Software Requirements Specification (SRS)

Adaptive Driving Beam (ADB) System

Authors: Anthony Ghaith, Matthew Vazquez, Almostafa Aalabdulrasul, Seth Neubauer, Naod Ghebredngl

Customer: Mr Jacob Rhodes, Program Manager - Electronics, Flex-N-Gate

Instructor: Dr. Betty H.C. Cheng

1 Introduction

Adaptive Driving Beam (ADB) System is used to help improve driving safety and performance when driving at night or during weather conditions where visibility isn't clear. This document describes the requirements, functionality, and use cases of this system. It begins with background information of the system, followed by the requirements. The requirements are then modeled in diagrams. Finally, a demonstration of the system shows off the intended scenarios.

1.1 Purpose

The purpose of this document is to clearly define the requirements for developing the ADB System. This document covers all of the specifications for the customer, as well as allowing for the development team to understand the requirements of the system before beginning construction.

1.2 Scope

ADB is an embedded system in a vehicle that automatically adjusts the headlight beams to maximize visibility for the driver in many situations. To do this, the ADB System utilizes the vehicle's built in vehicle and environmental detection measures to dynamically change the state of the beam. The ADB System is activated and deactivated by the driver with a button. When the ADB System is active, it analyzes the sensor data and adjusts the beam accordingly. If there are no other vehicles in front of the vehicle and weather conditions are clear, then the beam is set to maximum illumination. If there are other vehicles present or heavy weather conditions, then the beam is set to a lower illumination. The ADB System also looks at the road ahead and points the beam in the direction that the vehicle is traveling in. It accounts for changes in road curve as well as elevation to point the beam in the direction of travel.

1.3 Definitions, acronyms, and abbreviations

- Define all terms, acronyms, and abbreviations needed to understand the SRS. If this section is extensive, then move to an appendix. It is also possible to provide a link to other resources for extensive terminology explanation.

Start of your text.

- Adaptive Driving Beam (ADB)
- Environmental Detection Subsystem (EDS)
- Vehicle Positioning Subsystem (VPS)
- Human-Machine Interface (HMI)
- Cybersecurity Subsystem (CSS)

1.4 Organization

- Describe what the rest of the SRS contains
- Give the organizational structure of the SRS.

Start of your text.

2 Overall Description

The following section will describe how the ADB system functions. Section 2.1 contains the context of the product, understanding the constraints on the entire system whether that's the interface (hardware, software, user, etc…) or adaptations to the system. Section 2.2, summarizes the main functions that the software will use to perform adjusting the headlights of the driver's vehicle. Section 2.3, will describe the average user expected to use the system, things like skill level will be talked about. Section 2.4 will explain into more detail about the constraints on the system. Section 2.5 will explain the assumptions and dependencies required on the system. Section 2.6 will talk about requirements that are determined to be beyond the scope of the current project and may be addressed in future versions/releases.

2.1 Product Perspective

The ADB system provides visual clearance when driving and improves safety to the driver and to the people around them when operating a motor vehicle. Although this product is mainly software based, the system still requires components that are built onto the car. The system itself is composed of smaller subsystems that work together to help provide the driver with maximum visibility when driving.

The ADB system will be activated/deactivated with an on/off switch button located to the left of the steering wheel. When the user activates the ADB system it will begin automating the headlights system, when they turn off the system it will revert to the driver manually adjusting the headlights themselves.

The vehicle must have a camera and sensors for detecting other vehicles and environment conditions. In addition to these hardware constraints, the headlight units must be capable of dynamic adjustments such as matric LED technology, digital micro-mirror devices, and moving lenses.

As for software constraints, the ADB system must have software to allow communication between the other subsystems (Beam Control, Environmental Detection, Vehicle Positioning, Human-Machine Interface, Cybersecurity). In addition to communication throughout the system there must be software that will help connect the ADB system to the hardware to maintain real time calculations, allowing for maximum visibility for the driver at all times.

With the system primarily being software related, memory constraints become an issue, the system embedded into the vehicle must have enough memory to maintain the safety critical system to allow for constant safety of the driver and the actors within the environment.

2.2 Product Functions

Beam Control Function

Given road conditions, other objects, and weather conditions, will dynamically adjust vehicle headlights in order to maximize visibility while minimizing glare for other vehicles.

Environmental Detection Function

Uses various cameras and sensors to detect various weather conditions and other objects to help position beams.

Vehicle Positioning Function

Uses GPS and sensors to determine road conditions and layout, further maximizing input to beams for maximized visibility.

Human-Machine Interface (HMI) Function

Shows the user various ADB system metrics, in order to allow the user to easily, and thoroughly, understand the status of the ADB system.

Cybersecurity Function

Constantly monitors the ADB system for any faults or vulnerabilities in order to keep the system functioning and user protected.

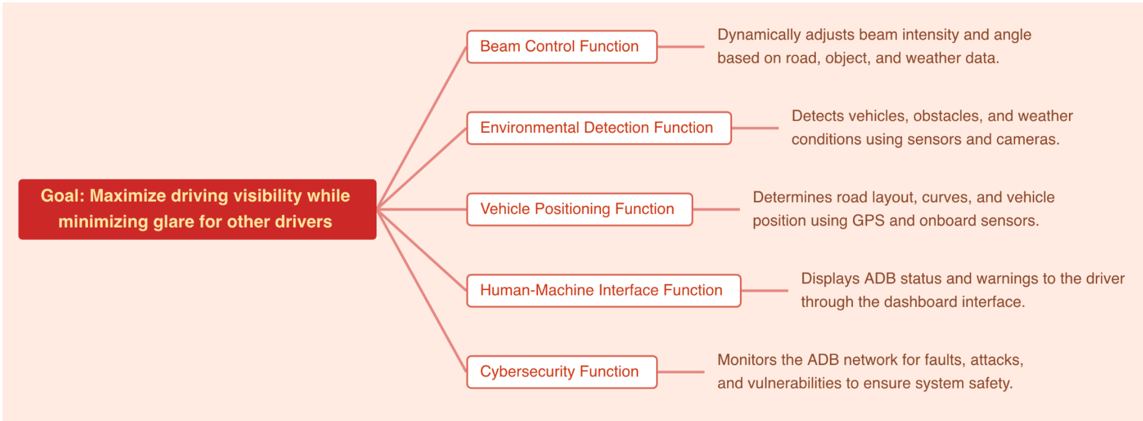

Figure 1: Goal Diagram

Figure 1 showcases the goal of the ADB system, along with the functions our system implements in order to achieve this goal.

2.3 User Characteristics

The primary users of the ADB system are licensed drivers operating vehicles equipped and safety tested with the ADB system installed. The system needs to be activated with a button, but no further input is required from the driver. The system expects the driver to stay on the road, in the right lane, following all laws and regulations of operating a vehicle.

2.4 Constraints

Since the Adaptive Driving Beam (ADB) system controls the vehicle's headlights and affects visibility, there are several safety-critical constraints. Firstly, the system must be able to fail safely by reverting to a standard low-beam mode if any major component, such as the front-facing camera or sensors malfunctions. Communication between the ADB subsystems and the vehicle's control modules must be low latency to ensure that the headlight beam is adjusted in real time as surrounding conditions change. The ADB must also not interfere with manual lighting controls or any other existing vehicle safety systems. Furthermore, environmental factors such as fog, snow, or heavy rain can limit the accuracy of detection systems (cameras or sensors unable to detect). In such cases, the system should automatically reduce functionality or deactivate to maintain safety. The system must also adhere to all automotive lighting standards and ensure that glare to other drivers is prevented. Lastly, as a cybersecurity-aware system, the ADB must ensure that all internal and external communications are authenticated and protected from unauthorized access that could compromise system integrity or vehicle safety.

2.5 Assumptions and Dependencies

For the development of the Adaptive Driving Beam (ADB) system, it is assumed that the vehicle is equipped with functional hardware components including front-facing cameras, environmental sensors, and headlights capable of dynamic beam adjustment. The electrical and communication systems within the vehicle are assumed to be fully operational, allowing communication between the ADB and its subsystems, potentially other modules as well. The driver is expected to understand the system, particularly how to enable it, and when the system will revert to low-beam mode. Furthermore, the ADB software and firmware are assumed to be properly installed, securely updated, and free of corruption.

