Task

The lighting drone is an autonomous vandal-resistant flying contraption for night strolls in forest parks, which is able to find a person in the dark, follow him and light his way until he leaves the forest park. It can communicate with the drones of the same type and coalesce into a swarm, when several drones light the way for one person or vice versa, delegate lighting to another drone so that one drone illuminates the road for several people moving in the same direction. Describe the concept model for the Light Drone system as you see it. The description must include at least one diagram. Comment on significant entities and attributes. The detail and method of description is at the discretion of the performer.

1. Description of the system

The purpose of the "Lighting Drone" system development is the safety of people in poor lighting conditions in parks, forest parks, industrial zones and any other open areas that require additional lighting.

The Lighting Drone system is a combination of a certain number of flying drones and a server that centrally controls all drones.

1.1. Drones and charging stations

1.1.1. Drones

The flight of each drone is carried out autonomously according to GPS coordinates. It is assumed that the drone will fly only in open areas, not trespassing into premises, pipelines and other places where the GPS signal may be unavailable.

Before start, each drone in the system is given initial parameters:

- boundaries of the working area, where people should be escorted with path lighting (in the form of coordinates array);

- default flight altitude;

- default speed at which the drone moves around the area while not escorting people;

- charge level at which the drone should start flying to the charging station.

There may be different models of drones in the drone fleet with different characteristics, such as maximum speed and altitude, range of weather conditions suitable for flying, flashlight power, maximum flight time without recharging, etc.

1.1.2. Drone launch

After the launch, each drone that starts to carry out an escort mission must start moving towards the closest point from the territory boundaries array of coordinates. During a mission, the coordinates of the drone must be fixed and recorded once per second (another interval can be selected), except for the time spent at the charging station.

Each drone must turn on the built-in flashlight at the start of the session, i.e. upon entering the work area. The angle of the canopy must be set in the initial settings of the drone - for example, 30° to the horizontal plane.

1.1.3. Charging stations

Drone charging stations are located in the working area (for example, in a park). Charging stations are an optional element of the system, since not in every park it is possible to connect stations to circuit. The presence of charging stations reduces downtime of drones since drones do not need to fly to a base located outside the drone’s working area to recharge. Each station has several charging slots.

When the drone’s charge level reaches the threshold value specified in the settings (for example, 15%), the drone should start moving towards the nearest charging station that has available slots. The calculation of the nearest free station is carried out by the central control system located on the server.

1.1.4. Obstacle avoidance

It is also necessary to develop (or use ready-made) algorithms for avoiding obstacles in the way of the drone. All obstacles should be divided into known and unknown:

- To bypass known obstacles, a map of obstacles in the drones operational area must first be drawn up.

- To bypass unknown obstacles (other drones or objects not fixed on the obstacle map), a separate algorithm should be developed (or taken ready-made) for each drone model, which will depend on the number, type and characteristics of sensors and their location on the drone hull.

- At the first stage of system development, a ready-made algorithm for avoiding obstacles can be taken and implemented for drones with characteristics suitable for the algorithm.

1.1.5. Weather

The drone can operate only in certain weather conditions. For each drone model, the following parameters must be defined:

- minimum allowable air temperature,

- maximum allowable air temperature,

- maximum allowable wind speed,

- maximum allowable precipitation level.

All drones should only operate in the absence of extreme weather conditions such as natural disasters or heavy hail.

To control weather conditions it is necessary to record air temperature, wind speed and other parameters and compare them with the permissible operating conditions every 10 minutes (another time interval can be selected).

1.2. Groups of people and groups of drones, escorting people

1.2.1. The appearance of a person in the system

In this model, it is assumed that drones follow the people possessing a device with an enabled GPS module: a smartphone, tablet, GPS tracker, fitness bracelet, etc. When expanding the functionality, you can think about how to recognize a person without an active GPS device: for example, using infrared cameras that "see" objects in the dark and neural networks for recognizing a person's figure.

When a new GPS device appears on the drones operational area, information about the identification number of the device, its coordinates and time should be recorded in the database every second. Data on the movement of a device with an active GPS module should be recorded until the device is outside the operational area, i.e. outside the polygon formed by the given coordinates.

1.2.2. Definitions of "group of people" and "group of drones"

Each person can be accompanied by one drone or a group of drones, consisting of one master drone and (optionally) one or more slave drones. In addition, one drone or a group of drones can accompany several people walking in the same direction at a certain distance from each other. In the current paragraph and beyond, human refers to an active GPS device.

