(A Report of a project facilitated by)

PX4 SITL With Gazebo

BACHELOR OF TECHNOLOGY IN COMPUTERSCIENCE & ENGINEERING

(ARTIFICIAL INTELLIGENCE& MACHINE LEARNING)

Submitted by:
Chitturi Madhava badari narayana

Under the guidance of

Avijith Ashe

(TEACHING ASSISTANT)

DEPARTMENT OF COMPUTER SCIENCE &ENGINEERING

KAKINADA INSTITUTES OF ENGINEERINGAND TECHNOLOGY

(Approved by AICTE & Permanently Affiliated to JNTUK,KAKINADA
and accredited by NAAC) Yanam Road,Korangi, Andhra Pradesh,pin-533461

2025-2026

Table of Contents

Week 1 Documentation:

Proposal Selection & Initial Setup

1. Project Proposal Selection

Title:

- Simulation and Implementation of GPS Failure Failsafe for a Quadrotor with PX4 SITL using Gazebo

Objective:

- This project aims to simulate a failsafe mechanism for GPS failure in a quadrotor in a simulated setup using PX4 SITL and Gazebo. The failsafe will detect GPS loss and execute predefined safety measures like position hold, landing, or return to home.

Methodology:

- Employ SITL simulation of PX4 using Gazebo.
- simulate GPS failure mode and analyze how the quadrotor responds to it.
- build a failsafe system that assures recovery safely.
- Use QGroundControl for flight monitoring and param setup.
- Assure results under various test conditions.

Expected Outcome:

- A functioning GPS failsafe system that will be testable in a simulated quadrotor environment.
- Logging of performance indices and safety protocol.

2. Environment Setup

- **2.1 Installing ROS 2 & Dependencies**

- ROS 2 (optional) or ROS 1 is required for simulation. The following were run:

```
sudo apt update && sudo apt upgrade -y
sudo apt install ros-humble-desktop
source /opt/ros/humble/setup.bash
```

- ## 2.2 Installing Gazebo Classic (Gazebo 11)

- Gazebo 11 is the version I choose for PX4 SITL compatibility:
  `sudo apt install gazebo11 libgazebo11-dev`
- If the command does not work run: `sudo apt install gazebo` and check version (gazebo —version)

- ## 2.3 Installing MAVLink Libraries

- MAVLink enables communication between PX4 and Gazebo:
  `sudo apt install python3-pip`
  `pip3 install pymavlink MAVProxy`

- ## 2.4 Cloning and Building PX4 from Source

```
git clone --recursive https://github.com/PX4/PX4-Autopilot.git
cd PX4-Autopilot
bash./Tools/setup/ubuntu.sh
```

- Build PX4 for SITL with Gazebo:
  `cd PX4-Autopilot`
  `make px4_sitl gazebo`

- ## 2.5 Testing Basic ROS-Gazebo-PX4 Integration

- Execute test simulation to verify setup:
  `source ~/PX4-Autopilot/Tools/setup_gazebo.bash`
  `make px4_sitl gazebo`
- Expected result: Gazebo should boot up with a quadrotor, and the PX4 console should display that the system is running.

- ## 2.6 Installing QGroundControl

- QGroundControl was installed to monitor and control the drone. The installation procedure :
- Downloading QGroundControl from the official website: [QGroundControl Downloads](QGroundControl)

- Installing QGC on Ubuntu using:

  sudo apt install qgroundcontrol

- Then make it executable by chmod +x ./QGroundControl.AppImage
- To launch ./QGroundControl
- Opening QGC and checking installation.

Summary:

- This project step involves simulating and setting up the GPS failure failsafe for a quadrotor with PX4 SITL and Gazebo. The goal is to create a failsafe mechanism to identify GPS loss and execute pre-programmed responses like position hold, landing, or return home. The approach includes establishing the simulation environment, testing GPS failure, and verifying the failsafe system using QGroundControl for monitoring. The environment setup involved the installation of ROS 2, Gazebo 11, MAVLink libraries, and PX4, then their integration to facilitate effortless use. Preliminary tests were conducted to confirm the successful installation and operation of the SITL simulation.

Outcome

- Completed the setup of PX4-Autopilot and sitl with gazebo.