2.6 Apporportioning of Requirements

To prevent an unreasonable scope size for this current project, a few requirements have been determined to not be included and may be addressed in the future:

- The method of detecting other vehicles (will use vehicles built in detection)
- The method of detecting weather conditions

3 Specific Requirements

- Give an enumerated list of requirements.
- As appropriate, use a hierarchical numbering scheme.

1. Sample requirement at the top level
   1.1. Level 2 requirement example
   1.2. Another Level 2 requirement
2. Select the “Requirement” Style.
3. Separately list the Cybersecurity requirements
   3.1. Identify the threat vectors (pathways for attacks)
   3.2. Identify the threat actors (types)
   3.3. What kind of cybersecurity vulnerabilities exist for your subsystem? How can you prevent, detect, and/or mitigate?

3.1 Global Invariant Requirements

1. The ADB shall revert to low-beam mode in the event of sensor failure.
2. User must be able to deactivate ADB system by pressing On/Off button
3. The ADB’s cybersecurity subsystem should call for secure communication between subsystems, authentication protocols for system updates, and real-time threat detection mechanisms.
   a. The system must only respond to secure communication between the ADB components.
4. When there is oncoming traffic/vehicles it should change to low beam in that opposing vehicles direction
5. The ADB system should never exceed 3000 lumens

3.2 Hardware Requirements

1. The ADB shall be activated/deactivated with an On/Off button located to the left of the steering wheel
    a. When the button is pressed and the ADB is deactivated, the Environmental Detection Subsystem checks if the environment is dark enough to activate the ADB. If it is, then the ADB activates first in low beam mode
    b. When the button is pressed and the ADB is activated, then it will become deactivated
    c. If the ADB is activated and the Environmental Detection Subsystem validates that the environment is not dark enough for the ADB, then it will deactivate it

3.3 Software Requirements

1. The ADB system shall constantly monitor the current driving conditions and adjust the beam accordingly
    a. The Beam Control Subsystem controls illumination levels and direction of headlight beams
    b. The Beam Control Subsystem receives inputs from the Environmental Detection Subsystem
2. The ADB shall adjust the headlight beam's shape and direction depending on the current scenario
    a. When no other drivers within 160 meters in front of vehicle, use high beams to fully illuminate road [1]
    b. The ADB shall switch to low beam mode and set strength to 1000 lumens in the event of oncoming vehicle detection.
    c. The ADB shall switch to low beam mode and set strength to 1000 lumens in the event of trailing a vehicle by 95 meters in the same lane [1]
3. The Environmental Detection Subsystem will detect other vehicles, road signs, and environmental conditions through a combination of built in sensors and cameras
    a. Radar sensors on the front of the vehicle will detect approaching vehicles
    b. The front view camera will capture the forward facing image and detect other vehicles and road signs using deep learning
4. The Vehicle Positioning Subsystem will work with the Environmental Subsystem to map out the relative position of the vehicle
    a. The Environmental Subsystem will detect the position of other vehicles and the road
    b. The Vehicle Positioning Subsystem will take these inputs and calculate the position and direction of the vehicle relative to these other entities
5. The ADB system should constantly monitor what is in front of the vehicle
    a. The Environmental Detection Subsystem and the Vehicle Positioning Subsystem will work in conjunction to determine where the vehicle is in relation to the road and other vehicles

6. The ADB system must be capable of responding to environmental changes, such as fog, precipitation, and road curve.
    a. In the event of heavy fog, switch to low beam mode
    b. In the event of heavy rain, switch to low beam mode
    c. The Environmental Detection Subsystem will monitor the curve of the road and send messages to the Beam Control Subsystem. The Beam Control Subsystem will interpret these messages to adjust the brightness and direction of the beams.
7. The ADB shall revert to low-beam mode in the event of sensor failure.
    a. The ADB shall alert the driver of reverting to low-beam mode.
8. The Human Machine Interface (HMI) Subsystem provides feedback to the driver of the status and warnings of the ADB
    a. If all conditions are met (no oncoming traffic within 160 meters, not trailing a vehicle within 95 meters, clear weather conditions) activate high beam and set strength to 3000 lumens [2]
    b. If an error is detected in the system, the HMI will send a warning message to the vehicle's infotainment system with an audio cue
    c. If an oncoming vehicle is detected by the Environmental Detection Subsystem, the HMI will send a message to the infotainment system that low beam mode has been activated
    d. If extreme weather conditions (fog or heavy rain) are detected by the Environmental Detection Subsystem, the HMI will send a message to the infotainment system that low beam mode has been activated
    e. When the ADB On/Off button is pressed, the HMI will send a message to the infotainment system that the ADB has been activated/deactivated

3.4 Security Requirements

1. The ADB shall be equipped with a cybersecurity subsystem that should be capable of protection, monitoring, detection, and mitigation of cyber threats.
    a. The ADB’s cybersecurity subsystem should call for secure communication between subsystems, authentication protocols for system updates, and real-time threat detection mechanisms.

4 Modeling Requirements

4.1 Use Case Diagram

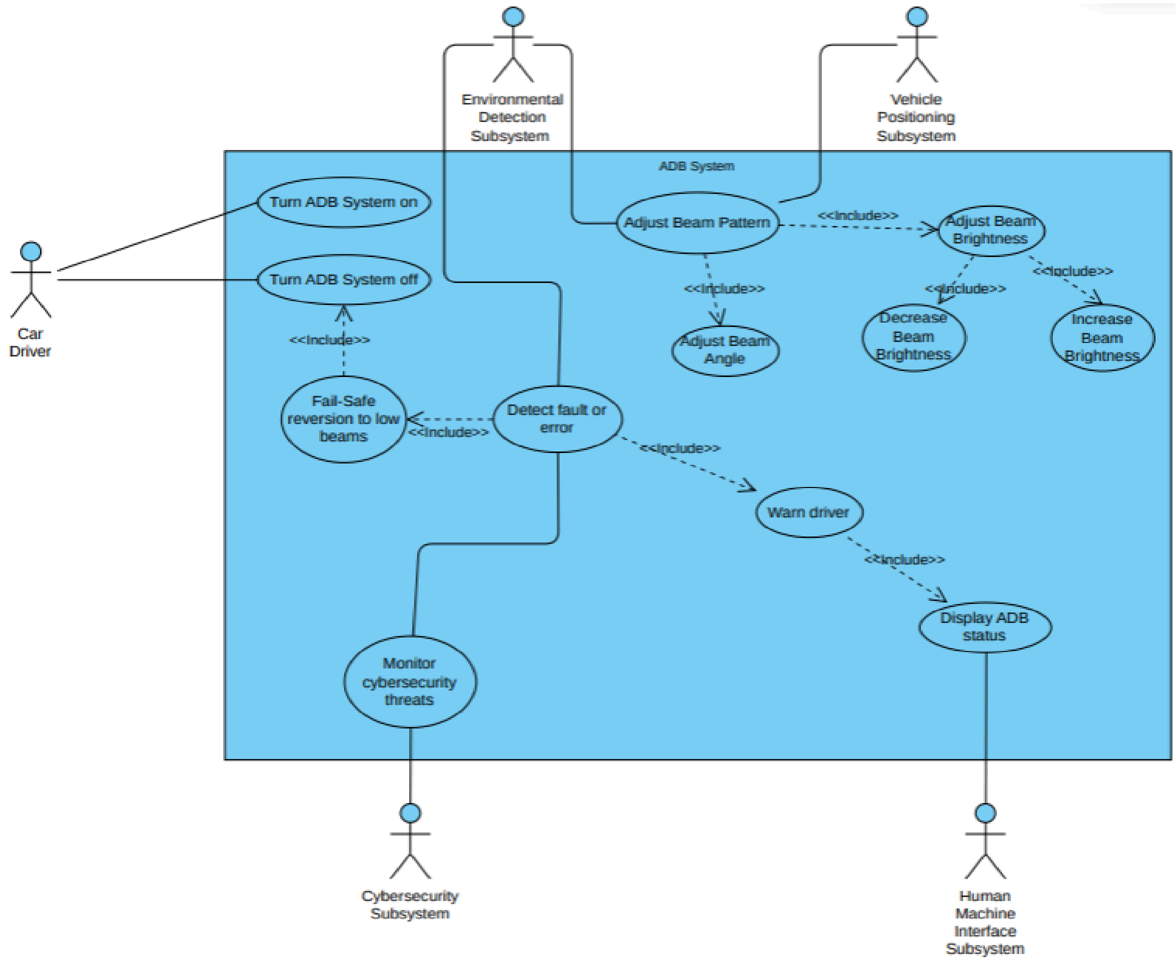

Figure 2: Use case diagram for the ADB System

Use Case:	Turn Adaptive Driving Beam System (ADB) on
Actors:	Car Driver
Description:	After the user/driver turns on the car they have the option to press a button to activate the ADB system (turn on), from here the ADB system will then engage and begin to change the beam pattern based on the situation its currently in
Type:	Primary (essential)
Includes:	-
Extends:	-
Cross-refs:	-

