ISTANBUL TECHNICAL UNIVERSITY

SOFTWARE ENGINEERING

ASSIGNMENT 4

PROJECT TITLE	Package Deliverer Drones
REPORT NAME	Design Specification
TEAM NAME	Team Ratchet
GROUP NUMBER	**6**

TEAM MEMBERS	
NAME	**STUDENT NUMBER**
Mehmet Taner ÜNAL	150130702
Cem Yusuf AYDOĞDU	150120251
Ozan ÖZYEĞEN	150120115

CONTENTS

DESIGN DPECIFICATION

1. INTRODUCTION

This section introduces the design specification document for the Package Deliverer Drones. It provides the purpose of the system and the organization of contents in this document.

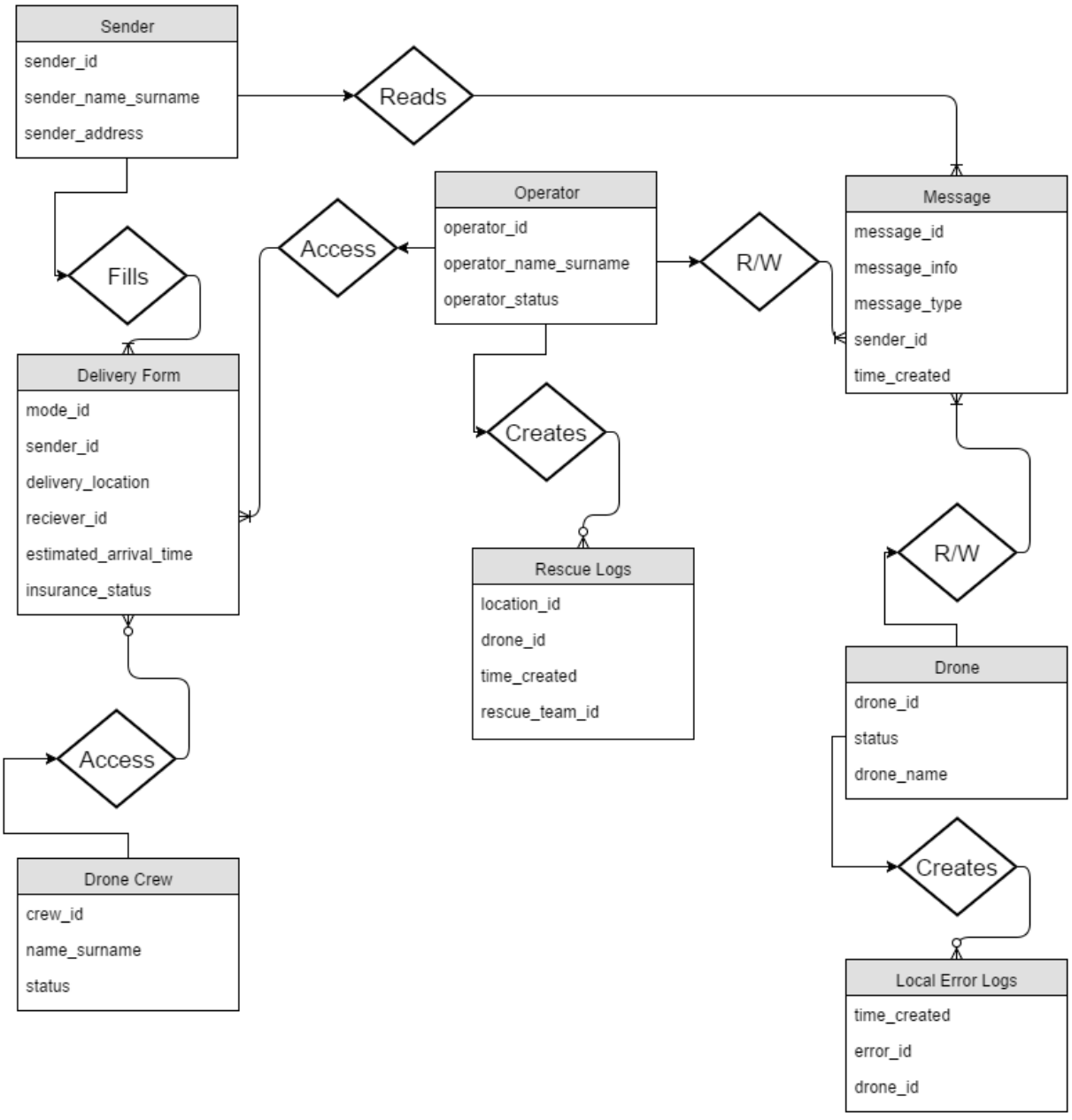
The drones take the package from the sender and bring it to the receiver. In addition, all the activity of the drones can be monitored from software that is used by the Control Operator and Drone Crew. This system helps greatly to the cargo companies, reduces the manpower needed to deliver packages and makes easier to send and receive packages.

This design specification document describes the models specified for this Package Deliverer Drones System. It brings detailed information about the design of the system, such as data structures, components and the user interface. The development of this program is involved in the “Software Engineering” course. The objectives are to be able to develop a product by managing its several phases by following some methods and models. It enables us being familiar with the way products are developed in a professional context.

2. DATA MODEL

2.1 General Data Model

This entity relationship diagram is based on the conceptual model which is given in the requirements specification document. Thus, they contain many similarities. Both of the models have aimed to Show how the system Works by using entities and relationships. However, since the conceptual model is an early model it is not as specific as the entity relationship diagram. The entity relationship diagram is created by considering how the data will be stored in the system.



2.2 Important Data Considerations

In this part, the information about the other data formats is given. The system gets the weather information in order to decide whether the drones should fly or not. To do this, our system uses the API of openweathermap.org. System gets the weather information by making JSON calls. Example JSON calls and responds are given below:

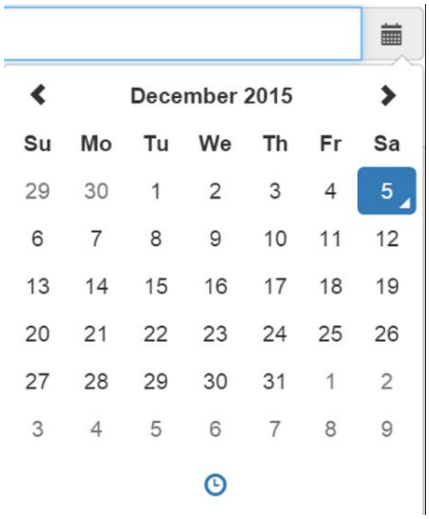
JSON

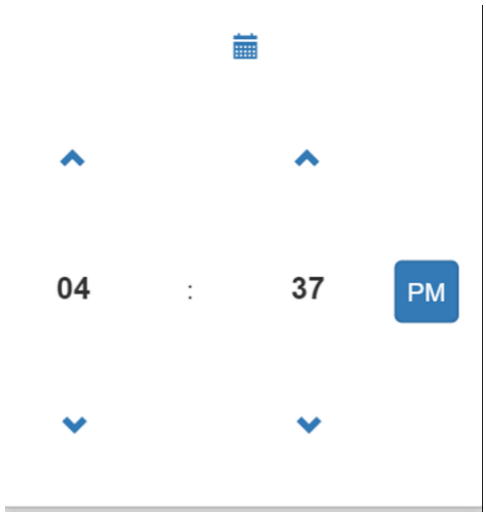
Example of API respond:


```
{"coord":
{"lon":145.77,"lat":-16.92},
"weather":[{"id":803,"main":"Clouds","description":"broken clouds","icon":"04n"}],
"base":"cmc stations",
"main":{"temp":293.25,"pressure":1019,"humidity":83,"temp_min":289.82,"temp_max":295.3
7},
"wind":{"speed":5.1,"deg":150},
"clouds":{"all":75},
"rain":{"3h":3},
"dt":1435658272,
"sys":{"type":1,"id":8166,"message":0.0166,"country":"AU","sunrise":1435610796,"sunset
":1435650870},
"id":2172797,
"name":"Cairns",
"cod":200}
```

