|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Software Design Specifications**  Drone Navigation Using Brain-Computer Interface (BCI)  Version: 1.0   |  |  | | --- | --- | | Project Code | 19S-16 | | Supervisor | Dr. Ahsanullah Abro | | Co-Supervisor | Dr. Ahmed Ali Shah | | Project Manager | Muhammad Raheal Safdar | | Project Team | Naneeta  Madiha | | Submission Date | 20th September 2022 |   Signature: |

**Table of Contents**

[**1. Introduction 3**](#_Toc114614964)

[**1.1. Purpose of Document 3**](#_Toc114614965)

[**1.2. Scope of the development project 3**](#_Toc114614966)

[**1.3. Definitions, acronyms, and abbreviations 3**](#_Toc114614967)

[**2. Design Considerations 4**](#_Toc114614968)

[**2.1. Assumptions 4**](#_Toc114614969)

[**2.2. Constraints 4**](#_Toc114614970)

[**2.2.1. Platform 4**](#_Toc114614971)

[**2.2.2. Operating System 4**](#_Toc114614972)

[**3. Architecture 4**](#_Toc114614973)

[**3.1. Overview: 4**](#_Toc114614974)

[**4. Detailed Software Design 5**](#_Toc114614975)

[**4.1. Use Case Diagram: 5**](#_Toc114614976)

[**4.2.** **Sequence Diagram: 6**](#_Toc114614977)

1. Introduction
   1. Purpose of Document

This document aims to build a BCI system that control drone navigation. In this document, we will create use case diagram, sequence diagram and system architecture, which help us to create our system easily.

* 1. Scope of the development project

Drone navigation using BCI is not a new idea but it can be game changer. There are plenty of solutions in market that are used to control the drone but they all involve the physical interaction of user with controller e.g., a mobile app and remote. In a mobile app user will have to use his hand to control drone and it is the case with remote controller. There is another technological advancement, which we have proposed where less physical interaction is needed and that is BCI which uses brain signals to control the drone with the help of OpenBCI headset.

After designing and development of the project, we will be able to provide a system that will help user to control drone by wearing EEG headset. A user will have to wear EEG headset and will think of any of the six commands that are left, right, takeoff (up), land (down), forward and backward to move the drone in respective direction. One of the limitations of this project is that drone will be controlled only in six directions and no more movements like rotation, bending and flipping will not be entertained and secondly, as EEG signal is sensitive input so there will be latency tradeoff because we cannot match the same speed of remote control or mobile app with brain signals. Furthermore, this project can be extended by adding functionalities like weight lifting and photography using drone.

* 1. Definitions, acronyms, and abbreviations

1. BCI – Brain Computer Interfaces
2. EEG – Electroencephalography
3. SRS – Software Requirement Specification
4. SDLC – Software design life cycle
5. OS – Operating System
6. Design Considerations
   1. Assumptions

A1: OpenBCI headset is connected with the OpenBCI GUI.

A2: Wi-Fi and Bluetooth are available.

A3: Tello Edu drone is switched on.

* 1. Constraints
     1. Platform

C1: Works well on PC.

* + 1. Operating System

C2: It can work windows operating system 8 or higher.