Use cases	-

Use Case:	Turn Adaptive Driving Beam System (ADB) off
Actors:	Car Driver
Description:	When the system runs into an issue where it struggles to calculate where other vehicles are in its current situation it will disengage (turn off) the ADB system to prevent harm to other actors in that current scenario. If there is a cyberattack the ADB will automatically turn off. Lastly, the user has the option to manually turn off the ADB system if they want to simply manually adjust the headlights.
Type:	Primary (essential)
Includes:	-
Extends:	-
Cross-refs:	-
Use cases	Fail-Safe Reversion to low beams

Use Case:	Adjust Beam Pattern
Actors:	Environmental Detection Subsystem, Vehicle Positioning Subsystem
Description:	Uses inputs from the Environmental Detection Subsystem and the Vehicle Positioning Subsystem to dynamically adjust the angle and brightness of the beam.
Type:	Primary (essential)
Includes:	Adjust Bream Brightness, Adjust Beam Angle
Extends:	-
Cross-refs:	-
Use cases	Adjust Bream Brightness, Adjust Beam Angle

Use Case:	Adjust Beam Brightness
Actors:	-
Description:	Adjust brightness of the headlight beam depending on the given information, its primary job is to increase or decrease brightness of the headlight beam
Type:	Primary (essential)

Includes:	Increase Beam Brightness, Decrease Beam Brightness
Extends:	Adjust Beam Pattern
Cross-refs:	-
Use cases	Increase Beam Brightness, Decrease Beam Brightness

Use Case:	Increase Beam Brightness
Actors:	-
Description:	Increase the brightness of the headlight beam
Type:	Primary (essential)
Includes:	-
Extends:	Adjust Beam Brightness
Cross-refs:	-
Use cases	-

Use Case:	Decrease Beam Brightness
Actors:	-
Description:	Decrease the brightness of the headlight beam
Type:	Primary (essential)
Includes:	-
Extends:	Adjust Beam Brightness
Cross-refs:	-
Use cases	-

Use Case:	Adjust Beam Angle
Actors:	-
Description:	This use case includes all beam adjustments related to angle, including minor left, right, upward, and downward corrections performed by the Beam Control Subsystem
Type:	Primary (essential)

Includes:	-
Extends:	Adjust Beam Pattern
Cross-refs:	-
Use cases	-

Use Case:	Warn driver
Actors:	-
Description:	Notification that appears on the drivers dash to alert the driver that the ADB system that something has changed?
Type:	Primary (essential)
Includes:	-
Extends:	-
Cross-refs:	-
Use cases	Display ADB Status

Use Case:	Display ADB Status
Actors:	HMI
Description:	Shows system status, including any errors that may need to warn driver.
Type:	Primary (essential)
Includes:	Warn Driver
Extends:	-
Cross-refs:	-
Use cases	-

Use Case:	Monitor Cybersecurity Threats
Actors:	Cybersecurity Subsystem
Description:	Actively monitors system to reduce/prevent cybersecurity threats to the system.
Type:	Primary (essential)
Includes:	-

Extends:	-
Cross-refs:	-
Use cases	Fail-Safe Reversion to low beam

Use Case:	Detect fault or error
Actors:	Environmental Detection subsystem
Description:	Manages faults and/or foul conditions, activating the fail-safe and warning driver of error.
Type:	Primary (essential)
Includes:	Fail-Safe Reversion to low beams, Warn Driver
Extends:	-
Cross-refs:	-
Use cases	Fail-Safe Reversion to low beams, Warn Driver

Use Case:	Fail-Safe Reversion to low beams
Actors:	
Description:	In event of system failure or adverse weather conditions, ADB system will revert to low beams and deactivate.
Type:	Primary (essential)
Includes:	Warn Driver
Extends:	-
Cross-refs:	-
Use cases	Turn Adaptive Driving Beam System (ADB) off

4.2 Domain Model

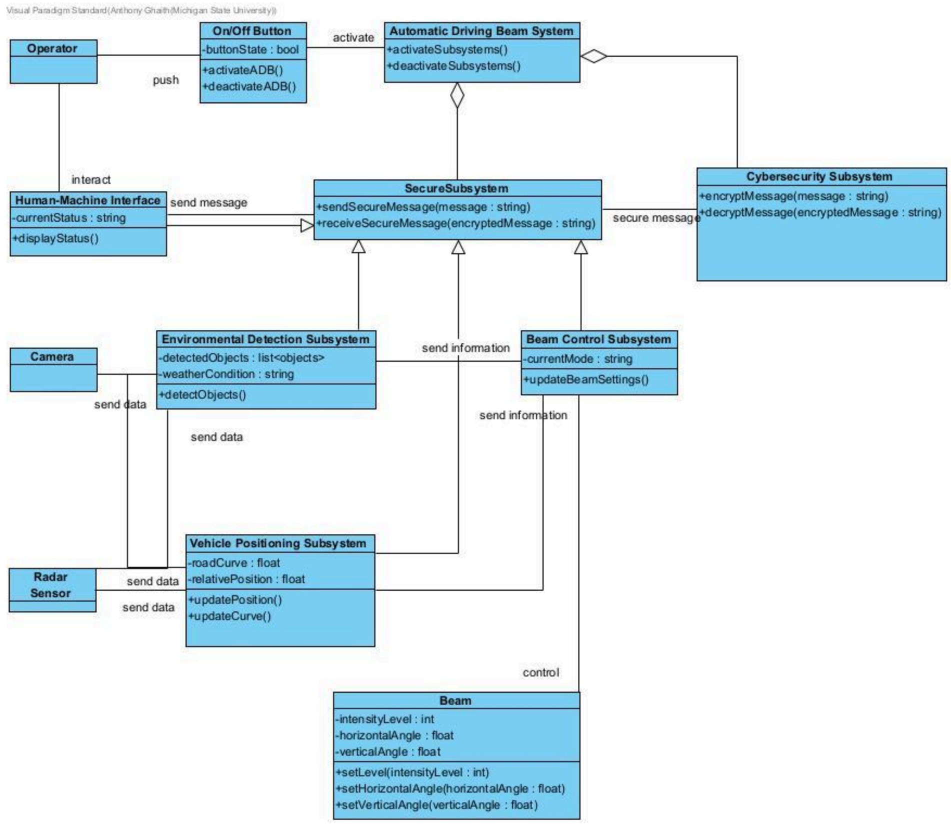

Figure 3: Domain Model for the ADB System

Element Name	**Cybersecurity Subsystem**
Description	Monitors communications and activates responses to detected security threats.
Attributes	
Operations	encryptMessage(message : string) decryptMessage(encryptedMessage : string) activateSecurityResponse()
Relationships	ADB System – Aggregation Secure Subsystem – encrypts and decrypts messages
Export Control	Public

Element Name	**Secure Subsystem**
Description	Parent class for Subsystems that make secure message transfers
Attributes	

Operations	sendSecureMessage(message : string) receiveSecureMessage(encryptedMessage : string)
Relationships	ADB System – Aggregation Cybersecurity Subsystem – sends messages to secure
Export Control	Public

Element Name	**Environmental Detection Subsystem**
Description	Detects oncoming/following vehicles and analyzes weather/environment.
Attributes	detectedObjects : list<object> weatherCondition : string
Operations	detectObject()
Relationships	ADB System – Aggregation Secure Subsystem – Inheritance Beam Control Subsystem – sends environmental data
Export Control	Public

