

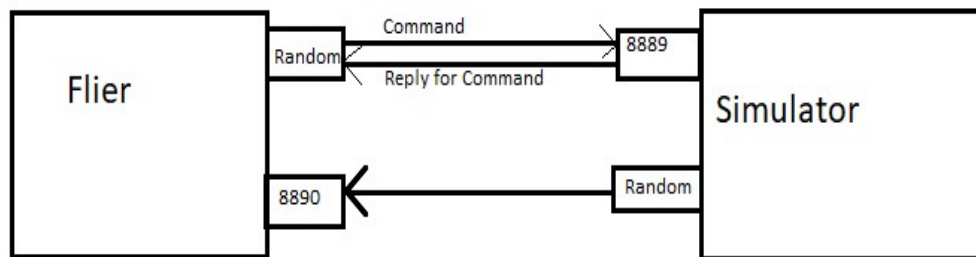
REPORT: ASSIGNMENT-2

INTRODUCTION:

In this project I became familiar with AME principles, Template method pattern, Observer method pattern, UML class and Interaction diagrams and lastly the unit testing techniques.

ENHANCEMENTS MADE:

1. Periodic Status tracking of the drones state using multithreading so that there is an updated state of the drone every 100ms.
2. Drone State is used to improve the mission execution, the execution of complex messages like flip and go aborts if the battery percentage goes below 25% or temperature exceeds 80, the drone performs a left or right move instead of flip if the battery percentage is in between 20-40.
3. Uses move and rotate method in the DroneState.java to get an estimate of drone's position after every maneuver and resets the drone state if the land command has been sent.
4. **Attempted** (GUI) but stopped as per the discussion with professor so as to correctly apply the Observer Pattern in the coming assignment.
5. **(Point 6)** The retries for sending a command, drone ports and address are taken as an input from the user to meet the requirement so that the user can set the 8890 and 8889 from the cmd accordingly, manage the number of retries for every command as per the need.
6. **(Point 7)**(a) The network ports that are communicating and their addresses are displayed to the user also the battery and temperature values are indicated if they go below a threshold.
(b) The program asks the user to enter the distance for Left/Right/Forward/Back commands, the degrees to rotate in CW/CCW commands and also the type of flip he wants (l/r/f/b).
(c) The drone's flip command is handled depending upon the battery percentage, if it is between 20-40 the drone performs a right move instead of flip and if the battery goes below 20, the drone lands safely.
7. **(Point 8)**The other developers can easily add new type of mission just by over-riding the Mission class method and also the template design pattern has been applied that sets the skeleton of a mission to be executed, the developer just has to over-ride the sendOtherCommands in Mission class to select the maneuvers he wishes to perform. The order is Command-Takeoff-otherCommands-Land. The users can also perform a custom mission by sending the commands through cmd these commands can be first validated with the simulator and then tested with drone. *(added after review)*
8. **(Point 9)**The components responsible for the network communication are used by both Drone Flier and Simulator (ports and communicator).



9. **(Point 10)** The component responsible for message serialization and de-serialization is used by both Drone Flier and Simulator.
10. **(Point 11)** The drone state is used by both Flier and Simulator (FlierStatusThread and SimulatorStatusThread).

