**Driverless Car System Implementation**

This assignment focuses on implementing the software components necessary to support the operation of a driverless car, as outlined in the provided class diagram from system design, Figure 1.

A screenshot of a computer

Description automatically generated

[Figure 1]

This README includes the implementation of software components necessary to support the operation of a driverless car. Below, you'll find a description of the code and the testing code specifically for the system along with instructions on how to execute the code.

**driverless\_system.py**

It consists of several classes that simulate the functionality of a driverless car system. Here's an overview of each class:

**DriverlessSystem**:

This class represents the main system controlling the driverless car.

It has methods to make decisions, detect obstacles, and store data.

The make\_decision method decides whether to proceed or halt based on obstacle detection.

The detect\_obstacle method simulates the detection of obstacles using a camera. To avoid any doubt, it will need to be replaced once being implemented to the real system.

The store\_data method stores captured image data in the Data class.

**Camera**:

This class represents the camera module of the driverless car.

It has a method capture\_image that simulates capturing an image. To avoid any doubt, it will need to be replaced once being implemented to the real system.

For demonstration purposes, it returns a placeholder image.

**Data**:

This class represents the data storage module of the driverless car.

It has a method store\_data to store captured image data.

**Obstacle**:

This class represents the obstacle detection module of the driverless car.

It has an attribute detected to simulate obstacle detection.

**Decision**:

This class represents the decision-making module of the driverless car.

It has an attribute result to store the decision (proceed or halt).

**test\_driverless\_system.py**

The testing code uses the unittest framework to test the functionality of the driverless car system. Here's a description of the test cases:

**test\_make\_decision:** Tests the make\_decision method of the DriverlessSystem class. It verifies whether the system makes the correct decision based on obstacle detection.

**test\_detect\_obstacle:** Tests the detect\_obstacle method of the DriverlessSystem class. It ensures that the system detects obstacles correctly.

**test\_store\_data:** Tests the store\_data method of the DriverlessSystem class. It checks whether the system stores data correctly after capturing an image.