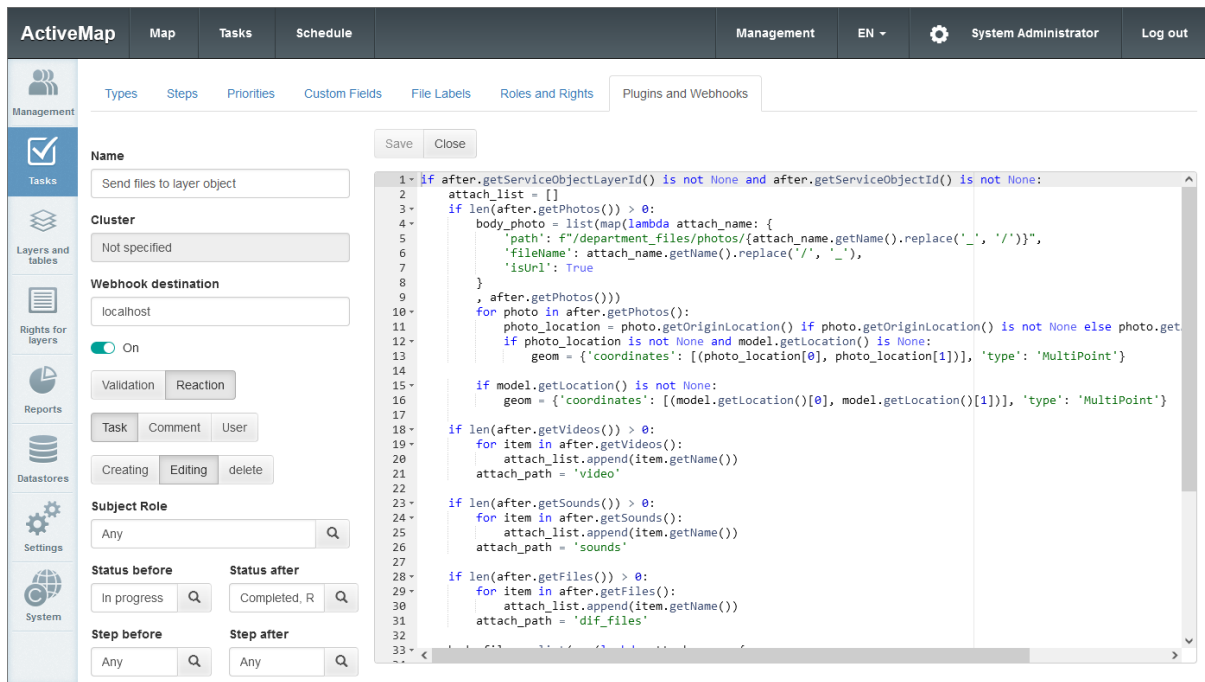
Fig. 2.68: Plugin creation

In the left part of the opened window, configure the parameters: fill in the "Name" and "Description" fields, specify the cluster if necessary, switch the "Enabled" toggle to the desired position.

Select the values of the following fields:

- "Type" – type of plugin (validation or reaction).

Validation is performed before saving, reaction – after saving the task in the database.

Validation allows checking the correctness of a task when creating/editing it, performing certain actions when creating/editing/deleting users. The validator plugin should return one of three possible values:

- *valid( )* – everything is correct, you can save it.

- *invalid( )* – error, you cannot save it.

- *invalid(String error)* – error, you cannot save it, show the given error message.

Reaction allows you to send a request for another operation on the same task or several such requests in response to a successful operation on the task. For example, in response to attaching new files to a task, send a request to change the custom field in the same task that stores the number of files. The request is executed in a separate thread from the reaction. In reaction you can specify the user on behalf of whom the request to create/edit the task is executed. If the reaction has an attached server and generates a relative request, the request is sent to the attached server (the request is ignored if the server is turned off or removed).

Webhooks are the special kind of reaction. This is an automated launch of http requests in response to operations on entities (tasks or com-

ments). Unlike regular reactions, you can write webhooks both for tasks and comments. For webhooks on tasks, operations for creating/editing/deleting tasks are available. Only the operation of creating a comment is available for webhooks on comments. To send a request, the reaction should return the result of executing one of the special functions:

- *return post(url),*
- *return patch(url),*
- *return put(url),*
- *return delete(url),*
- *return get(url).*
- "Model" – the entity on which the operation is performed (task or comment).
- "Operation" – the operation that triggers the plugin (creation/editing/deletion for tasks, creation for comments).

Depending on the selected type, model, and type of operation, the following fields are displayed:

Validation when creating/deleting a task:

- "Subject Role" – user roles that can use the plugin.
- "Organization" – task-creating organizations that have access to the plugin.
- "Type of work" – types of work for which the plugin works.
- "Priority" – priorities for which the plugin works.

Validation when editing a task:

- "Subject Role" – user roles that can use the plugin.
- "Status before" – task statuses for which the plugin works.
- "Status after" – the status to which tasks are transferred after the plugin is triggered.
- "Step before" – steps of task execution for which the plugin works.
- "Step after" – the step to which tasks are transferred after the plugin is triggered.
- "Organization" – task-creating organizations that have access to the plugin.
- "Type of work" – types of work for which the plugin works.
- "Priority" – priorities for which the plugin works.

Validation when creating a comment:

- "Subject Role" – users with which role can use the plugin.

---

In all of the above fields, you can select multiple values. "Destination server" can be added to this list of fields – the address of the server to which relative requests are sent.

The field on the right should be filled with a Python script. Roles, statuses, before/after steps, organizations, types of work, and priorities can be specified directly in the script in more complex combinations than can be enabled by turning toggles in the left part of the window. If these field values are mentioned in the script, default values ("any") can be left for them in the left part.

You can see script examples in the *Plugin script examples* (page 272) section.

After filling in all the required fields, click "Create". The created plugin appears in the general list of plugins. If necessary, you can edit or delete each plugin in the list.

### "Webhook destinations" subsection

Destination servers are needed for grouping webhooks and for the fast change of the destination server. For example, if you have 5-10 webhooks and their API entry point has changed, you have to change only the destination server address. If the reaction has an attached server and generates a relative request, the request is sent to the attached server (the request is ignored if the server is turned off or removed).

To add a new server, go to the appropriate subsection and click "Create". In the window that opens, fill in the "Name", "Cluster" (if necessary), and "URL" fields, switch the "Enabled" toggle to the desired position (Fig. 2.69).

Fig. 2.69: Creating a destination server

The created server appears in the general list of webhook destination servers ([Fig. 2.70](#)). If necessary, you can edit or delete the server data in the list (except for the cluster). You can set the cluster when the destination server is created, but you cannot change or delete it.



Fig. 2.70: List of webhook destination servers

### 2.3.3.3 "Layers and tables" block

The "Layers and tables" block is intended for working with cartographic layers of the system, tables, and their groups. If you switch to the "Layers", "Groups", "Tables", or "Icons" tabs, you get access to the following controls: sorting of tables, search bar, adding new records, editing records, deleting records, and exporting data (Fig. 2.71).
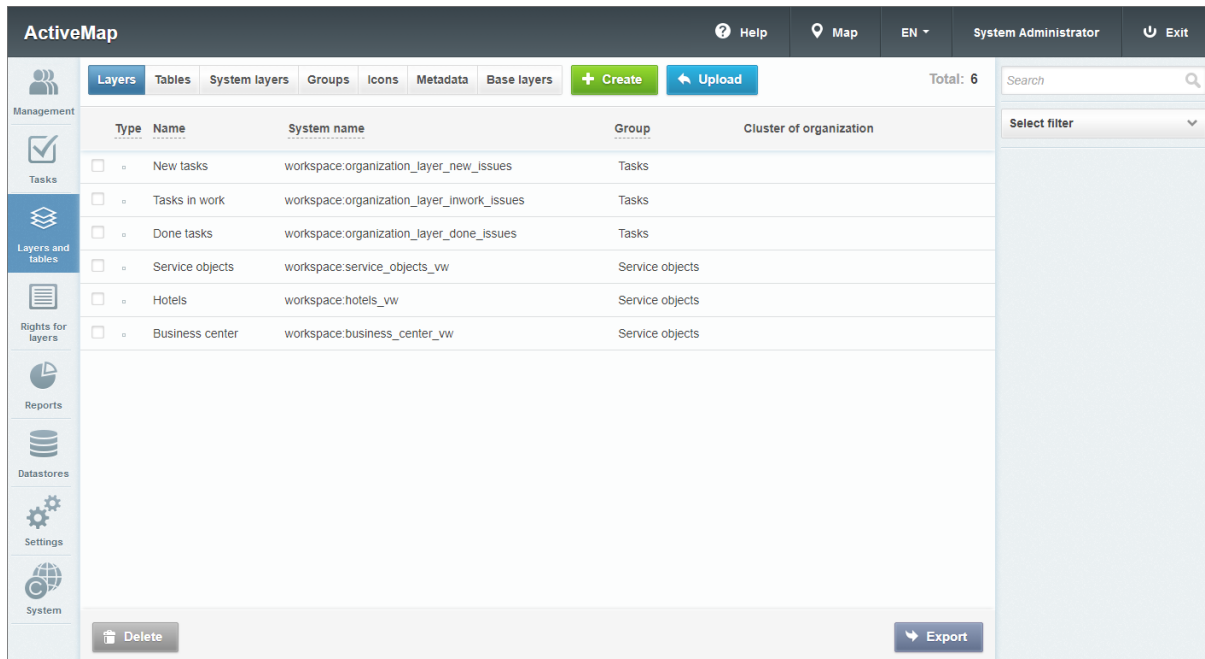


Fig. 2.71: "Layers" block

### 2.3.3.3.1 "Layers" tab

In the "Layers" tab you can view detailed information about the system layers, create, or load new layers. A layer is a visual representation of a set of geographic data on any digital map.

When you switch to the "Layers" tab in the administration window, a table with the following columns becomes available (Fig. 2.71):

- "Type" – geometry type (point, line, polygon, or raster) that determines how the layer object appears on the map.

- "Name" – name of the layer in the interface.

- "System name" – name of the layer in the database.

- "Group" – belonging to a certain group of the system.

- "Cluster of organizations" – an association of several organizations for operational monitoring of the work of departments (for more information about clusters, see *"Clusters of organizations" tab* (page 47)).

In the "Layers" tab you can use search strings and filters (by group, data store, geometry type, belonging to raster layers, by service objects and

cluster), tools for creating new layers, and editing/deleting the existing ones.

### Creating a new layer

To create a new layer in the system, click **+ Create** . The layer creation window opens, containing the tabs: "Main", "Attributes", "Service Objects" and "Default rights".

**"Main" tab**

The first tab that opens is the "Main" tab, where you have to fill in the following fields (Fig. 2.72):

- "Layer cluster" – belonging to the cluster of organizations. Cluster selection is available when creating a layer as a System Administrator. When creating a layer as an Cluster Administrator, affiliation with the cluster is automatically determined.

- "Name" – name of the layer in the interface.

- "System name" – name of the layer in the database. It should consist of letters from the Latin alphabet, without spaces or special characters. It is generated automatically when entering information in the "Name" field. If a non-Latin script title is entered in the "Name" field, transliteration is used. If you are not satisfied with the received name, you can enter your own version in this field. Automatic input does not work if you first fill in the "System name" and then the usual "Name". Unlike the name, you cannot edit the system name after the layer has been created.

- "Group" – group of layers in which the layer is displayed.

- "Datastore" – database in which the layer is stored.

- "Geometry type" – point, line, or polygon.

- "Layer protocol":

  - WMS – providing information in the form of a geographically referenced images;

  - WFS – providing information in the form of geospatial data.

- "Projection" – code of one of the common geographic projections.

- "Use for search" – indexing the layer to search for its objects.

- "Style" – description of the layer display properties on the screen (color, size, transparency, and other properties of the layer objects and their labels).

Fig. 2.72: Creating a layer

When creating and editing a layer, you can select one of the following styles:

- "Basic" – default style (point, line, or polygon).
- "Simple" – style, where you can select an attribute for the caption and set its color, background, transparency and size (Fig. 2.73).

Fig. 2.73: Simple layer style

If you select the simple style and the point geometry type, you can set the form of displaying points for each object (circle, triangle, or square) or choose a style with an icon. You can set the background color and icon size and outline the shape. If you use the style with icon, you should select one of the icons from the drop-down list and specify its size. The system also supports loading your own icons (more details in *"Icons" tab* (page 93)).

If you select the line as the geometry type, you can set the background color and line thickness. To set a color for the stroke, select the appropriate stroke option.

If you select polygon as the geometry type, you can set not only the colors and sizes for the stroke, but also the background transparency.

- "Advanced" – a style generated using the GeoCSS language with support for filters and a legend. When you select this style, a separate form with a code (Fig. 2.74) appears to the right of the input fields. See *Attachment 2. Examples of advanced layer styles* (page 273) for examples of extended styles. For more details on the rules for creating geocss styles, see https://docs.geoserver.org/stable/en/user/styling/workshop/css/css.html.

Fig. 2.74: Advanced layer style

**"Attributes" tab**

To add new attributes, fill in the "Name" and "Type" fields and click

.



Fig. 2.75: Adding layer attribute data

The following types of attribute fields are supported:

- String – a short text field;

- Integer – an integer field;

- Big integer – a numeric type that makes it possible to work with integers of arbitrary length;

- Boolean – a choice of true and false;

- Float – a field with a real numeric value;

- Date – selection of date from the calendar (day-month-year);

- Date and time – selection of date (day-month-year) and time (hours-minutes);

- Dictionary – selection from a list of values from the specified reference table (dictionary);

- Data table – selection from a list of values from the specified data table;

- Layers – selection from a list of objects from the specified layer;

- Organization connection – selection from a list of available organizations;

- Cluster connection – selection from a list of available clusters;

- User connection – selection from a list of available users;

- Work type connection – selection from a list of available work types;

- Priority connection – selection from a list of available work priorities.

    After creating an attribute, a table with the name, system name, and attribute type opens in the administration area. The system name is automatically assigned to the attribute based on transliteration of the entered name or based on the type for the following attribute types:

- Organization connection – sys_org_id;

- Cluster connection – sys_clr_id;

- User connection – sys_user_id;

- Work type connection – sys_typ_id;

- Priority connection – sys_prt_id.

If there are several fields in a layer with one of the specified types, then {current date} is automatically added at the end of the system attribute name.

New attribute appears in the first line of the table (Fig. 2.75). The following actions are available to the administrator:

- changing the attribute's name;

- defining additional parameters of the attribute:

  - "Title" – clicking on the line makes attribute the title of the object's card displayed when you click on the object on the map.

  - "Subtitle" – clicking on the line makes attribute the subtitle of the object's card displayed when you click on the object on the map.

  - "For search" – toggle switch to use the attribute for search.

  - "View" – toggle switch for showing atribute for users.

  - "Can be edited" – toggle switch for attribute editing availability.

  - "HTML escape" – toggle switch to interpret the attribute as HTML content (expands the possibilities of filling in attribute fields, for example, their content can be a link or a formatted text).

  - "View in pop-up window" – toggle switch to display the attribute in the object's card that is displayed when you click on the object on the map.

To add the next attribute, you have to fill in the form with fields again and click **+ Add**.

In the upper part of the window, there are the "Title format" and the "Subtitle format" fields. You can form the title/subtitle using a mask for the layer from one or several attributes. To set up a new mask, click "Title" or "Subtitle" on the desired attribute. You can add a brief explanation for better perception.



| Title format | Subtitle format |
| --- | --- |
| {name} | {address} |

Fig. 2.76: Example of setting a title and subtitle mask

You can limit the visibility of objects of the same layer for different clusters, organizations, and users in the system. To do this, you have to:

1. Create a field with one of the following data types: "Organization connection", "Cluster connection", or "User connection".

2. Select its name from the drop-down list in the "Limit the visibility of objects by" field at the top of the window.

3. After saving the layer attribute structure, fill the connection field with values by selecting from the drop-down list.

Objects of this layer are visible only for the users of the cluster (or organization) specified in this field or for the users listed in the "User connection" field.

Users with "System Administrator" and "System Inspector" roles can leave the system filter field empty when creating an object and the ob-

ject becomes available to all users. For other users, the field is filled by default with the name of the main organization or main cluster of the user. The object is accessible only to users of this organization or cluster.

To facilitate the process of filling the created attribute fields and to filter objects on the map by attribute values, you can set the link to a reference table, to a data table, or other thematic layers. To do this, create a table in the "Tables" tab of the "Layers" block (*"Tables" tab* (page 83)) and select "Reference table (dictionary)" or "Data Table" as the field type of the current layer. To link to a thematic layer, select the "Layers" type and select the thematic layer. A drop-down list with the names of available tables appears to the right of the field type. After selecting a particular table, specify the "Value Field" with "integer" data type (the source field for the link) and the "Name Field" (the field that stores the names of the elements) (Fig. 2.77).



Fig. 2.77: Attaching a reference table to the layer field

After attaching a reference or data table (Fig. 2.78) you can select one of the values from the drop-down list instead of entering an attribute value when creating new layer objects and when filtering.
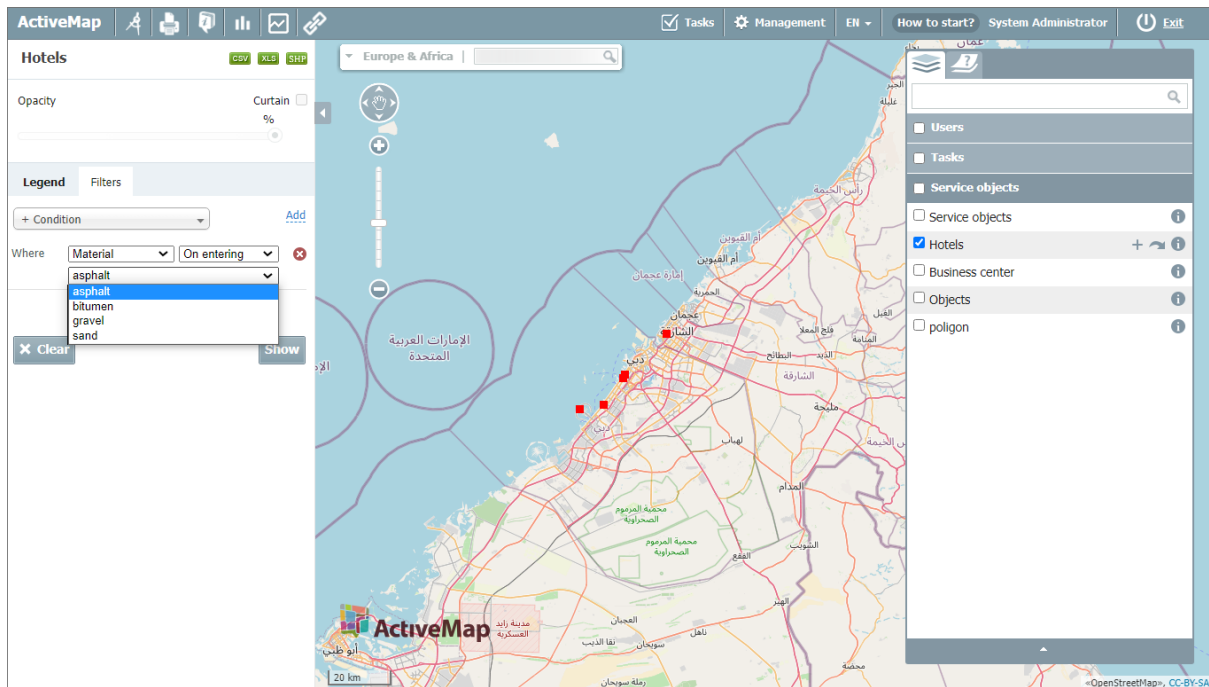
Fig. 2.78: Using reference table to filter layer objects

Fields with "Cluster connection", "Organization connection", "User connection", "Work type connection", and "Priority connection" types allow you to connect system reference tables (dictionaries). System dictionaries are generated automatically based on data entered into the system.

**"Service Objects" tab**

Service objects are layers containing objects of interest of the user organization related to its activities. In this tab, you can set the mapping between the fields of this layer and the fields of tasks that are created based on the service objects. This means that when creating tasks with a link to service objects, all or part of the task fields are automatically filled with data about this service object. The mapping determines which fields it is.

You can define a layer as a service layer in the "Rights for layers" block (for more information, see *"Rights for layers" block* (page 100)). Depending on the settings in the "Rights for layers" block, some users can use the layer as a service layer, while others can use it as a regular layer.

To configure service objects, enable the corresponding toggle switch. Then select an attribute for the name of the service object from the layer fields and click ➕. The name format displays its mask. The name can consist of several attributes. To do this, create an appropriate mask by adding new attributes.

To set the mapping of service layer attributes and task fields, click **➕ Add a match**. Select the layer attribute and the task field from

the drop-down lists. To delete the mapping, click ![delete icon] next to the mapping. The attribute format can include several layer fields. You can also add an explanation to this line to help you understand the information in the tasks.
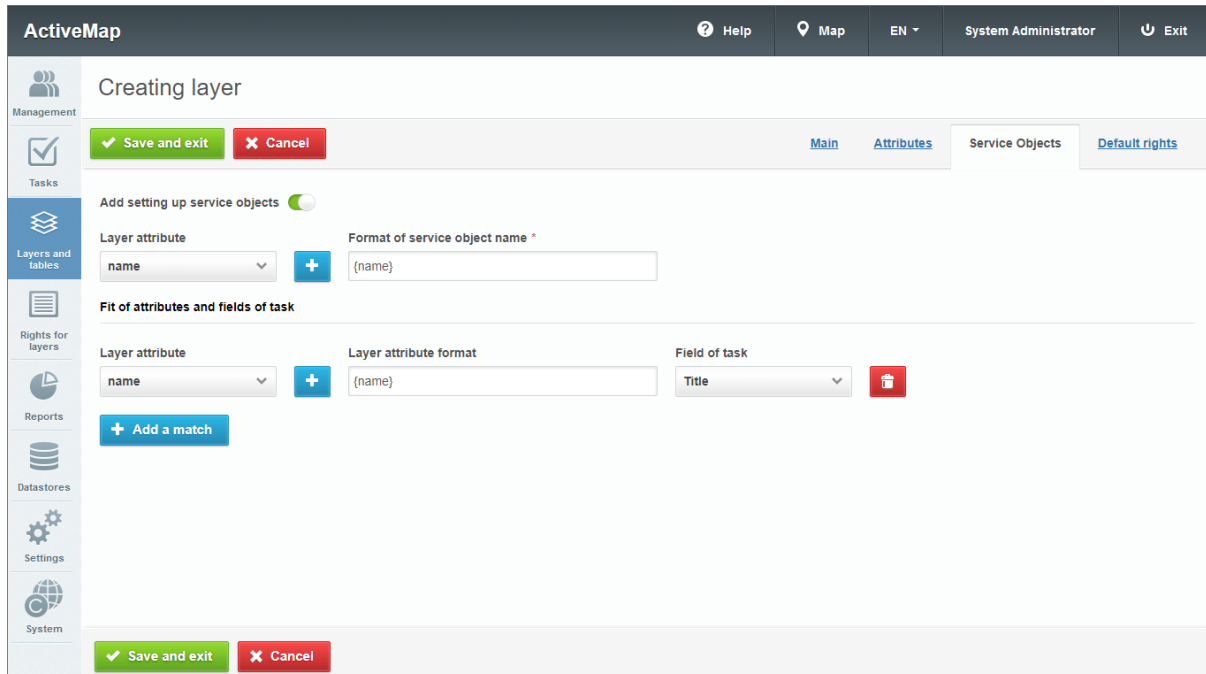


Fig. 2.79: "Service objects" tab

**"Default rights" tab**

In this tab you can set the default rights for viewing/editing/managing and set the rights for using the layer as a layer with service objects. To set the rights, select the options from the drop-down lists:

- "All Roles" – granting rights to all roles;

- "Roles" – selecting roles to set rights for;

- "Multiple organizations" – selecting all client organizations or all non-client organizations, or all regardless of this criterion;

- "Organization" – selecting the organization to set rights for.

After selecting the parameters, click ![Add button]. A new row appears in the table below. Set the necessary rights using the "Service Objects", "View", "Edit", and "Manage" toggle switches (for more information about the rights, see *"Rights for layers" block* (page 100)).

Fig. 2.80: "Default rights" tab

### Editing layer information

To change layer information, click  or double-click on the row with the name of the selected layer. A form (similar to the add form) opens in the administration area. Here you can fill in/change the fields of the layer (Fig. 2.81).
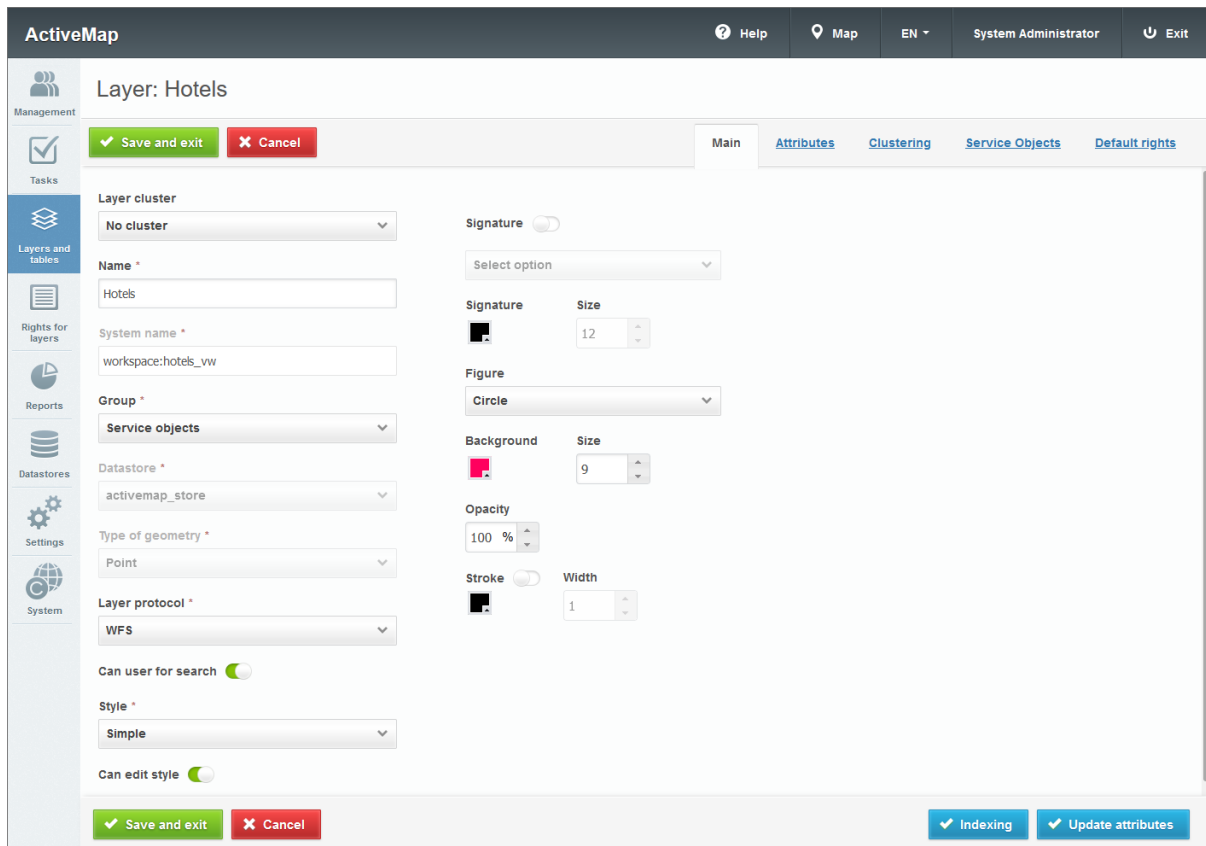
Fig. 2.81: Editing layer information

When you change cluster membership, users in the new cluster are automatically granted rights to the layer, configured by default. Users in the previous cluster retain their rights to the layers and can edit them if necessary.

In the editing window, the "Index"  and "Update attributes"  buttons become available to the administrator. Clicking  generates a full update of the information about the selected layer. The administrator can use this button if new information has been added and is temporarily absent in the search results.

To change information about layer attributes, switch to the "Attributes" tab in the layer editing window. Here you can delete an attribute by clicking  on the right side of the line. The  button allows you to reset the cache when adding attributes to a layer and reload them from the database.

To save the changes made, click . To cancel editing the layer, click .

When editing a point layer (i.e., a layer with the "Point" geometry type), the "Clustering" tab appears in the administration area in addition to the "Main" and "Attributes" tabs.

Clustering is the display of a group of point layer objects located nearby with a single mark on the map. Clustering is possible only for point layers with the WFS display method. The amount of objects grouped into a cluster is displayed as a number. The proximity of objects to be included in the cluster is calculated based on the scale.

When you switch to the "Clustering" tab in the editing window, a form opens. Here you can enable clustering, set the maximum zoom level, and create a new cluster (Fig. 2.82). Here you can see the graphs with the following headers: object count, icon, and label color.
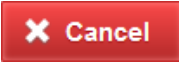


Fig. 2.82: Layer clustering settings

To add a new cluster, specify in the input field the number of objects contained in this cluster. Select the image displayed on the map, set the label color, and click [+ Add].

The "Object Count" column shows the range of the number of objects that corresponds to a specific cluster. If the number 5 is next to the first cluster, 10 is next to the second, and 15 is next to the third (as shown in Fig. 2.82), it means that up to 5 objects fall into the first cluster, from 6 to 10 in the second, and from 11 to 15 in the third. If the third cluster with 15 objects is the last in the list, then there will be no finite number of objects for it.

The "Icon" column stores cluster images that become available when viewing the map. The "Label color" column displays the color used for the caption on the map.

### Deleting a layer

To delete a layer, click ![trash icon] on the right side of the layer row. To delete multiple layers at once, select the corresponding rows and click ![Delete button] at the bottom of the screen. The delete confirmation window appears with a choice of layer deletion modes: "Delete from geoportal", "Delete from geoserver", and "Delete from database" (Fig. 2.83). You can choose multiple options. To completely delete a layer, select all 3 items. Click ![OK button] to confirm the deletion. Click ![Cancel button] to cancel.
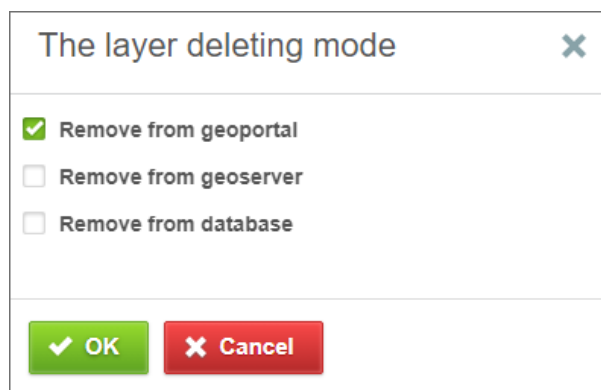


Fig. 2.83: Layer deletion confirmation

### Searching for a layer

You can work with the search bar and filters in the "Layers" tab. There are filters by group, geometry type, layer type (raster/vector), service objects, and clusters.

For example, when choosing the filter "By geometry type", a form appears where you have to select one of the types (point, line, or polygon) from the drop-down list to filter out layers. After that, layers with the selected geometry type appear in the administration area.

To clear the filtering results, click "Clear all".

### Loading a layer

To import a layer into the system, click ![Upload button] at the top of the "Layers" tab. A pop-up window appears where you can choose a layer from the computer. You can upload archived vector shape-files in zip format and georeferenced raster images in GeoTIFF format.

When importing a layer from a shapefile, the system automatically identifies some attribute types if the attribute field name starts with the following words:

- sys_org_id – "Organization connection" type;

- sys_clr_id – "Cluster connection" type;

- sys_user_id – "User connection" type;

- sys_typ_id – "Work type connection" type;

- sys_prt_id – "Priority connection" type.

It is desirable to compress (lzw) the geotiff file and build pyramids for it. You can do it using the tools of the GDAL library (https://gdal.org/).

The stages of layer loading are displayed in the information window (Fig. 2.84).
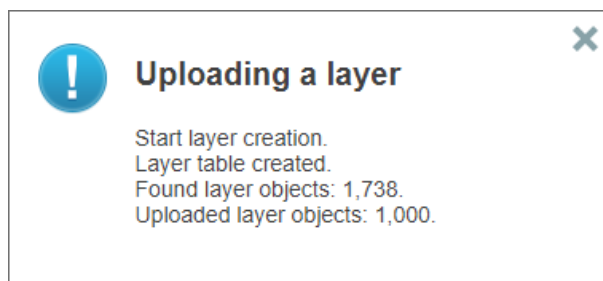


Fig. 2.84: Displaying layer loading stages

After loading, the same window opens as when creating a layer. The geometry type and layer display protocol are automatically determined, while the remaining fields need to be filled in. Meanwhile, attributes (if any) are also set automatically. You can independently select a data storage from the presented list in the "Main" tab.

### "Tables" tab

The "Tables" tab contains information about data tables and reference tables (dictionaries) of the system. Data tables and reference tables, unlike layers, do not contain spatial information about the location of objects such as geometry or coordinates of objects.

Reference tables and associated data tables are used to solve the following tasks:

- Simplification of filling attribute fields. With linked reference or data table, users can select the value of the attribute field from the suggested variants instead of entering it manually.

- Filtering layer objects on the map;

- Creation of thematic maps based on reference tables;

- Applying style according to the reference table.

When switching to the "Tables" tab, a table with the following columns becomes available (Fig. 2.85):

- "Title" – name of the table in the interface.

- "name in DB" – name of the table in the database (consisting of the letters of the Latin alphabet, transliteration of the "Title" field by default).

- "Table type" ("Data table" or "Dictionary" ("Reference table")).

- "Cluster of organization" – name of the organization cluster to which the table belongs.
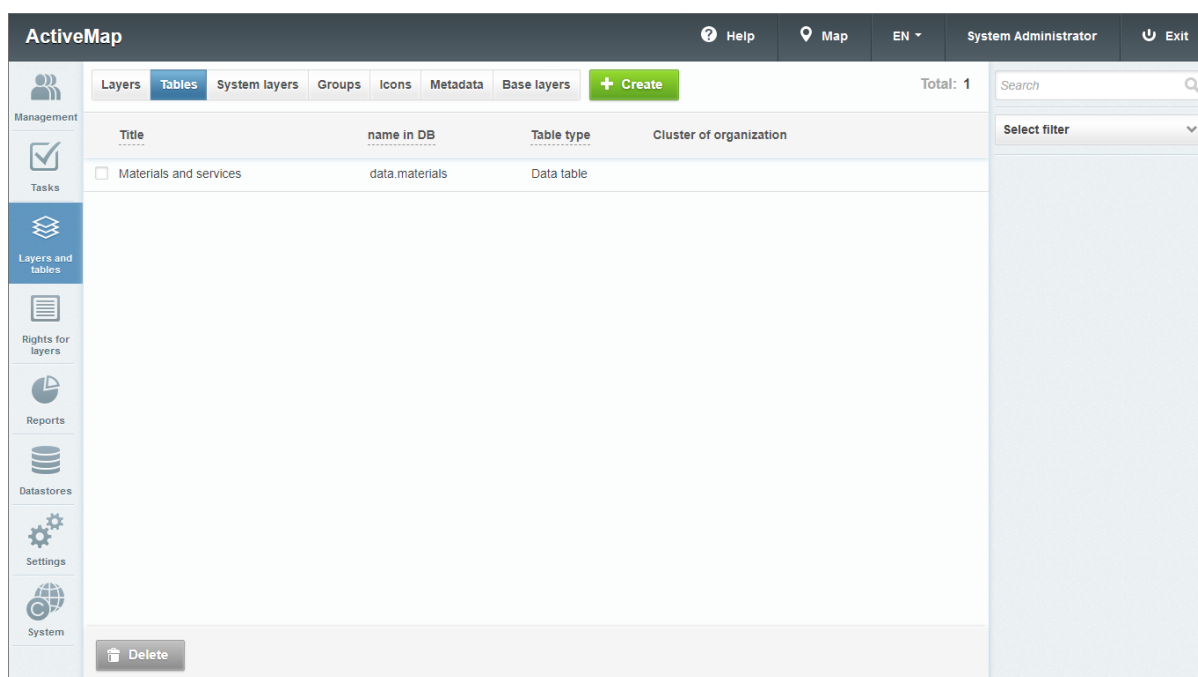


Fig. 2.85: "Tables" tab

To add a new table, click ![Create] at the top of the window. The "Main" tab appears, similar to the layer creation form (Fig. 2.86).

Fig. 2.86: Creating a table

Fill in the following fields:

- "Type" – type of the table (reference table (dictionary) or data table).

- "Data table cluster" – belonging to the cluster of organizations. Cluster selection is available when creating a table under the System Administrator role. When creating a table under the Cluster Administrator role, the cluster belonging is determined automatically.

- "Name" – name of the table in the interface.

- "System name" – name of the table in the database, consisting of the letters of the Latin alphabet, without spaces and special characters. It is formed automatically when entering information in the "Name" field. If you enter a non-latin title in the "Name" field, the transliteration is used. If you are not satisfied with the received name, you can enter your own variant in this field. Automatic entry does not work if you fill in the "System name" first, and then the usual "Name". Unlike the name, you cannot edit the system name after the table is created.

- "Group" – the group in which the table is displayed.

- "Datastore" – the database in which the table is stored.

You can enable the following toggles:

- "Can use for search" – include layer data in the full-text search in the ActiveMap Desktop desktop application.

- "Attach files and photos" – enable the ability to attach files and photos to the layer objects.

The "Attributes", "Service Objects" and "Default rights" tabs are filled in similarly to the tabs of the same name in the layer creation form (section *Creating a new layer* (page 70)). After entering all the necessary

information, click ![Save and exit](green button).

To modify a table, click ![edit icon] on the right side of the corresponding line. A window similar to the table creation window opens where you can change or add the data. When changing the cluster membership, default permissions for the table are automatically granted to the users of the new cluster. Permissions for the table for users of the previous cluster are saved. You can edit them, if necessary.

To delete a table, click ![delete icon] on the right side of the corresponding line. To delete multiple tables at once, check the corresponding lines and

click ![Delete](red button) at the bottom of the screen. For more information about deleting system elements, see *Deleting an element* (page 29).

### Reference tables

A reference table (dictionary) is a table with systematically organized data, designed to facilitate the user's actions when working with attribute information on objects.

To form a reference table (dictionary), create and fill in a table with the "Dictionary" data type with attribute fields, among which there must be a source field for linking with integer values ("id" or "gid") and a value field – an attribute field, in which the names of reference elements are stored (Fig. 2.87). Attaching a reference to a layer field is described in *Creating a new layer* (page 70).

> **Attention:** Values in the source field for linking must be unique. Without this, it is impossible to attach the reference table (dictionary) to another table.

Fig. 2.87: Reference table (dictionary) attributes

## Relationships between tables

Objects of different tables can be logically related to each other. Tables are connected by a "one-to-many" relationship, which means the ability to link several objects of one table (child table) to one object of another table (parent table).

To create a "one-to-many" relationship, add a link field (integer type) to the child table (Fig. 2.88) and set up the relationship with the parent table (described in *Creating a new layer* (page 70)). The parent table must have a text field to describe its objects with the "String" type (Fig. 2.89). This field is used to display values in the key field of child table objects.
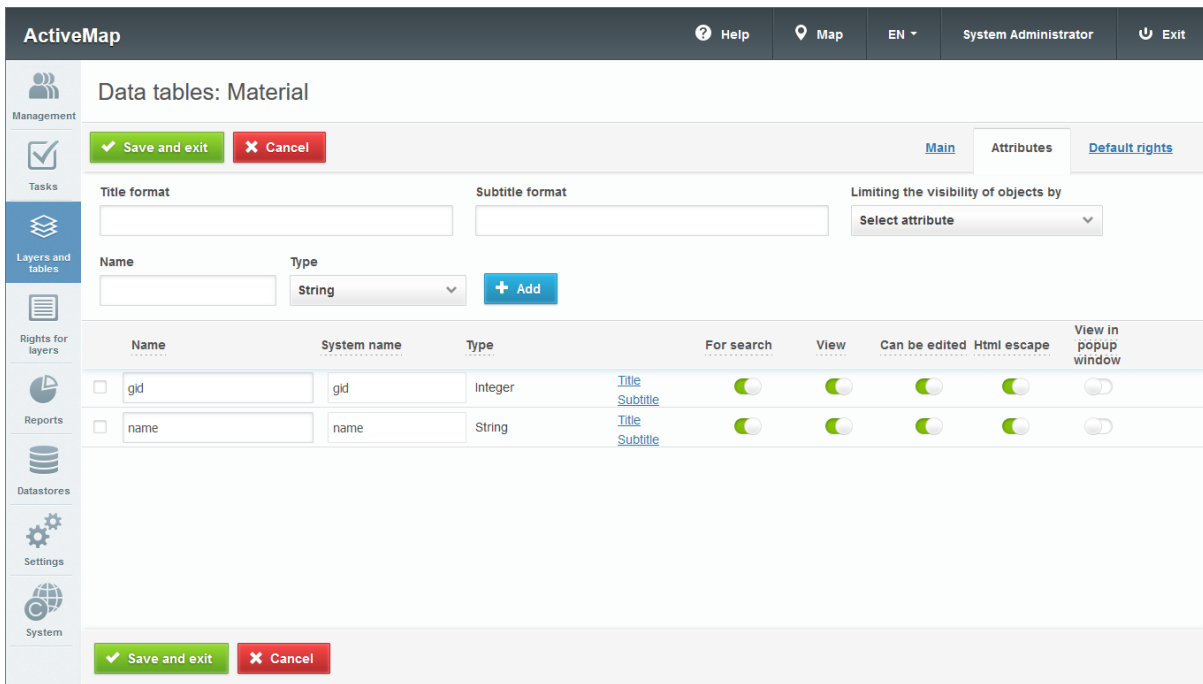
Fig. 2.88: Parent table attributes
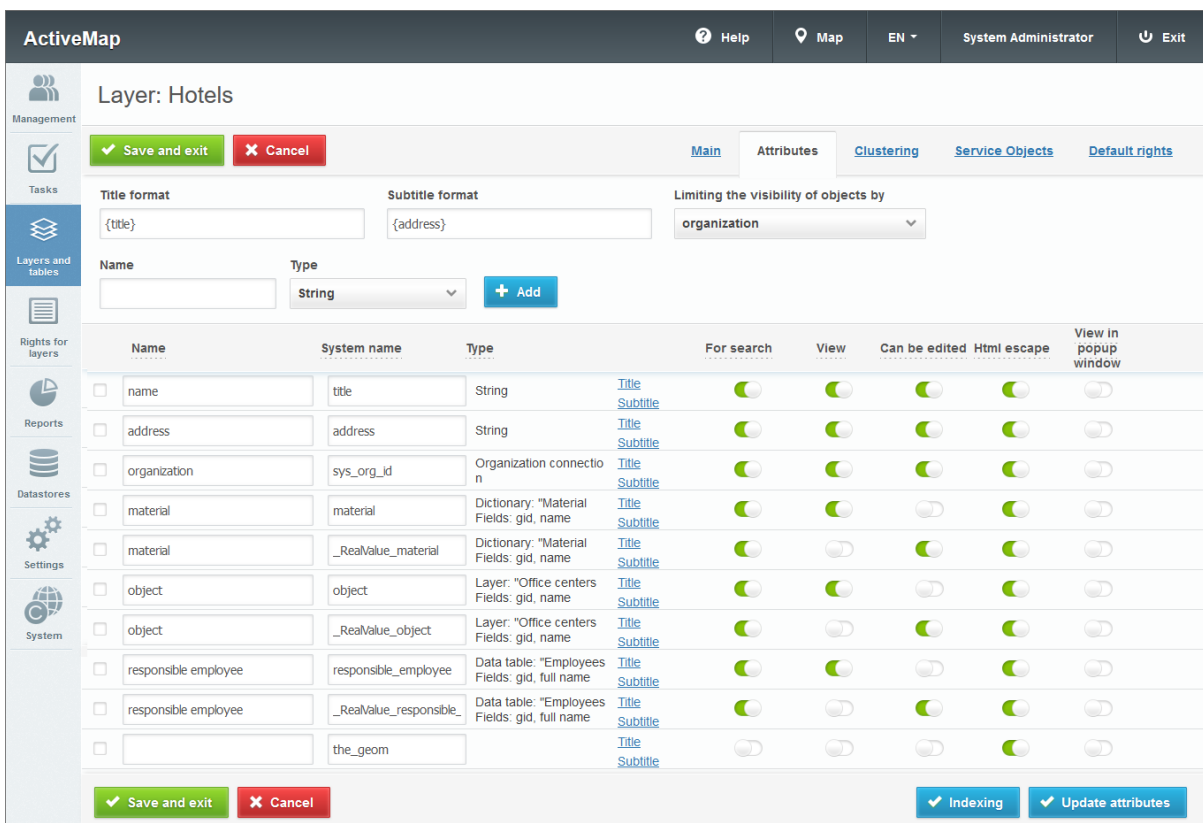


Fig. 2.89: Child table attributes

### 2.3.3.3.2 "System layers" tab

The "System layers" tab displays system-created layers (Fig. 2.90). Usually, there is only one system layer – "Users". Geographic data for it is taken from the coordinates of ActiveMap system mobile application users. The style for this layer is set programmatically.
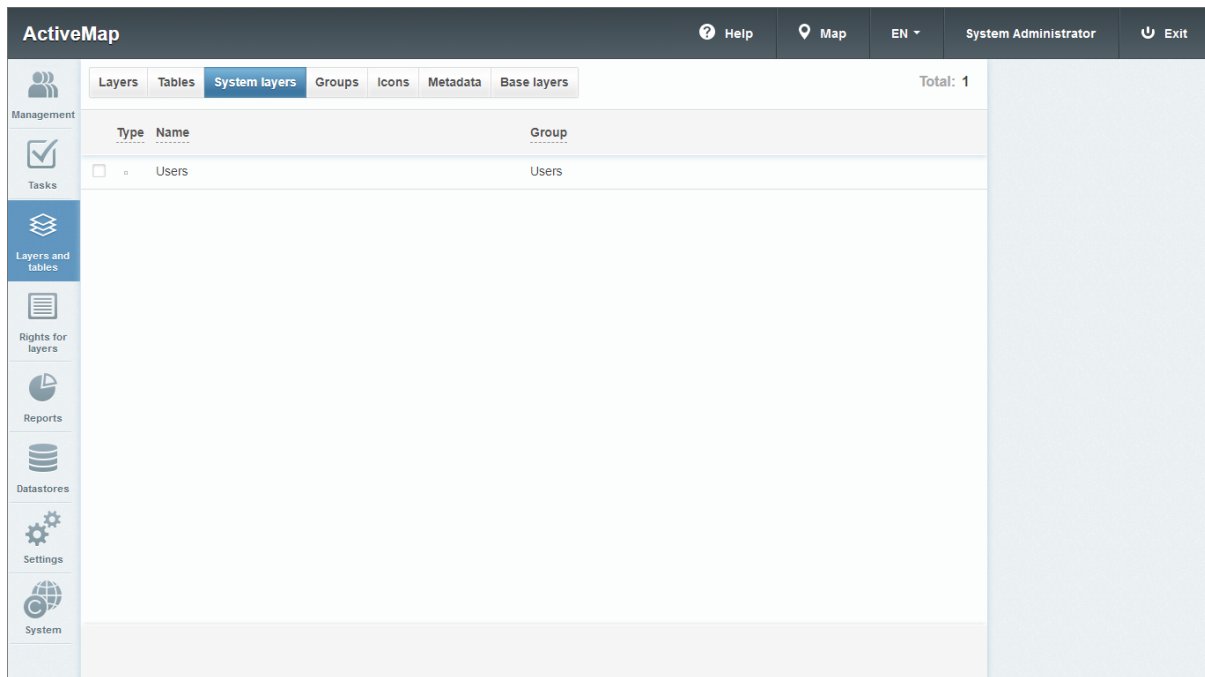


Fig. 2.90: "System layers" tab

Only the System Administrator can edit the "Users" system layer. Other roles, if they have rights to this layer, can only grant permissions to other users within their cluster or organization.

The System Administrator can set a custom color palette and layer legend elements for the system layer, as well as define default permissions.

To do this, click ✏ on the right side of the layer row or double-click on the row. The "Main" tab opens, similar to the form for creating a regular layer, except for the possibility of adding attribute fields.

Users in this layer are divided into groups based on the activity interval – the time when the system last received their coordinates. Users in these groups have their own icon color.

The System Administrator can configure the following fields of the system layer (Fig. 2.91):

- "Name" – name of the layer.

- "Group" – group of layers in which the system layer is located.

- "Activity intervals of the last coordinates transmitted by users" – time intervals. They are divided into three groups by default:

- – Active – coordinates received no more than 15 minutes ago (green color);

- – More than 15 minutes ago – coordinates received from 15 to 60 minutes ago (yellow icon);

- – More than an hour – coordinates received more than an hour ago (red icon).

- • "Group name for users outside of defined intervals" – name for the users whose coordinates were last transmitted before the latest time interval (by default, before a day).

- • "Group name for users with no information about their location" – name for users whose geolocation is disabled.

You can set your own color for each parameter.

You can also set your own activity intervals. To delete an existing interval, click ![delete] to the right of the interval. To create a new interval, click the "Add interval" button at the bottom of the window, fill in the field "Name of interval in the interface", and set the time interval and color of the icon.

After all the changes have been made, click ![Save and exit].
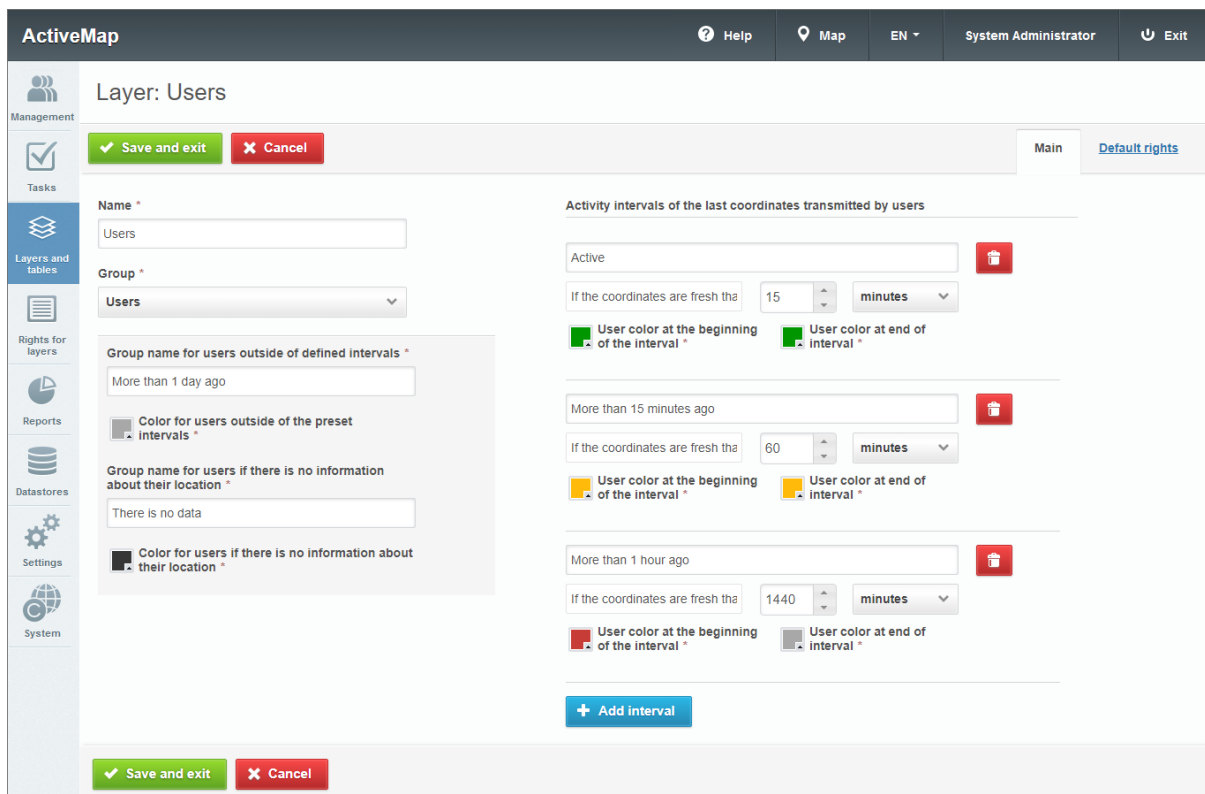


Fig. 2.91: Editing the display of the "Users" system layer

In the "Default rights" tab you can set the rights to view/edit/manage, as well as to use the layer as a layer with service objects for different

user roles in different organizations (Fig. 2.92).



