Software Requirements Specification

for

Self Driving Car - Tesla Y

**Version 2.0 approved**

**Prepared by JJ Javier, Aaron Hamilton, Dheeraj Singavarapu**

**JAD**

**12/04/23**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 2

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 3

2.6 User Documentation 3

2.7 Assumptions and Dependencies 3

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.2 Hardware Interfaces 4

3.3 Software Interfaces 4

3.4 Communications Interfaces 4

**4. System Features 4**

4.1 Auto-Steer 4

4.2 Auto-Brake 4

4.3 Traffic Light / Stop Sign Check 5

4.4 Location / Route Management 6

4.5 Speed Management 6

4.6 Auto-Park 7

**5. Other Nonfunctional Requirements 8**

5.1 Performance Requirements 8

5.2 Safety Requirements 8

5.3 Security Requirements 8

5.4 Software Quality Attributes 8

5.5 Business Rules 9

**6. Other Requirements 9**

**Appendix A: Glossary 9**

**Appendix B: Analysis Models 9**

**Appendix C: Requirements Engineering 10-33**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| JJ Javier, Aaron Hamilton, Dheeraj Singavarapu | 10/17/23 | Initial Draft | 1.0 |
| JJ Javier, Aaron Hamilton, Dheeraj Singavarapu | 12/4/23 | Final Draft | 2.0 |

# Introduction

## Purpose

The purpose is to create a safe self driving mode application in pursuit of a safer driving environment. The application will allow unwavering decision making and communications between the environment and the vehicle. Drivers with difficulty in making these decisions while driving can rest assured as the self-driving vehicle will have efficient and accurate priority, reaction time, and security. As the target audience are those incapable of driving themselves, elderly folk, or people who have high interest in the technology that comes from the self-driving vehicle, the car’s purpose is to allow driverless motion to get from one destination to another.

Created by,

JJ Javier - [javie032@csusm.edu](mailto:javie032@csusm.edu)

Aaron Hamilton - [hamil097@csusm.edu](mailto:hamil097@csusm.edu)

Dheeraj Singavarapu - [singa002@csusm.edu](mailto:singa002@csusm.edu)

## Document Conventions

This document follows IEEE formatting conventions. The document is written in Arial, double spaced, and font sizes for main headers are 18 pt, and subsection headers are 14pt and body sections are 12pt. Section headers and subsections are numbered and bolded. Every requirement statement has its own priority.

## Intended Audience and Reading Suggestions

The intended audience of our proposal document are those incapable of driving themselves, the elderly who can no longer drive themselves, or those who have a high interest in the car and the technology that comes from it. The SRS will describe the vehicle's main and intended functions along with the non-functional requirements. Within this document you will find the product’s design, user cases, hardware and software specifications, and many of its features that come with it.

## Product Scope

The scope of this product and the purpose behind it is to create a safer driving environment. With the use of a highly advanced, self-driving car will be used to lower driving accident rates and prevent injury to passengers. With the car being able to use its sensors to communicate the environment to its software, it can react, responde, and properly take effective action based on its surroundings. Goals would include a lowering of driving accidents per year, reduce the congestion of traffic, and the ability to communicate from car to corporate about how they’re doing and what more can be done to improve. This product aims to make the roads a safer place, not just for those who cannot drive, but those who may drive on the same roads as one.

## References

“Model Y Owner’s Manual.” Tesla, [www.tesla.com/ownersmanual/modely/en\_us/](http://www.tesla.com/ownersmanual/modely/en_us/).

# Overall Description

## Product Perspective

The origin of the Tesla Model Y can be traced to the product line known as Tesla. Co-founded most notably by Elon Musk, it is a new car added to the line of products and is very similar to its predecessors and other models of the same line such as: Model S, Model 3, and Model X. The Model Y is a cheaper version of the Model X in price and other aspects. Aspects like size, battery power, and functionality of the doors make the Model X much more expensive. Its system is within the shared ecosystem of other tesla models and uses both shared software and technology.

## Product Functions

Functional requirements include the following, auto-direction with the steering wheel that keeps the car on the road, object detection around the vehicle that ensures no crashing, automatic gas usage so that the car remains on the speed limit without the driver having to push the gas pedal, automatic brake usage so that the vehicle will automatically begin to brake in accordance to a stop or traffic, and finally a system that detects roads, roadlines, and different terrain features so that the vehicle can stay on the road in its lane without failure.

## User Classes and Characteristics

Some classes may include one for the steering, acceleration, braking, sensor input data, and the communication between sensor and software. Most, if not all product functions are used as the operator uses the vehicle and its system. The experience needed is for those who are capable of driving and passed the driver’s license exam. One is not able to operate the vehicle without the acquisition of a driver’s license. The most important classes are the main function of the car itself, the acceleration, braking and steering functions.

## Operating Environment

The environment where the hardware platform operates is a computer system built into the Tesla Model Y by Tesla themselves. The software is mostly operating on a system Tesla continuously updates as time goes on. The latest update to the software was made around October 12, 2023 known as 2023.32.9. Other software components of the Tesla Model Y include, Autopilot and Full Self-Driving (FSD) Software, Touch screen infotainment system, as well as its Connectivity software. FSD is self explanatory as it is the system which allows self driving as a feature, the infotainment system comes with a touchscreen pad in the center of the car and is a hub for all tesla applications, finally the connectivity software is how tesla is able to receive digital updates or OTA(over the air) updates.

## Design and Implementation Constraints

Tesla’s design allows the owner / user to have easy access and easy understanding of how the system operates. Users must follow all car use regulations and protocols in order to use the vehicle on public roads. The car must be charged in order to run accompanied by the key card given in order to unlock and use the car with its features. The car itself has memory storage for the user’s data and to keep their preferences. The car overall has everything it needs when acquired including different tools, accessabilities, and security. The operator must maintain the vehicle if in contact with any damages to the car itself or the system. Software updates will be available to the user at the given moments, the user must be able to navigate to the update settings in order to accomplish the software update altogether.

## User Documentation

User manual and guide will be available within the system. Or the user may find the online manual located at [www.tesla.com/ownersmanual/modely/en\_us/](http://www.tesla.com/ownersmanual/modely/en_us/).

## Assumptions and Dependencies

First and foremost it is assumed that the Tesla Model Y will comply with basic regulatory standards and rules placed on electric vehicles that also include autonomous driving. It is also assumed that both vehicle and user data is highly protected from any sort of data leaks as well as cyber attacks. It is also assumed that there will be consistency in the manufacturing of the Tesla Model Y with little to no defects in released products as well as meet quality standards. Finally we assume that the OTA or over the air updating system will continue to put out updates for the Tesla Model Y. Now as far as dependencies go the Tesla Model Y depends a lot on the software updates provided by Tesla themselves. The Tesla Model Y also depends on the availability of charging stations around the world. Another thing the Model Y depends on is the mobile apps functionality and compatibility. It also depends on the changes that can happen to regulations as far as autonomous and electric vehicles go. Finally the Model Y is dependent on the access to the internet or a cellular network in order to update the car and other remote access features.

# External Interface Requirements

## User Interfaces

The user interface will be a tablet-like screen within the front center of the car in order to easily navigate between the many features including the self-driving application. When starting the car, the screen will turn on and show a homepage for the user to navigate. The screen will show the vehicle's battery status, location, exterior display, and multiple app icons. From there the user may go through the different tabs and change settings and customize the vehicle to their suitability. The tablet-like screen is a touch screen so the user should be able to easily go through the car’s features and learn what the vehicle entails specifically for the self-driving application. Buttons used will be on the steering wheel of the vehicle. There are two buttons on the left and right side of the wheel that allow the user to press, open or turn features of the vehicle itself.

## Hardware Interfaces

The Tesla Model Y uses a variety of sensors and cameras in order to get input to the software in communications in order to tell its environmental surroundings and relationship in distance. Hardware overall consists of the car and its components to function a car, the steering wheel for additional navigation, its side, front, and back cameras, as well as sensors that communicate the environment with the software. Another huge hardware component is the built-in tablet-like screen used at the front of the car, right of the steering wheel. This component allows the user to navigate through the software features and help guide the user in operating the vehicle as a whole.

## Software InterfacesThe software interfaces of the Tesla Y self-driving car facilitate critical communication and data exchange for the system's seamless operation. The vehicle's central processing unit (CPU) and embedded software work in tandem with the self-driving system, enabling real-time decision-making and control. Mapping and navigation databases provide essential location and route information, ensuring accurate route planning. The self-driving system also relies on Tesla's over-the-air (OTA) update system to continuously improve and update its software. In addition, third-party API integration allows the system to obtain real-time traffic data and road conditions. All these interfaces, supported by Tesla's proprietary operating system, contribute to the vehicle's safe and efficient autonomous operation on the road.

## Communications Interfaces

Communication mainly comes from outside sensors of the vehicle to the software. The sensor to determine what object is around the car such as another vehicle, bike, pedestrian and other foreign objects. These sensors tell the software what the objects around them are and how far it is from the vehicle. With so, the car is able to determine how far it needs to be before deceleration or how far is far enough to begin accelerating. This also allows the vehicle to stay in lane and be aware of when to change lanes based on the sensory input of how far an object is within a lane to the right or left of the vehicle. The vehicle may also submit any crash reports or feedback from the user to the manufacturer. With customer service available at their given hours, users may get access to their data or get help in understanding how functions within the vehicle as a whole work entirely. All the communications from the user specifically come from the tablet screen as the users are able to navigate through to access communications.

# System Features

System features will include auto-steer, auto-brake, traffic light / stop sign check, location/route management, speed management, and auto-park

## Auto-Steer

4.1.1 Description and Priority

Keeping the vehicle in proper road lanes while following speed limits

Priority: High

Benefit: 9

Cost: 5

Penalty: 5

Risk: 3

4.1.2 Stimulus/Response Sequences

User Action: User selects on the auto drive application

System Response: From the moment auto drive is selected, the vehicle will take over all steering actions taking in the environment information to determine what actions need to be taken.

4.1.3 Functional Requirements

REQ-1: User should be the owner/operator of the vehicle having accessibility to the vehicle

REQ-2: User should navigate through the menu of the vehicle’s tablet and select the auto pilot feature

## Auto-Brake

4.2.1 Description and Priority

Decelerating approaching the speed of 0 depending on the distance relationship from the front of the vehicle to the object in front of the vehicle

Priority: High

Benefit: 9

Cost: 3

Penalty: 4

Risk: 3

4.2.2 Stimulus/Response Sequences

User Action: User selects on the auto drive application

System Response: From the moment auto drive is selected, the vehicle will take over all steering actions taking in the environment information to determine what actions need to be taken.

4.2.3 Functional Requirements

REQ-1: User should be the owner/operator of the vehicle having accessibility to the vehicle

REQ-2: User should navigate through the menu of the vehicle’s tablet and select the auto pilot feature

* 1. **Traffic Light / Stop Sign Check**

4.3.1 Description and Priority

The communication between the environment checking from sensor input if a traffic light is red, yellow, or green or if approaching a stop sign. Based the check will auto-brake or maintain speed.

