**BARTIN ÜNİVERSİTESİ MÜHENDİSLİK, MİMARLIK ve TASARIM FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**

**2023-2024 GÜZ DÖNEMİ**

**BSM401 – BİTİRME PROJESİ HAFTALIK RAPORU**

ÖĞRENCİNİN

giyim, insan yüzü, kişi, şahıs, duvar içeren bir resim

Açıklama otomatik olarak oluşturuldu

Fotoğraf

Adı Soyadı : Berkay ERGİN Öğrenci No : 21030310001 Tel 05388351273

E-posta : [21030310001@ogrenci.bartin.edu.tr](mailto:21030310001@ogrenci.bartin.edu.tr)

DERS DANIŞMANI BİLGİLERİ

Adı Soyadı, Ünvanı : Doç. Dr. Eyüp Burak CEYHAN

|  |
| --- |
| HAFTALIK ÇALIŞMA ÇİZELGESİ |
| **İNSANSIZ HAVA ARACI** |
| İHA ile LOJİSTİK |
| **Öğrencinin Çalışma Faaliyetleri** |
| SİMÜLASYON YAZILIMI :  Simülasyon yazılımı Oluşturulan Simülasyon yazılımının temeli DroneKit Framework üzerine kurulduğundan kütüphanenin sunduğu çok sayıda temel komut kullanılabilmektedir. Bu temel komutların çalıştırılabileceği  araç kütüphane üzerinden Vehicle adıyla yazılım geliştiricilere sunulmaktadır. Böylelikle geliştirilen Vehicle objesi sayesinde anlık bilgiler yazılım çalışma zamanında kullanıcıya iletilebilmektedir. Vehicle objesi  tarafından kullanıcıya aktarılacak anlık bilgiler doğrultusunda otomatik olarak verilecek komutların tayini sağlanabilmektedir. Örneğin pil seviyesinin belirlenen seviyeden düşük olması durumunda sistemin nasıl hareket edeceği veya anlık GPS sensör koordinatlarının belirlenememesi sonucu uçuşun başlayıp  başlayamayacağı gibi görev tayinleri bu obje aracılığı ile programcı tarafından gerçekleştirilmektedir. Görev mekanizmaları simülasyon yazılım katmanında belirlendiği gibi fiziksel kontrolcü katmanında da  tanımlanmaktadır. Kütüphanenin yazılımcıya sunduğu diğer bir komut ise LocationGlobalRelative olarak  adlandırılmaktadır. Temel altyapıda LocationGlobalRelative komutu oluşturularak drona istenilen sayıda görev koordinatları tayini ataması gerçekleştirilebilmektedir. Sistem bir döngü altyapısında çalıştığı için anlık olarak her komut sonrası Vehicle objesinin durumu sorgulanması gerekmektedir. Sistemin çalışmasında Simple\_goto komutu önemli bir yere sahiptir. Simple\_goto komutu temelde bir LocationGlobalRelative alarak koordinatlara gerçek veya simülasyonda tanımlanmış dronu koordinatlara ulaştırmayı hedef alır. Python dilinin sıralı olarak çalışma özelliğinden dolayı belirlenen komutların birer gerçekleştirilip gerçekleştirilmediğinin gözlenmesi hayati önem taşımaktadır. Örnek vermek gerekirse Simle\_goto komutu sonrası ikinci Simple\_goto komutunun verilmeden önce drone’un gerçekten o koordinatlara ulaşıp ulaşmadığı bir yazılım döngüsü içerisinde test  edilmektedir. Dron planlanan koordinata ulaştığı takdirde ikinci simple\_goto komutu kullanılmaktadır. Güvenli ve iyi düzeyde uçuşun temelleri göz önüne alındığında döngü sisteminin hatasız olarak kullanılması büyük önem arz etmektedir. Hazırlanmış olan yazılımın drone üzerinde çalıştırılması için Linux komutlarını Windows 10 test sistemi üzerinde çalıştırmak için kullanılan Cygwin girdi olarak $sim\_vehicle.py –map –console komutunu almaktadır. Cygwin tarafından sim\_vehicle.py dosyası çalıştırıldıktan sonra map ve console adlı iki adet parametre simülasyon sistemine gönderilmektedir. Konsol ve harita uygulamalarını anlık olarak izlemeyi kolaylaştıran komut uygulamaları kullanıcıya sunmak için oluşturulmuşlardır. Test sisteminde 42 Mission Planner gözlem uygulaması kullanıldığından bu parametrelerin kullanımı önem teşkil etmemektedir. Şekil 2’de belirtildiği gibi Missin Planner Uygulamasının ve yazılan python uygulamasının birbiriyle bağlanması için  MAVProxy üzerinden iki adet portun sistem üzerinde ayrılmış olması gerekmektedir. Öngörülen olarak bir |

|  |  |  |  |
| --- | --- | --- | --- |
| adet 14550 numaralı port kullanıma sunulmuştur. İkinci port ise MAVProxy komut satırı üzerinden açılmaktadır. | | | |
| **Danışman Önerileri** | | | |
|  | | | |
| Öğrencinin İmzası: |  | Danışmanın İmzası: |  |