Fig. 2.92: The "Default rights" tab for the "Users" layer

### 2.3.3.3.3 "Groups" tab

Groups combine layers and tables by thematic or other defined criteria. When you switch to the "Groups" tab in the administration area, a table appears with the following headings (Fig. 2.93):

- "Ordinal number" – place in the list of layer groups on the main page of ActiveMap Web, occupied by the group;

- "System group" – indication that a group contains system layers (by default, "User monitoring" is a system group);

- "Name" – name of the group;

- "Cluster of organizations" – belonging to the cluster of organizations;

- "Layers" – number of layers and tables in the group.
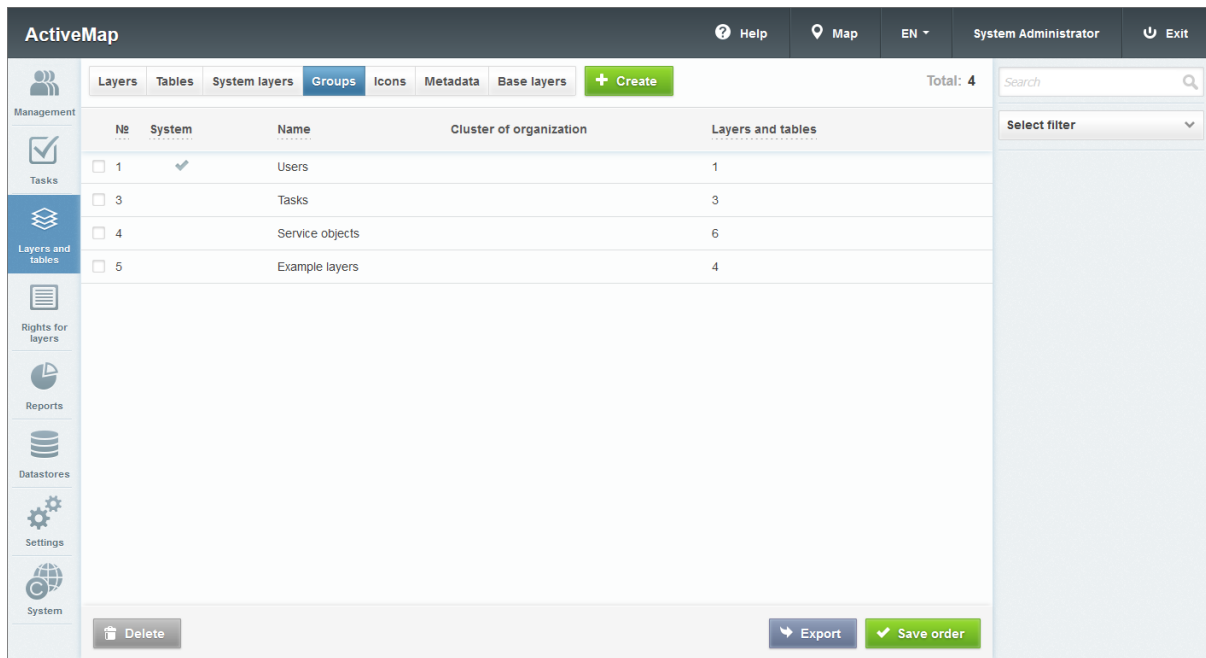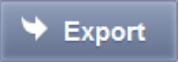
Fig. 2.93: "Groups" tab

In the "Groups" tab, the administrator has access to the "Save order"

button ![Save order]. It is used to sort groups alphabetically (or in ascending/descending order of layers) and save the resulting order in the system. Clicking the button automatically changes the sequential numbers (in the "No." column) for the sorted groups.

To rearrange groups, click on a certain group and, without releasing

the button, move it to the desired line, and click ![Save order].
When switching to the "Groups" tab, you can use the search bar, as well as tools for creating new groups and editing/deleting existing ones.

In addition, in the "Groups" tab, you can save information about groups

to your computer. To do this, click ![Export] and select a directory to save the Excel file. This file contains a table with the names of groups and the number of layers and tables in each group.

### Adding a new group

To add a new group to the system, click at the top of the window. A form opens in which the user can enter the name of the new group.

### Editing a group

When editing a group in the administrative area, a new window opens with the following tabs: "Main" and "Layers". In the "Main" tab you can edit the name of the group. When switching to the "Layers" tab, you can view information about the layers belonging to the group (sequential number, name, and system name).

To swap layers, click on a layer name and move the layer to the desired row without releasing the button.

### Deleting a group

To delete a group, click on the right side of the corresponding row. To delete several groups at once, select the checkboxes of corresponding

rows and click at the bottom of the screen. For more information about deleting system elements, see *Deleting an element* (page 29).

### Searching for a group

You can use the search bar to search for groups by their names and filter by clusters in the "Groups" tab.

#### 2.3.3.3.4 "Icons" tab

When switching to the "Icons" tab, you can see a table with the following columns: icon and name (Fig. 2.94). You can use icons when creating styles and adding clustering to the point layers (for more information about clustering, see *Editing layer information* (page 79)).
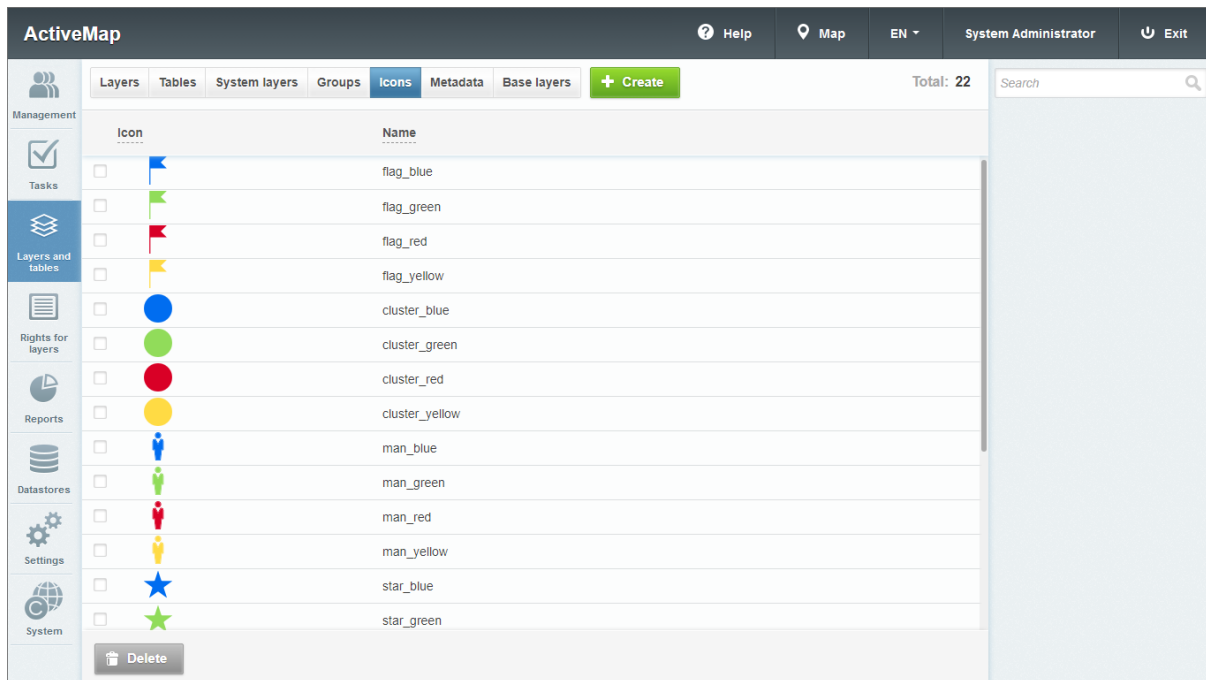
Fig. 2.94: "Icons" tab

The "Icons" tab contains a search bar and tools for adding new icons and editing/deleting existing ones.

### Adding an icon

To add a new icon, click  at the top of the window. A window opens to enter the icon's name and upload the image by clicking on the corresponding upload form.

### Editing an icon

When clicking  (or double-clicking the row with the selected icon), a form opens in the administration window (with the same fields as the form for adding an icon) where you can fill/change the fields with information about the selected icon.

### Deleting an icon

To delete an icon, click  on the right side of the corresponding row. To delete multiple icons at once, select the corresponding rows with checkboxes and click  at the bottom of the screen. For more information about deleting system elements, see *Deleting an element* (page 29).

### Searching for an icon

You can use the search bar to search for images by their names.

### Getting a link

To get a link to an icon, hover over it, right-click, and select "Copy image URL" or "Copy link to image" depending on the browser used. You can use this link when creating an advanced layer style with icons.

#### 2.3.3.3.5 "Metadata" tab
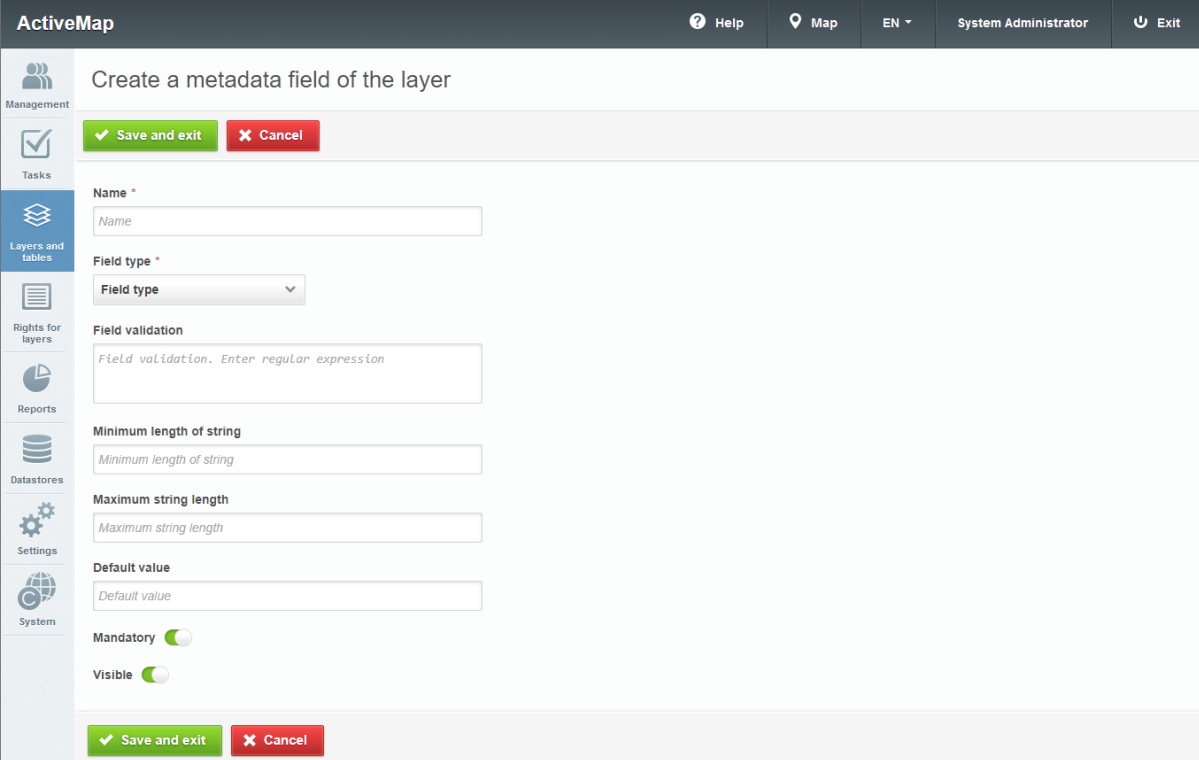
In the "Metadata" tab, you can view the custom fields that are displayed in the "Additional" tab when creating or editing a layer, as well as in the information window in the client part of the site. When switching to the "Metadata" tab, you can see a table with the following columns:

- "Name" – name of the custom field;

- "Field type" – input data type (string, integer, fraction, list, condition, or date);

- "Default value" – data that is initially specified in additional field;

- "Mandatory" – mandatory flag for filling in this field;

- "Visible" – the flag for displaying the field in the "Metadata" tab when creating or editing a layer, as well as in the information window in the client side of the site.

In the "Fields" tab, you have access to a search bar as well as tools for adding new fields and editing/removing existing ones.

### Adding layer metadata

In the "Metadata" tab, you can add a new field for entering additional information about the created layer. To do this, click [+ Create] at the top of the window. A field creation window opens. Fill in the suggested fields (Fig. 2.95).



Fig. 2.95: Adding metadata

### Editing layer metadata

To edit the information, click [✏] or double-click on the row with the name of the selected field. A form similar to the creation form opens, where you can modify the information about the additional fields of interest.

### Deleting metadata

To delete a field, click [🗑] in the right part of the corresponding row. To delete multiple fields at the same time, select the corresponding rows by selecting their checkboxes and click the [🗑 Delete] button at the bottom of the screen. For more information about deleting system elements, see *Deleting an element* (page 29).

---

**Metadata search**

In the "Metadata" tab, you can use the search bar to search for additional fields by their names. In addition, there is a filter located under the search bar to search by criteria such as "Field Type", "Required Field" and "Visible".

### 2.3.3.3.6 "Base layers" tab

A base layer (basemap) is a layer that is the main or primary one in a particular map. Users visualize tasks, service objects, and thematic layers above the basemap, as well as use it for navigation through a map and getting general information about the area of interest. When switching to the "Base layers" tab, you can see a table of basemaps with the following columns (Fig. 2.96):

- "Code" – an identifier automatically created by the system when adding a new basemap.

- "Name" – the title of the basemap displayed in the system.

- "Main" – the marked basemap is displayed when opening the ActiveMap Web start window, as well as on maps in the ActiveMap system's mobile applications and in task creation.

- "Space image" – the marked basemap is interpreted as a satellite image (raster image).



Fig. 2.96: "Base layers" tab

### Adding a basemap

To add a new basemap, click ![+ Create] at the top of the window.
A window opens where you have to fill the following fields (Fig. 2.97):

- "Name" – name of the basemap displayed in the system.

- "Main base layer" toggle switch – makes basemap the main one,
  i.e. it is displayed when opening the start window of ActiveMap
  Web, on maps in the mobile applications of ActiveMap, and when
  creating tasks.

- "Space image" toggle switch – marks the basemap as a satellite
  image.

- "Class name" – selects the class that defines the mechanism for
  accessing the basemap.

- "Url/key" – depending on the selected class, it may contain a link to
  access basemap data or access keys (API key) for rights-protected
  third-party organizations.

- "Options" – other map parameters (for example, copyright, pos-
  sible display scales of the basemap, etc.). When listing multiple
  parameters separated by commas, they are enclosed in curly braces
  {}.

- "Custom class" – a map defined outside of existing map classes (for
  example, a combined map from several existing classes).

You can found examples of class names, keys, and parameters in the *At-
tachment 3. Examples of baselayer settings and parameters* (page 283)
section.

Fig. 2.97: Adding a basemap

### Editing a basemap

To edit the parameters of a basemap, click  in the right part of the row of the selected basemap. A form similar to the creation form opens, where you can modify the information about the base map.

### Deleting a basemap

To delete a basemap, click  in the right part of the corresponding row. To delete multiple basemaps simultaneously, select the corresponding rows and click the  button at the bottom of the screen. For more information about deleting system elements, see *Deleting an element* (page 29).

### Searching for a basemap

To search for a basemap, you can use the search bar to search by title. Under the search bar there is a filter, that helps to quickly find the base layer by selecting the "Main" option in the filter, or the satellite image by selecting the "Satellite" option. If you first select "Main" or "Satellite" and then uncheck the box next to the option, all base maps that are not the main base map or satellite image, respectively, are displayed in the results.

### 2.3.3.4 "Rights for layers" block

Access control is the process of granting users the necessary permissions to access system tables. Users can have the following rights: read, write, edit, and use the table objects as service objects.

The right to read allows seeing the layer in all programs within the system. The right to write allows adding new objects to the layer, editing and deleting the old ones. The right to edit allows changing the layer structure, styles, etc. The right to use table objects as service objects allows creating tasks with linkage to these objects with automatic filling of the task fields according to the configured mapping (matching a table attribute with a task field).

To search for rights, you can use the search string or the filter panel.

### 2.3.3.4.1 "Access control" tab

In the "Access control" tab, you can view the list of rights, available to the users of the system (Fig. 2.98). When switching to this tab, a table with the following columns becomes available:

- "User" – use name (login is specified in brackets).

- "Group" – group to which the layer belongs.

- "Layer" – name of the layer for which the access is defined.

- "Service Objects" – toggle switch for granting the right to use the table as a list of service objects. The same table can be used as a service by some users and as a normal by others.

- "Read" – toggle switch for granting the right to view the layer.

- "Write" – toggle switch for granting the right to edit the layer.

- "Edit" – toggle switch for granting the right to manage the layer.

Fig. 2.98: "Access control" tab

When enabling the toggle switch in the column header, the right to read/write/edit/use as service objects is granted to all users currently displayed in the administration area. Thus, you can use a filter to select users by any parameter and grant them all rights at the same time. Similarly, permissions can be revoked by enabling the toggle switch in the layer row for a specific layer, or by enabling the toggle switch in the column header for user categories. After enabling the toggle switch in the column header, the system asks to confirm the action.

To sort rows alphabetically by "User", "Group", or "Layer" parameters, click on the header of the corresponding column. To reverse the sorting direction, click on the header again.

The "New tasks", "Tasks in progress", and "Executed tasks" layer rows do not have the "Write" and "Edit" toggle switches. This is explained by the fact that the system enters objects into these layers on the basis of tasks, and users are not allowed to enter objects.

Clicking  opens a window for selecting a directory for saving the accesses.xlsx Excel file. The structure of this file repeats the structure of the table in the current tab. Field sorting is not preserved during export, but if you apply a filter before export, only selected rows are exported.

### 2.3.3.4.2 "Copy access" tab

In the "Copy access" tab, you can copy user rights (Fig. 2.99).



Fig. 2.99: "Copy access" tab

To do this, select a user for whom you want to copy the rights and click

Copy rights . Next select the user whose rights you want to copy
in the opened window (Fig. 2.100).
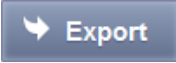


Fig. 2.100: User selection window for rights source

### 2.3.3.4.3  "Access for groups" tab

In the "Access for groups" tab, you can grant organizations the appropriate rights by enabling the "Access" toggle switch in the line of the desired group (Fig. 2.101). You can select a specific group, organization, user, or cluster group with a filter for convenience.



Fig. 2.101: "Access for groups" tab

You can save the table of organizations with the rights granted to them for groups to a computer. To do this, click ![Export]. A window for selecting a directory for saving the groups.xlsx Excel file opens.

### 2.3.3.4.4  "Access to tables" tab

In the "Access to tables" tab, you can grant users the rights to read, write, and edit tables (Fig. 2.102). You can filter a specific organization, user, user cluster, cluster of tables with data and select users with read, write, and edit rights.

Fig. 2.102: "Access to tables" tab

### 2.3.3.5 "Reports" block

In the "Reports" block, you can manage report templates JasperReports[1]
displayed in the system (Fig. 2.103).



Fig. 2.103: "Reports" block

---

[1] https://ru.wikipedia.org/wiki/JasperReports
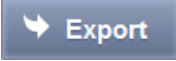
### 2.3.3.5.1 "List of reports" tab

In the "List of reports" tab, you can view all system reports, upload new ones, and edit/delete existing ones.

In this tab you can also change and save the order in which the reports are displayed in the system. Using the [➜ Export] button, you can export the Excel file jreports.xlsx containing information about the reports and the fields they contain.

### Uploading a report

To add a report to the system, click [➜ Upload] at the top of the window. A pop-up window opens to select a report from your computer. You can upload the files in jrxml format.

After successfully uploading the report to the system, the report creation window opens (Fig. 2.104). Fill in the following fields in the "Main" tab:

- "Name" – the name under which the report is displayed in the system.

- "Data storage" – selection from the list of available databases.

- "Preferred report format" – selection of the format in which the report is created by default (when creating the report, you can choose another format: PDF, Word, RTF).

- "Report description" – the description that accompanies the report in the system.

- "Report type" – selection of the report type: general, by tasks, or by the layer. A general report is generated for all types of work and tasks in the system. Report by tasks allows you to set a particular type of work. Report by the layer allows you to specify the thematic layer for which the report is generated.

- "Languages for which the report is available" – selection of the system language(s) for which the report is displayed when switched to. The list of available languages corresponds to those activated in the "System" block. If the language is not set, then the report is available for all languages. The System Administrator can see the entire list of reports for easy management in the "Administration" mode.

- "System Report" toggle switch – marks the system reports.

- Report parameters:

- "Parameter name" – name of the parameter;

- "Mandatory" – toggle switch for the parameter that is mandatory to fill in for report generation;

- "System" – toggle switch for internal use (the parameter is not displayed in the generated report);

- "Combine with next in the period" – toggle switch combining parameters in the "from-to" period and automatically filling in parameters with a date type when selecting a standard period (today, yesterday, this week, etc.);

- "Parameter description" – description of the properties and usage of the parameter;

- group of parameters: "Start date", "Sign", "Days", "Hours", "Minutes and seconds" – creating a default date displayed when filling in report parameters (it can be changed).
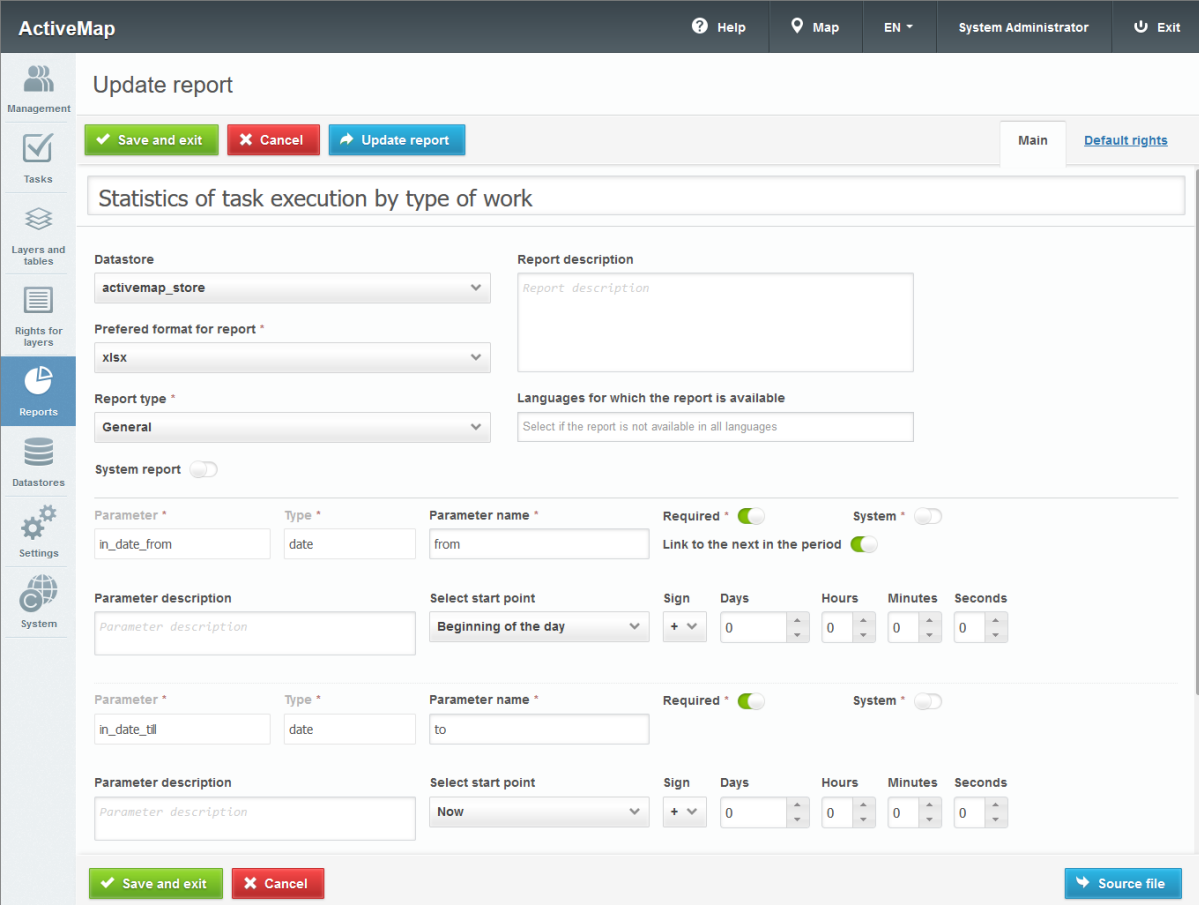


Fig. 2.104: Creating a report

In the "Default rights" tab you can set the rights to view/manage the report. The same logic is implemented here as for granting default rights to layers (for more information, see *Creating a new layer* (page 70)).

### Editing a report

Clicking ![pencil icon] (or double-clicking on a row with a selected report) opens a form similar to the creation form, where you can fill in/edit the fields with information about the selected report (Fig. 2.105). The following buttons become available while editing:

**→ Update report** – reload the original jrxml report file.

**↪ Source file** – export the created report template in jrxml format.



Fig. 2.105: Editing a report

---

**Deleting a report**

To delete a report, click [trash icon] on the right side of the corresponding row. To delete several reports at the same time, check the corresponding lines and click [Delete button] at the bottom of the screen. See *Deleting an element* (page 29) for more information about deleting system items.
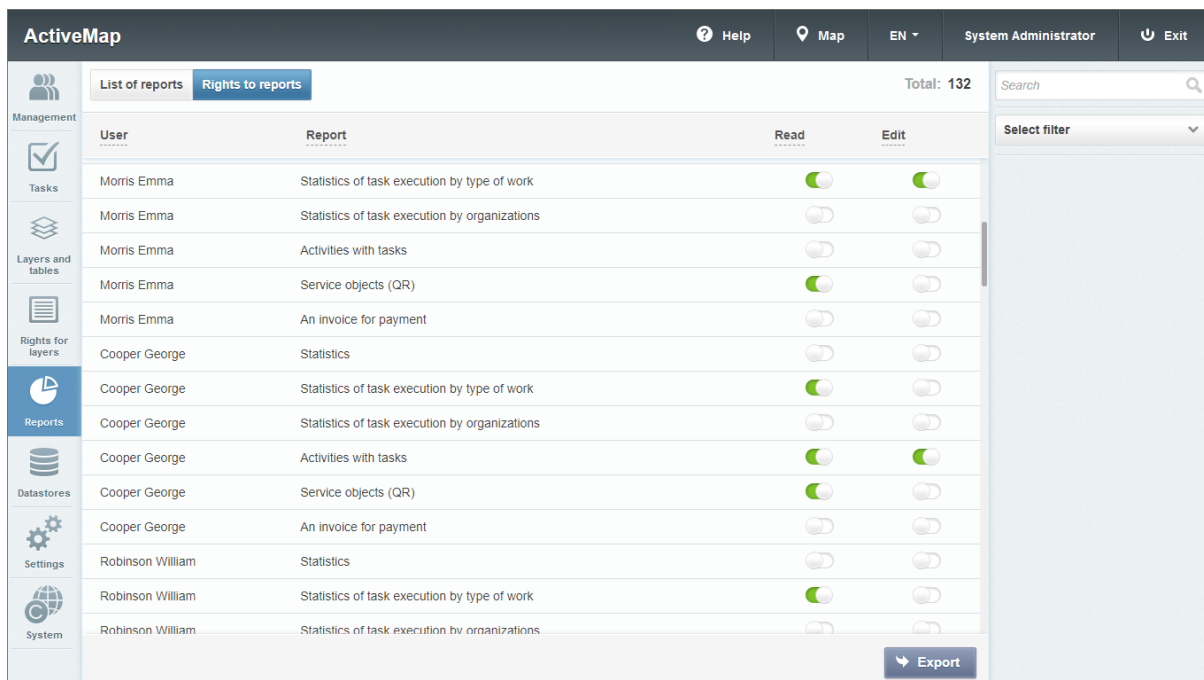
**Searching for a report**

In the "Report List" tab, you can use the search bar to search for reports by their names.

### 2.3.3.5.2 "Rights to reports" tab

When switching to the "Rights to reports" tab in the administration area, the following columns appear:

- "User" – user's full name;

- "Report" – name of the report;

- "Read" – toggle switch indicating whether this report is available to the selected user for viewing in the information window;

- "Edit" – toggle switch indicating whether this report is available to the selected user for management.



Fig. 2.106: "Rights to reports" tab
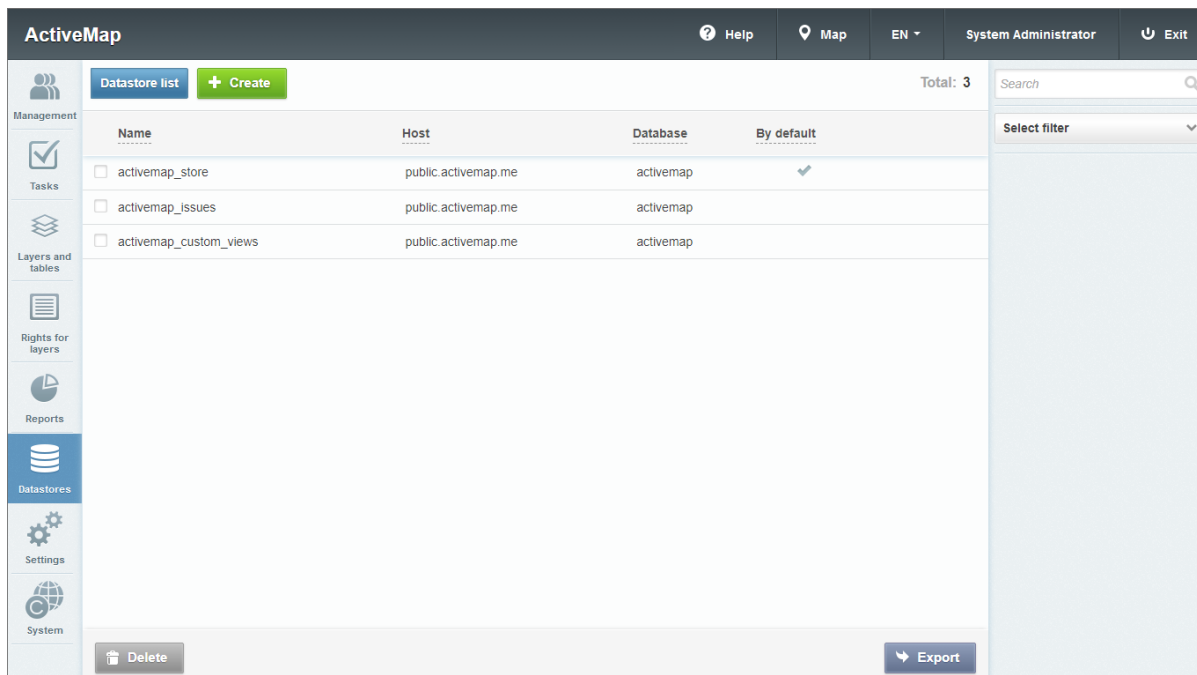
Using the [**Export**] button, you can export an Excel file accesses.xlsx containing information about users and their rights to reports.

### Search for rights to reports

When switching to the "Rights to reports" tab, you can use the search bar to search for reports by their names and a filter selected from the drop-down list for searching by the user's full name, report name, main organization, user cluster, role, and rights to view and delete reports.

### 2.3.3.6 "Datastores" block
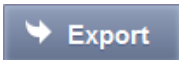
In the "Datastores" block, you can view detailed information about databases presented in a table with the following columns: name, host, database, by default (Fig. 2.107).



Fig. 2.107: "Datastores" block
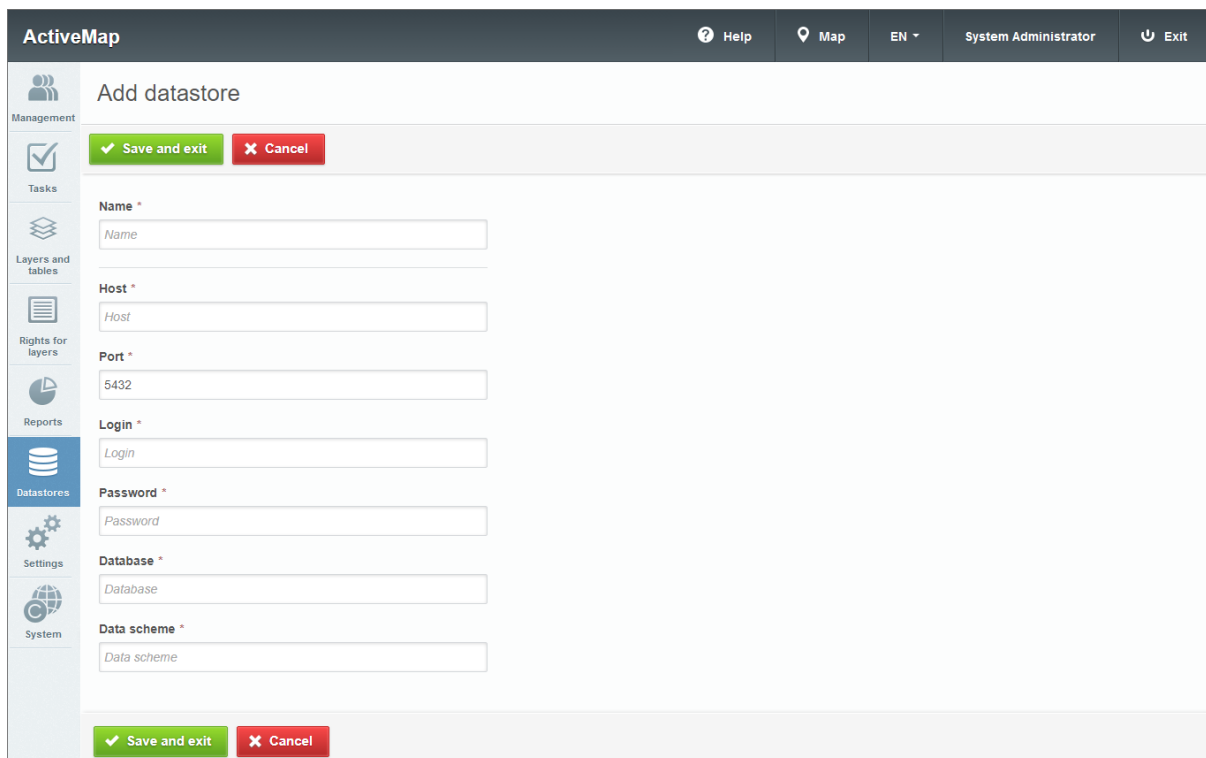
A mark next to the storage name in the "Default" box means that when you create or load a layer (report), the system automatically moves the layer (report) to the default storage (when you do not select a specific data store from the presented list).

To export data about datastores, click [**Export**]. A window for selecting a directory opens to save the "datastores.xlsx" Excel file on the computer. The exported table contains the following data: name, host, port, datastore, and data schema.

### 2.3.3.6.1 Adding a new datastore

To add a new datastore to the system, click ![Create] at the top of the window. Fill in the following fields in the opoened window (Fig. 2.108):

- "Name" – datastore name;

- "Host" – server address;

- "Port" – port for database connection;

- "Login" – login for access to the datastore;

- "Password" – password for access to the datastore;

- "Database" – name of the database in which the datastore is created;

- "Data scheme" – element of the data storage structure.



Fig. 2.108: Adding a new datastore

---

### 2.3.3.6.2 Editing a datastore

To edit datastore information, click  (or double-click the row with the selected data store). A form appears in the administration area with the same fields as in the add form. Here you can fill in/change the fields of interest with information about the selected datastore. The administrator has access to all fields for editing, except for the "Name" field.

### 2.3.3.6.3 Deleting a datastore

To delete data store, click  on the right side of the corresponding row. To delete several data stores at once, check the corresponding rows and click  at the bottom of the screen. For more information about deleting system elements, see *Deleting an element* (page 29).

### 2.3.3.6.4 Layer publishing

To publish a layer in the system, hover over the row with the datastore name and click the "Available tables" button . A new window opens where you can view the full list of available datastore layers and their geometry types (Fig. 2.109).

In addition, you can view which layers are available for publishing and which ones are already published. For published layers, the "Published" column has a checkmark, and to the right of the names of unpublished layers, there is a "Publish" button.
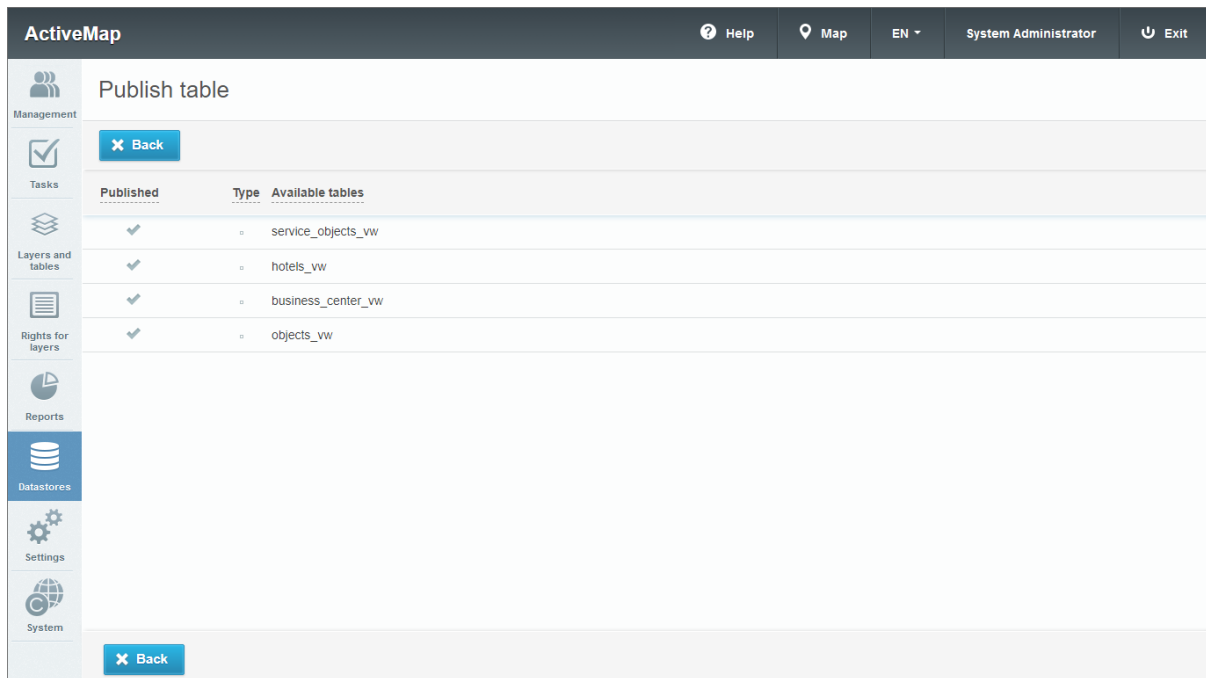
Fig. 2.109: Available tables in the datastore

Clicking "Publish" takes you to the window for editing the selected layer. In this window, you have to select a group for publishing the layer, define its style, change its name, and, if necessary, add new attributes (in the "Attributes" tab) and additional fields (in the "Additional" tab).

To publish a layer in the system, fill in the required input fields and click ✔ Save and exit. To cancel publication, click ✘ Cancel.
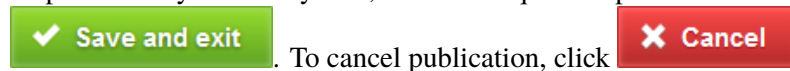
### 2.3.3.6.5 Searching for datastores

In the "Datastores" block, you can use the search bar to search for a datastore by its name, database name, or host name. In addition, you can search using the "From infrastructure" filter, which allows users to find datastores that work with the MapEditor program.

### 2.3.3.7 "Settings" block

> **Danger:** Developers verified the values of the existing settings during debugging. Changes in this block may cause the system to malfunction or stop working.

In the "Settings" block, you can configure various elements of the system (Fig. 2.110).
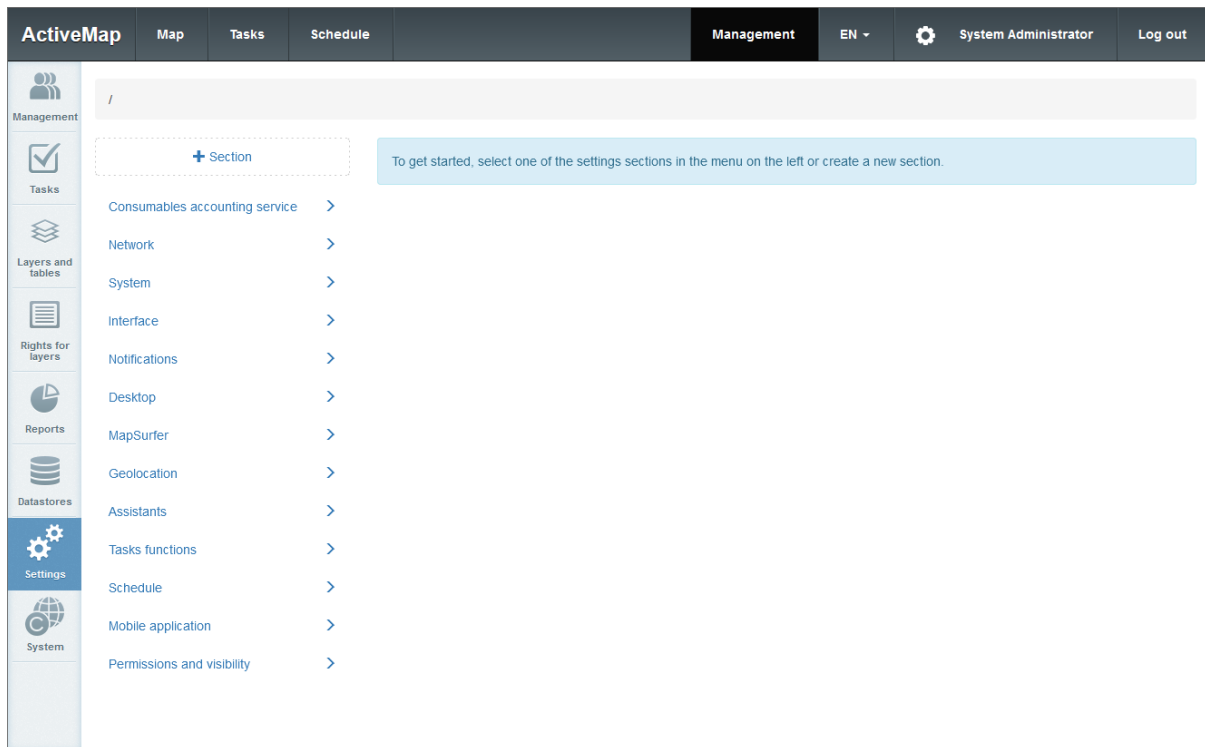
Fig. 2.110: "Settings" block

They are divided into thematic sections. Each section contains folders and subfolders with settings and their values (Fig. 2.111).
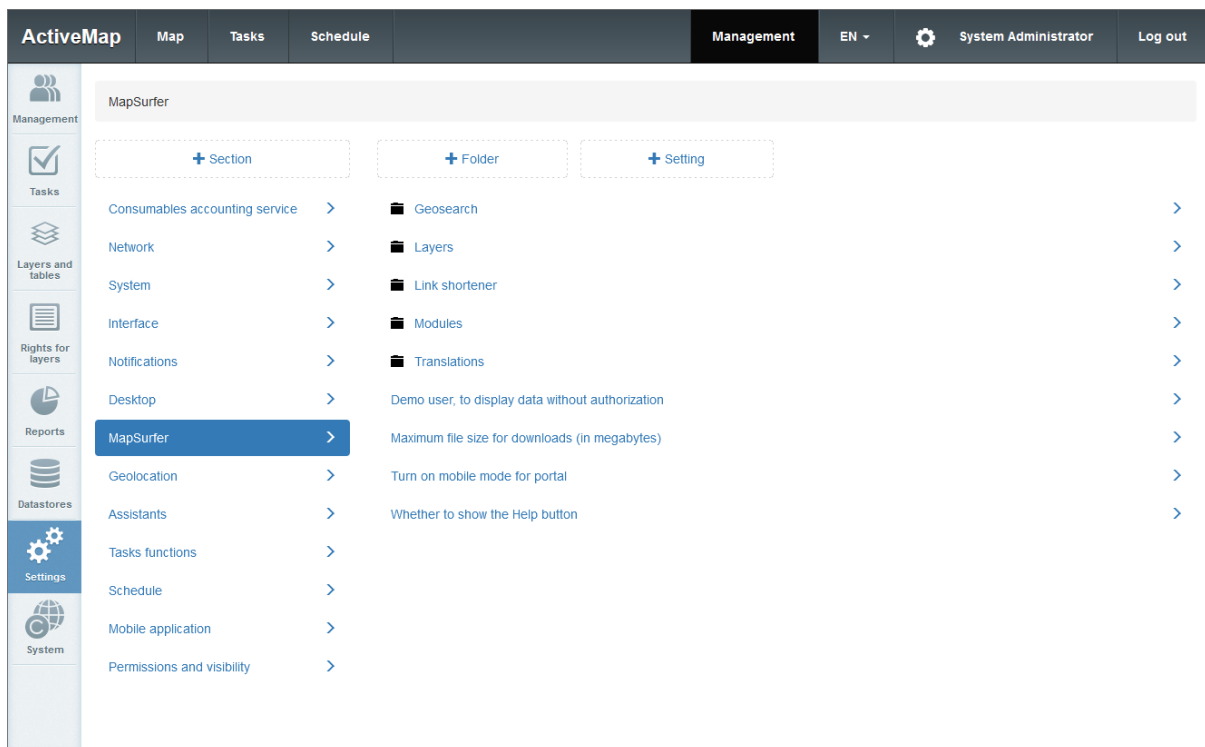


Fig. 2.111: Structure of the "Settings" block

Editing and deleting the default settings values is not available.

---

**2.3. Management module**                                                113

To add a new setting to an element of the system, select the appropriate section, then click the "+ Settings" button. A form opens with fields to fill in (Fig. 2.112):

- "Key" – name in the system in Latin alphabet characters;

- "Name" – name in the interface;

- "Type" – data type (string, integer number, logical value, or real number).
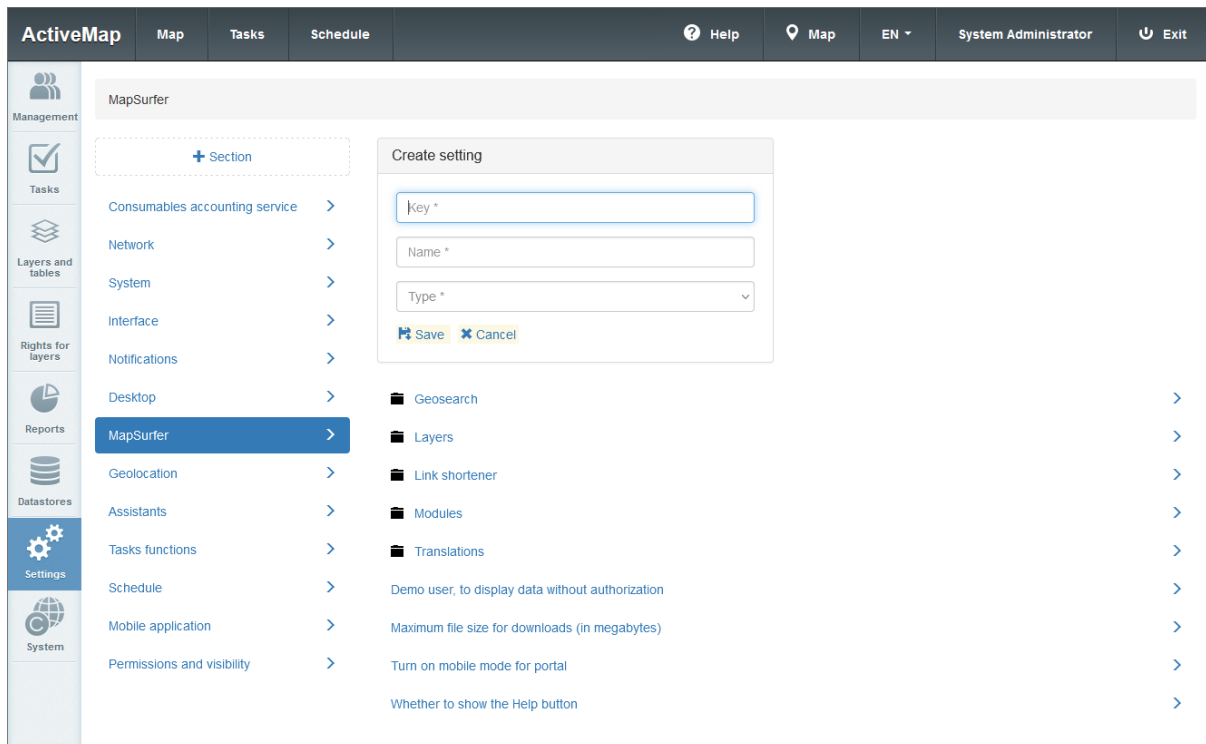


Fig. 2.112: Creating the setting

To set a new value for the setting, select the setting and click "+Value". A window opens with fields to be filled in (Fig. 2.113):

- "Value" – value the setting is equal to in the system (depends on the data type specified when creating the setting);

- "Organization" – organizations to which this setting is applied;

- "User" – users to whom the setting is applied;

- "Role" – user roles to which the setting is applied;

- "State" – statuses of the task to which the setting is applied.
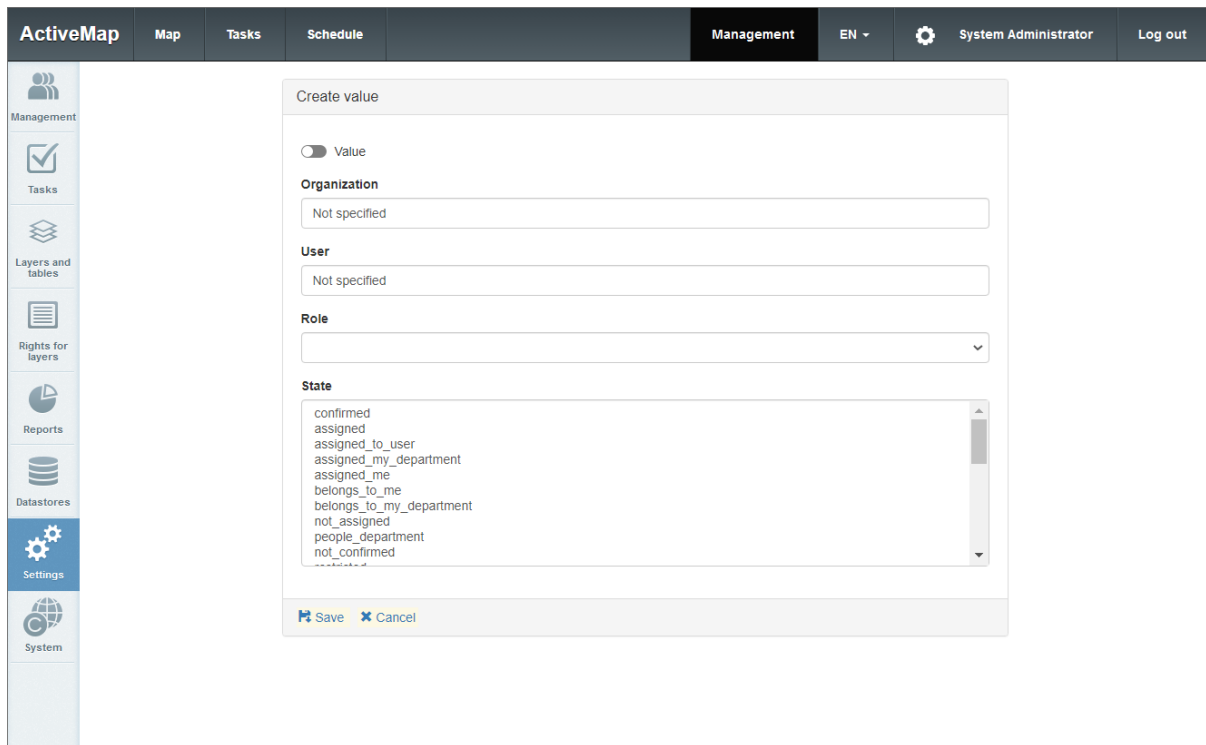
Fig. 2.113: Adding a value

> **Attention:** The setting can be specified either for a particular user or for a role in the system. If the "Organization", "User", and "Role" fields are not filled in, the setting is available to all users of the system.

To edit the added value, click ![pencil icon]. Make the changes in the opened window and click "Save". To delete the added value, click ![trash icon]. Confirm or cancel the deletion in the dialog box.