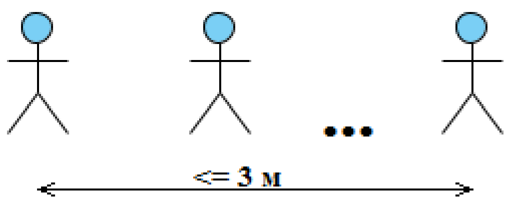
In summary, this model assumes that:

- One group of drones at one time (time interval = 1 second, because the data on the coordinates of the drone and the person are recorded in the database every second) accompanies one group of people;

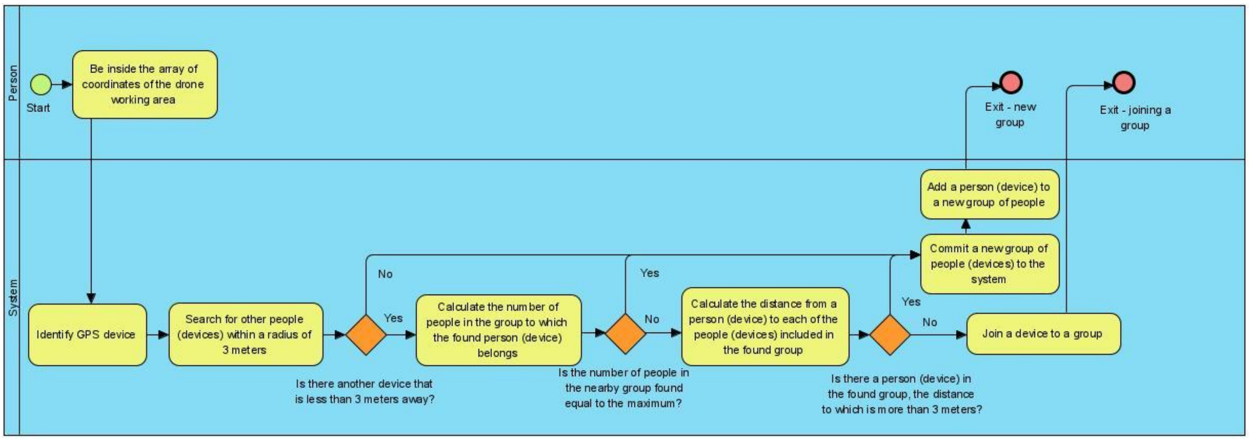
- One group of drones is considered: one master drone (mandatory) + one or more slave drones (optional).

1.2.3. Creation of a group of people

One group of people is considered to be one or more people (devices with active GPS) moving in the same direction at a distance from each other not exceeding the specified distance. In order for several people to be considered a group, every second of time between any two people in the group, the distance must not exceed 3 meters:



Therefore every second a check must be made for each person (GPS device) in the system, the algorithm of which is shown below:



The distance calculation around each device is carried out within a radius of 3 meters.

The maximum number of people in one group must be specified in the global setting in the code. The number of people in one group should not exceed the maximum.

1.2.4. Creating of a drone group

The composition of the drone group should be determined by the central drone control system. A drone group consists of 1 master drone from 0 to {N-1} slave drones, where N is the maximum possible number of drones in the group.

A new group of drones is assembled each time a new group of people is created, as follows:

1. The master drone is determined;

2. It is determined whether a new slave drone can be added to the group;

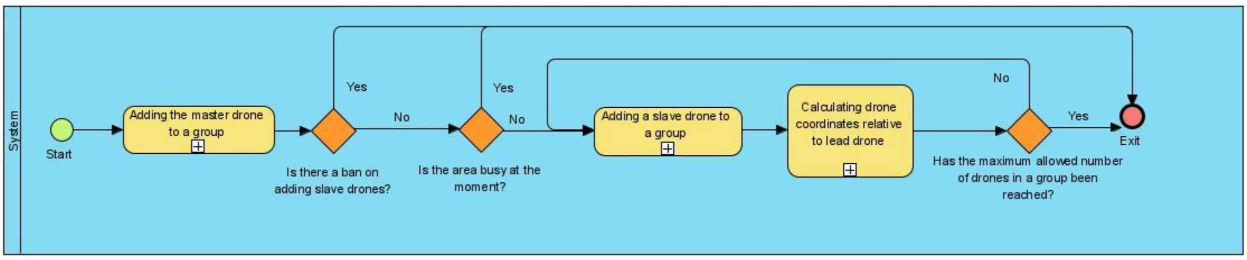
3. A slave drone is added if possible;

4. If adding a slave drone is not possible, then the group gathering is completed;

5. Items 2-4 are repeated until the group of drones is completely assembled.

At the same time a key role in the drone selection is played by the coordinates of a new group of people who should be followed, and the remaining percentage of the drone's charge.