Element Name	**Vehicle Positioning Subsystem**
Description	Determines lane, heading, and path to support beam direction decisions.
Attributes	roadCurve : float relativePosition : float
Operations	updatePosition() detectCurvature()
Relationships	ADB System – Aggregation Secure Subsystem – Inheritance Beam Control Subsystem – sends positioning data
Export Control	Public

Element Name	**Beam Control Subsystem**
Description	Determines proper beam settings based on environment and road conditions.
Attributes	currentMode : string
Operations	updateBeamSettings()

Relationships	ADB System – Aggregation Secure Subsystem – Inheritance Beam - sends control messages
Export Control	Public

Element Name	**Human Machine Interface**
Description	Interface between operator and ADB system for displaying information and capturing input.
Attributes	currentStatus : string
Operations	displayStatus()
Relationships	ADB System – Aggregation Secure Subsystem – Inheritance Operator – displays messages
Export Control	Public

Element Name	**Radar Sensor**
Description	Detects environmental information such as lighting and distance.
Attributes	sensorType : string status : boolean
Operations	GetDistanceData()
Relationships	Environmental Detection Subsystem - sends sensor data Vehicle Positioning Subsystem - sends sensor data
Export Control	Public

Element Name	**Camera**
Description	Captures and sends image data for environment analysis and detection.
Attributes	frameRate : int cameraId : string
Operations	sendImageData()
Relationships	Environmental Detection Subsystem - sends image data Vehicle Positioning Subsystem - sends image data
Export Control	Public

Element Name	**Beam**

Description	Controls light emission direction, pattern, and on/off behavior.
Attributes	intensityLevel : int horizontalAngle : float verticalAngle : float
Operations	setLevel() setHorizontalAngle() setVerticalAngle()
Relationships	Beam Control System - Receives control messages
Export Control	Public

Element Name	**On/Off Button**
Description	Enables or disables the adaptive driving beam feature.
Attributes	buttonState : boolean
Operations	activateADB() deactivateADB()
Relationships	Operator – Operator pushes to activate system ADB System – activates system
Export Control	Public

Element Name	**Operator**
Description	Human drivers who receive info, and may override ADB systems at any time.
Attributes	
Operations	
Relationships	On/Off button – Operator pushes to activate system Human Machine Interface - reads messages
Export Control	External Actor

4.3 Sequence Diagrams

Scenario 1:

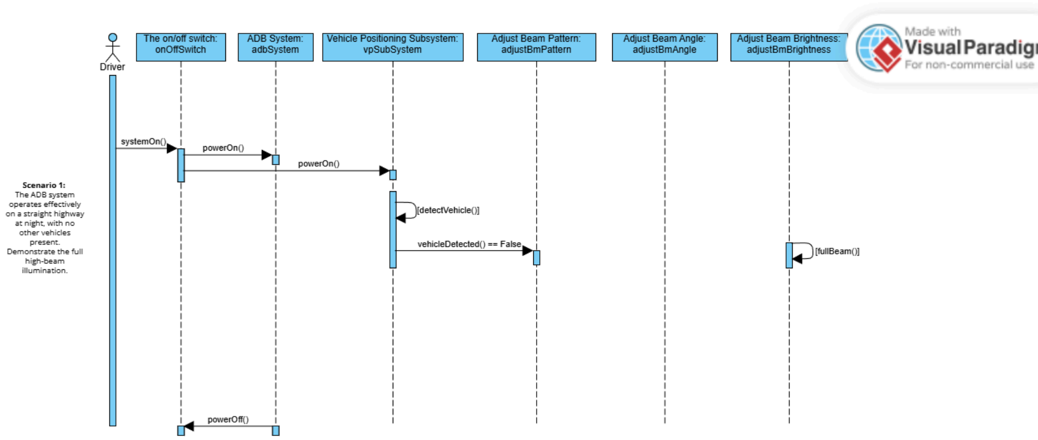

Figure 4: Sequence Diagram Scenario 1

Scenario 2:

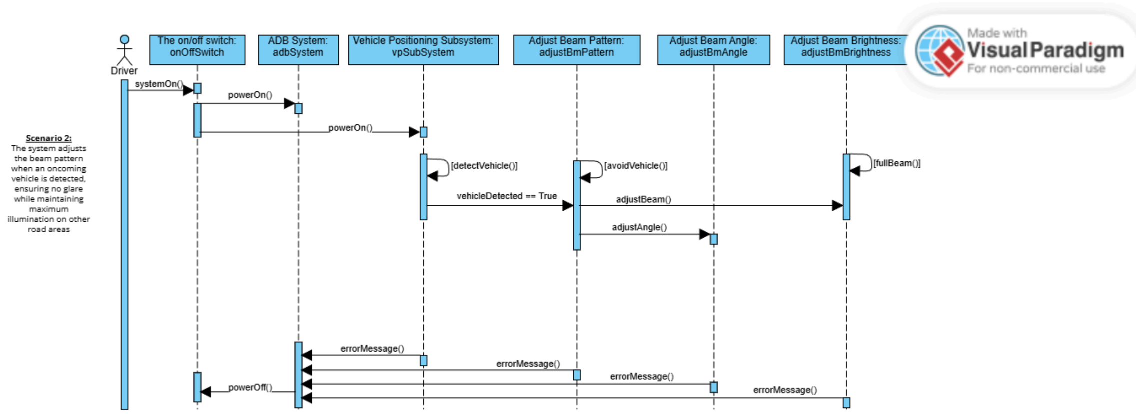

Figure 5: Sequence Diagram Scenario 2

Scenario 3:

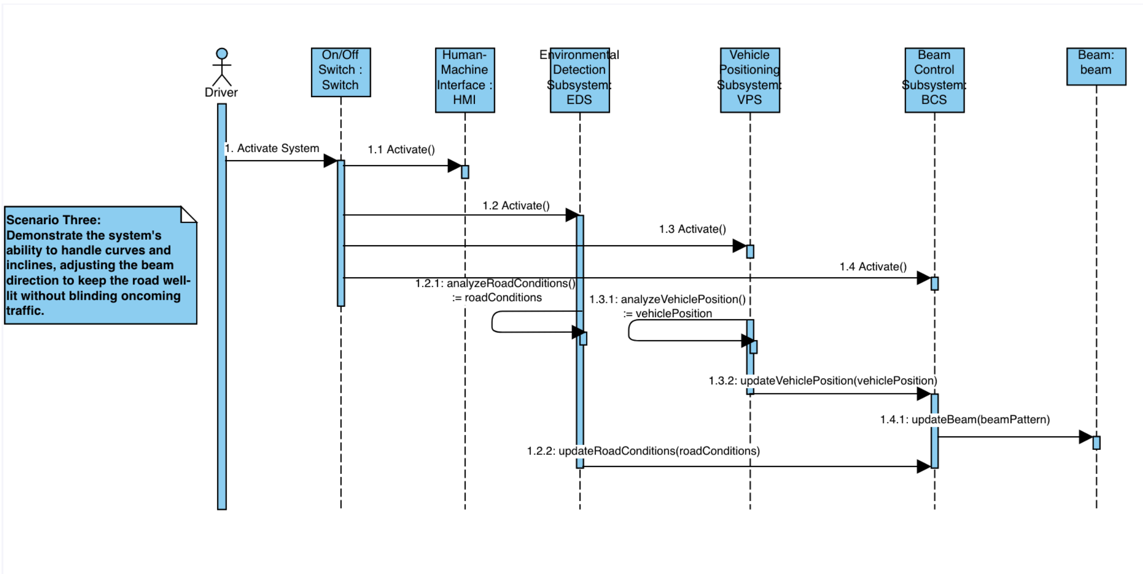

Figure 6: Sequence Diagram Scenario 3

Scenario 4:

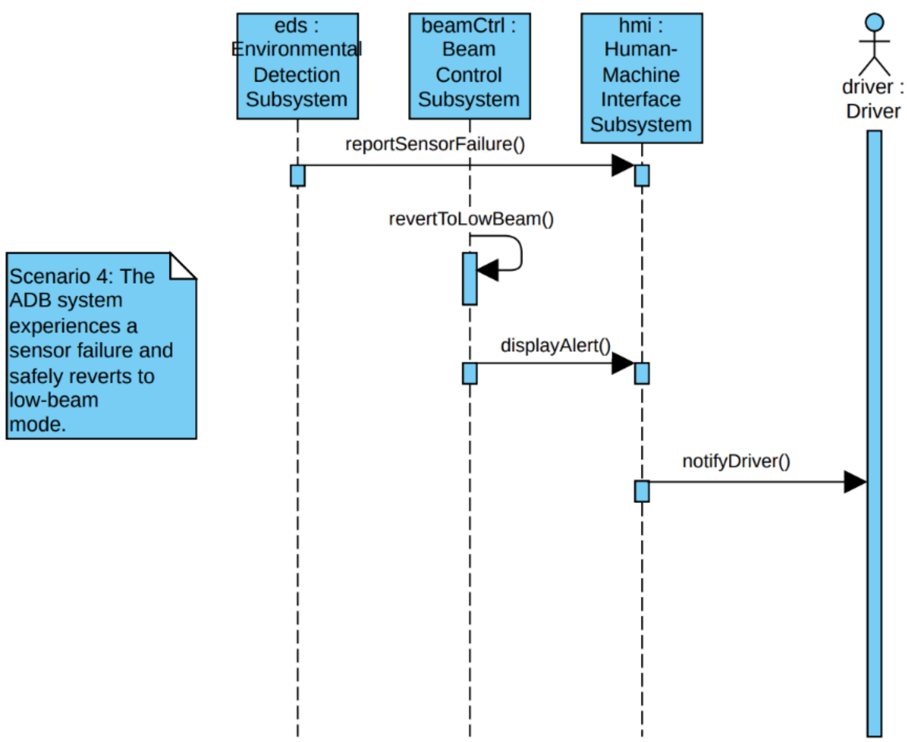

Figure 7: Sequence Diagram Scenario 4

Scenario 5:

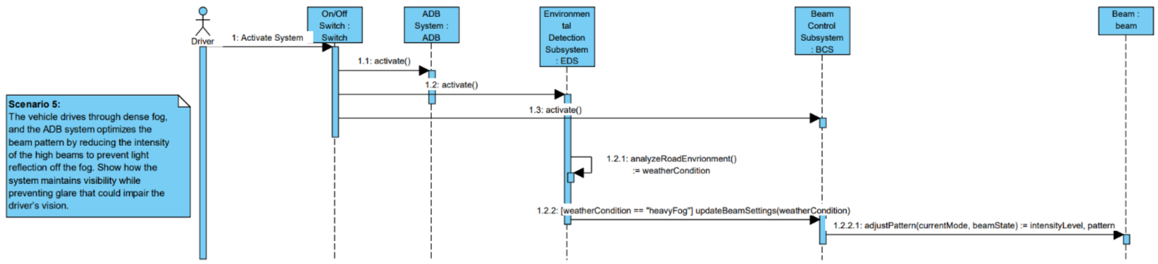

Figure 8: Sequence Diagram Scenario 5

Scenario 6:

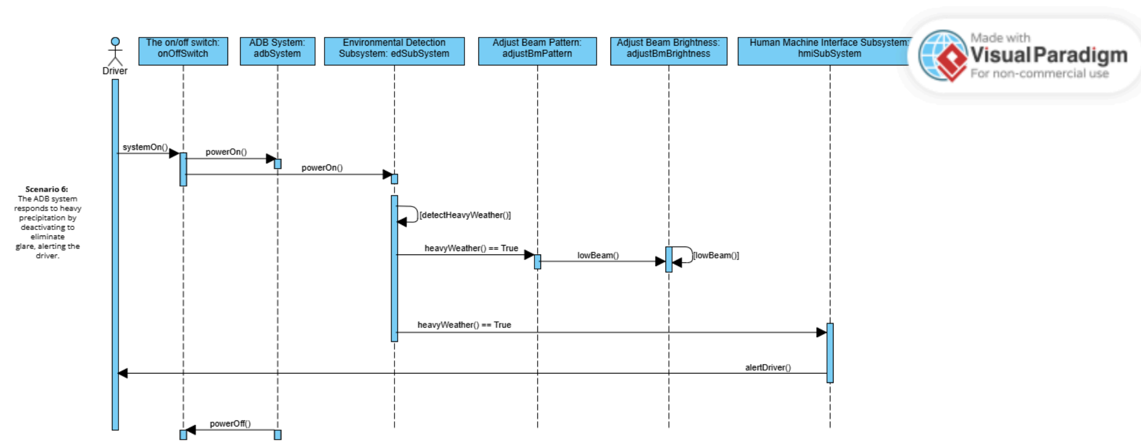

Figure 9: Sequence Diagram Scenario 6

4.4 State Diagrams

This section contains state diagrams that describe how different components of the ADB system move from state to state. In the following state diagrams, states are represented by blue rectangles, and transitions between states are denoted by arrows. The text on each arrow indicates the transition condition that causes the state to change. In a state diagram, the beginning state is depicted by a black-filled circle with an arrow leading to the initial state.

In Figure 10, the driver has initiated the ADB system, with this activation the system then sends the connection signals to the other subsystems (Environment, Cybersecurity, Vehicle Positioning, HMI, and Beam Control Subsystems). This will allow the entire rest of the system to begin monitoring their specific tasks. If there is ever an issue with any of the subsystems it will send an error message to the ADB system to deactivate the system until the issue is resolved.

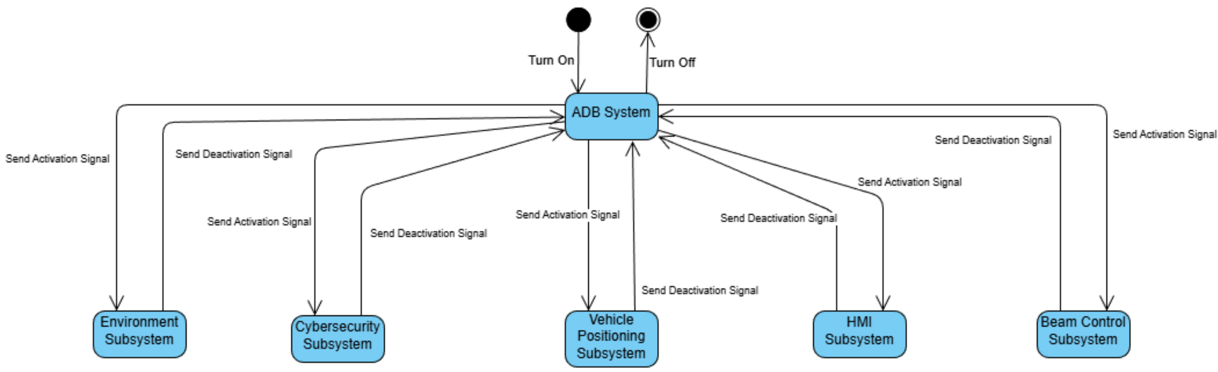

Figure10: State Diagram for the ADB System

In Figure 11, the environment subsystem has been activated and begins to constantly monitor the environment around the vehicle, once it detects that there is not vehicle or the visibility is clear due to no bad weather conditions it will send a message to activate the high beam mode, this will go back to low beam mode if there are any vehicles in the scene or visibility worsens due to bad weather conditions. Of course not all subsystems will be perfect and therefore if there are any errors, a message will be sent to the error handling function and from here it will deactivate the system (default back to low beams) and send a message back to the subsystem to be sent out to the other systems.

