Smart Parking System

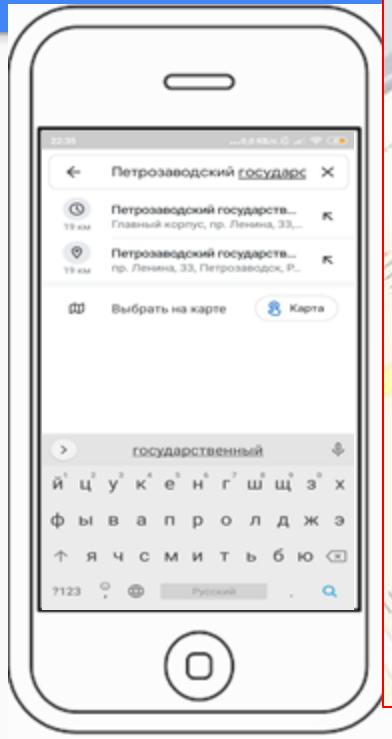
Made by Olga Masaeva

Part 1. Problem statement

- Conceptual model. Main functions. Service building scripts. The hardware used to run the agents. Similar existing smart application solutions. Intelligence. Application dimension

Idea

Creation of an intelligent parking space, allowing you to search, view, interact with a parking space



Main services:

- recommendation of a parking place at the selected destination
  - Disabled parking recommendation
  - Recommending parking spaces for staff/visitor only if the user is one
- parking warnings
- viewing the parking situation in a specific place
- Reservation of a parking space

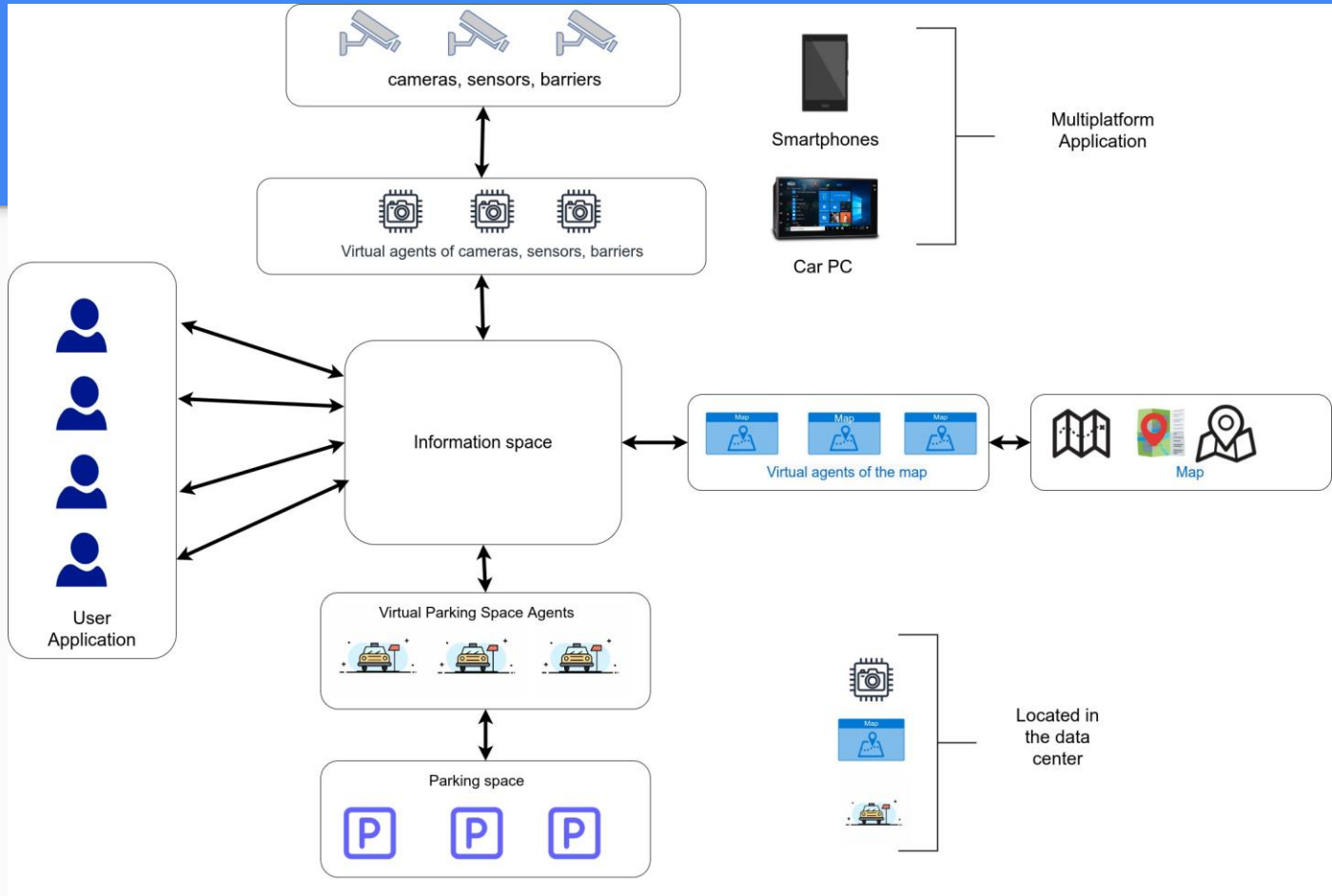
Equipment used:

Surveillance cameras

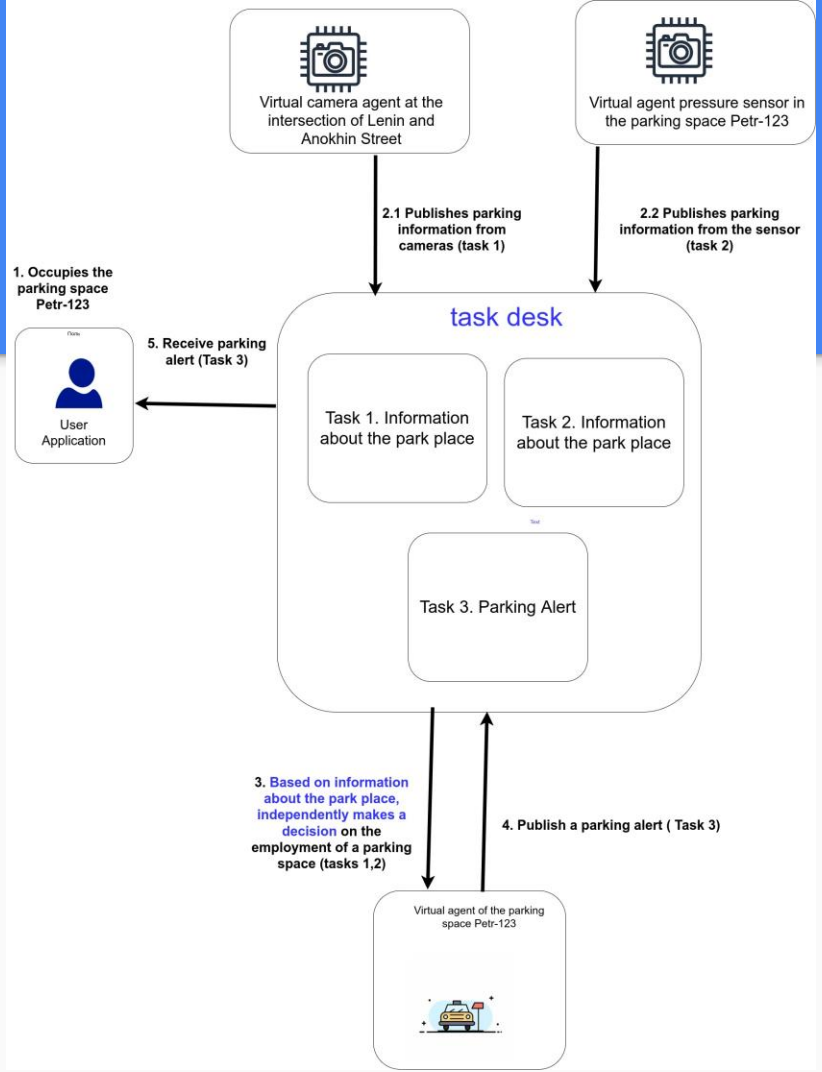
Pressure sensors

Automatic space reservation barriers

Hardware and software part

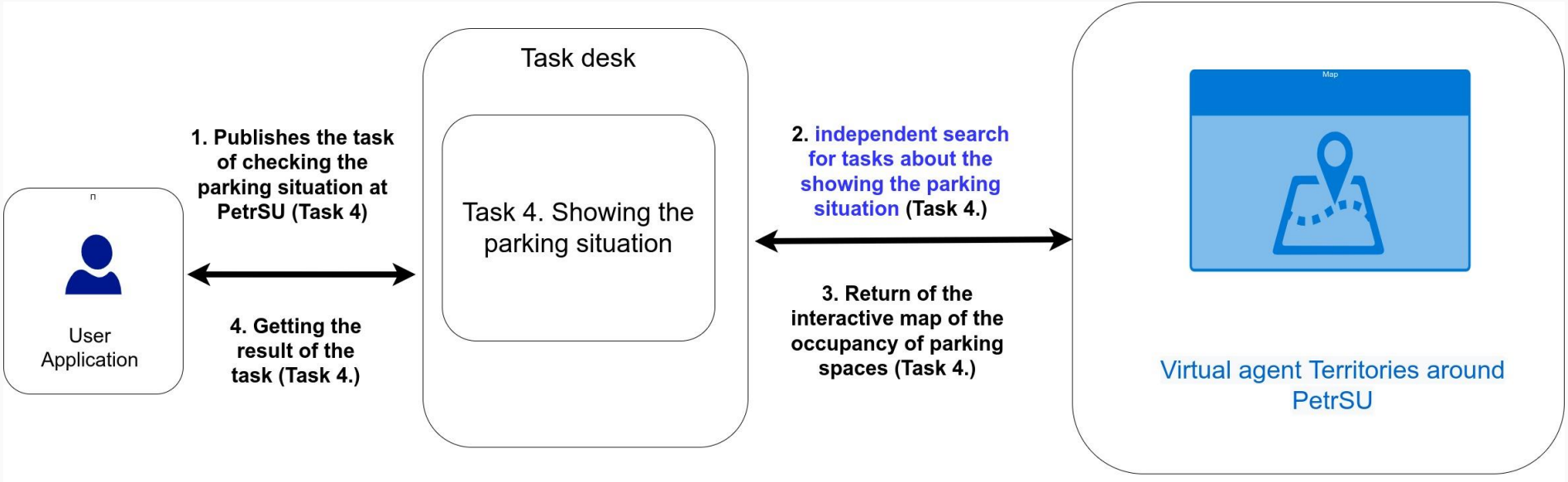


Parking Response System Scenario



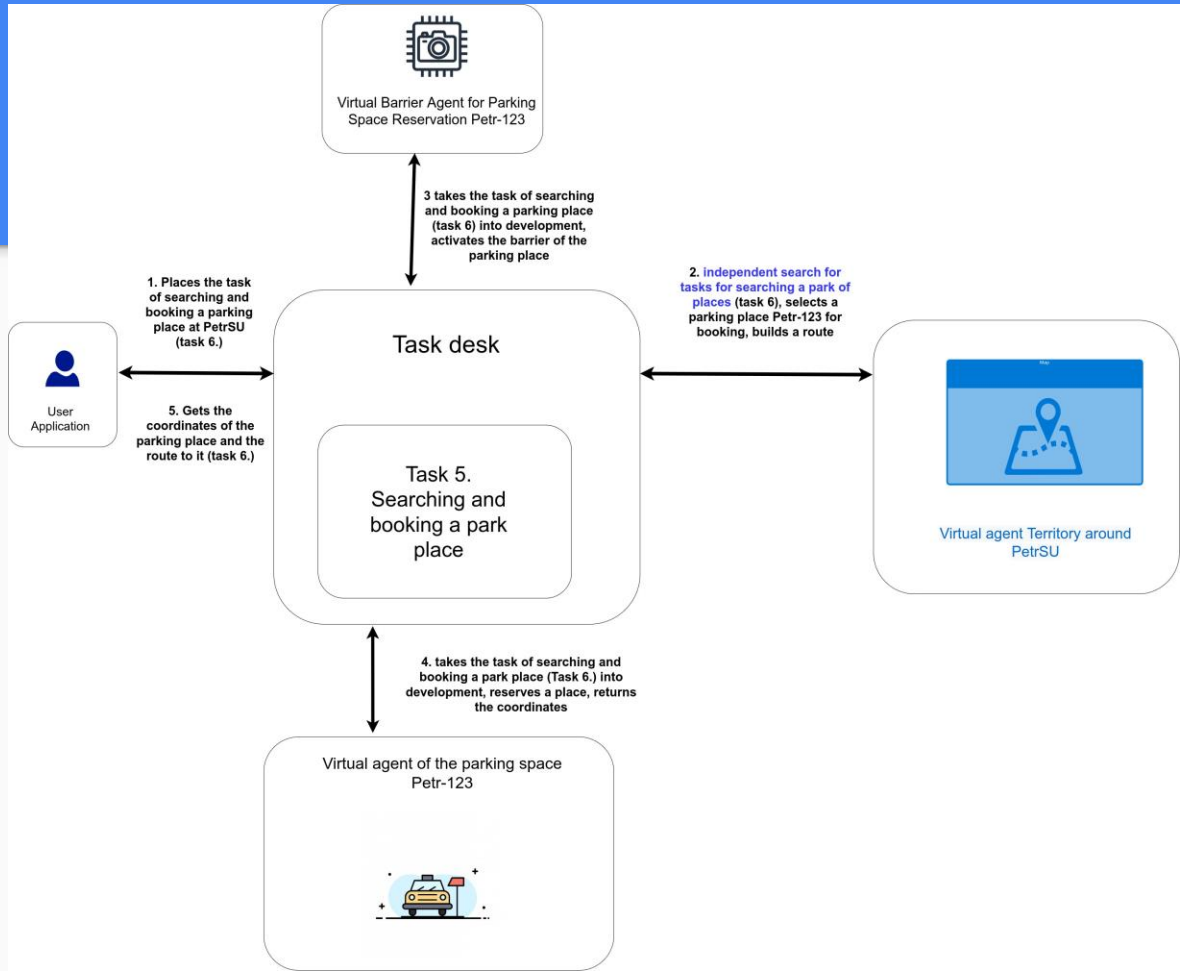
Blue represents intelligence

Scenario for viewing the parking situation in a specific place



Blue represents intelligence

Scenario for searching and booking a parking space at PetrSU



Blue represents intelligence

Analogues

Google maps, Yandex Maps, 2GIS and others.

Common features: only looking for a way to a destination, can find a place to park, but cannot find a place to temporary stay.

Analogues

- Smart parking systems such as iPark.

Common features: work in a strictly limited marked area inside the parking lot, traffic control with the help of light signals / mechanical actions

What we offer - Intelligent selection of a parking / parking place in the city

- Our app gives you real-time, personalized directions to a city parking/parking spot

Comparative characteristics

Feature	2GIS	Ipark	Out system
Working area	Everywhere in a city	Only inside of a parking space	Everywhere in a city
Finding a parking / temporary stay place nearby	Only parking space	Only parking space	Parking space and place for temporary stay
Application existance	+	-	+
Access to reserve a parking space	-	+	+
Access of making a route to a place	+	+	+

Minimum application dimensions

Camera agent

1 per camera/sensor/barrier

Number of users - N people

User's application

2*N applications, 2 per each user (assuming that the user will work from both Car PC and mobile device)

Parking space agent

1 per each parking space

Map agent

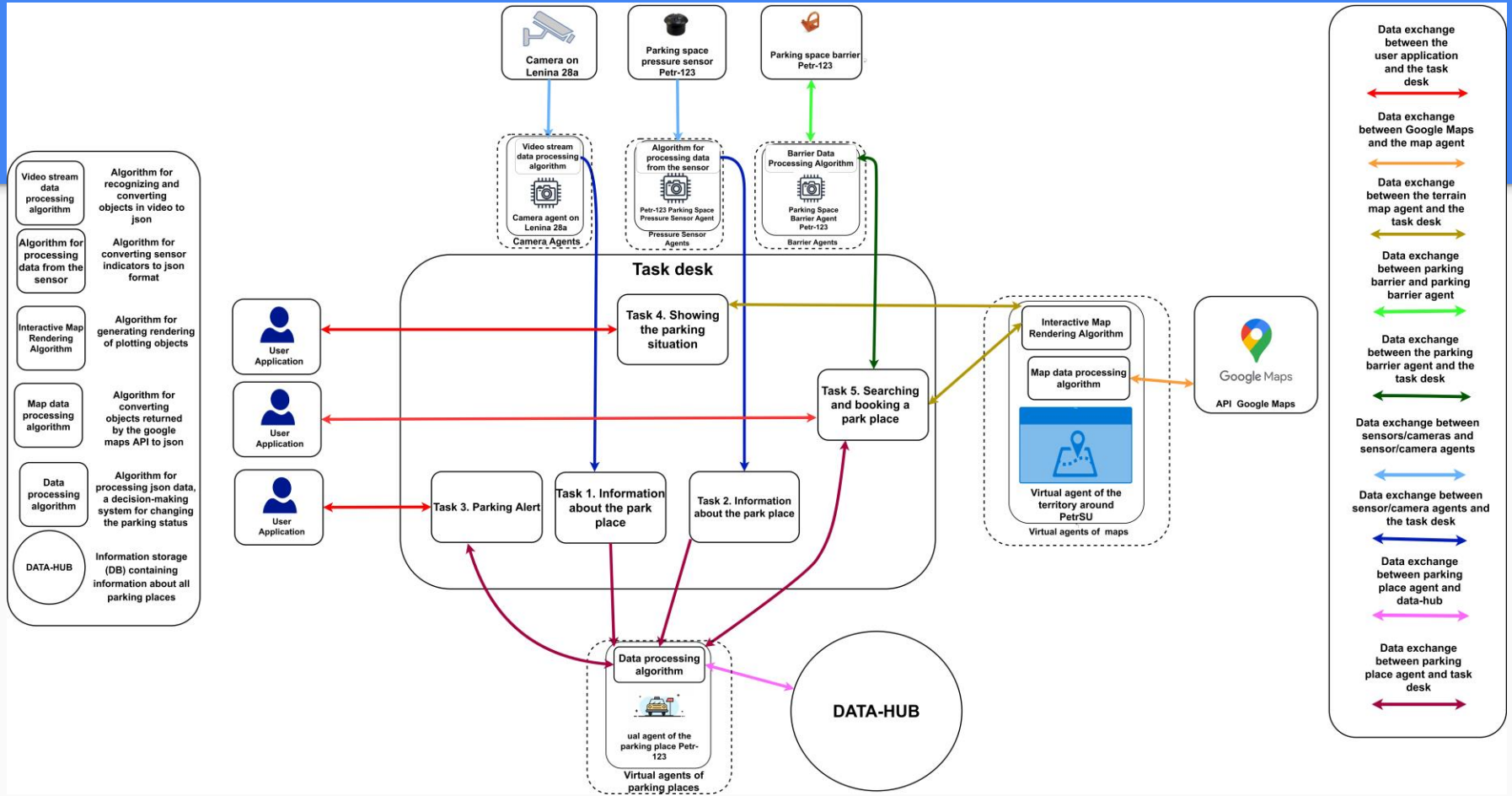
1 per each building in a city

For N users, the estimated data volume will be ≈50×N Mb/day.

Part 2: Multi-Agent Architecture and Detailed Design

- Detailing requirements in use cases. Common information space. Intelligence analysis within the developed use cases.