The algorithm for creating a group of drones is shown below:

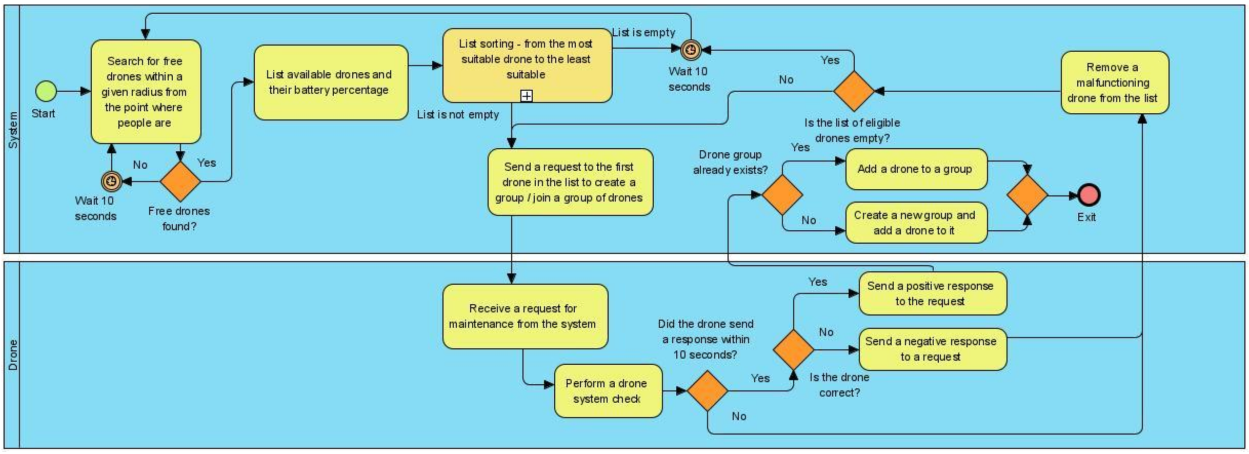


Slave drones should not be added to a drone group in the following cases:

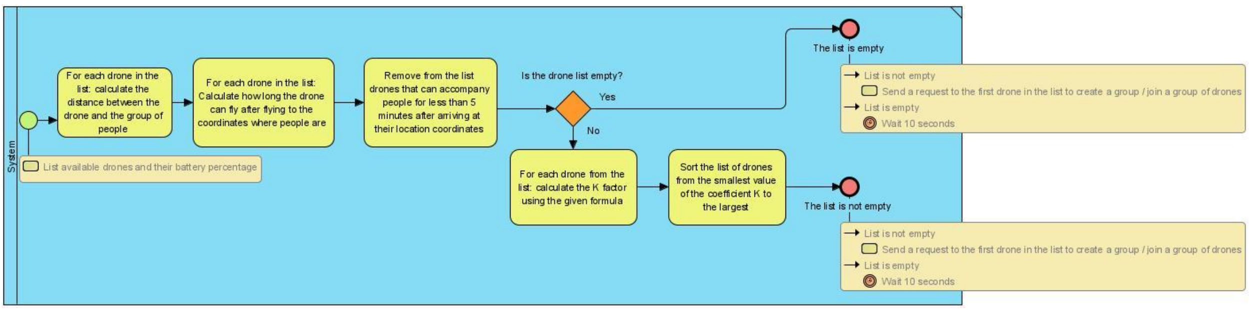
- If a large flow of people is expected in the next 30 minutes (the flow of people depends on the day of the week and time, as well as on the holiday schedule) and the number of drones in the working area is less than 1 drone per 1000 square meters (the value can be changed);

- Any time if the number of drones in the working area is less than 1 drone per 10,000 square meters (the value can be changed) the slave drones should not be added to the group.

