Jennifer Cafiero

SSW 555 Homework 2

I pledge my honor that I have abided by the Stevens Honor System.

1. Three features relevant to car changing lane software module:

* The car must not crash into any vehicles or objects on the road.
* The system must change lanes to place the vehicle between two other vehicles or objects on the road.
* The system must successfully move the vehicle to be in the correct lane.

1. Use Case 1

* Name: Avoid accidents with nearby objects or vehicles
* Description: The user initiates changing lanes by triggering the turn signal. The system will check its surroundings on the appropriate side and make note of the objects and other vehicles around it. The system will proceed to change lanes if there is enough room to do so. If there is no longer a safe amount of room for the car in the new lane, the vehicle will stop changing lanes and it will go back to the lane it came from, if that is still a safe situation.
* Actors: The vehicle, the driver, nearby objects/pedestrians
* Basic Flow: 1. The driver initiates changing lanes by triggering the turn signal and the vehicle begins to do so. 2. The vehicle becomes too close to another object or vehicle. 3. The system recognizes that the vehicle is too close to another object. 4. The vehicle moves away from the object and stops changing lanes.
* Alternate Flow: The driver could change his or her mind and not want to change lanes anymore. The driver could manually change lanes. An external object can suddenly move too close to the car.

Use Case 2

* Name: Change lanes to put car between two other vehicles or objects
* Description: When the driver decides to switch lanes, the vehicle will autonomously move itself to be in between two other vehicles or objects.
* Actors: The vehicle, the driver, other cars and objects to place the vehicle between
* Basic Flow: 1. The driver decides to switch lanes. 2. The driver initiates changing lanes by triggering the turn signal. 3. The vehicle will check its sensors to make sure there is adequate space in the neighboring lane for it to move into. 4. The vehicle will move into the new lane successfully.
* Alternate Flow: The driver could manually change lanes. The driver could change his or her mind and not want to change lanes anymore. An external object can move too close to the car.

Use Case 3

* Name: Successfully enter new lane
* Description: When the driver wants to change lanes, he or she can press a button and the vehicle will leave the lane it is driving in and enter the desired new lane.
* Actors: The vehicle, the driver, nearby objects/pedestrians
* Basic Flow: 1. The driver makes the decision to change lanes. 2. The driver initiates changing lanes by triggering the turn signal. 3. The vehicle leaves the lane it was operating in and enters its new lane.
* Alternate Flow: External objects can move to be in the way of the vehicle changing lanes. The driver could manually change lanes.

|  |  |  |
| --- | --- | --- |
| Title: Avoid Accidents | | |
| Acceptance Test: avoidAccident | Priority: 1 | Story Points: 3 |
| As a user, I want to avoid crashing with other objects so that other pedestrians and myself remain safe while my vehicle is on the road. | | |

|  |  |  |
| --- | --- | --- |
| Title: Change lanes | | |
| Acceptance Test: changeLane | Priority: 1 | Story Points: 8 |
| As a user, I want to have my vehicle change lanes autonomously so that myself and others remain safe and the vehicle is in the lane of my choosing. | | |

|  |  |  |
| --- | --- | --- |
| Title: Enter new lane | | |
| Acceptance Test: enterLane | Priority: 2 | Story Points: 5 |
| As a user, I want the vehicle to safely enter the new lane so that the vehicle is in the lane of my choosing for my driving needs. | | |

1. Advantages/Disadvantages of use cases vs user stories

Because the *Driverless Cars* project is on a larger scale, using use cases can be advantageous because of the amount of detail and structure they provide for the development team. Having detailed basic and alternate flows available to the team can provide a good starting point for developing the software. However, use cases have no prioritizing of tasks so the team might not clearly understand which tasks will be more difficult for development. User stories are used in agile planning and each task is simple to read and understand. Use cases have clearly labeled priorities as well as the addition of story points. These story points allow for developers to quickly understand how long a task might take to develop, but user stories don’t allow for as much of an in-depth analysis as use cases can provide.