Detailed architecture



Detailed scenario of the parking response system

Blue represents intelligence

1. The user occupies the parking space Petr-123

2.1 The algorithm for processing the video stream of the camera agent continuously receives data from the camera, which it converts into data on the status of parking places

2.1 The algorithm for processing data from the sensor agent of the sensor continuously receives data from the pressure sensor, which it converts into data on the status of the parking place above it

3.1 Publication of processed data as task 1

3.2 Publication of processed data as task 2

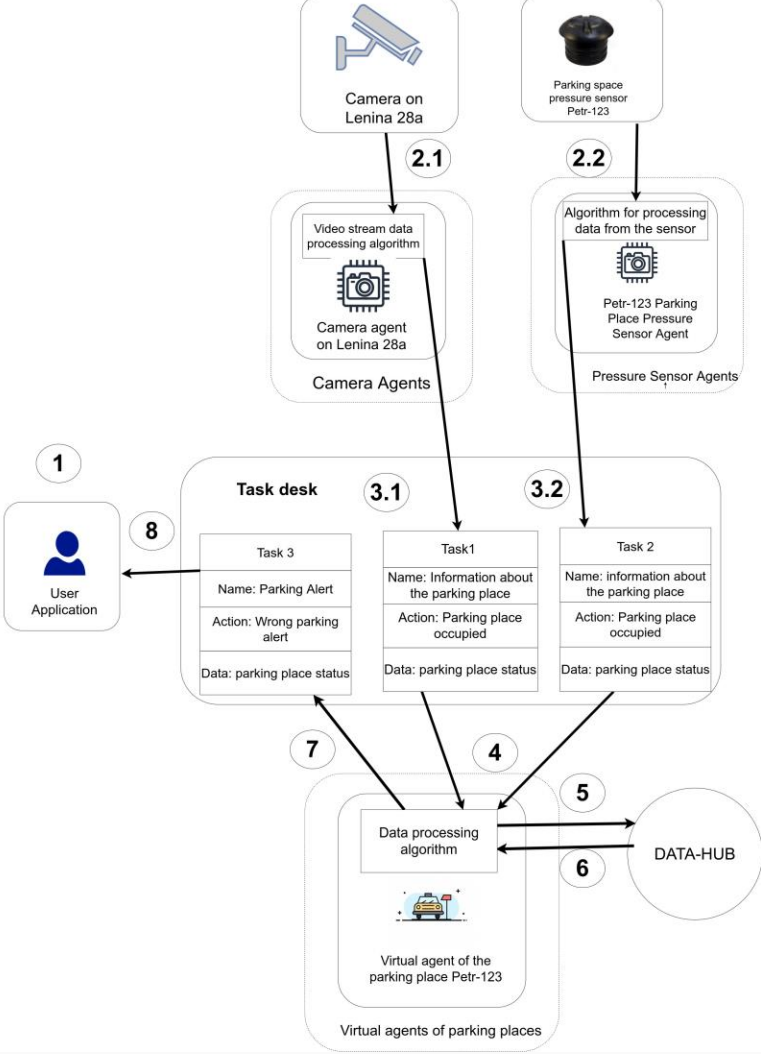
4. Obtaining by subscription the agent of the parking place Petr-123 of information about parking place occupation from task 1, 2

5. Forming a request in the data-hub to change the status of the parking place Petr-123 and obtain data about the user who occupied it

6. Getting the result of requests, processing

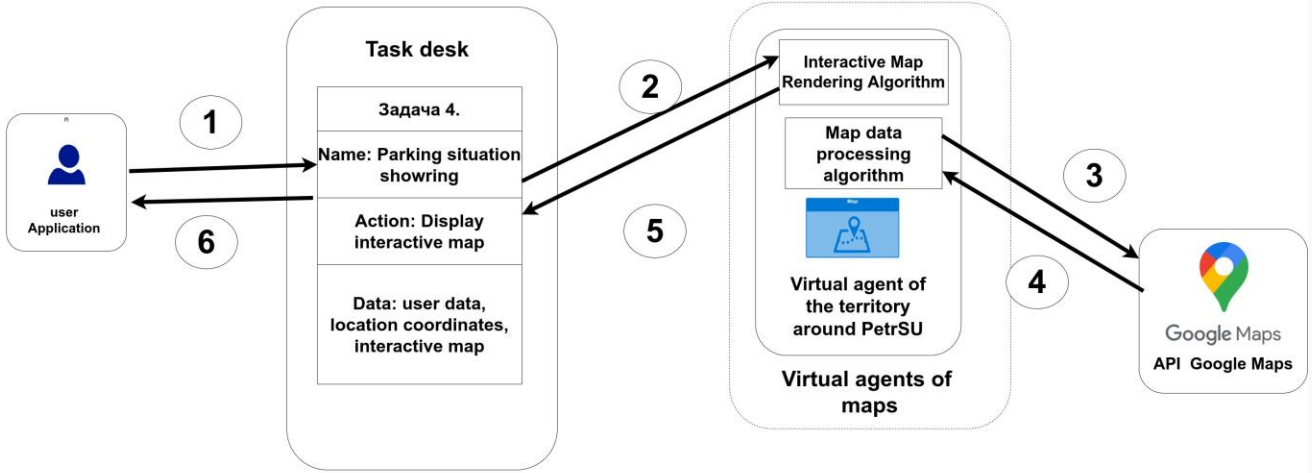
7. Publication of the task of alerting the user about parking

8. Subscription of a custom application to receive an alert for illegal parking



Scenario for viewing the parking situation in a certain place

1. Publication of the task with obtaining an interactive map around PetrSU
2. Obtaining by subscription the task of displaying an interactive map of parking places
3. 3 Forming a request for a map of the area from google maps
4. 4 Getting the result of a request to get a map of the area
5. 5. Publication of the processed data in the task
6. 6. Obtaining data on the task by the user application by subscription

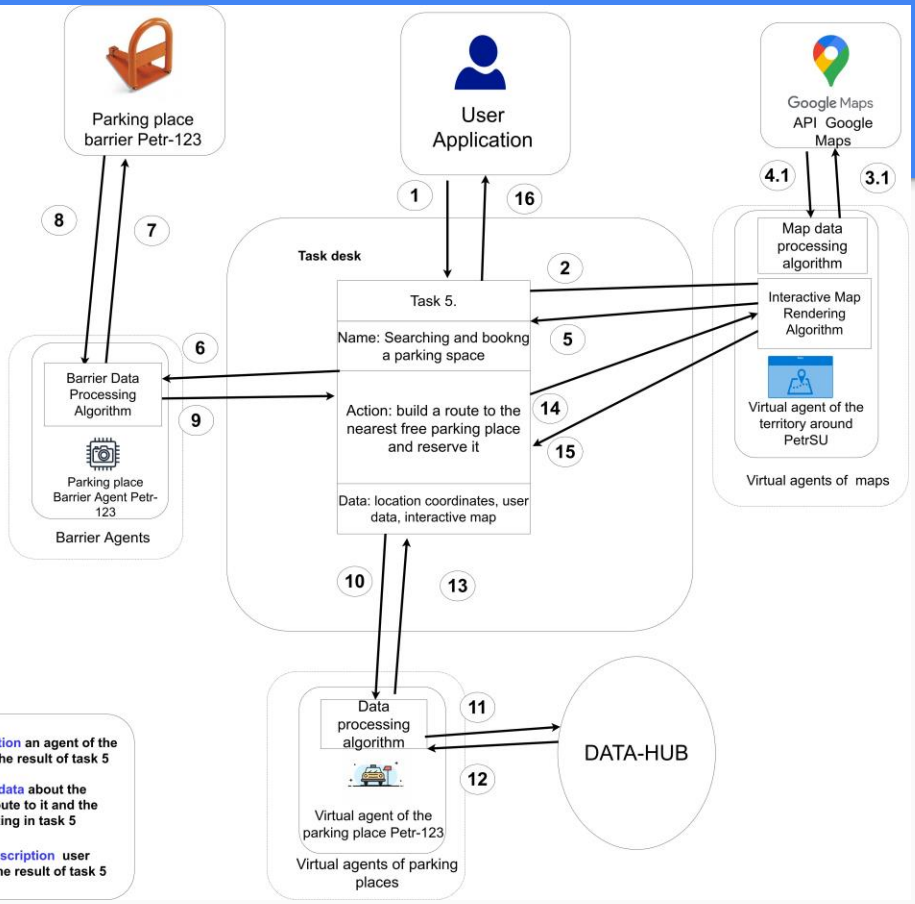


Blue represents intelligence

Detailed Scenario for searching and booking a parking space in a specific location