The algorithm for adding a master and a slave drone is the same:



The algorithm for selecting the most suitable drone and compiling a list of drones in order from most suitable to least suitable is shown in the sub-process "Sort the list - from most suitable drone to least suitable":



For each drone, this algorithm calculates the coefficient K according to the formula K = S2 / P, where S is the distance between the drone and the group of people who need to be accompanied, P is the remaining charge of the drone in the range from 0 to 1.

This formula is used because the S and P parameters are the most important for choosing the right drone. The drone should fly to people in the shortest possible time (with the same speed, the distance between the drone and people should be considered the key by default), and the drone should accompany people for as long as possible, i.e. what matters is the percentage of charge that the drone has left. In this case, the distance to a person plays a more important role, so it should be squared. The smaller the K coefficient, the more suitable the drone is for the mission of escorting a group of people.

If the lead drone has a charge percentage close to the threshold, a new lead drone must be assigned. If there are slave drones in the group, then among them the drone with the longest remaining flight time without charging should be selected as the leader. If the group consists of only one drone, then at a certain time before reaching the threshold value (such time should be defined as a global variable), the system should start searching for a new drone according to the abovementioned algorithm. After the new lead drone reaches the coordinates of the group of people, the old lead drone should start moving towards the charging station.

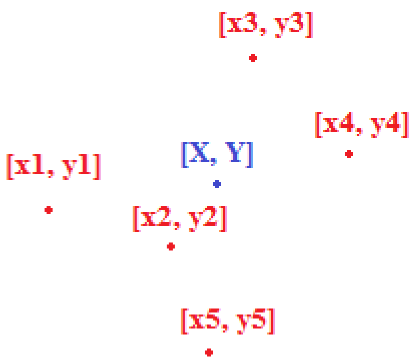
Note: in case when there are no charging stations in the operational area, or when there are few of them, it is necessary to assign a drone with a charge percentage calculated for a longer operational time to the mission to accompany a new group of people. This indicator depends on the distance to the drone base, or the distance between the charging stations. The indicator must also be set as a global variable in the system. For example, a condition can be set that the drone’s charge must allow it to operate for at least 45 minutes without recharging.

The maximum number of drones in one group must be specified in the global setting in the code. The number of drones in one group should not exceed the maximum.

1.2.5. Escorting people with drones

When carrying out an escort mission, a group of drones must fly at some distance from a group of people and have the same motion vector in the horizontal plane.

The coordinates of a group of people are the coordinates of the central point of the area occupied by a group of people. In practice, the coordinates of the center point can be calculated simply as the arithmetic average of the X and Y coordinates of all people in the group:



$$X = (x1 + x2 + ... + x5) / 5$$
$$Y = (y1 + y2 + ... + y5) / 5$$

The coordinates of the drone group should be the coordinates of the master drone.

Every second, the following actions should be performed in sequence:

- Calculation of the center point of a group of people.

- Calculation of the distance between a group of drones and a group of people accompanied by it.

- Calculation of the drone flight time to the coordinates where the group of people was 2 seconds ago, at a constant speed V, set by default in the drone settings.

- The movement of a group of drones towards the coordinates in which the group of people was 2 seconds ago, with speed (the speed depends on the parameter N - the time it takes the drone to reach the group of people at a speed V):

• If the predicted flight time to the coordinates is >= 2N minutes (seconds), then speed = 1.5V;

• If the predicted flight time to coordinates is >= N and < 2N minutes (seconds), then speed = 1.25V;

• If the predicted flight time to coordinates is >= 0.1N and < N minutes (seconds), then speed = 1V;