Priority: Medium

Benefit: 9

Cost: 5

Penalty: 7

Risk: 2

4.3.2 Stimulus/Response Sequences

User Action: User must not interfere when car is checking for the traffic light / stop sign

System Response: The vehicle will determine based on the sensor indication if it’s green to continue forward and maintain proper speed, if yellow to auto-brake and slow down, and if it’s red or a stop sign and come to a complete stop and wait until either the light turns green or it’s the vehicles turn to begin driving again.

4.3.3 Functional Requirements

REQ-1: User should be the owner/operator of the vehicle having accessibility to the vehicle

REQ-2: User should navigate through the menu of the vehicle’s tablet and select the auto pilot feature

* 1. **Location / Route Management**

4.4.1 Description and Priority

Gives the vehicles location on a map in real time while giving the best routes when wanting to go from one location to another.

Priority: low

Benefit: 7

Cost: 2

Penalty: 1

Risk: 1

4.4.2 Stimulus/Response Sequences

User Action: User selects turns on and begins using the vehicle

System Response: The vehicle will start and the system will be online

User Action: User selects the settings application from the built in screen tablet

System Response: The system will open up its settings where the location indicator will be displayed

User Action: User selects to turn on location at all times

System Response: The vehicle will use the information from the vehicle and display the current location in real-time.

4.4.3 Functional Requirements

REQ-1: User should be the owner/operator of the vehicle having accessibility to the vehicle

REQ-2: User must go into the vehicle’s settings and allow the vehicle’s location to be shown in real-time.

* 1. **Speed Management**

4.5.1 Description and Priority

The user may set a speed limit manually before engaging in auto pilot or the vehicle will be able to determine the speed limit based on location and type of area it’s being driven in

Priority: High

Benefit: 9

Cost: 5

Penalty: 5

Risk: 5

4.5.2 Stimulus/Response Sequences

User Action: User selects on the auto drive application

System Response: Based on the type of area road the vehicle is located at, the car should automatically base its speed off the speed limitations

User Action: User selects auto pilot speed management

System Response: User may set a maximum speed the car can go while in auto pilot mode

4.5.3 Functional Requirements

REQ-1: User should be the owner/operator of the vehicle having accessibility to the vehicle

REQ-2: User should navigate through the menu of the vehicle’s tablet and select the auto pilot feature

REQ-3: User should navigate through the auto pilot’s features and be able to set a speed limit if necessary

* 1. **Auto-Park**

4.6.1 Description and Priority

When arriving at the user’s chosen destination, the vehicle will turn and align itself based on its surroundings given by the sensors to properly put itself in position to park.

Priority: Low

Benefit: 7

Cost: 3

Penalty: 2

Risk: 1

4.6.2 Stimulus/Response Sequences

User Action: User selects on the auto drive application

System Response: Vehicle will take the sensor input data to determine the wheel adjustment, speed adjustment and position itself to park

4.6.3 Functional Requirements

REQ-1: User should be the owner/operator of the vehicle having accessibility to the vehicle

REQ-2: User should navigate through the menu of the vehicle’s tablet and select the auto pilot feature

# Other Nonfunctional Requirements

## Performance Requirements

Rigorous testing, redundancy in critical systems, and continuous improvement of technology are essential to mitigate technical conflicts. The constant checking of the sensors within and outside the car as well as its communications between the sensors to the car will elevate the experience and bring reassuring performance. Performances include lane management, traffic light change indicator, route management, acceleration control, speed maintenance, brake control, and pedestrian watch. The performance should be available and reliable 24/7 as the user inputs the use of the self driving application

| Non-Functional Req | Weight | Self Driving - Tesla Y | Manual |
| --- | --- | --- | --- |
| Response time to other cars & traffic lights | 0.10 | 0.20 | 0.30 |
| Route / Lane Management | 0.40 | 0.70 | 0.70 |
| Priority of vehicles Vehicles | 0.50 | 0.90 | 0.70 |
| Total | 1 | 0.75 | 0.66 |

## Safety Requirements

Establishing clear liability frameworks and insurance policies to address these issues is essential. Government regulations and industry standards can play a role in shaping liability rules. Though the operator may not by physically driving the car, he or she is required to have a driver’s license. Safeguards include emergency breakage at a certain distance between car and objects in front and multiple cameras from the front, back and both sides of the vehicle to ensure safety between the vehicle and the surrounding environment including cars, pedestrians, and buildings.

## Security Requirements

Implementing robust cybersecurity measures and adhering to data protection laws are crucial for addressing privacy and security conflicts. The feature to report feedback or report or a crash in real time allows the vehicle to adjust while being protected against potential hackers.

## Software Quality Attributes

With the use of software, the possibility of software updates to give users improved versions on a basis will allow an increase in the products adaptability, availability, correctness, flexibility and maintainability. With consistent checks between all products in use, there will be a constant inflow of new information from those who use the product. The vehicle will have a guide built in to show those who are operating the product how to use it and a list of contact information to further ask if needed.

5.4.1 Adaptability: Users will be able give feedback based on experience

5.4.2 Availability: The application will be available to users 24/7.

5.4.3 Correctness: The application should meet its functional requirements.

5.4.4 Flexibility: Changes and modifications will be tested and run over and used to improve the application

5.4.5 Interoperability: The software should be able to communicate with the manufacturer and developers of the product

5.4.6 Maintainability: The application should be easy to update and troubleshoot.

5.4.7 Portability: The application will be only accessible with the specific car type

5.4.8 Reliability: The application should not crash under normal operation and repeated inputs.

5.4.9 Reusability: Parts of the codebase should be reusable for future add-ons.

5.4.10 Robustness: The software should be able to handle unexpected inputs and errors without causing a crash or data loss in the system.

5.4.11 Testability: Testings will be run by use of the given feedback from multiple users

5.4.12 Usability: The software will be user friendly and easy to navigate.

## Business Rules

Governments and industry stakeholders need to work together to establish clear, consistent, and adaptable regulatory frameworks that support the development and deployment of autonomous vehicles. Though passengers won’t be driving the vehicle, it is required that the operator of the vehicle has a driver’s license. Defining ethical guidelines for self-driving car behavior and engaging in public discussions to shape these guidelines can help address conflicts. Only the owner of the vehicle will be able to operate the car and its features including the self-driving application.