### 2.3.3.7.1 "Consumables accounting service" section

To use the method of calculating the cost of work online, configure the "Invoice" module in the application. This section allows users to add settings to this module (Fig. 2.114). You can set a new value by selecting the setting, then clicking "+ Value". In the window that opens, enter the required name and fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
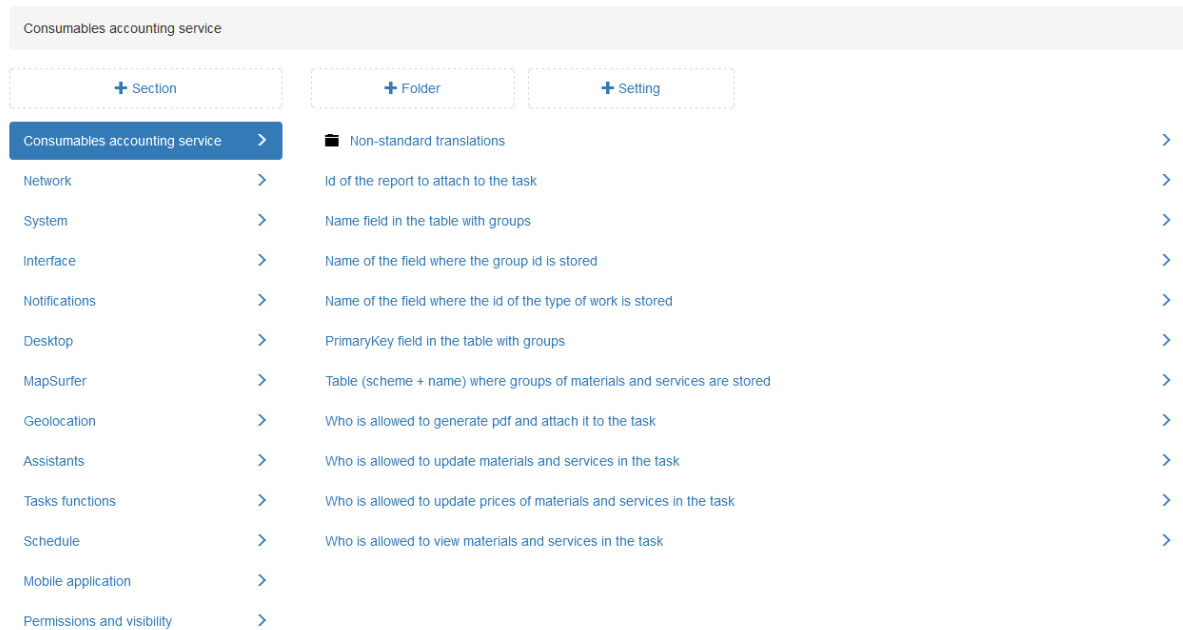
Fig. 2.114: "Consumables accounting service" section

### Non-standard translations

In this folder, you can rename fields of the module. Select a language from the list presented, then click "+ Setting". Enter the appropriate values in the opened window (Fig. 2.115):

- "Key" – key in the system to be renamed;

- "Name" – name of the field to be renamed;

- "Type" – data type (in this case, a string).



Fig. 2.115: Changing the field name

Next, set a new value for this setting (Fig. 2.116). Restart the service to apply the new value.
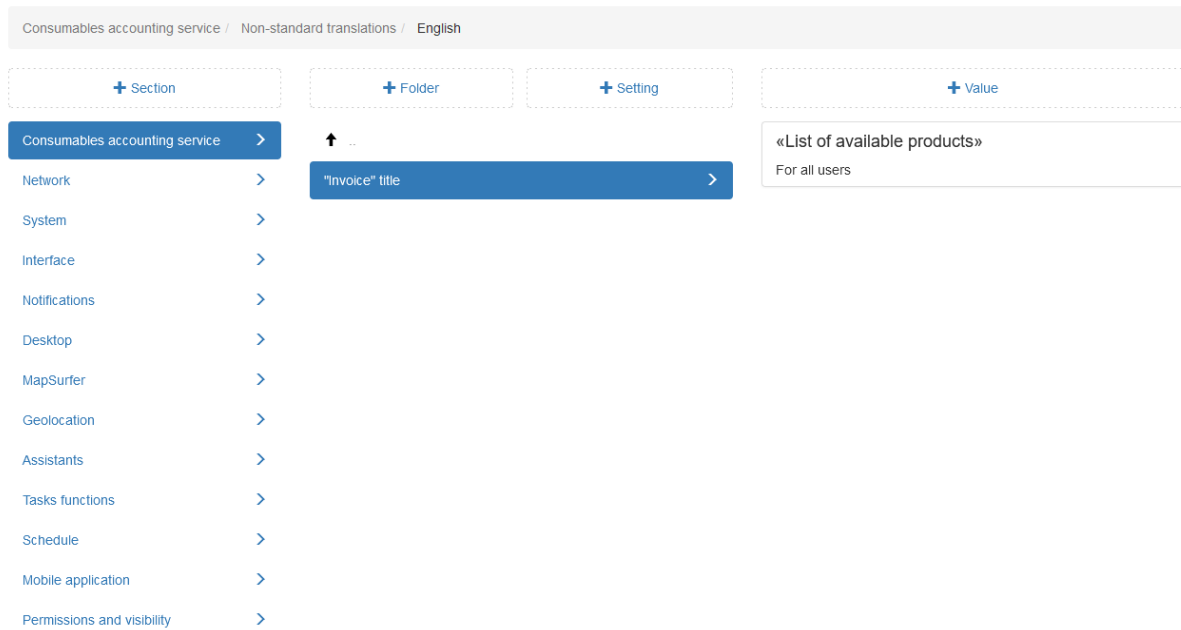
Fig. 2.116: Adding a new value

### Id of the report to attach to the task

This setting displays the ID of an existing report in the system, which is loaded when an invoice is generated in the task card in PDF format. By default, the system uses a report with ID 343 (Fig. 2.117). Restart the service to apply the new value.
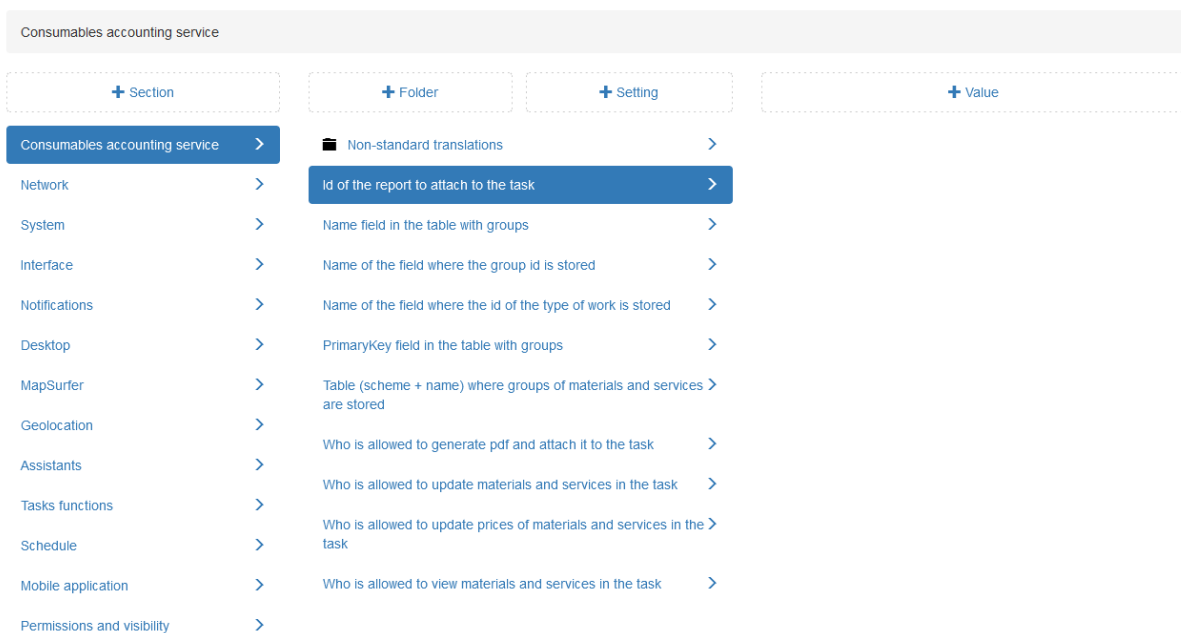


Fig. 2.117: Specifying the report ID

### Name of the field where the group id is stored

To add a filter by material and service groups in the ActiveMap Mobile application, create a table with material and service groups, for example, "Group of materials". Then add this reference table (dictionary) to "Materials and services" table. The field should have **sys_group_id** database name (Fig. 2.118), as it is recognized in the database by default.



Fig. 2.118: The name of the field where the group ID is stored

Next, fill in the **Table (scheme + name) where groups of materials and services are stored** setting, specifying where the group of materials and services is stored (Fig. 2.119).

Fig. 2.119: Table where material and service groups are stored

In the **PrimaryKey field in the table with groups**, add a new value, for example gid (Fig. 2.120).



Fig. 2.120: PrimaryKey field in the table with groups

In the **Name field in the table with groups** setting, specify the name of the field where the group names from the material group table are stored (Fig. 2.121).

Fig. 2.121: Name field in the table with groups

Next go to the MapEditor or ActiveMap Desktop desktop application and fill in the newly created reference table, adding the appropriate groups, and then the "Materials and services" table with the appropriate values. Restart the service to apply the new setting.

### Name of the field where the id of the type of work is stored

To add a filter by types of work in the "Invoice" module of the ActiveMap Mobile application, define the name of the attribute in the table with materials and services where the work type ID is stored. The default name is **sys_type_id** (Fig. 2.122). Next, add the corresponding attribute to the table with materials and services ("Integer" field type). Then open the table management in the MapEditor desktop application and select the "Materials and services" table. Fill in the **sys_type_id** field by specifying the IDs of the necessary work types. Restart the service to apply the new value.

Fig. 2.122: Name of the field where the id of the work type is stored

## Who is allowed to generate pdf and attach it to the task

This setting allows users to generate a PDF invoice and attach it to the task (Fig. 2.123). However, the specified user must first be granted rights to the corresponding report. Otherwise, it is not possible to generate an invoice.



Fig. 2.123: Providing access to generate an invoice

### Who is allowed to update materials and services in the task

This setting allows you to grant the right to edit an invoice (add materials, change the quantity) and save the changes made within the task. However, this setting does not allow you to generate an invoice within the task (Fig. 2.124).



Fig. 2.124: Granting access to edit an invoice

### Who is allowed to update prices of materials and services in the task

This setting allows you to specify the roles or users who can change prices of materials and services in the invoice within the current task (Fig. 2.125).
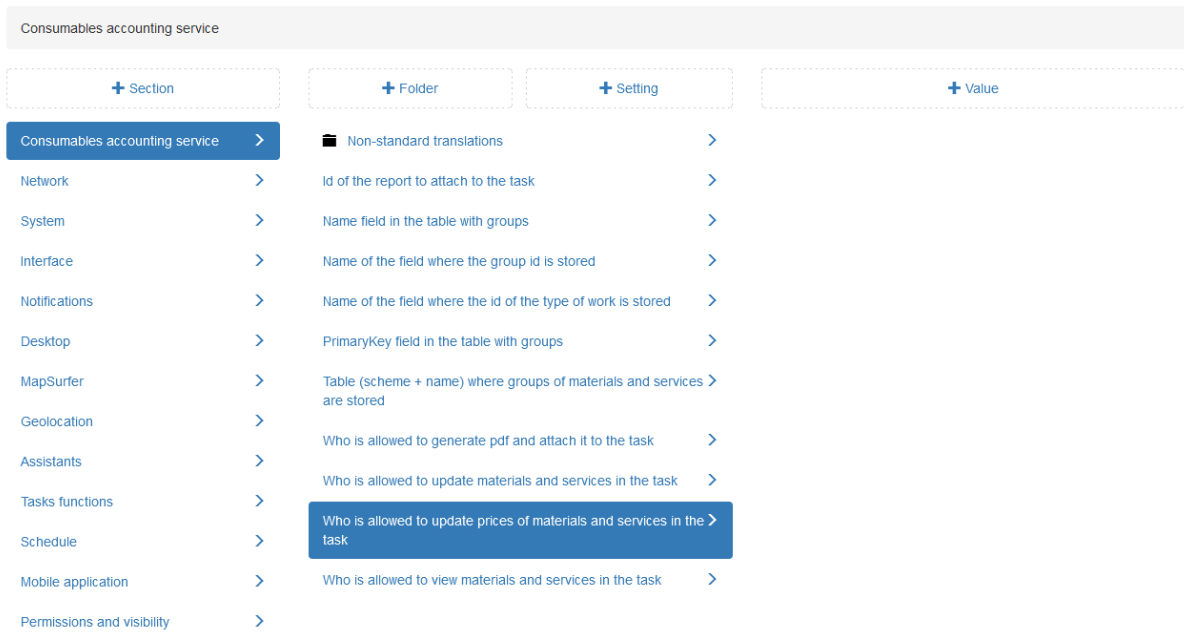
Fig. 2.125: Granting access to edit the invoice

### Who is allowed to view materials and services in the task

This setting allows you to specify users who can view an invoice generated by another user (:Fig. 2.126).
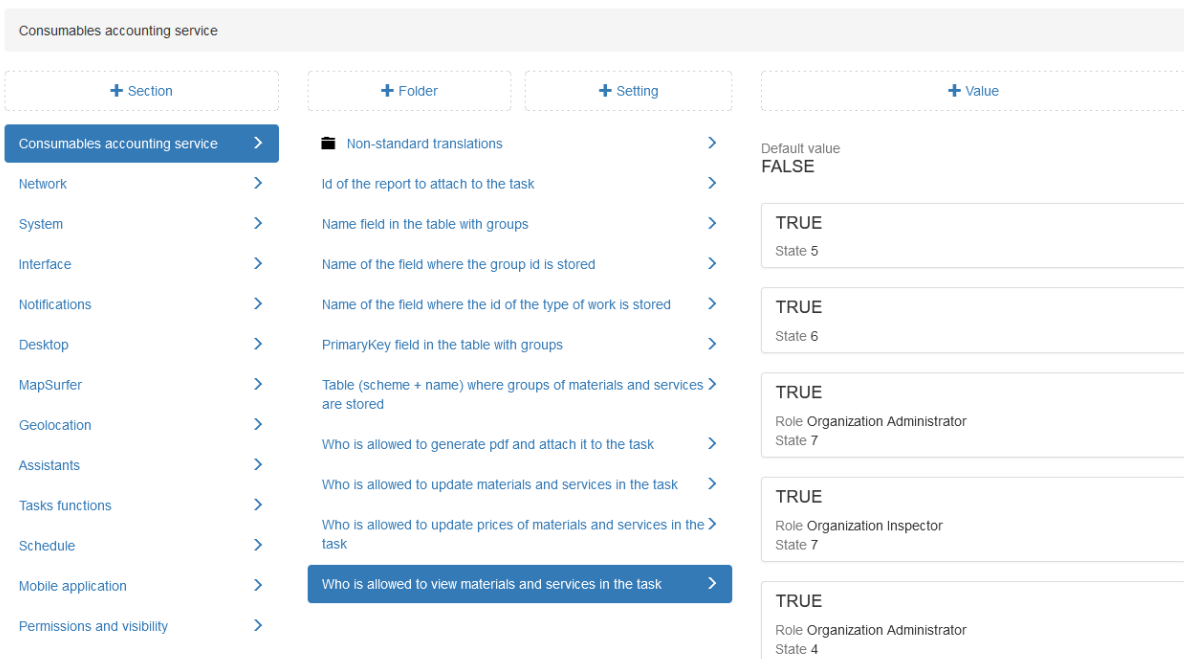


Fig. 2.126: Providing access to view an invoice

### 2.3.3.7.2 "Network" section

In this section, you can add a public server that is automatically displayed when logging into the mobile application. In this case, you do not need to enter the address manually. Click on the server and proceed to enter the login and password (Fig. 2.127). If necessary, you can set a new value by selecting the setting and then clicking "+Value". In the opened window, turn on/off the toggle switch or enter the required name, and then fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
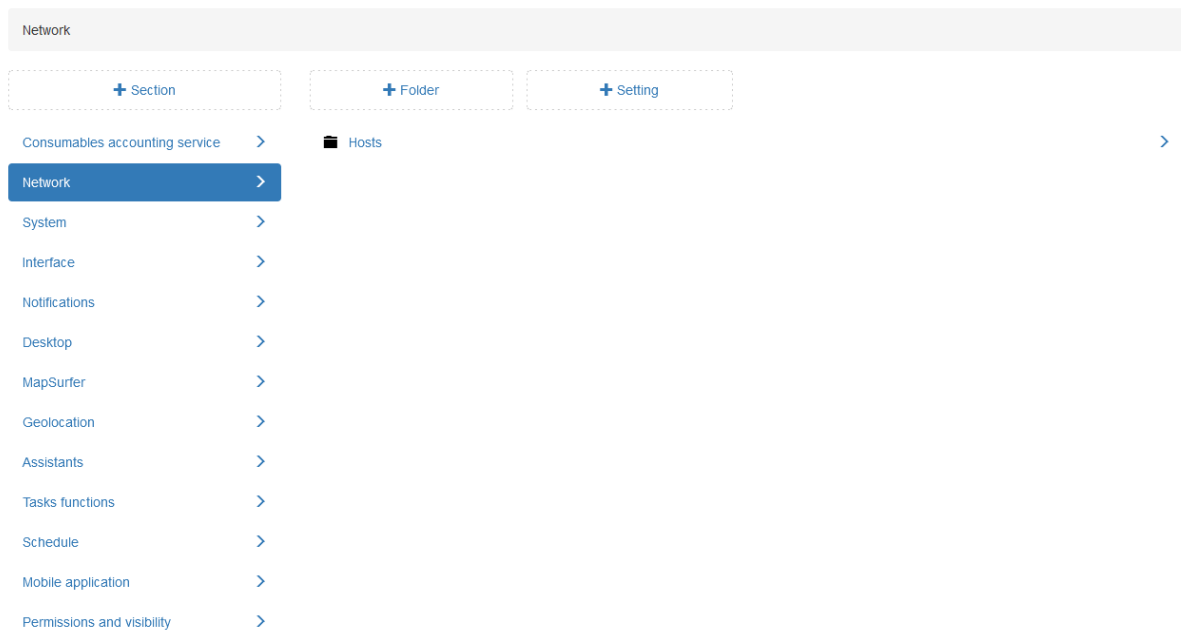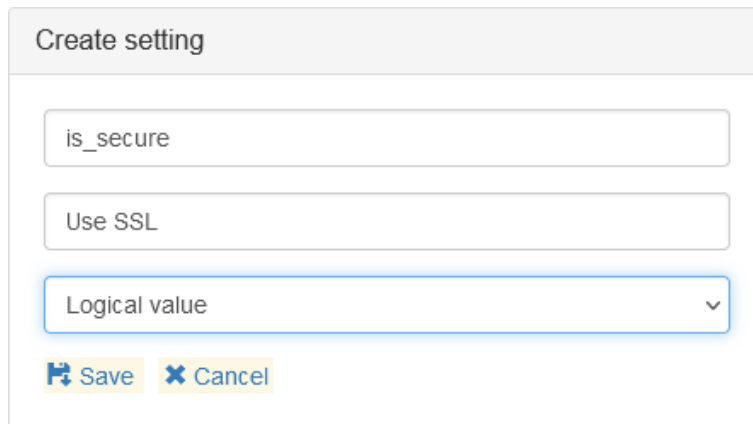
Fig. 2.127: "Network" section

**Attention:** The setting works only if the server address is registered in the mobile application.

### Hosts

To add a public server, add the following settings in this folder by clicking "+ Setting" and entering the corresponding names:
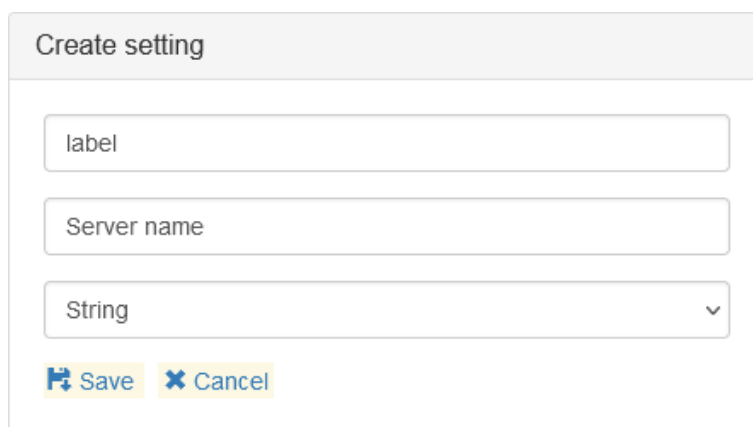
- Use SSL (key – is_secure, data type – boolean) (Fig. 2.128);

- Server name (key – label, data type string) (Fig. 2.129);

- Host (key – host, data – type string) (Fig. 2.130).
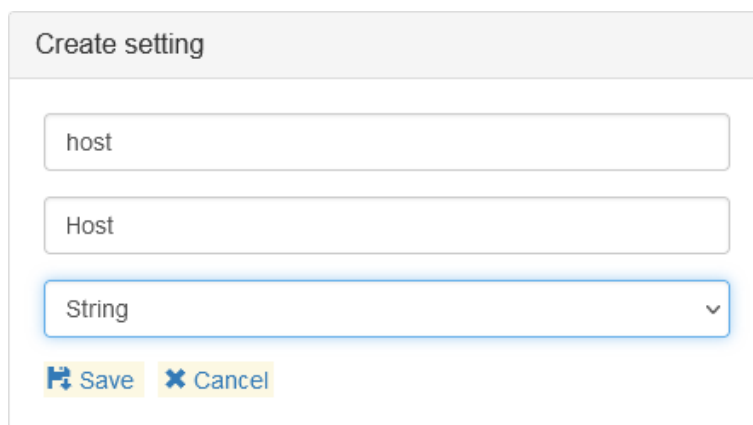
Fig. 2.128: Use SSL



Fig. 2.129: Server name



Fig. 2.130: Host

Next, set the corresponding value for each setting by selecting it from the list (Fig. 2.131).
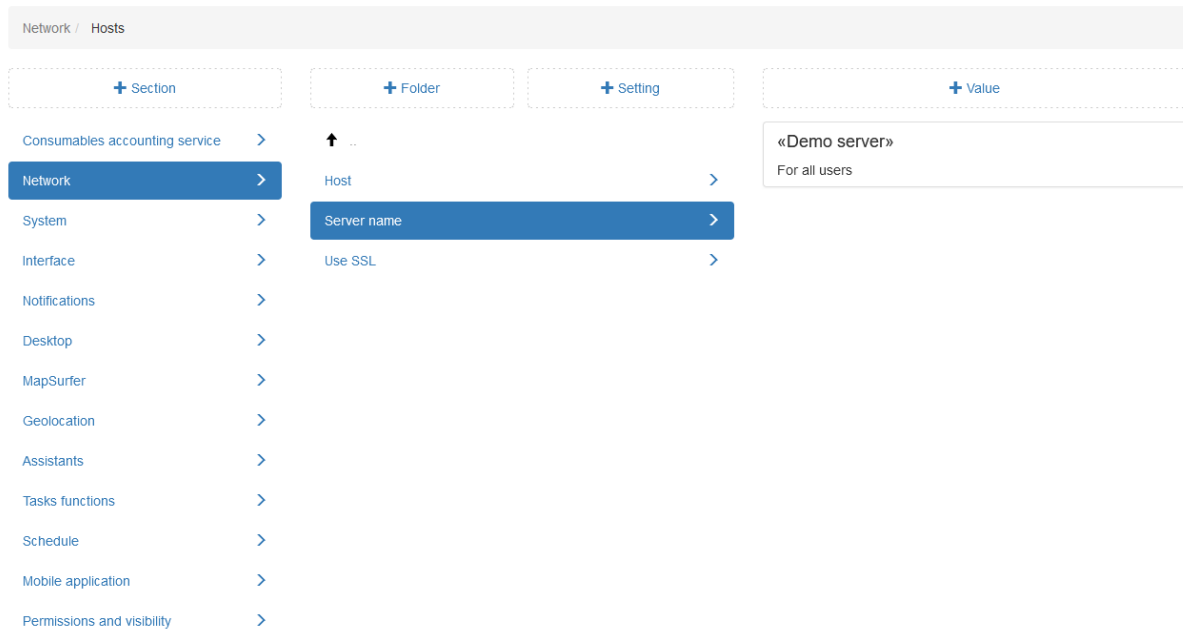
Fig. 2.131: Name of the demo server

### 2.3.3.7.3 "System" section

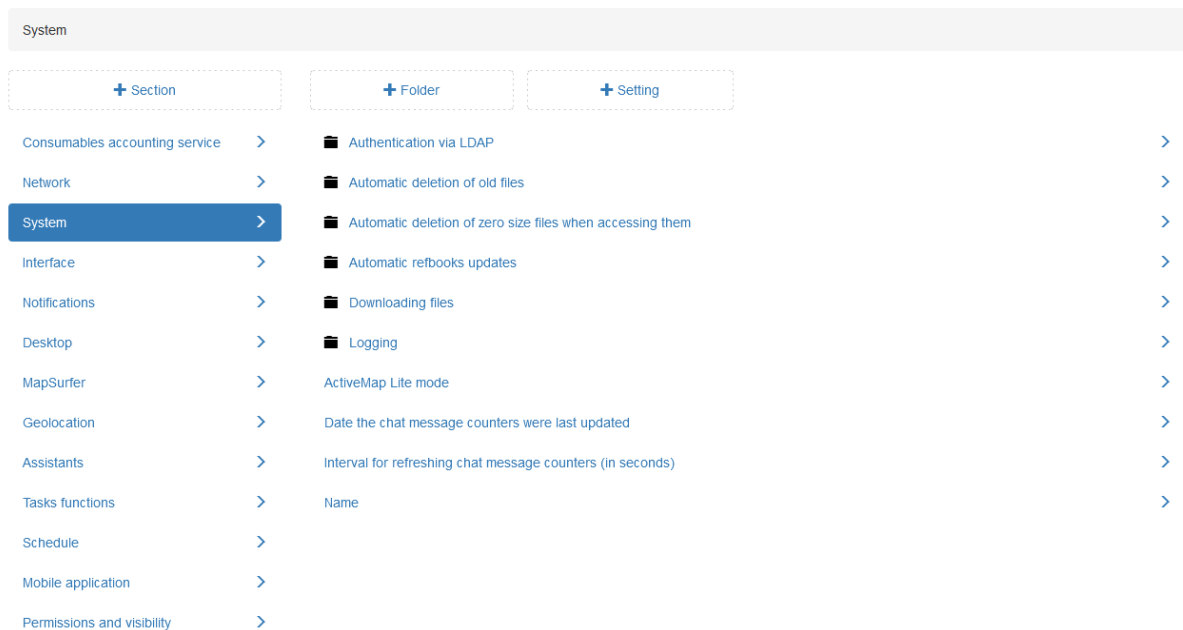This section contains additional settings for the Cerebellum (Fig. 2.132).



Fig. 2.132: "System" section

## Authentication via LDAP

The system has a setting for integration with LDAP. LDAP is a single authorization system used by all software products in an organization. By default, LDAP is disabled.

To connect user authorization via LDAP, set the appropriate values in the **"Host"** and **"Port"** settings. Next, fill in **"Administrator login"** and **"Administrator password"** by entering the LDAP user data under which authorization and search of other users is performed (Fig. 2.133).
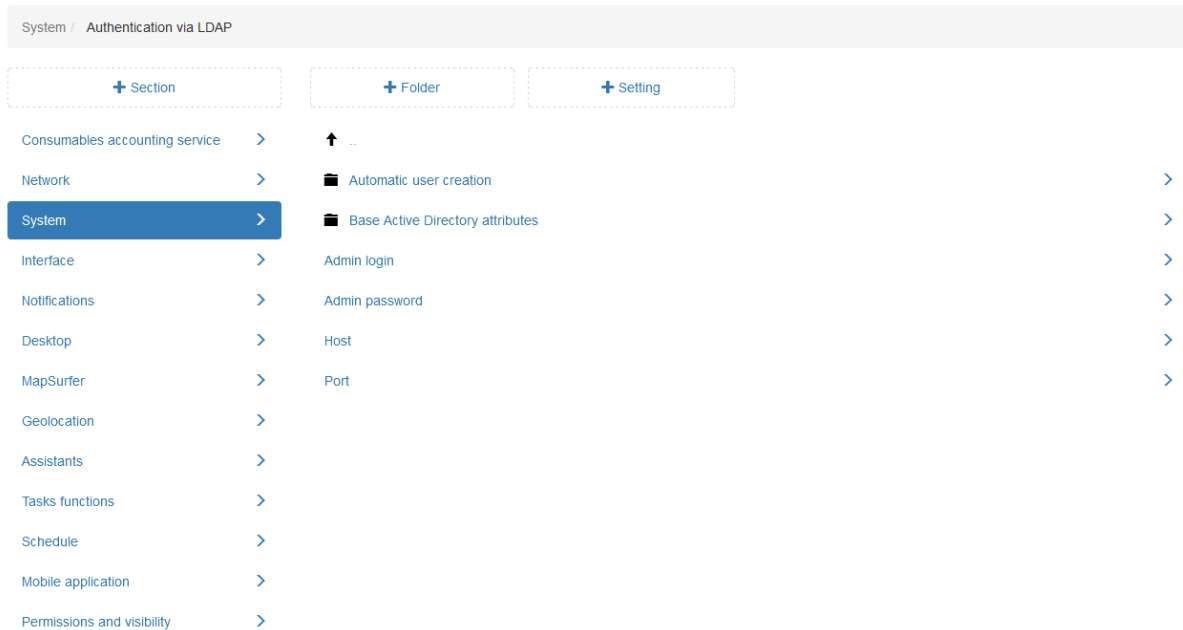


Fig. 2.133: Authentication via LDAP

## Automatic user creation

The system has a setting for automatic creation of a new user if this user account exists in the Microsoft Active Directory. Users are created in one organization with the role of "Executor". Later, the administrator can transfer the data of these users to the necessary organization and change their role in the system (Fig. 2.134).
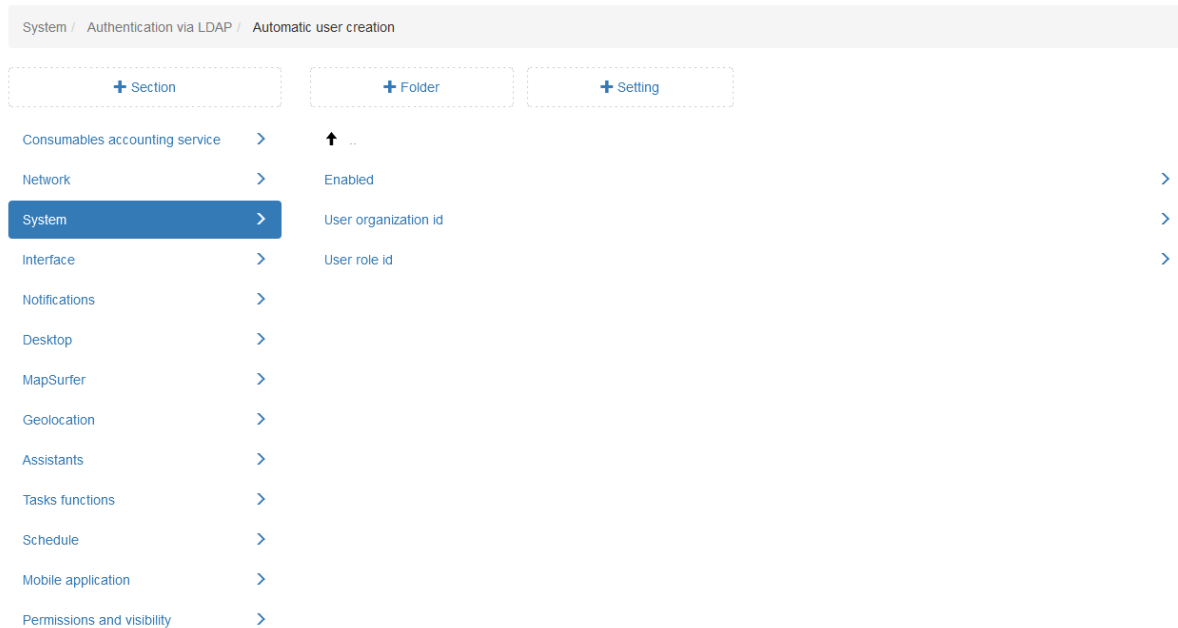
Fig. 2.134: Automatic user creation

## Base Active Directory attributes

In this folder, the administrator should register the correspondence of the standard fields of the ActiveMap system user and LDAP fields (Fig. 2.135).



Fig. 2.135: Base Active DireCtory attributes

## Automatic deletion of old files

The system has a limit on the maximum total volume of task files. This includes all files, for which there are records in the database, including logically deleted files, template task files, and others. Cerebellum receives information from the database about files to be deleted and cleans them at the specified time, if necessary. Files to be deleted are stored in the following order:

- Deleted files,

- Files of deleted tasks,

- Files of tasks not in the "In Progress" step.

Cerebellum periodically checks whether the volume of files exceeds the limit on the total volume. If it exceeds the limit, files are retrieved for deletion in the specified order. This setting is disabled by default (Fig. 2.136).



Fig. 2.136: Automatic deletion of old files

**Allow to delete files in tasks only with the following priorities**

In this setting, you can specify the IDs of the task priorities where files should be deleted (Fig. 2.137). You can specify priorities using identifiers (IDs) separated by commas.

Fig. 2.137: Allow to delete files in tasks only with the following priorities

**Enable downloading meta-information about files**

This setting enables downloading metadata about files that are to be deleted. Metadata can include various file attributes such as name, size, creation date, etc. By default, this setting is disabled (Fig. 2.138).



Fig. 2.138: Enable loading file metadata

**Limit the number of files in the queue**

Here you can set the maximum number of files that can be queued for deletion. When files are added to the queue, the auto-deleter gradually deletes them according to the set parameters. The default is 10000 files (Fig. 2.139).



Fig. 2.139: Limit on the number of files in the queue

**Maximum number of files to delete in one launch (7:00 – 23:00)**

The setting specifies a limit on the number of files that can be deleted at one time in the specified time interval. For example, if the value is set to 50, the auto-deleter deletes no more than 50 files within the specified time period. The default is 1 file (Fig. 2.140).

Fig. 2.140: Maximum number of files to delete in one launch (7:00 – 23:00)

**Maximum number of files to delete in one launch (23:00 – 7:00)**

The setting specifies a limit on the number of files that can be deleted at one time during the specified time interval. For example, if the value is set to 50, the auto-deleter deletes no more than 50 files within the specified time period. The default is 1 file (Fig. 2.141).



Fig. 2.141: Maximum number of files to delete in one launch (23:00 – 7:00)

**Number of files per meta-information loading launch**

This setting determines the number of files for which metadata is requested in each upload cycle. For example, if set to 100, the auto-deleter requests metadata about 100 files per operation. The default is 1000 files (Fig. 2.142).



Fig. 2.142: Number of files per meta-information loading launch

**Delete files only older than (sec)**

The setting specifies the storage time for task files (in seconds). If files have been stored for longer than the specified time, they are automatically deleted. The default is 180 days (Fig. 2.143).

Fig. 2.143: Delete files only older than (sec)

**Delete videos only older than (sec)**

The setting specifies the storage time for the task video files (in seconds). If a video has been stored for longer than the specified time, it is automatically deleted. The default setting is 45 days (Fig. 2.144).



Fig. 2.144: Delete videos only older than (sec)

**Period between downloads of meta-information about files (sec)**

This setting determines the time interval (in seconds) after which the auto-deleter downloads file meta-information. For example, if the value is set to 3600, then the meta-information is loaded every hour. The default is 300 seconds (Fig. 2.145).



Fig. 2.145: Period between downloads of meta-information about files

**Period between launches**

The setting determines the time interval (in seconds) between successive launches of the auto-deleter. For example, if the value is set to 3600, the auto-deleter runs every hour. The default is 1 second (Fig. 2.146).

Fig. 2.146: Period between launches

### Automatic deletion of zero size files when accessing them

When a file is deleted, information about it may remain in the database. Storing such information increases the storage space used. Use this setting to delete such empty records. The setting is disabled by default (Fig. 2.147).

Fig. 2.147: Automatic deletion of zero size files when accessing them

## Automatic refbooks update

This folder contains system settings responsible for updating data in reference tables and types of work (Fig. 2.148).



Fig. 2.148: Automatic updating of reference tables

**Delay in changing the version of refbooks, seconds**

When adding a value to a reference table (work type, priority, custom field, etc.), the system can automatically update data in the applications. This setting specifies, how often (in seconds) the system sends data to all applications (Fig. 2.149). In this case, the user does not need to update the data manually to get the actual values.



Fig. 2.149: Delay in changing the version of reference tables, seconds

## Downloading files

When connecting an external server for data storage, you can specify the storage location for each directory in this setting. After that, this directory is created on the server, and the files are stored in the specified location (Fig. 2.150).
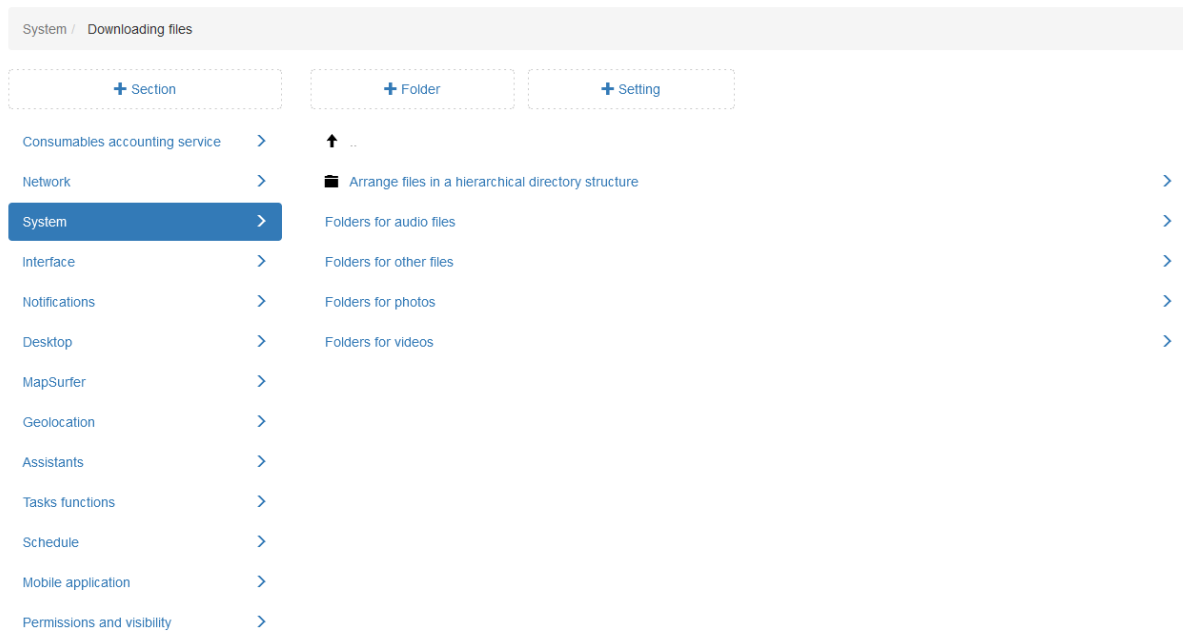
Fig. 2.150: Downloading files

### Logging

The setting allows you to automatically record in chronological order the events that occur with the task (creation, commenting). By default, this setting is enabled (Fig. 2.151).
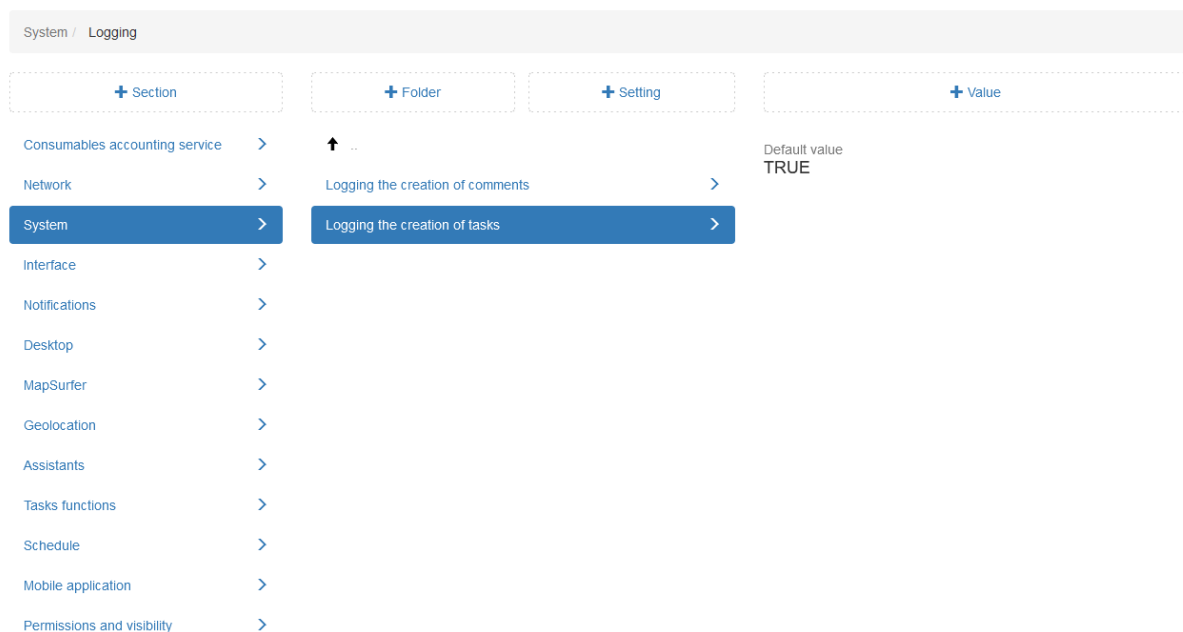


Fig. 2.151: Logging

### ActiveMap Lite mode

This setting allows using the system without providing a dedicated server for each organization. By default, this setting is disabled.

### Date the chat message counters were last updated

Engineering setting. This value is for developer use only during debugging.

### Interval for refreshing chat message counters

Engineering setting. This value is for developer use only during debugging.

#### 2.3.3.7.4 "Interface" section

This section allows you to configure the display of task ratings (Fig. 2.152). Users set ratings in the ActiveMap Informer mobile application. You can also view them in the ActiveMap Desktop. If necessary, you can set a new value for each setting in this folder by selecting it and clicking "+ Value". Turn on/off the toggle switch or enter a new value and then fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
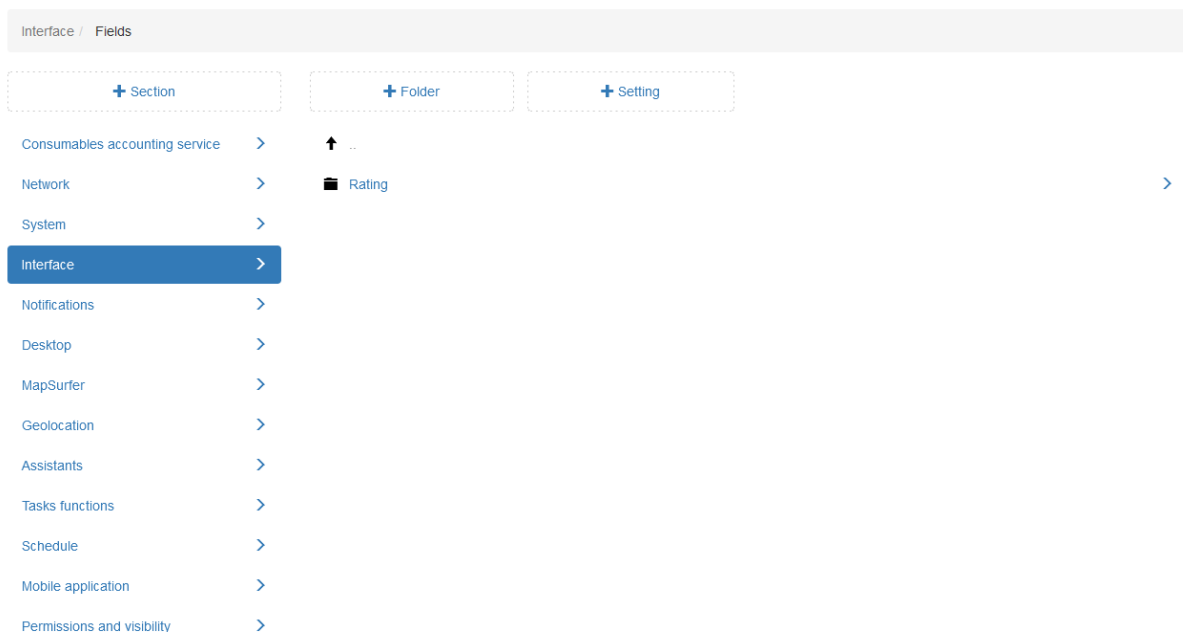
Fig. 2.152: "Interface" section

### Fields/Rating

**Scale [3-5]**

The system allows you to set a scale (from 3 to 5) for rating completed tasks. By default, a 4-point system is set to display ratings in the ActiveMap Desktop (Fig. 2.153).
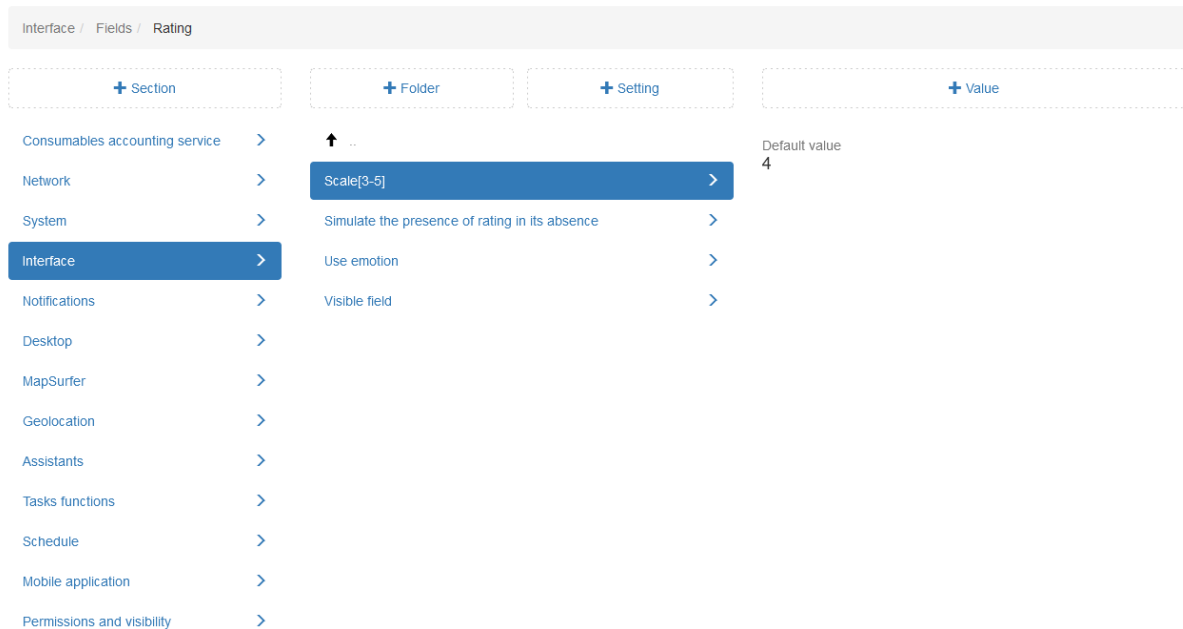


Fig. 2.153: Scale [3-5]

**Simulate the presence of rating in its absence**

This setting is a system setting that allows you to avoid errors when there is no set rating in the task.