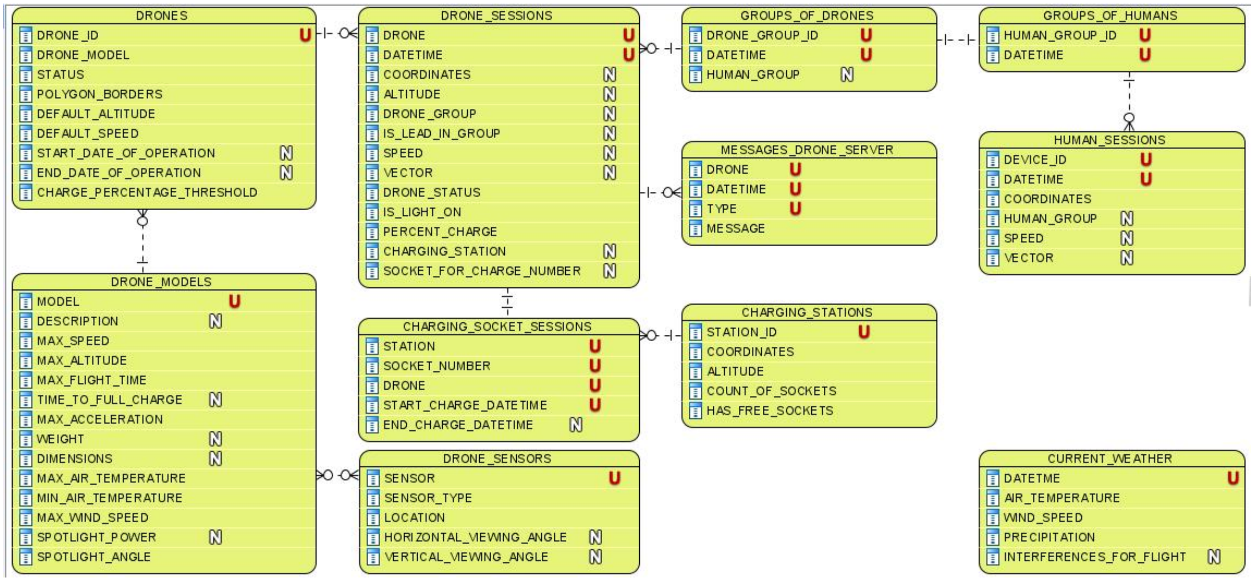
• If the predicted flight time to the coordinates is < 0.1N seconds, then the speed should decrease to the average speed of people over the last 3 seconds + 2 km/h.

- When a group of drones reaches the coordinates in which the group of people was 2 seconds ago, it is necessary to start flying to the coordinates in which the group of people was 1 second ago. The flight speed must be equal to the average speed of the group of people over the last 3 seconds.

It is necessary to provide for a smooth increase and decrease in speed: set a parameter that determines the allowable acceleration limit for the drone.

2. Conceptual data model

2.1. Conceptual model in the form of an ER diagram



2.2. Description of entities and attributes of the conceptual model

2.2.1. DRONE entity

Contains information about the drones in the drone park.

Logical key: DRONE_ID.

Entity attributes:

- DRONE_ID – unique drone identifier.

- DRONE_MODEL - drone model.

- STATUS - information about the status of the drone. Possible status options: "On a mission", "At the base", "Breakdown".

- POLYGON_BORDERS - an array of coordinates that outlines the drone operational area.

- DEFAULT_ALTITUDE - default altitude at which the drone should fly.

- DEFAULT_SPEED - default speed at which the drone should fly, heading towards the coordinates of the group of people that this drone should accompany. That is, this is the parameter V, described in paragraph 1.2.5.

- START_DATE_OF_OPERATION - date of the first operational flight of the drone.

- END_DATE_OF_OPERATION - date of the last operational flight of the drone. The field must be completed after the drone is decommissioned.

- CHARGE_PERCENTAGE_THRESHOLD - remaining drone charge threshold in percent, at which the drone should start moving towards the charging station or base for recharging.

2.2.2. DRONE_MODELS entity

Contains information about drone models present in the drone fleet.

Logical key: MODEL.

Entity attributes:

- MODEL – drone model name.

- DESCRIPTION - a brief description of the drone model.

- MAX_SPEED - maximum possible speed of the drone of this model.

- MAX_ALTITUDE - maximum possible altitude that a drone of this model can reach.

- MAX_FLIGHT_TIME - maximum flight time of the drone with the flashlight on and the active GPS module.

- TIME_TO_FULL_CHARGE - time to fully charge the drone from 0 to 100%.

- MAX_ACCELERATION - maximum possible acceleration of the drone.

- WEIGHT – drone weight.

- DIMENSIONS - drone dimensions.

- MAX_AIR_TEMPERATURE - maximum possible air temperature at which the drone can operate.

- MIN_AIR_TEMPERATURE - minimum possible air temperature at which the drone can operate.

- MAX_WIND_SPEED - maximum possible wind speed at which the drone can operate.

- SPOTLIGHT_POWER - drone flashlight power.

- SPOTLIGHT_ANGLE – tilt angle or range of possible tilt angles of the drone flashlight.

2.2.3. DRONE_SENSORS entity

Contains information about all possible sensor options located on the drone models present in the drone fleet.