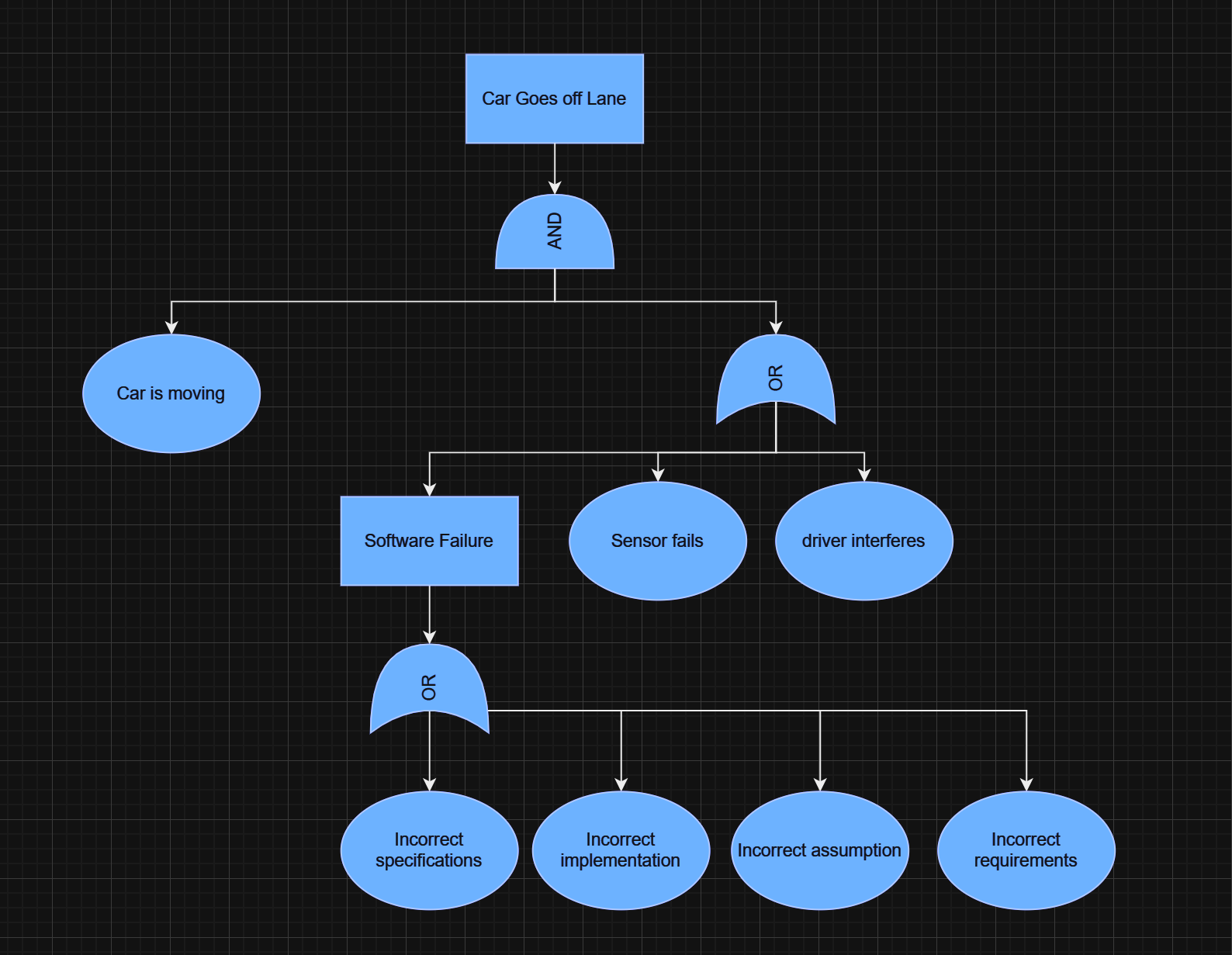
# Other Requirements

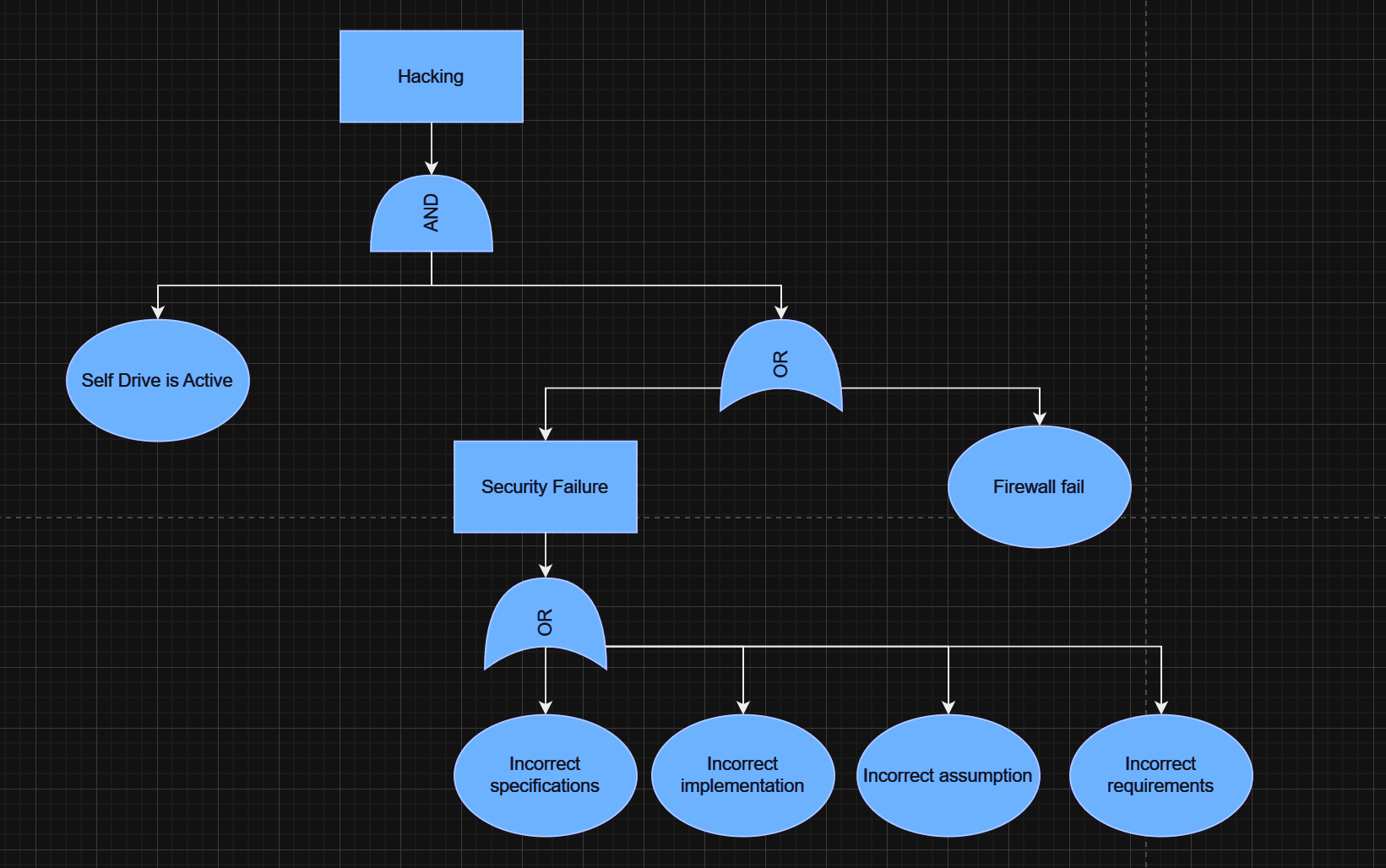
**Appendix A: Glossary**

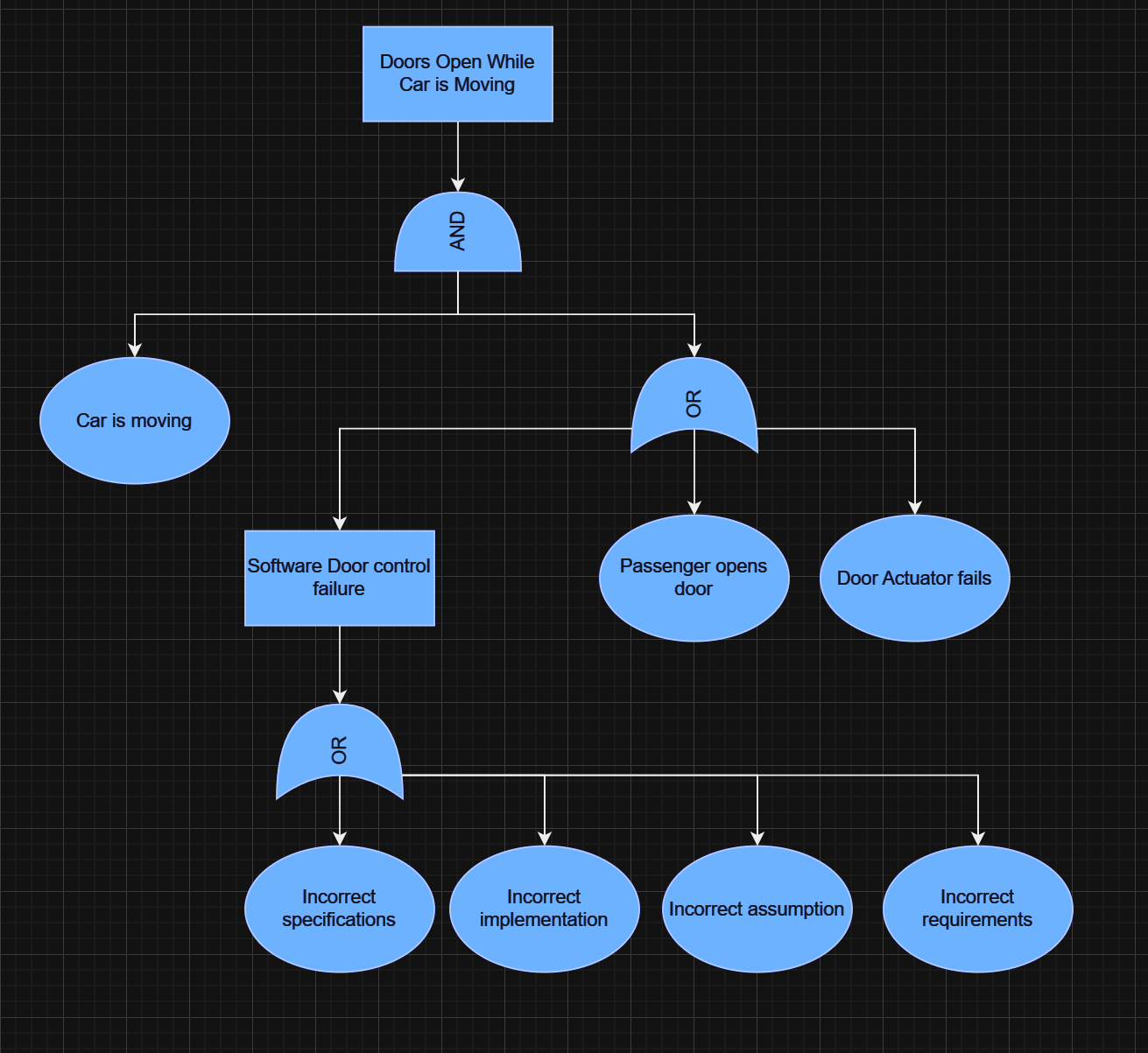
**Real time**: The actual time in which something takes place