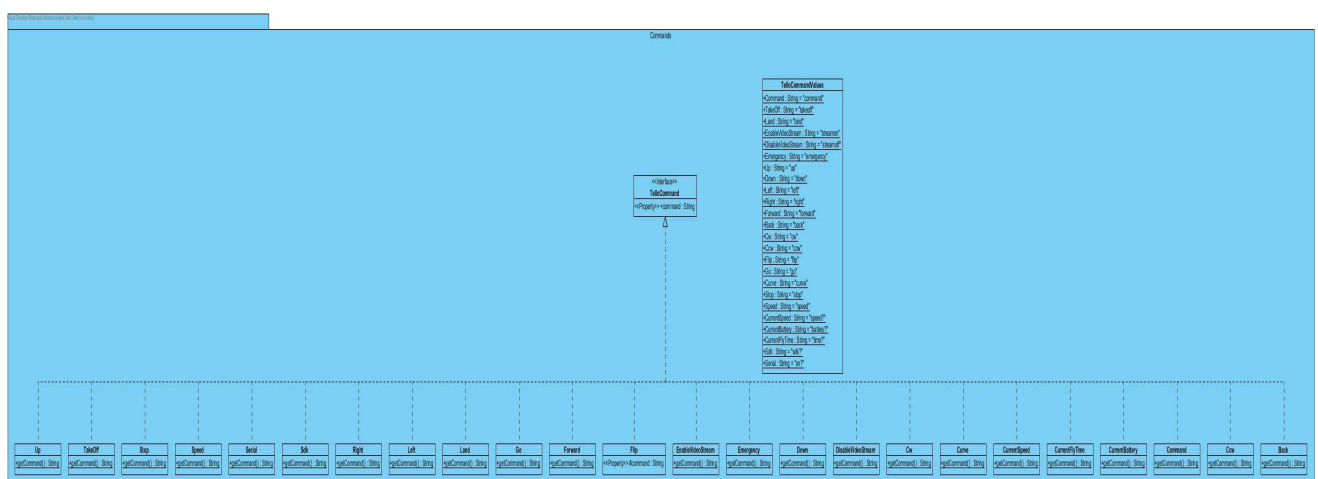
Design Patterns Applied:

Template method design pattern is to define an algorithm as skeleton of operations and leave the details to be implemented by the child classes. The overall structure and sequence of the algorithm is preserved by the parent class. Used in the execution of missions in my program. The order of execution is set to Command -> TakeOff -> otherCommands -> Land.

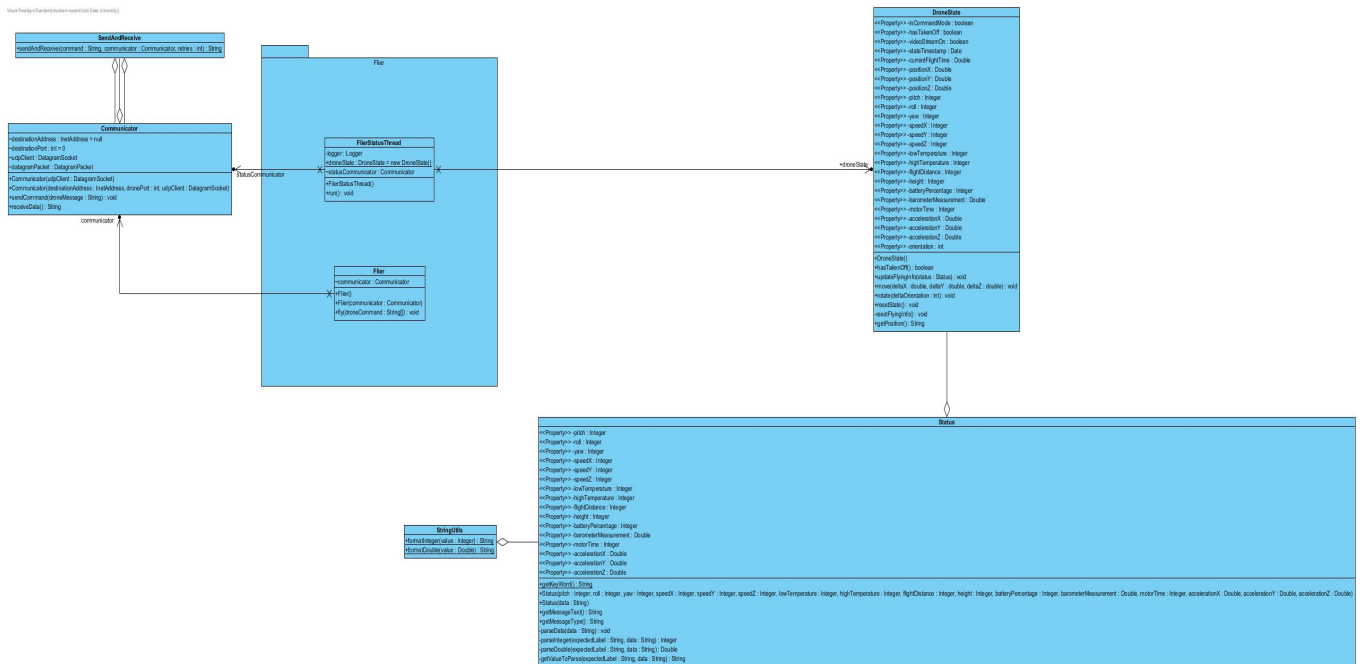
Observer Pattern: Observer pattern is used when there is one-to-many relationship between objects such as if one object is modified; its dependent objects are to be notified automatically. In my program as soon as the drone state is modified the status is updated on both the flier and simulator, multi-threading is used to execute different threads on flier and simulator side.