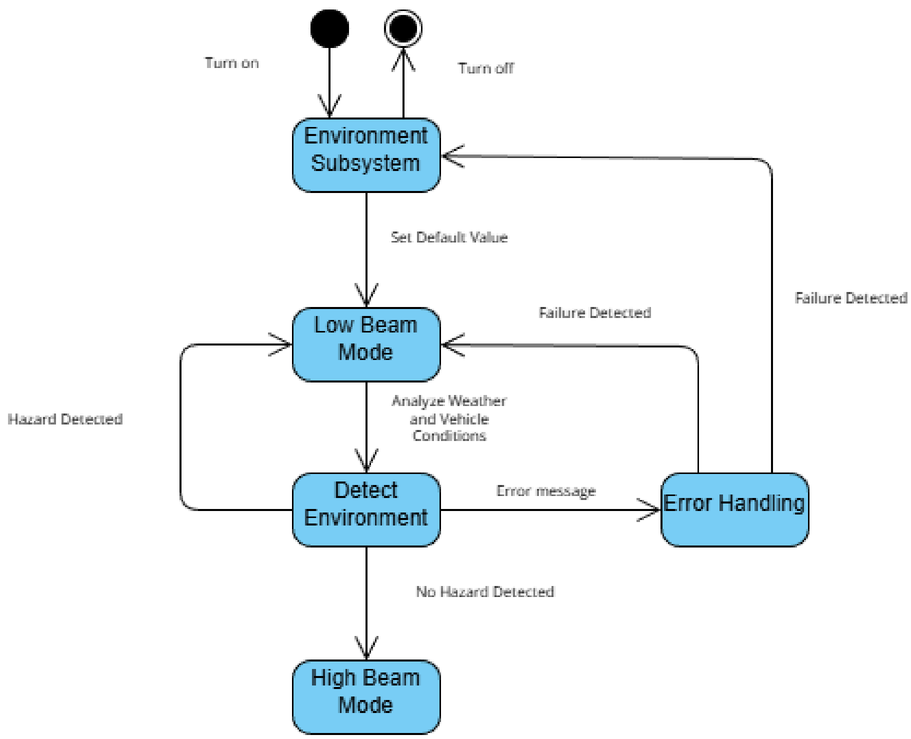

Figure 11: State Diagram for the Environmental Subsystem

In Figure 12, the cybersecurity subsystem is monitoring the entire system making sure that there are not cyber threats attacking the system, if it finds a threat it will send the message back to the main subsystem to then be distributed to the other systems. Error handling has been implemented and if for any reason there is an error during the monitoring process it will send this message to the error handling function which will send this back to the subsystem to be solved.

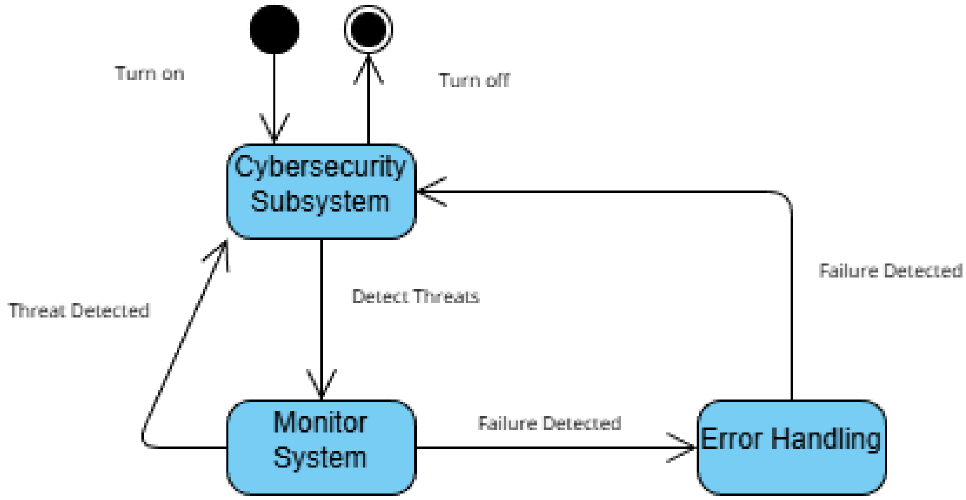

Figure 12: State Diagram for the Cybersecurity Subsystem

In Figure 13, the vehicle positioning subsystem is in charge of managing any change in curve or slope of the road in the direction in front of the vehicle. When initiated the beam will be initiated at a default angle position described in the requirements, from it it will begin to monitor any changes in road, if the system begins to detect that the vehicle is entering a curve or ascending/descending road conditions it will recalculate the horizontal level and send that new angle back to have it sent to the other beam subsystems to ensure that the driver has full visibility at all times. Error handling is taken care of through the error handling function, where if the subsystem runs into any errors it will send the message to the error handling function which will send that back to the subsystem and have it facilitated through the ADB System.

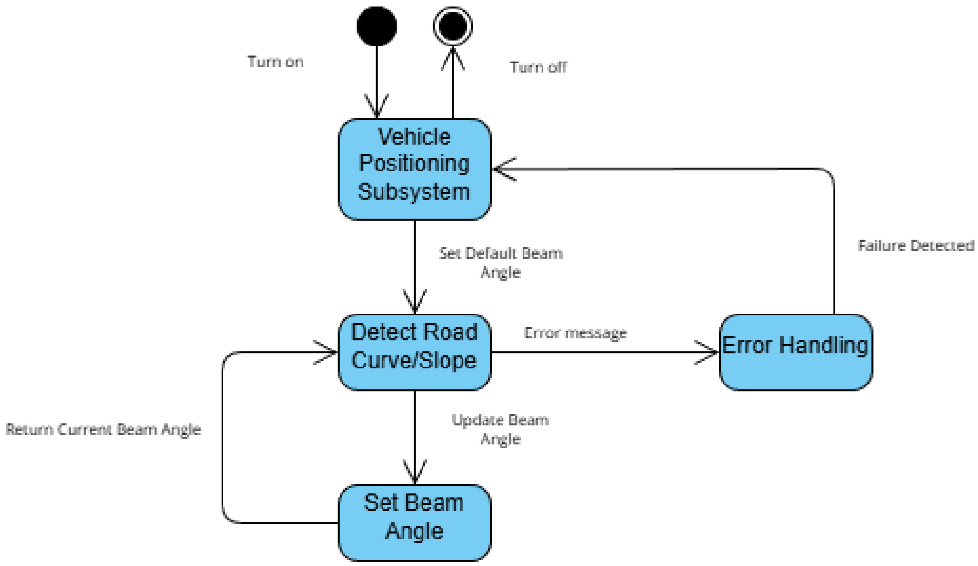

Figure 13: State Diagram for the Vehicle Positioning Subsystem

In Figure 14, the HMI subsystem is on display, this system simply monitors any messages that need to be sent to the driver whether that's simply a notification of them turning on/off the ADB system, a change in beam angle position, or if there has been an error in the system. Once the subsystem is initiated it will monitor for any messages that are being sent throughout the system, once it detects a message that needs to be sent it will then create the message and send that directly to the driver. Any errors are taken care of through the error handling function where it will receive the error message and process that message and have it sent back to the subsystem and facilitated throughout the entire system.

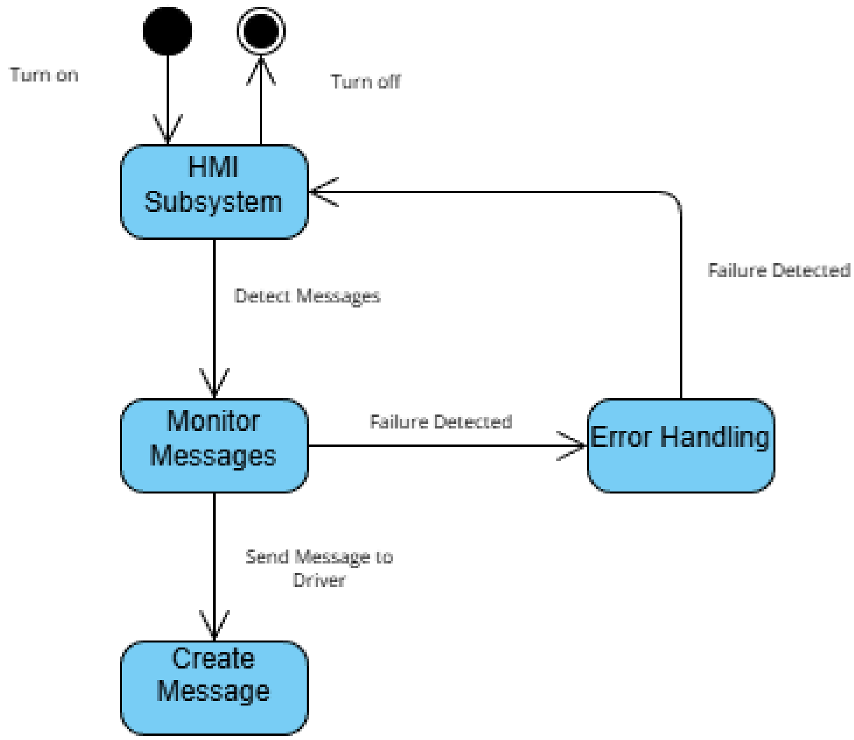

Figure 14: State Diagram for the Human Machine Interface Subsystem

In Figure 15, after low beam mode has been activated, the system collects data from the sensors. Based on this data, it determines whether the beams should be updated to high beam mode, which is initiated through the updateBeams function. Of course, not all systems are perfect, and error handling must be accounted for. Errors are constantly monitored through the sensorFail and beamFailure functions. If either error is activated, the system initiates fail-safe mode and reverts the beams from their current state back to low beam mode.