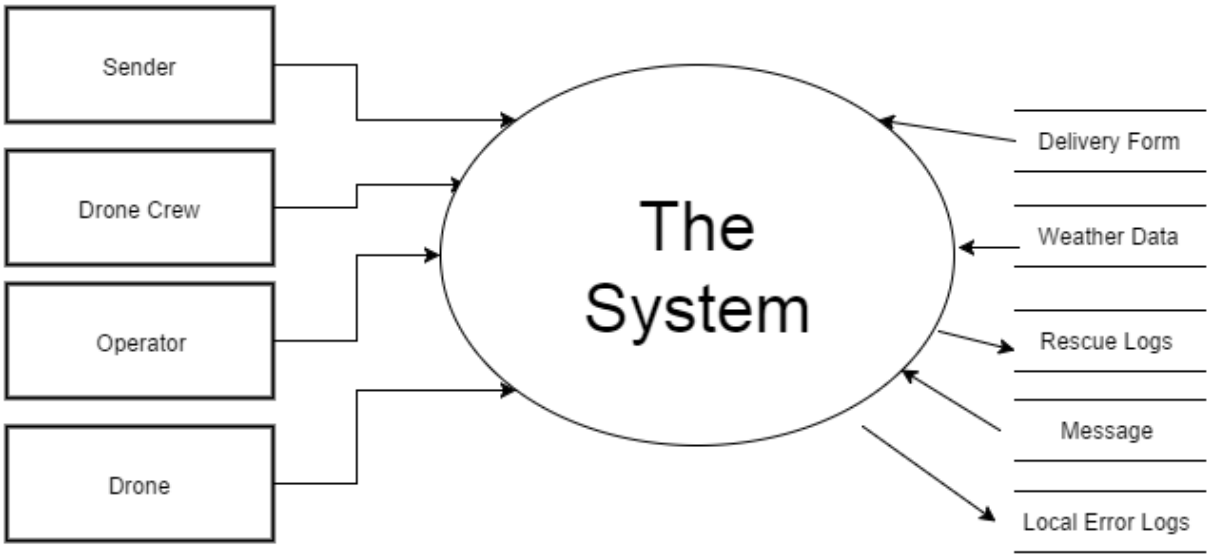
Another important data consideration for the drone delivery system is delivery form page. The sender needs to fill the delivery form on a browser. Hence, a delivery form web page is created. PHP, Ajax and Bootstrap are used on the delivery form page. After the user click send bottom at the page, it sends the information to the system in JSON form. Then the system receives and processes the delivery request.

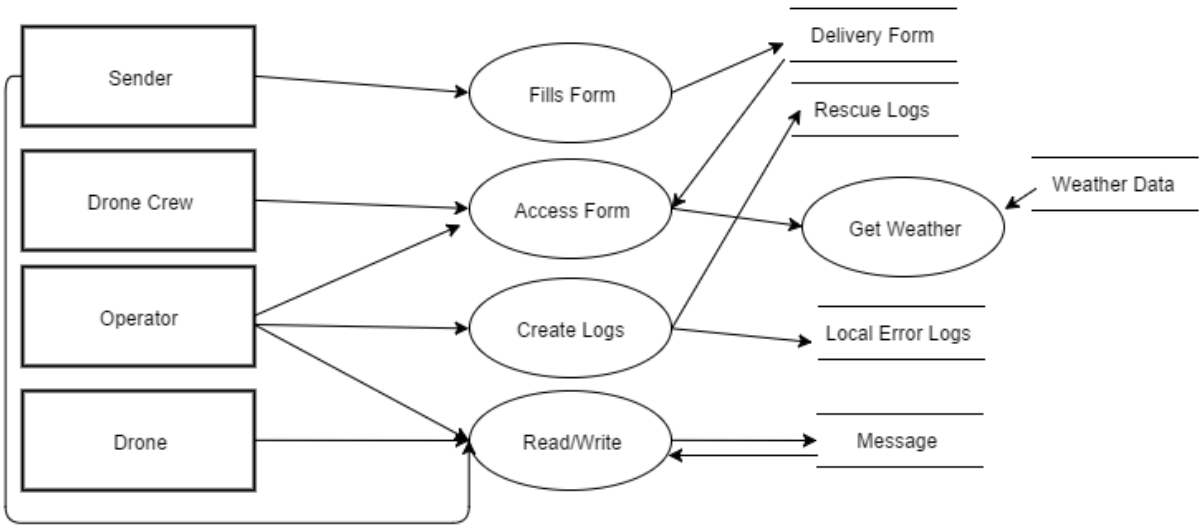
Screenshots of the Bootstrap Datetimepicker which is used in the delivery form is given below:





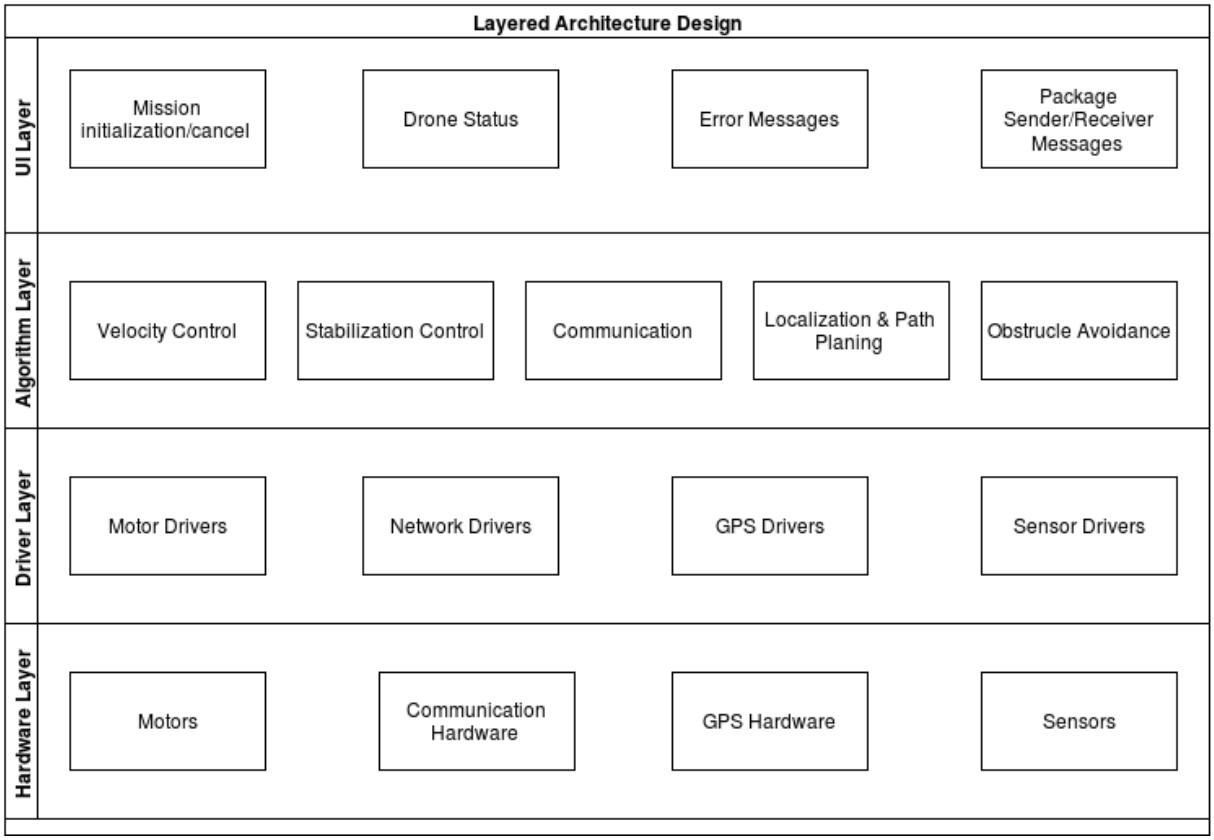
2.3 Data Flow


Sender
Drone Crew
Operator
Drone
The System
Delivery Form
Weather Data
Rescue Logs
Message
Local Error Logs

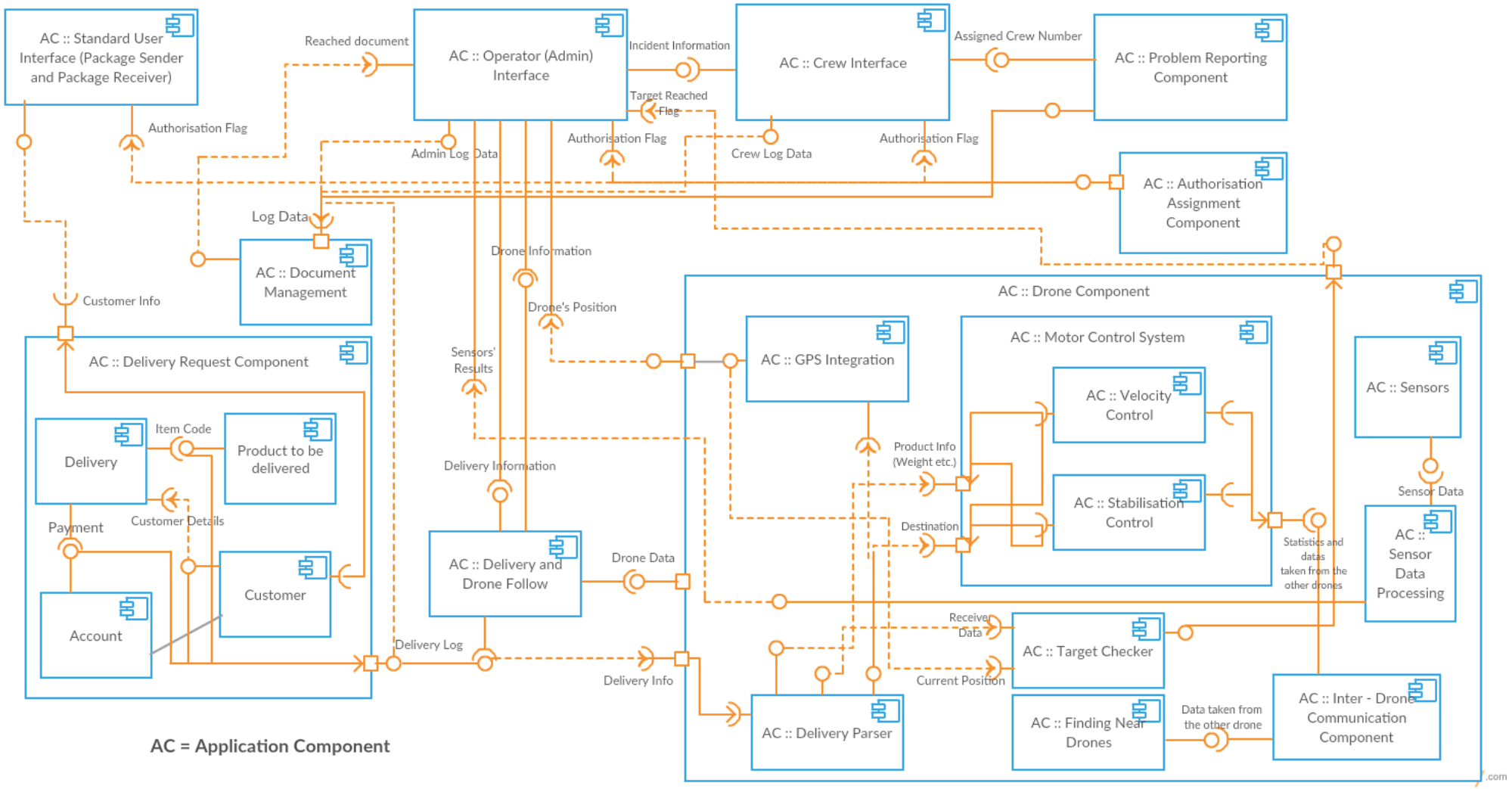

Sender
Drone Crew
Operator
Drone
Fills Form
Access Form
Create Logs
Read/Write
Delivery Form
Rescue Logs
Get Weather
Weather Data
Local Error Logs
Message

3. SOFTWARE MODEL

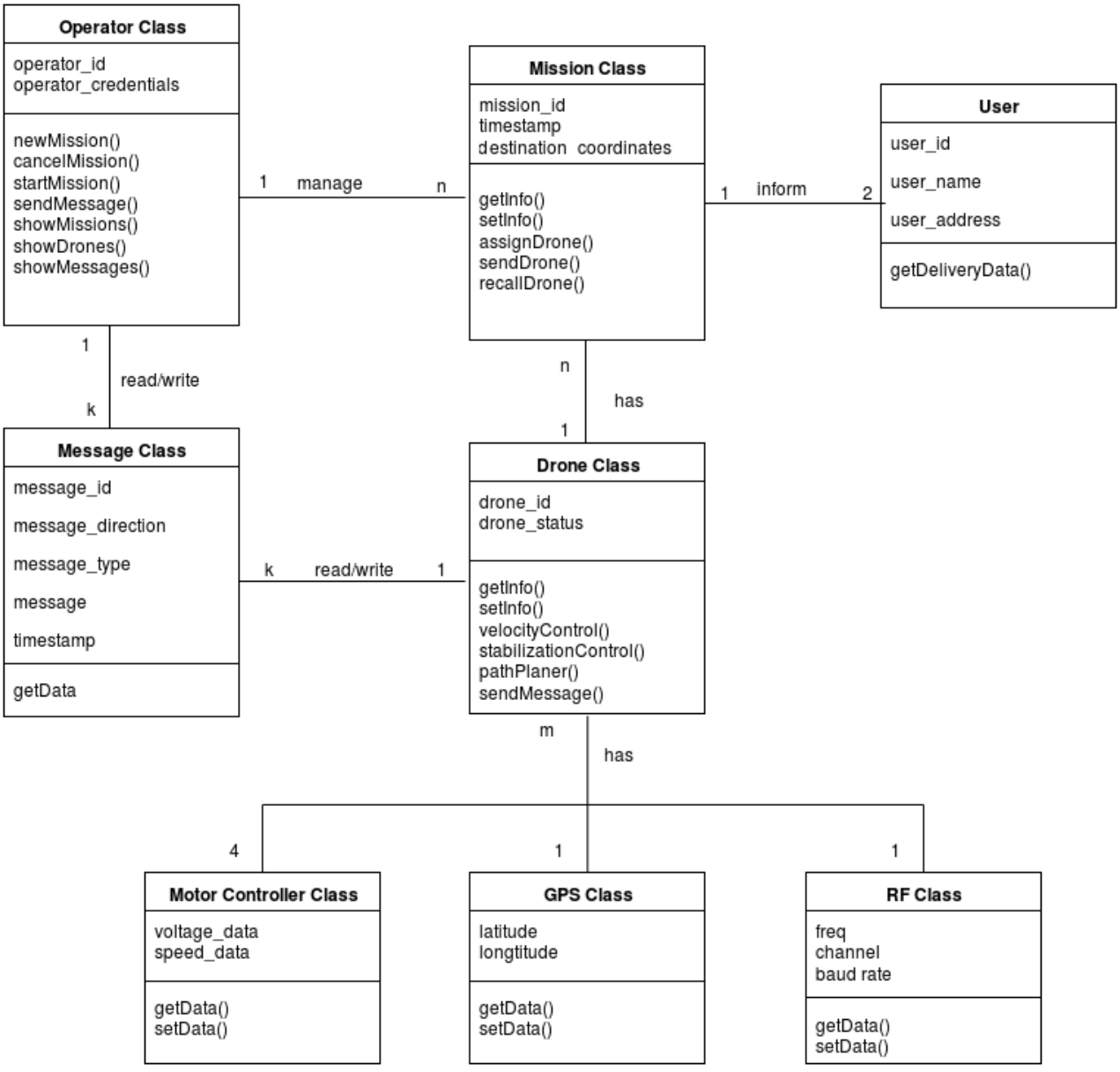
3.1 System Architecture



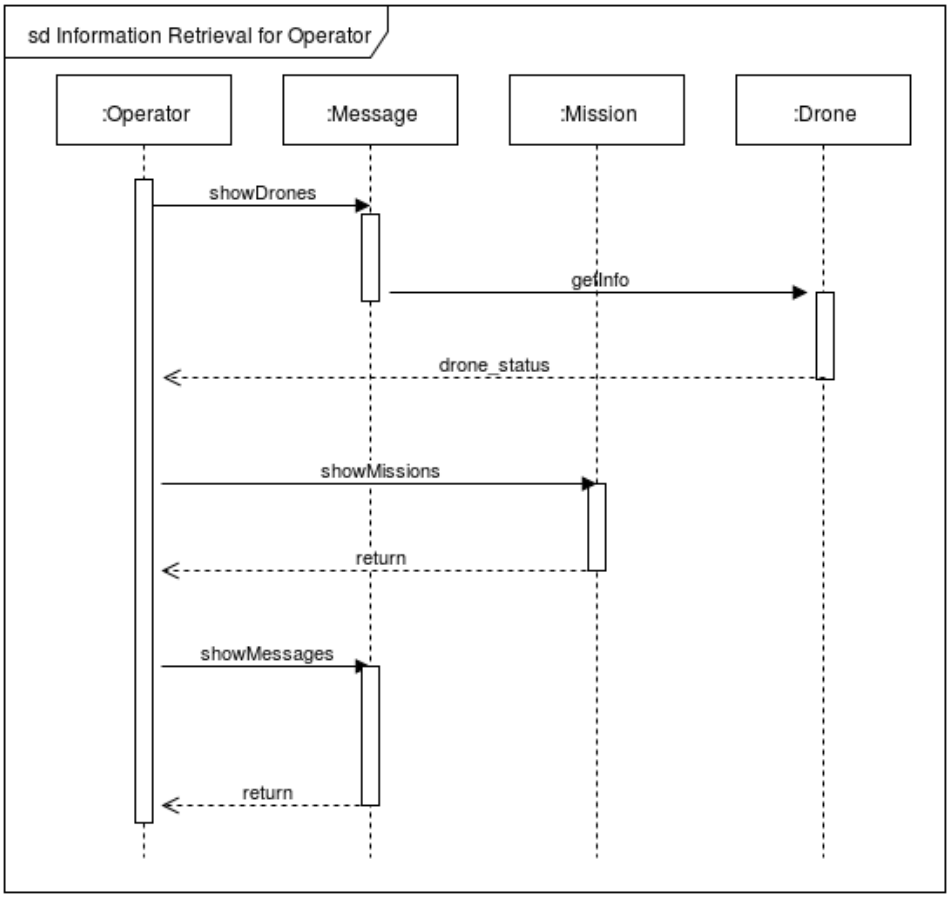
3.2 Component Diagram

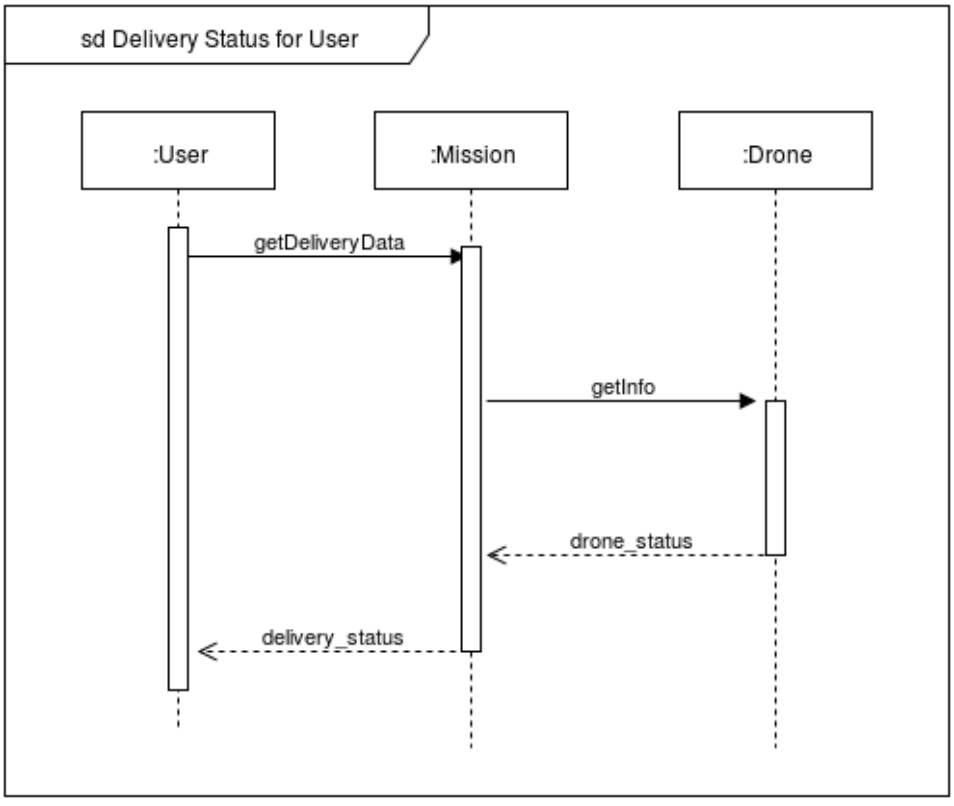


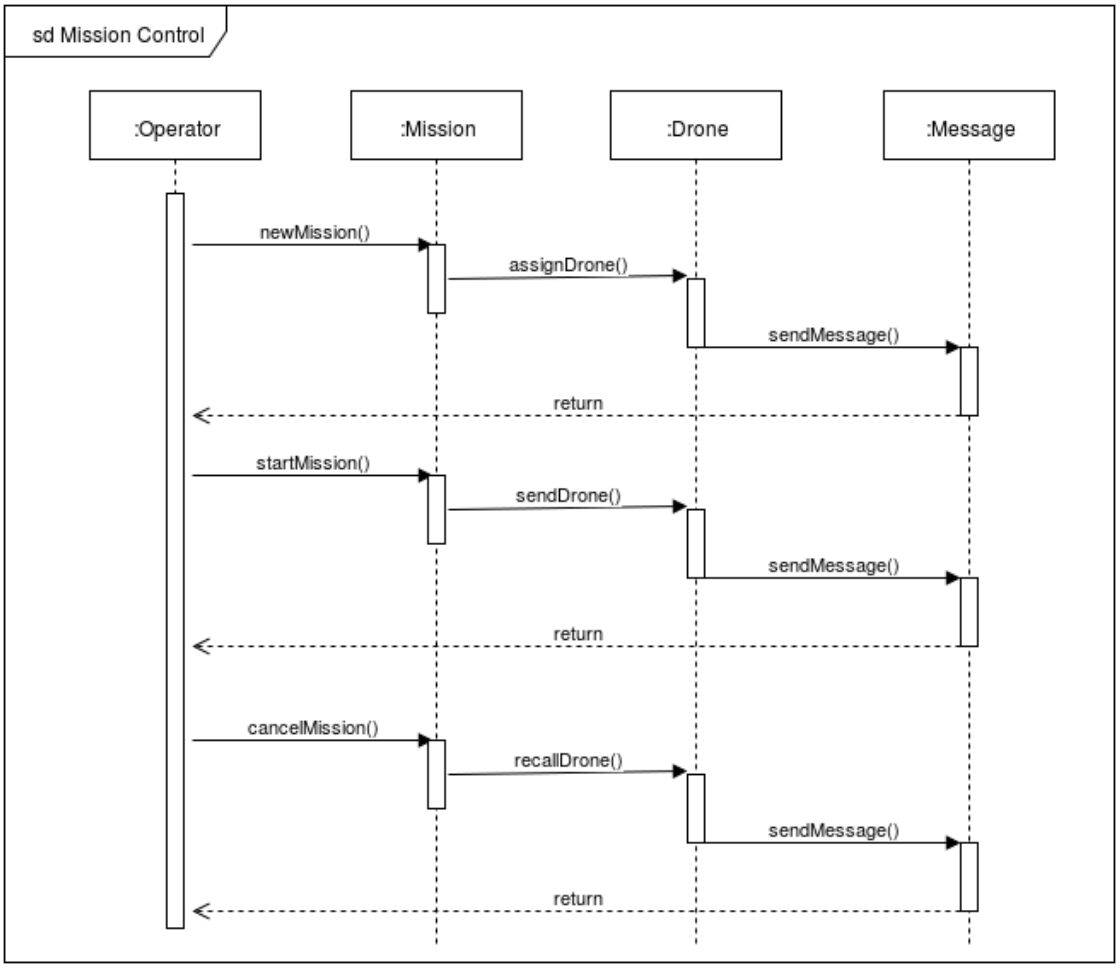
3.3 Class Diagrams

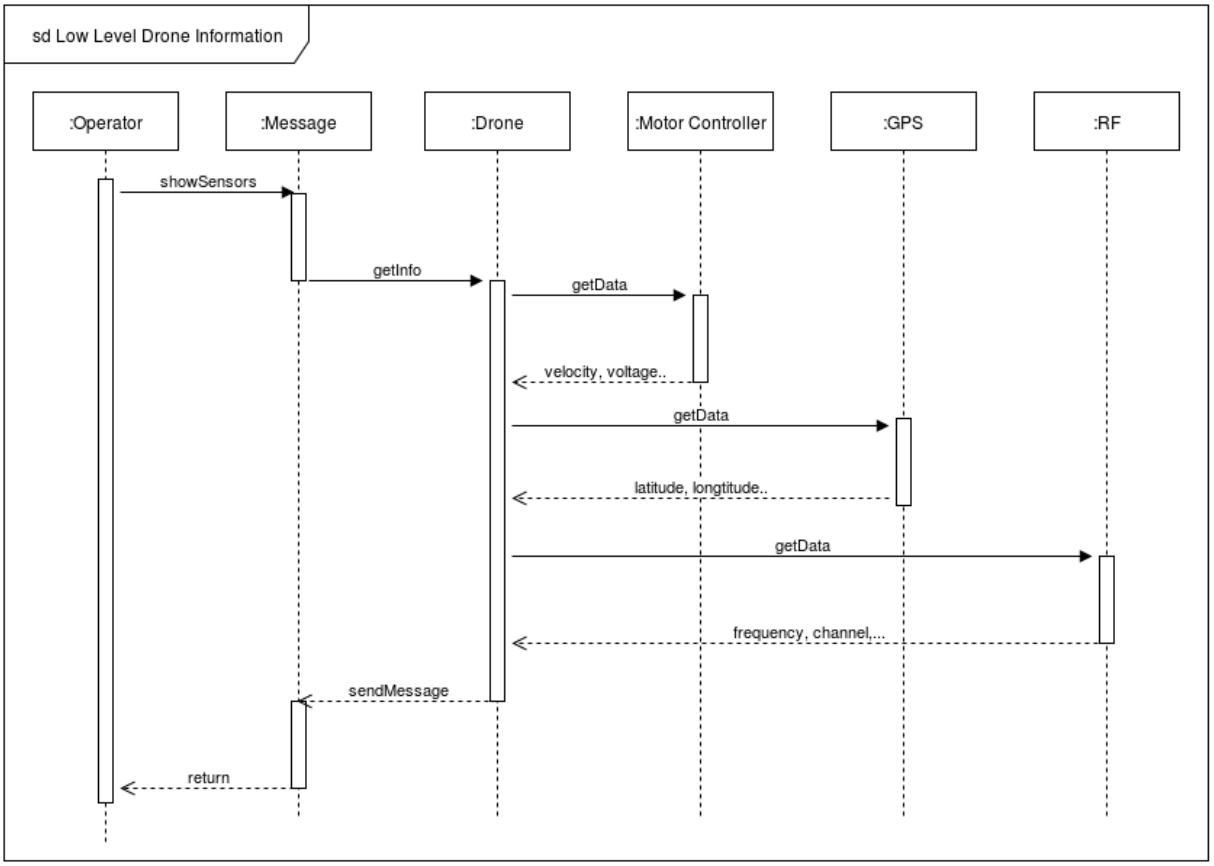


3.4 Sequence Diagrams

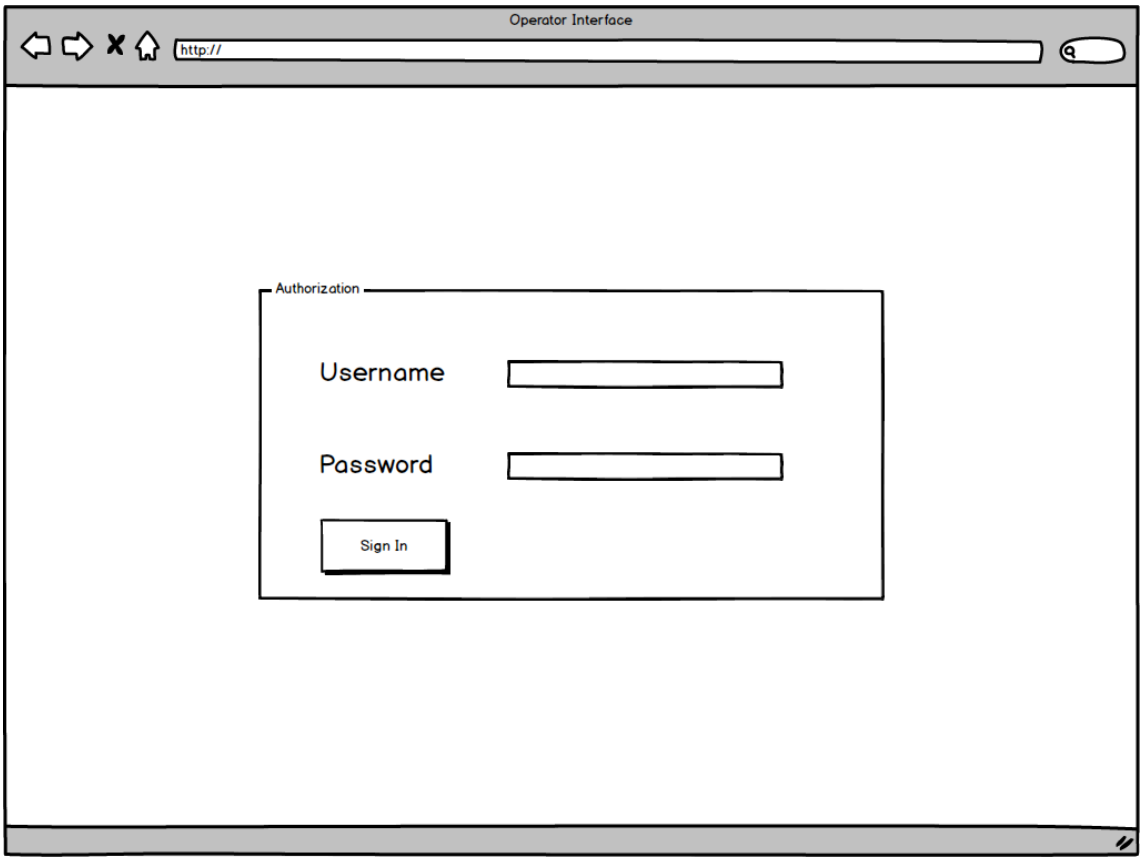


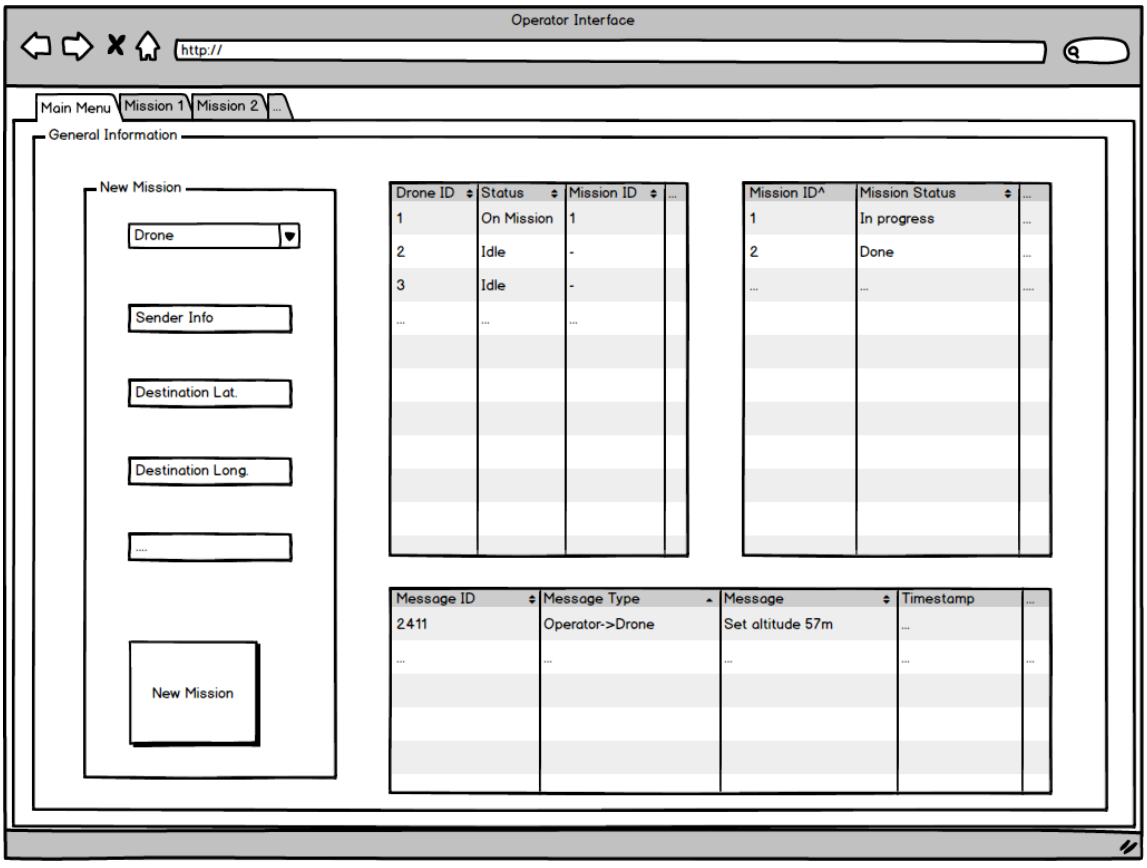


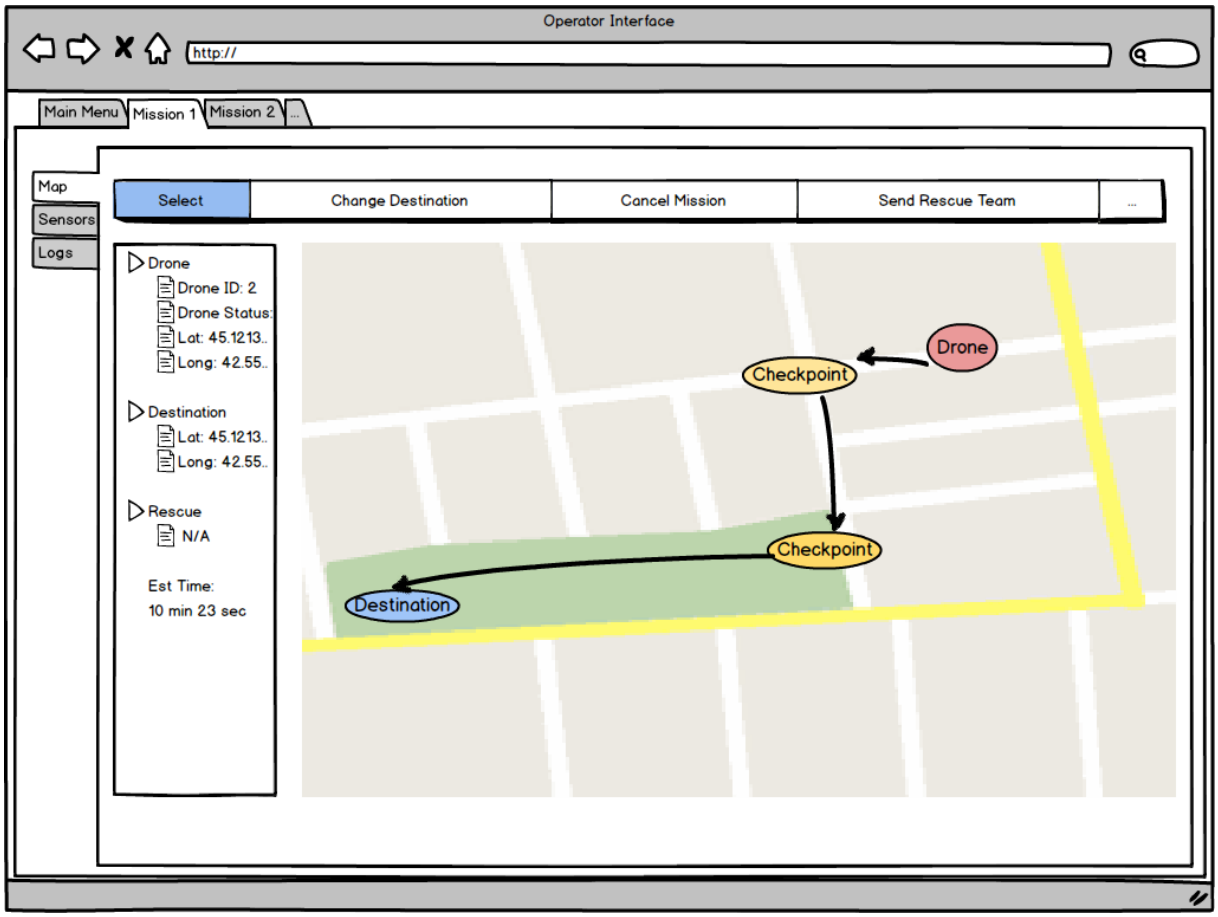

sd Mission Control
:Operator
:Mission
:Drone
:Message
newMission()
assignDrone()
sendMessage()
return
startMission()
sendDrone()
sendMessage()
return
cancelMission()
recallDrone()
sendMessage()
return

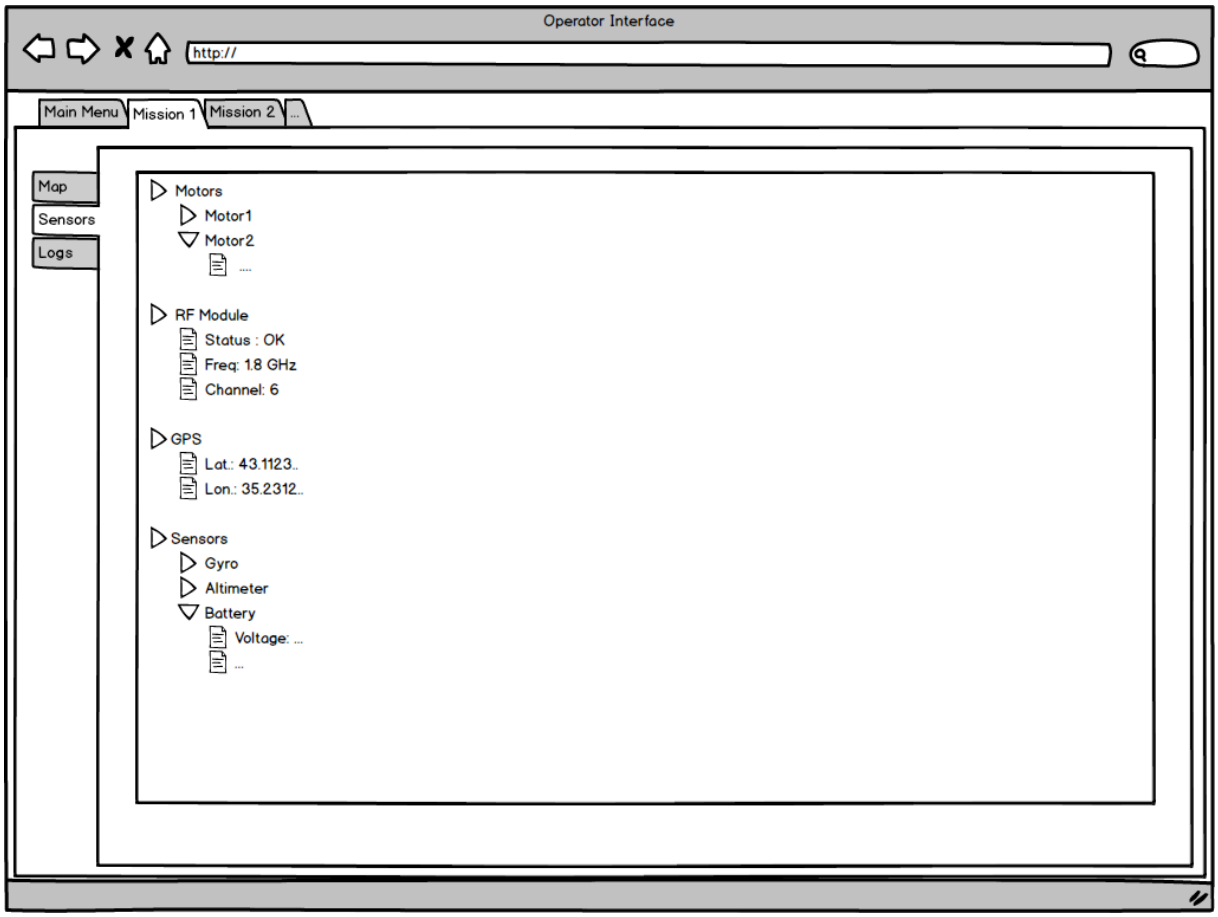

sd Low Level Drone Information
:Operator
:Message
:Drone
:Motor Controller
:GPS
:RF
showSensors
getInfo
getData
velocity, voltage..
getData
latitude, longtitude..
getData
frequency, channel,...
sendMessage
return