UML Class Diagrams:

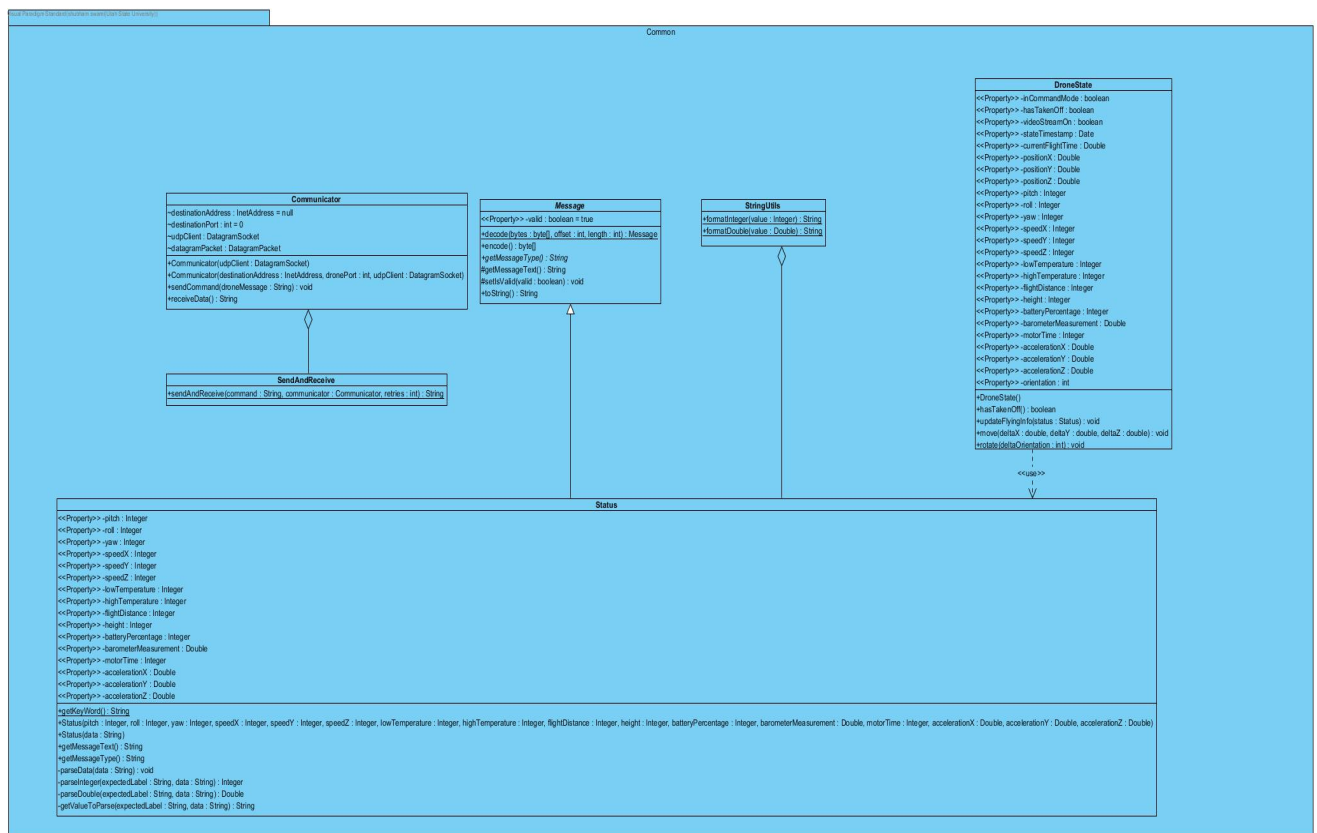
Messages