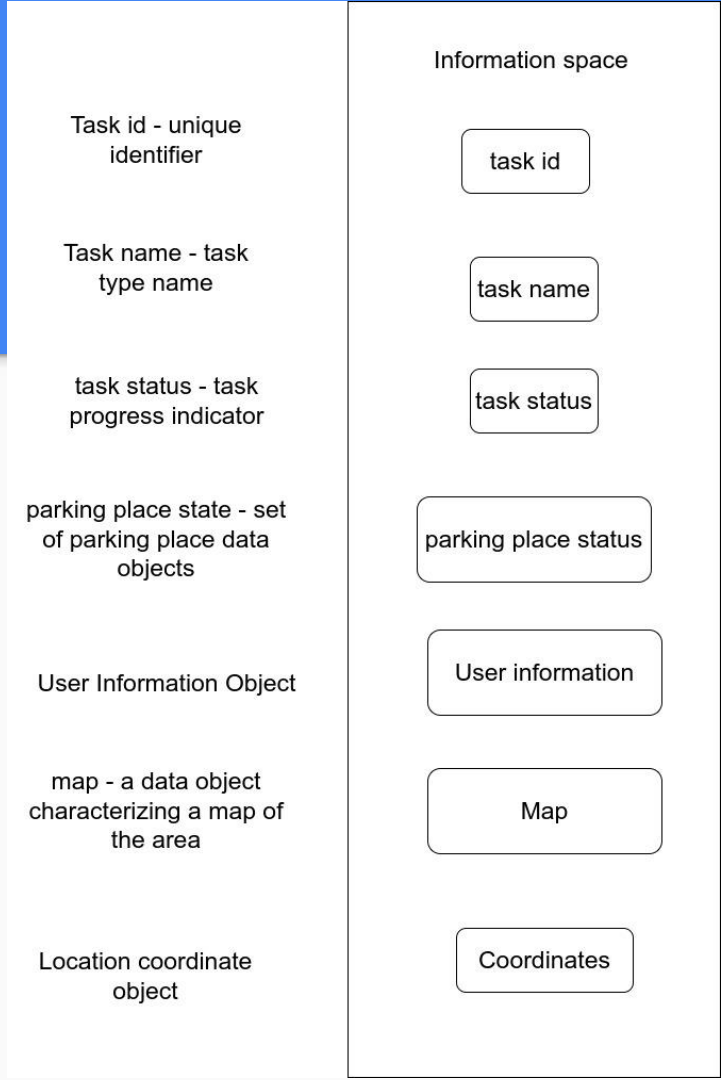
1. Publication of the task of constructing a route to the nearest free parking place at PetrSU, as well as its reservation
2. Obtaining by subscription an agent of the territory around PetrSU about the task of searching and booking a parking space
3. 3 Forming a request for a map of the area from google maps
4. 4. Getting the result of a request to get a map of the area
5. 5. Publication of the task of reserving and blocking a parking space Petr-123
6. 6. Obtaining by subscription agent blocking barrier for parking place Petr-123 activation tasks
7. 7. Forming for agents of the blocking barrier request to the blocking barrier for activation
8. 8. Receiving the result of the request, processing the data of the sensors of the barrier
9. 9. Publication of processed data in task 5
10. 10. Obtaining by subscription the agent of Parking place Petr-123 of the status update task (task 5)
11. 11. Forming a request to update the status of the parking place Petr-123 to Data-hub
12. 12. Get the query result
13. 13. Publication of processed data in task 5

14. Obtaining by subscription an agent of the area around PetrSU the result of task 5
15. Publication of the data about the parking place, the route to it and the success of blocking in task 5
16. Obtaining by subscription user application agent the result of task 5



Blue represents intelligence

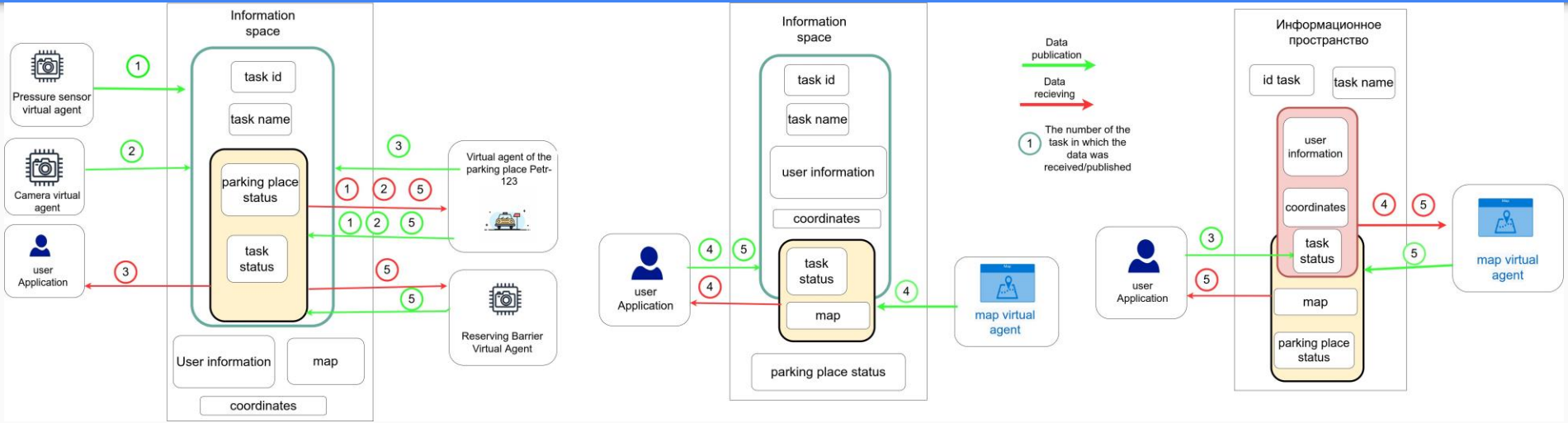
Common information space



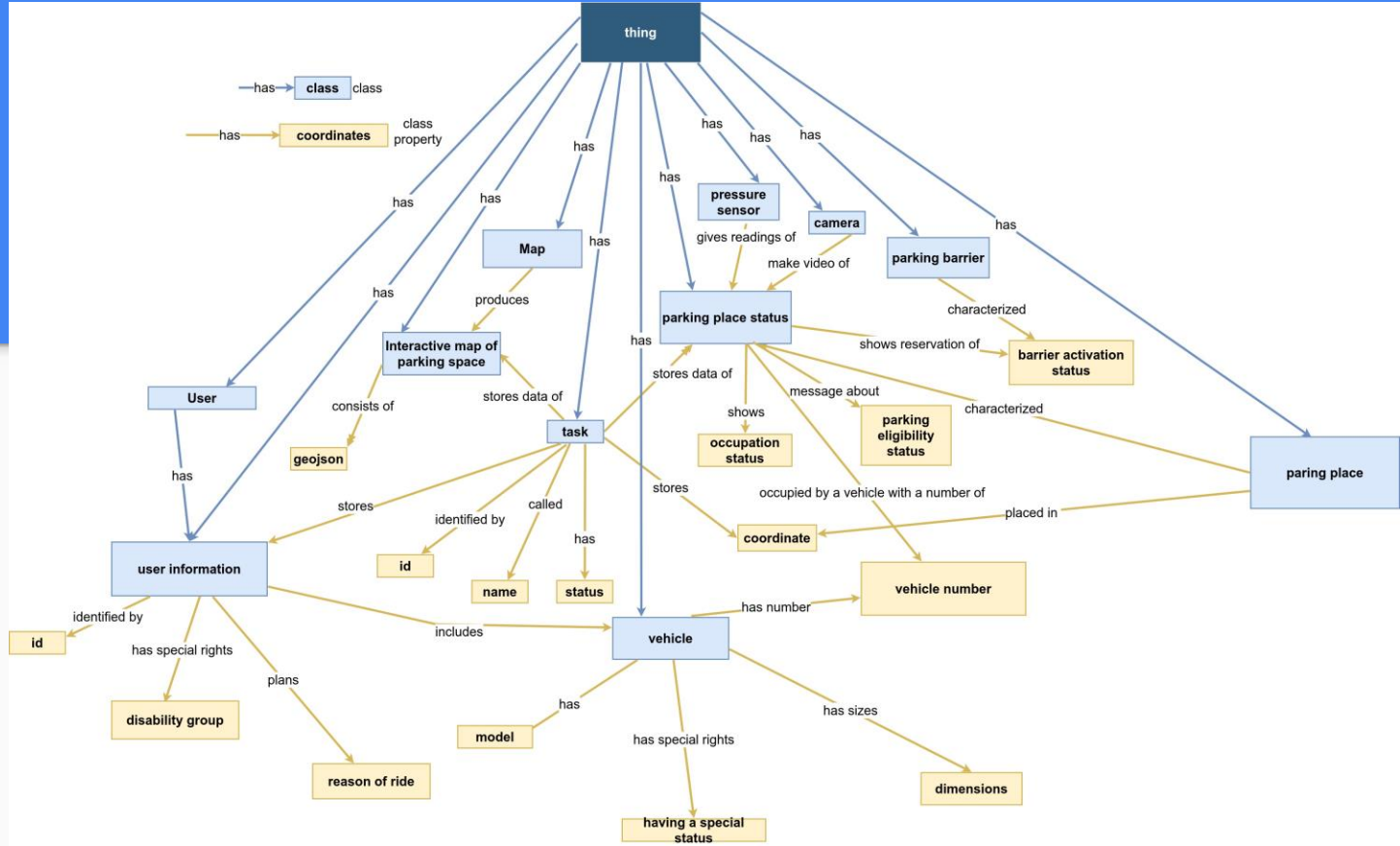
Part 3. Ontological Modeling

- Intellectual space as a knowledge base. Graph of ontological classes. Graph of ontological individuals.