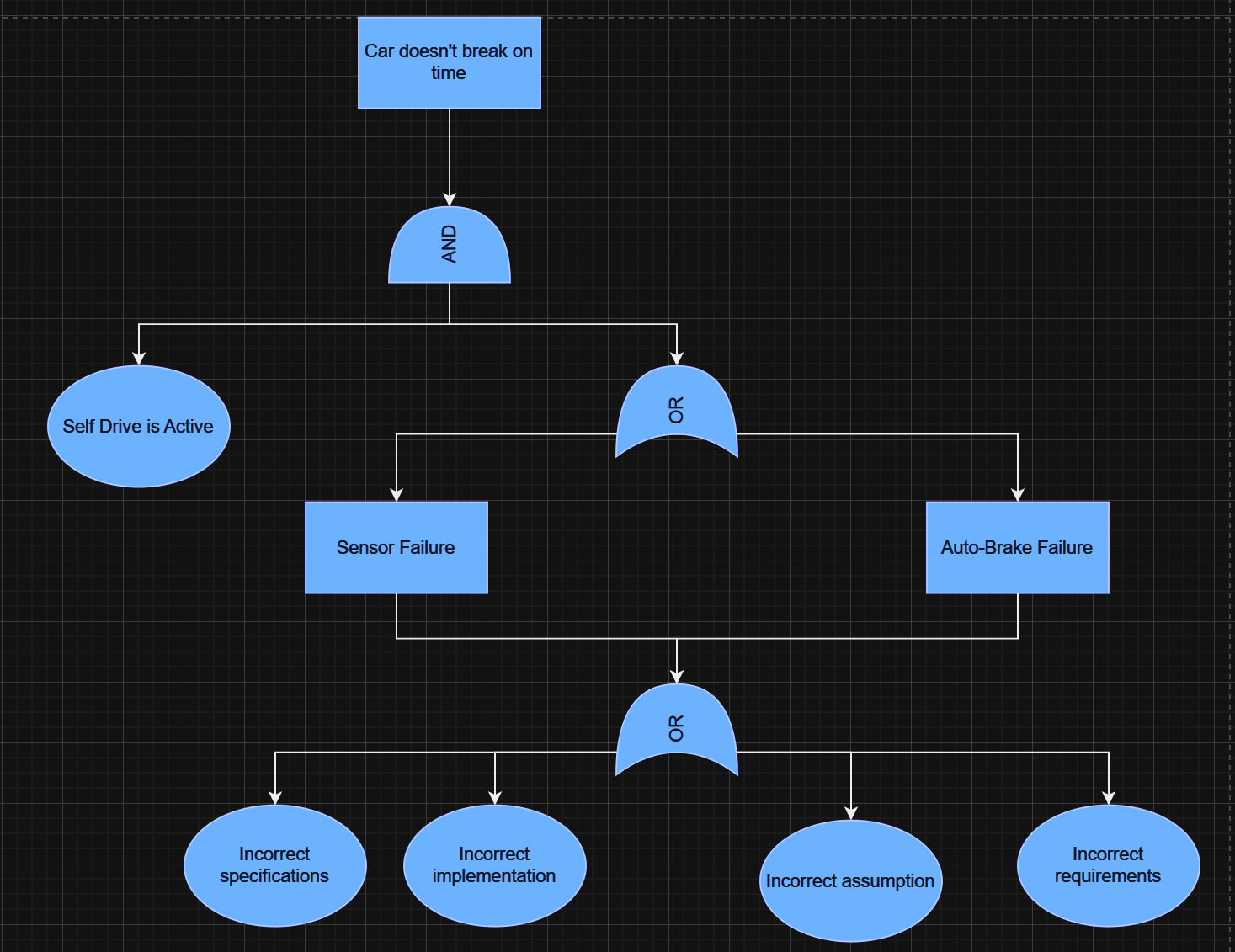
**Appendix B: Analysis Models**

**Risk Trees:**

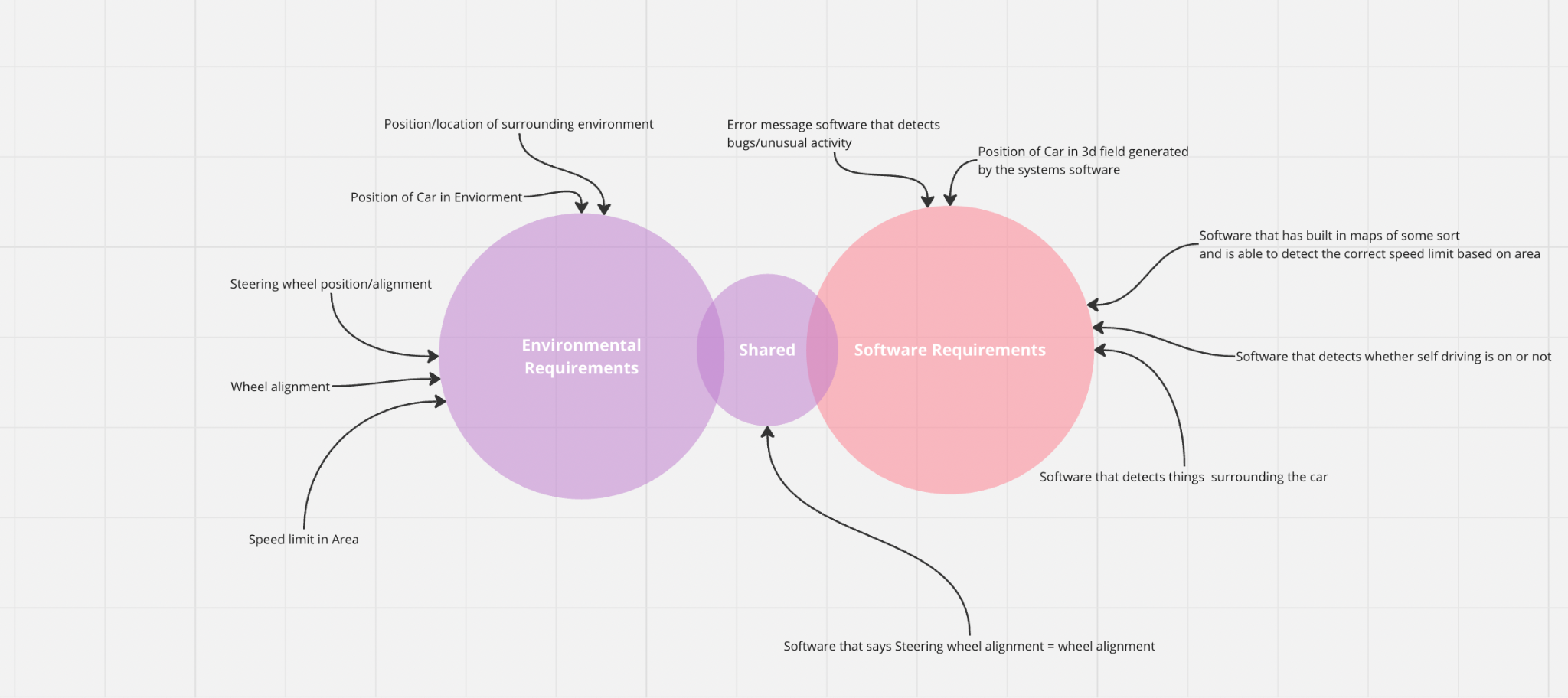








**Software-Environment Diagram**



**Appendix C: Requirements Engineering**

**1. In a paragraph, write down the objective of your project? Focus on the questions**

**“What is the criteria for our project to become successful”. Next, write down the key**

**stakeholders of your project.**

The criteria for our project to be successful is the ability for the driver of a Tesla Model Y vehicle to keep their hands off the wheel and have the car self-drive without assistance smoothly and without accidents. This system has a multitude of stakeholders such as the programming team at Tesla, Tesla shareholders, owners of a Tesla who use the autopilot system and finally passengers who will be in the car as it drives itself. This is a critical project since the lives of the public are at stake so it is essential that development of such a system prioritizes safety and quality of the feature above all else. Keeping the driver safe while also driving on its own without any accidents will be key indications of success involving this particular system.

**2. List 5 functional software requirements**

5 functional requirements can include the following, auto-direction with the steering wheel that keeps the car on the road, object detection around the vehicle that ensures no crashing, automatic gas usage so that the car remains on the speed limit without the driver having to push the gas pedal, automatic brake usage so that the vehicle will automatically begin to brake in accordance to a stop or traffic, and finally a system that detects roads, roadlines, and different terrain features so that the vehicle can stay on the road in its lane without failure.

**3. List 5 non-functional software requirements**

5 non-functional requirements can include the following, fast yet smooth responsiveness to incoming object or traffic, smooth handling of the wheel rather than sharp direction changes, system keeps car under the speed limit, easy to switch system on and off for driver, absolutely no issue or errors for detection system.

5 non functional requirements - respond to user inputs such as button clicks,text field inputs, etc, also should be up and running 24/7, protect users personal information,

**4. Are there any environmental requirements? If yes identify the system**

**requirements based on the following**

**System requirements = software requirements + domain requirements (environment)**

Environmental Requirements for a Self-Driving Car (Tesla Model Y):

1. Geographical Environment:

- The tesla’s self driving feature should be designed to operate in a wide range of environments

- It should be able to handle certain weather, including rain, snow, fog, and any other weather conditions.

- The car should adapt to different types of roads, such as paved roads, gravel roads, and dirt roads.

2. Traffic Conditions:

- The system should be able to navigate through different traffic scenarios, including heavy traffic, intersections, roundabouts, and pedestrian crossings.

- It should follow local traffic rules and regulations, including speed limits and other rules.

3. Obstacle detection:

- The self-driving car should be able to detect and respond to a wide range of obstacles, such as other vehicles, pedestrians, and objects like road signs and things of that nature.

4. Communication with surroundings:

- The system may require communication with traffic signals and other connected vehicles to increase safety.

5. Sensor Range and Accuracy:

- The car's sensors must have enough range and accuracy to be able to detect the environment around it despite whatever conditions the car may be in.

6. Mapping:

- The system needs access to accurate map data to be able to navigate correctly, for example high-definition maps and real-time map updates.

system requirements based on these environmental requirements:

System Requirements for a Self-Driving Car (Tesla Model Y):

1. Software Requirements:

- Object Recognition: The software should be able to understand data from sensors to recognize objects in the car's environment.

- Decision Making:The software must make decisions in the moment based on sensor data, traffic rules, and navigation.

- Path Planning:The system should be capable of planning safe and accurate paths for the vehicle as well as time efficient ones.

- Control System:There must be a control system to make certain controls occur like acceleration, braking, and steering.

2. Domain Requirements (Environment):

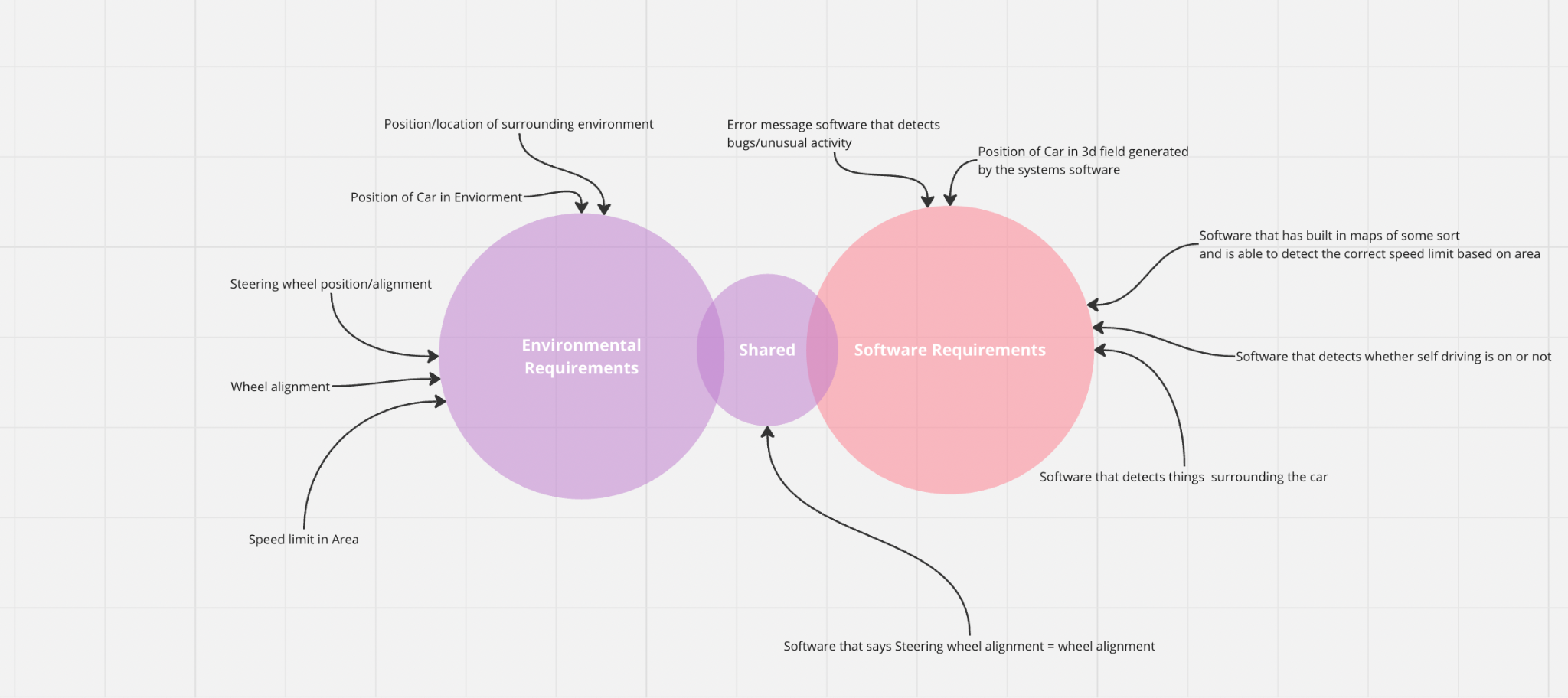
-Mapping: The system must communicate with a mapping service to get up to date maps and accurate data.

- Communication with surroundings: Implement communication to interact with other vehicles, infrastructure, and traffic light systems when needed.

- Weather Resistance: The software should include things to handle different types of weather conditions and in turn maintain safe driving conditions.

**5. Draw the software-environment diagram for the various requirements of your**

**Project**



Changes Made: Position of car in 3d field generated by systems software moves to shared area.

**6. Identify critical conflicts to your project**

Critical conflicts and challenges related to self driving cars:

**Safety:**

Conflict: Determining the liability in case of accidents involving self-driving cars can be difficult. It’s hard to pinpoint who to assign the blame to the manufacturer, the software developer, the owner, or a combination of these people ?

Resolution: Establishing clear safety and liability frameworks and insurance policies to address these issues is needed. Government regulations and industry standards can play a role in shaping liability rules.

**Ethical problems:**

Conflict: Self-driving cars may face situations where they must make ethical decisions, such as choosing between protecting the occupants or pedestrians. Resolving these moral dilemmas is challenging.

Resolution: Defining ethical guidelines for self-driving car behavior and engaging in public discussions to shape these guidelines can help address these conflicts.

**Technical Challenges:**

Conflict: Ensuring the reliability and accuracy of sensors, software, and hardware components is critical for safety. Failures in these components can lead to accidents.

Resolution: Rigorous testing, redundancy in critical systems, and continuous improvement of technology are essential to mitigate technical conflicts.

**Data Privacy and Security:**

Conflict: Self-driving cars generate vast amounts of data, including location, behavior, and sensor data. Protecting this data from cyberattacks and ensuring passenger privacy can be challenging.