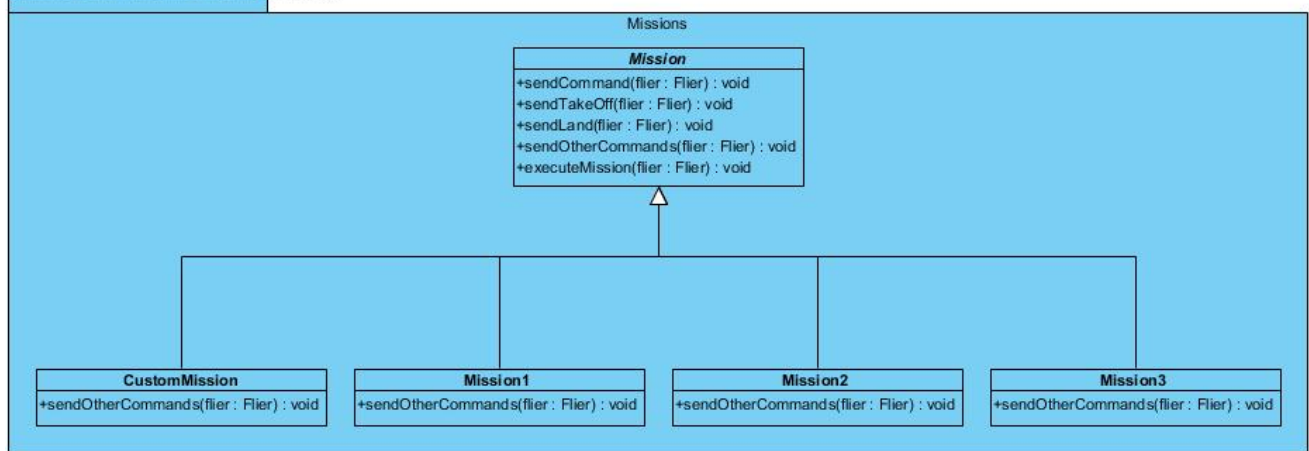
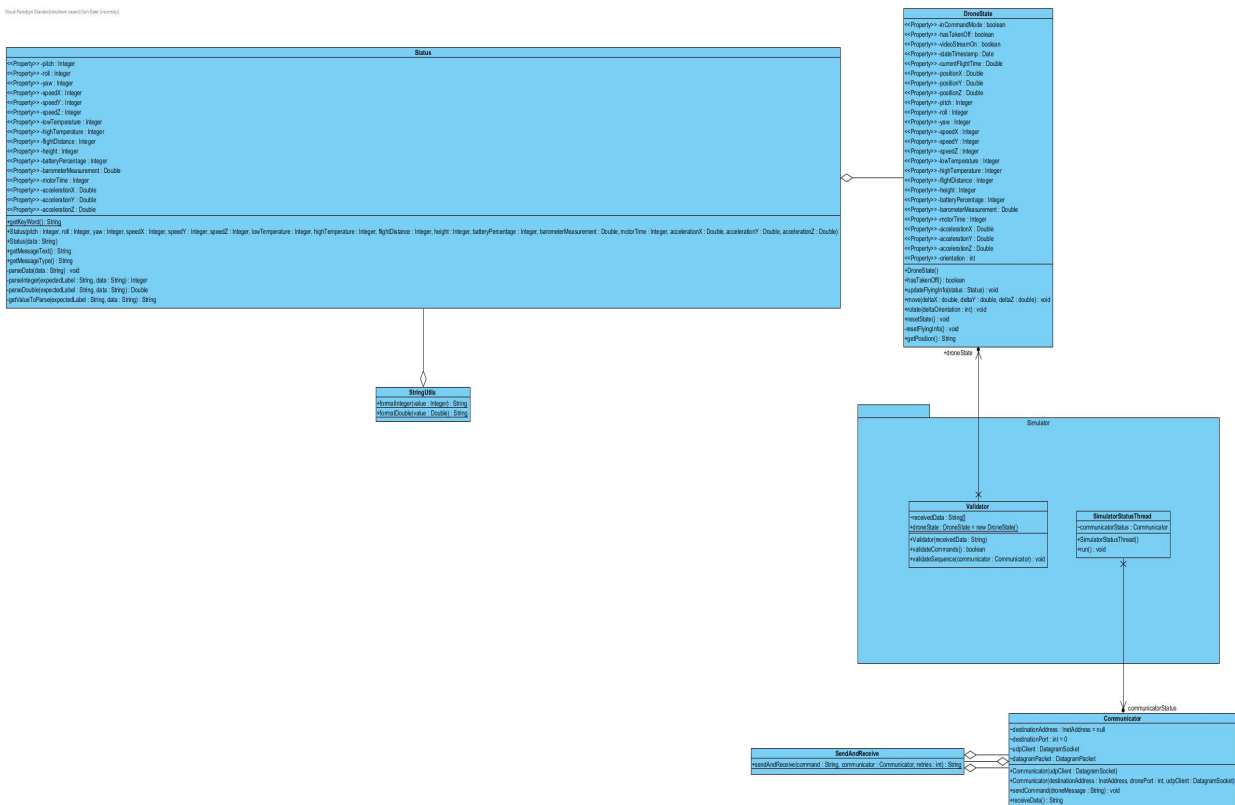
Flier:

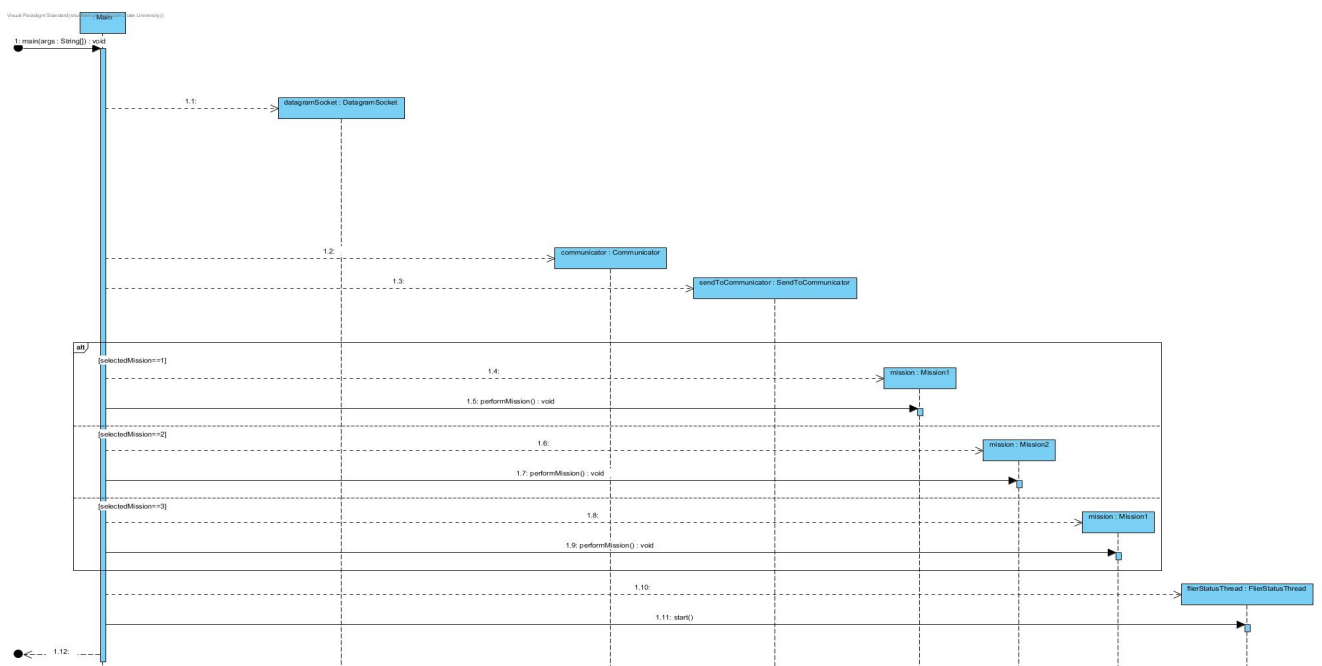


Common:



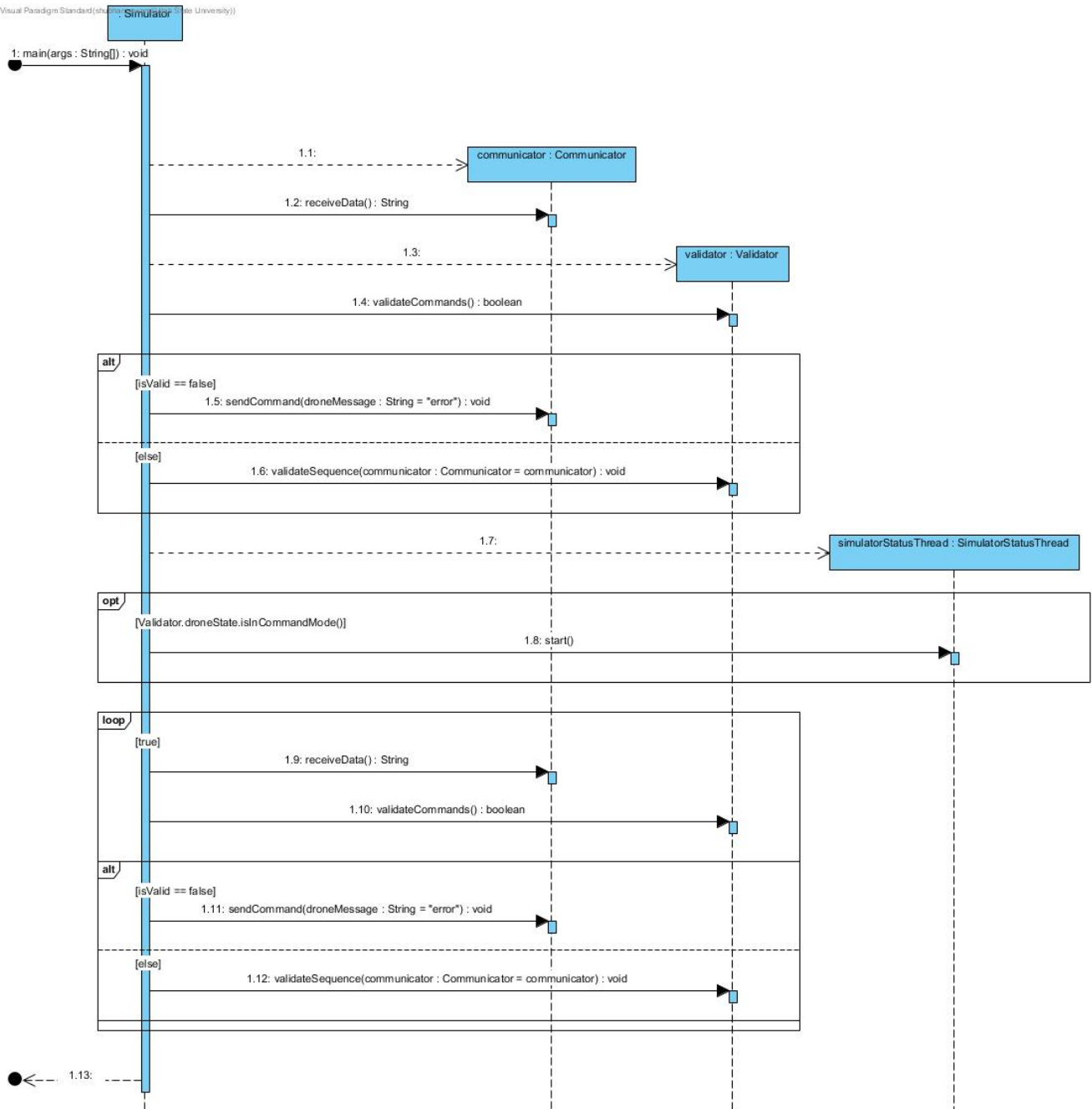
Visual Paradigm Standard (shubham swam(Utah State University))

Visit PenguinDirect.com/Free to see if you qualify for this offer.



Interaction Diagram Simulator Main

Visual Paradigm Standard (shuangshuang@shuangshuang.com)



Simulator Status Thread:

