Sprint Evaluation

After all researches are made about ROS and required installations are completed, we gave some basic commands like takeoff, land, togglecam etc. to quadrotor simulated on gazebo. We achieved to send and implement those and more commands on quadrotor .Now, we are more familiar with ROS, its working environment after researching and reading related tutorials and articles and we can manage and control the system on simulation. The goal we want to carry out in this sprint is to give a little complicated command set to quadrotor to follow a certain path. Since the first thing that we need to do is to operate the quadrotor on simulation environment before passing the real world operation and in order to operate the quadrotor manually, we need to use joystick after launching. Thus, we decided to install some additional packages for joystick control. Moreover, we made small changes in the plan. Since we are now working in simulation environment, installation processes and implementation on simulation environment (gazebo) took more time we expect, we could not pass the real world operation and we postponed this action to the next sprint. Moreover, we needed to make some updates on backlogs which are specified in details in part 3.

Team evaluation

We should work together for the first phase of our project as a team, since all team members should understand what the working principle of ROS environment and how the system is going to be operated.

Thus, we pay more importance to work together to understand the concept and tasks. So far, we have completed the required tools' installation and achieved to run quadrotor on simulation environment. Moreover, we did more than 5 meetings including with assistant and professor, which contributes us to progress on our project correctly and helps us in terms of improving our workings and researches on the project. After completing this phase (working on simulation environment), we are planning to work partial separate according to our tasks which we decided at the beginning of the term for each team member. As a team, we consider that we chose a correct strategy for this phase and next phases, and will try to continue applying those strategies at the end of the project.

Task	Assigned Member	4th Week				5th Week				6th Week				7th Week			
SRS Documentation	**1.** Mehmet Akif Akpınar **2.** Mustafa Buğra Tamer **3.** Rana Aygül **4.** Kadir Ekmekçi	🟥	🟥	🟥	🟥												
tum_simulator research, installation & creating first simulation	**1.** Mehmet Akif Akpınar **2.** Mustafa Buğra Tamer **3.** Rana Aygül **4.** Kadir Ekmekçi		🟦	🟦	🟦	🟦	🟦	🟦									
tum_ardrone research, installation & creating first mission	**1.** Mehmet Akif Akpınar **2.** Mustafa Buğra Tamer **3.** Rana Aygül **4.** Kadir Ekmekçi									🟦	🟦	🟦	🟦				
First Simulation: Creating first complete simulation which includes advanced plans	**1.** Mehmet Akif Akpınar **2.** Mustafa Buğra Tamer **3.** Rana Aygül **4.** Kadir Ekmekçi											🟩	🟩	🟩	🟩	🟩	🟩
Prepared 2nd Retrospective Document	**1.** Mehmet Akif Akpınar **2.** Mustafa Buğra Tamer **3.** Rana Aygül **4.** Kadir Ekmekçi													🟪	🟪	🟪	🟪

Backlog Updates

1) As a result of our researches, we back out of using Hector as a simulation environment. Although Hector is a valid simulation environment for any four motor quadrocopter, it's real-life and simulation synchronization is supported for only certain type of quadrocopter which we will not using. Because of this synchronization is a must for us we decided to not using Hector.

2) We decided to work with tum_simulator which is a simulation package for AR Drone 2.0. This package runs on Gazebo and it is more suitable to our project because it is developed for AR Drones and it has many features including simulation/real-life synchronization which we mentioned above.

3) We decided to run tum_simulator with a patch implemented on ROS Indigo instead of running on ROS Fuerte which is an out-date version of ROS and not supported after Ubuntu 12.04.

4) We decided to work with tum_ardrone as a mission planner.

5) We think that if we can handle most part of the project in simulation, we can handle in real-time also. Therefore we postponed real-time flying processes one or two weeks and decided to work on simulation environment before all planned real-time processes.

6) We decide to implement some pyton scripts which will send our control commands from terminal to tum_simulator for creating our auto-pilot for the project.

7) We decided to use joystick for manually controlling the UAV both in simulation and in real-life. Tum_simulator has a support for these goals.

9) We have to shift our schedule because of patch accordance problems between ROS and its libraries.