Resolution: Implementing robust cybersecurity measures and adhering to data protection laws are crucial for addressing privacy and security conflicts.

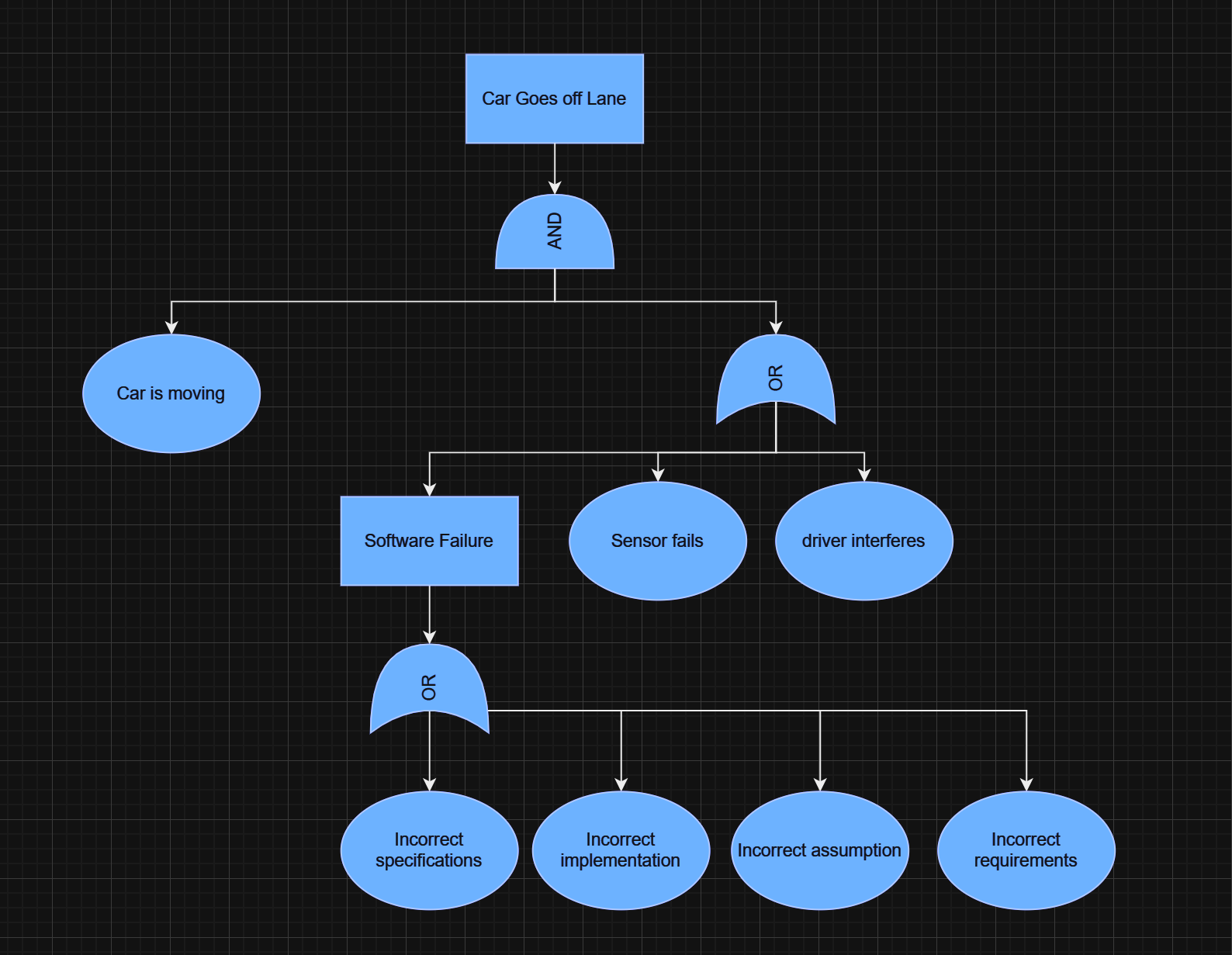
**Legal Issues:**

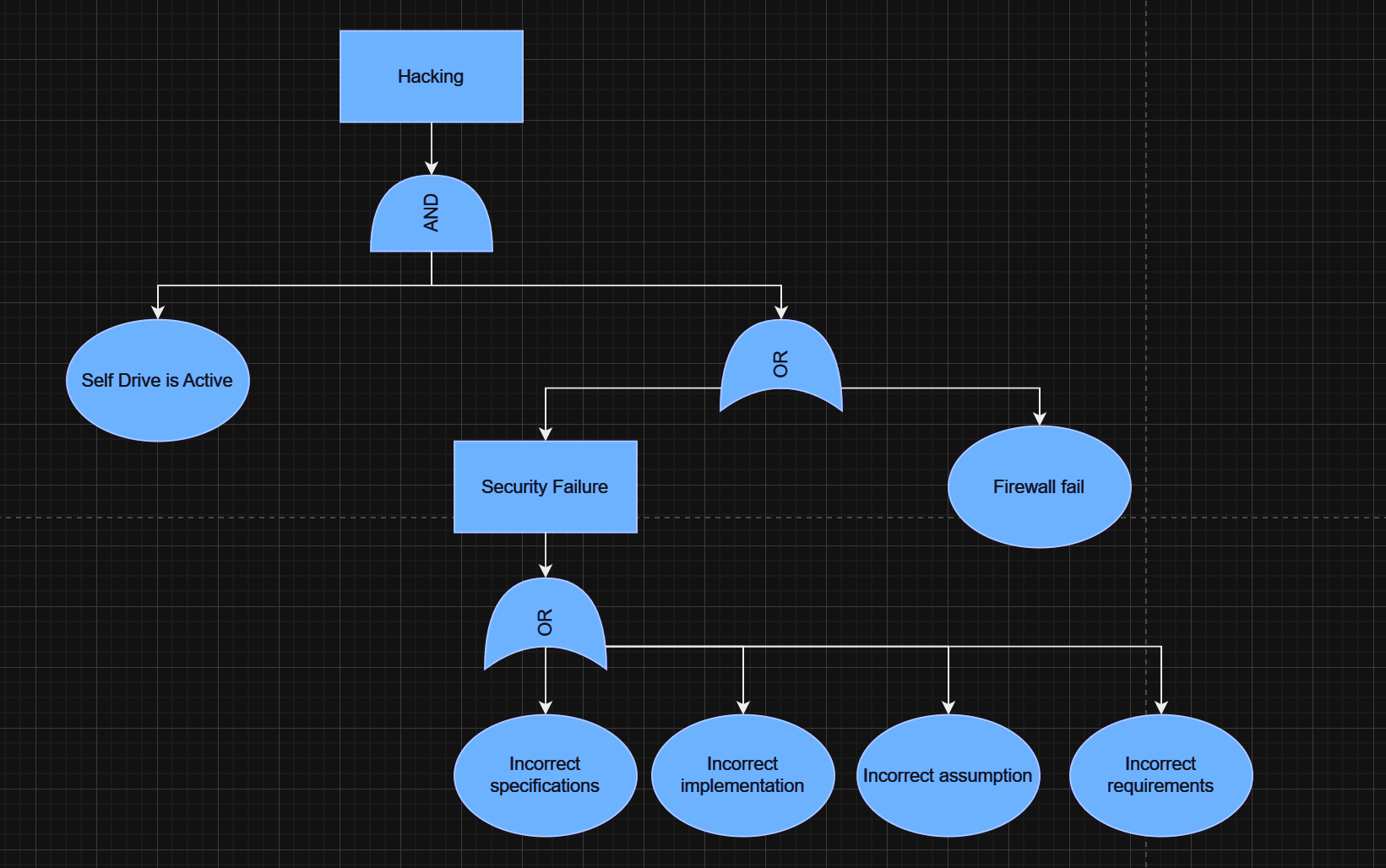
Conflict: The lack of specific/standardized regulations for self-driving cars can hinder their deployment and create uncertainty.

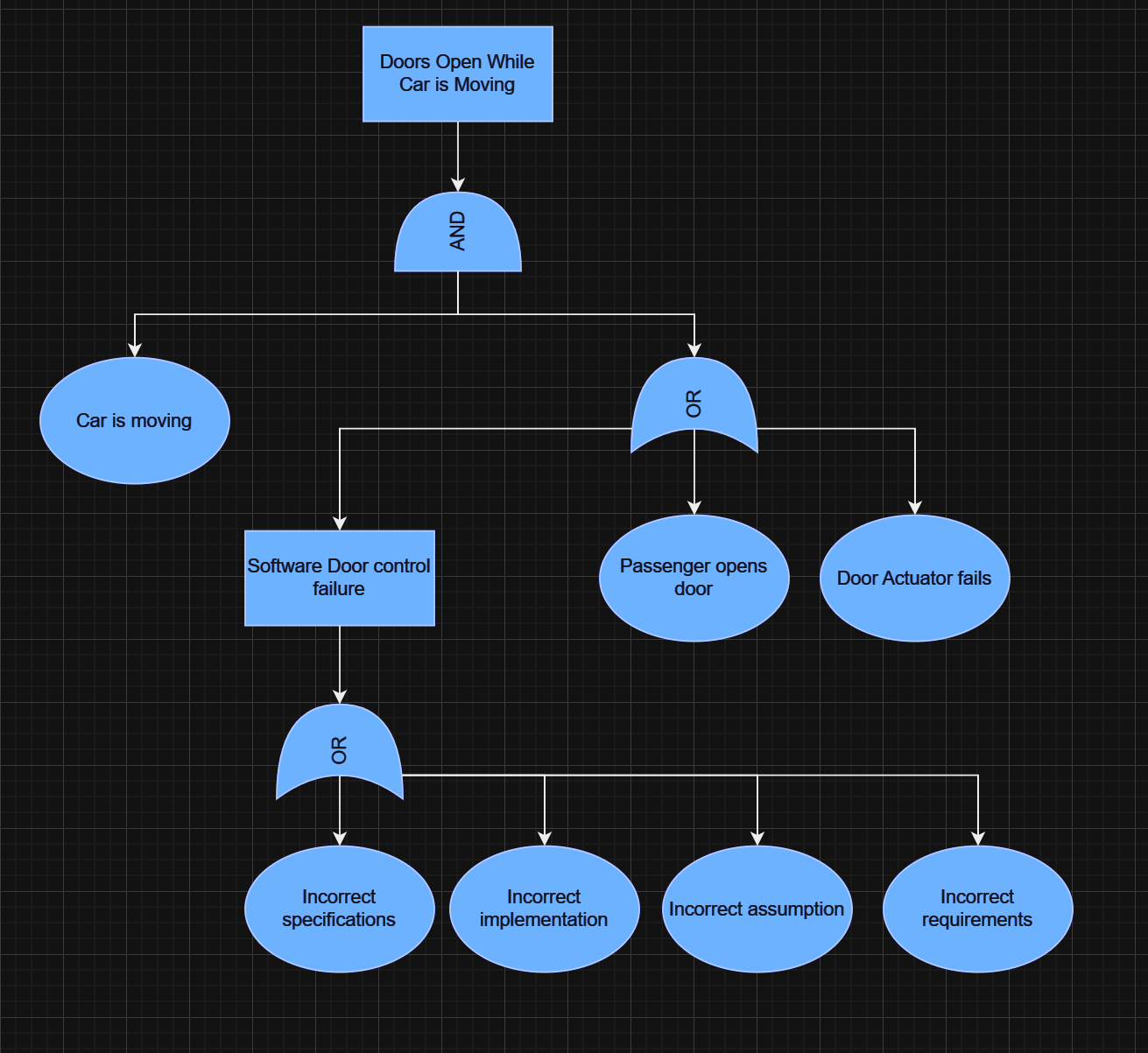
Resolution: Governments and industry stakeholders need to work together to establish clear, consistent, and adaptable regulatory frameworks that support the development and deployment of autonomous vehicles.