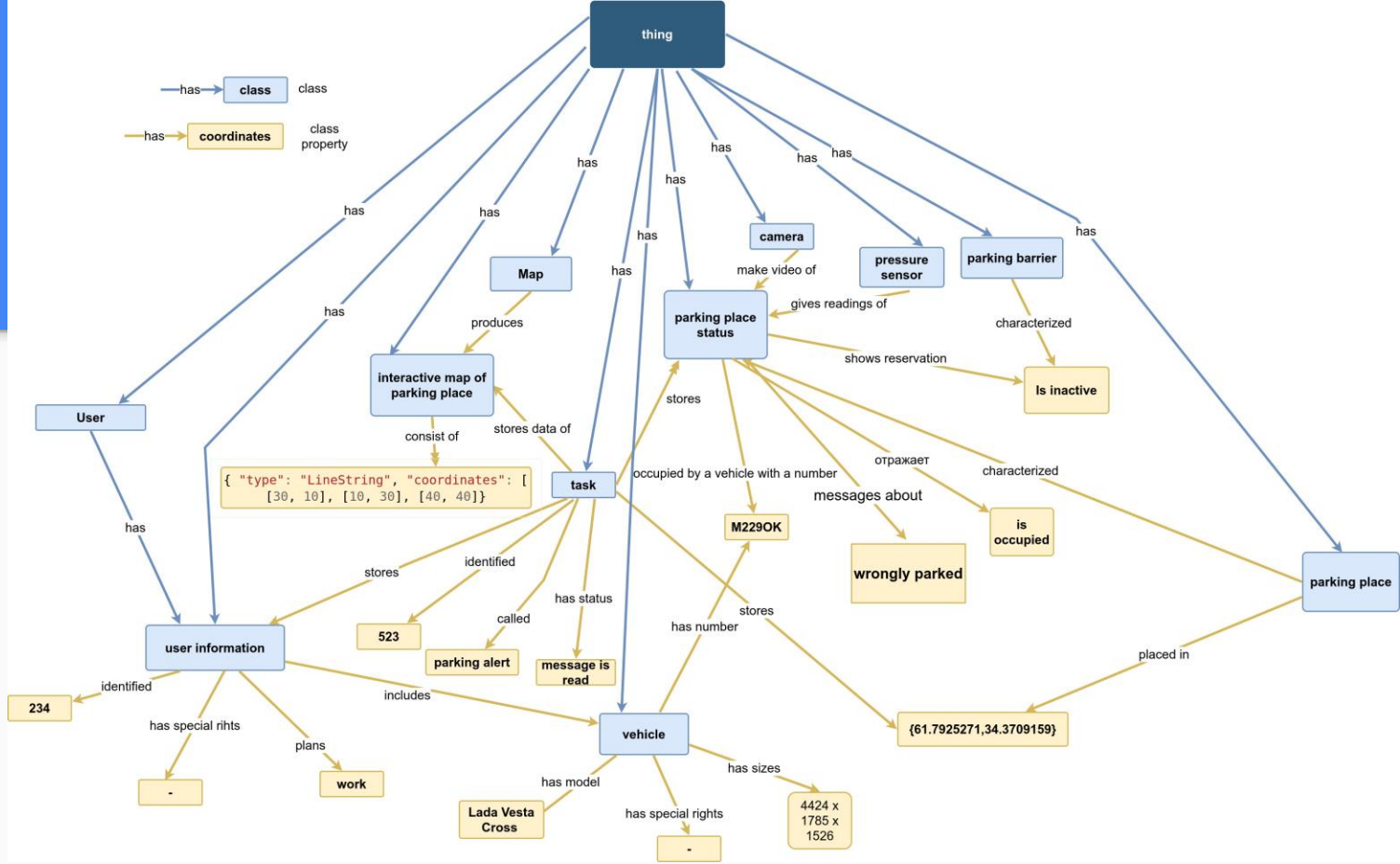
Intellectual space as a knowledge base



Ontological class graph



Graph of ontological individuals

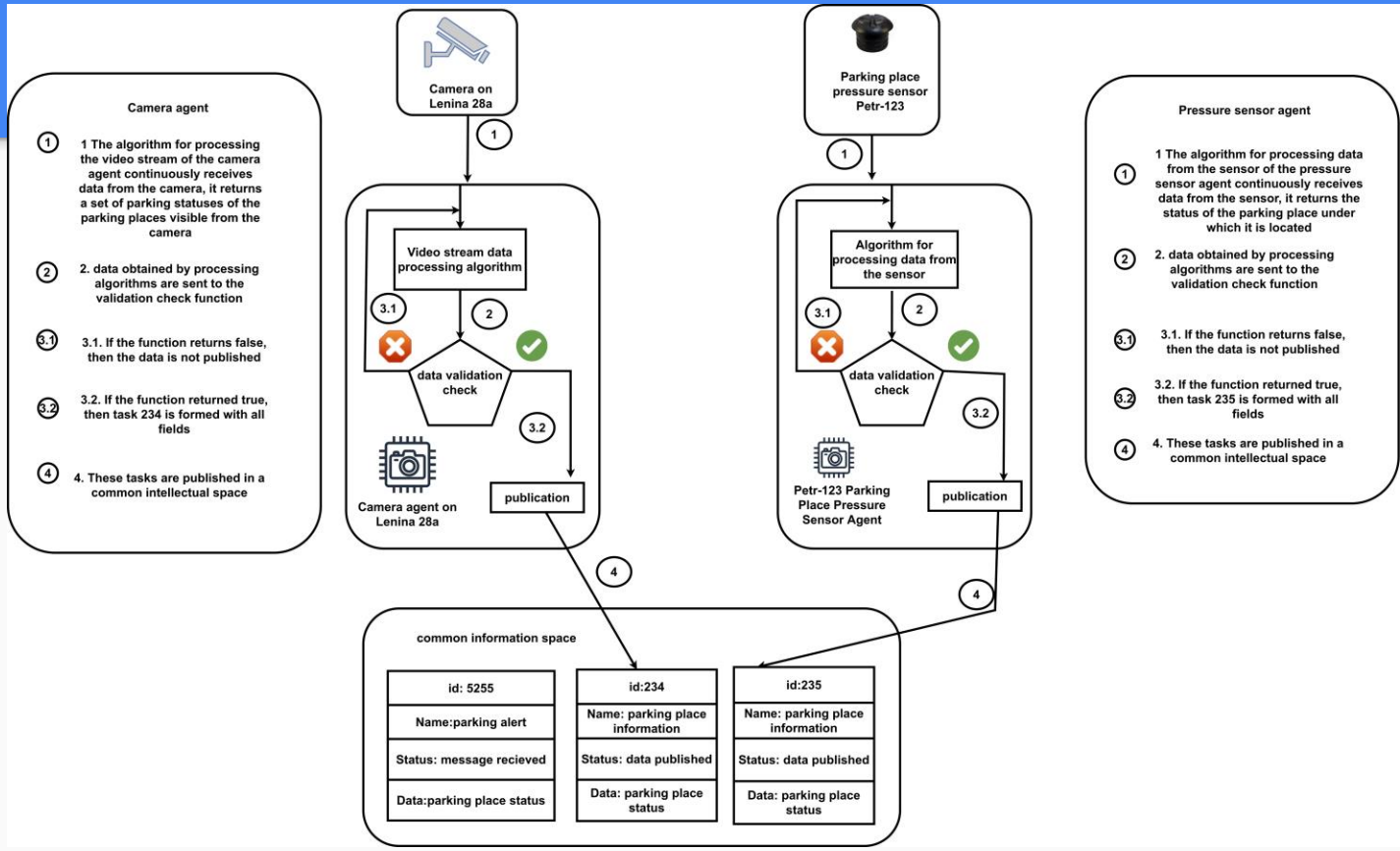


Part 4. Knowledge Processors

- The actions of each agent in each service building scenario. Sequence diagrams for scenarios. Actions to access the intellectual space. Actions to deliver the service to the user.

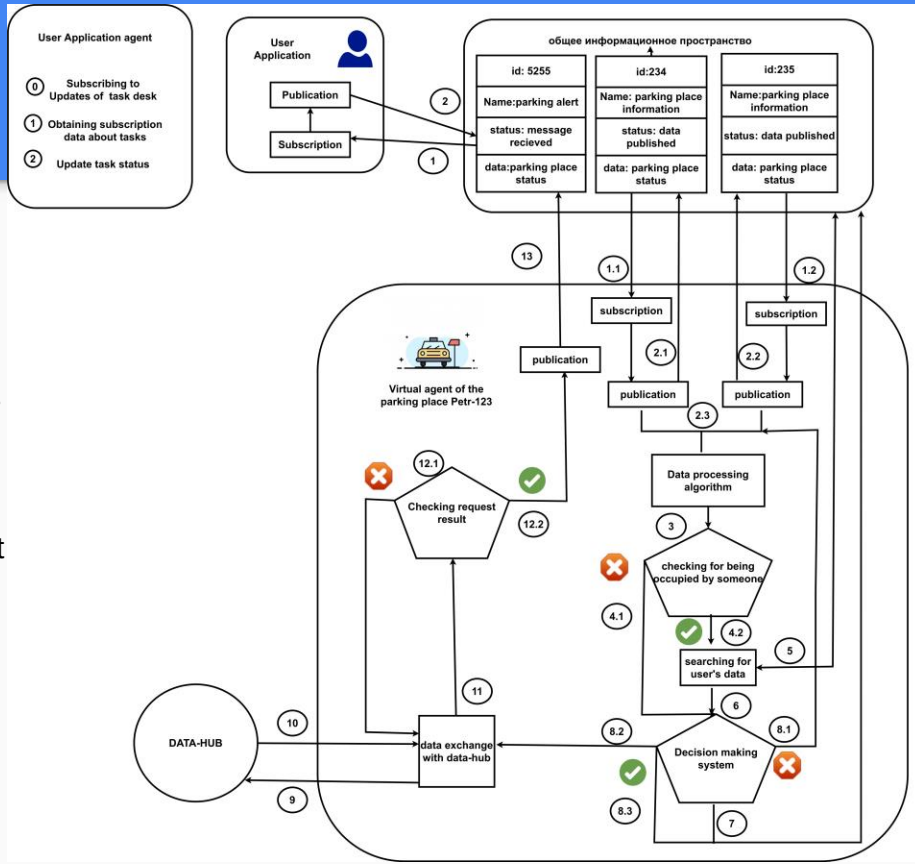
Parking Response System Scenario: Sensor and Camera Agents

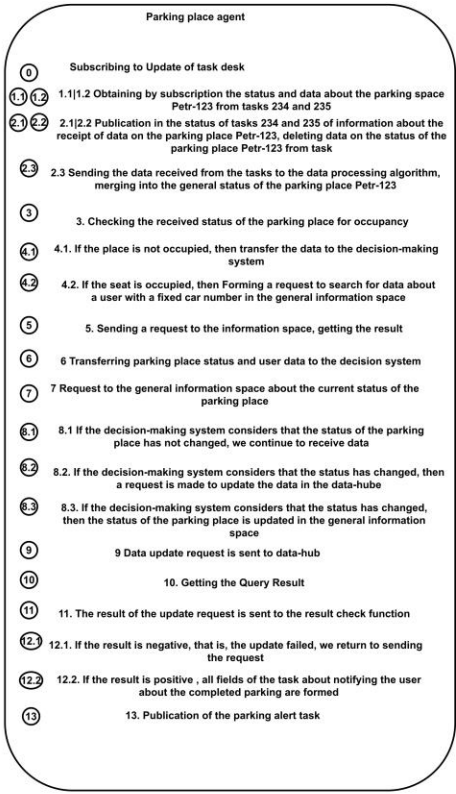
Actions to access the smart space:
5 - publish all task fields:
5.1 insert task
5.2 insert task fields