**Use emotion**

This setting allows using emojis to display ratings in the ActiveMap Desktop. By default, this setting is enabled (Fig. 2.154). With the toggle turned off, users see the scores in the ActiveMap Desktop in the form of stars, not emoji.
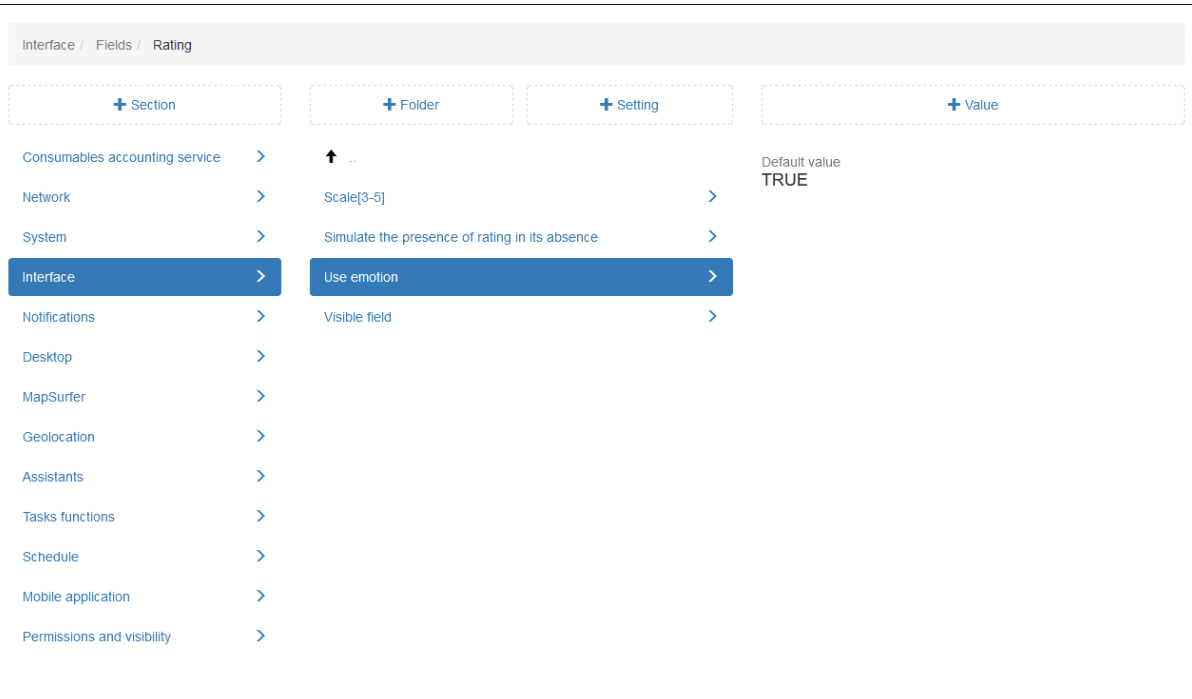
Fig. 2.154: Using emoji

**Visible field**

In this setting, you can enable the visibility of the rating in the Ac-
tiveMap Desktop. By default, this setting is disabled (Fig. 2.283).
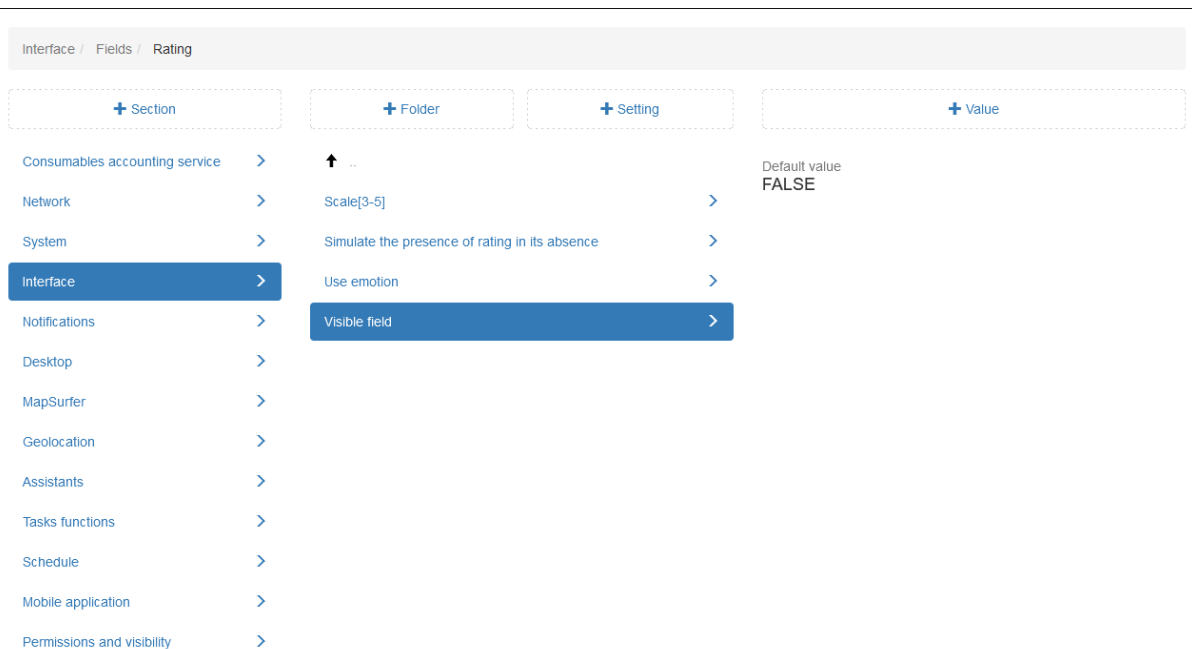


Fig. 2.155: Setting the visibility of the rating

### 2.3.3.7.5 "Notifications" section

This section allows configuring PUSH notifications and webhooks (Fig. 2.156). You can set a new value by selecting the setting and then clicking "+ Value". In the window that opens, enable/disable the toggle switch or enter a value, then fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
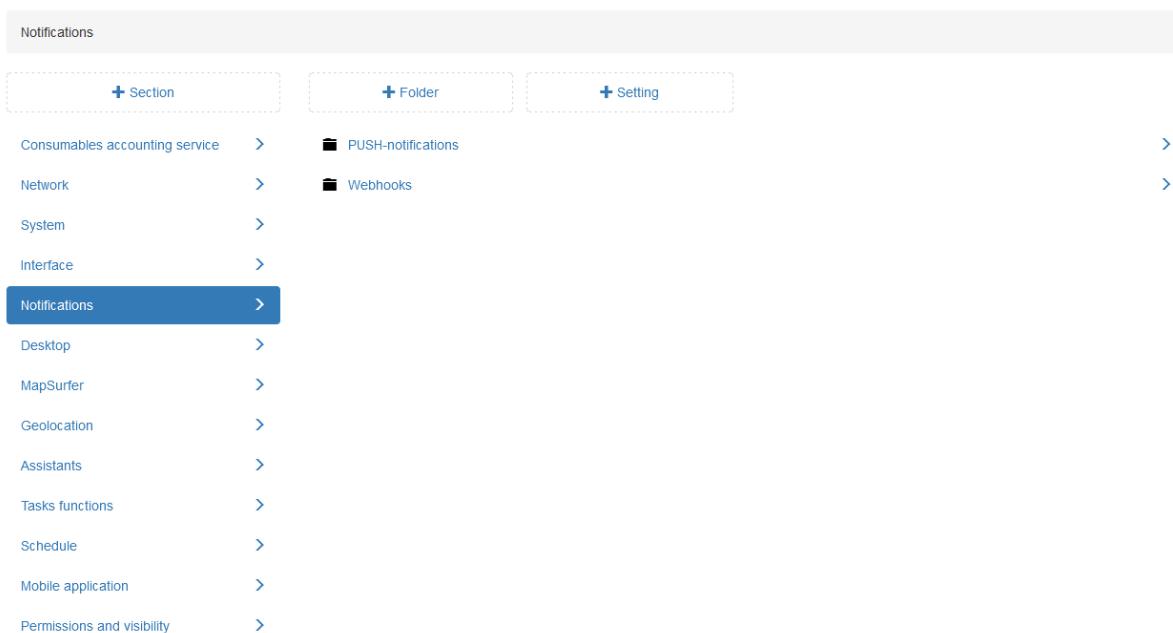


Fig. 2.156: "Notifications" section

### PUSH-notifications

In this setting administrator sets PUSH-notifications for such actions as creating, changing, deleting tasks, adding comments, etc. (Fig. 2.157).
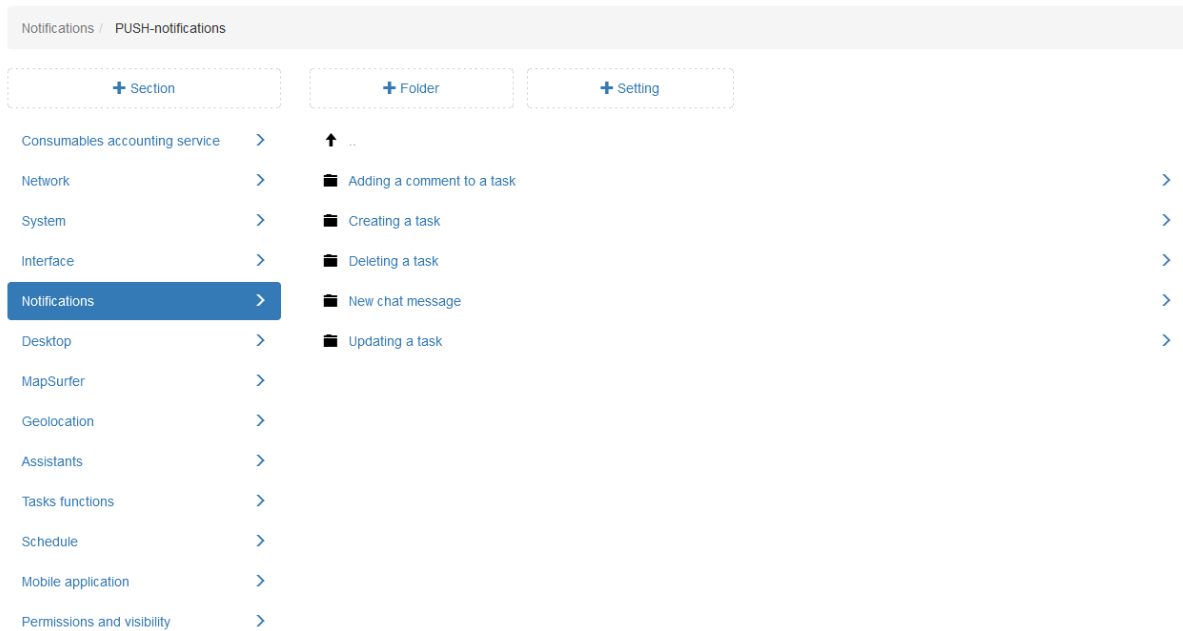
Fig. 2.157: PUSH-notifications

PUSH-notifications for each of the presented actions are configured for user roles (Fig. 2.158).
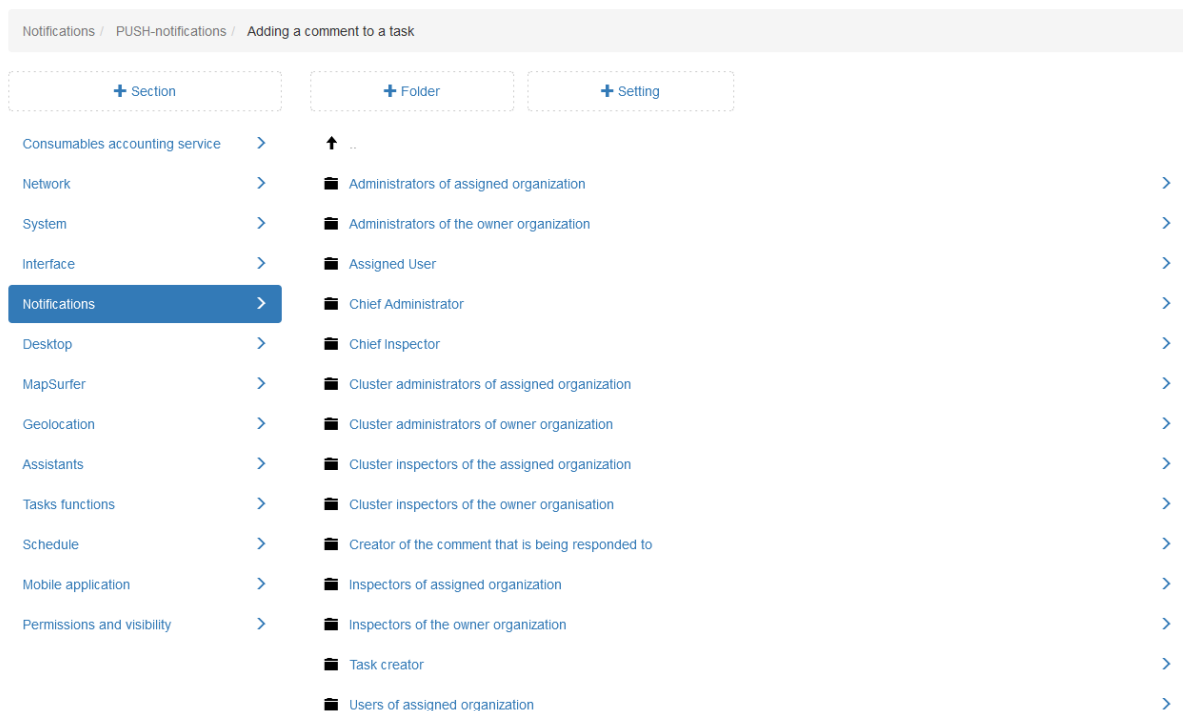


Fig. 2.158: PUSH-notifications configured for user roles

**Webhooks**

This setting specifies how many requests can be sent simultaneously to the server with the selected data transfer protocol. The default is 10 requests (Fig. 2.159).
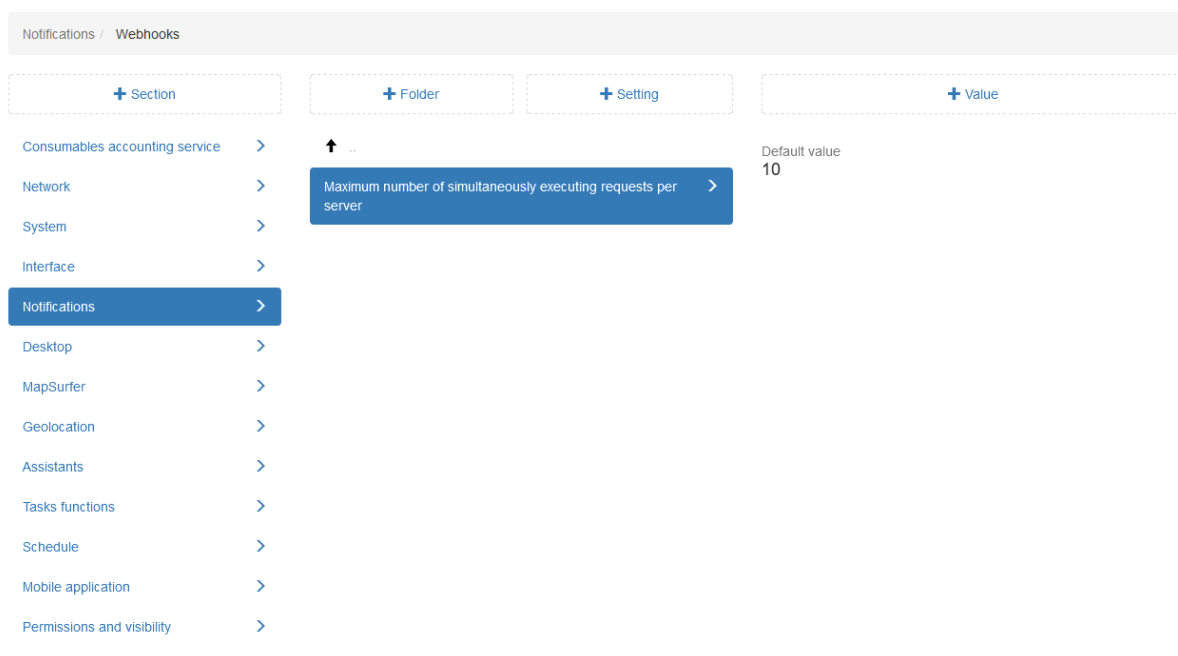


Fig. 2.159: Maximum number of simultaneous requests per server

### 2.3.3.7.6 "Desktop" section

In this section, you can configure additional options to work in the ActiveMap Desktop (Fig. 2.160). You can set a new value by selecting the setting and clicking "+ Value". In the window that opens, enable/disable the toggle switch or enter the desired name and then fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
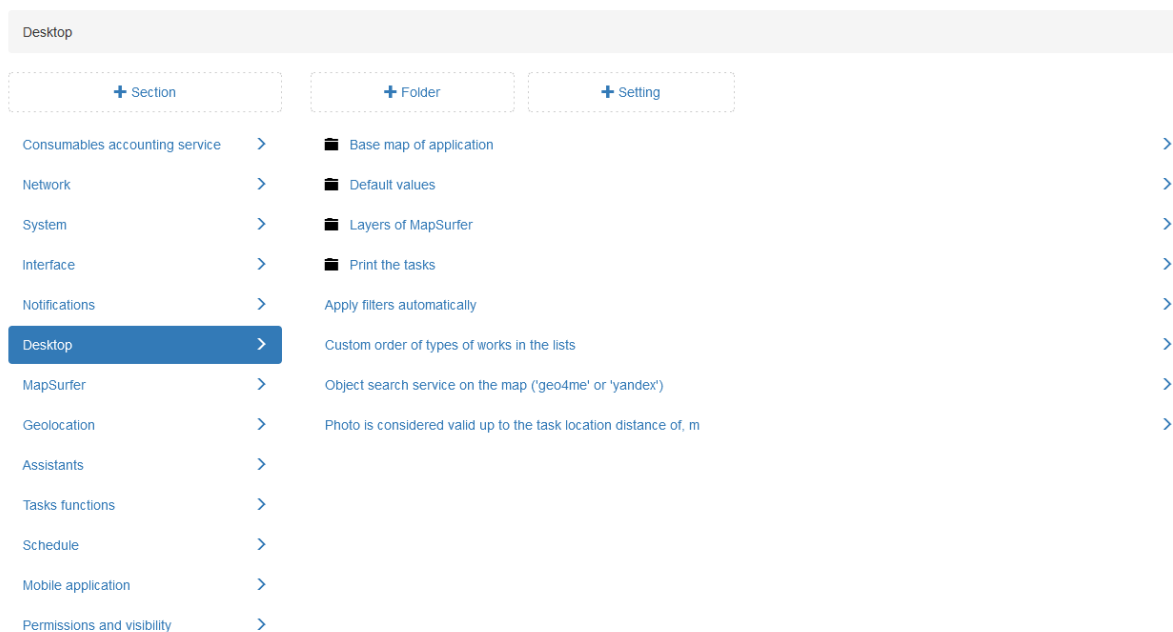
Fig. 2.160: "Desktop" section

## Basemap of application

The setting allows adding a map as a base layer to work in the ActiveMap Desktop.

**Name of the folder for the cache**

The setting allows setting a new folder to store the map cache to work in the ActiveMap Desktop.

**Projection of basemap**

In this setting, you can enter the projection for the map's base layer (in PROJ.4 format).

**URL of TMS-service**

In this setting, you can enter the URL of the TMS service to specify the map as the base layer (Fig. 2.161).

Fig. 2.161: Application basemap

## Default values

You can specify values that are automatically filled in when tasks are created in the ActiveMap Desktop (Fig. 2.162). Thus, after adding a value, the user does not need to fill in these fields manually when creating a task. They are filled in automatically.
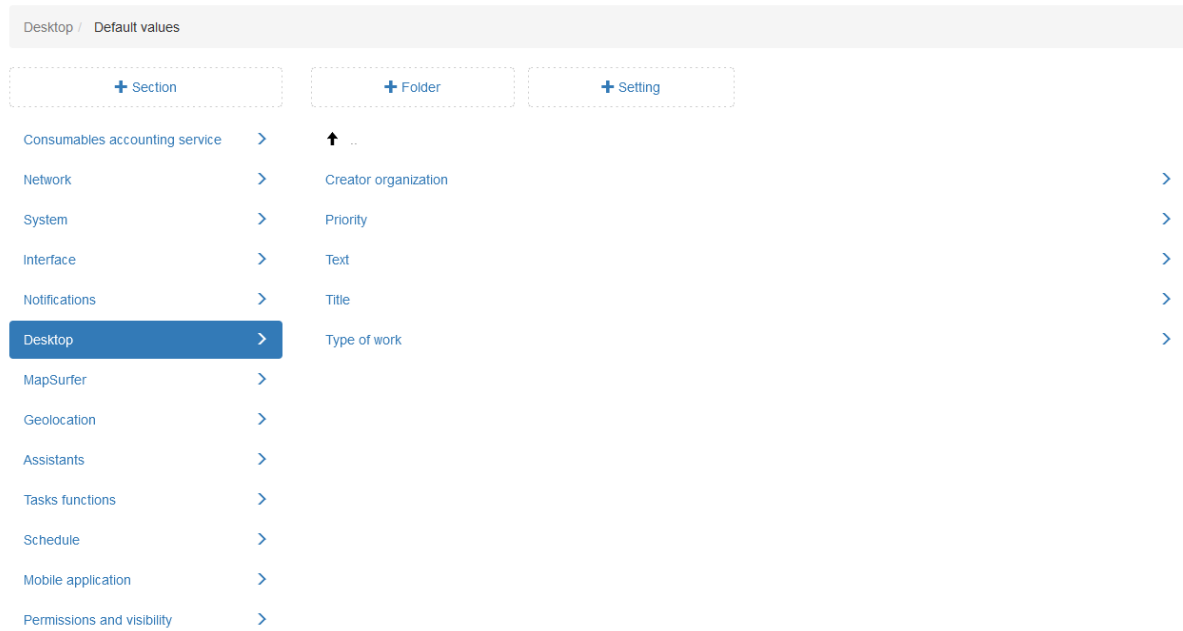
Fig. 2.162: Default values

## Layers of MapSurfer

This setting allows you to specify the comma-separated IDs of the layers that are displayed by default on the task map window (Fig. 2.163).
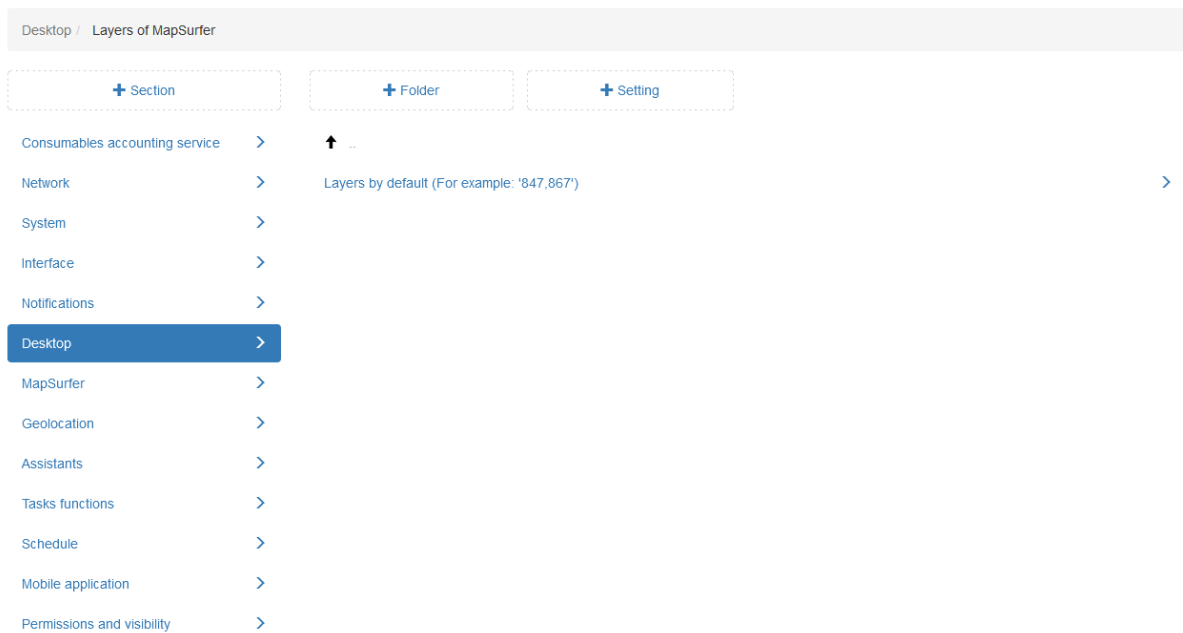


Fig. 2.163: Default values

### Print the tasks

The setting allows you to optionally hide the map when printing tasks in the ActiveMap Desktop (Fig. 2.164).
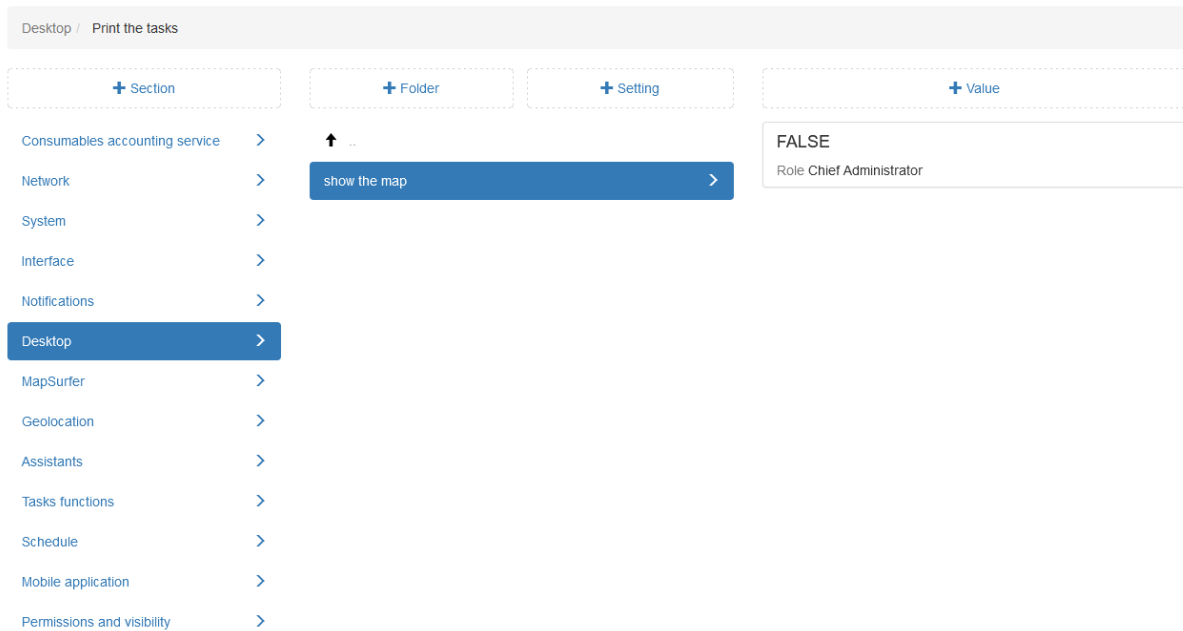


Fig. 2.164: Printing a task

### Apply filters automatically

In this setting, you can enable automatic application of the filter to tasks in the ActiveMap Desktop. By default, this setting is enabled (Fig. 2.165). If a value is created with the toggle switch turned off, the user should click "Apply" to apply the filter.

Fig. 2.165: Automatic filter application

## Photo is considered valid up to the task location distance of, m

The setting allows users to specify the radius of the task point. The default radius is 150 meters (Fig. 2.166).



Fig. 2.166: Radius of the task point for taking photos

### 2.3.3.7.7 "MapSurfer" section

This section allows configuring additional options for working in the ActiveMap Web (Fig. 2.167). You can set a new value for each option by selecting it and then clicking "+ Value". Enable/disable the toggle switch or enter a new value in the opened window and fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
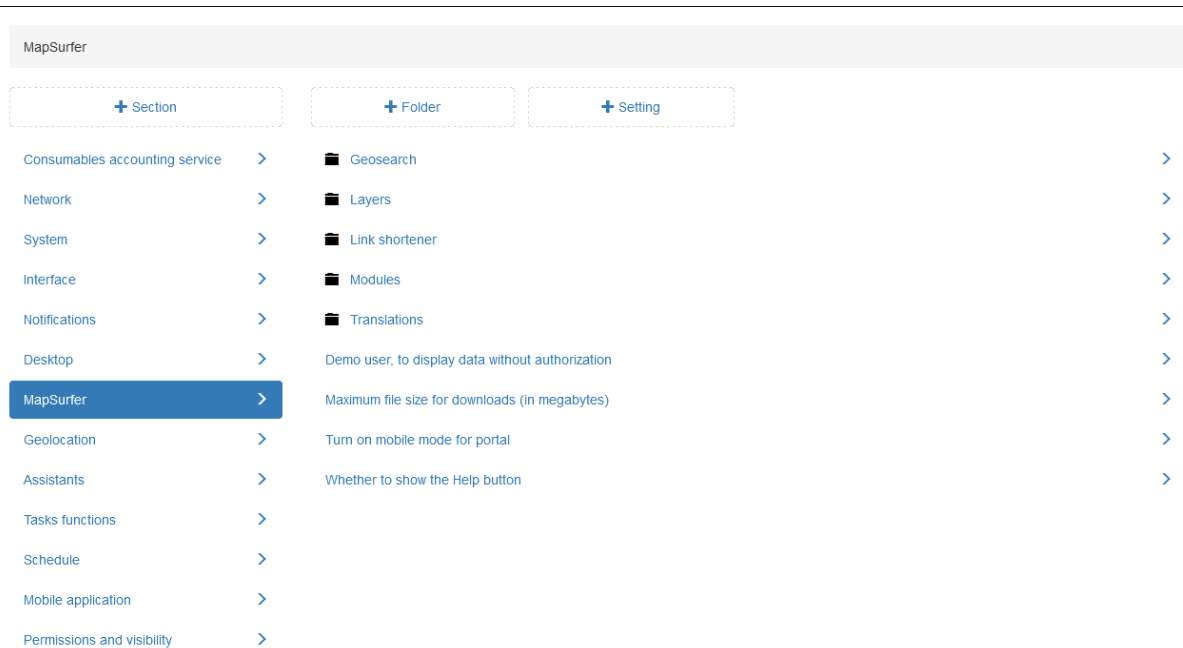


Fig. 2.167: "MapSurfer" section

### Geosearch

The "GeoSearch" folder contains services for geocoding.

**Buffer for reverse geocoding by layers (in meters)**

To search by coordinates in layers, form a small area where the search is performed. Here you can set the buffer in which the search works. The default setting is 50 meters (Fig. 2.168).

Fig. 2.168: Buffer for reverse geocoding

**Limit the search to the transmitted area for Nominatim (GeoCodingNominatim)**

This setting defines the limits for searching through Nominatim. If the setting is enabled, the search is carried out only in the area that is transmitted in the request. In particular, when searching in the ActiveMap Web in map mode, this is the area of the map that is currently displayed in the browser. By default, this feature is disabled (Fig. 2.169).



Fig. 2.169: Limit the search to the transmitted area for Nominatim (GeoCodingNominatim)

**Limiting the number of entries in the response for geocoders**

A setting that limits the number of records found in the "Search results" window in "Map" mode. The larger value is set in this setting, the longer it takes to process the request. The default setting is 20 records (Fig. 2.170).
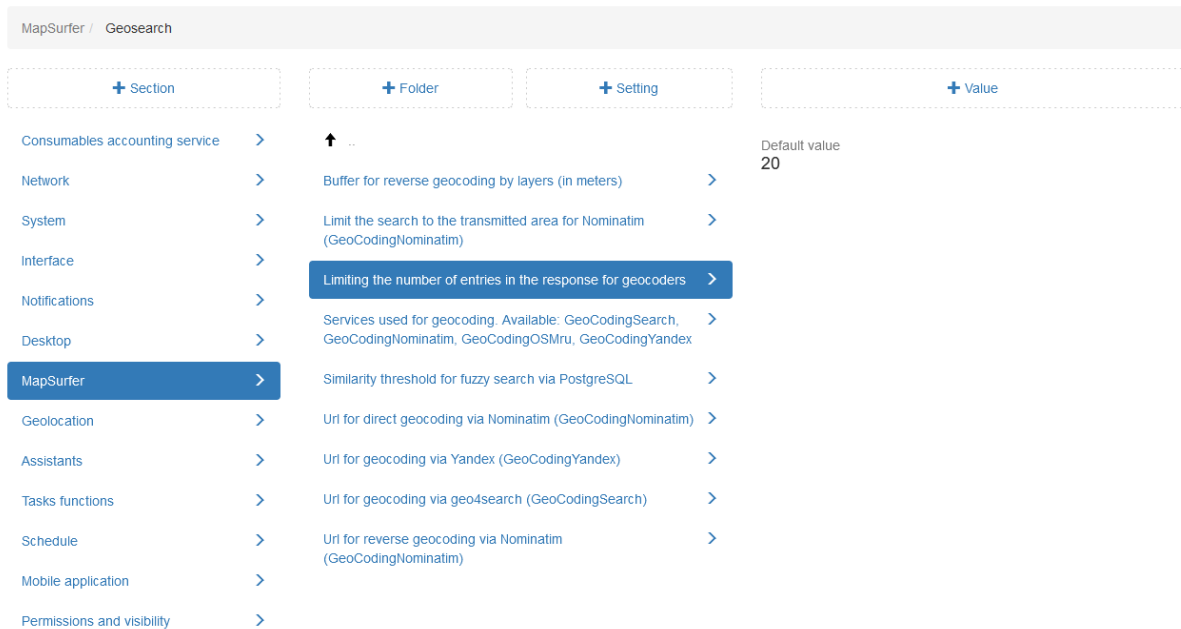


Fig. 2.170: Limiting the number of entries in the response for geocoders

**Services used for geocoding**

The following services are available by default:

- GeoCodingSearch – search.geo.pro;

- GeoCodingNominatim – a service that works through nominatim;

- GeoCodingOSMru – https://openstreetmap.ru/api/search;

- GeoCodingYandex – yandex service (works only with an API key).

The administrator can connect several geocoders for geosearch by sequentially adding several values. When multiple services are used, the search starts with the first one in the list. If nothing is found, the second one is used, and so on (Fig. 2.171).

Fig. 2.171: Services used for geocoding

## Layers

This folder contains additional settings for working with thematic layers
([Fig. 2.172](#)).



Fig. 2.172: Layers

### Importing objects

In this folder, you can configure the settings for importing service objects from a text file in the ActiveMap Mobile mobile application.

### Maximum number of rows to import

This setting specifies a limit on the number of rows that can be loaded at one time. The default setting is 10000 rows (Fig. 2.173).
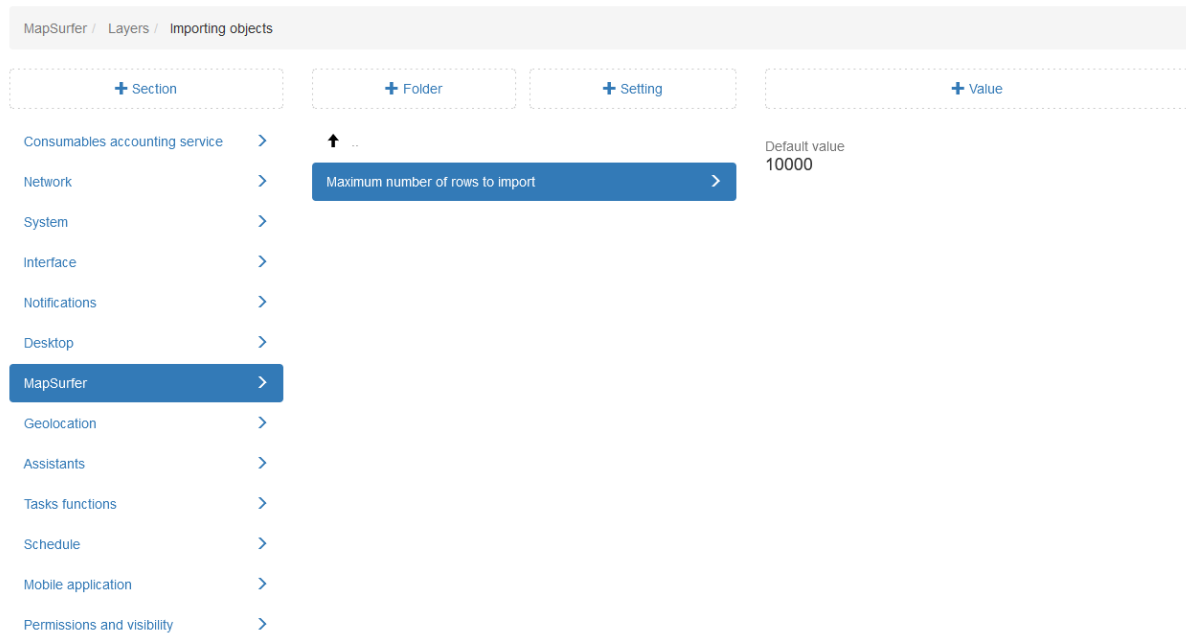


Fig. 2.173: Maximum number of rows to import

### Always show signatures in the user layer

In this setting, you can enable automatic display of the user's signature when the "Users" layer is connected in the ActiveMap Web web application. The signature is formed on the basis of the "Name" field in the user card. By default, the setting is disabled (Fig. 2.174).

---

Fig. 2.174: Enabling user signatures

**Buffer for finding objects in a point**

The setting allows users to set the size of the buffer for getting additional information on objects in the "Map" mode. The default is 200 meters (Fig. 2.175).

Fig. 2.175: Buffer for finding objects in a point

**Clearing the GWC when notified of layer changes**

This setting allows the administrator to disable GeoWebCache deletion when notified of layer changes. By default, the setting is disabled (Fig. 2.176).

Fig. 2.176: Clearing the GWC when notified of layer changes

**Coding for layer loading/unloading (csv, shape)**

The setting specifies the encoding used to export and import data in .csv and .shape formats. The default encoding is Windows-1251 (Fig. 2.177).



Fig. 2.177: Encoding for layer loading

---

**How often to check for updates of layer features (in seconds)**

Events about changes to thematic layer objects are collected for the interval specified in this setting (e.g. creation of a new object through the ActiveMap Mobile mobile application). If such changes have accumulated, an event is sent via WebSocket that it is necessary to redraw the corresponding layers in the ActiveMap Web. After that, the new object appears automatically, i.e. there is no need to update the data or reconnect the layer. If there are no changes, no event is sent. The default setting is 15 seconds (Fig. 2.178).



Fig. 2.178: Frequency of checking for layer object updates

**Limit the number of entries in the response when searching for objects by area**

This setting limits the number of records in the "Search results" window when searching for objects using the "Area object list" tool in the "Map" mode. The larger the value specified in this setting, the longer the request is processed. The default setting is 30 records (Fig. 2.179).

Fig. 2.179: Limit the number of entries in the response when searching for objects by area

**Limit the number of entries in the response when searching for objects in a point**

This setting limits the number of entries in the "Search results" window when searching for objects in the "Map" mode. The larger the value set in this setting, the longer the request is processed. The default is 30 entries (Fig. 2.180).

Fig. 2.180: Limit the number of entries in the response when searching for objects in a point

**Number of records in one transaction when indexing a layer**

This setting specifies the number of objects requested at the same time to index the layer. The default is 1000 objects (Fig. 2.181).



Fig. 2.181: Number of records in one transaction when indexing a layer

### The number of layer objects in one transaction when creating a layer from a shapefile

This parameter specifies the number of layer objects in a transaction for loading into the database when importing a layer from a shapefile. The default is 1000 objects ([Fig. 2.182](#)).



Fig. 2.182: The number of layer objects in one transaction when creating a layer from a shapefile

### Unloading objects of a geometry layer

This setting allows including information about the object geometry in the layer export in the ActiveMap Web. By default, this setting is disabled ([Fig. 2.183](#)).

Fig. 2.183: Exporting a layer with object geometry setting

**Whether to run daily reindexing**

If the layer is edited via REST (in ActiveMap Desktop, ActiveMap Web, ActiveMap Mobile), the changes automatically appear in the index. By default, daily layer reindexing is disabled (Fig. 2.184).



Fig. 2.184: Daily layer reindexing setting

### Link shortener

A service that allows you to shorten long links in the "Fixed link" tool when setting map parameters and connecting thematic layers in the ActiveMap Web in the "Map" mode (Fig. 2.185).
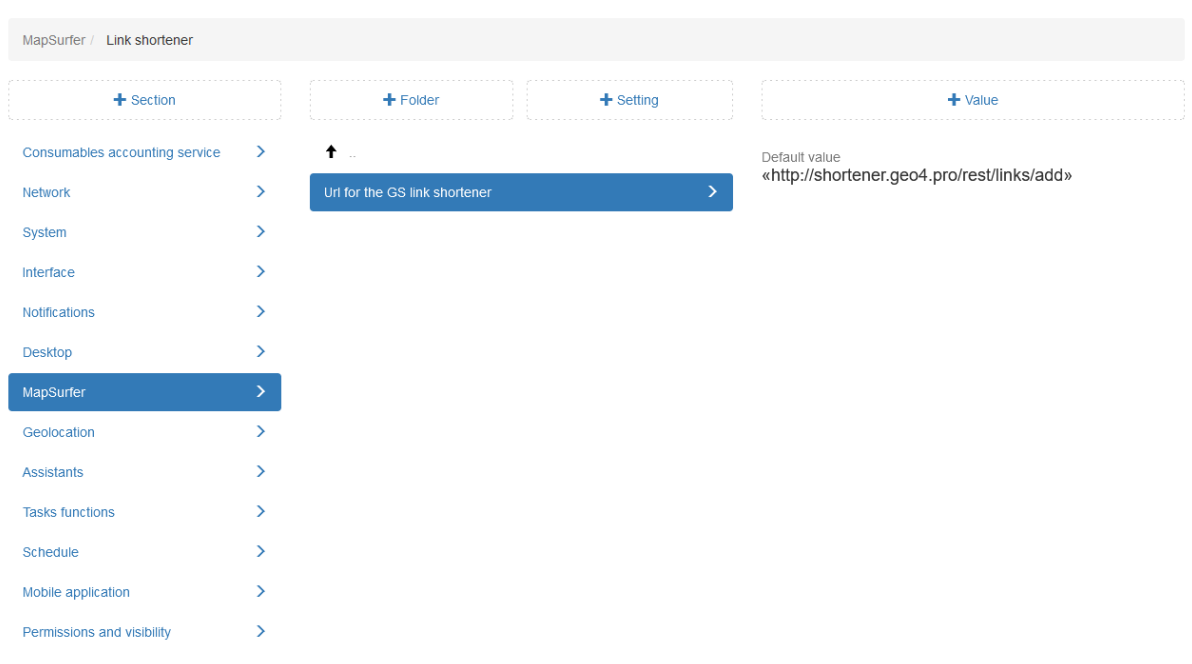


Fig. 2.185: Link shortener

### Modules

**Statistics module** is a report that is generated (updated) after a specified period of time based on the collected data in online mode. This report is available to users in the "Map" mode (Fig. 2.186).
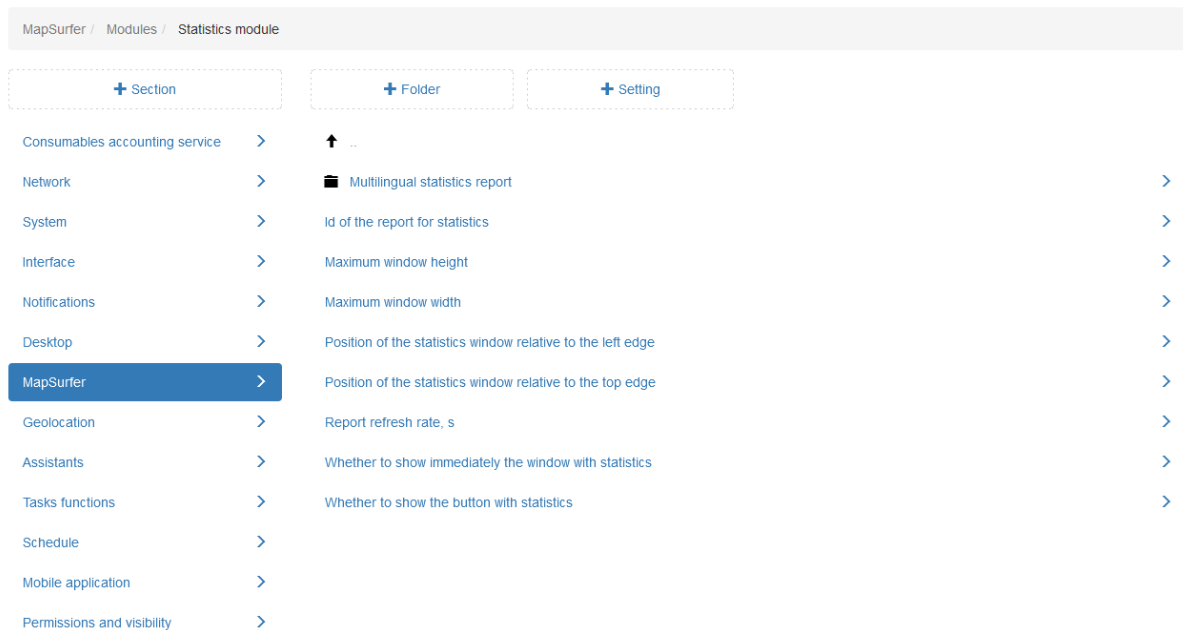
Fig. 2.186: Statistics module

**Multilingual statistics report**

The statistics module supports report generation for different interface languages. In this folder, you can set the required report for the specified language. To add a new language, click the "+ Setting" button and create a new folder. Specify the two-letter language code in the "Key" field and enter the language name in the "Name" field (Fig. 2.187). Next, add a value containing the ID of the required report for this language.
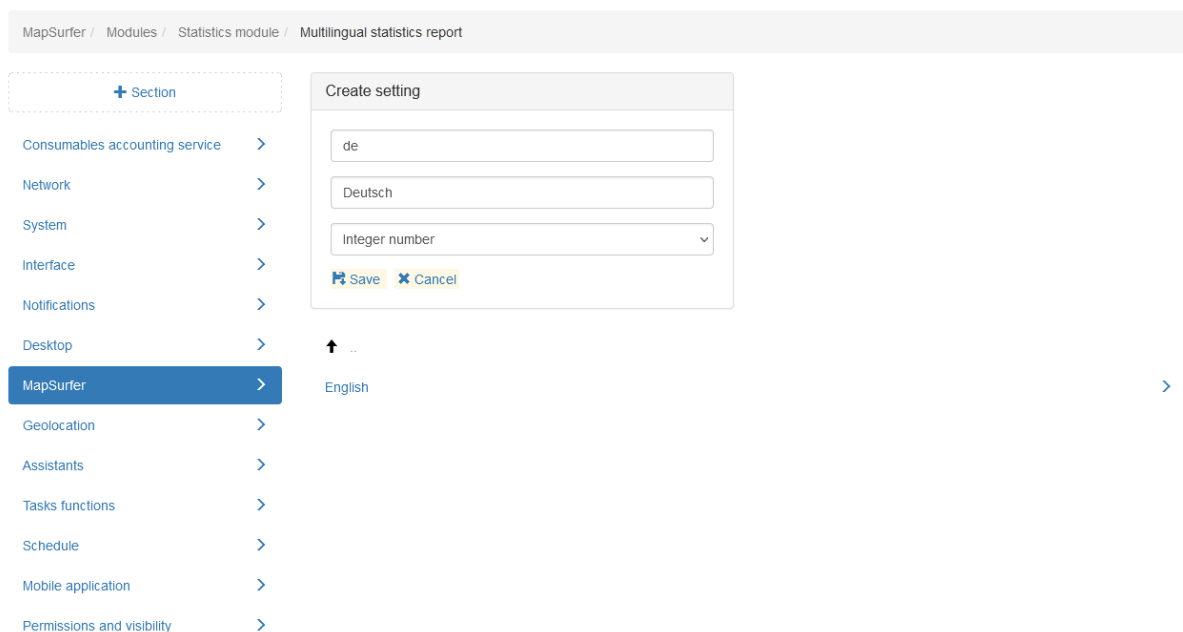


Fig. 2.187: Adding a language

**2.3. Management module**

If the language is not set, then the report specified in the "**Id report for statistics**" setting is used by default.

**Id of the report for statistics**

In this setting, you can specify the id of the report, which is used to display the statistics. You can specify different reports for different roles or organizations (Fig. 2.188).



Fig. 2.188: Specifying the report for the Statistics module

**Maximum window height** and **Maximum window width**

By default, the statistics window is optimized for the report content (Fig. 2.189). If necessary, you can change the window size by specifying new values.

Fig. 2.189: Maximum window height

**Position of the statistics window relative to the left edge** and **Position of the statistics window relative to the top edge**

The default location of the statistics popup in the browser is relative to the top and left edges (Fig. 2.190). The administrator can change the window size by specifying new values.



Fig. 2.190: Statistics window location relative to the top

---

### Report refresh rate

This setting allows you to set the frequency of updating data in online statistics. The default is 180 seconds (Fig. 2.191).



Fig. 2.191: Report update frequency (in seconds)

### Whether to show immediately the window with statistics

When this setting is enabled, a pop-up window with online statistics is automatically displayed in the "Map" mode after authorization in the ActiveMap Web. This feature is disabled by default (Fig. 2.192).

Fig. 2.192: Show statistics window immediately

**Whether to show the button with statistics**

This setting allows you to display a button for connecting online statistics in the ActiveMap Web toolbar in the "Map" mode. By default, this feature is disabled (Fig. 2.193).



Fig. 2.193: Show statistics button

### Translations

In this folder, you can override the standard translation of phrases in the ActiveMap Web in the desired language (Fig. 2.194).
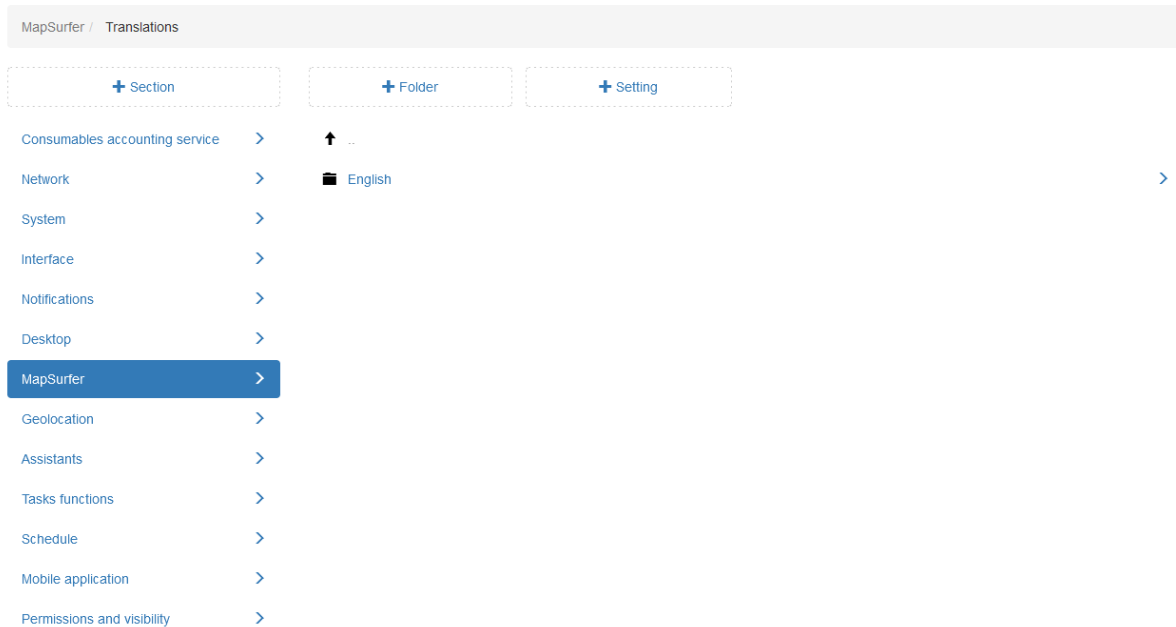


Fig. 2.194: Translations

You can also add another desired language. To do this, create a new folder by clicking "+ Folder". Specify the two-letter language code in the "Key" field and enter the name of the language in the "Title" field (Fig. 2.195).



Fig. 2.195: Adding a new folder

Next, add the setting. Specify the key used by the system for this translation in the "Key" field. In the "Name" field, specify any name that indicates what the translation refers to. Select "String" in the "Type" field (Fig. 2.196).

Fig. 2.196: Adding a new setting

After saving the setting, add a new value for using in this language for translation.

### Demo user to display data without authorization

In this setting, you can set a public user. After that, layers to which the specified user has access are displayed on the portal without authorization (Fig. 2.197).



Fig. 2.197: Public user

### Maximum file size for uploading (in megabytes)

A setting that allows specifying the maximum file size for import in the ActiveMap Web. The default is 2048 megabytes (Fig. 2.198). If the size of the uploaded file exceeds this value, an error message appears and the file is not uploaded.
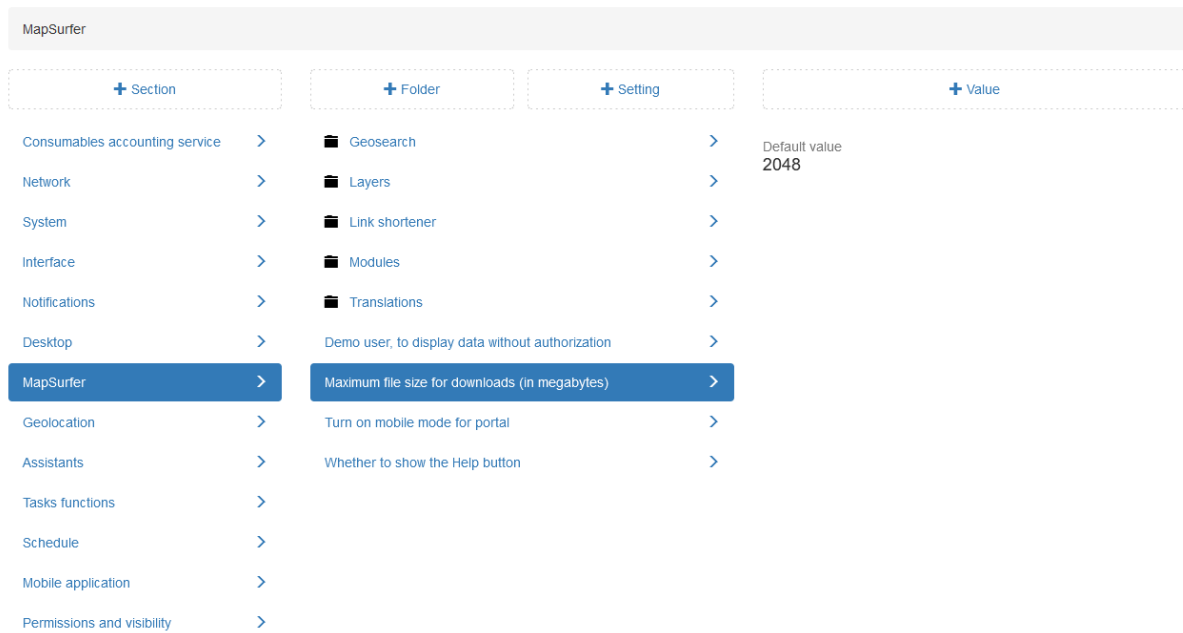


Fig. 2.198: Maximum file size for uploading

### Turn on mobile mode for portal

The setting allows you to disable the display of ActiveMap Web in the mobile mode. By default, this setting is enabled (Fig. 2.199).

Fig. 2.199: Enable mobile mode for portal

### Whether to show the "Help" button

This setting allows hiding the "Help" button in the administration panel.
By default, the setting is enabled (Fig. 2.200).



Fig. 2.200: Disabling the "Help" button

### 2.3.3.7.8 "Geolocation" section

The section presents additional settings for geolocation monitoring in the ActiveMap Mobile (Fig. 2.201). These settings allow you to determine independently which mode is set on the server. You can set a new value by selecting the setting, then clicking "+ Value". In the window that opens, enable/disable the toggle switch or enter the required name and then fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
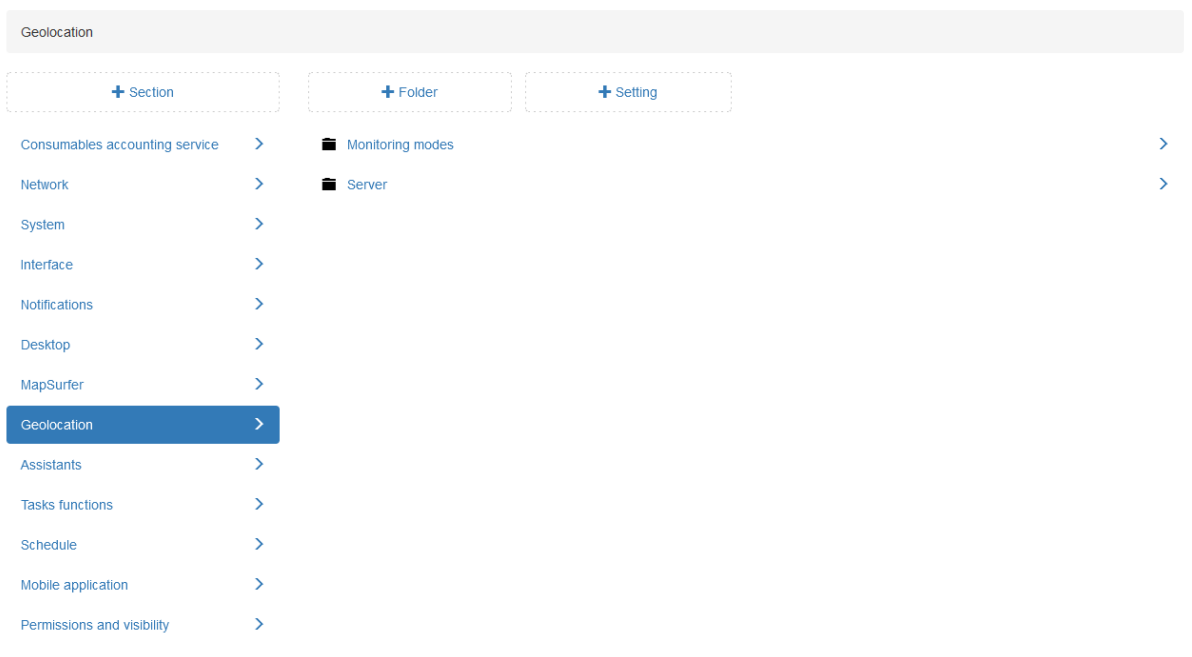


Fig. 2.201: "Geolocation" section

### Monitoring modes

There are 5 monitoring modes for user devices in the mobile application, which differ in location accuracy. In the maximum accuracy mode, the frequency of transmitting location points to the server is increased. In the mode of minimum accuracy, the location of devices is transmitted less frequently. However, this mode saves the battery power of the user's device.