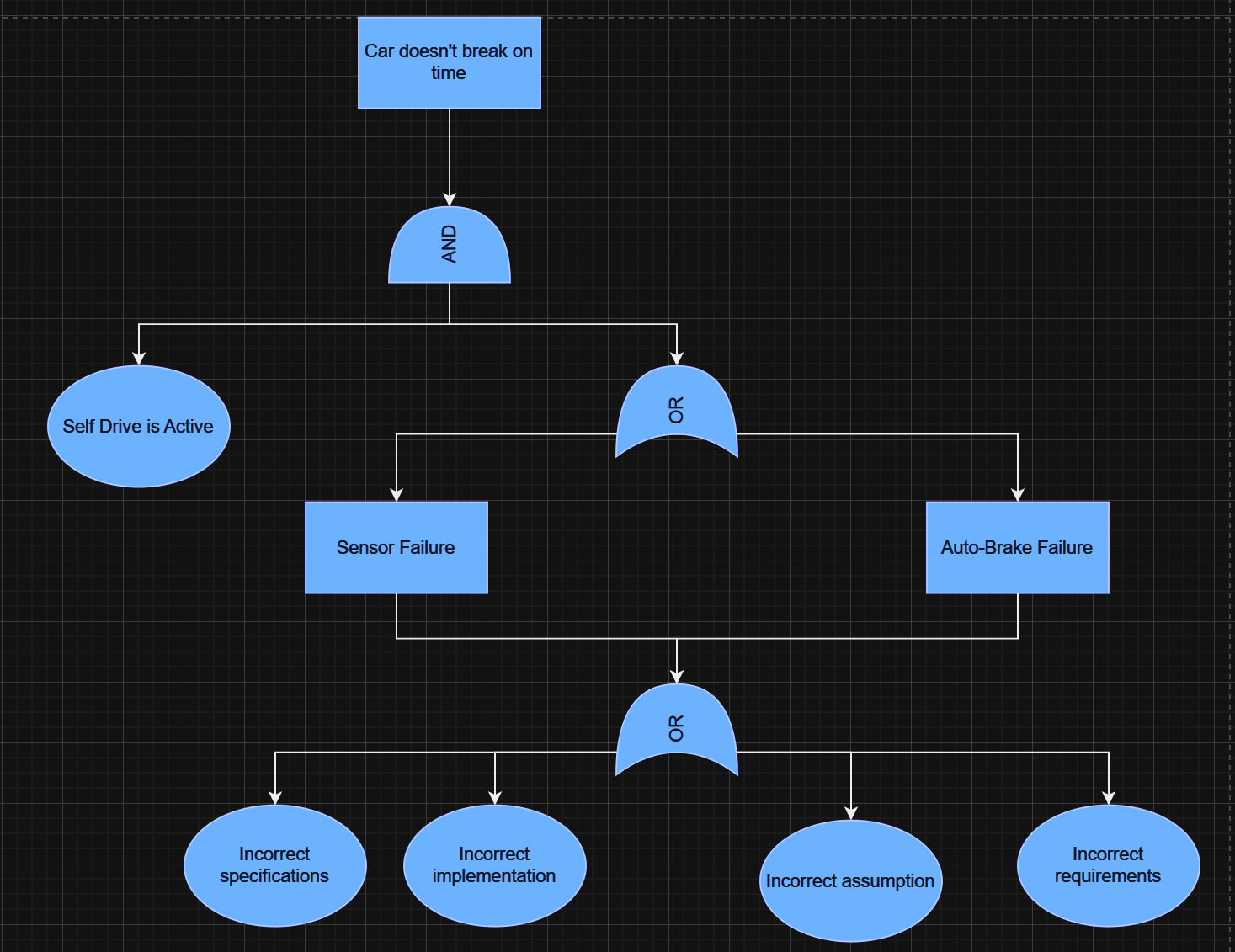
**7. Identify 4 risks related to your project and draw the risk tree for them. Also try to**

**identify the risk cut set from the tree.**









**8. Work with your team to find alternative requirements like the library case study**

**of subscribing to journals? Once found, determine the quantitative evaluation of the**

**alternatives based on at least 3 Non-Functional-Requirements. You can provide**

**weightage points and criteria points for the options yourself as a team. Complete the**

**table and state which option is better.**

| Non-Functional Req | Weight | Self Driving - Tesla Y | Manual |
| --- | --- | --- | --- |
| Response time to other cars & traffic lights | 0.10 | 0.20 | 0.30 |
| Route / Lane Management | 0.40 | 0.70 | 0.70 |
| Priority of vehicles Vehicles | 0.50 | 0.90 | 0.70 |
| Total | 1 | 0.75 | 0.66 |

**9. Consider the SE-451 project on which you are working during this semester.**

**Extract six requirements from your project that might need to be prioritized in view**

**of overall value, assign weightages for their contribution to the project relative to**

**each other, and perform requirement prioritization using the AHP process.**

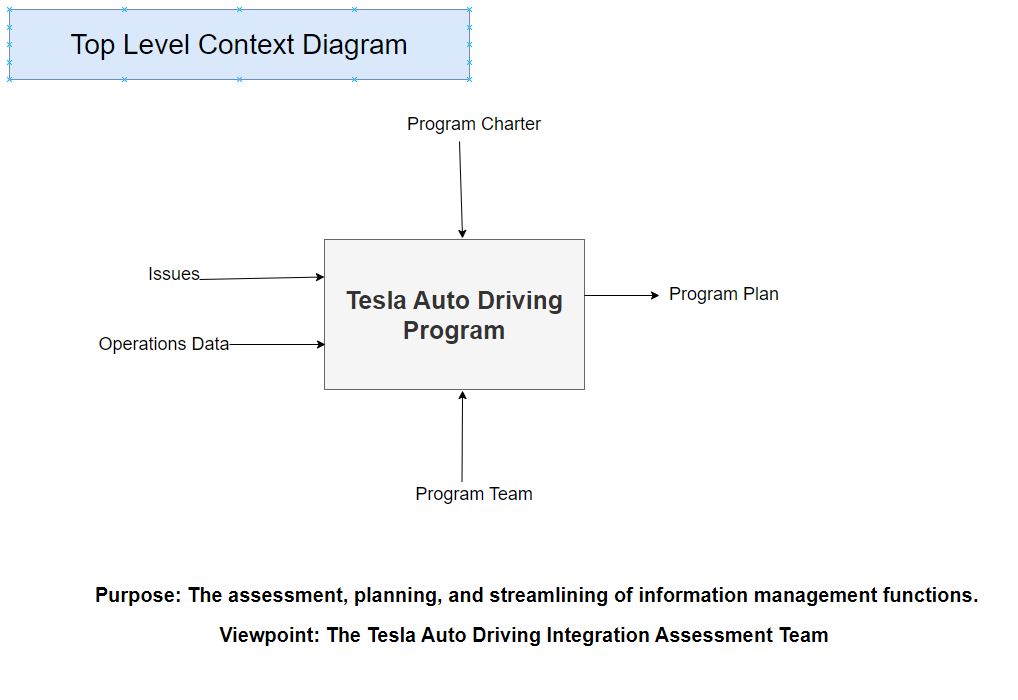
| Func Reqs | Detects traffic light changes | Priority of cars and actions | Auto-  Brake | Gives location in Real Time | Manage Car Routes | Speed Handler |
| --- | --- | --- | --- | --- | --- | --- |
| Detects traffic light changes | 1 | 2 | 1 | 9 | 5 | 1 |
| Priority of cars and actions | 0.5 | 1 | 1 | 4 | 2 | 1 |
| Auto-  Brake | 1 | 1 | 1 | 9 | 4 | 1 |
| Gives location in Real Time | .11 | .25 | .11 | 1 | .17 | .14 |
| Manage Car Routes | .2 | .5 | .25 | 6 | 1 | .14 |
| Speed Handler | 1 | 1 | 1 | 7 | 7 | 1 |

Priority Ranking -

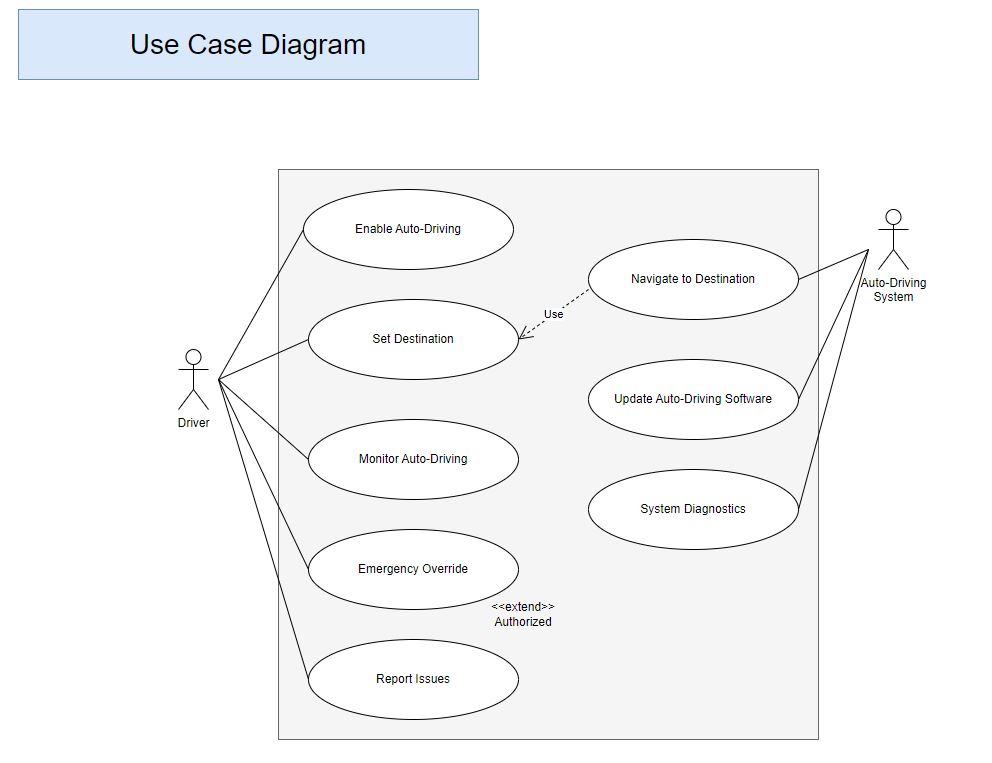
1. Detect Traffic light Changes - 26.3%
2. Speed Handler-25.2%
3. Auto Brake-22.6%
4. Priority of Cars and actions-16.1%
5. Manage Car routes- 7.2%
6. Gives location in real time-2.7%