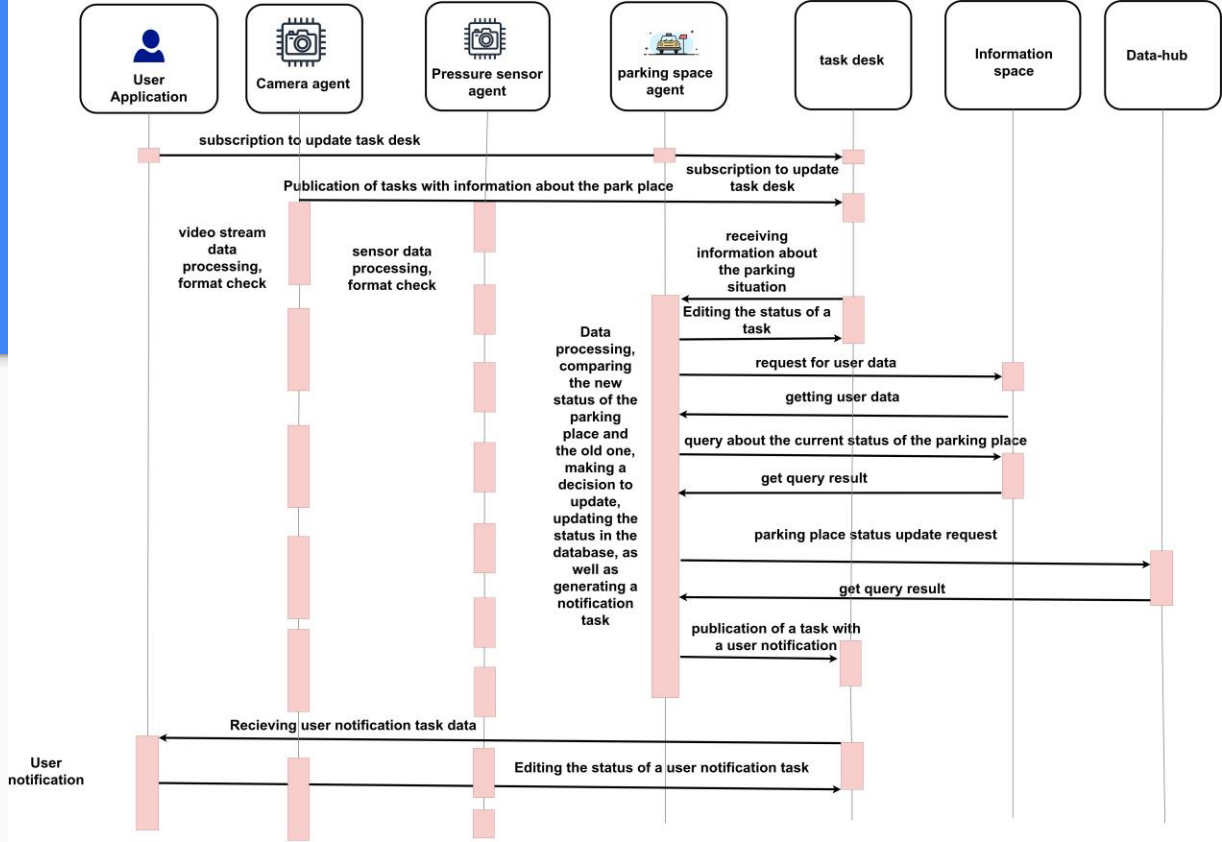
Parking Response System Scenario: Parking place Agents and Applications

Actions to access the smart space:
0 - subscribe: subscription by the agent of the parking place and the user application to changes in the task board
Agent Park Locations:
1.1,1.2 - query getting task fields 234 and 235
2.1,2.2 - publish tasks 234 and 235: status update
6 - query getting user information by car number
7.2 - query getting information about a parking space by id
15 – publish task 5255:insert task, insert fields
Application agent
1 - query getting information about the state of the park place and status
2 – update notification task status

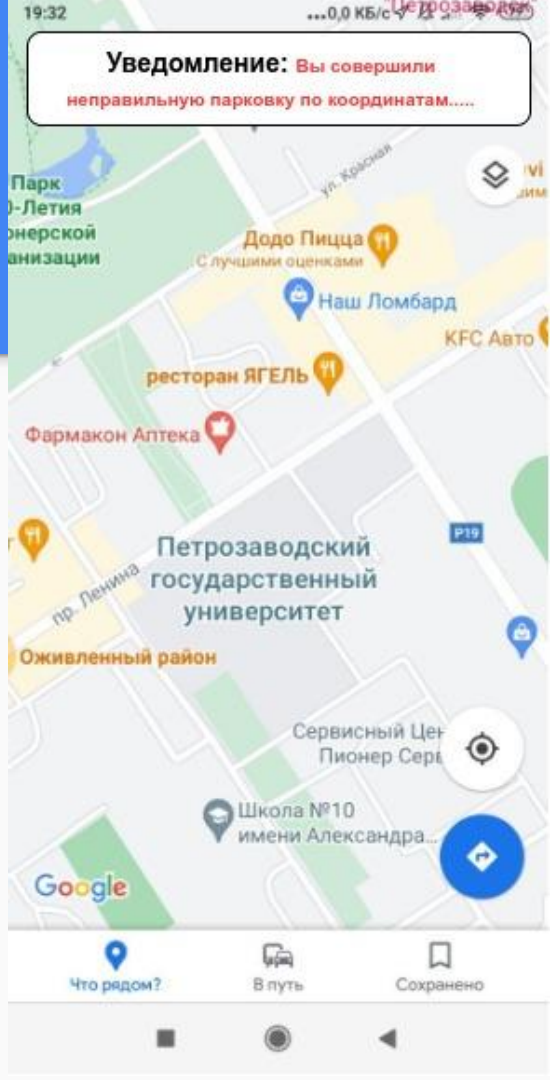




Sequence Diagram for Parking Response Scenario



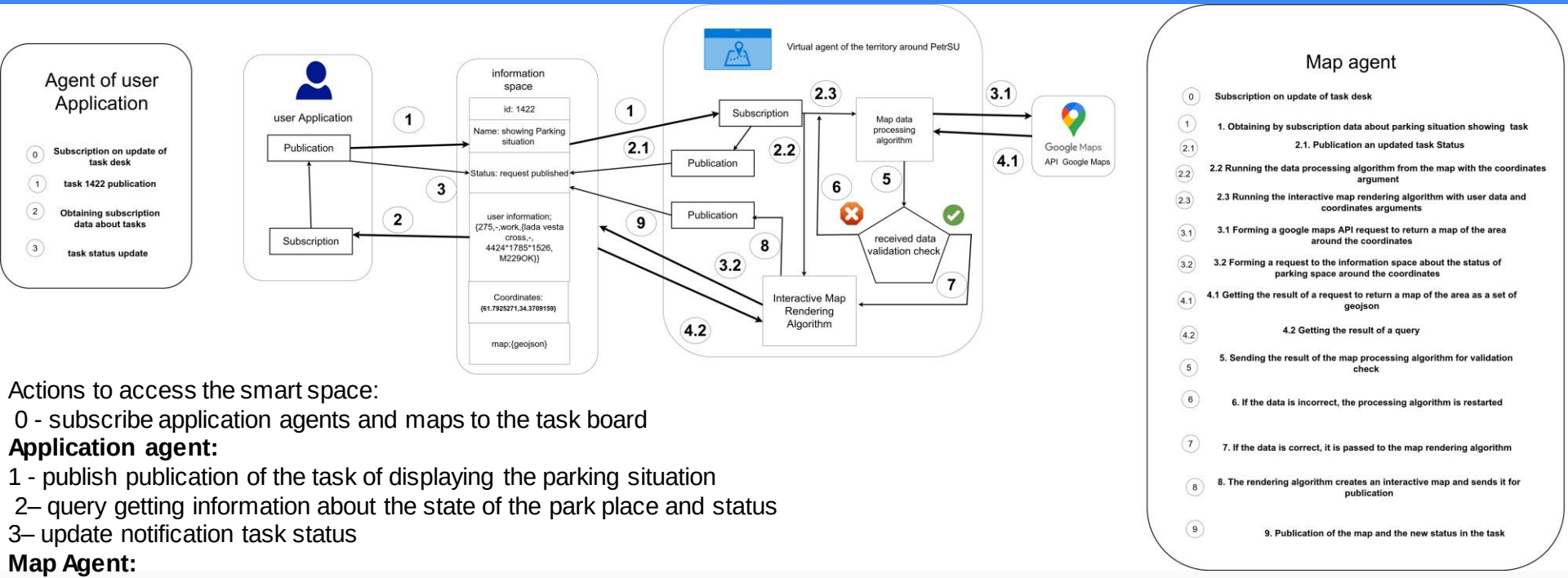
Delivery steps to the user of the parking response system



Data Analysis Activities in the Parking Response System

- Video stream data processing algorithm - a neural network trained to recognize cars, license plates of a car and a marked parking space, the recognition results are correlated and form the states of several parking spaces
- Algorithm for processing data from sensors - according to the data from the sensors, it is calculated whether the parking space is occupied, the result is returned as the state of the parking space
- Data processing algorithm - takes several states of a parking space and combines them into one, eliminating conflicts in the data
- Decision system - compares the new state of the parking space and the old one and decides whether it has changed, as well as the correctness of the perfect parking

Scenario for viewing the parking situation in a specific place



Actions to access the smart space:

0 - subscribe application agents and maps to the task board

Application agent:

1 - publish publication of the task of displaying the parking situation

2– query getting information about the state of the park place and status

3– update notification task status

Map Agent:

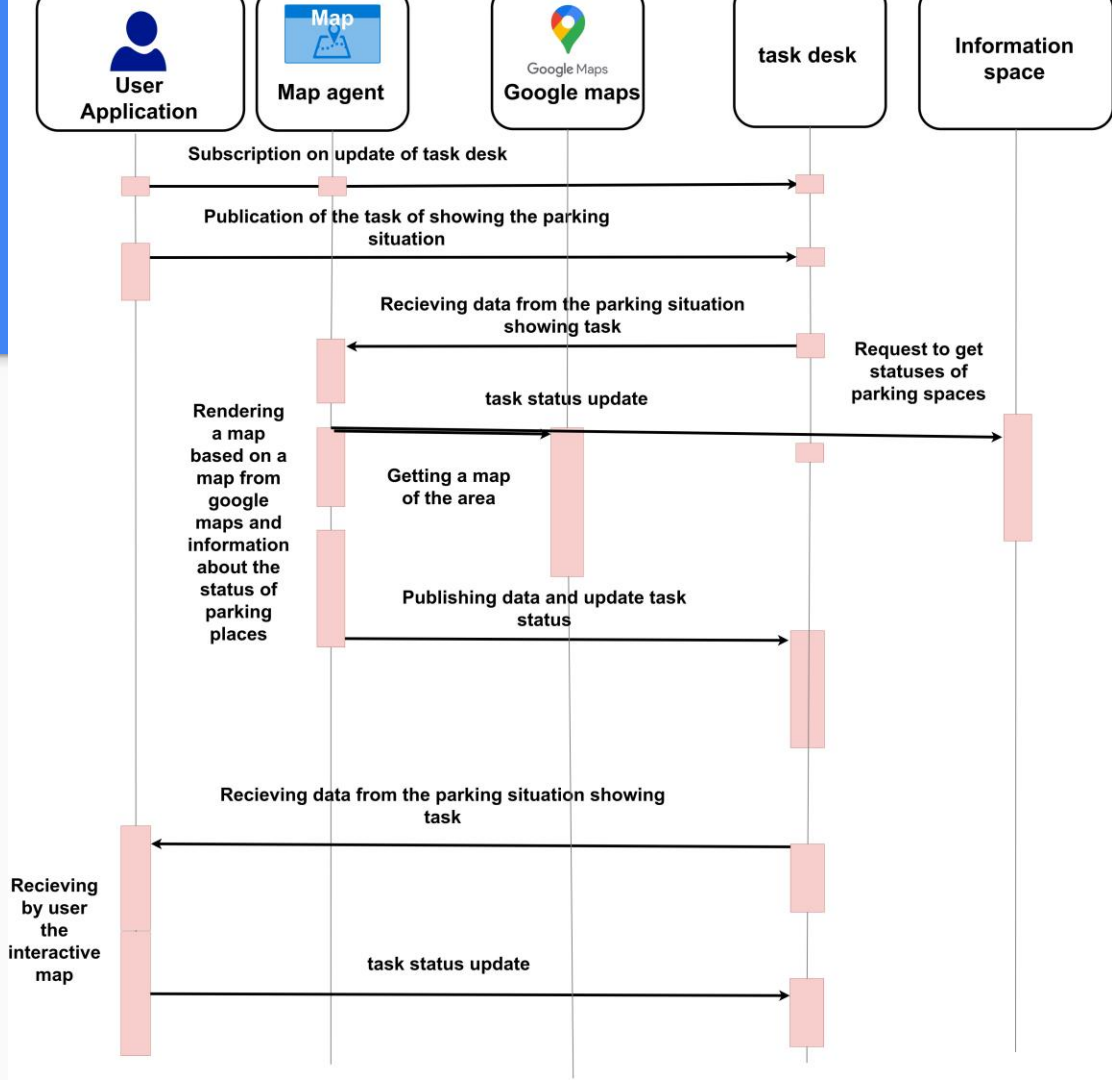
1 - Query getting information about the task

2.1 - update task status

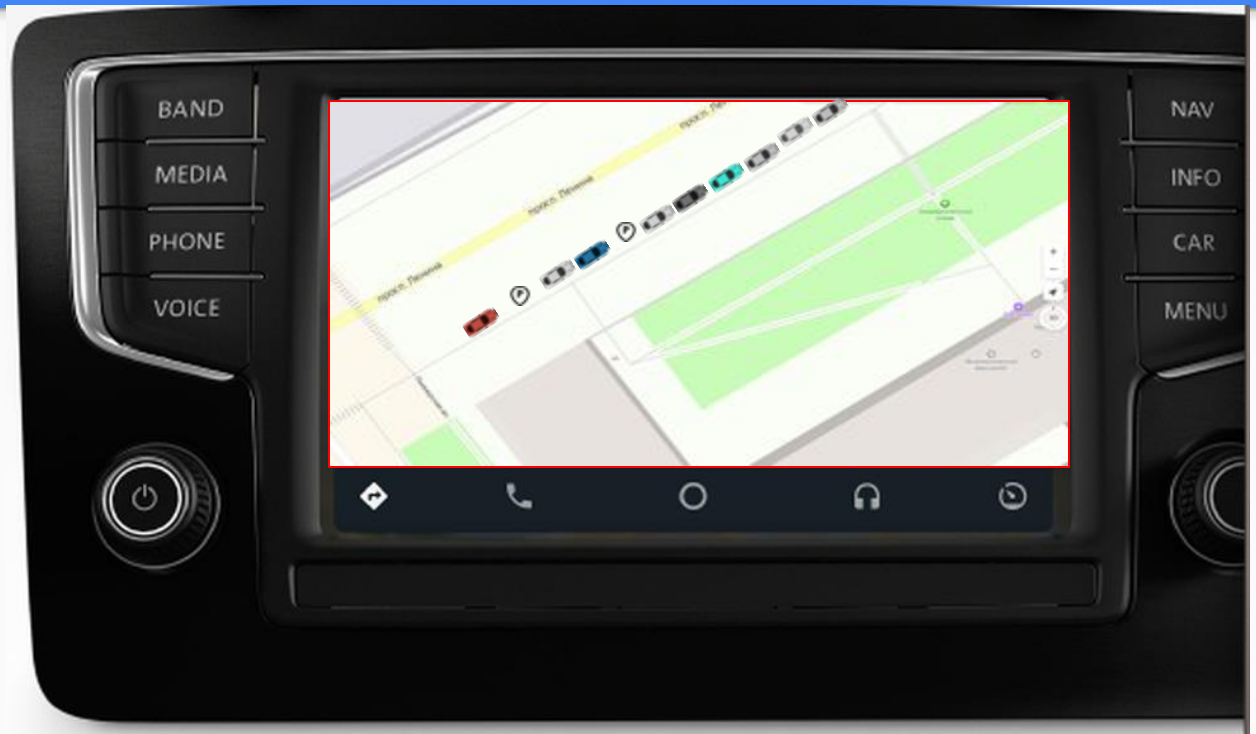
3.2 - Query to find the states of parking spaces at a distance of no more than 1 km from the coordinate

9 – publish task result: insert map , update status

Sequence Diagram for Parking Situation View Scenario



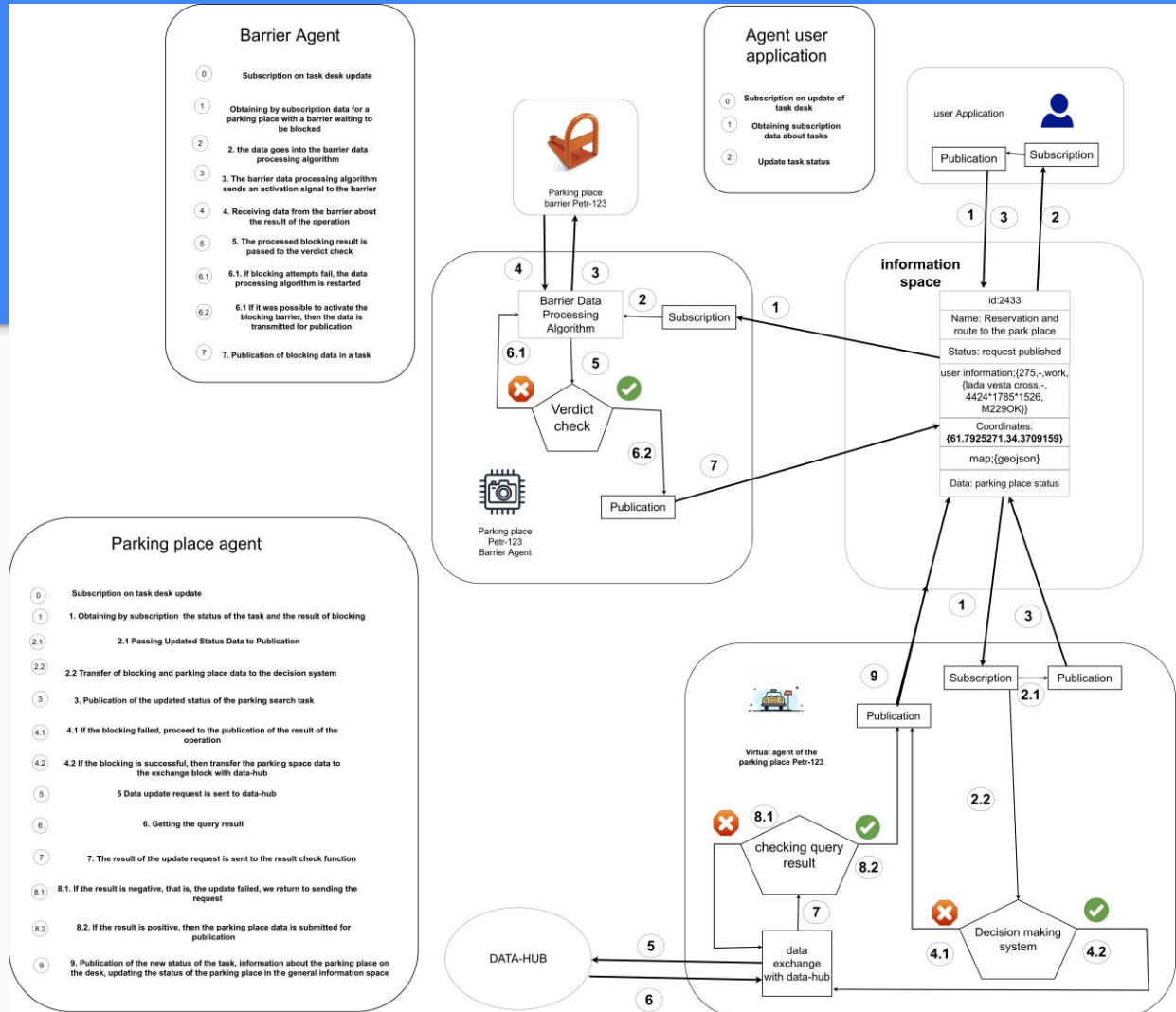
Actions to deliver to the user the output of the parking situation in the city