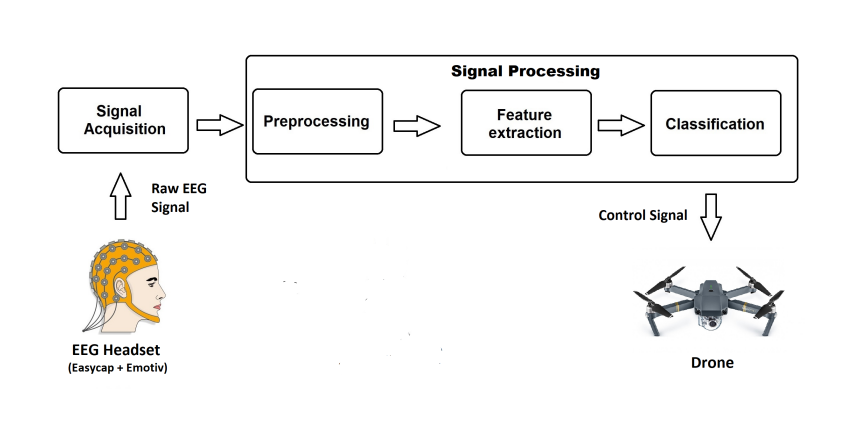
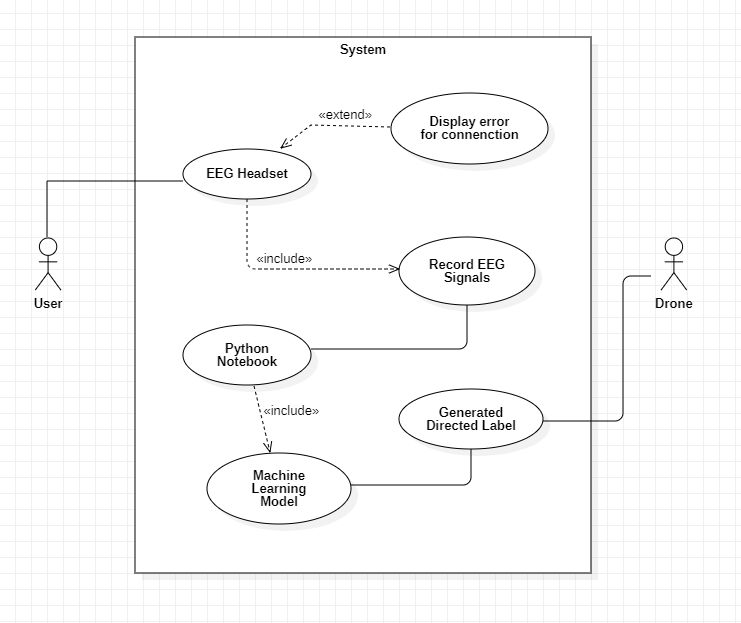
1. Architecture
   1. Overview:
2. Detailed Software Design
   1. Use Case Diagram:

Figure : Use Case Diagram



The Figure 1 shows the use case of our project. The user wears the headset. It is determined; weather the headset is placed correctly or not. If for some reasons the headset is not properly placed over the head, then it will be displayed as an error. If the headset is properly placed, we will start recording the brain signals. These brain signals are then sent to the Python Notebook. In that Python Notebook, first these signals will be cleaned and after cleaning these signals will be passed by the Machine Learning Model for its classification. The model gives us the directed label. This label will then sent to the Drone using the Python Notebook.

* 1. Sequence Diagram:

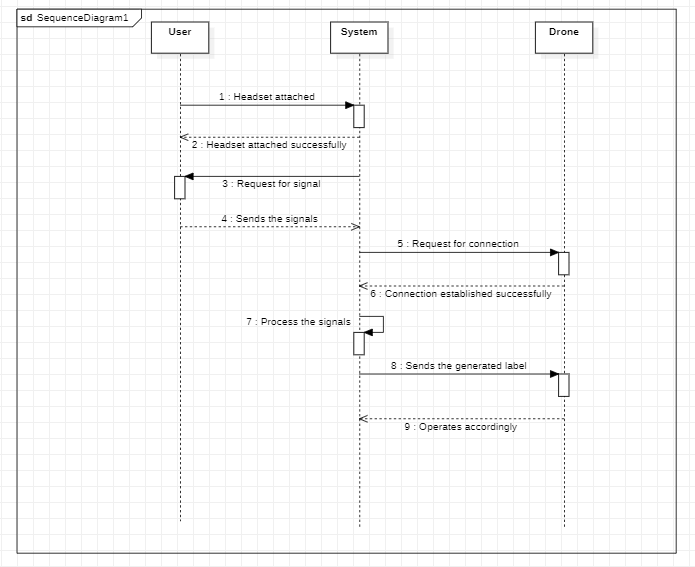


Figure : Sequence Diagram

The Figure 2 shows the sequence diagram of our project. It is straightforward. The requests are response will run in the sequence. Starting from the headset attachment. The user will wear the headset and the system will response with the success of the headset attachment. After this, the system will request for the signals and the headset will record the signals and response the system request by sending the signals to the system. After this, the connection with the drone is established. The System will request the drone for the connection using the Wi-Fi. After successful connection with the drone, the drone will response to the system. Once the connection is done with both the headset and the drone, the system will process the brain signals that were requested earlier and these signals will be used to generate an output as a command. The system will then send these generated labels to the drone and the drone will act according to that command.