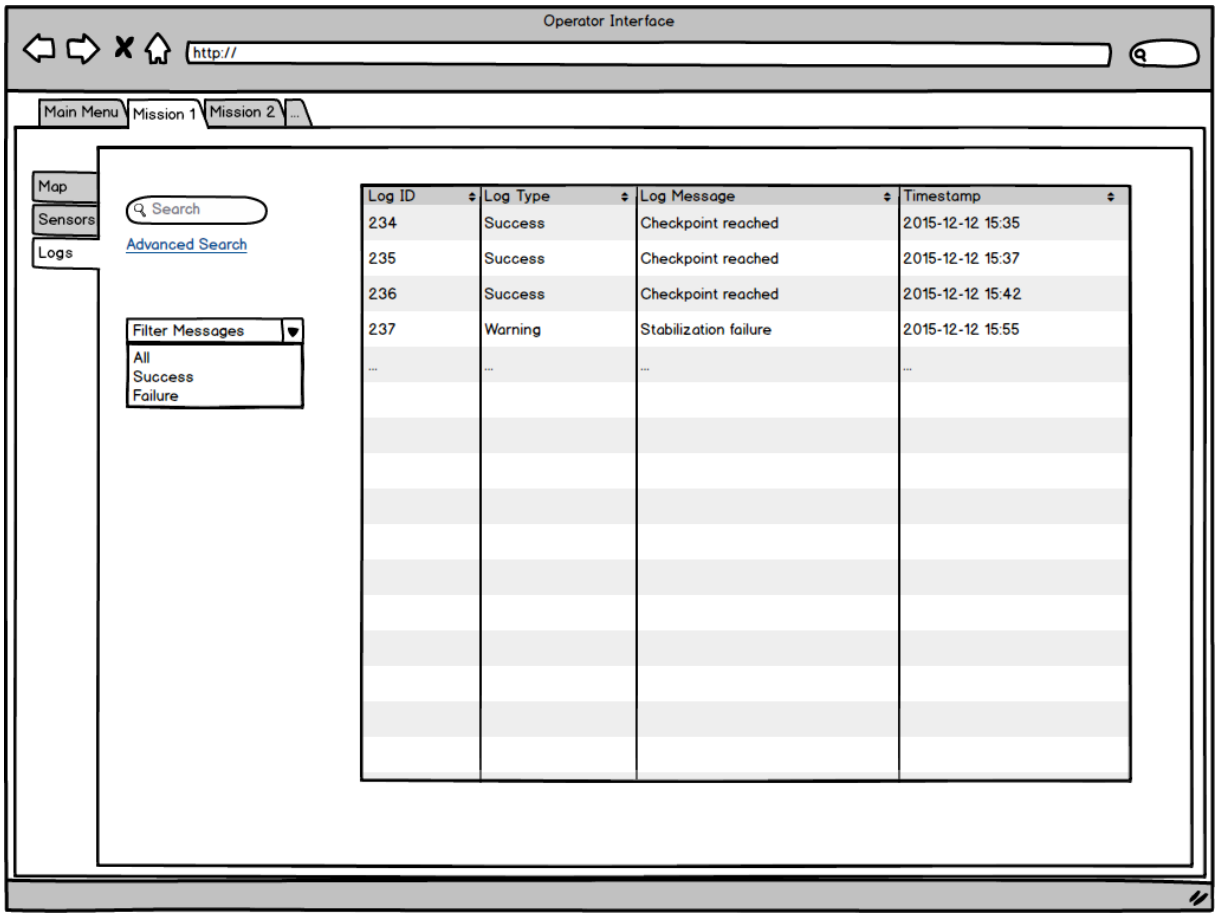
4. USER INTERFACE MODEL

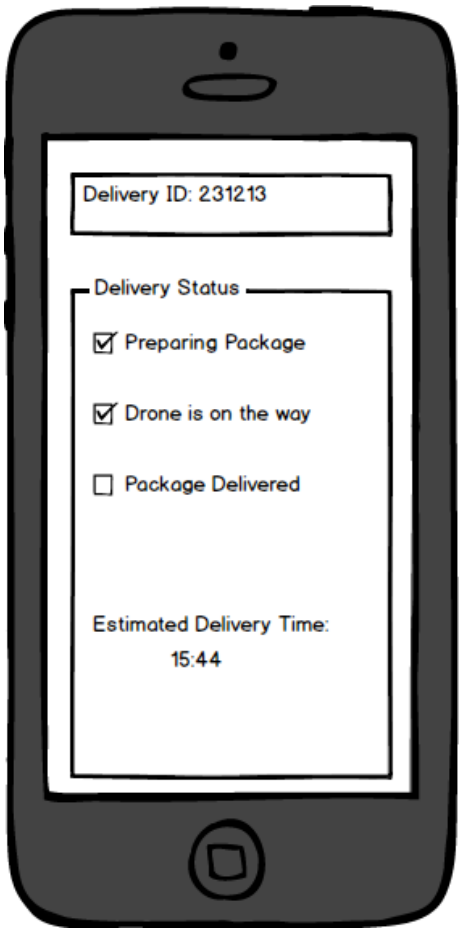





Operator Interface
http://
Main Menu
Mission 1
Mission 2
...
Map
Sensors
Logs
Select
Change Destination
Cancel Mission
Send Rescue Team
...
Drone
Drone ID: 2
Drone Status:
Lat: 45.1213..
Long: 42.55..
Destination
Lat: 45.1213..
Long: 42.55..
Rescue
N/A
Est Time:
10 min 23 sec
Drone
Checkpoint
Checkpoint
Destination


Operator Interface
http://
Main Menu
Mission 1
Mission 2
...
Map
Sensors
Logs
Motors
Motor1
Motor2
....
RF Module
Status : OK
Freq: 1.8 GHz
Channel: 6
GPS
Lat.: 43.1123..
Lon.: 35.2312..
Sensors
Gyro
Altimeter
Battery
Voltage: ...
...


Operator Interface
http://
Main Menu
Mission 1
Mission 2
...
Map
Sensors
Logs
Search
Advanced Search
Filter Messages
All
Success
Failure
Log ID
Log Type
Log Message
Timestamp
234
Success
Checkpoint reached
2015-12-12 15:35
235
Success
Checkpoint reached
2015-12-12 15:37
236
Success
Checkpoint reached
2015-12-12 15:42
237
Warning
Stabilization failure
2015-12-12 15:55


Delivery ID: 231213
Delivery Status
Preparing Package
Drone is on the way
Package Delivered
Estimated Delivery Time:
15:44