The mode corresponds to the page number next to the operating system of the device (from 0 to 4):

- 0 – maximum location accuracy,
- 4 – minimum location accuracy (Fig. 2.202).
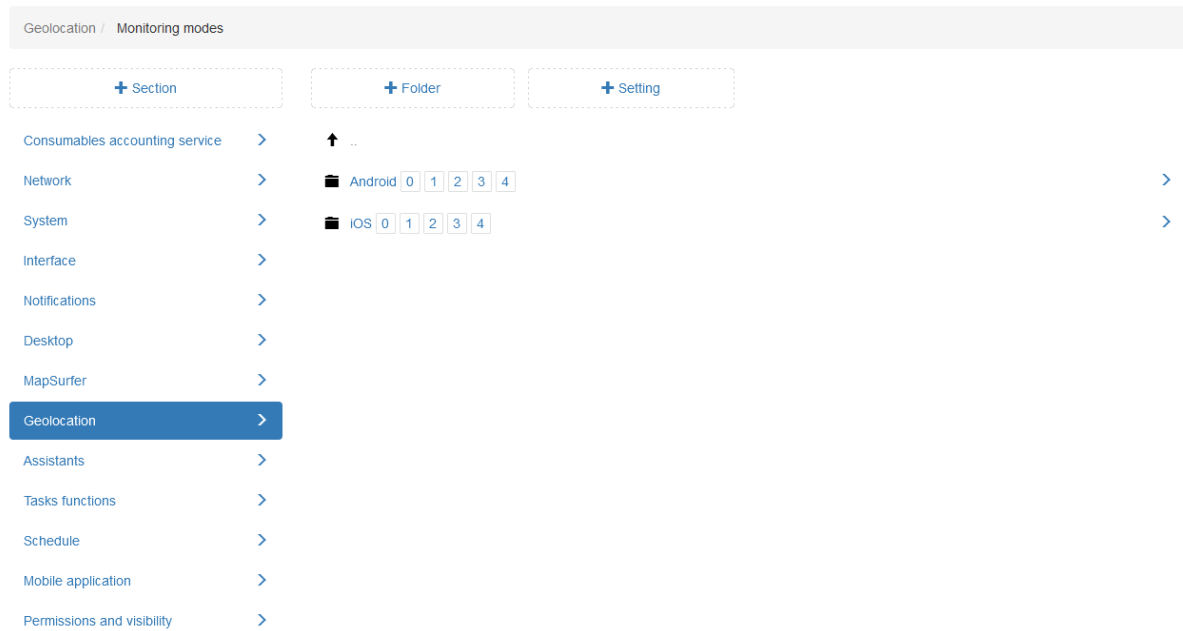
Fig. 2.202: Monitoring modes

Each mode has its own set of settings. To go to the settings of a particular mode, click the page number.

### Android

This folder contains geolocation settings in the ActiveMap Mobile for the Android operating system (Fig. 2.203). After adding a new value to the presented settings, the user needs to update the data in the mobile application.
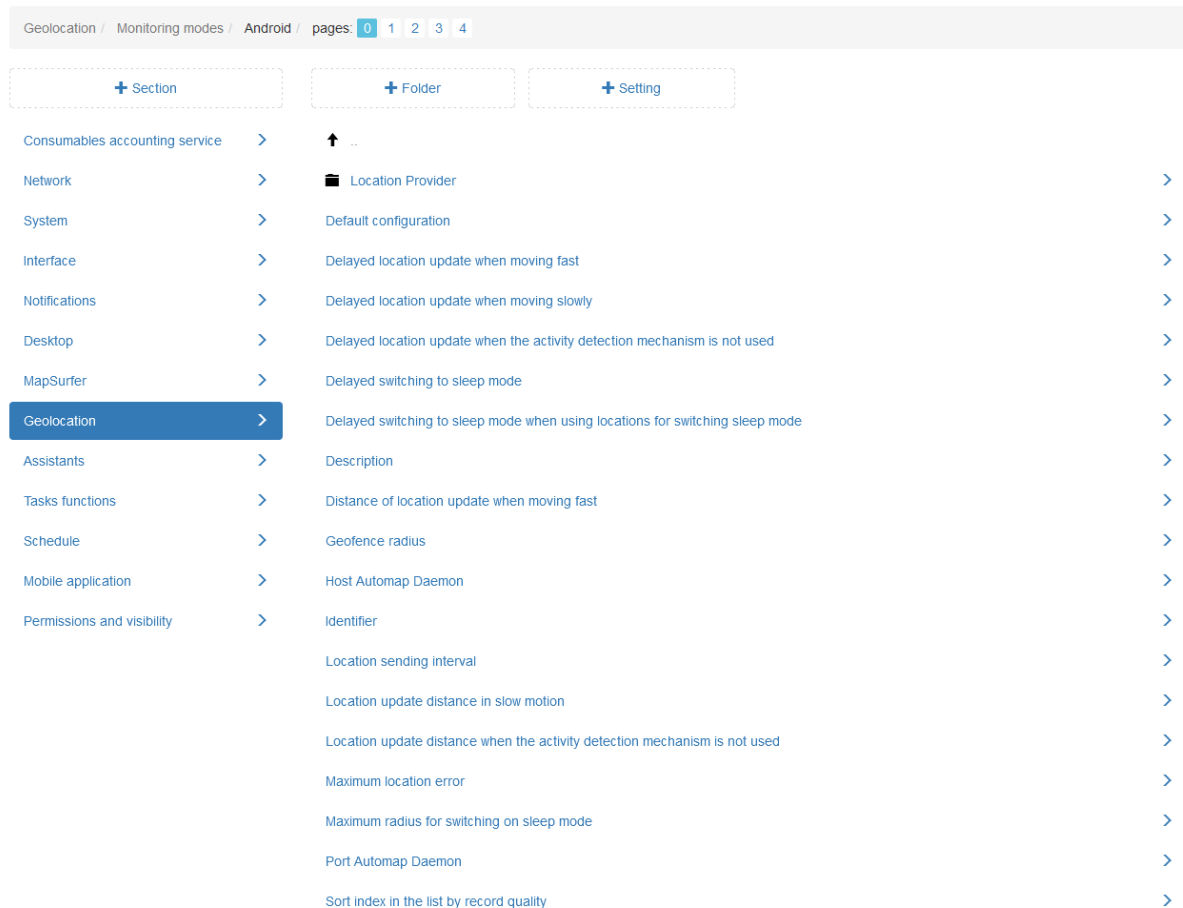
Fig. 2.203: Monitoring mode settings for the Android operating system

**Location Provider**

This setting allows specifying a provider for the "Mark my location" window in the ActiveMap Mobile. The following providers are available:

- gps (default) – this provider determines the location using satellites. Depending on the conditions, it may take some time to determine the location.

- network – determines the location based on the availability of a cell tower and Wi-Fi access points. The results are extracted using a network search.

- passive – the least accurate provider. This provider receives location data when other applications or services request it, without actually requesting locations itself.

**Default configuration**

This setting specifies which of the presented geomonitoring modes is activated on the server.

**Delayed switching to sleep mode**

This setting specifies how many seconds it takes for location monitoring to go into sleep mode when the user is not moving.

**Description**

The setting contains a brief description of the selected monitoring mode.

**Distance of location update when moving fast**

The setting is used to filter location points during fast motion, such as in a car. You can set how many meters the device needs to request the user's location for a given type of movement.

**Identifier**

The setting specifies the ID of the selected mode of user location monitoring. The setting is used for the «Geo4.me» mobile application. There are 5 geo-position monitoring modes. Based on this index, they are sorted in the interface of the mobile application.

**Location sending interval**

This setting specifies how often (in seconds) the user's location points are sent to the server.

**Location update distance in slow motion**

The setting is used to filter location points when moving slowly, such as when walking. You can set how many meters the device needs to request the user's location for a given type of movement.

**Maximum location error**

The setting specifies the allowable coordinate error. If the coordinate exceeds this error, then it is not taken into account.

The **Port AutoMap Daemon** and **Host AutoMap Daemon** settings specify the address and port for connecting to the server.

**Title**

This setting displays the name of the selected user location monitoring mode.

### iOS

This folder contains geolocation settings in the ActiveMap Mobile for the iOS operating system. After adding a new value to the presented settings, the user needs to restart the mobile application.

## Modules

### Using GPS monitoring

This setting specifies that location determination on iOS devices is also done via GPS. This setting is enabled by default (Fig. 2.204).
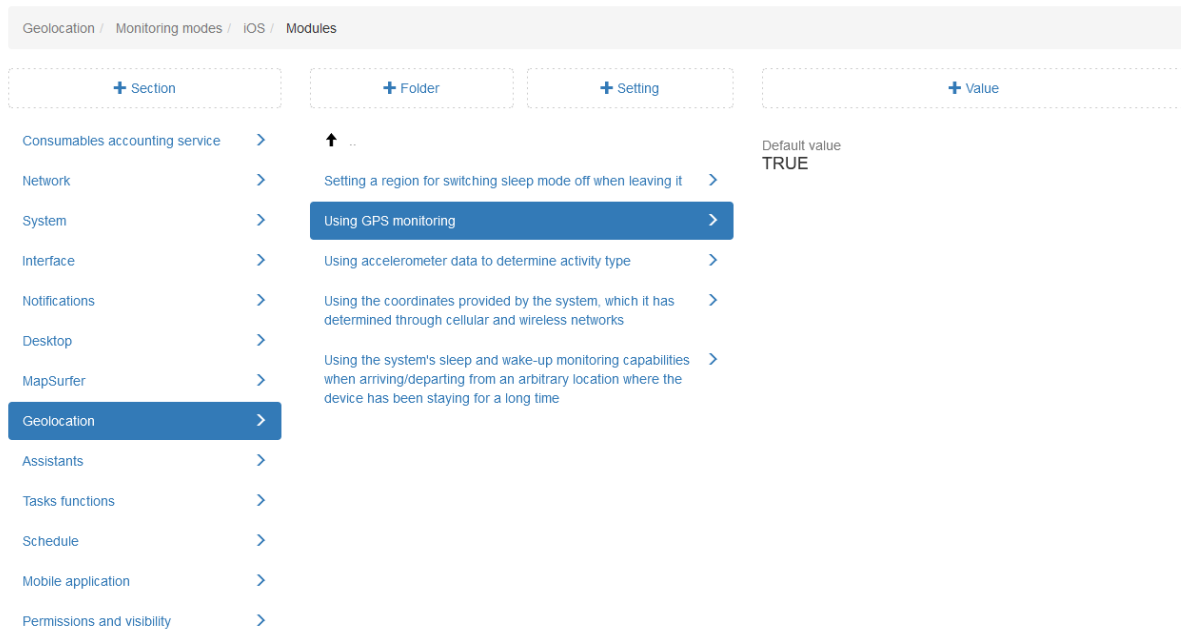


Fig. 2.204: Using GPS monitoring

### Using accelerometer data to determine activity type

In this setting, you can enable using the accelerometer in the mobile device, which improves the quality of the transmitted data (Fig. 2.205). Connecting this sensor also provides information about speed and direction of movement. However, using mobile device sensors involves increased battery consumption.

Fig. 2.205: Using accelerometer data

**Use the coordinates provided by the system, which it has determined through cellular and wireless networks**

By default, iOS devices use cellular and wireless networks to determine the location. You cannot select only one of these options. This setting is enabled by default (Fig. 2.206).



Fig. 2.206: Determining coordinates using cell and wireless networks

**Use the system's sleep and wake up monitoring capabilities when arriving/departing from an arbitrary location where the device has been staying for a long time** and **Setting a region for switching sleep mode off when leaving it**

Applying these settings allows users to stop sending data to the system if the user stays in one place for a long time (home, office). The setting puts the application into sleep mode after the time specified in the **Sleep Delay** setting (Fig. 2.207).
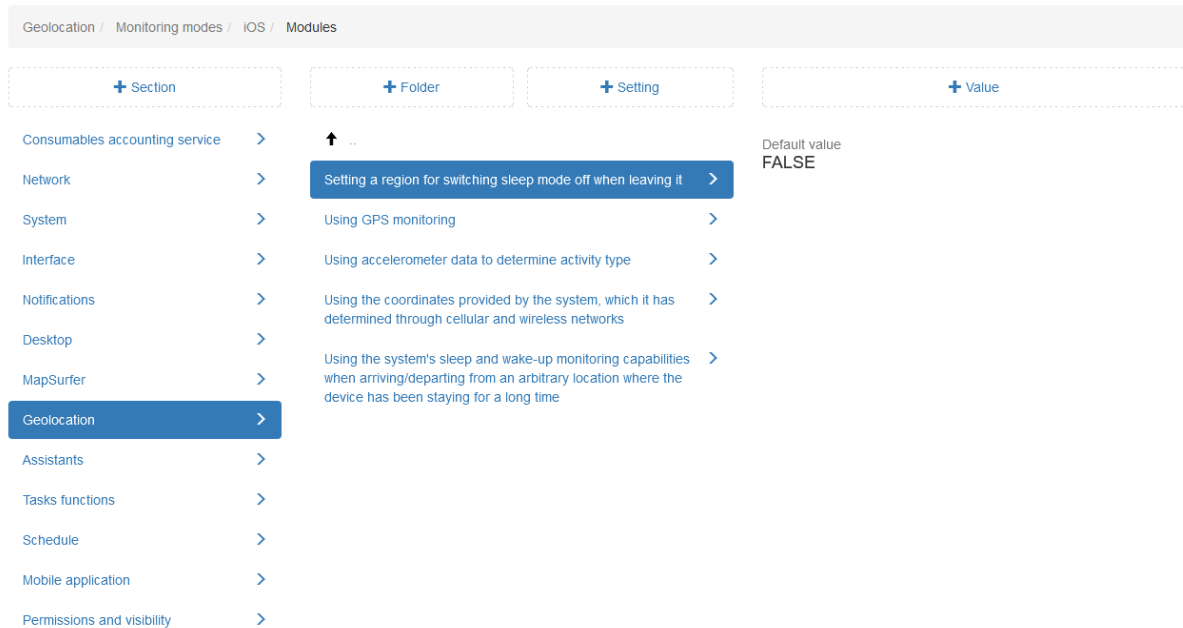


Fig. 2.207: Setting the region to wake up when coming out of it

## Fast motion monitoring settings

**Desired accuracy**

The setting is used to filter location points during fast motion, such as in a car. If the point radius is greater than the set value, these points are not used (Fig. 2.208).

Fig. 2.208: Desired accuracy during fast motion

**Minimal change of movement angle to get location update**

The setting is used to filter points. You can set the minimum angle of movement in degrees. If the angle matches the minimum value, the change is considered insignificant and the motion angle is not changed (Fig. 2.209).



Fig. 2.209: Minimal change of movement angle

**Minimum distance between points when receiving a location update**

This setting is used to filter points. You can specify how many meters should pass before a new point is recorded (Fig. 2.210). Frequent location updates can lead to increased battery consumption of the device.
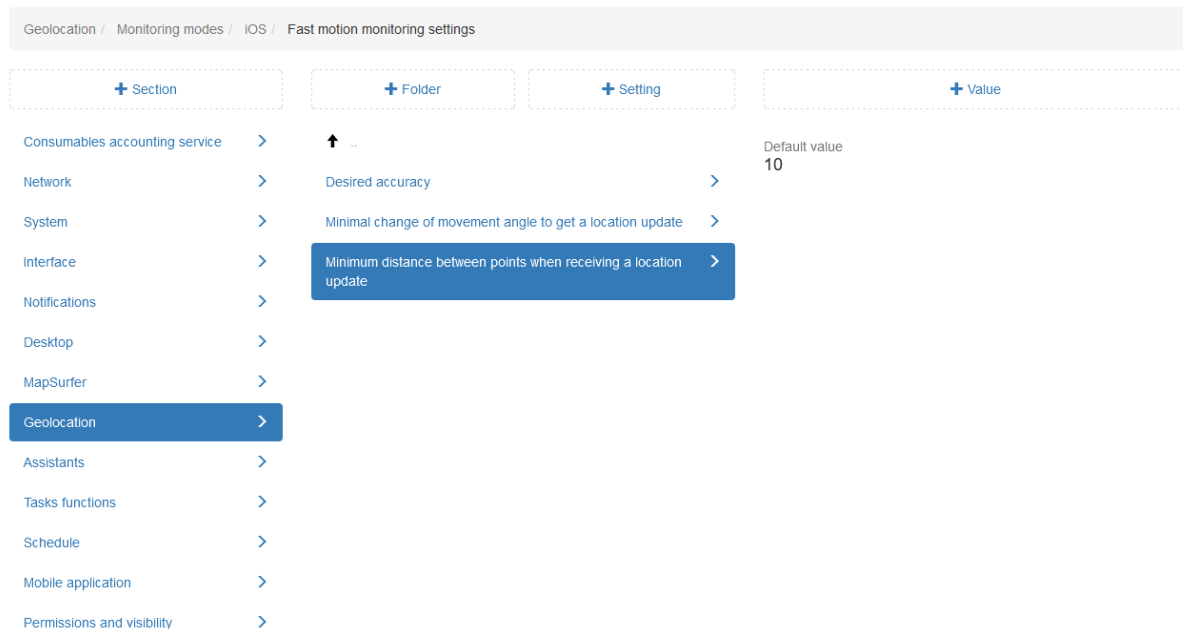


Fig. 2.210: Minimum distance between points when receiving a location update

## Slow motion monitoring settings

**Desired accuracy**

The setting is used to filter location points when moving slowly, such as when walking. If the point radius is larger than the set value, these points are not used (Fig. 2.211).

Fig. 2.211: Desired accuracy during slow movement

### Minimal change of movement angle to get a location update

The setting is used to filter points. You can set the minimum movement angle in degrees. If the angle matches the minimum value, the change is considered insignificant and the motion angle does not change (Fig. 2.212).



Fig. 2.212: Minimal change of movement angle

---

**Minimum distance between points when receiving a location update**

The setting is used to filter points. You can specify how many meters should pass before a new point is recorded (Fig. 2.213). However, frequent location updates can lead to increased battery consumption of the device.
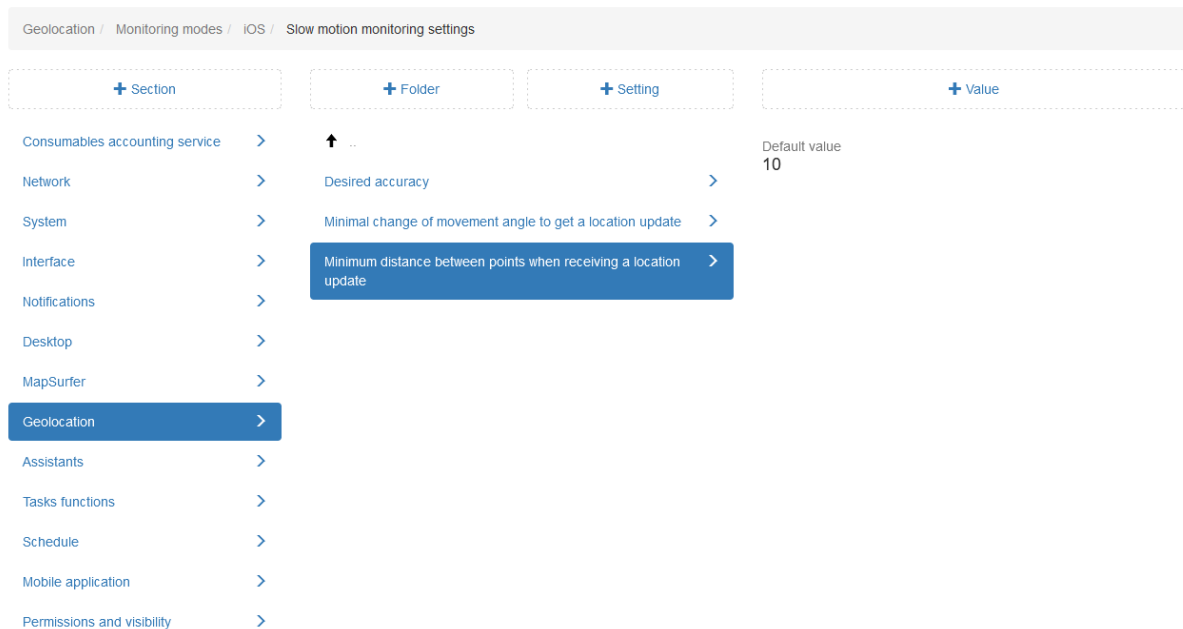


Fig. 2.213: Minimum distance between points when receiving a location update

**Default configuration**

The setting displays the geomonitoring mode activated on the server (Fig. 2.214).

Fig. 2.214: Default configuration

**Delayed switching to sleep mode**

The setting specifies how many seconds to wait before switching the location monitoring to sleep mode if there is no user movement (Fig. 2.215).



Fig. 2.215: Sleep delay

**Description**

The setting contains a brief description of the selected monitoring mode
([Fig. 2.216](#)).
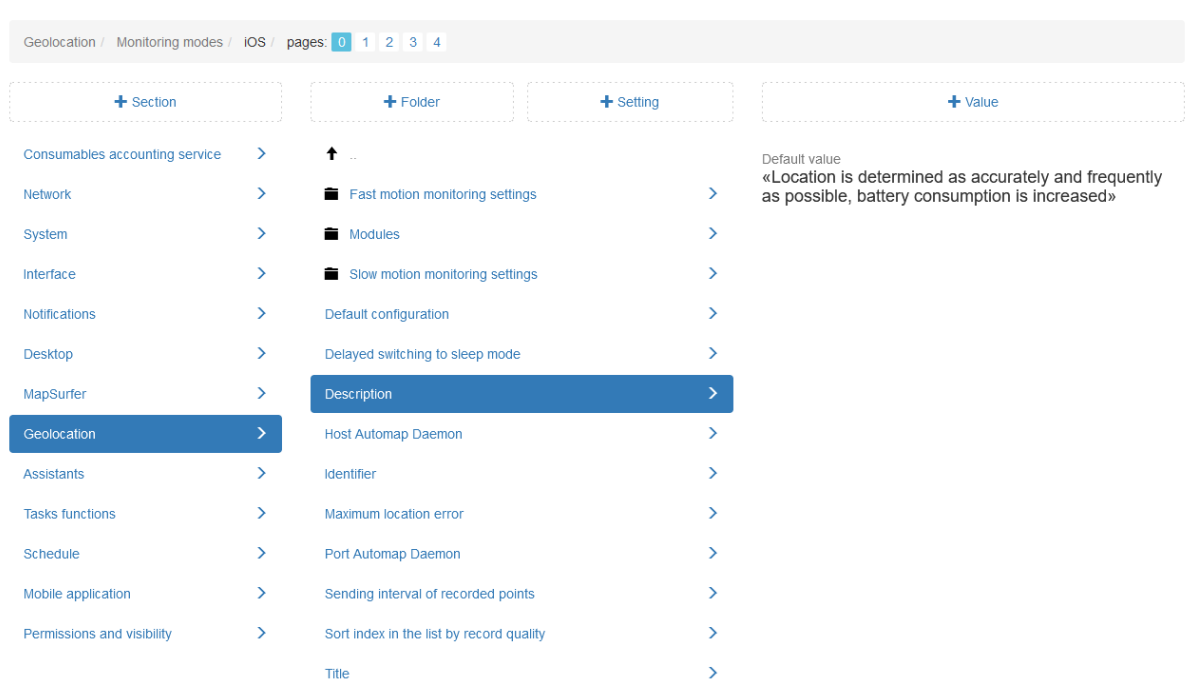


Fig. 2.216: Description of geolocation monitoring mode

**Identifier**

This setting specifies the ID of the selected location monitoring mode
([Fig. 2.217](#)).

Fig. 2.217: Identifier of the monitoring mode

**Maximum location error**

The setting specifies the allowable coordinate error. If a coordinate exceeds this error, it is not taken into account (Fig. 2.218).



Fig. 2.218: Maximum location error

The **Port AutoMap Daemon** and **Host AutoMap Daemon** settings specify the address and port to connect to the server (Fig. 2.219).



Fig. 2.219:  AutoMap daemon host

### Sending interval of recorded points

This setting determines the frequency (in seconds) of sending location points to the server (Fig. 2.220).

Fig. 2.220: Recorded points sending interval

**Sort index in the list by record quality**

The setting is used for the «Geo4.me» mobile app. There are 5 modes of geo-position monitoring. They are sorted in the interface of the mobile application based on this index (Fig. 2.221).



Fig. 2.221: Sort index in the list by record quality

**Title**

This setting displays the name of the selected user location monitoring mode (Fig. 2.222).
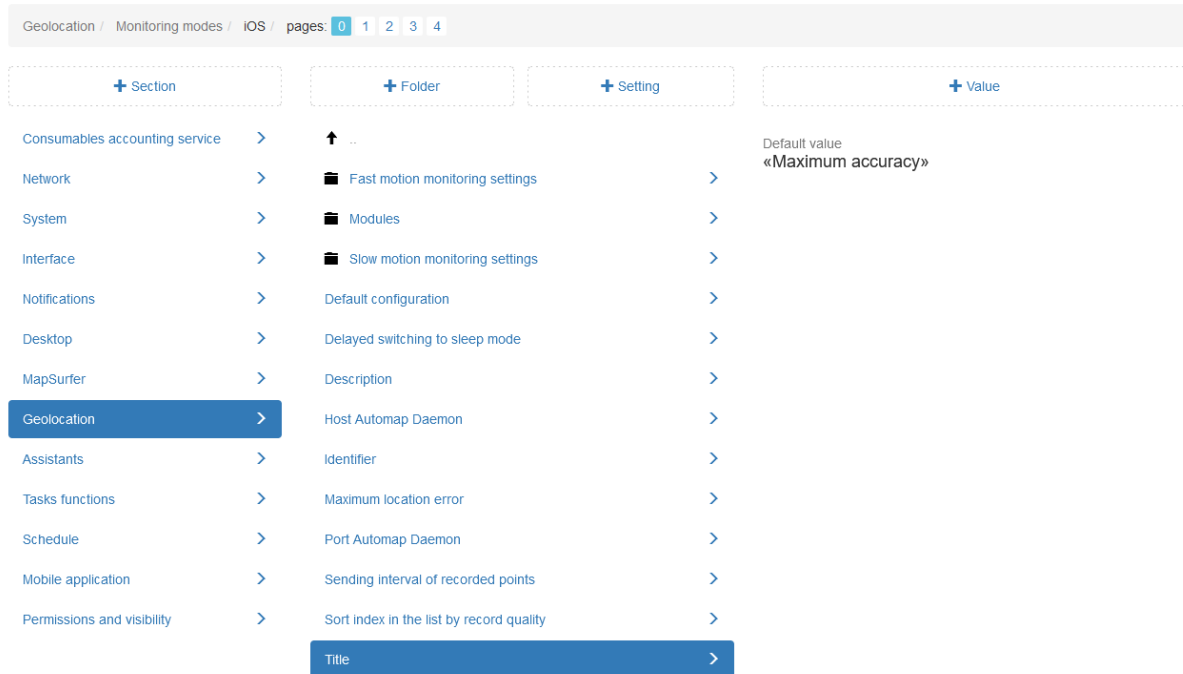


Fig. 2.222: Name of geolocation monitoring mode

## Server

In this folder, you can configure server settings for monitoring user geolocation (Fig. 2.223).
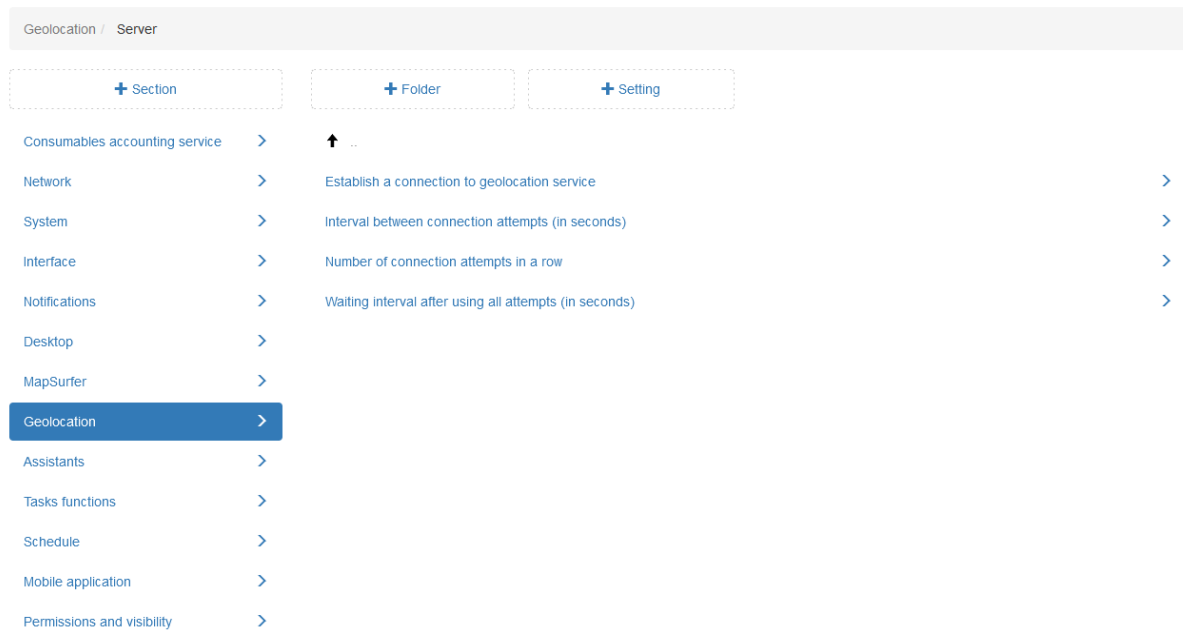
Fig. 2.223: Server settings for monitoring geolocation

**Establish a connection to geolocation service**

When the setting is enabled, Cerebellum gets the location of users by establishing a connection with the location tracking service. If this setting is disabled during operation, Cerebellum resets the connection. The response to the setting change is immediate, but the connection itself may take some time to establish. This setting is enabled by default (Fig. 2.224)
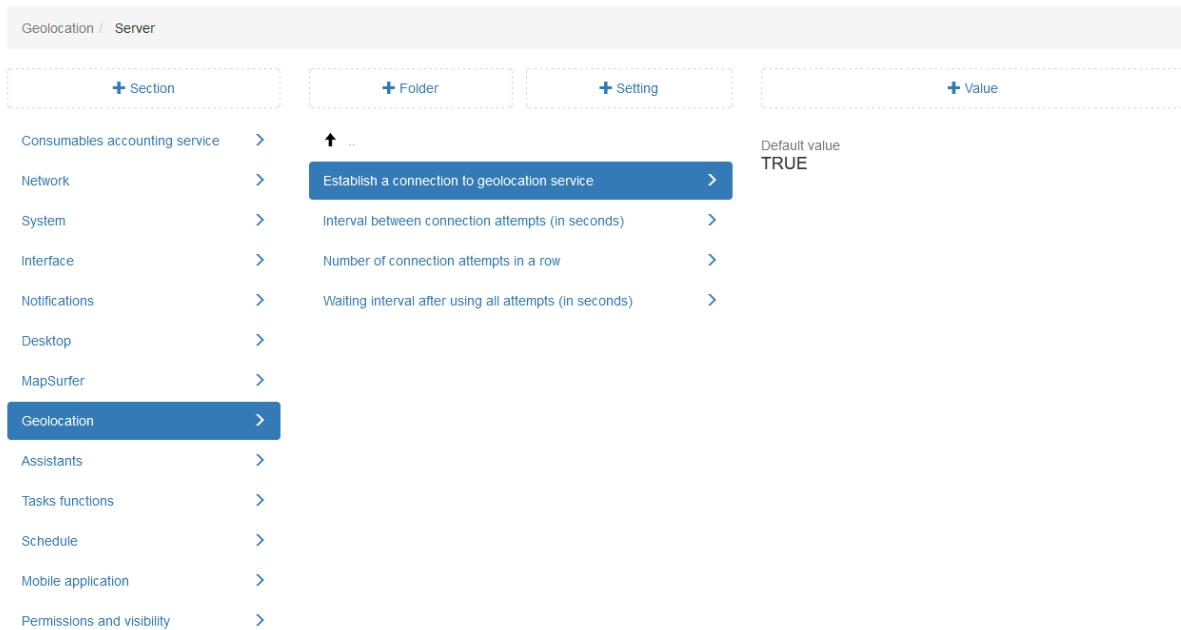
Fig. 2.224: Establishing a connection to geolocation service

**Interval between connection attempts (in seconds)**

If the connection to the location tracking service fails, the next attempt is made after the time specified in this setting (Fig. 2.225).
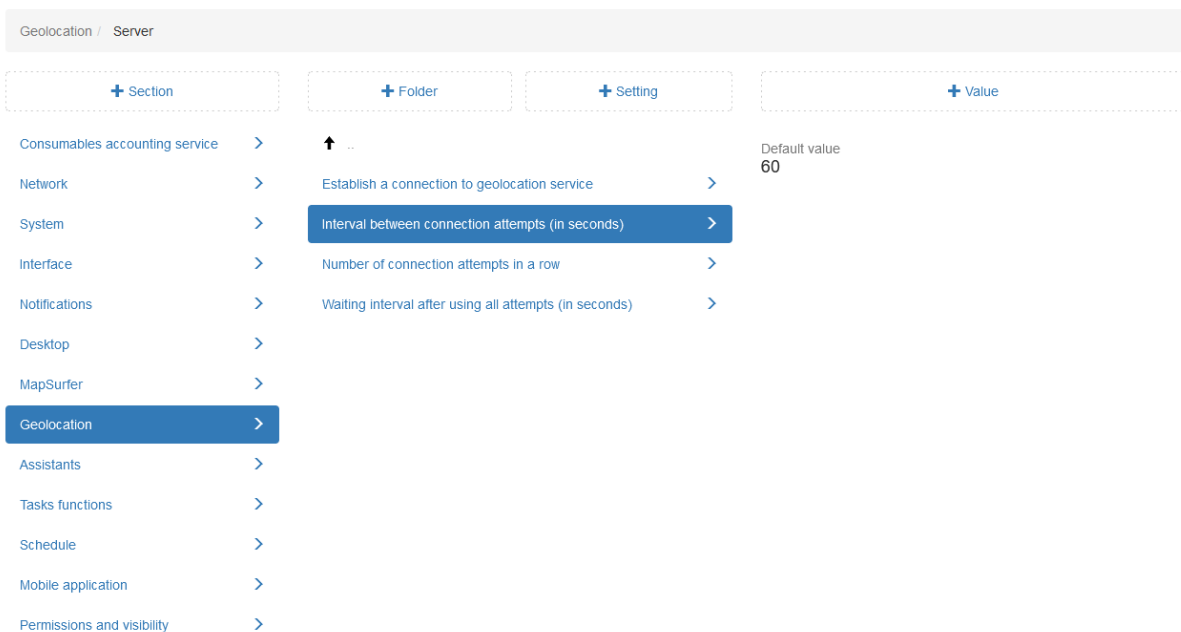


Fig. 2.225: Interval between connection attempts

**Number of connection attempts in a row**

Unsuccessful connection attempts are combined into groups. The maximum number of attempts in a group is specified in this setting (Fig. 2.226).
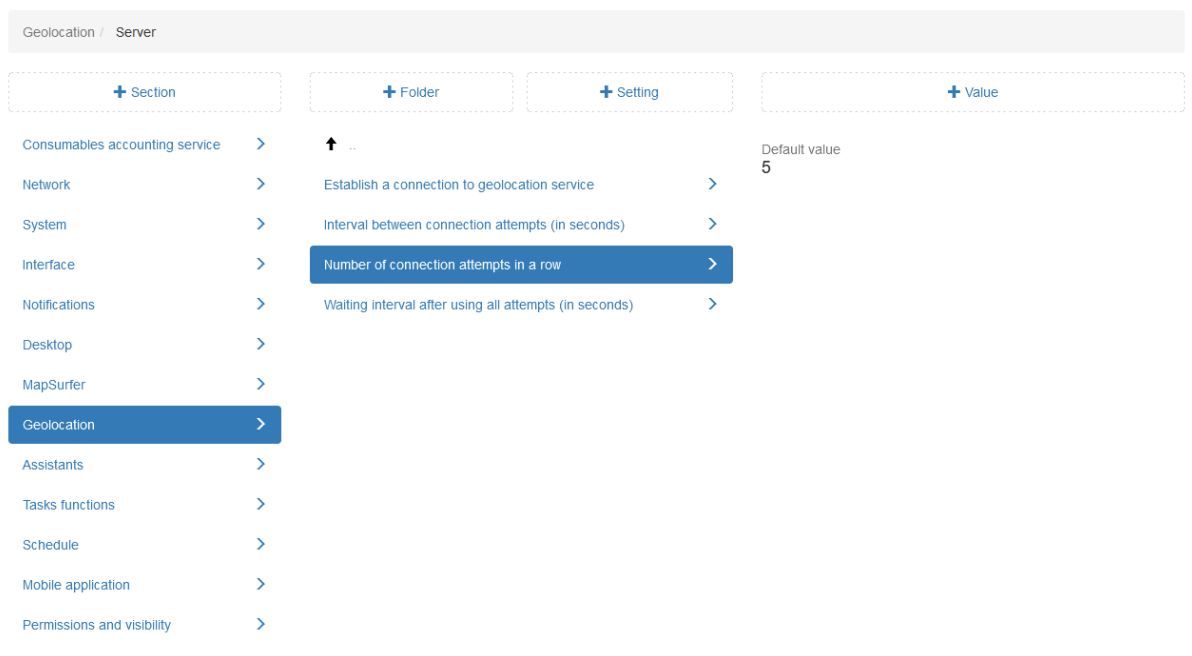


Fig. 2.226: Number of connection attempts

**Waiting interval after using all attempts**

After using all the attempts in a group, the next group of attempts is made after the time specified in this setting (Fig. 2.227).
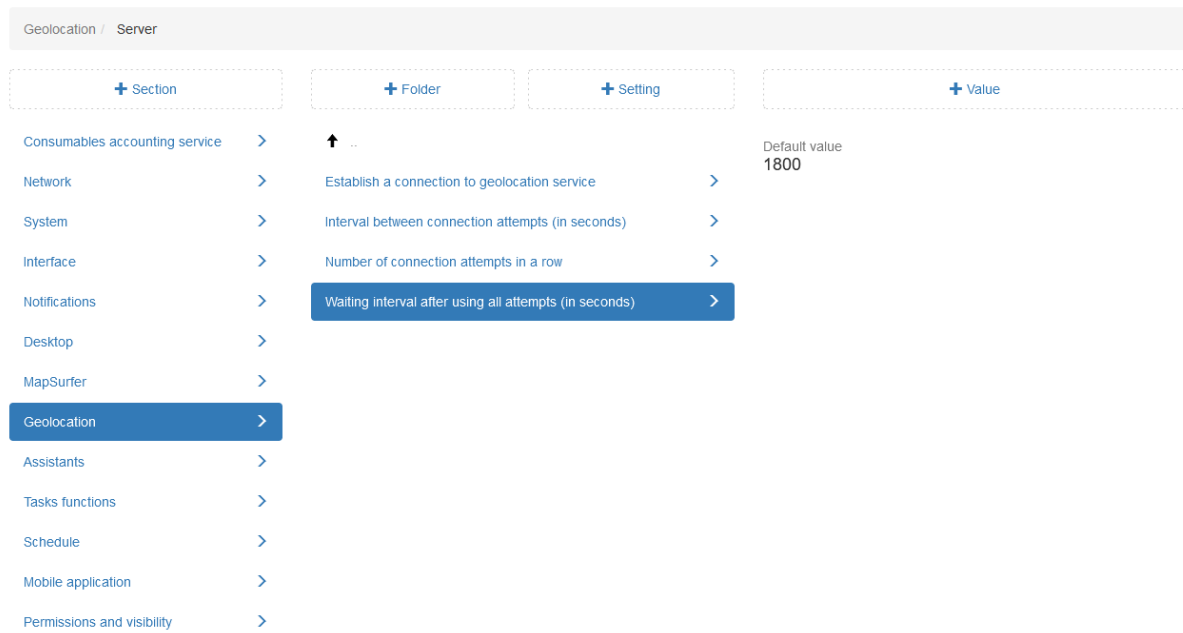
Fig. 2.227: Waiting interval after using all attempts

### 2.3.3.7.9 "Assistants" section

The section allows users to customize existing algorithms that perform certain actions (Fig. 2.228). You can set a new value by selecting the setting, then clicking "+ Value". In the window that opens, enable/disable the toggle switch or enter the required name, and then fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
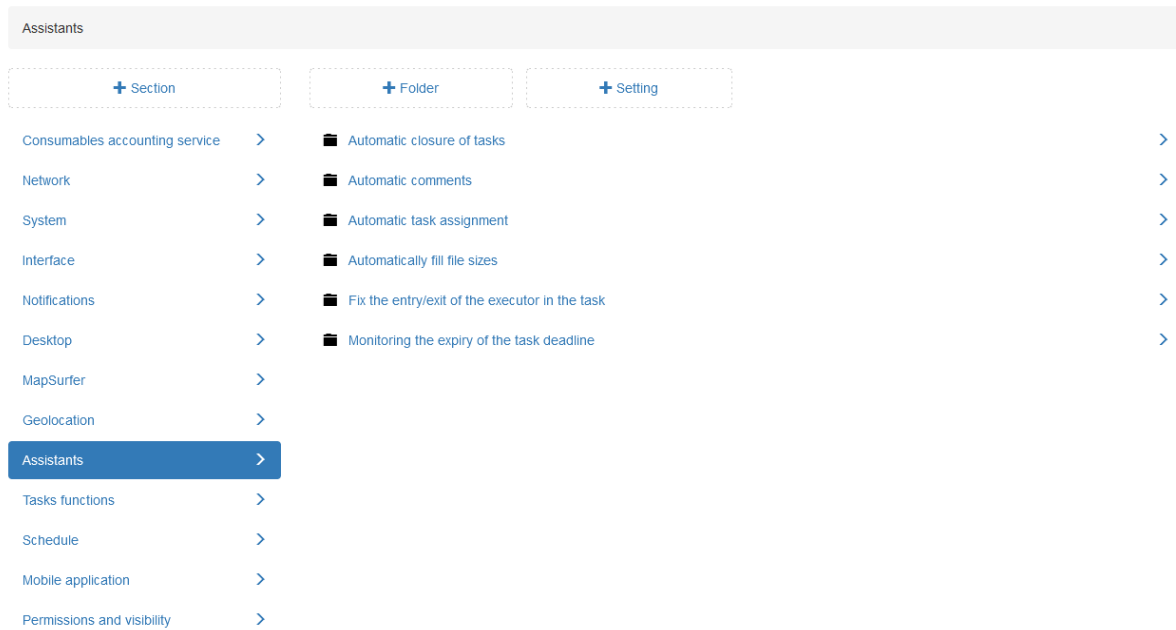
Fig. 2.228: "Assistants" section

## Automatic closure of tasks

A setting that allows users to transfer the task automatically at the closing step to the "Completed" stage. This setting is disabled by default (Fig. 2.229).
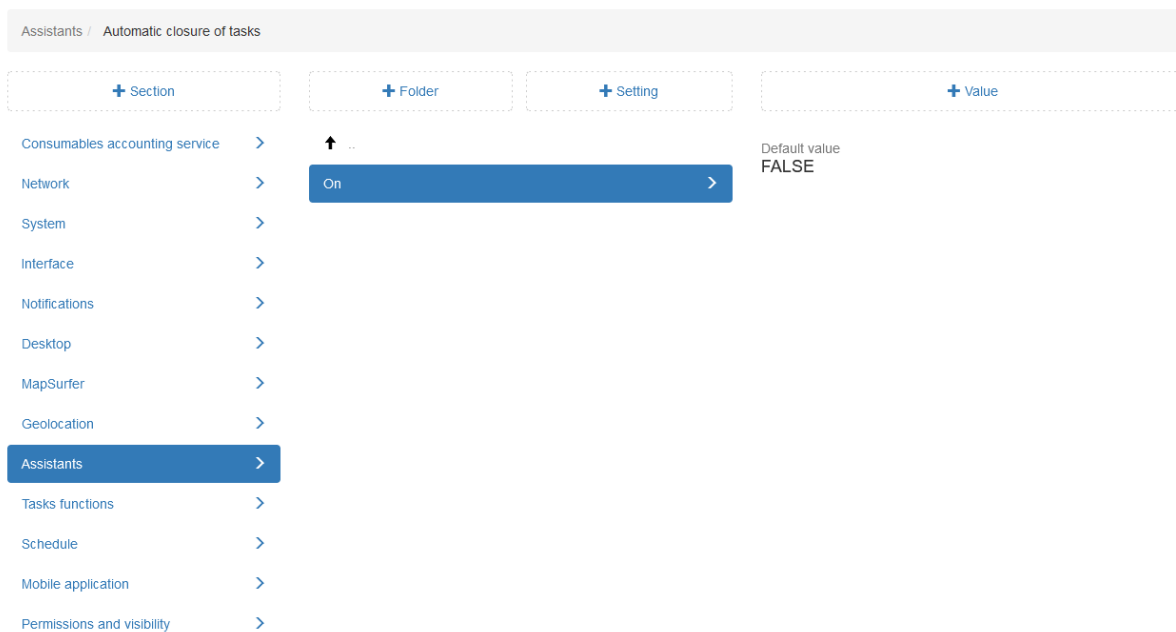


Fig. 2.229: Automatic task closing

---

## Automatic comments

This setting allows adding comments about attaching a file to the history at the time of task creation. The setting is enabled by default (Fig. 2.230).
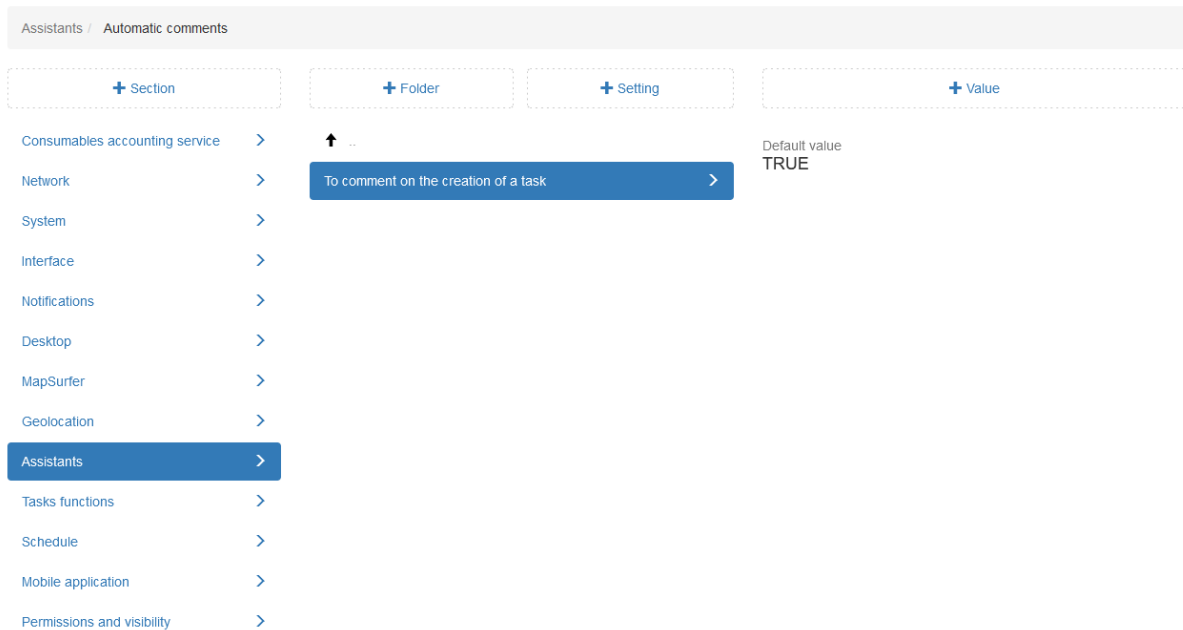


Fig. 2.230: Automatic comments

## Automatic task assignment

This setting is responsible for automatic task assignment. Auto assignment is enabled by default only for the "Client" role, for others this setting is not available (Fig. 2.231).
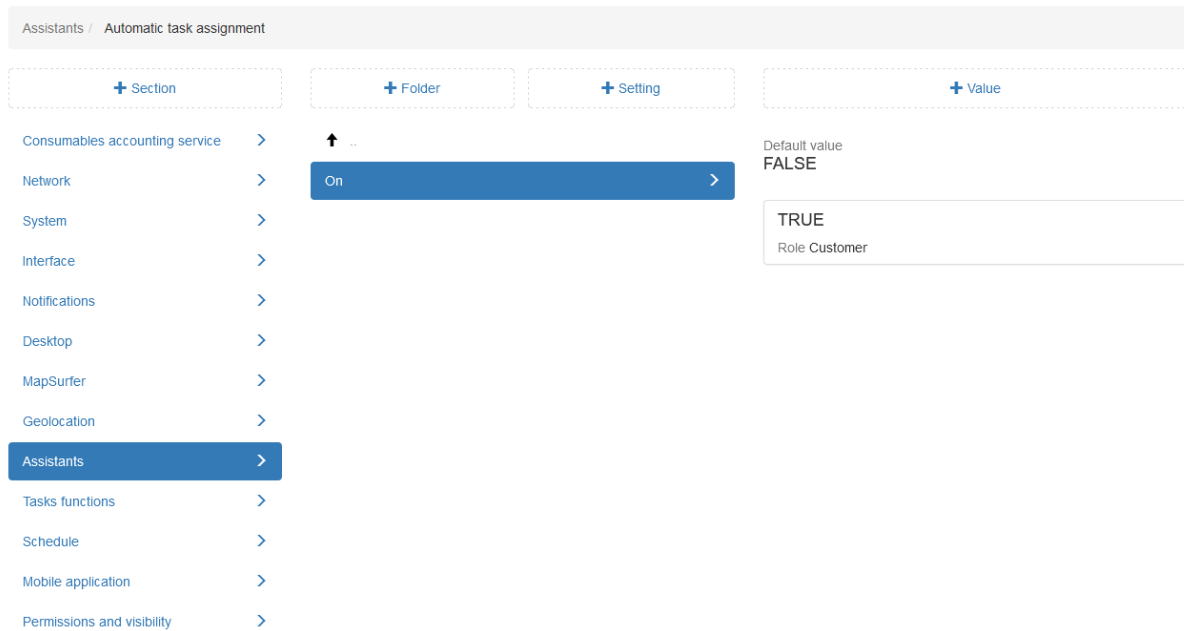
Fig. 2.231: Automatic task assignment

## Automatically fill file sizes

This setting allows filling in the "File size" column in the database for old files. To start the process of gradual filling with data, enable this setting. It loads the specified number of files from the database in the **Number of files to process at a time** setting with the frequency specified in the **Startup period** setting. It searches for them on the disk and collects information by the size. This setting is disabled by default (Fig. 2.232).
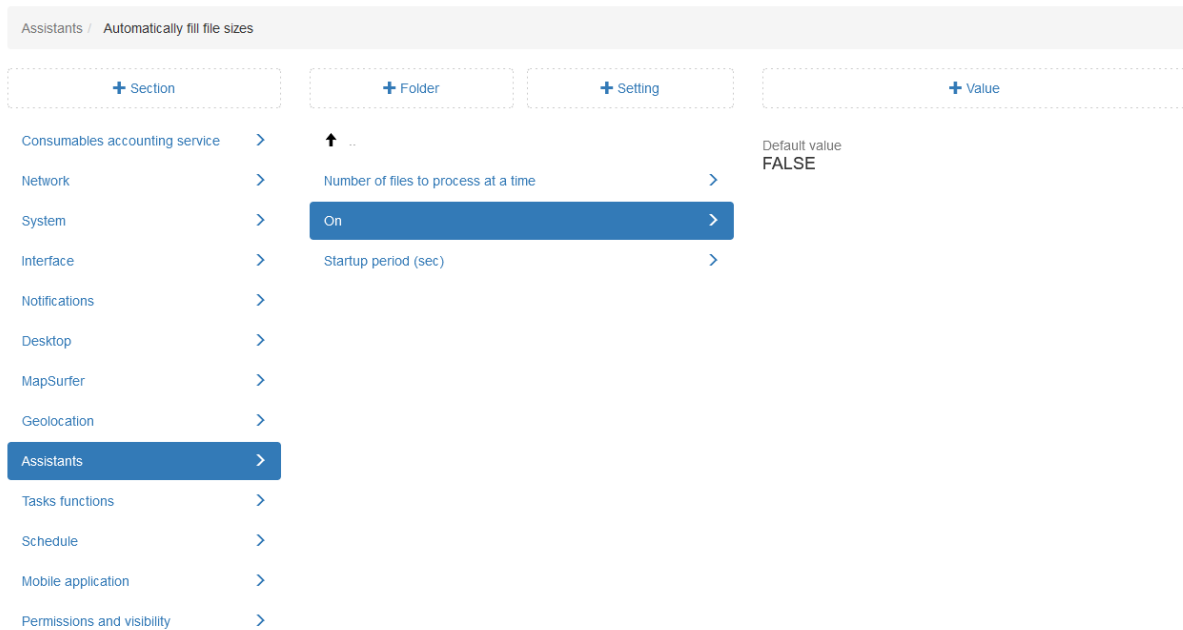
Fig. 2.232: Automatic filling of file sizes

## Fix the entry/exit of the executor in the task

Cerebellum has a mechanism for controlling the entry/exit of executors into/from the task zones. Zones are defined as circles around the task points. To ensure the correct work of the mechanism, enable the connection with the location determination service.