**10. Identify the following diagrams for your project**

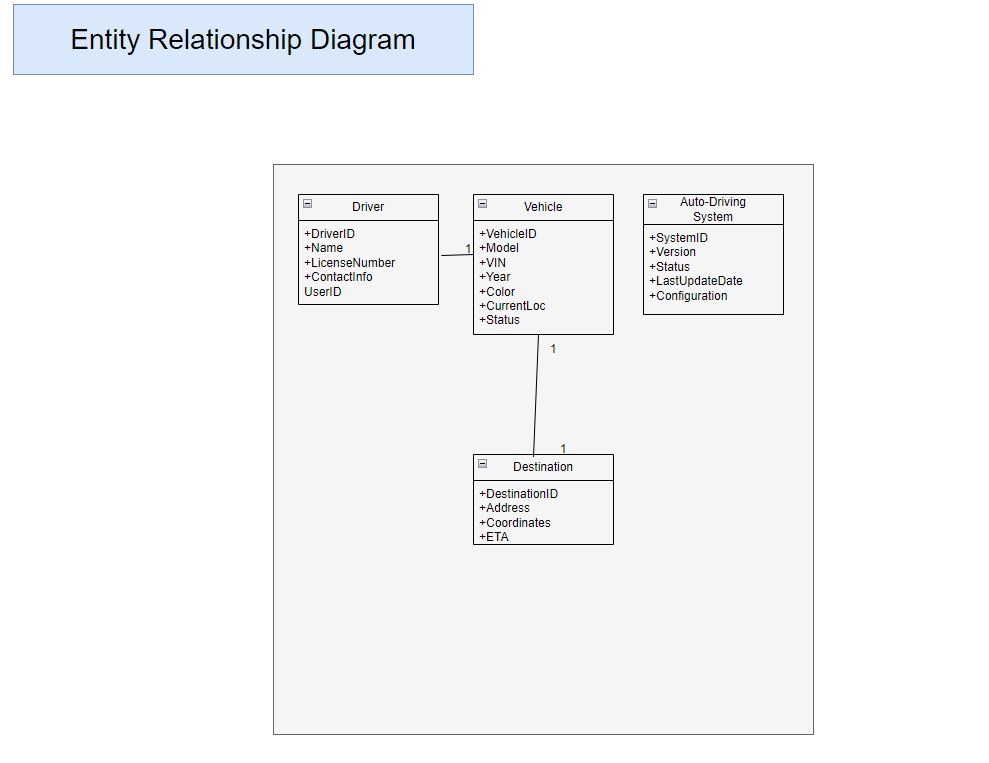
**a. Top level context diagram**

****

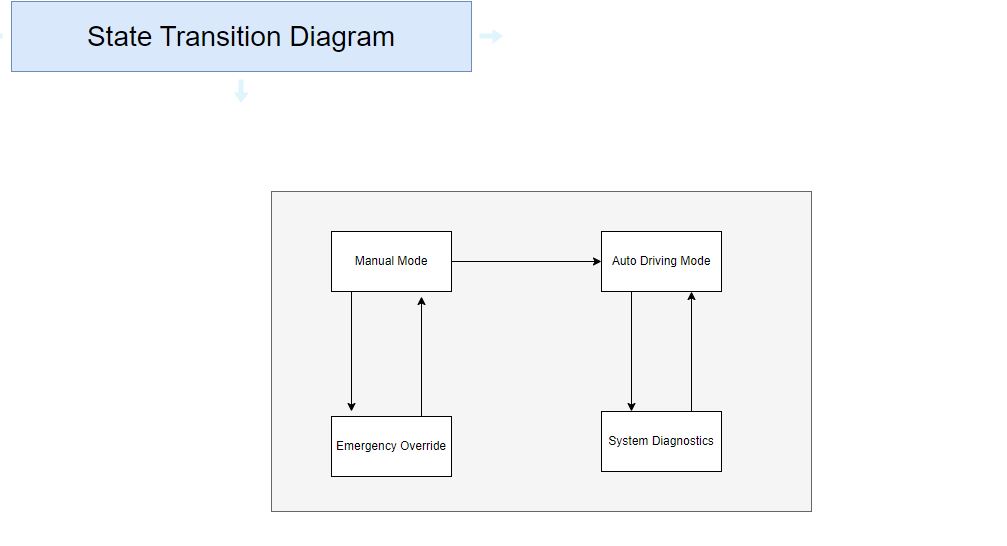
**b. Use case diagrams**

****

**c. Entity Relationship diagrams**

****

**d. State Transition diagram**

****

**11. Identify six behavioral goals and three soft goals for your project**

Behavioral Goals:

* Door closed while moving
* Brakes active when approaching objects
* Always yield to pedestrians
* Maintain proper lane position when on road
* Speed changes when in certain zones
* Activate turn signals when turning

Soft Goals:

* Minimize abrupt movement and provide a smooth course
* Optimize driving routes and vehicle operation
* User-friendly navigation for all levels of technical and physical abilities

**12. State the fit criterion for the goals identified in Question 11**

Spec: Door closed while moving

Fit Criterion: Doors should remain closed 100% of the time when the car is active and moving.

Spec: Brakes active when approaching objects

Fit Criterion: Brakes should approach the speed of zero when within 100 feet of an object.

Spec: Always yield to pedestrians

Fit Criterion: When a pedestrian detected car should be alerted and adjusted accordingly to the distance between vehicle and pedestrian.

Spec: Maintain proper lane position when on road

Fit Criterion: Steadiness occurring when at a constant speed errors made no more than 3%.

Spec: Speed changes when in certain zones

Fit Criterion: The determined location and speed limit zone corresponding with the vehicle in real time, changes happen in less than 3 seconds.

Spec: Activate turn signals when turning

Fit Criterion: Turn signals flick on when needed within 5 seconds of changing lanes or turning.

Spec: Minimize abrupt movement and provide a smooth course

Fit Criterion: Vehicle to calculate most efficient course and remain steady with no swerves or incorrect and unnecessary movements.

Spec: Optimize driving routes and vehicle operation

Fit Criterion: To take accurate turns and transitions throughout the duration of the relocation and operate with minimal instruction.

Spec: User-friendly navigation for all levels of technical and physical abilities

Fit Criterion: The graphical user interface and operation navigation shall remain consistent and unchanged without error 100% of the time.

**13. Formulate each behavioral goal in Question 11, as instantiations of Achieve or Maintain/Avoid patterns**

Maintain[DoorClosedWhenMoving]:

Always (if the car is moving then the doors are closed)

Achieve[BrakesActiveWhenApporaching]

If [object close with 100 feet] then sooner-or-later [the brakes activate reaching the speed of zero]

Achieve[AlwaysYieldForPedestrians]

If [pedestrians detected by sensors] then [the vehicle will decrease in speed and take proper action in according to the pedestrian in the most efficient and accurate course]

Maintain[ProperLanePositionOnRoad]

Always (if the car is moving then autosteer centers car within lane)

Avoid[Speeding]

Never (Goes above or below the speed limit in according to its location and the determined limit of the road/freeway)

Achieve[ActivateTurnSignalsWhenTurning]