Logical key: SENSOR.

Entity attributes:

- SENSOR – name of the sensor.

- SENSOR_TYPE – sensor type. Examples: "Laser", "Optical".

- LOCATION – location of the sensor on the hull, for example, "Left on the front".

- HORIZONTAL_VIEWING_ANGLE – horizontal viewing angle.

- VERTICAL_VIEWING_ANGLE – vertical viewing angle.

2.2.4. DRONE_SESSIONS entity

Contains information about drone sessions: coordinates and other information for each drone in every second of its flight over the operational area.

Logical key: DRONE + DATETIME.

Entity attributes:

- DRONE – drone identification number.

- DATETIME – current date and time in "YYYY-MM-DD HH24:MI:SS" format.

- COORDINATES - drone coordinates tracked by GPS.

- DRONE_GROUP - identifier of the drone group to which the drone belongs at the current time. If the drone does not escort people and therefore is not added to the drone group, then the value is left blank.

- IS_LEAD_IN_GROUP - if the drone is attached to a group of drones, then whether it is the leader in this group. Possible options: "Yes", "No". If the drone does not escort people, then the value remains empty.

- ALTITUDE - drone altitude tracked with the distance sensor pointing down.

- SPEED - speed at which the drone flew the last second. The field is optional and can be removed during the logical or physical design phases.

- VECTOR - a 3D vector showing the last second direction of the drone. The field is optional and can be removed during the logical or physical design phases.

- DRONE_STATUS - drone status. The options are: Free, On Mission, Charging, Heading to Base, Faulty.

- IS_LIGHT_ON - shows if the flashlight is on. Possible options: "Yes", "No".

- PERCENT_CHARGE - drone charge percentage.

- CHARGING_STATION - identification number of the charging station where the drone is located. The field is filled in when the drone is charging.

- SOCKET_FOR_CHARGE_NUMBER - number of the charging station socket where the drone is located. The field is filled in when the drone is charging.

2.2.5. GROUPS_OF_DRONES entity

Contains information about the drone groups formed to escort people every second.

Logical key: DRONE_GROUP_ID + DATETIME.

Entity attributes:

- DRONE_GROUP_ID – drone group ID.

- DATETIME – current date and time in "YYYY-MM-DD HH24:MI:SS" format.

- HUMAN_GROUP - a group of people that this group of drones escorts currently.

2.2.6. CHARGING_STATIONS entity

Contains information about charging stations for drones in the work area.

Logical key: STATION_ID

Entity attributes:

- STATION_ID – identification number of the charging station.

- COORDINATES – charging station coordinates.

- ALTITUDE - altitude at which the charging station is located.

- COUNT_OF_SOCKETS - number of sockets for charging drones at the station.

- HAS_FREE_SOCKETS - charging sockets availability, possible options: "Yes", "No".

2.2.7. CHARGING_SOCKET_SESSIONS entity

Contains information about drones at charging stations. The entity is optional: it was added to make it easier to access data about the presence of drones at charging stations.

Logical key: STATION + SOCKET_NUMBER + DRONE + START_CHARGE_DATETIME

Entity attributes:

- STATION – charging station identification number.

- SOCKET_NUMBER – number of the charging station's socket where the drone is currently charging.

- DRONE – drone identification number.

- START_CHARGE_DATETIME – start date and time of drone charging at the station.

- END_CHARGE_DATETIME - end date and time of drone charging at the station.

2.2.8. HUMAN_SESSIONS entity

Contains information about sessions of people (active GPS devices), i.e. about the coordinates of each person every second while the person is in drones operational area

Logical key: DEVICE_ID + DATETIME

Entity attributes:

- DEVICE_ID – identification number of a device with an active GPS module.

- DATETIME – current date and time in "YYYY-MM-DD HH24:MI:SS" format.

- COORDINATES - person coordinates tracked by GPS.

- HUMAN_GROUP - identifier of the people group accompanied by drones, to which this person is connected at the current second.

- SPEED - speed with which the person moved the last second. The field is optional and can be removed during the logical or physical design phases.

- VECTOR - a two-dimensional vector showing the movement direction of a person at the last second. The field is optional and can be removed during the logical or physical design phases.

2.2.9. GROUPS_OF_HUMANS entity

Contains information about groups of people that accompany groups of drones every second.