The zone entry control works when the following conditions are met:

- the zone entry control mechanism is enabled;

- the executor's location tracking is enabled;

- the task has a point;

- the task is not a template;

- the task is in the "In progress" or in the "Completed" step, but the executor is in the task zone.

> **Attention:** If the task is completed, but the executor is in the zone, the control mechanism waits for the executor to leave the zone and then stops monitoring the task. However, this does not refer to the tasks that are in the "Completed" step at the time of launching Cerebellum. For such tasks, it is impossible to determine correctly whether the executors are in the zone or not, so their entry/exit is not tracked.

If the conditions described above are met for the task and the executor enters/exits the task zone, the following happens:

- a record is added to the corresponding table with a fixation of:

  – the user;

  – the task point;

  – date and time;

  – entry/exit sign;

  – indication of the event occurrence directly at the moment of assigning to executor;

- one of the following comments is created on behalf of the executor in the task:

  – "The executor has left the task area";

  – "The executor has entered the task area";

  – "The executor is out of range of the task" if the event occurred at the moment of assignment to the executor or at the moment of changing the zone radius setting;

  – "The executor is inside the zone of the task" if the event occurred at the moment of assignment to the executor or at the moment of changing the zone radius setting.

By default, this setting is enabled (Fig. 2.233).
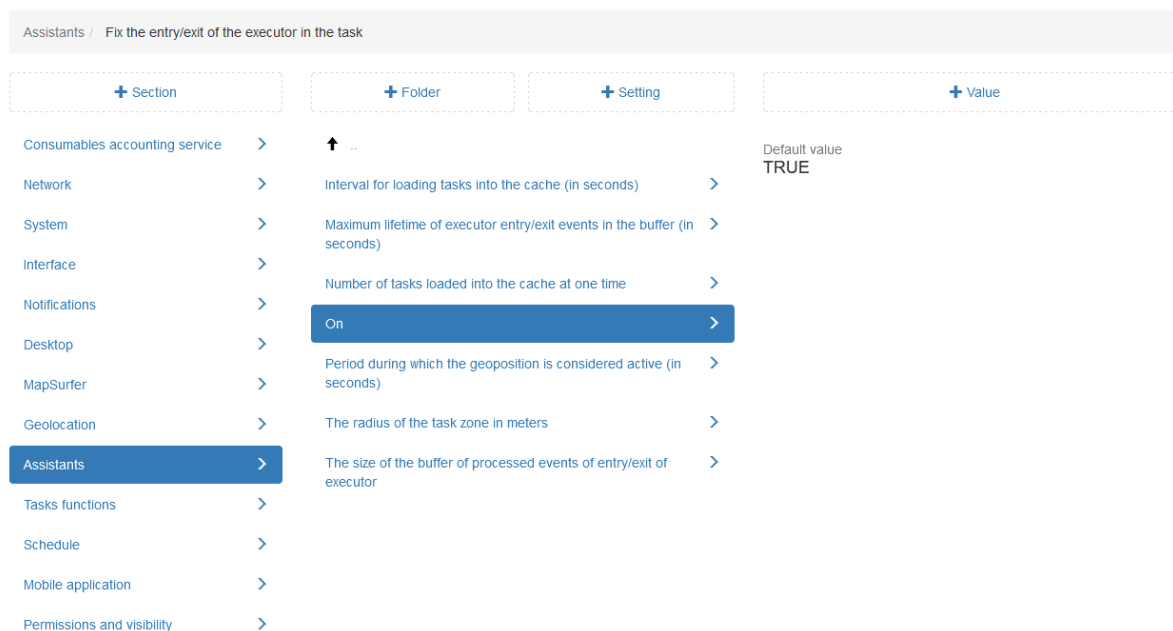


Fig. 2.233: Fixing the entry/exit of the executor in the task

**Interval for loading tasks into the cache (in seconds)**

The setting controls the process of filling the internal cache with tasks in the mechanism of tracking the user entry/exit from the task zone.

Namely, it specifies how often the next set of tasks should be loaded from the database into the cache (Fig. 2.234).
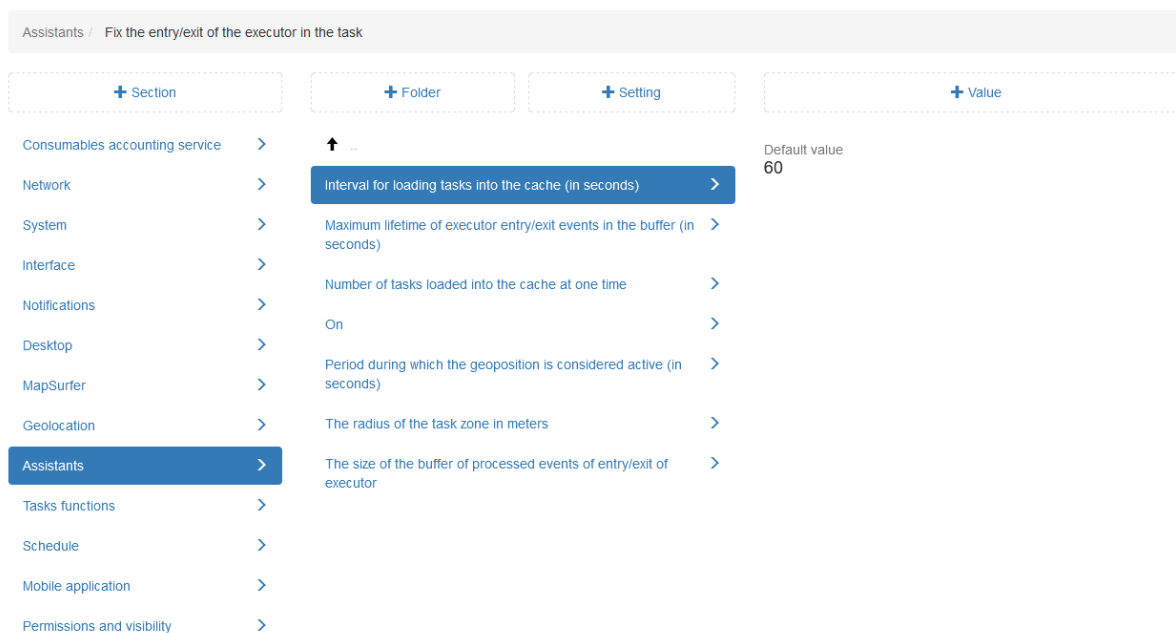


Fig. 2.234: Interval for loading tasks into the cache

**Maximum lifetime of executor entry/exit events in the buffer (in seconds)**

The engineering setting. This value is for developer use only during debugging. It specifies the frequency of sending events from the buffer for processing regardless of buffer fullness. The timeout is set for the timely processing of events in case the entry/exit event from the task zone occurs very rarely (Fig. 2.235). The default is 1 second.

Fig. 2.235: Maximum lifetime of executor entry/exit events in the buffer (in seconds)

**Number of tasks loaded into the cache at one time**

To track the entry/exit of executors from the task zone, Cerebellum uses a task cache. It stores the part of the information necessary to calculate the entry and exit events. When Cerebellum is started, the cache is filled with information. However, this happens gradually to avoid excessive load on the server. With a certain periodicity, Cerebellum saves the next tasks in the cache, in descending order of their IDs. This setting specifies how many tasks are loaded from the database into the cache at once (Fig. 2.236).

Fig. 2.236: Number of tasks loaded into the cache at once

**Period during which the geoposition is considered active**

The setting sets the duration for which the user's geoposition is considered active. The default is 3 hours (Fig. 2.237).



Fig. 2.237: Period during which the geoposition is considered active

**The radius of the task zone in meters**

You can set the distance from the task point at which the executor is considered to have entered or left the task zone. The default is 75 meters (Fig. 2.238).
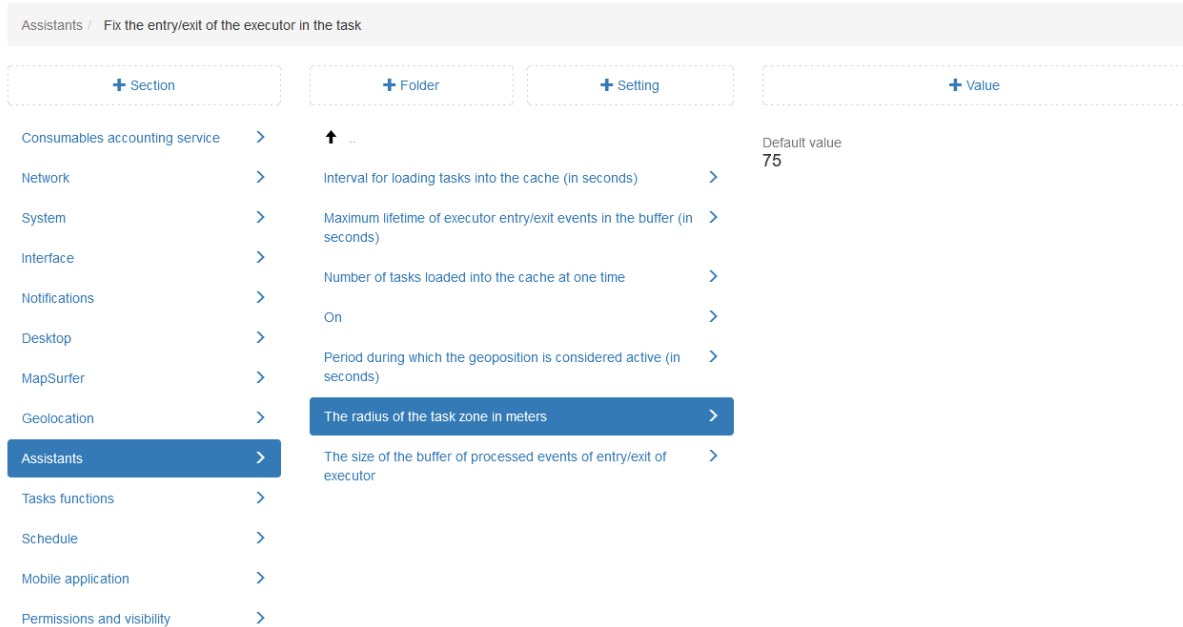


Fig. 2.238: Task zone radius

> **Attention:** If you change this setting while Cerebellum is running, a large number of executors may get in/out of the task radius. This can severely overload the system by causing a mass addition of comments.

**The size of the buffer of processed events of entry/exit of the executor**

Engineering setting. This value is for developer use only during debugging. There can be a large number of tasks in the system, where the event of entering/leaving the zone of the task occurs. For all these tasks, the system has to add the appropriate comment. First of all, the system takes the task from the database, since only its id is stored in the cache. Performing this operation one by one for all tasks can cause a large load on the database. Therefore, the buffer of processed events is used. This setting specifies the size of the buffer (Fig. 2.239). The default is 20 events.

Fig. 2.239: The size of the buffer of processed events of entry/exit of the executor

## Monitoring the expiry of the task deadline

These settings allow recording the expiration of the task in the task history. By default, this setting is enabled (Fig. 2.240).

Expiration control functions under the following conditions:

- the "Control of the expiration of tasks" setting is enabled;

- the task is not a template;

- the task execution deadline is in the future;

- the task is at the "In progress" step.
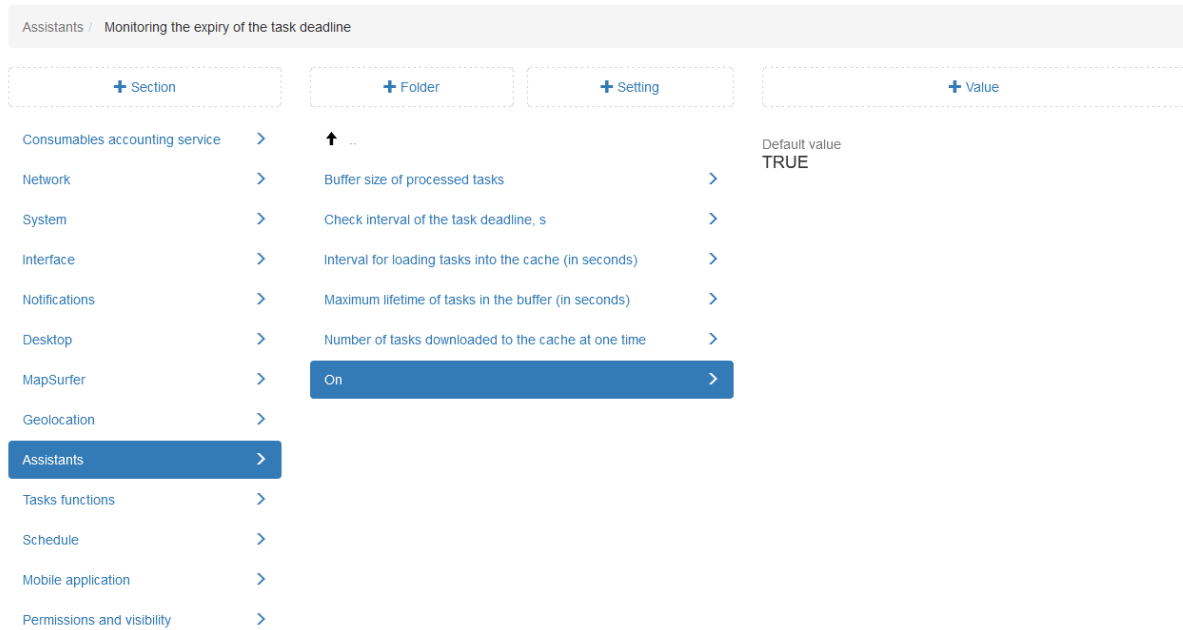
Fig. 2.240: Deadline control for tasks

---

**Attention:** The deadline is monitored regardless of the value of the setting. The value only affects whether events of deadlines expiration are recorded in the database or in comments in the task.

---

**Buffer size of processed tasks**

Engineering setting. This value is for developer use only during debugging. There may be a large number of tasks in the system with the same execution deadline, such as scheduled tasks. Upon expiration, the system has to change the corresponding column and add a comment in each of these tasks. First of all, the system takes the task from the database, since only its ID is stored in the cache. Performing this operation one by one for all expired tasks can cause a large load on the database. Therefore, there is a buffer of expired tasks that fills up as the due date arrives. Expired tasks from the buffer are sent for processing only after the buffer is fully filled, as well as on timeout. This setting specifies the size of the buffer (Fig. 2.241). The default is 50 tasks.

---

Fig. 2.241: Buffer size of processed tasks

**Check interval of the task deadline**

This setting specifies how often Cerebellum checks for expired tasks (Fig. 2.242). The default is 30 seconds.



Fig. 2.242: Expiration check interval

**Interval for loading tasks into the cache (in seconds)**

This setting controls the process of filling the internal cache of the deadline control mechanism with tasks. It specifies how often the next sets of tasks should be loaded from the database into the cache (Fig. 2.243).



Fig. 2.243: Interval for loading tasks into the cache

**Maximum lifetime of tasks in the buffer (in seconds)**

Engineering setting. This value is for developer use only during debugging. It specifies the frequency of sending expired tasks for processing from the buffer regardless of the buffer size. The timeout is used for timely processing of expired tasks if the expiration event occurs very rarely (Fig. 2.244). The default is 1 second.
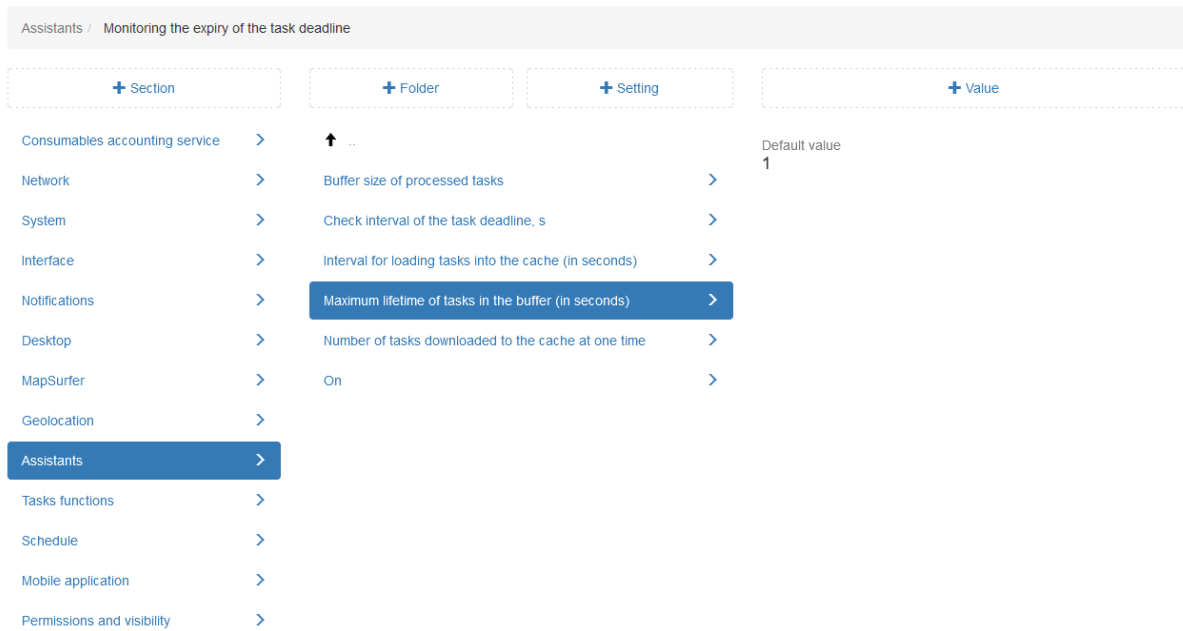
Fig. 2.244: Maximum lifetime of tasks in the buffer

**Number of tasks downloaded to the cache at one time**

To track the deadline for execution, Cerebellum uses a cache that stores the mapping between tasks and their execution deadlines in ascending order. The cache is populated and updated when tasks are created, modified, and deleted, as well as when Cerebellum starts to load information about existing tasks. Such a load happens gradually to avoid excessive server load. With a certain periodicity, Cerebellum saves the next set of tasks in the cache in ascending order of their execution deadlines. This setting specifies how many tasks are loaded from the database into the cache at once (Fig. 2.245).

Fig. 2.245: Number of tasks downloaded to the cache at once

### 2.3.3.7.10 "Task functions" section

This section contains settings for server operation with tasks (Fig. 2.246). You can set a new value for the setting by selecting it, then clicking "+ Value". Turn on/off the toggle switch or enter a new value in the opened window and fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
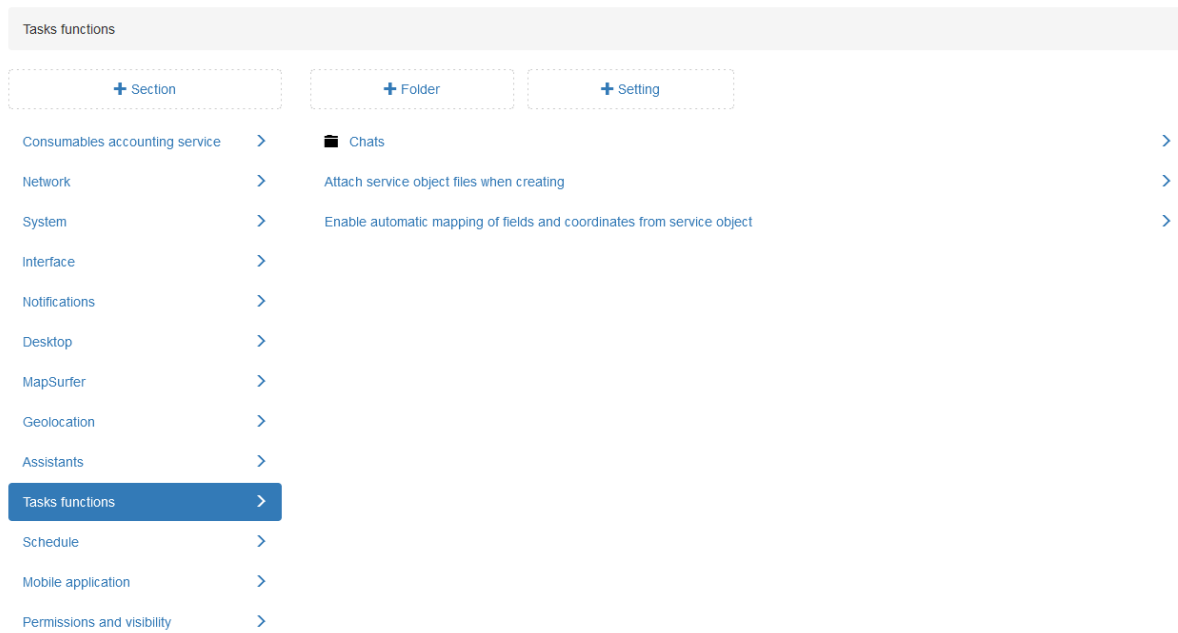
Fig. 2.246: "Task functions" section

## Chats

### Number of chats loaded at a time when the application starts

Engineering setting. Only developers change it during debugging. It specifies how many comments are loaded at once when starting (Fig. 2.247).

Fig. 2.247: Number of chats loaded at a time when the application starts

### Attach service object files when creating a task

Setting that allows users to copy files of the related service object when creating a task. By default, this setting is disabled (Fig. 2.248).



Fig. 2.248: Attachment of service object files

### Enable automatic mapping of fields and coordinates from the service object

Server setting that allows users to enable copying of fields and coordinates from the service object to custom fields when creating a task. By default, the setting is enabled (Fig. 2.249).
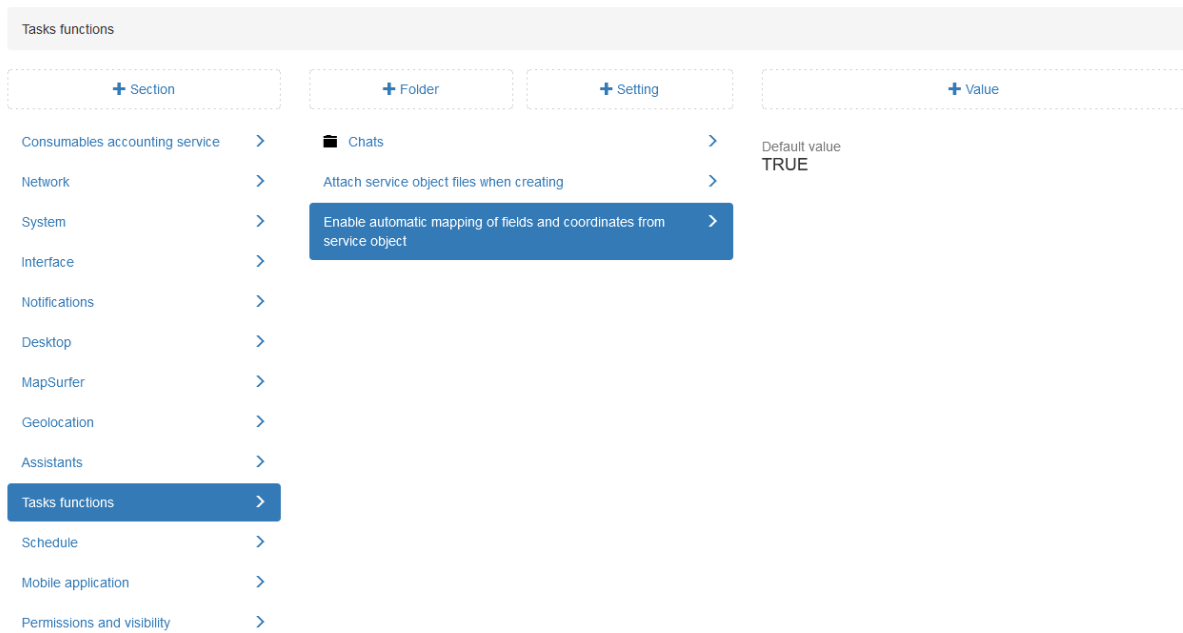


Fig. 2.249: Automatic mapping of fields and coordinates from the service object

### 2.3.3.7.11 "Schedule" section

This section specifies the maximum number of simultaneously created tasks by schedule. The default setting is 2 tasks. This means that no more than 2 schedule tasks are created at once (Fig. 2.250).
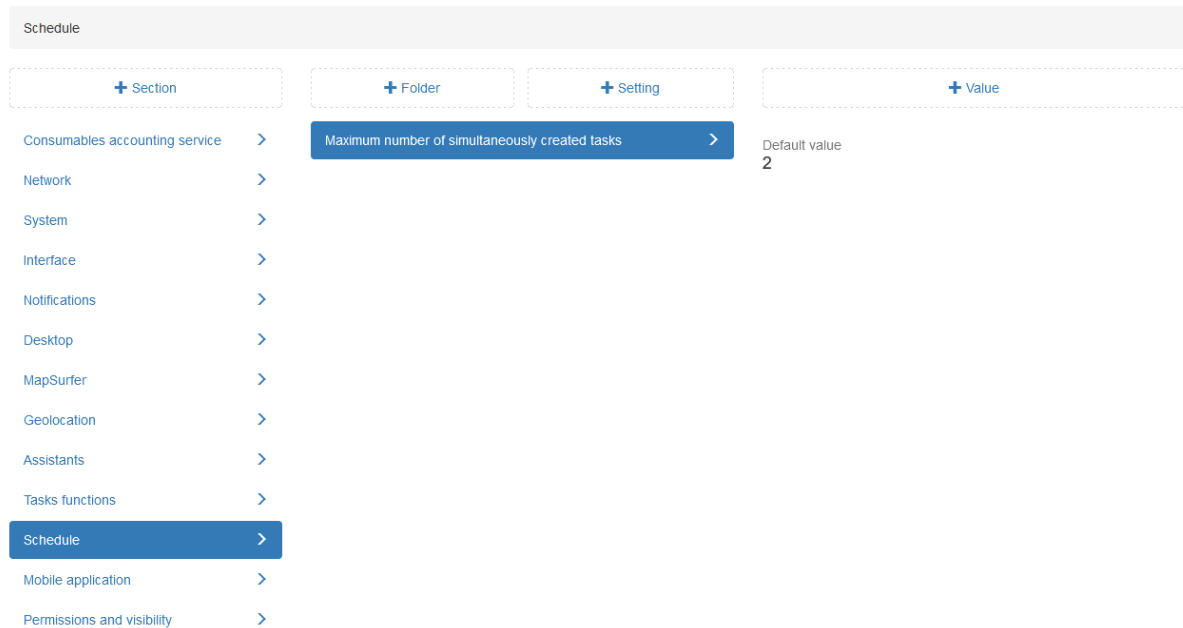
Fig. 2.250: "Schedule" section

#### 2.3.3.7.12 "Mobile application" section

The "Mobile application" section allows configuring additional options for working in ActiveMap Mobile (Fig. 2.251). You can set a new value by selecting the setting, then clicking "+Value". Enable/disable the toggle switch or enter the required name in the opened window and fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
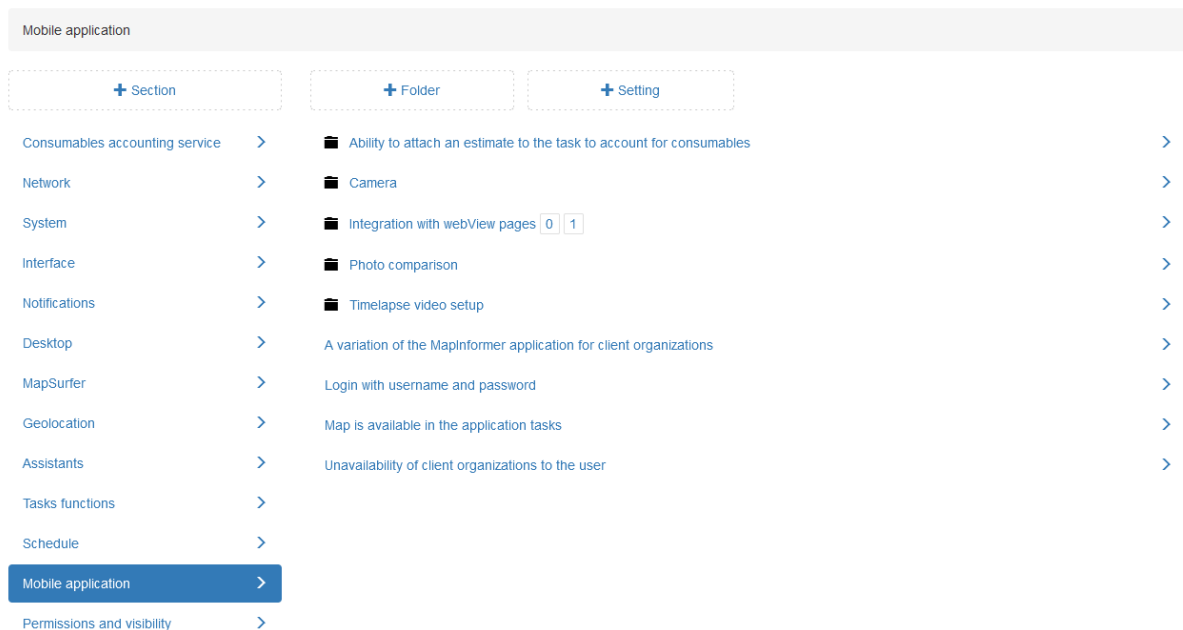


Fig. 2.251: "Mobile application" section

### Ability to attach an estimate to the task to account for consumables

In this folder, you can add the invoice module (the "Invoice" button in the task card).

**Paragraph text**

This setting allows you to set the name of the module in the application (Fig. 2.252).
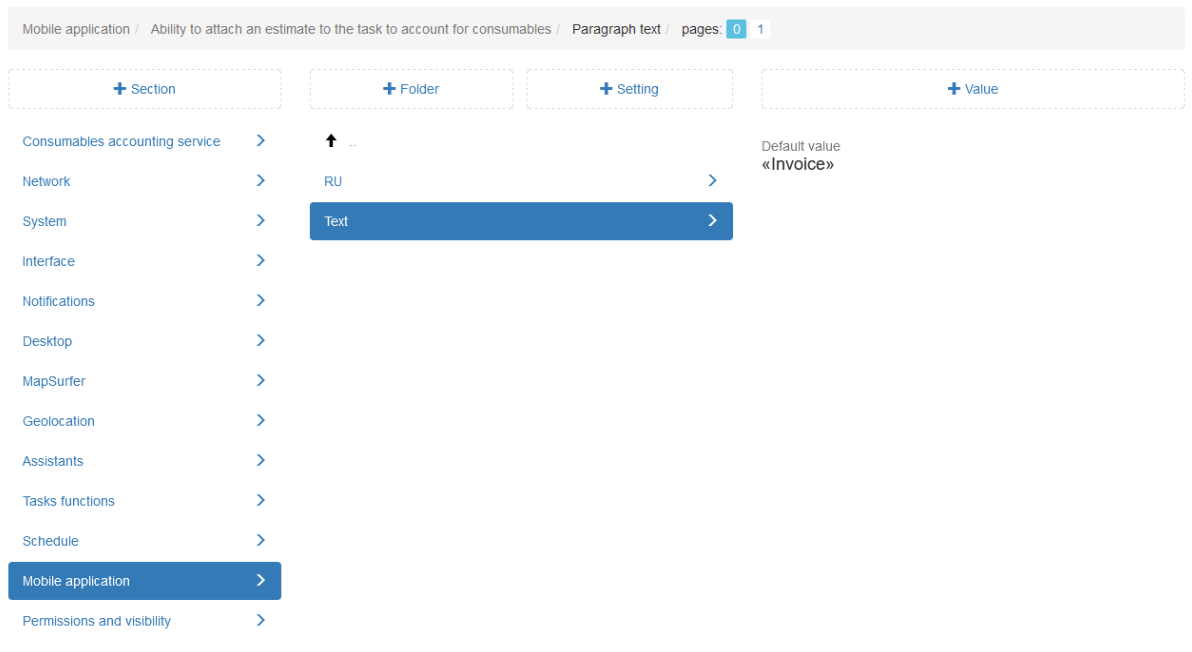


Fig. 2.252: Specifying the name of the module

**Activate**

This setting allows you to activate the invoice module. By default, this setting is disabled (Fig. 2.253).
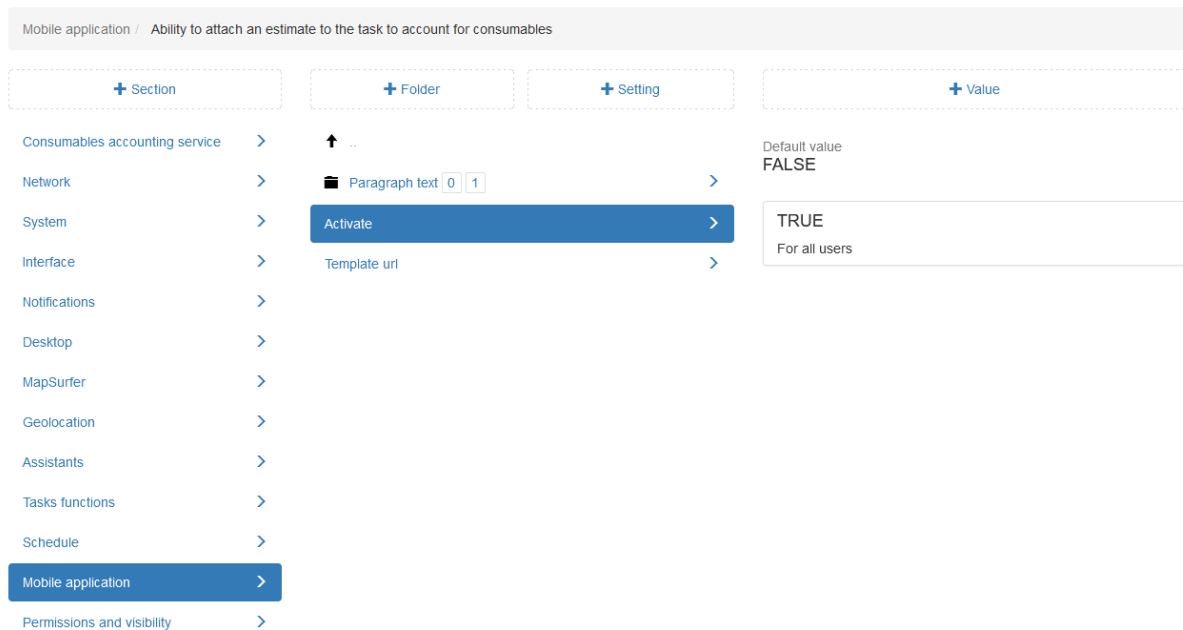
Fig. 2.253: Activating the invoice module

### URL template

This setting shows the web address of the page that opens when you click the "Invoice" button.

### Camera

In this folder, you can add settings for the custom camera in the mobile application.

**Consider the distance from the task point**

If both this setting and **"Prohibit photographing"** are enabled and the user is far from the task point, a message appears when trying to add a photo. It indicates that a photo cannot be taken due to a large location error. If **"Prohibit photographing"** is disabled and this setting is active, you still can take a photo, but the geolocation icon turns red. By default, this setting is disabled (Fig. 2.254).
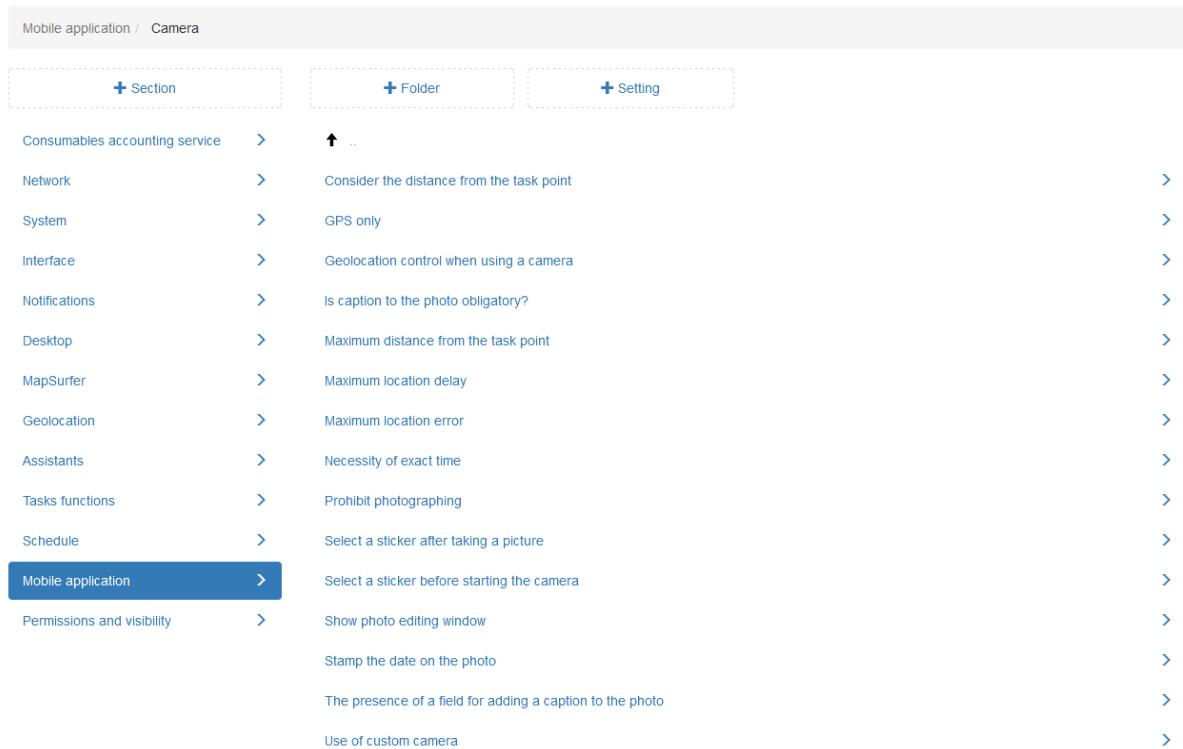
Fig. 2.254: Distance from the task point

**GPS Only**

The setting allows users to set the location source. If the setting is enabled, only information from the GPS source is used to determine the location. When the setting is off, coordinates can also be determined by cellular networks. By default, this setting is disabled (Fig. 2.255).

Fig. 2.255: Coordinate source

**Geolocation control when using a camera**

When the camera is launched, the application begins to search for the user's current location. If the location cannot be found, the photo has no coordinates. This setting prevents taking photos until the user's location is determined. By default, this setting is enabled. If a value is added that prohibits controlling the location when using the camera, the application does not prevent taking photos, even if the coordinates have not yet been determined (Fig. 2.256).

Fig. 2.256: Geolocation control when using a camera

**Is caption to the photo obligatory?**

This setting makes it mandatory to add a caption to the attached photo. Thus, the user cannot attach a photo to a task without adding a caption first. By default, this setting is disabled (Fig. 2.257).

Fig. 2.257: Add a caption to the photo

**Maximum distance from the task point**

This setting sets the maximum distance (in meters) from the task point at which a photo can be taken (Fig. 2.258).

Fig. 2.258: Maximum distance from the task point

**Maximum location delay**

This setting specifies the time (in milliseconds) of the relevance of the device's geolocation after losing the GPS signal when using the built-in camera. If the location was obtained more than the specified value of milliseconds ago and the "Prohibit photographing" setting is enabled, the application does not allow taking a photo until it receives a point that meets the requirements (Fig. 2.259).

Fig. 2.259: Maximum location delay

**Maximum location error**

The setting allows specifying the permissible error (in meters) of determining the device's geolocation when using the built-in camera. If the received location has a larger error than the set value and the "Prohibit photographing" setting is enabled, then the application does not allow taking a photo until it receives a point that meets the requirements (Fig. 2.260).

Mobile application / Camera

| + Section | + Folder | + Setting | + Value |

Consumables accounting service  >

Network  >

System  >

Interface  >

Notifications  >

Desktop  >

MapSurfer  >

Geolocation  >

Assistants  >

Tasks functions  >

Schedule  >

**Mobile application**  >

Permissions and visibility  >

↑ ..

Consider the distance from the task point  >

GPS only  >

Geolocation control when using a camera  >

Is caption to the photo obligatory?  >

Maximum distance from the task point  >

Maximum location delay  >

**Maximum location error**  >

Necessity of exact time  >

Prohibit photographing  >

Select a sticker after taking a picture  >

Select a sticker before starting the camera  >

Show photo editing window  >

Stamp the date on the photo  >

The presence of a field for adding a caption to the photo  >

Use of custom camera  >

Default value
50

Fig. 2.260: Maximum location error

**Necessity of exact time**

When this setting is enabled, the application does not allow taking a photo until the time is synchronized with accurate time (from the Internet or GPS). Synchronization is required once and is saved until the device is turned off. If this setting is disabled, synchronization is still performed, but it does not prevent taking photos, creating a situation where the device time is attached to the photo. By default, this setting is disabled (Fig. 2.261).

Fig. 2.261: Necessity of exact time

**Prohibit photographing**

The setting prohibits taking a photo if the user is outside the task zone geolocation, or if the geolocation function on the device is turned off, or if the geolocation services cannot determine (with a given accuracy) the location of the device. By default, this setting is disabled (Fig. 2.262).

Fig. 2.262: Prohibition of photography

**Select a sticker after taking a picture**

The setting makes it mandatory to select a sticker after taking a photo. The application does not allow attaching a photo to the task without specifying a sticker first. The sticker selected from the list is added to the taken photo. This setting is disabled by default (Fig. 2.263).

Fig. 2.263: Selecting a sticker after taking a picture

**Select a sticker before starting the camera**

This setting activates the window for selecting a sticker when clicking the "Add photo" button on the task card. You should specify the sticker first and then take the photo. Otherwise, you cannot add the photo. The sticker selected from the list is added to the photo taken. By default, this setting is disabled (Fig. 2.264).

Fig. 2.264: Selecting a sticker before starting the camera

**Show photo editing window**

A setting that controls whether the photo preview and editing window appears when you use the camera. By default, this setting is enabled (Fig. 2.265).

Fig. 2.265: Show photo editing window

**Stamp the date on the photo**

This setting allows adding a date and time stamp to the photo. By default, this setting is enabled ([Fig. 2.266](#)).

Fig. 2.266: Stamp the date on the photo

**The presence of a field for adding a caption to the photo**

The setting allows adding photo descriptions in the photo editor in the mobile application. By default, this setting is enabled (Fig. 2.267).

Fig. 2.267: Adding a caption to a photo

**Use of custom camera**

This setting allows you to prohibit using the standard device camera when working in a mobile application. By default, this setting is enabled. If a value has been added that prohibits the use of a custom camera on the device, then the settings in this folder are ignored (Fig. 2.268).
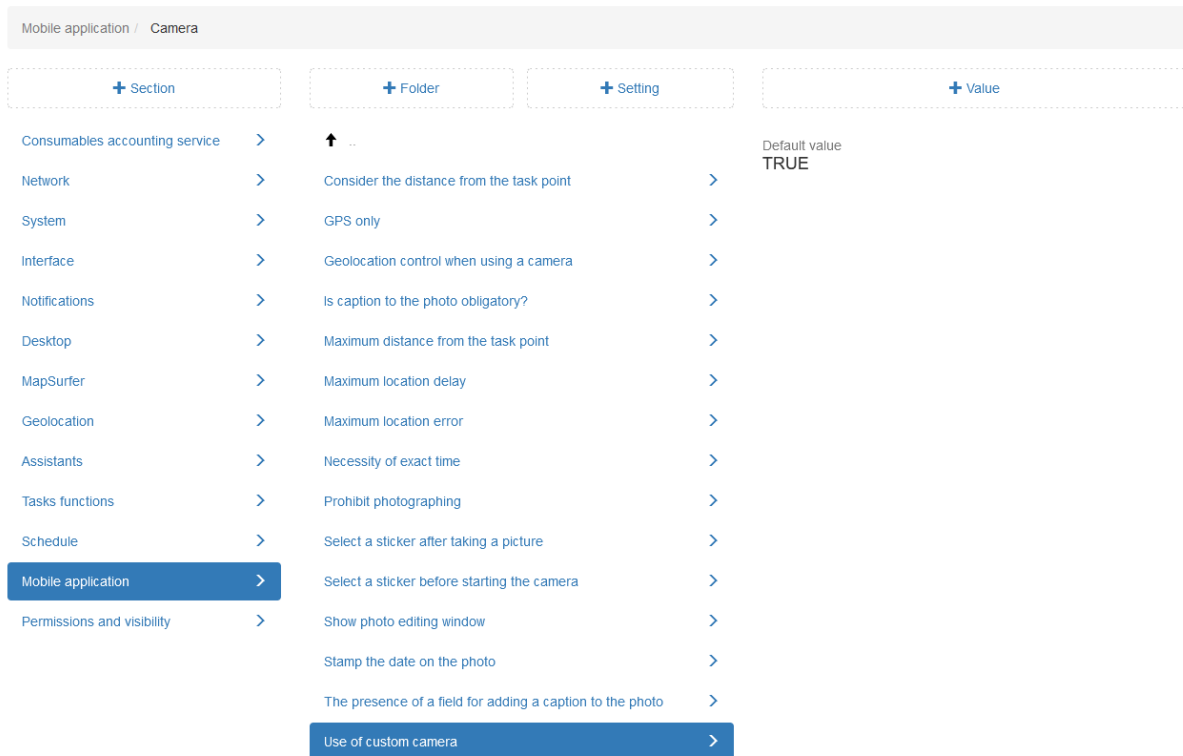
Fig. 2.268: Use of custom camera

### Integration with webView pages

In this folder, you can add sections of third-party web applications and services to the navigation menu of the ActiveMap Informer mobile application.

**Text**

To specify the name of a new section in the ActiveMap Informer mobile application, go to the **"Menu name"** folder, select the **"Text"** setting, and enter the necessary value (Fig. 2.269).
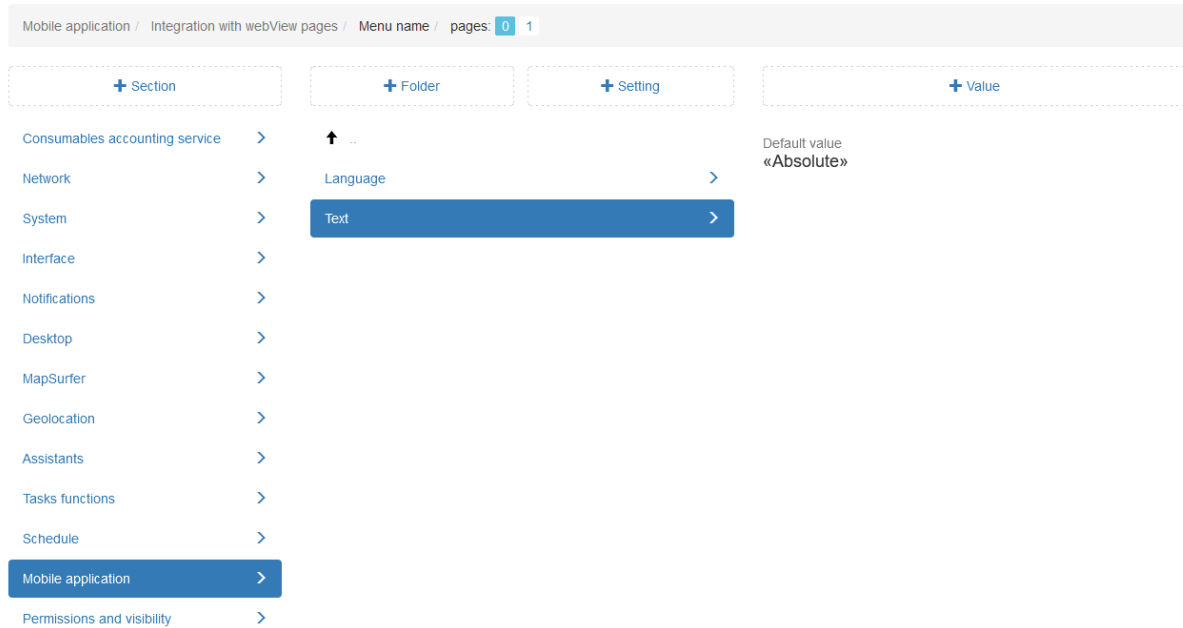
Fig. 2.269: Name of the connected web service

**On**

The setting allows enabling the connection of a third-party service. By default, this setting is disabled (Fig. 2.270).
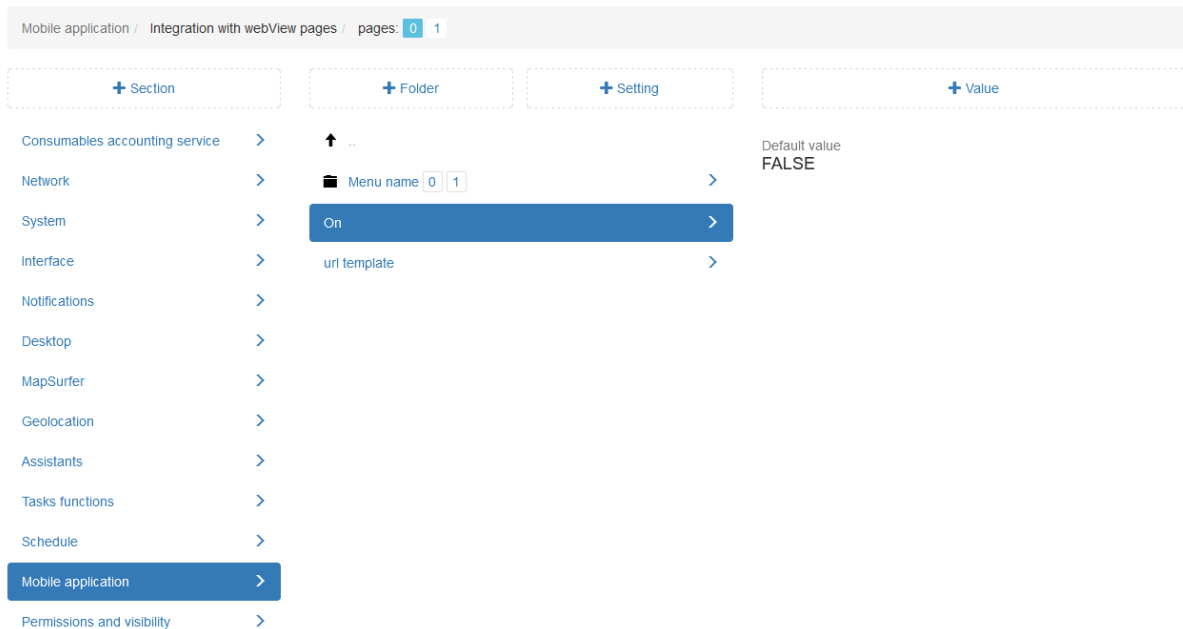


Fig. 2.270: Connecting web applications

**URL template**

In this setting you can specify the address of a third-party web application or service (Fig. 2.271).
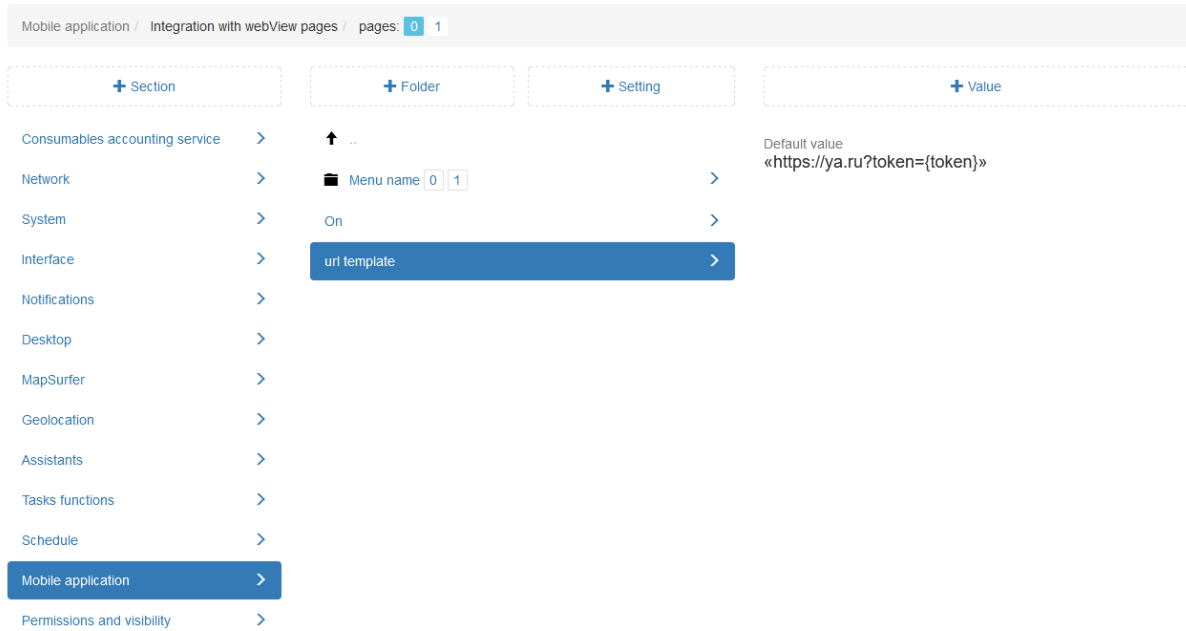
Fig. 2.271: Name of the connected web application

### Timelapse video setup

There is an option to record timelapse videos in the ActiveMap Mobile application for Android. In this mode, the geographic coordinates of the camera are recorded along with the video itself. This allows creating tasks from frames of this video with the attachment of the current frame (as a task photo) and geoposition in ActiveMap Desktop. In this section, you can add settings for recording timelapse videos in the mobile application.

