Driverless Car System Design

Overview:

This design proposal document demonstrates some systems needed to operate a driverless car. The document uses different UML diagrams to demonstrate how the system is designed.

Systems described in this document:

* Collision and object detection: This system uses a series of camera sensors to detect, classify and identify signs, objects, lanes, vehicles, pedestrians, and more. Once the system detects these objects, it adds them in real-time to a queue of the most dangerous objects to the least dangerous objects. Since a queue (Queue Data Structure - GeeksforGeeks, 2022) is (FIFO) First In First Out data structure, it is excellent for this use. Once an object is added to the queue, it will be dealt with first. Once it is dealt with it will be popped out of the queue or dequeued, and the next object in the queue will be dealt with next.
* Driver monitoring system: It is important to keep track of the driver's vitals to ensure the driver is awake and in good health. The system keeps track of the driver's vitals with various sensors placed on the steering wheel, driver's seat, and vehicle dashboard. If the system detects that the driver is incapacitated, it will alert the driver with an audible noise. If the driver does not respond, the vehicle will pull over to the side of the road and call the police and emergency services. The system uses a Stack data structure (Stack Data Structure - GeeksforGeeks, 2022). A stack is a Last In First Out (FILO) structure that is perfect for keeping track of any kind of history. In the case of this system, the driver's health vitals will always be at the top of the stack. The stack can be popped to view the previous vital health measurements.
* Security system: Once the driver leaves and locks their vehicle, the vehicle will activate sentry mode. Sentry mode turns on proximity sensors and cameras around the car and ensures no one is breaking into the car or has hit the car. The cameras will always be recording the previous 5 minutes. If the vehicle detects a hit or breaking it will save the previous 5 minutes and start streaming everything to the cloud. The vehicle will also alert the driver by sending an SMS and an email. This system uses a dictionary (Dictionary Operations in Data Structure, 2022) to store a key/pair value of each recording and the recording name. The dictionary data structure is perfect for this use case since it uses the key as an index to and the recording as a value.
* Logging System: It is important to know what happened and when something happened. The logging system uses a List (How do Dynamic arrays work? - GeeksforGeeks, 2022) of strings as a buffer. The great thing about a List data structure is that it can grow and shrink depending on how many elements you add or remove from the list. Since we have no idea how many messages the system will log, this data structure is perfect for this use case, unlike an array with a fixed size.

UML digrams link using Draw.io

<https://drive.google.com/file/d/1s3WgADj0IJyZHFy15Gl2hcNEldimDyow/view?usp=sharing>

References:

GeeksforGeeks. 2022. *Queue Data Structure - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/queue-data-structure/> [Accessed 8 August 2022].

GeeksforGeeks. 2022. *Stack Data Structure - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/stack-data-structure/> [Accessed 8 August 2022].

Tutorialspoint.com. 2022. *Dictionary Operations in Data Structure*. [online] Available at: <https://www.tutorialspoint.com/dictionary-operations-in-data-structure> [Accessed 8 August 2022].

GeeksforGeeks. 2022. *How do Dynamic arrays work? - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/how-do-dynamic-arrays-work/> [Accessed 8 August 2022].