Logical key: HUMAN_GROUP_ID + DATETIME.

Entity attributes:

- HUMAN_GROUP_ID - ID of a group of people accompanied by drones.

- DATETIME – current date and time in "YYYY-MM-DD HH24:MI:SS" format.

2.2.10. MESSAGES_DRONE_SERVER entity

Contains information about all messages sent between the drones and the central control system.

Logical key: DRONE + DATETIME + TYPE

Entity attributes:

- DRONE – identification number of the drone sending/receiving the message.

- DATETIME – date and time of sending/receiving a message.

- TYPE – message type. If the drone receives a message, then type = "REQUEST", if it sends, then "RESPONSE".

- MESSAGE – message content.

2.2.11. CURRENT_WEATHER entity

Contains information about current weather conditions. In its direct form, an entity has no links to other entities. The information in the table should be used to compare the current weather conditions with the conditions under which each of the drone models can operate (this information is stored in DRONE_MODELS).

Logical Key: DATETIME

Entity attributes:

- DATETIME – date and time of weather conditions measurement.

- AIR_TEMPERATURE – air temperature.

- WIND_SPEED - wind speed.

- PRECIPITATION - the level of precipitation.

- INTERFERENCES_FOR_FLIGHT - interference that can hinder drone’s flight.

2.3. Description of relationships between entities of the conceptual model

2.3.1. DRONE_MODELS - DRONES

- Relationship type: 1 : N – one row in DRONE_MODELS corresponds to several rowa in DRONES.

- Attributes by which the connection is made:

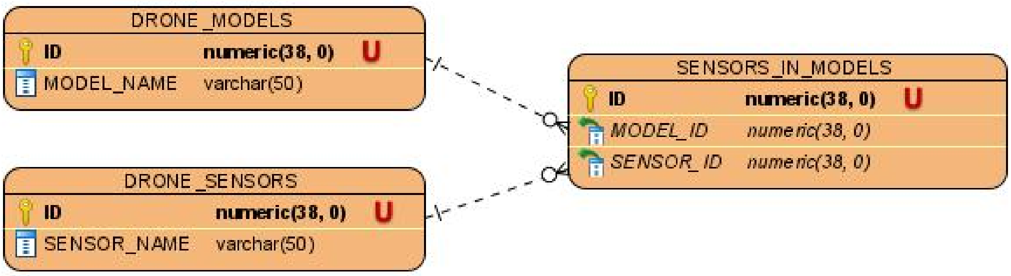
• DRONE_MODELS.MODEL = DRONES.DRONE_MODEL

- Description: Each drone in the drone park belongs to a specific drone model. There can be many drones of the same model.

2.3.2. DRONE_MODELS - DRONE_SENSORS

- Relationship type: N : M – several rows in DRONE_MODELS can correspond to several rows in DRONE_SENSORS.

- Attributes used for communication: Not shown in the conceptual model. The "many-to-many" relationship must be broken at the stage of creating a physical database model into 2 "one-to-many" relationships by adding a new entity "SENSORS_IN_MODELS":



- In the physical model, a relationship must exist between the attributes:

• SENSORS_IN_MODELS.MODEL_ID = DRONE_MODELS.ID

• SENSORS_IN_MODELS.SENSOR_ID = DRONE_SENSORS.ID

- Description: Each drone model can have multiple sensors. Each type of sensor can be present on several drone models.

2.3.3. DRONES-DRONE_SESSIONS

- Relationship type: 1 : N – one row in DRONES corresponds to several rows in DRONE_SESSIONS.

- Attributes by which the connection is made:

• DRONES.DRONE_ID = DRONE_SESSIONS.DRONE

- Description: Drone flight information is recorded in DRONE_SESSIONS every second. Information is not recorded when the drone is charging, except for the first second of charging. Also, information is not recorded while the drone is outside the operating area.

2.3.4. GROUPS_OF_DRONES - DRONE_SESSIONS

- Relationship type: 1 : N – one row in GROUPS_OF_DRONES corresponds to several rows in DRONE_SESSIONS.

- Attributes by which the connection is made:

• GROUPS_OF_DRONES.DRONE_GROUP_ID = DRONE_SESSIONS.DRONE_GROUP

• GROUPS_OF_DRONES.DATETIME = DRONE_SESSIONS.DATETIME

- Description: Multiple drones can be in the same group at the same time.