**Allow location recording**

This setting enables the recording of the device's movement and location while recording the current frame. You can play the recording in the ActiveMap Desktop. By default, this setting is disabled (Fig. 2.272).

Fig. 2.272: Allow location recording

### FPS (View)

This setting indicates the frequency of video frame playback per second.
The default setting is 5 video frames (Fig. 2.273).



Fig. 2.273: FPS (View)

### Maximum error in meters

This setting specifies the acceptable coordinate error when recording
timelapse videos. If the coordinate exceeds this error, it is not taken

into account when recording. By default, the permissible error is 100 meters (Fig. 2.274).
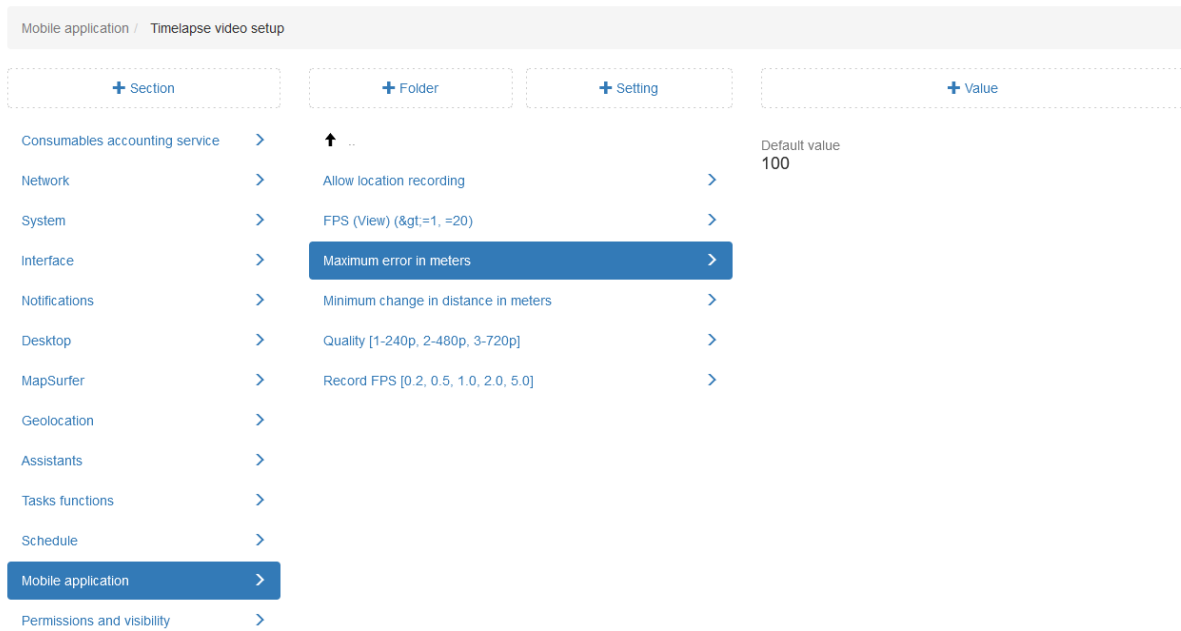


Fig. 2.274: Maximum error in meters

### Minimum change in distance in meters

This setting specifies the minimum distance between location updates when recording a video. The default setting is 5 meters (Fig. 2.275).
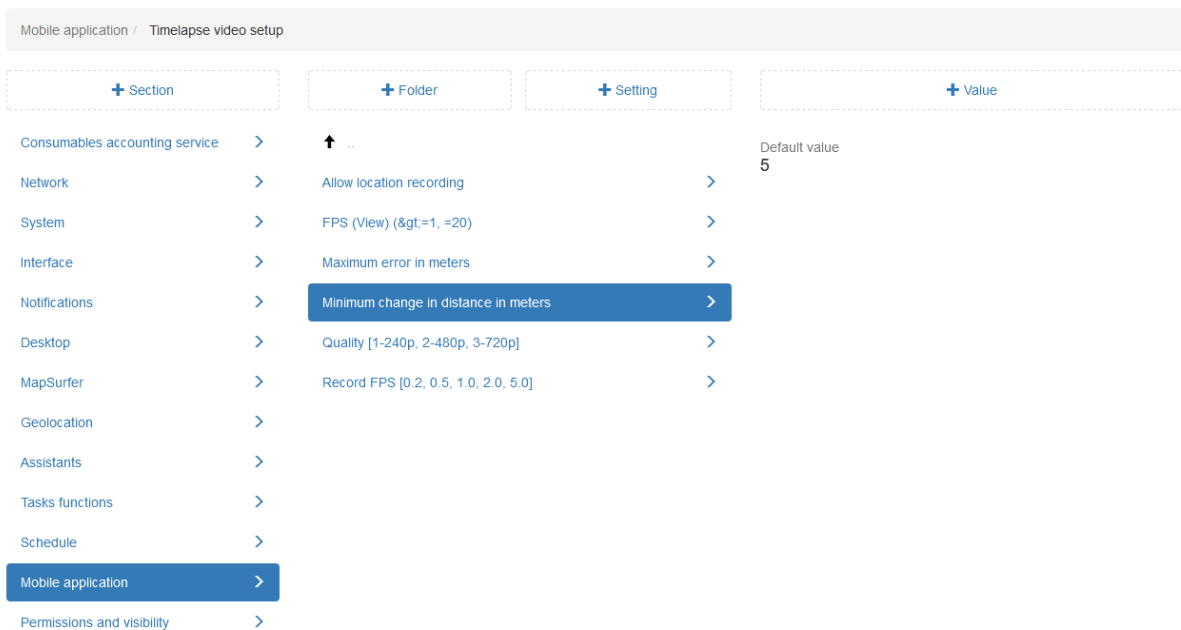


Fig. 2.275: Minimum change in distance in meters

### Quality

This setting allows you to select the quality of recorded video files. The default is 480 pixels (Fig. 2.276).



Fig. 2.276: Video quality

**Record FPS**

The setting specifies how many seconds it takes to record one frame. The default is 2 seconds (Fig. 2.277).



Fig. 2.277: FPS (Record)

### Photo comparison

In this folder, you can add URLs for online and offline neural network models. The selected model compares the sample photo and the result, then displays the percentage of matching in the photo editing window of the mobile application.

**Allowable percentage comparison of two photos**

In this setting, you can specify a number that determines the percentage threshold after which the photo can be considered similar to the sample. If the percentage is higher than the specified value, then the background with the percentage information is colored in green, if lower – in red. If this setting is not filled, the background is not colored (Fig. 2.278).
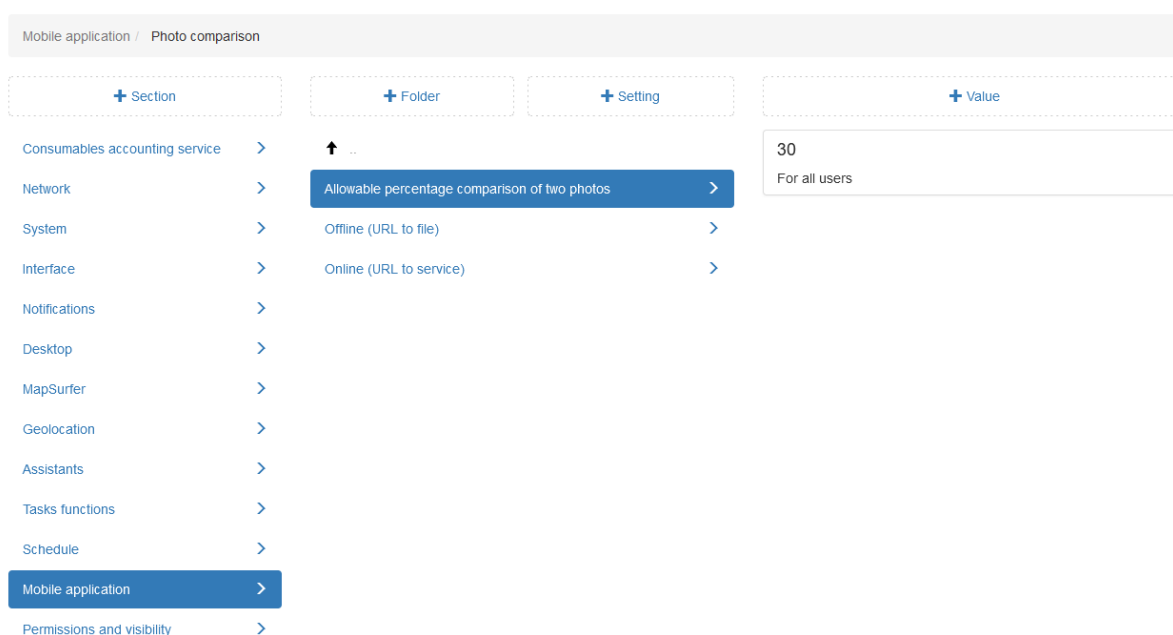


Fig. 2.278: Positive comparison percentage of two photos

**Offline (URL to file)**

The setting allows specifying the URL to the file that is loaded with the reference tables (dictionaries). Once the file is successfully uploaded, the photo editing window displays the percentage of similarity with the sample (Fig. 2.279).

Fig. 2.279: Offline (URL to file)

**Online (URL to service)**

The setting allows specifying the URL of the service that compares the sample photo and the result. If no value is added, then the comparison button is not available in the result photo editing window (Fig. 2.280).



Fig. 2.280: Online (URL to service)

### Login with username and password

The setting activates user authorization in the ActiveMap Informer mobile application using a login and password instead of a phone number. This setting is disabled by default (Fig. 2.281).
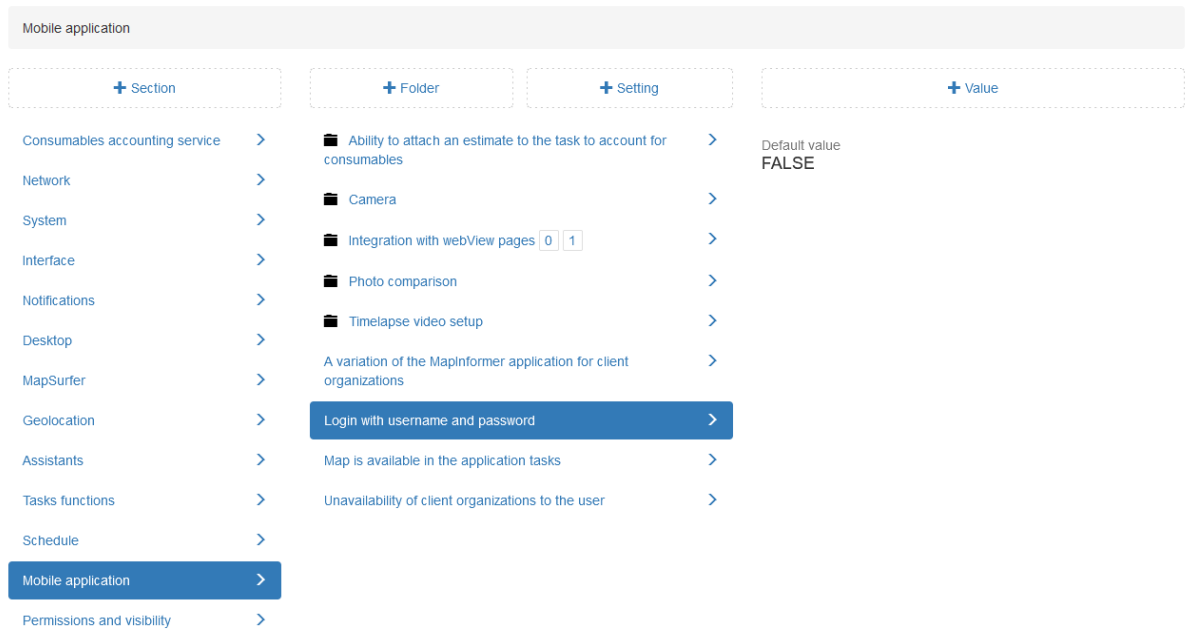


Fig. 2.281: Login with username and password

### Map is available in the application tasks

This setting allows you to enable or hide the map window in tasks in mobile applications. By default, this setting is enabled (Fig. 2.282).

Fig. 2.282: Map availability in the application

## Unavailability of client organizations to the user

The setting is intended for selecting an organization when registering a user in the ActiveMap Informer mobile application. If this setting is enabled, the user cannot select an organization. This setting is disabled by default (Fig. 2.283).



Fig. 2.283: Unavailability of client organizations to the user

### 2.3.3.7.13 "Permissions and visibility" section

The section allows you to add settings to block user actions that can lead to incorrect operation of the system (Fig. 2.284). You can set a new value for each setting in this folder by selecting it, then clicking "+ Value". Enable/disable the toggle switch or enter a new value in the opened window and fill in the remaining fields. For more information about the process of adding a new value, see *"Settings" block* (page 112).
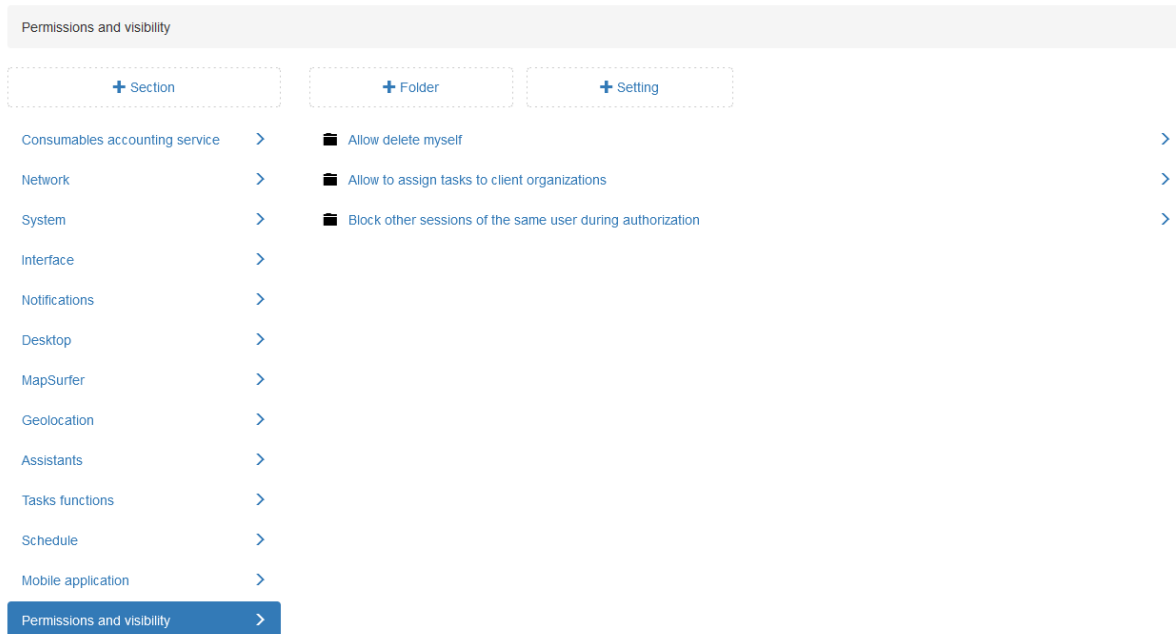


Fig. 2.284: "Permissions and visibility" section

### Allow delete myself

In this setting, you can specify the role, organization, or users who can delete their user accounts in the ActiveMap Mobile iOS application (Fig. 2.285).
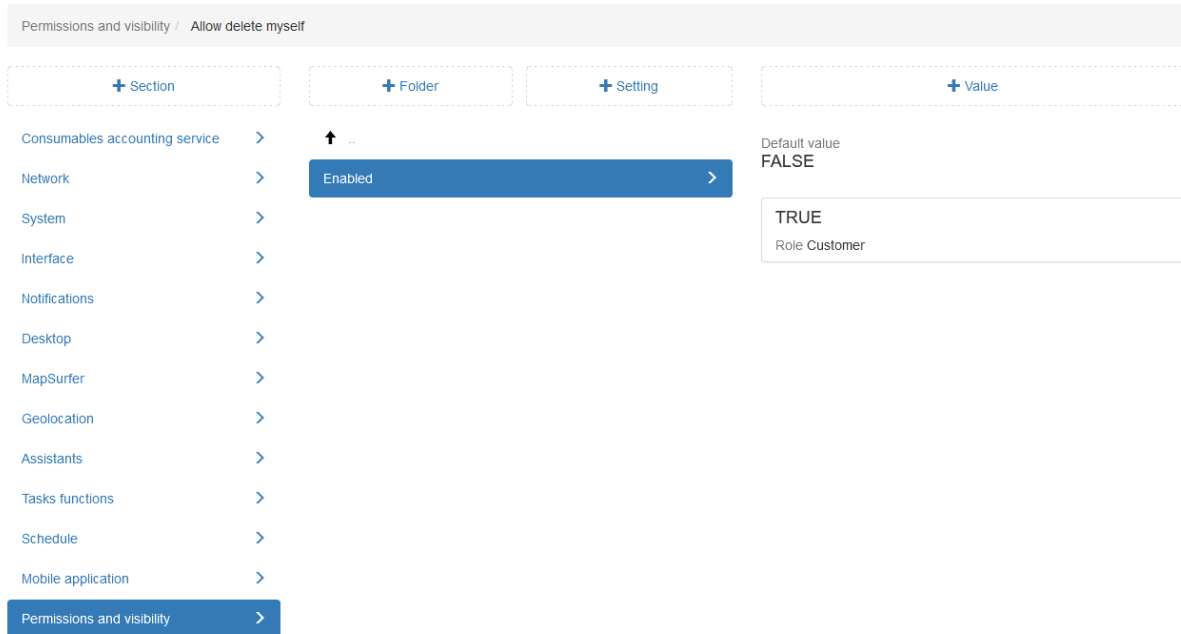
Fig. 2.285: Deleting your account

### Allow to assign tasks to client organizations

In this setting, you can prohibit assigning tasks to a client organization. If you try to assign a task, you see the blocked action information, but the task is created and the "Performing Organization" field remains blank. This setting is disabled by default (Fig. 2.286).
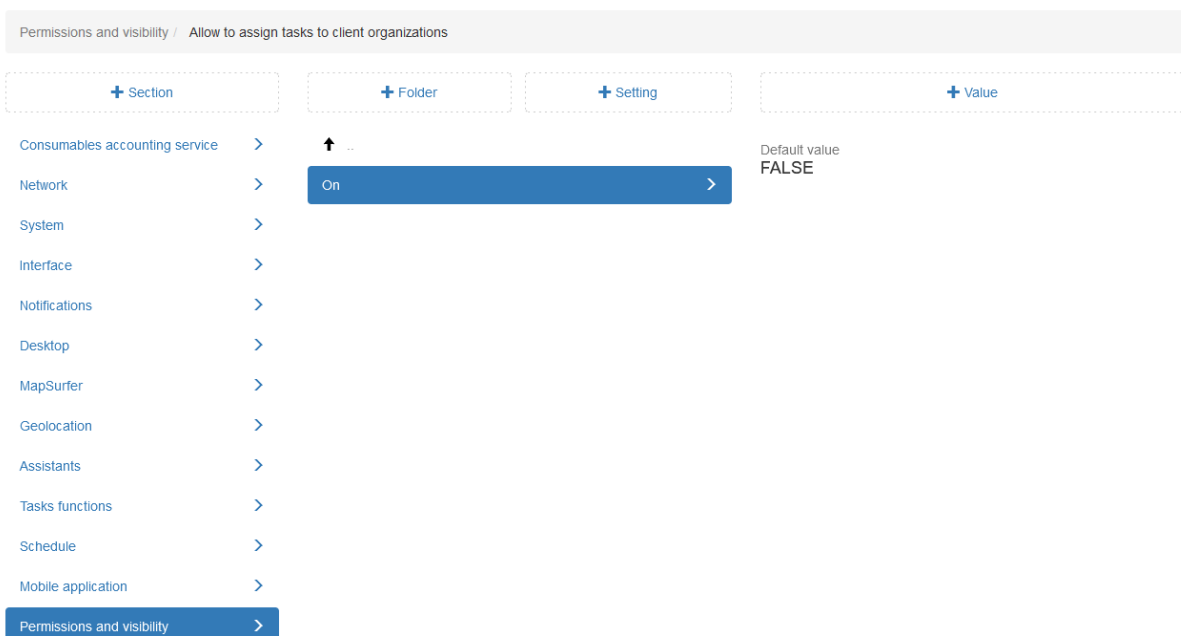


Fig. 2.286: Prohibit assigning tasks to client organizations

### Block other sessions of the same user during authorization

This setting allows you to block a user's session in the application when re-authorizing with the same account in the same application on another device. For example, if a user was authorized in the ActiveMap Desktop, access to another user under this account in the desktop application is blocked. By default, this setting is disabled (Fig. 2.287).
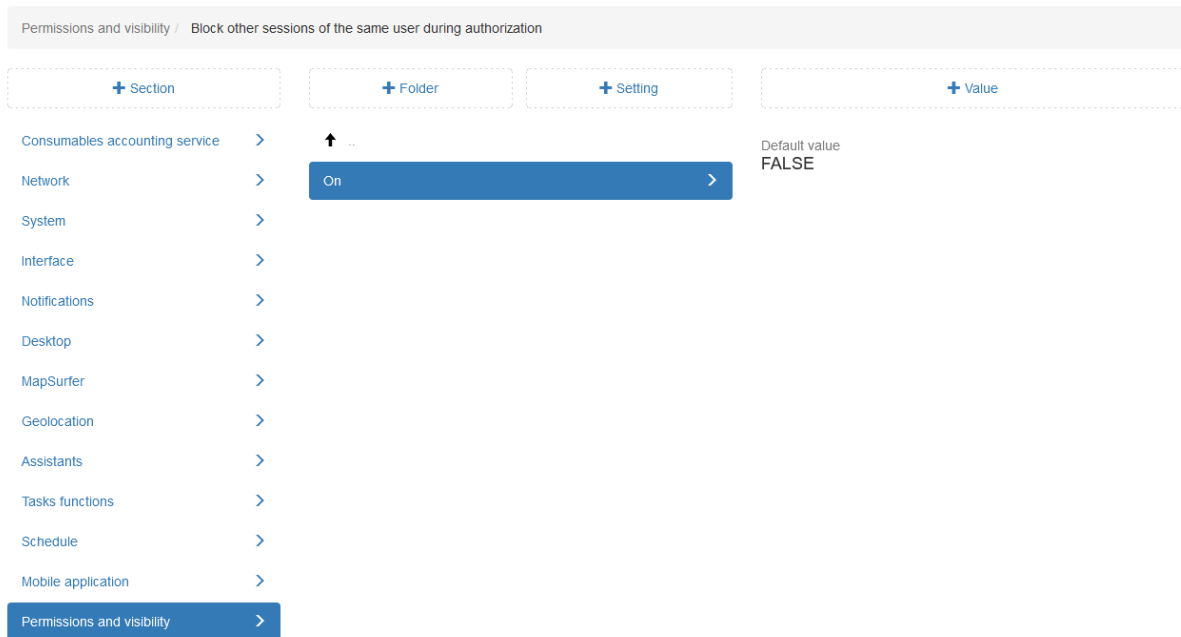


Fig. 2.287: Blocking other sessions of the same user during authorization

### 2.3.3.8 "System" block

This block contains custom settings that you can apply to the system.

#### 2.3.3.8.1 "Global" tab

In this tab you can change the title of the system, add your logo and website icon, and enable automatic assignment of an organization upon creation.

When you activate the "Assign an organization to yourself when creating" setting, the "Assigned organization" field is automatically filled with the name of the created organization after saving new organizations. In the organization card, you can specify another assigned organization by selecting it from the drop-down list and then save the changes.

---

**Important:** The assigned organization should belong to the creator's organization cluster.

---

In this tab you can also change the visible area (the part of the map that is loaded when the web application is launched) by selecting the desired boundaries and clicking "Apply" (Fig. 2.288).
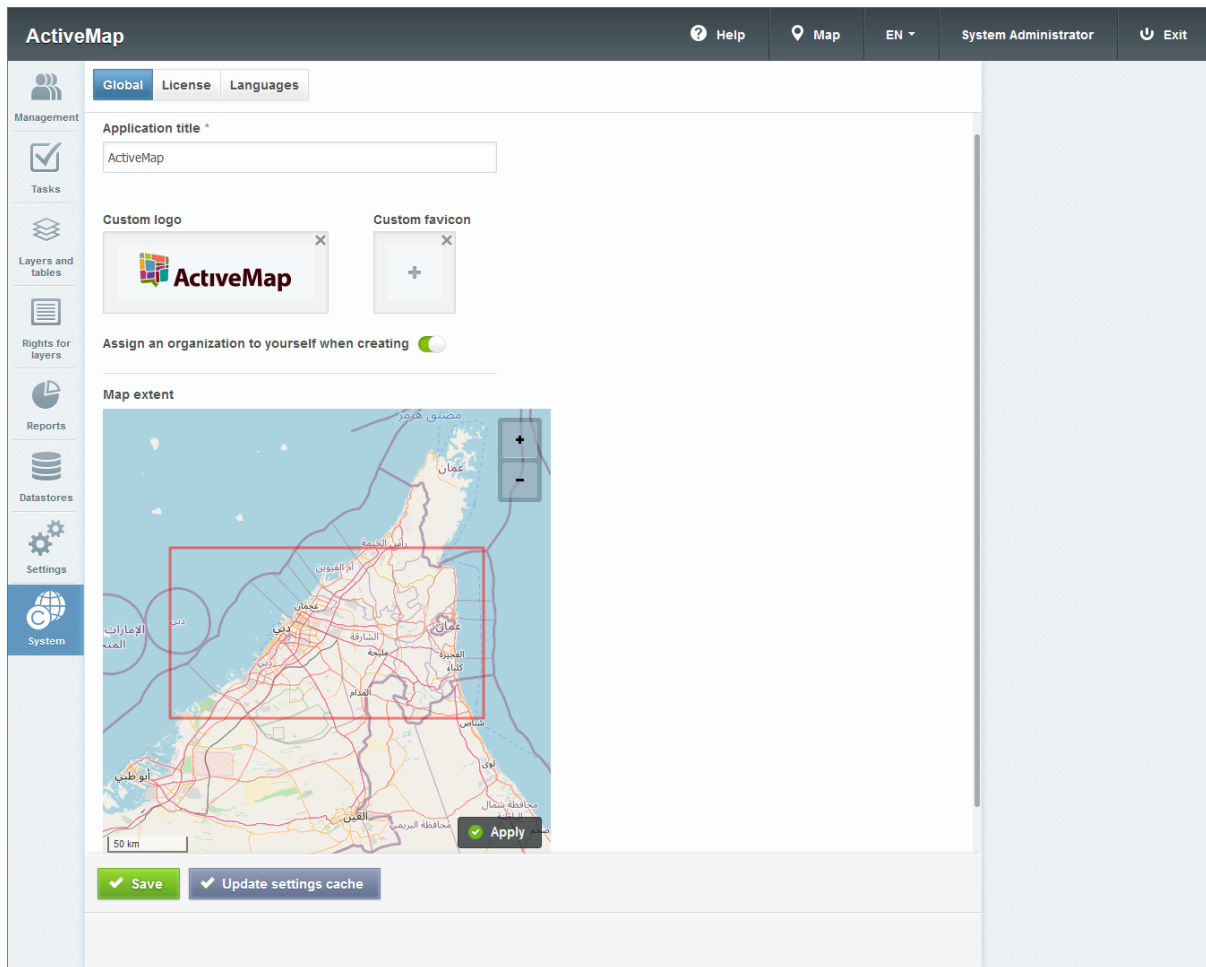


Fig. 2.288: Global system settings

Click [✔ Save] to save your changes. Clicking [✔ Update settings cache] starts the process of updating the internal cache of the web application to download the settings from Cerebellum.

### 2.3.3.8.2 "License" tab

Once the first user with the "System Administrator" role gains access to the system, he/she should perform the license activation procedure. Information about the license activation requirement is displayed in a pop-up window on the administration panel (Fig. 2.289).

Fig. 2.289: "License is not active" pop-up window

If the license is not activated, you cannot create users, and an appropriate message is displayed if you try to save data.

To activate the license, go to the "System" block and select the "License" tab or follow the link in the "License is not active" pop-up window. Enter the license key you received from your service provider and click  (Fig. 2.290).
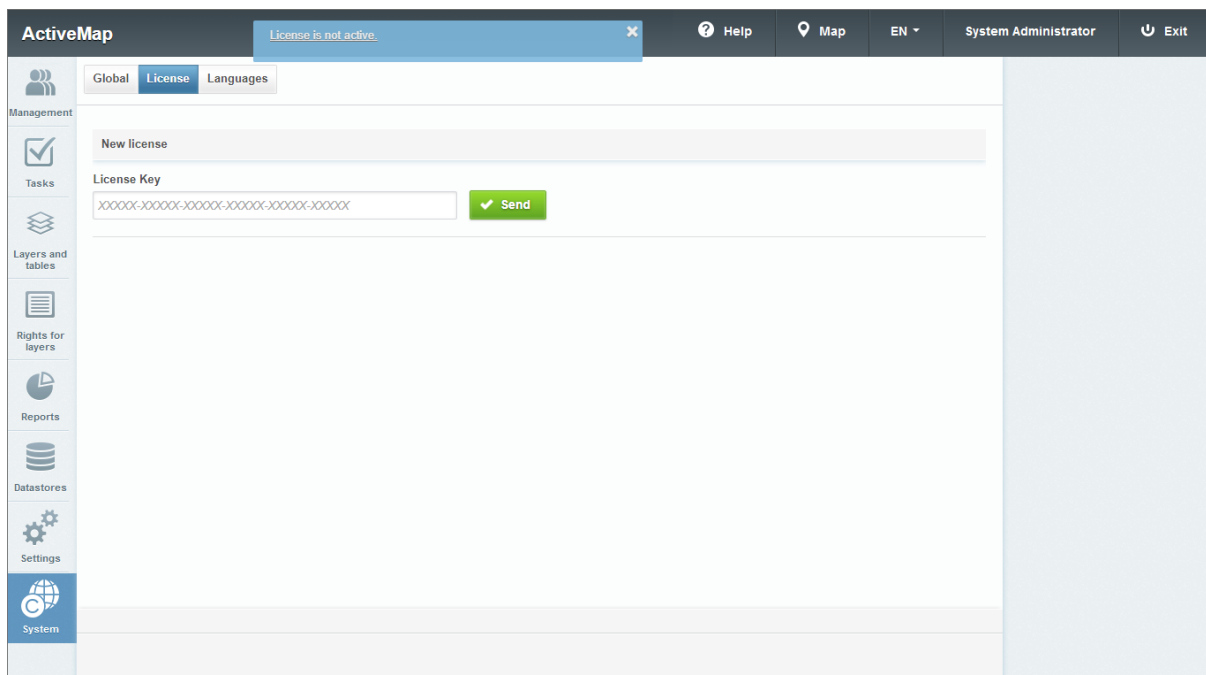


Fig. 2.290: License key entry field

The window displays information about the current license. If the license key is correct, when you click , the system asks you to enter the number of users required for activation (Fig. 2.291).
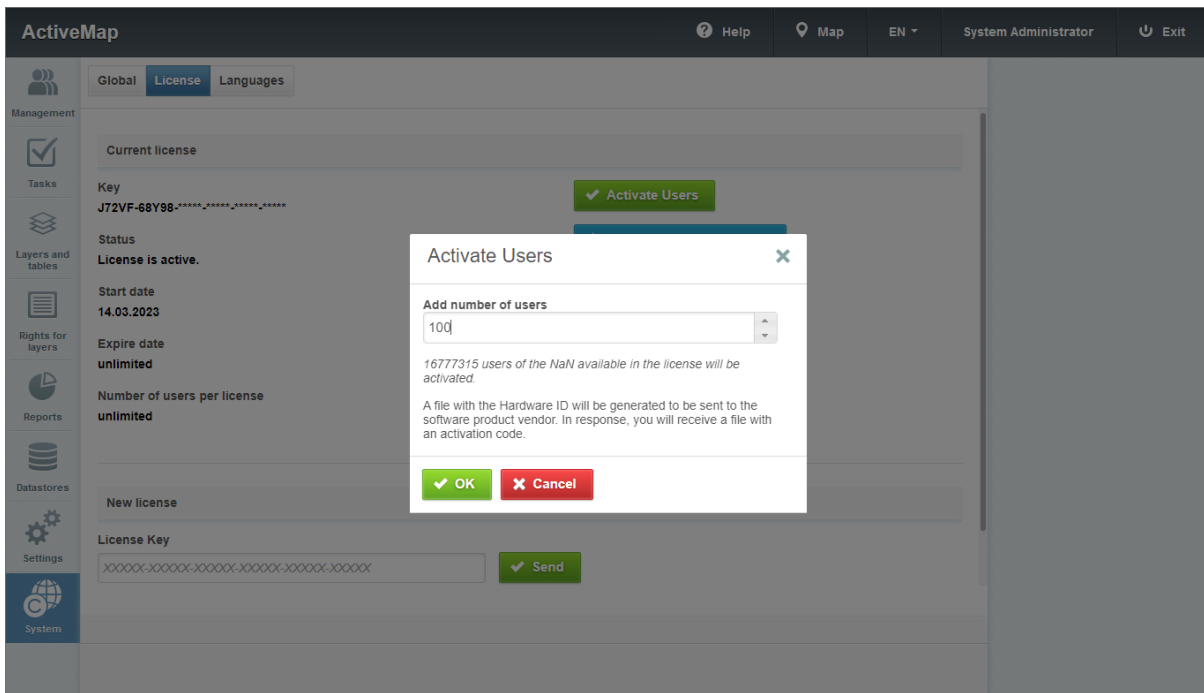
Fig. 2.291: "Activate users" window

Save the "Hardware Code" file, which contains encrypted information about the server characteristics and the license key (Fig. 2.292).
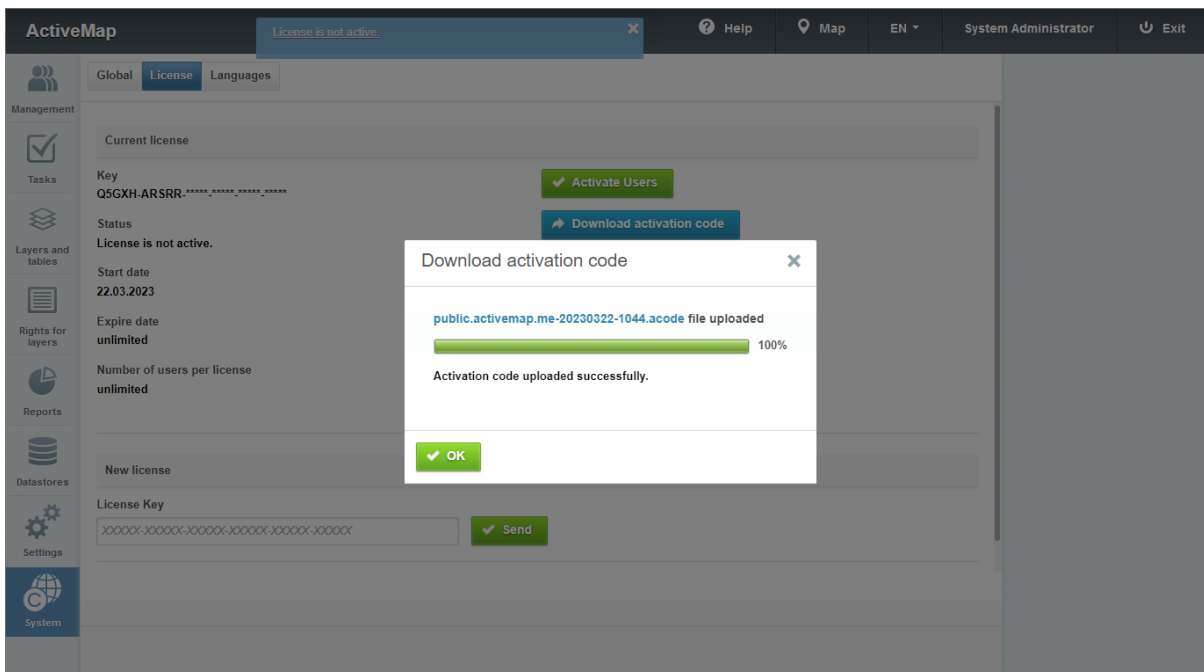


Fig. 2.292: Saving the hardware code

Transfer this file to the service provider for activation by sending a corresponding request to the e-mail. The email should include organization name, number of user licenses required, server domain name where the license key is activated. Service provider verifies the received informa-

tion and checks that the requested number of users does not exceed the value specified in the license. If the data is correct, the service provider sends a file with an activation code, valid for the server specified in the email. Upload the received file to the "License" tab by clicking

. After successful license activation the system displays the following data: status, validity period, maximum allowed number of users, and number of activated users in the system (Fig. 2.293).
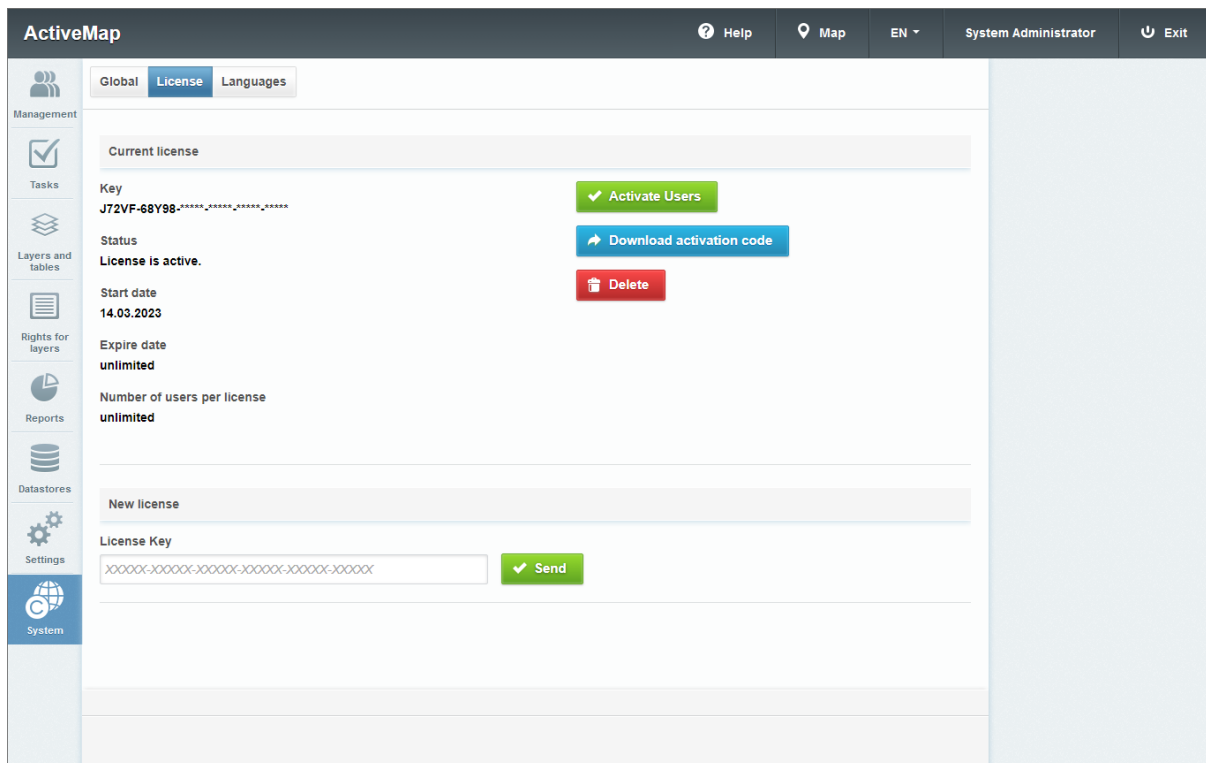


Fig. 2.293: Information about the current license

If necessary, you can request additional licenses for users by following the same steps. You can repeat this process until the total number of users exceeds the allowed number of licenses for this server. Activating this license on another server is not possible, since the activation server stores information that this license key has already been used.

To switch to another license (perpetual or unlimited in number of users), purchase a new license key and enter it into the system. Then load the activation code using both the old and the new hardware code. The system checks the activation code against both hardware codes. If the activation code matches the new hardware code, the old files are deleted and the new activation code is applied.

To terminate the license on a given server, click . A deletion confirmation message appears (Fig. 2.294).
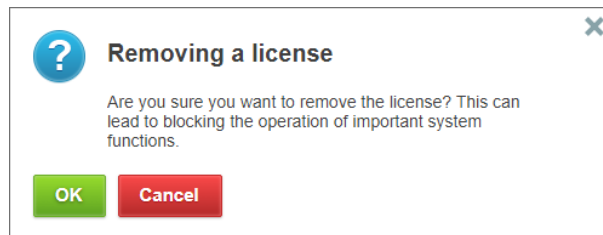
Fig. 2.294: Deletion confirmation message

### 2.3.3.8.3 "Languages" tab

This tab contains a list of available interface languages (Fig. 2.295). You can activate the language by turning on the toggle switch in the line. The activated language is displayed in the drop-down list on the user panel. You can also change the order in which languages appear in the list. To do this, drag the line above or below and click **✔ Save order**.



Fig. 2.295: List of available interface languages

## 2.4 Exit the program

To exit the program, click "Exit" in the upper right corner of the window and close the browser tab.

# FREQUENTLY ASKED QUESTIONS

## 3.1 Starting the Program

If you have problems with starting the Program, try to open the program page in another browser or contact technical support via the hotline phone number indicated on the https://activemap.me/ website, or write an email to support@activemap.me.

## 3.2 Log in to the Program

If you are experiencing authorization problems, please contact technical support by calling the hotline number listed on the https://activemap.me/ website or by sending an email to support@activemap.me.

# GLOSSARY

**Account** is a set of data about a user stored in the system, necessary for the authentication and providing access to personal data and settings.

**Activation code** is a file containing an encrypted hardware code, information about the number of users, and the license period.

**Applied software suite** is a set of interconnected programs designed to solve problems of a certain class of a particular subject area and interact with the user.

**Attribute data** are values describing features of the objects. Attribute data types are: integer, real, text, date, date and time, geometry.

**Band** is an object that is placed directly on the report page. It is a container for the other objects, such as "Text", "Picture", etc.

**Basemap** is the dominant or underlying layer in a given map that provides geographical context to the map and other dataset layers above it. Users visualize tasks, service objects, and thematic layers above the basemap. They use it for navigation through a map and for getting general information about the area of interest.

**Centroid** is the center of a geographical object on a map. For most objects, the centroid coincides with the center of the rectangle described around the object.

**Client organization** is an association of users who make their requests via the mobile application, monitor their status, who are capable of evaluating the work performed. User rights for operating the System are restricted.

**Cluster** is an association of several organizations for the purpose of enabling the in-process control of the performance of departments.

**Cluster Administrator** is a user role in the System, responsible for cluster administration, namely: managing organizations and users of the cluster, granting access rights to layers and reports within the cluster, and managing cluster tasks.

**Cluster Inspector** is a user role in the System, responsible for managing tasks within the cluster.

**Clusterization** is the representation of raster layer objects located nearby by a single label on a map.

**Contract** is an entity for accounting and planning the task to be performed by organizations under contractual obligations.

**Custom fields** are attribute fields, which can be customized in the system versus features of a project underway, and be referenced to the certain work items.

**Data export** is a data loading from the Program database to an external file.

**Data table** is a set of the related data stored in a structured format in a database.

**DBF data format** is a data storage format used as one of the standard ways of storing and transmitting information by database management systems, spreadsheets, etc.

**Drag and Drop** is a way to manipulate interface elements in the user interfaces using a mouse or a touch screen. The method is implemented by "grabbing" (pressing and holding the left mouse button) the object displayed on the screen, which is available for such operation, and then moving it to another place (to change its location) or "dropping" it to another element (to call the corresponding action in the program).

**Executor** is a user role for creating new tasks and performing the assigned tasks in the System.

**GDAL** (Geospatial Data Abstraction Library) is a translator library for raster and vector geospatial data formats. As a library, it presents a single raster abstract data model and a single vector abstract data model to the calling application for all supported formats.

**Geographic coordinates** are the mathematical values that designate a position on the earth relative to a given reference system.

**GeoJSON data format** (Geographic JavaScript Object Notation) is a format for representing various geographic data structures. A GeoJSON object can be represented by a geometry, a feature, or a feature collection. GeoJSON supports the following geometry types: Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon and GeometryCollection. A feature in GeoJSON consists of geometry and additional properties. Feature collection consists of a set of features.

**Geographic Information System (GIS)** is an information system designed to collect, store, analyze, and display spatial data and related information about presented GIS objects.

**GPS** is a satellite navigation system that measures distance, time and determines the location in the WGS 84 world coordinate system. It can accurately determine the three-dimensional coordinates of an object equipped with a GPS receiver: latitude, longitude, height above sea level, as well as its speed, direction of movement, and current time.

**File label (sticker)** is a textual mark in a picture.

**Hardware code** is a file that contains encrypted information about the server characteristics and the license key.

**Hatching** is a set of drawings and colors used to fill polygonal objects.

**Image sticker (file label)** is a text mark on the photo.

**Import object coordinates** is a data loading from external files into the Program database.

**Information display panel** is a panel designed to display specific information related to user actions, as well as messages that correct user actions (warning messages, tips).

**Installer** is a program that installs files on the end user's computer.

**Interval** is a data table that is used to configure the display styles of layer objects on the map depending on their specific numerical characteristics. The Program uses intervals of (a, b) type.

**Invitation (an invite link)** is a link containing information on the server address, login, and password of a user to simplify the process of authorization in the mobile application.

**Layer** is a visual representation of geographical data in the environment of any digital map.

**Layer group** is a set of layers grouped according to thematic or other specified criteria.

**Layer object visibility on the map** is a displaying the layer object on the map as a certain symbol, line, or polygon.

**Layer visibility on the map** is a displaying of all layer objects on the map as a group of symbols, lines, or polygons.

**LDAP (Lightweight Directory Access Protocol)** is an open, vendor-neutral, industry standard application protocol for accessing and maintaining distributed directory information services over an Internet Protocol (IP) network.

**Legend** is a set of symbols and explanations on a map.

**License** is a file containing information on the acceptable quantity of users and validity period, allowing to link the server software of the System to the equipment.

**License key** is a character string provided to the customer by the software vendor after purchasing the license, used to activate the product and obtain a digital license for a fixed server. Contains the maximum number of users and the license period in an encrypted form.

**Linear object** is an object on a digital map that represents a place or item that has length but no area at a given scale.

**Managing map layers** is the set of actions for managing layer visibility, creating and editing the geometry of layer objects on the map.

**Map scale** is the ratio of a distance on a map to the corresponding distance on the ground. A scale of 1:100,000 means that one unit on the map corresponds to 100,000 of the same units of measurement on the ground.

**Mapping** is a correspondence between a layer attribute and a task field.

**MapInfo Interchange Format (MIF)** is a MapInfo text data format that includes geographic data (objects) and a description of the data table containing attribute information related to objects.

**Multi-object** is a combination of several objects. Multi-objects can be of point, line, and polygon geometric types.

**Multiservice** is the ability to represent any layer as a layer with service objects.

**Node** is the point representing the beginning or ending of an edge of a linear or polygonal object, topologically linked to all the edges that meet there.

**Object attributes (attribute data)** are values describing the object properties. Attribute data types are: integer, real, text, date and time, geometry.

**Object geometry** is the measurements and properties of points, lines and surfaces. In GIS, geometry represents spatial components of geographic objects.

**One-to-many relationship** is a relation between two sets of data where one record in a parent table can be associated with one or more records in another table (child data table).

**Operational tasks** are the tasks created to solve current issues.

**Organization Administrator** is a user role in the System, responsible for administering the organization, namely: creating users, granting access rights to layers and reports within the organization, and managing tasks of the organization.

**Organization Inspector** is a user role in the System, responsible for managing tasks within the organization.

**Point object** is a cartographic object that does not have length or area in the accepted scale.

**Polygonal (area) object** is a cartographic object that bounds the area at a given scale.

**Program user (User)** is a person (employee) or organization that uses the current Program to perform a specific function.

**Raster layer** represents data in the form of geographically-referenced images as well as fragments of raster images displayed in the same projection and prepared for each level of map detail.

**Reference table (dictionary)** is a table with systematically organized data intended to help users to handle attribute information on objects.

**Service objects** are the layers containing the objects of interest of the user organization due to their relation to business activity of the involved organization.

**Schedule** is a tool that allow users to automatically create and assign template tasks at a certain time with a specified periodicity.

**SHP data format** is a vector format of geographic files. It allows users to store the following types of geometric objects: points (polypoints), lines (polylines), polygons, and other objects. A file can contain only one object type. Each entry in the SHP file can have multiple attributes to describe its geometry.

**Scheduled tasks** are the tasks created at a specified date and time according to a template.

**Spatial database** is a database optimized to store and access spatial data or data that defines a geometric space.

**SQLite** is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine.

**SQLite Data Format** is the SQLite relational database file format.

**Sub-object** is an object included in the multi-object.

**Symbol** is a graphical representation of a geographic object or a class of spatial objects, which helps to identify and distinguish them from other spatial objects on the map.

**System Administrator** is a user role in the System with the maximum rights, responsible for its configuration, including managing clusters, organizations, users of all roles, contracts, directories, as well as for distributing access rights to layers and reports.

**System Inspector** is a user role in the System, responsible for managing tasks across all clusters.

**System reference table** is a reference table generated automatically based on data entered into the system. System reference tables include tables of system users, priorities and types of work.

**TAB data format** is the format of MapInfo vector spatial data files (MapInfo files).

**Task** is a key element of the system, assigned to a user, which can contain instructions for execution, information about the contract, service object, type of work, creation date, deadline, priority, and execution step. Additional files (documents, photos, videos) can be attached to the task.

**Task priority** is a characteristic of the urgency of the task.

**Task status** is a characteristic of the completion degree of work on the task, determined by the dispatcher or administrator when accepting the task.

**Task step** is a stage in the sequence of actions for completing a task changed by the task executor.

**Thematic layer** is a spatial data bank layer which objects are interrelated by the same topic.

**Timelapse-video** is a video file comprising a series of pictures taken via a video camera during a long time period.

**Tile Map Service (TMS)** is a specification for storing and retrieving cartographic data that provides access to the map tiles rendered at a specific scale level. These resources are accessed via the "REST" interface.

**Toolbar** is a graphical user interface with buttons for performing Program commands.

**Tiled Web Map Service (TWMS)** is a specification for storing and retrieving map data that provides pre-built georeferenced map images. TWMS relies on technologies for building and transmitting large images to the Internet using tiles – small, standard-sized image fragments. A TWMS service may also include one or more styles, dimensions, or tiling schemes to define how the TWMS layer is displayed. Accessing data via the TWMS protocol requires preprocessing of the source cartographic data by creating tiles for the full range of scales, over the entire area. This technology allows locally caching an image by building a tile grid.

**User profile** is a characteristic of an individual system user, represented by a set of attributes, such as full name, email, phone number, etc.

**User rights management** is a set of actions for registering and managing user rights in the Program.

**User tags** is an entity allowing to group users against a specified attribute (e.g., the phone model).

**User type** is a user characteristic (a human being or a vehicle) to determine the user mapping settings versus the type selected.

**Vector image** is a representation of graphical objects and images based on the use of geometric primitives such as points, lines, and polygons.

**Webhook** is an automated launching of http requests in response to operations on entities (comments and tasks).

**Web Feature Service (WFS)** is a web service for querying spatial data that includes a standardized API. Unlike the Web Map Service (WMS), which returns a map image (rendered data), the WFS service returns actual objects with geometry and attributes that can be used in any type of geospatial analysis. WFS services also support filters that allow users to perform spatial and attribute queries on the data.

**Web Map Service (WMS)** is a standard protocol for serving geographically referenced images over the Internet, generated by a cartographic server based on data from the GIS database. The WMS service may also include a Styled Layer Descriptor (SLD) to define how the WMS layer should be displayed. The WMS service layer consists of three elements arranged hierarchically in the table of contents. At the top is the name of the WMS service, which contains all the layers of the WMS map. The next level down contains the WMS composite layers whose only function is to organize the WMS sublayers into appropriate groups. There is at least one WMS composite layer, but there can be any number of composite WMS layers (and even nested groups within groups). WMS composite layers do not contain map layers. This is the third group, WMS sublayers that actually contain map layers.

## ATTACHMENTS

# 5.1 Attachment 1. Python plugins

## 5.1.1 General information

Plugins are Python scripts that run before and after changing (creating, editing, or deleting) of system server core entities.

The system supports the following types:

**Validation** is a plugin that runs before modifying a system server core entity, which allows you to check the correctness of the entered data.

**Reaction** is a plugin that runs after modifying a system server core entity, which allows you to trigger another action in response to data changes.

Version 0.38 of the system's server core supports plugins invoked in response to actions with tasks and comments (creating, editing, or deleting). However, in this version, it is not possible to edit or delete comments from plugins.

Version 0.41 of the system's server core adds support for plugins invoked in response to user actions (creating, editing, or deleting).

The plugin receives entity states before and after changes, as well as system utilities. Plugin execution can trigger a response action in the server core of the system. To do this, the plug-in code must return the result of its execution to the system's server core.

All plugins are executed on behalf of the "Plugin system" system user with the System Administrator role.

When creating and deleting tasks, users, or task comments, the following variables are available in Python code:

- model (also known as after) – the task, user, or comment being created;

- subject – the creator of the task, user, or comment.

When editing tasks and users the following are available:

- model (also known as after) – the task or user after modification;

- before – the task or user before modification;

- subject – the user making the request.

For security reasons, the variables store wrappers over the models (Java objects) which are read-only.

## 5.1.2 Wrapper methods

All wrappers, except those explicitly listed below, have only one method, `getId()`.

**model/before/Mockups.Task**

| Return type | Method name (parameter list) | Description |
|---|---|---|
| | getId() | task id |
| Mockups.User | getUser() | task creator |
| | getDate() | task date |
| | getOrganization() | creator organization |
| | getWorkgroup() | task project |
| | getType() | type of work |
| | getPriority() | priority |
| | getStage() | task step |
| | getStatus() | task status |
| | getText() | description |
| | getTitle() | title |
| | getAs signedOrganization() | assigned organization |
| Mockups.User | getAssignedUser() | assigned user |
| Mockups.Contract | getContract() | contract |
| | getServiceObjectId() | service object id |
| | getSe rviceObjectLayerId() | service object layer id |
| | get ServiceObjectTitle() | service object title |
| | getScheduleId() | shedule id |
| | getTemplateId() | template id |
| | getParentId() | parent task id |
| | getLocation() | point coordinates [lon, lat] |
| Attachment | getMainPhoto() | main photo |
| [ Mockups.Attachment ] | getFiles() | files |
| [ Mockups.Attachment ] | getPhotos() | photos |
| [ Mockups.Attachment ] | getVideos() | video files |
| [ Mockups.Attachment ] | getSounds() | audio files |
| | get(String translit) | value of a custom field whose 'translit' is requested |
| | getDeadline() | deadline |
| | getExpiredDate() | expire date |
| | getSampleMatching() | minimum photo matching percentage with the sample |
| | getAddedPhotoCount() | number of added photos |

**subject/Mockups.User**

| Return type | Method name (parameter list) | Description |
|---|---|---|
| | getId() | user id |
| Mockups.Role | getRole() | user role |
| Mockups.Cluster | getCluster() | user cluster |
| Mockups.Organization | getOrganization() | main organization |
| Mockups.Organization | getOrganizations() | list of available organizations |
| | getFio() | full name |
| | getLogin() | user login |
| | getPassword() | user password |
| Mockups.UserInfo | getInfo() | detailed user information |
| | getEmail() | user email |
| | getAddress() | user address |
| | getPhone() | user phone |
| | getPassport() | user passport |
| | getSystem() | system user or not |
| Mockups.UserType | getType() | user type |
| Mockups.Tag | getTags() | user tags |
| | getBlocked() | user blocked or not |
| | getTracking() | tracking enabled or not |
| | getGlonassId() | GLONASS-id for tracking |
| | getAvatarFileName() | avatar file name after upload to the server |
| | g etAvatarUpdateDate() | avatar update date |
| | get LdapAuthentication() | user authentication through LDAP |
| | get LastAuthentication() | user's last authorization date |
| | getGisEditorAccess() | user integration with GIS-editor |

**model/Mockups.Comment**

---

**5.1. Attachment 1. Python plugins**

| Return type | Method name (parameter list) | Description |
|---|---|---|
| | getId() | comment id |
| | getUuid() | comment UUID4 |
| | getReferenceId() | id of the comment to which this comment is a reply |
| | getLevel() | comment level |
| Mockups.User | getUser() | comment creator |
| | getUpdateText() | update text (for task update comments) |
| | getComment() | comment text |
| | getType() | comment type (regular or system) |
| Mockups.Diff | getDiff() | information about the changes made to the task (for system comments) |
| Mockups.Chat | getChat() | channel information |
| | getChatMessageNumber() | message number in the chat channe |
| | getSystemMessageCode() | system message code |
| Mockups.Task | getTask() | information about the task to which the comment relates |

**Mockups.Attachment**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getSticker() | sticker |
| getAttachmentLocation() | attachment file location coordinates in [lon, lat] format |
| getOriginLocation() | coordinates of the file creation location in [lon, lat] format |
| getName() | filename on the server |
| getType() | file type; possible values: photos, sounds, video, dif_files |
| getDescription() | file description |
| getNum() | file serial number |
| getIsRemote() | remote file or not |
| getFile() | physical file |
| getParentPhotoId() | parent photo id (available only for photos) |
| getSampleMatching() | photo match percentage with the sample |

**Mockups.File**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getAuthor() | author |
| getDate() | creation date |

**Mockups.Sticker**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getName() | name |

**Mockups.Type**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getName() | name |

**Mockups.Organization**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getName() | name |

**Mockups.UserInfo**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getEmail() | email |
| getAddress() | address |
| getPhone() | phone |
| getPassport() | passport |

**Mockups.UserType**

| Method name (parameter list) | Description |
|---|---|
| getId() | user type id |
| getTitle() | user type name |
| getIconFileName() | icon file name after uploading to the server |
| getIconUpdateDate() | date of the last icon change |
| getIsDefault() | is this user type the default |
| getUseInitials() | whether to use user's initials in the marker on the map (instead of an icon) |
| getUseAvatar() | whether to use user's avatar in the marker on the map (instead of icons and initials) |

**Mockups.Tag**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getTitle() | name |

**Mockups.Contract**

| Method name (parameter list) | Description |
|---|---|
| getId() | id |
| getTitle() | title |
| getStartDate() | contract start date |
| getFinishDate() | contract end date |
| getGrantTaskCreation() | contractor's rights to create contract tasks |
| getCluster() | cluster |
| getCustomer() | customer organization |
| getAssignedOrganization() | executing organization |

**Mockups.Diff**

| Method name (parameter list) | Description |
|---|---|
| getId() | id of the entry with information on changes made to the task (for system comments) |

**Mockups.Chat**

| Method name (parameter list) | Description |
|---|---|
| getId() | id of the entry with information on the message channel |

### 5.1.3 Validations

Validator plugins allow you to check the correctness of an entity when performing actions on it. Validators are executed before the entity is saved in the database.

The plugin should return one of three possible values:

- valid() – everything is correct, the entity can be saved;

- invalid() – error, the entity cannot be saved;

- invalid(String error) – error, the entity cannot be saved, show error message.

```
if(len(model.getPhotos()) == 0):
    return invalid()

for photo in model.getPhotos():
    if(photo.getAttachmentLocation() is None or photo.getOriginLocation() is
 →None):
        return invalid()

    dist = utils.dist(photo.getAttachmentLocation(), photo.
 →getOriginLocation())

    if(dist > 1000):
        return invalid('too much distance between points')

return valid()
```

### 5.1.4 Reactions

A reaction is the ability to launch a request for one or more operations on the same task or user in response to a successful operation on a task or user. For example, in response to attaching new files to a task, run a request to change a custom field storing the number of files. The server core of the system processes requests asynchronously, therefore, although the reaction launches the request, it is processed in a thread separate from the reaction and almost certainly ends after its execution.

A reaction can launch a single response request:

```
return upd().setTitle(after.getTitle() + '!')
```

Or launch several response requests:

```
return [
    upd().setText('JEP-modified'),
    upd().setTitle('Modified by JEP')
]
```

Also, the reaction may return a message to the system's server core that no action is required:

```
return noop()
```

In reactions, you can specify the user on whose behalf the request to create or modify a task is made.

```
return ins().by(101).set...
return fullclone().by(101).set...
return briefcopy().by(101).set...
return upd().by(101).set...
```

In version 0.38 of the system's server core, the `ins()`, `upd()`, `briefcopy()`, `fullclone()` methods are available only when writing a reaction to a task operation. Starting with version 0.41 of the system's server core, the `ins()` and `upd()` methods become available when writing a reaction to a user operation.

### 5.1.5 Creating new tasks and users

```
# create a new task/user
return ins()

# create a full copy of the current task (including files)
return fullclone()

# create a partial copy of the current task (excluding files):
return briefcopy()
```

Calls to the `ins()`, `briefcopy()`, `fullclone()` methods for a task create a ready-made request, which you can modify. For example, change the fields for the created task: title, text, date, add files, etc. For the `ins()` method, these modifications are mandatory, because `ins()` for a task creates an empty task without the following mandatory fields: type of work, priority, and organization.

Calling the `ins()` method for a user creates an empty user without login, password, full name, and organization (these are mandatory fields that must be filled in).

```python
# fill in the mandatory fields when creating a task
return ins().setTitle('Hello').setType(1).setPriority(1).setOrganization(4)

# create a full copy of the current task and change the title at the same time
 return fullclone().setTitle(after.getTitle() + ' - Modified')

# create a new user
return ins().setLogin('test_123').setPassword('123').setFio('test_123').
↪setOrganization(3).setRole(7).setAddress('Moscow').setEmail('test@ya.ru')
```

By default, the `briefcopy()` and `fullclone()` methods use `after` as the data source for copying (`before` in the case of reaction to task deletion). However, you can explicitly specify which of the two task states to use:

```python
# If this is a reaction to a change:
# Create a copy of the task state BEFORE the change
# for each transaction
return fullclone(before)
```

## 5.1.6 Methods for setting task fields

| Method name (parameter list) | Description |
|---|---|
| set(String translit, Object value) | value of a custom field by transliteration |
| add(List<Attachment> attachments) | add a list of files |
| add(Attachment attachment) | add a file |
| setMainPhoto(Attachment photo) | set new main task photo |
| setLocation(Double lon, Double lat) | set coordinates |
| setLocation(List<Double> coordinates) | set coordinates |
| setTitle(String title) | set title |
| setText(String text) | set description |
| setOrganization(Long id) | set creator's organization id (only when creating a task) |
| setWorkgroup(Long id) | set project id (only when creating a task) |
| setDate(Date date) | set task date in java.util.Date format (only when creating a task) |
| setDate(Double date) | set task date in python timestamp format (only when creating a task) |
| setType(Long id) | set work id |
| setContract(Long id) | set contract id |
| setPriority(Long id) | set priority id |
| setAssignedOrganization(Long id) | set assigned organization id |
| setAssignedUser(Long id) | set assigned user id |
| setStage(Long stage) | set task step |
| setStatus(Long id) | set status id |
| setParent(Long id) | set parent task id |
| setServiceObject(Long id) | set service object id |
| setServiceObjectLayer(Long id) | set service object layer id |
| setArchive(Boolean archive) | set task as archived |
| setDeadline(Date date) | set deadline in java.util.Date format |
| setDeadline(Double date) | set deadline in python timestamp format |
| setExpiredDate(Date date) | moment when the task was overdue in the java.util.Date format |
| setExpiredDate(Double date) | moment when the task was overdue in the java.util.Date format |

### 5.1.7 Methods for setting user fields

| Method name (parameter list) | Description |
| --- | --- |
| setFio(String text) | set user name |
| setLogin(String login) | set user login |
| setPassword(String password) | set user password |
| setEmail(String email) | set user email |
| setAddress(String address) | set user address |
| setPhone(String phone) | set user phone |
| setPassport(String passport) | set user passport |
| setRole(Long id) | set user role id |
| setOrganization(Long id) | set id of the main organization |
| addOrganizations(List<Long> ids) | add user access to organizations, list of organization ids |
| removeOrganizations(List<Long> ids) | remove user access to organizations, list of organization ids |
| setType(Long id) | user type id |
| addTags(List<Long> ids) | add tags to the user, list of id tags |
| removeTags(List<Long> ids) | remove user tags, list of id tags |
| setBlocked(Boolean blocked) | enable or disable blocking of user |
| setTracking(Boolean tracking) | enable or disable tracking of user |
| setAvatarFileName(String avatarFileName) | add user's avatar (avatar file name after uploading to server) |
| setLdapAuthentication(Boolean ldapAuthentication) | enable or disable user authentication via LDAP |
| setGisEditorAccess(Boolean gisEditorAccess) | enable or disable integration with GIS editor (available only to the System Administrator) |

### 5.1.8 Working with dates

The `setDate()`, `setDeadline()`, and `setExpiredDate()` methods have two versions that accept `Date` and `Double` respectively. You can call the version with `Date` using special utilities for generating `Date` objects:

```
if model.getDeadline() is not None:
    return noop();


return upd().setDeadline(dates.get(2021, 12, 31, 12, 30, 0))
# or
return upd().setDeadline(dates.get("2022-02-12 13:05:30"))
# or
return upd().setDeadline(dates.get(1637668042))
# or
return upd().setDeadline(dates.now())
```

The version with `Double` is intended for working with the `datetime` package:

```
import datetime

if model.getDeadline() is not None:
```

```
    return noop();

tomorrow = datetime.datetime.now() + datetime.timedelta(days=1)

return upd().setDeadline(tomorrow.timestamp())
```

### 5.1.9 Adding files to a task

In reactions, you can initiate the attachment of files to a task.

To do this, you need to:

- Pass the name of the file on the disk relative to the server core's file directory to one of the functions: `photo`, `sound`, `video`, or `anyfile`. The file should already be in this directory.

- Call (if necessary) methods to specify additional file fields (`setSticker`, `setDescription`).

- Add the result of the function call to the task using the `add` methods (you can transfer either a single file or several ones).

```
file1 = photo('photo-2/2021/06/22/16-06/d392a0c4-eb12-1004-8555-4a467d53f32e.
↪jpg')
file2 = photo('nas-2/2021/06/22/16-06/d3bf55b0-eb12-1004-8555-4a467d53f32e.jpg
↪').setSticker(1163)
file3 = photo('photo-2/2021/06/22/16-06/d392a0c4-eb12-1004-8555-4a467d53f32e.
↪jpg').setDescription('test')
file4 = photo('photo-2/2021/06/22/16-07/d3d27ba4-eb12-1004-8555-4a467d53f32e.
↪jpg').setDescription('test').setSticker(1163)
return upd().add([file1, file2]).add(file3).add(file4)
```

You can attach an existing photo from `after` or `before`:

```
return [
    # "short" copy with additional
    # attachment of the first photo from the original task
    briefcopy().add(before.getPhotos()[0]),
    # similar to to fullclone()
    briefcopy().add(after.getPhotos())
]
```

You can also attach an existing photo, but at the same time set a new (different) sticker or description for it:

```
return ins().add(photo(after.getPhotos[0]).setSticker(4))
```

In reactions, you can attach stickers to existing files:

```
u = upd()
u.getPhoto(0).setSticker(3)
return u
```

---

### 5.1.10 Editing tasks and users

Returning the result of the upd() function from reaction initiates a modification of the current task or user. Just like with ins(), you can call methods that set the required fields of the task or user:

```python
# modifying task
u = upd().setTitle(after.getTitle() + ' - Modified')
u.add(photo('photos/somfilehere.jpg))
return u
```

```python
# modifying user
u = upd().setFio(after.getFio() + ' - Modified')
u.setOrganization(11).addOrganizations([4,5]).removeOrganizations([1,3])
return u
```

### 5.1.11 Modifying task files

You can use the ins() and upd() functions to call methods that add files to a task:

```python
u = upd().add(photo('photos/somfilehere.jpg))
return u
```

You can also modify existing task files. To do this, you need to:

- get a link to a file by calling one of the methods: getPhoto(), getFile(), getSound(), getVideo();

- call one of the required methods: setSticker() or setDescription().

```python
u = upd()
# The getPhoto(int) method fetches the file link
# by the sequential method in the getPhotos() array.
# Numeration starts from 0.
# The serial number is not num! (not a file number)
u.getPhoto(0).setDescription('The first photo')
u.getFile(0).setSticker(3)
return u
```

The result of the getPhoto(i) method call differs from the result of the getPhotos()[i] call. The first call returns a link to a file to which you can make changes. The second returns a wrapper object that only allows you to read some file properties:

```python
u = upd()
# this does not work! you need to use getPhoto(i)
u.getPhotos()[0].setSticker(3)
return u
```

Also in version 0.38 of the system's server core, writing the entire action into one line without an intermediate variable does not work. Reactions expect that the response returns the result of the upd() call, not getPhoto():

```
# this does not work!
return upd().getPhoto(0).setSticker(3)
```

### 5.1.12 Setting the main photo of the task

You can set the main photo both when creating a task and when changing it. The installed main photo can be from the "old" files and from the "new" ones.

```
# making a copy of the task, setting the first photo as main
return fullclone().setMainPhoto(after.getPhotos()[0])
```

```
# making a copy of the task, adding a new photo,
# making it the main one
f = fullclone()
p = photo('photos/somephoto.jpg')
return f.add(p).setMainPhoto(p)
```

```
# making the first photo the main one
return upd().setMainPhoto(after.getPhotos()[0])
```

```
# adding a copy of the first photo,
# making the copy of the main one
u = upd()
p = photo(after.getPhotos()[0])
return u.add(p).setMainPhoto(p)
```

And an example of incorrect use:

```
# photo(...) creates an auxiliary photo object,
# which is used when adding;
# but in this case the photo is not added,
# so it's useless to try to make it the main one
return upd().setMainPhoto(photo(after.getPhotos()[0]))
```

### 5.1.13 Webhooks

Among the reactions, **webhooks** is particularly noteworthy. This is an automated launch of http requests in response to operations on entities. Unlike regular reactions, webhooks can be written **for tasks or users, as well as for comments**. Python code can construct the url and body of the request, or specify that no request needs to be made. To run a query, the reaction must return the result of one of the special functions:

```
return post(url)
return patch(url)   # or
return put(url)     # or
return delete(url)  # or
return get(url)     # or
```

The passed URL can be either absolute or relative. A relative URL is completed to the address of the system's local server core (usually http://localhost:9099). The URL may contain a placeholder/:id, which is replaced with the ID of the current entity. If the request is sent to http://localhost / http://127.0.0.1, then the "Plugin System" system user token with the "System Administrator" role is automatically added to it.

For example, you can write a reaction that adds a comment to any task change:

```
# Sends { "comment": "Hello" } to http://localhost:9099/rest/tasks/:id/
↪comment?token=...
# The initial / can be omitted.
# The prop() method allows you to add properties to the JSON request body;
# more on it below.
return post('/rest/tasks/:id/comment').prop('comment', 'Hello')
```

The reaction can trigger any number of webhooks:

```
return [
    post('/rest/tasks/:id/comment').prop('comment', '#1'),
    post('/rest/tasks/:id/comment').prop('comment', '#2'),
    post('/rest/tasks/:id/comment').prop('comment', '#3')
]
```

Finally, in the reaction, you can freely mix requests to create or change the task and webhooks:

```
return [
    post('/rest/tasks/:id/comment').prop('comment', 'Comment'),
    upd().setTitle(after.getTitle() + '!')
]
```

Webhooks solve the same tasks that can be solved by the `requests` module.

The first difference from the `requests` is the automatic completion of the URL and the addition of a token.

But the second difference is much more important. Requests made via `requests` are executed immediately. They are blocking, i.e. the plugin is stopped until a response to the request (successful or unsuccessful) is received. If a request is executed for a long time (for example, a minute), and there are several such requests, then they can block the entire work of the server core. In contrast, the `return post()` call does not start the request. It only builds and schedules it for sending. After the request is formed, the work of plugins continues as usual. Incoming requests are processed in a separate thread. The server for sending the request is determined for each of them. For all such servers, the server core keeps a record of the number of sent requests for which a response has not been received yet. If this value is less than the threshold value N (controlled by the *Webhooks* (page 145) setting), then the request is sent immediately. Otherwise, the request is sent after receiving a response to any of the requests in processing. For example, if N = 10, then the first 10 requests to the https://some.server server are sent immediately, and the 11th is sent only after at least one of the first 10 has completed.

Finally, the third difference from the `requests` module is that you can create and attach destination servers for webhooks in the system (actually, an http address that that you can name, enable/disable, and delete) (Fig. 5.1).

Fig. 5.1: Webhook destination servers

If a reaction has an attached server and generates a *relative* request (tasks/:id/comment instead of https://some.server/tasks/:id/comment), the request is sent exactly to the attached server.

For example:

```
# Assuming the https://am37.activemap.ru server is attached to the reaction␣
↪script
# Then the script below tries to send { "comment": "Hello" }
# to https://am37.activemap.ru/rest/tasks/:id/comment
# (unsuccessfully, because at least a token is not added).
return post('/rest/tasks/:id/comment').prop('comment', 'Hello')
```

The request is ignored if the server is turned off or deleted.

Destination servers are required to group webhooks and quickly change the destination server. If there are 5-10 webhooks and their API entry point has changed, it is enough to change only the address of the destination server.

Finally, special methods have been added to build requests that allow you to specify the request body in JSON format.

```
req = post('/herewego')
# props sets several properties in the body
req.props({
    'x': 2,
    'y': 3
})
# prop adds one property to the body
req.prop('op', '+')
# props adds properties rather than replacing them
req.props({
```

(continues on next page)

```
    'ans': 5
})

return req
```

Each `prop` argument can be a string, number, boolean type, date, list of values, or dictionary of values. In addition, you can pass `after/before` (for tasks and comments) to `prop`. The data about them is automatically converted to JSON when you submit a request (see its format in `/ rest/docs`):

```
return post('/someurl')
    .prop('before', before)
    .prop('after', after)
```

Finally, there are methods that allow you to set the entire body as a `map`. The difference from `props` is that `props` does not clear the previously passed body:

```
body = {
    'before': before,
    'after': after
}
```

In `body` you can pass not only `map`, but also `before/after`:

```
# subsequent calls to prop/props
# add new properties to the body
return post('/someurl').body(after).prop('operation': 'update')
```

In conclusion, one more example:

```
return post("/users")
    .prop("login", "user")
    .prop("paswd", "password")
    .prop("fio", "Test user")
    .prop("organization_id", 3)
    .prop("workgroup_ids", [1, 3, 4])
    .prop("role_id", 7)
    .prop("type", {"id": 2})
    .prop("tracking", False)
```

### 5.1.14 Utilities

Reference tables are available in the plugin code:

```
refbooks().types().byName(...)
refbooks().stickers().byName(...)
refbooks().users().byName(...)
refbooks().statuses().byName(...)
refbooks().fields().byName(...)
```

The `byName()` method for each refrence table searches for a model with the required "name/title/fio". This depends on which field of this model is more suitable in meaning to `byName()`. Returns the desired `Mockups.*` model.

Utility functions are also available:

```
// Returns the approximate distance
// between two points on the surface of the Earth (in kilometers).
// It is assumed that the Earth is a perfect sphere.
dist(List<Double> point1, List<Double> point2)

// The function gets two collections of photos, and returns those,
// which are in "after", but which are not in "before".
diffPhotos(List<Attachment> before, List<Attachment> after)
```

### 5.1.15 Imports in Python

You can install additional packages if needed. One of the easiest ways to install a package is to use the `pip` utility: `pip install <packagename>`. In the script, the package is connected using the `import` operator.

```python
import requests
import warnings

warnings.filterwarnings("ignore")
url = 'https://integration.dev.geo4.pro/rest/'

def getToken():
  response = requests.post(url + 'auth/by-login?apiVersion=2.0', json = {
→'login': 'depadmin', 'password': '123456789'}, verify = False)
  return None if response.status_code != 200 else response.json()['token']

def postComment(id):
  return requests.post(url + 'tasks/%d/comments?token=%s&apiVersion=2.0' %
→(id, getToken()), json = {'comment': 'jep\'s comment'}, verify = False)

postComment(model.getId())
return noop()
```

### 5.1.16 Potential vulnerability: recursive triggering of the change reaction

It is important to note that the creation or modification of a task in the reaction is the same as all other requests. Validations and reactions are also launched for it. Therefore, it is easy to write a script that triggers an endless series of reactions. For example, if you need to change the deadline (set it a day ahead) with each update of the task, the following script triggers an infinite chain of reactions:

```python
import datetime

now = datetime.datetime.now()
```

```
tomorrow = now + datetime.timedelta(days=1)

return upd().setDeadline(tomorrow.timestamp())
```

## 5.1.17 Plugin script examples

**Upload files to a layer object**

```
if after.getServiceObjectLayerId() is not None and after.getServiceObjectId()␣
↪is not None:
    attach_list = []
    if len(after.getPhotos()) > 0:
        body_photo = list(map(lambda attach_name: {
            'path': f"/department_files/photos/{attach_name.getName().replace(
↪'_', '/')}",
            'fileName': attach_name.getName().replace('/', '_'),
            'isUrl': True
        }
        , after.getPhotos()))
        for photo in after.getPhotos():
            photo_location = photo.getOriginLocation() if photo.
↪getOriginLocation() is not None else photo.getAttachmentLocation()
            if photo_location is not None and model.getLocation() is None:
                geom = {'coordinates': [(photo_location[0], photo_
↪location[1])], 'type': 'MultiPoint'}

        if model.getLocation() is not None:
            geom = {'coordinates': [(model.getLocation()[0], model.
↪getLocation()[1])], 'type': 'MultiPoint'}

    if len(after.getVideos()) > 0:
        for item in after.getVideos():
            attach_list.append(item.getName())
        attach_path = 'video'

    if len(after.getSounds()) > 0:
        for item in after.getSounds():
            attach_list.append(item.getName())
        attach_path = 'sounds'

    if len(after.getFiles()) > 0:
        for item in after.getFiles():
            attach_list.append(item.getName())
        attach_path = 'dif_files'

    body_files = list(map(lambda attach_name: {
        'path': f"/department_files/{attach_path}/{attach_name.replace('_', '/
↪')}",
```

```python
        'fileName': attach_name,
        'isUrl': True
    }
    , attach_list))

    req = patch(f"/layers/{after.getServiceObjectLayerId()}/features/{after.
 ↪getServiceObjectId()}/files")
    req.props({'photos': body_photo, 'files': body_files})

    return [
        req,
        patch(f"/layers/{after.getServiceObjectLayerId()}/features/{after.
 ↪getServiceObjectId()}").prop('geometry', geom)
        ]
```

**Checking the number of photos**

```python
if after.get('photo_count') is not None and after.get('photo_count') !=
 ↪before.get('photo_count') and subject.getId() != after.getUser().getId():
    return invalid("\nOnly the task creator can change the 'Number of Angles'
 ↪field.\n"
                "\nPlease return the previous value.")

if len(after.getPhotos()) is not None and after.get('photo_count') is not
 ↪None and after.getStatus().getId() != before.getStatus().getId():
    if len(after.getPhotos()) < int(after.get('photo_count')) and after.
 ↪getStatus().getId() == 4:
        send_photo_count = int(after.get('photo_count')) - len(after.
 ↪getPhotos())
        return invalid(f"Not enough photos to change the step. You need to
 ↪add more {send_photo_count}")
```

# 5.2 Attachment 2. Examples of advanced layer styles

For more information about creating geocss styles, see https://docs.geoserver.org/stable/en/user/styling/workshop/css/css.html.

** Example of a point layer style using standard icons (circles) for categories**

```css
/* @title Offices */
[fclass = 'bank'] {
 mark: symbol('circle');
}
[fclass = 'bank']
 :mark{
  fill:#68904D;
  stroke:black;
  stroke-width:1;
```

```
    size:18;
}


/* @title ATMs */
  [fclass = 'atm'] {
 mark: symbol('circle');
}
[fclass = 'atm']
:mark{
  fill:#EE9B01;
  stroke:#000000;
  stroke-width:1;
  size:9;
}
```

Where:

```
/* @title Offices */ - category name to be displayed in the legend.
[amenity = 'bank'] - the field to filter by and the value of the field.
mark: symbol('circle') - icon shape (circle).
fill:#68904D - icon fill color. You can use the name or the hex color code.
stroke: black/#000000 - color of the icon stroke. You can use the name or hex
→color code (black = #000000).
stroke-width:1 - icon stroke width in pixels.
size:18 - icon size in pixels.
```
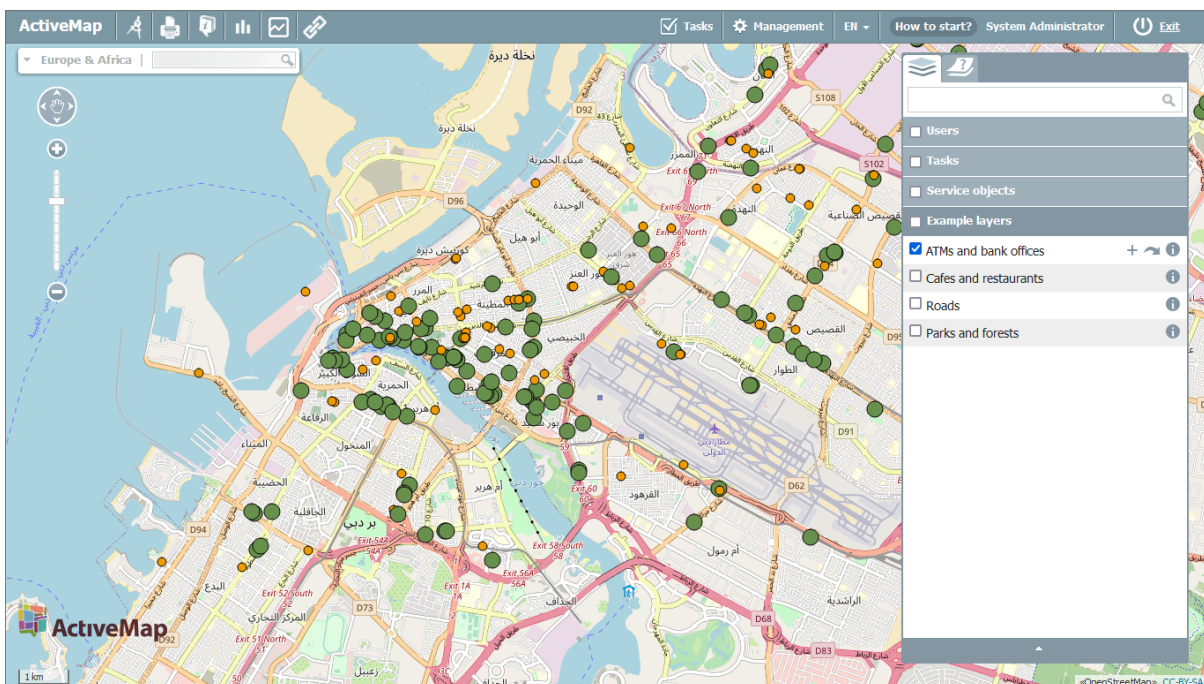


Fig. 5.2: Example of a point layer style for bank offices and ATMs using standard icons (circles) for categories

**Example of a point layer style using icons for categories**

```
 /* @title Fast food */
[ fclass = 'fast_food' ] {
   mark-opacity: 1;
   mark-rotation: 0;
   mark-size: 28;
   mark: url(https://public.activemap.me/dictionary/icons/79/view);
}
/* @title Cafes */
[ fclass = 'cafe' ] {
   mark-opacity: 1;
   mark-rotation: 0;
   mark-size: 28;
   mark: url(https://public.activemap.me/dictionary/icons/78/view);
}
/* @title Restaurants */
[ fclass = 'restaurant' ] {
   mark-opacity: 1;
   mark-rotation: 0;
   mark-size: 28;
   mark: url(https://public.activemap.me/dictionary/icons/80/view);
}
```

Where:

```
/* @title Cafes */ - category name to be displayed in the legend.
[fclass = 'cafe'] - the field to filter by and the value of the field.
mark-opacity: 1 - icon transparency (can be changed from 0 - full␣
↪transparency to 1 - full opacity).
mark-rotation: 0 - angle of icon rotation in degrees.
mark-size: 28 - icon size in pixels.
url(https://public.activemap.ru/dictionary/icons/78/view) - link to the icon.␣
↪You can get it by going to the "Layers" block, the "Icons" tab, right-
↪clicking on the icon and selecting "Copy image URL" (:ref:`icon_url`).
```
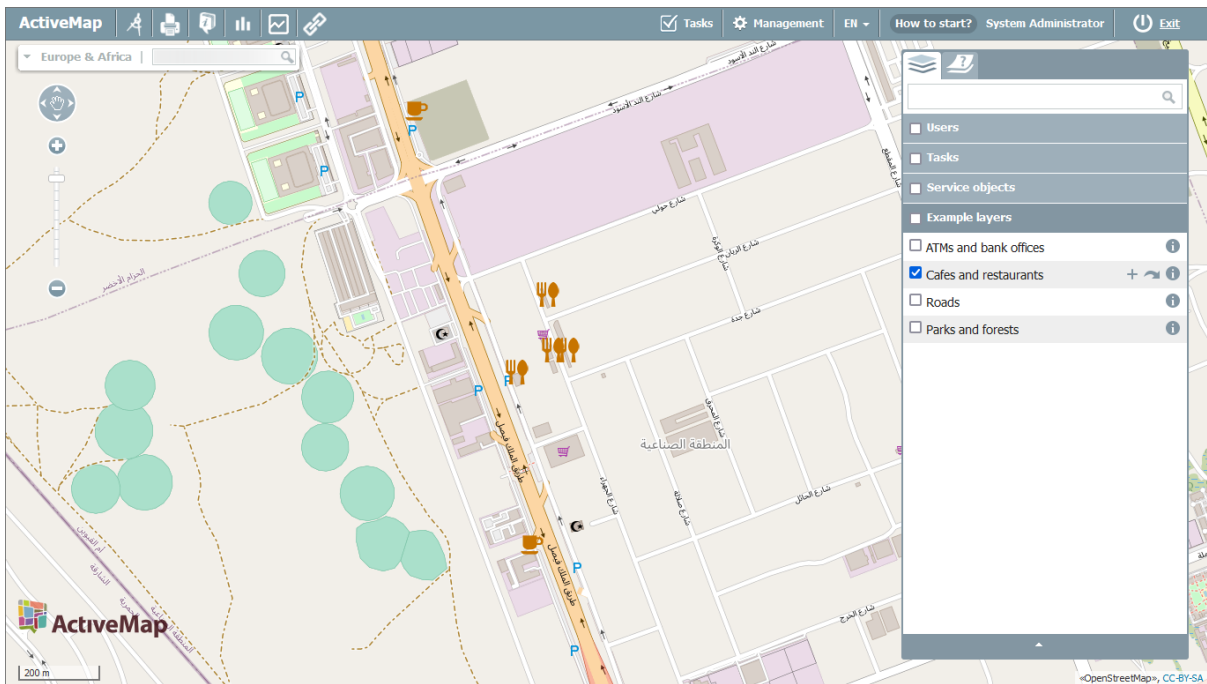
Fig. 5.3: Example of a point layer style for cafes and restaurants using icons for categories

**Example of a linear layer style with labels and line color information from the data**

```
* {
  stroke: [stroke_color];
  stroke-dashoffset: 0;
  stroke-linecap: butt;
  stroke-width: 4;
  label: [naimen];
  font-family: Arial;
  font-weight: bold;
  font-fill: black;
  font-size: 12;
  halo-color: white;
  halo-radius: 2;
  -gt-label-follow-line: true;
  -gt-label-max-angle-delta: 60;
  -gt-label-max-displacement: 400;
  -gt-label-repeat: 300;
}
```

Where:

```
stroke: [stroke_color] - color of the line stroke. In this case it is taken␣
→from the value of the specified field (stroke_color) and is not displayed␣
→in the legend.
stroke-dashoffset: 0 - offset of the stroke from its initial position in␣
→pixels.
stroke-linecap: round - parameter defining the shape of line ends (round -␣
→rounded corners, butt - cut at a right angle immediately after the end of␣
```

```
→the line, square - cut at a right angle after a distance equal to half the␣
→stroke-width).
stroke-width: 4 - line width in pixels.
label: [naimen] - name of the field which values are used to label objects.
font-family: Arial - font family for labeling objects.
font-weight: bold - font weight (thickness of label characters).
font-fill: black - font color. You can use name or hex color code (black =
→#000000).
font-size: 10 - font size in pixels.
halo-color: white - color of the caption stroke. You can use name or hex␣
→color code (black = #000000).
halo-radius: 1 - radius of the label stroke in pixels.
-gt-label-follow-line: - following the contours of linear object labels.
-gt-label-max-angle-delta: 90 - maximum bending angle of the caption in␣
→degrees.
-gt-label-max-displacement: 400 - maximum displacement of the label in pixels.
-gt-label-repeat: 150 - repetition of object's caption through the specified␣
→number of pixels.
```

**Example of a linear layer style with captions and different line types for categories**

```
 * {
 label: [name];
 font-family: Arial;
 font-weight: bold;
 font-fill: black;
 font-size: 10;
 halo-color: white;
 halo-radius: 1;
 -gt-label-follow-line: true;
 -gt-label-max-angle-delta: 90;
 -gt-label-max-displacement: 400;
 -gt-label-repeat: 150;
 }

/* @title Motorways */
[stylegroup = 'motorway'] {
stroke: #d1386f, #db798f;
stroke-width: 8px, 6px;
stroke-linecap: round;
z-index: 8, 9;
}
/* @title Main Roads */
[stylegroup = 'mainroad'] {
stroke: #be9239, #f8ce8c;
stroke-width: 6px, 4px;
stroke-linecap: round;
z-index: 6, 7;
}
```

```
/* @title Streets */
[stylegroup = 'minorroad'] {
stroke: #d9d6d0, #fefefe;
stroke-width: 4px, 3px;
stroke-linecap: round;
z-index: 4, 5;
}
/* @title Alleys */
[stylegroup = 'service'] {
stroke: #d9d6d0, #fefefe;
stroke-width: 3px, 2px;
stroke-linecap: round;
z-index: 2, 3;
}
/* @title Pedestrian zones */
[stylegroup = 'noauto'] {
stroke: #f99589;
stroke-width: 3px;
stroke-dasharray: 5 2;
z-index: 1;
}
/* @title Others */
[stylegroup = 'other'] {
stroke: #d9d6d0;
stroke-width: 2px;
z-index: 0;
}
```

Where:

```
1. General parameters determined for the entire layer:

label: [name] - name of the field which values are used to label objects.
font-family: Arial - font family for labeling objects.
font-weight: bold - weight of the font (the thickness of the caption␣
↪characters).
font-fill: black - font color. You can use name or hex color code (black =
↪#000000).
font-size: 10 - font size in pixels.
halo-color: white - stroke color of the caption. You can use name or hex␣
↪color code (black = #000000).
halo-radius: 1 - radius of the signature stroke in pixels.
-gt-label-follow-line: - follow line feature labels.
-gt-label-max-angle-delta: 90 - maximum label bend angle in degrees.
-gt-label-max-displacement: 400 - maximum label offset in pixels.
-gt-label-repeat: 150 - repeats the label of the object after a given number␣
↪of pixels.

2. Parameters for individual categories:
```

```
Simple line:

/* @title Others */ - category name to be displayed in the legend.
[stylegroup = 'other']- field that is filtered and field value.
stroke: #d9d6d0 - color of lines. You can use name or hex color code (black =
↪#000000).
stroke-width: 2px - line width in pixels.
z-index: 0 - the order in which the category is shown relative to other layer␣
↪categories (starts with 0, objects with z-index: 0 is displayed below all␣
↪other objects with larger index values).


Dotted line:

/* @title Pedestrian zones */ - category name to be displayed in the legend.
[stylegroup = 'noauto'] - the field that is filtered and the value of the␣
↪field.
stroke: #f99589 - color of the lines. You can use name or hex color code␣
↪(black = #000000).
stroke-width: 3px - line width.
stroke-dasharray: 5 2 - length of strokes (5) and spaces (2) in pixels.
z-index: 1 - the order in which the category is shown relative to other layer␣
↪categories.


Stroke line:

/* @title Motorway */ - category name to be displayed in the legend.
[stylegroup = 'motorway'] - the field that is filtered and the value of the␣
↪field.
stroke: #d1386f, #db798f - colors of strokes for lines. You can use name or␣
↪hex color code (black = #000000).
stroke-width: 8px, 6px - line width in pixels.
stroke-linecap: round - parameter defining the shape of the line ends (round␣
↪- rounded corners, butt - right angle break right after the line ends,␣
↪square - right angle break after half of the stroke-width).
z-index: 8, 9 - order of category showing relative to other layer categories␣
↪and lines within one category when imitating style with stroke-width.
```

For lines in CSS there is no concept of "fill", only "stroke". Thus, unlike points and polygons, it is not possible to style the "edge" of a line. However, this effect can be achieved by drawing each line twice: once with a certain width and again with a slightly smaller width. This creates the illusion of fill and stroke. The style uses CSS support for "multi-valued properties" with two colors and widths. In this case, the highways are colored first with a dark red line (#d1386f) of 8 pixels wide and then a thinner pink line (#db798f) of 5 pixels wide. Since each line is drawn twice, the render order determined by the z-index parameter is important. The wider line must have a smaller index value so as not to overlap the thinner line.
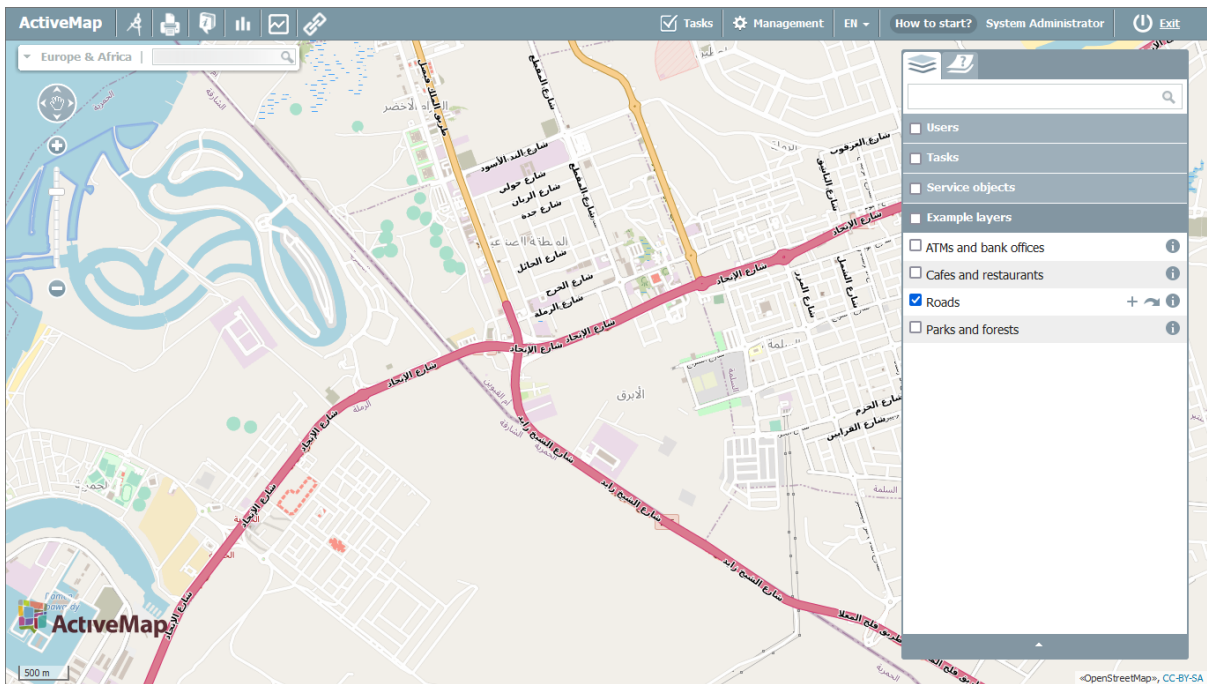
Fig. 5.4: Example of the road network linear layer style with captions and different line types for categories

**Example of a polygon layer style with fill by range**

```
*{
    fill-opacity:0.7;
    stroke:#254911;
    stroke-width:1;
  font-family: "Times New Roman";
  font-style: "normal";
  font-weight: "bold";
  font-size:10;
  font-fill:#000000;
  label-anchor: 0.5 0;
  label: [name];
  label-geometry: [centroid(the_geom)];
  -gt-label-max-displacement: 40;
  -gt-label-auto-wrap: 70;

}

/* @title Population < 20000 people */
[population_num < 20000] {
  fill:  #BDD880;}

/* @title Population between 20000 and 50000*/
 [population_num > 20000 and population_num < 50000]{
  fill:  #FFEB84;}

/* @title Population between 50000 and 100000*/
```

(continues on next page)

```
 [population_num > 50000 and population_num < 100000]{
  fill:  #FDBA7B;}


/* @title Population > 100000*/
 [population_num > 100000]{
  fill:  #F8696B;}
```

Where:

```
1. General parameters determined for the entire layer:

fill-opacity:0.7 - transparency of the polygon fill (ranges from 0 to 1).
stroke:#254911 - polygon stroke color. You can use the color name or␣
↪hexadecimal code.
stroke-width:1 - polygon stroke width in pixels.
font-family: "Times New Roman" - font family for labeling objects.
font-style: "normal" - font style (normal, italic).
font-weight: "bold" - weight of the font (weight of the caption characters).
font-size:10 - font size.
font-fill:#000000 - color of the signature symbols. You can use name or hex␣
↪color code (black = #000000).
label-anchor: 0.5 0 - anchor point that determines the placement of the label␣
↪relative to the centroid of the polygon. In this case, the label is offset␣
↪50% horizontally from the polygon's centroid and centered vertically.
label: [name] - name of the field which values are used to label objects.
label-geometry: [centroid(the_geom)] - relative location of the label␣
↪(location relative to the centroid).
-gt-label-max-displacement: 40 - maximum offset of the label in pixels␣
↪relative to the centroid of the polygon.
-gt-label-auto-wrap: 70 - breaks the label into lines if its length exceeds␣
↪the specified value in pixels.


2. Parameters for individual ranges:

/* @title Population < 20000 */*/ - range name to be displayed in the legend.
[population_num < 20000] - field to filter by and the value of the field.
fill: #BDD880 - the color of the polygon fill for the specified range. You␣
↪can use the color name or hexadecimal code.
```

**Example of a polygon layer style with hatching by category**

```
/* @title Parks */
  [fclass = 'park'] *{
  fill: symbol('shape://times');
  fill-size: 22px;
  stroke: darkgreen;
  }
  :fill {
  stroke: green;
  size: 8;
```

```
}

  /* @title Forests */
  [fclass = 'forest'] *{
  fill: symbol('shape://plus');
  fill-size: 12px;
  stroke: darkbrown;
  }
  :fill {
  stroke: brown;
  size: 8;
}
```

Where:

```
/* @title Parks */*/*// - name of the category to be displayed in the legend.
[natural = 'park'] - field to filter by and the value of the field.
fill: symbol('shape://times') - fill polygon with symbols for creating a⌐
↪shifting effect.
fill-size: 22px - size of symbols to fill in pixels.
stroke: darkgreen - color of polygons stroke.
:fill {
  stroke: green; - color of the characters stroke.
  size: 8; - symbols stroke thickness.
}
```
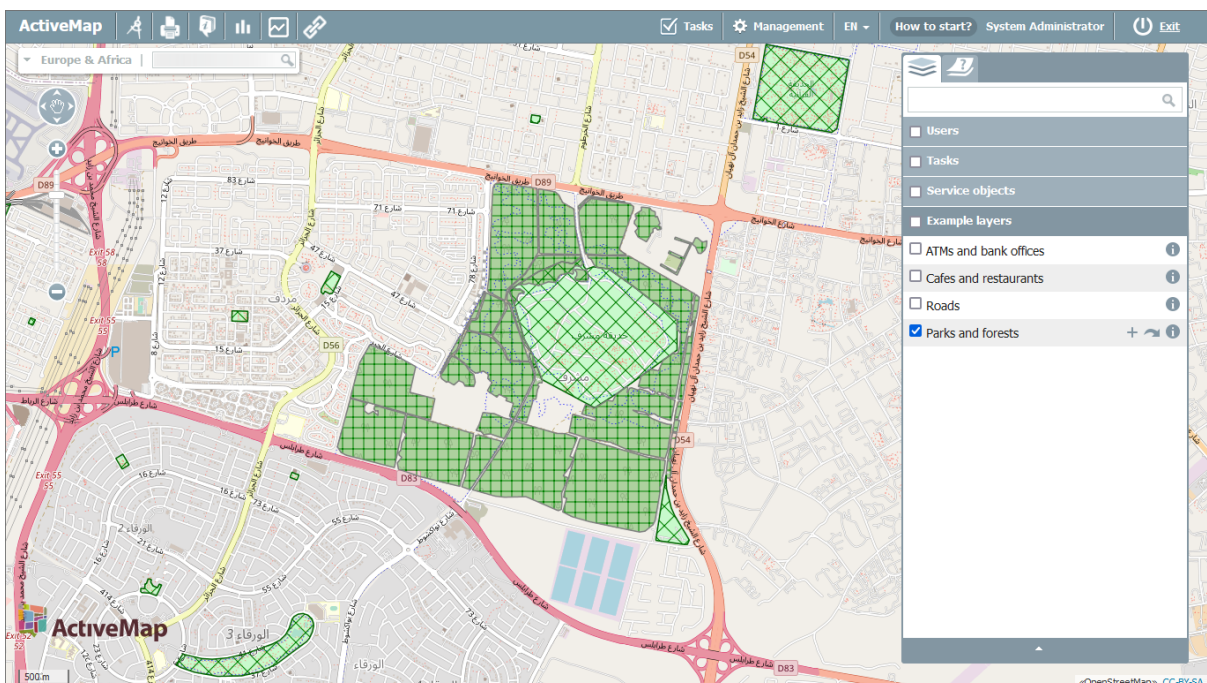


Fig. 5.5: Example of a polygon layer style for vegetation with hatching by category

---

**5.2. Attachment 2. Examples of advanced layer styles** 282

## 5.3 Attachment 3. Examples of baselayer settings and parameters

**Yandex**

```
Class name - M.Yandex;

Keys:
map - scheme,
satellite - satellite,
hybrid - hybrid (satellite + scheme),
publicMap - people's map,
publicMapHybrid - hybrid (satellite + people's map);

Map parameters - traffic: true|false (enable/disable traffic events layer).
```

**OpenStreetMaps**

```
Class name - M.TileLayer.OpenStreetMap.
```

**Dynamically created**

**M.TileLayer**

```
Class name - M.TileLayer;
url - http://{s}.domain/path/{x}/{y}/{z}.png;

Parameters:
minZoom: 0 - minimum zoom;
maxZoom: 18 - maximal zoom;
tileSize: 256 - tile size;
subdomains: 'abc' - subdomains;
errorTileUrl: '' - url of the tile with an error;
attribution: '' - description of copyright holders;
opacity: 1 - transparency;
scheme: '' - type xyz or tms;
zoomOffset: 0 - zoom shift;
crs: M.CRS.EPSG900913 - projection.
```

**M.TileLayer.WMS**

```
Class name - M.TileLayer.WMS;
url - http://domain/path;

Parameters:
minZoom: 0 - minimum zoom;
maxZoom: 18 - maximal zoom;
tileSize: 256 - tile size;
errorTileUrl: '' - url of the tile with an error;
attribution: '' - description of the copyright holders;
opacity: 1 - transparency;
zoomOffset: 0 - zoom shift;
crs: M.CRS.EPSG900913 - projection;
```

```
version: ''1.1.1'' - WMS version;
layers: '' - list of layers;
styles: '' - list of styles;
format: '' - image/jpeg' - image format;
transparent: '' - transparency.


Class name - M.TileLayer.WMS;
 url - http://domain/path;
```

## 5.4  Attachment 4. User rights grid

Table 5.2: us

| Rights/ Role | System Administrator | System Inspector |
|---|---|---|
|  |  |  |
| Change titles | + | – |
| Edit description | + | – |
| Change custom fields | + | +[2] |
| Change priority | + | – |
| Change step | +[1] | +[1] |
| Change deadline | + | – |
| Change status | + | + |
| Change the set point | + | – |
| Add a service object to a created task, if it has not been added before | + | + |
| Assign an executor | +[2] | +[2] |
| Assign an organization | +[2] | +[2] |
| View the creator | + | + |
| Attach photos/ videos (gallery) | + | +/- |
| Attach photos/ videos (camera) | + | +/- |
| Work with the invoice module | + | +/- |
| Delete | + | – |
|  |  |  |
| View | + | + |
| Create | + | – |
| Edit | + | – |
|  |  |  |
| View | + | + |
| Create | + | – |
| Edit | + | – |
| Delete | + | – |
|  |  |  |
| View | + | + |
| Create | + | – |
| Edit | + | – |
| Assign to other users | + | – |
|  |  |  |

| Rights/ Role | System Administrator | System Inspector |
|---|---|---|
| Edit the name | + | + |
| Change login | + | + |
| Change password | + | + |
| Add a tag | + | + |
| Edit contact information | + | + |
| Change role | + | – |
| Change user type | + | + |
| Change main organization | – | – |
| Change additional organization | – | – |
| Add an avatar | + | + |
|  |  |  |
| View | + | + |
| Create | + | – |
| Edit | + | – |

**1**

   task at the "in progress" step.

**2**

   task at the "in progress" step or draft.

**3**

   task at the "completed" or "new" step.

**4**

   task at the "in progress" step, step is not closed or draft.

## T

## U

## V

## W