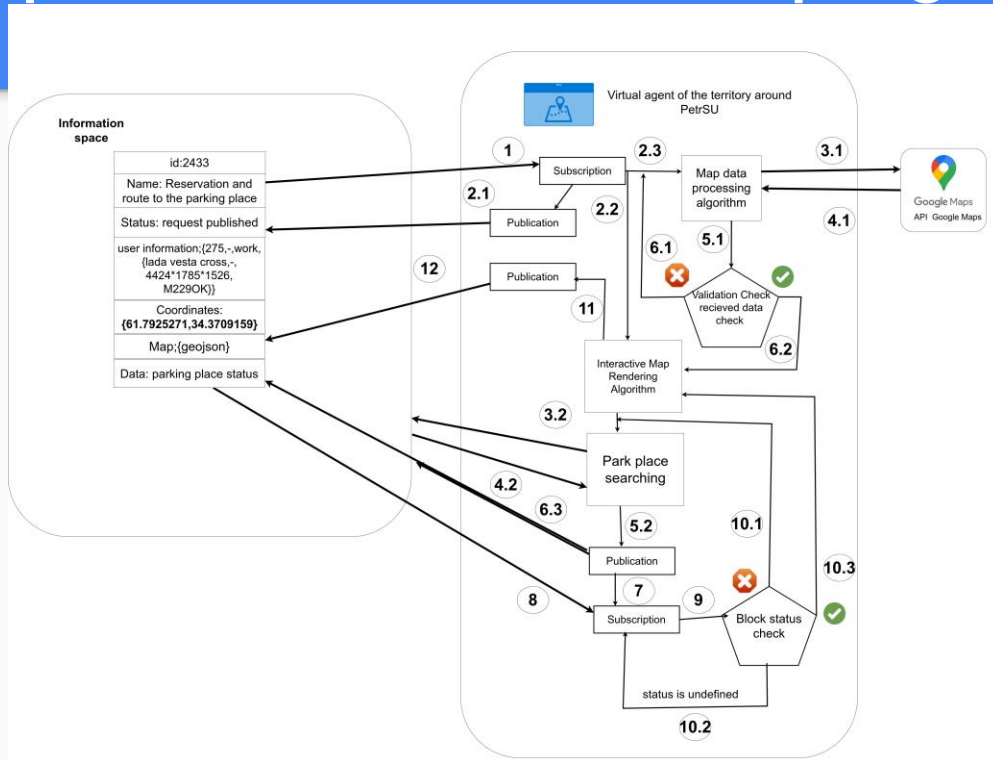
Actions for data analysis in the parking situation output system

- Algorithm for processing data from maps - API to google maps, requests maps, generates a list of geojsons
- Interactive map rendering algorithm - collects information about the current state of parking spaces and applies them to the resulting map

Scenario of searching and booking a parking space in a specific location - application, barrier and parking space agents



Scenario for searching and booking a parking space in a specific location - map agent



Map agent

0 Subscription on task desk update

1 1. Obtaining by subscription data about the task of searching and reserving a parking space

2.1 2.1. Publication an Updated task Status

2.2 2.2 Running the interactive map rendering algorithm with user data and coordinates arguments

2.3 2.3 Running the data processing algorithm from the map with the coordinates argument

3.1 3.1 Forming a google maps API request to return a map of the area around the coordinates

3.2 3.2 Forming a request to the information space about the status of parking place around the coordinates

4.1 4.1 Getting the result of a request to return a map of the area as a set of geojson

4.2 4.2 Getting the result of a query

5.1 5.1. Sending the result of the map processing algorithm for validation check

5.2 5.2 sending status update of parking place status data

6.1 6.1. If the data is incorrect, the processing algorithm is restarted

6.2 6.2. If the data is correct, it is passed to the map rendering algorithm

6.3 6.3 Publishing a new state of a parking place (conditionally occupied) in the task and in the general information space

7 7. moving to response to blocking of a parking space

8 8. Getting the status of a parking place and the status of a task as a result of blocking

9 9.checking blockng result

10.1 10.1 If the blocking result is negative, then another parking space is searched

10.2 10.2 If the blocking operation has not yet taken place, then continue reading the data

10.3 10.3 If the blocking was successful, then we pass the data about the parking place to the map rendering algorithm

11 11. The rendering algorithm creates an interactive map of the route to the parking place and sends it for publication

12 12. Publishing a map and a new status in a task

Steps for Accessing the Smart Space of the Search and Book a Parking place

0 - subscribe "application agents and maps to the task board

Application agent:

1 - publish "publication of the task of displaying the parking situation "

2– query getting information about the state of the park place, map and status

3– update the status of the search and reservation task

Map Agent:

1 - Query getting information about the task

2.1 - update task status

3.2 - Query to find the nearest free parking space suitable for the user

6.3 - publish update the state of the parking space is conditionally occupied

8 - Query getting information about the activation of a parking space

12– publish task result: insert map , update status

Barrier agent

1 - Query for getting information about the status of the task and the state of the parking space

7 - publication of the blocking result

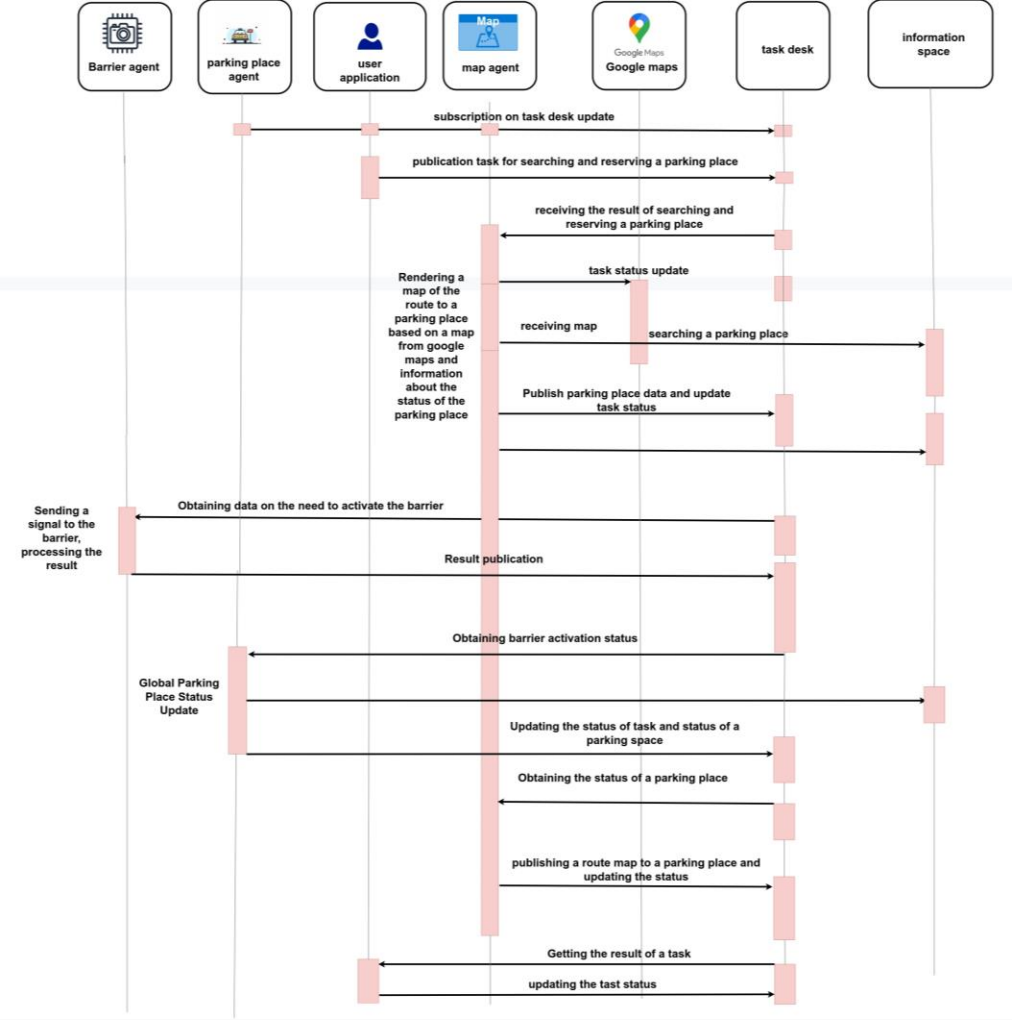
Parking space agent:

1 - Query getting information about the result of blocking

3 - publish update task status

9 - publish update parking space status and task status

Actions for accessing agents' smart space in a parking space search and reservation scenario



Scenario Data Analysis Steps

- Barrier data processing algorithm - sends a signal to the barrier, processes the result from the barrier
- Algorithm for processing data from maps - API to google maps, requests maps, generates a list of geojsons Interactive
- map rendering algorithm - collects information about the current state of parking spaces and applies them to the resulting map

Actions for delivery to the user of the service for searching and booking a parking space