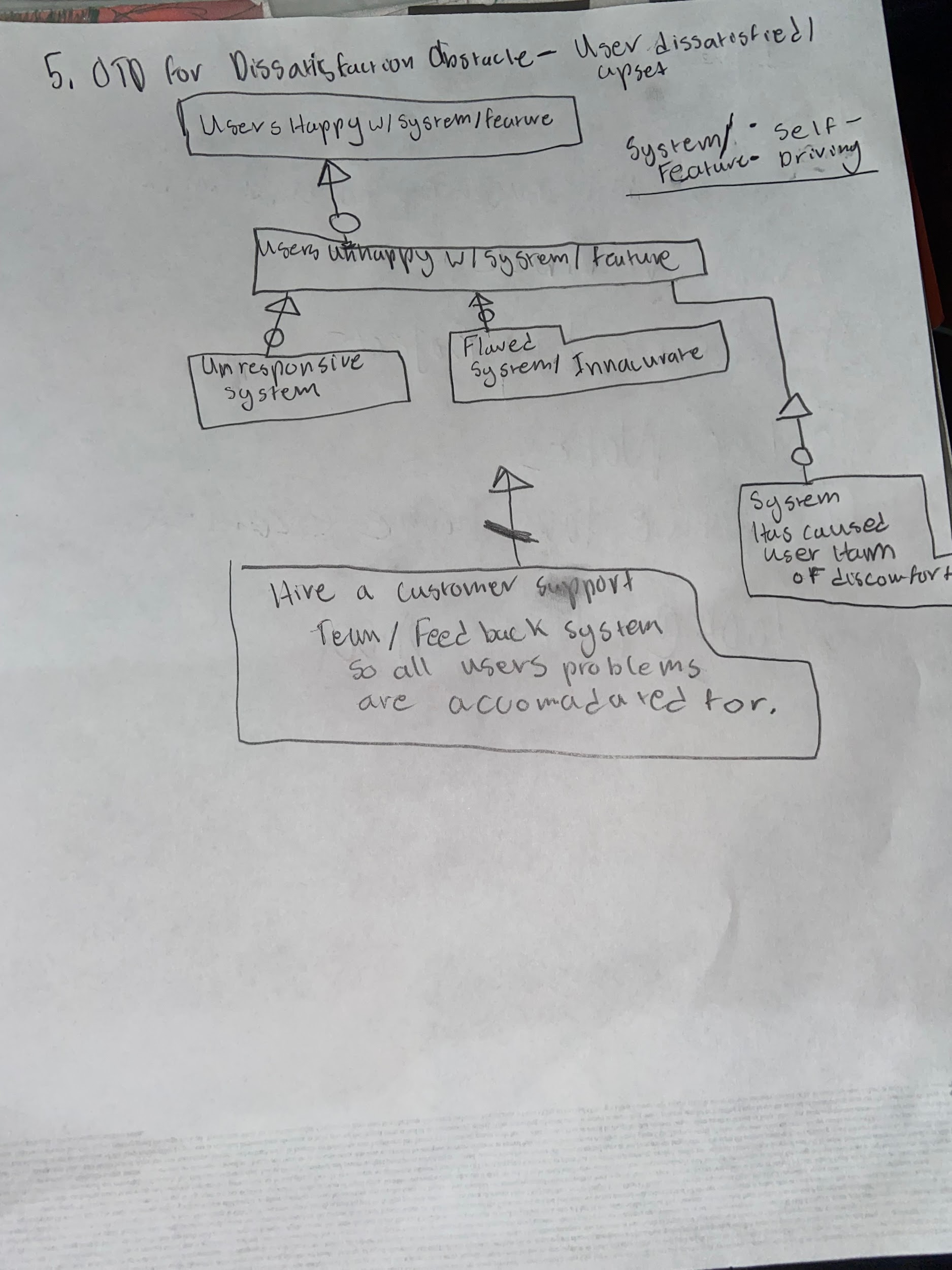
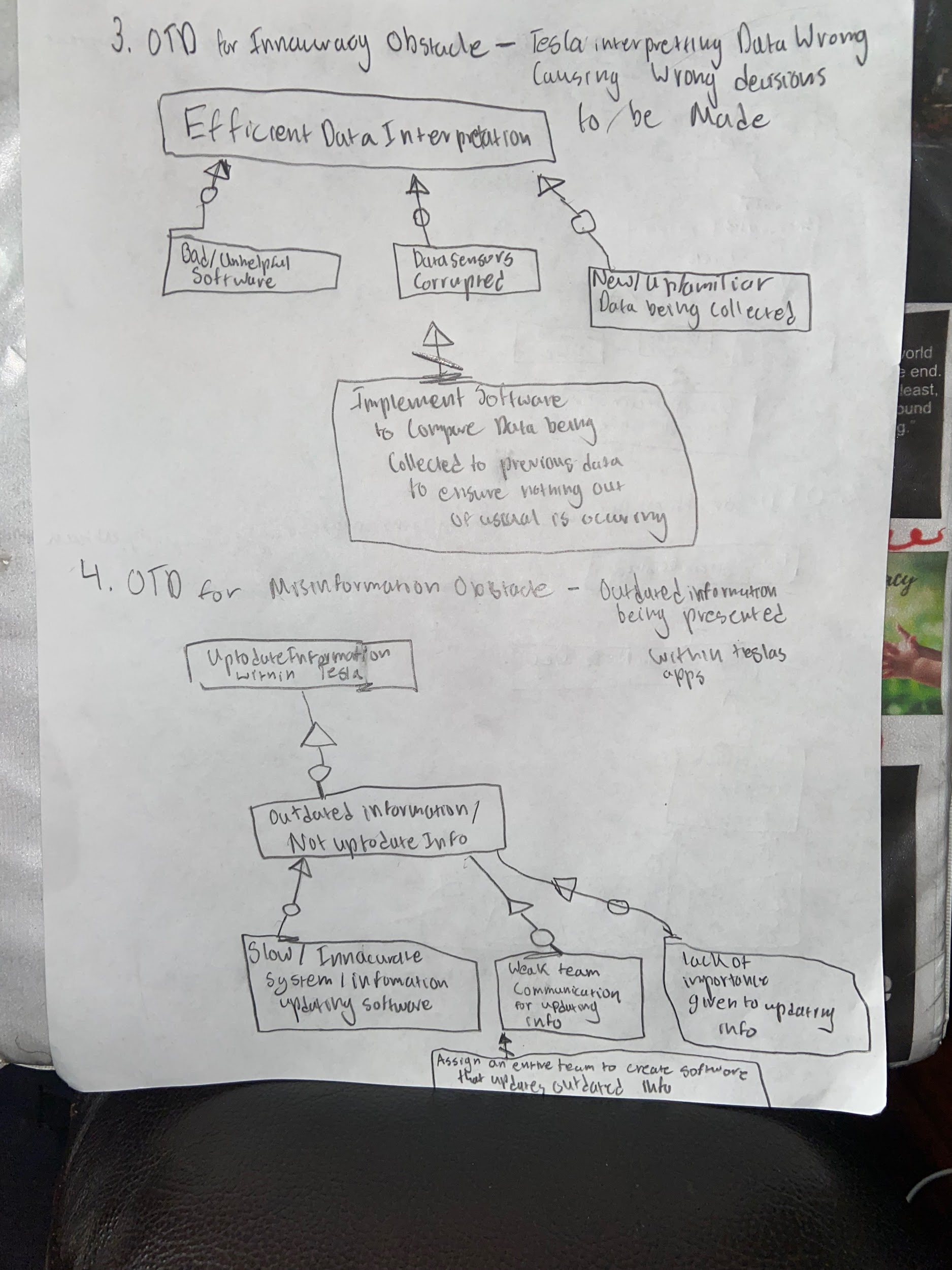
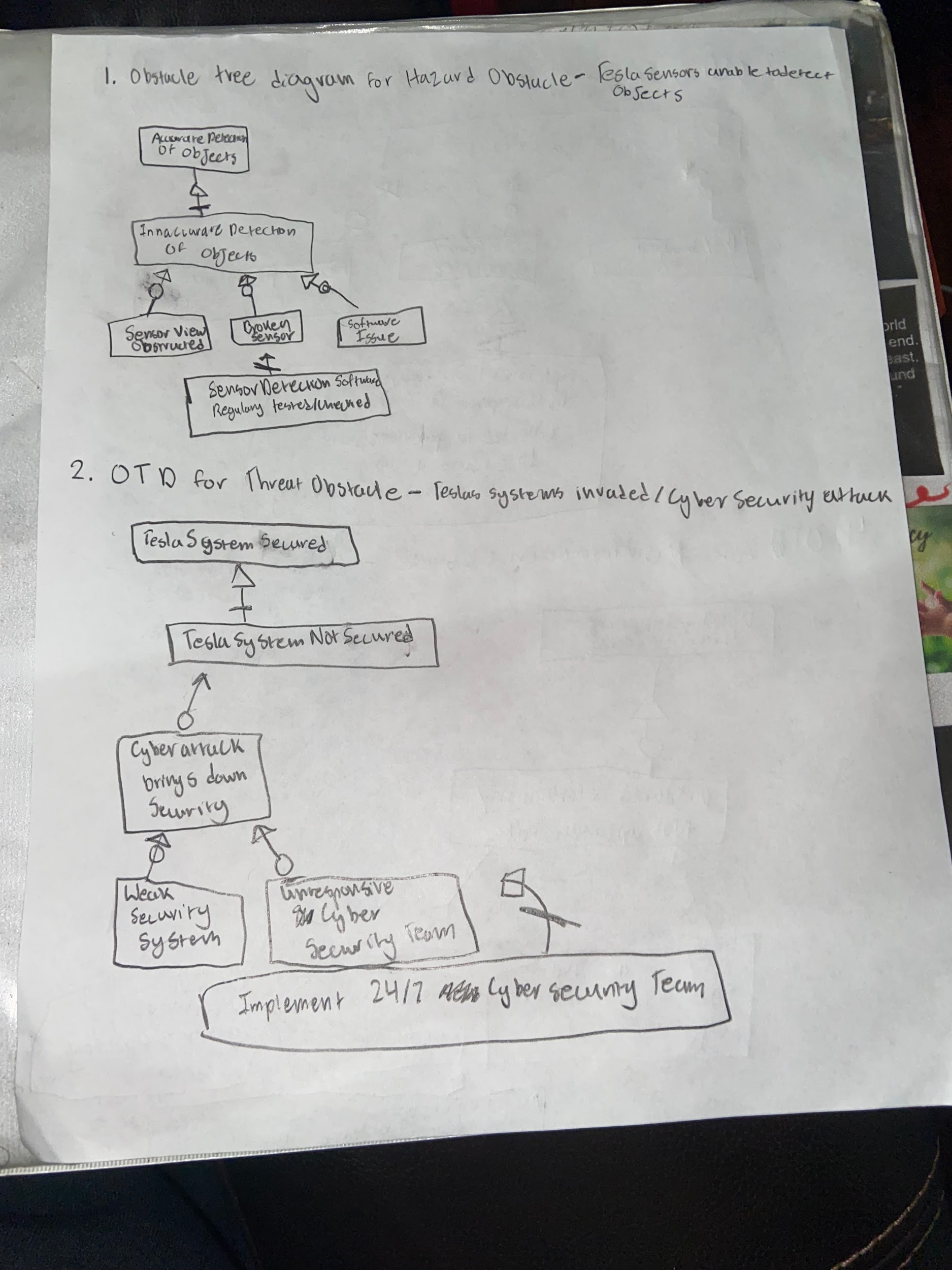
If [car approaches an intersection determined by location, route will determine and turn on the left or right signal accordingly]

**14. Identify five crucial obstacles for your project. Categorize them based on the**

**categories learned in class.**

1. Tesla being unable to detect objects on road/failure to detect (category-Hazard Obstacle which obstructs safety goal)
2. Compromise of the Teslas network/cyber attacks to it’s network or data which would affect the car's operation or affect it’s systems.(category - Threat Obstacle which obstructs Security goals)
3. Misinterpretation of data from tesla sensors data which causes the wrong decisions to be made(category- Inaccuracy Obstacle which affects Accuracy Goal)
4. Wrong/Outdated information present within teslas software whether that be outdated maps or information on things etc.(category - Misinformation obstacle which affects information goal)
5. Users getting angry with the self driving feature or any sort of discomfort from the user(category- Dissatisfaction obstacle which affects satisfaction goal)

**15. For the obstacles identified in Question 14, draw the obstacle diagrams.**

****

**16. Derive a obstacle resolution plan for the 5 obstacles in your project by adopting**

**one or more of the following strategies**

Doing 1 strategy for each of the 5 obstacles so we know how to use all of the strategies.

**a. goal substitution**

Using the threat obstacle for this strategy we can substitute the security goal for a more refined goal of higher quality security. To do this we can implement an advanced cyber security team that is available 24/7 that can constantly monitor the state of security for tesla systems. For example if you had a goal of SuspiciousActivityOccuring->NotifySecurityTeam you can substitute NotifySecurityTeam with RunCounterMeasure so that instead of waiting for a response the counter measure to find and prevent more suspicious activity is already in place.

**b. obstacle substitution**

Using the Dissatisfaction Obstacle for this strategy we can use the strategy to

Replace or substitute the obstacle itself. So in this case the obstacle is specifically user dissatisfaction with the self driving feature. To replace this make the obstacle Poor user reviews. Instead of waiting for the user to be dissatisfied, provide the user with a survey to continuously track their feelings on the product. Implement a whole team dedicated to the user experience and replace user dissatisfaction with low user feedback. So instead of users being dissatisfied being the obstacle now the obstacle is the lack of user feedback. Less user feedback either means there isn’t much wrong with the product or the users are not providing feedback. This is a better obstacle to work with than just the users being dissatisfied.

**c. obstacle prevention**

So for this strategy we are going to be using the Hazard obstacle. In this case we want to introduce a new goal in order to stay on top of obstacle prevention. So in this example if there was previously a detector for when the tesla sensors are uncalibrated: CensorUncalibrated we would improve on that with this strategy by adding a new goal called Avoid[CensorsUncalibrated].

**d. goal weakening**

For this strategy we will use the Misinformation Obstacle. The misinformation obstacle hinders the information goal and this is usually obstructed by Old information getting outdated. In order to fix this problem using this strategy we can weaken the obstructed goals formulation so it is no longer obstructed to do this instead of this: Maintain[UptoDateInformation] obstructed by InformationConstantlyOutdated you can change it to UptoDateInformation or DailyReviewofInformation. This would ensure that information isn’t constantly outdated and be up to date on a day to day basis.

**e. obstacle reduction**

So for the final strategy of obstacle reduction we will focus on the inaccuracy obstacle. This obstacle is caused by Tesla's sensors inaccurately interpreting data that it is taking in. In order to solve this problem using this strategy we can implement an ad-hoc countermeasure. For example one such as regularCrossCheck of data input and output. This would make sure all data is accurate and not being inaccurately taken in.