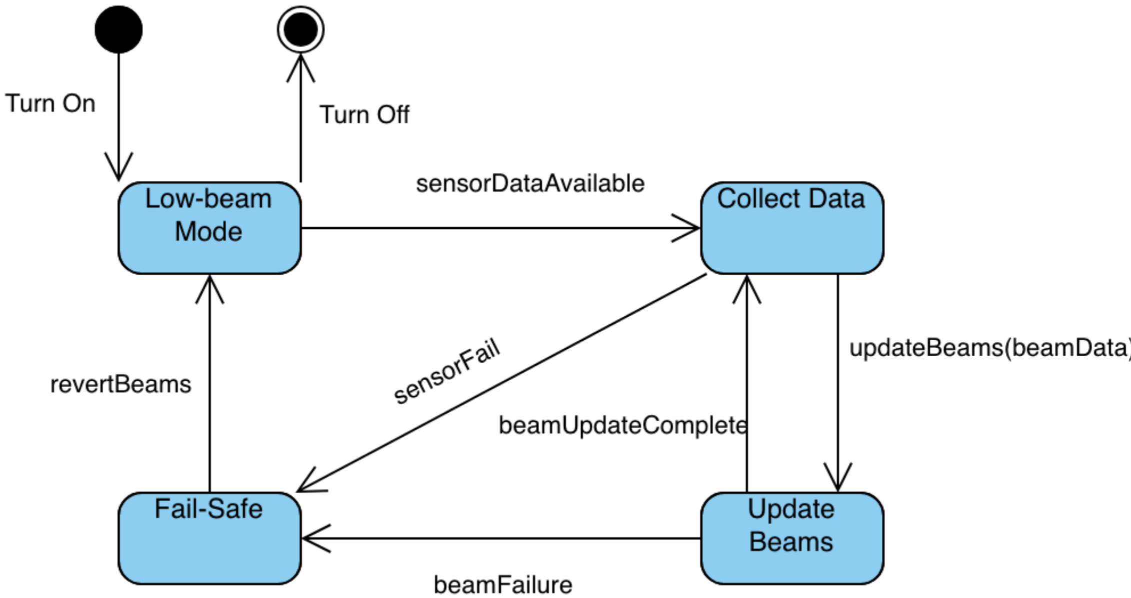

Figure 15: State Diagram for updating beam from sensor data

5 Prototype

The prototype will demonstrate multiple different scenarios that the ADB System will handle. There are a total of 5 different scenarios that can be tested in the prototype. They demonstrate the ADB's ability to adapt to different weather and road conditions and the methods it takes.

5.1 How to Run Prototype

The prototype can be run on this Itch.io page. It is also available on the project website.

5.2 Sample Scenarios

5.2.1 Scenario 1: Clear Highway

This scenario represents the best case scenario. There are no obstacles or heavy weather conditions, so the ADB system is in high beam mode.

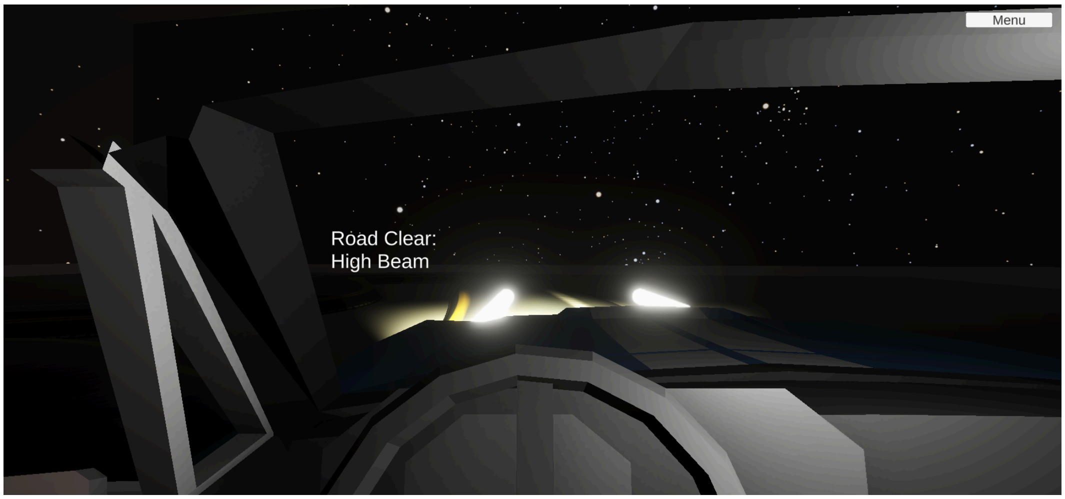

Figure 16: Scenario 1 Prototype

5.2.2 Scenario 2 : Oncoming Vehicle

This scenario represents when the system needs to adjust the beam pattern when an oncoming vehicle is detected, ensuring no glare while maintaining maximum illumination on other road areas. Figure 17 shows the system when the road is clear, activating high beam mode, illuminating the area in front of the driver, figure 18 shows when there is an oncoming vehicle, in order to prevent glare onto the driver and the oncoming vehicle the system must turn into low beam mode.