2.3.5. GROUPS_OF_HUMANS - HUMAN_SESSIONS

- Relationship type: 1 : N – one row in GROUPS_OF_HUMANS corresponds to several rows in HUMAN_SESSIONS.

- Attributes by which the connection is made:

• GROUPS_OF_HUMANS.HUMAN_GROUP_ID = HUMAN_SESSIONS.HUMAN_GROUP

• GROUPS_OF_HUMANS.DATETIME = HUMAN_SESSIONS.DATETIME

- Description: One group can include several people at the same time.

2.3.6. GROUPS_OF_DRONES - GROUPS_OF_HUMANS

- Relationship type: 1 : 1.

- Attributes by which the connection is made:

•GROUPS_OF_DRONES.HUMAN_GROUP=GROUPS_OF_HUMANS.HUMAN_GROUP_ID

• GROUPS_OF_DRONES.DATETIME = GROUPS_OF_HUMANS.DATETIME

- Description: One group of drones escorts one group of people at a time. Two given entities can be combined into one at subsequent design stages.

2.3.7. CHARGING_STATIONS - CHARGING_SOCKET_SESSIONS

- Relationship type: 1 : N – one row in CHARGING_STATIONS corresponds to several rows in CHARGING_SOCKET_SESSIONS.

- Attributes by which the connection is made:

• CHARGING_STATIONS.STATION_ID = CHARGING_SOCKET_SESSIONS.STATION

- Description: Each charging station has many drone charging sessions.

2.3.8. DRONE_SESSIONS - CHARGING_SOCKET_SESSIONS

- Relationship type: 1 : 1.

- Attributes by which the connection is made:

• DRONE_SESSIONS.DRONE = CHARGING_SOCKET_SESSIONS.DRONE

• DRONE_SESSIONS.CHARGING_STATION = CHARGING_SOCKET_SESSIONS.
STATION

•DRONE_SESSIONS.SOCKET_FOR_CHARGE_NUMBER=CHARGING_SOCKET_SESSIONS.SOCKET_NUMBER

•DRONE_SESSIONS.DATETIME=CHARGING_SOCKET_SESSIONS.START_CHARGE_DATETIME

- Description: Entities are connected by the time the drone arrives for charging. The 1:1 relationship type is because no data should be stored in DRONE_SESSIONS during the time the drone is charging, other than the first second of charging. This is done so that duplicate rows are not created. Communication between entities is carried out, including in the first second of drone charging. Two given entities can be combined into one at subsequent design stages.

2.3.9. DRONE_SESSIONS - MESSAGES_DRONE_SERVER

- Relationship type: 1 : N – one row in DRONE_SESSIONS corresponds to several rows in MESSAGES_DRONE_SERVER.

- Attributes by which the connection is made:

• DRONE_SESSIONS.DRONE = MESSAGES_DRONE_SERVER.DRONE

• DRONE_SESSIONS.DATETIME = MESSAGES_DRONE_SERVER. DATETIME

- Description: during the drone's flight a message can be sent from both the drone to the server and the server to the drone at the same time. It is also necessary to provide that the data is not written to DRONE_SESSIONS during charging, therefore a message received by the drone at this moment should be recorded by the drone in the first second after leaving the charging station.

3. Stages of development

1. Implementation of the flight of drones of one model, implementation of device identification with an active GPS module, implementation of escort of one person (GPS device) by one drone, search and implementation of a ready-made algorithm for avoiding obstacles.

2. Implementation of the possibility of escorting a group of people by a drone.

3. Implementation of the possibility of creating a group of drones to escort a group of people.

4. Adding charging stations.

5. Implementation of the flight of various models of drones, analysis of obstacle avoidance algorithms, development and implementation of improved obstacle avoidance algorithms for each drone model.

6. Adding the ability to identify a person without an active GPS device.

Stage 1 contains the basic tasks, without the implementation of which the system cannot function. In conditions of tight deadlines, it is necessary to implement this stage.

Also, with a short time frame, other models of the system can be considered and proposed: for example, the creation of a "grid" of drones in an unlit park with a flashlight turned down, i.e. each drone must always be in strictly specified coordinates. The grid should be placed in such a way as to illuminate the maximum number of paths that people move along. With a small area of the park, it is possible to offer a model in which drones "hang" along the perimeter of the park with a flashlight turned on, directed towards the park at a certain angle.

Time spent on the test task: approx. 20 hours.