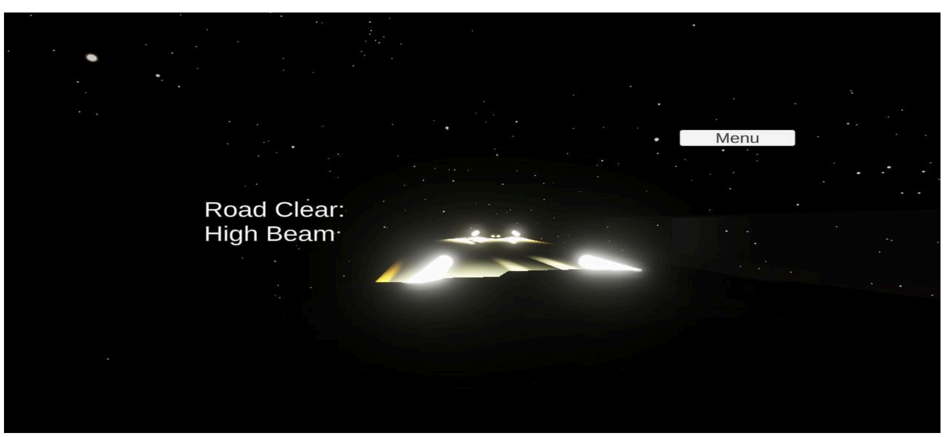

Figure 17: Scenario 2 High Beam Condition

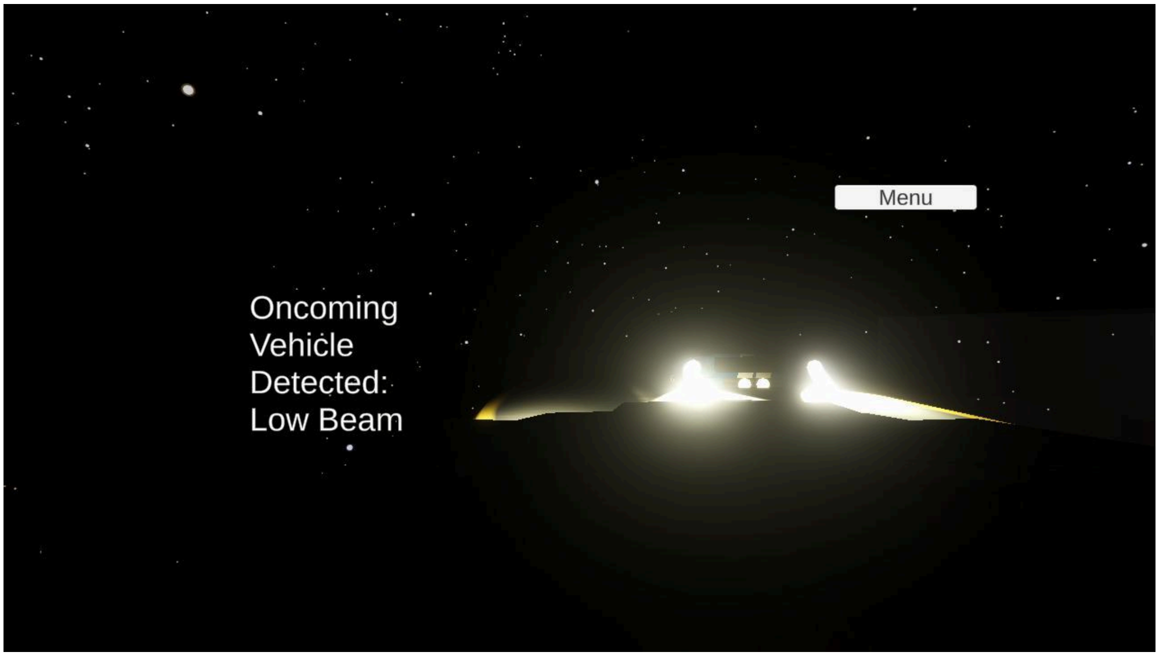

Figure 18: Scenario 2 Low Beam Condition

5.2.3 Scenario 3: Curves and Inclines

This scenario showcases the ADB system's ability to handle curves and inclines. The system uses GPS and various sensors to detect changes in the elevation and curvature of the road. These changes are then relayed to the Beam Controller Subsystem, ensuring that the beam pattern perfectly matches the road condition to ensure maximized visibility while still minimizing the potential glare to other vehicles on the road. Figure 19 showcases the ADB system automatically adjusting to the left turn in the road ahead, ensuring maximized visibility while entering the turn.

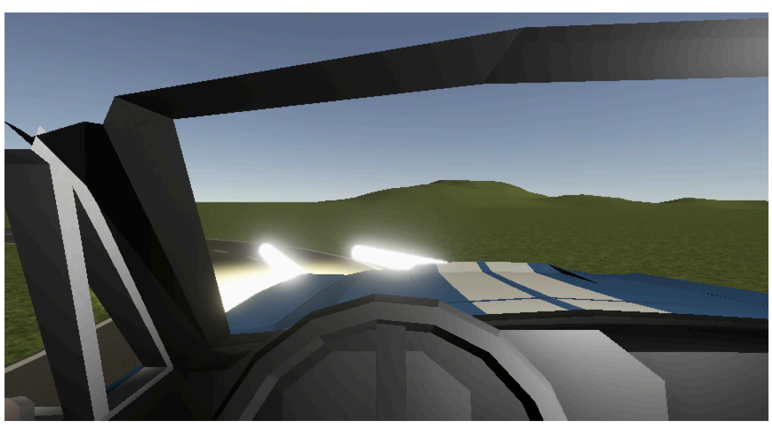

Figure 19: Scenario3 Adjust to Curves and Inclines

5.2.4 Scenario 4: Sensor Failure

Scenario 4 describes when the system experiences a sensor failure. The system responds by notifying the driver of the failure and reverting to low-beam mode.

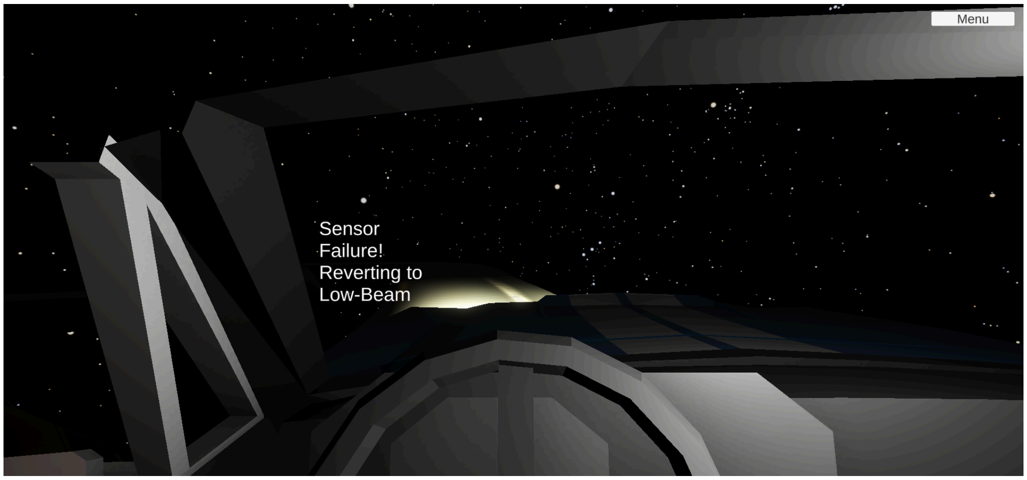

Figure 20: Scenario4 reverts to low beam on system failure

5.2.5 Scenario 5: Heavy Fog

Scenario 5 describes when the car approaches heavy fog. The ADB System responds by lowering illumination level and lowering the beam angle to avoid blinding the driver. Figure 20 demonstrates the ADB system automatically dimming the beams and adjusting the angle when approaching fog.

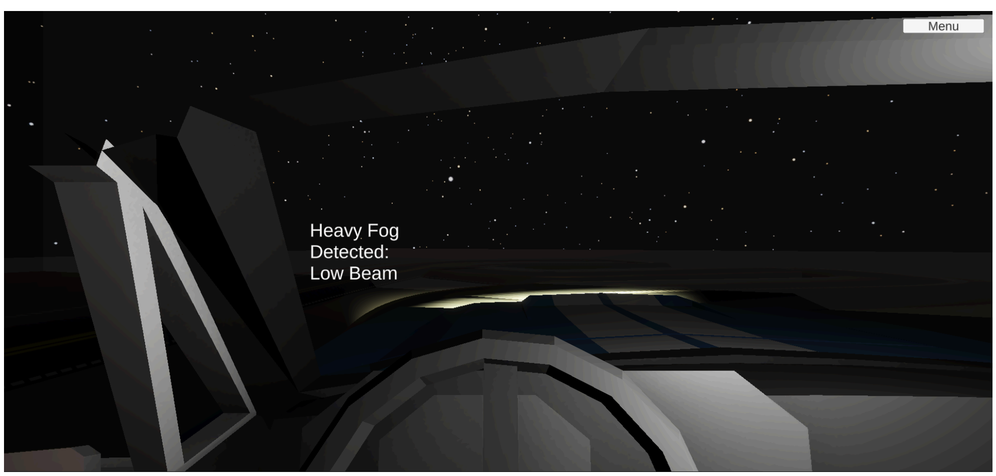

Figure 20: Scenario 5 adjusts to heavy fog

6 References

Start of your text.

[1] D. Thakore and S. Biswas, "Routing with Persistent Link Modeling in Intermittently Connected Wireless Networks," Proceedings of IEEE Military Communication, Atlantic City, October 2005.

[2] "California Driver's Handbook, " State of California DMV, 2025, Web..

[3] "Michigan Headlights Laws," Kensun, 2025, Web.

[4] Erich Hairston, Dean Dawson, Gabrie Italia, Evan Shoup, Kaela Burger, "Software Requirements Specification (SRS)", https://www.cse.msu.edu/~cse435/Handouts/SRS_examples/Final-SRS-Burger-Hairston-Dawson-Italia-Shoup.pdf , Used in Section 2, 2.1, and 5 for template modeling purposes

[5] S. Neubaur, "ADB2," *ADB2 Team Page*, 2025. [Online]. Available: https://www.egr.msu.edu/cse435/adb2/

[6] J. Rhodes, "Adaptive Driving Beam (ADB) System," 2025.

7 Point of Contact

For further information regarding this document and project, please contact **Prof. Betty H.C. Cheng** at Michigan State University (chengb at msu.edu). All materials in this document have been sanitized for proprietary data. The students and the instructor gratefully acknowledge the participation of our industrial collaborators.