Software Requirements Specification (SRS)

Project Automatic Parking Assist 2

Authors: Patrick Hogan, Mehrshad Bagherebadian, Andrew Alstead, Anurag Senapty

Customer: Eileen Davidson

Instructor: Dr. Betty Cheng

1 Introduction

In this Software Requirements Specification document, we will first cover the motivation behind the software to be developed, followed by the overall description of the software, then a comprehensive list of requirements, then diagrams such as use case, state, sequence as well as a domain model. Finally, we will show a prototype of how the system should behave.

1.1 Purpose

The purpose of this document is to provide an in-depth description of what this APA system does and how it will be used. The audience for this document includes software developers as well as project managers from Ford Motor Company.

1.2 Scope

The software product to be developed will be called "Automatic Parking Assist" and this software will be included in Ford vehicles.

The goal of this product is to provide user assistance when perpendicular or parallel parking their vehicle. This will be an embedded software application in Ford vehicles which utilizes stereo cameras as well as sonar sensors to swiftly but safely park a vehicle autonomously.

This application will be initially powered on by the user. Once powered on, it will scan the surroundings for potential parking spots that the car can be safely parked in. It will do this by scanning the area for spaces that meet the required width (1.2x width/length of the vehicle). Once spots are found, it will prompt the user to select and confirm a spot. Following confirmation by the user, the system will begin the parking maneuver and will continuously scan the area for obstacles or pedestrians. If an obstacle/pedestrian is detected, it will stop the vehicle until the vehicle deems it safe to continue. Once the maneuver is complete, the vehicle will stop and the application will terminate.

1.3 Definitions, acronyms, and abbreviations

APA = Automatic Parking Assist
HMI = Human Machine Interface

1.4 Organization

In the following sections, you will see an overall description of the product, the requirements that the product must meet, models describing behavior and edge cases, and finally a prototype modelling the behavior in a more accessible format.

2 Overall Description

In this section, we will cover the perspective of the product, functions of the product, characteristics of the user, constraints, assumptions and dependencies, as well as factors deemed beyond the scope of this product.

2.1 Product Perspective

This product is a stand alone application to be put in place utilizing sensors, cameras, and the human machine interface presently available in all Ford vehicles.

2.2 Product Functions

The Automated Parking Assist is activated by the user, a scanning process of the vehicle's surroundings begins in order to identify viable parking spaces that meet the required dimensions (1.2x the vehicle's width/length). Once spots are detected, the system prompts the user to select and confirm a spot. After that, it begins the autonomous parking maneuver while continuously monitoring the area for obstacles or pedestrians. If an obstruction is found, the system immediately stops the vehicle until it is deemed safe to continue. Once the maneuver is complete, the vehicle stops and the application terminates.

2.3 User Characteristics

Ideally, the user will have a drivers license and will be deemed competent by the users state DMV to operate a motor vehicle. Once these requirements are met, all the user needs to do is follow the prompt on the human machine interface while scanning the surroundings and pressing the brake to stop the vehicle should the vehicle not detect any obstacles.

2.4 Constraints

Safety-Critical Constraints

- The system must always give control back to the driver whenever a manual brake or steering input is detected.
- The system must never move the vehicle if sensor data is missing, invalid, or inconsistent.
- Vehicle speed must remain at or below 5 mph during all automatic maneuvers.
- If any safety-critical sensor or actuator fails, the system must stop the vehicle and alert the driver.
- The system must comply with automotive safety guidelines such as ISO 26262 and Ford's internal safety standards.

Operational and Design Constraints

- The software must run on existing in-vehicle hardware and use the vehicle's built-in cameras, radar, and ultrasonic sensors without hardware changes.
- The system must communicate through Ford's internal CAN bus and HMI framework.
- The software must maintain reliable performance under normal environmental conditions, including moderate rain, fog, or low light.
- Parking maneuvers must finish within 30 seconds, and all visual prompts must update with low latency.
- The system must handle degraded sensors by pausing or stopping rather than continuing blindly.
- All data exchange with the FordPass app must use an encrypted, one-time session for security.
- The system must log each activation, abort, and completion event for diagnostics.

Environmental and External Constraints

- The APA system depends on accurate sensor calibration and vehicle geometry; towing or significant load changes disable automatic parking.
- Operation requires functional network links and near field communication for remote use.
- The software must tolerate reasonable temperature ranges expected for in-vehicle electronics.

2.5 Assumptions and Dependencies

We assume that the user will not have ill intent while using this system and that the user will press the brake should the vehicle not detect a present obstacle. We also assume that the user will access the application via the Human Machine Interface or the FordPass app and that the person activating the APA system is actually the user and not an adversary. We assume that the hardware will function as expected and that the system will always be available.

2.6 Apportioning of Requirements

The following requirements are outside the scope of this Software Requirements Specification document but may be considered in future iterations of this APA system.

1. ML algorithm may classify and highlight non-critical obstacles (e.g., trash bags vs human baby).
2. There must be a measurable accuracy requirement for the ML model (TBD).
3. Adding side or 360° cameras for complete environmental awareness.

3 Specific Requirements

Functional Requirements:

1. The system must be activated through the Human Machine Interface (HMI).
2. The system must be able to be activated by the FordPass app.
3. The driver must be able to select “perpendicular” or “parallel” parking mode.
4. The system must scan for candidate parking spaces using front/rear cameras and side ultrasonics.
5. Once a space is detected, the driver must confirm the selection using the HMI before maneuvering.
6. After confirmation, the system must control gear selection (Drive/Reverse/Park), steering angle, and braking to follow a calculated trajectory into the spot.
7. During the maneuver, the system must continuously monitor surroundings (cameras/radar/ultrasonic) and vehicle pose.
8. If a new obstacle enters the path, the system must stop the vehicle immediately.
9. The system must notify the driver when the parking maneuver is complete.
10. The system must log APA start/stop, mode, completion/failure, and fault codes for diagnostics.
11. The driver must be able to abort the maneuver at any time using the brake pedal or HMI cancel (in-vehicle) or the app cancel.
12. In remote (FordPass) mode, the customer must be able to control the parking maneuver outside the vehicle after authentication.
13. The system must confirm that all commands (in-vehicle and remote) are from authenticated, authorized users.

14. The system must validate sensor inputs to ensure data integrity and reject spoofed or invalid data.
15. The system must handle temporary sensor blockages safely (pause or retry rather than continue blindly).
16. The system should safely deactivate if the driver takes over control (braking to a stop).
17. The dashboard camera feed must automatically switch between front/rear views based on intended motion, displaying steering direction and projected path.
18. A short delay should occur before movement begins after a camera view switch, allowing the driver to verify surroundings.
19. The system must not intentionally select a trajectory that intersects detected obstacles.
20. The system must report system failures to the user and immediately stop the maneuver.
21. The system should maintain effectiveness in poor weather or low lighting conditions.
22. The app's connection for remote parking must use a one-time encrypted session, established via NFC tap-to-authorize (similar to tap to pay).
23. The system must complete parking maneuvers (parallel or perpendicular) within 30 seconds.
24. The system must suspend operation if sensors detect degraded performance due to weather, lighting, or blockage, and notify the user accordingly ("Clean off sensor" or "Maintenance required").

25. The system must apply a 20% safety margin when verifying minimum parking space dimensions (1.2x vehicle length for parallel, 1.2x width for perpendicular).
26. The system must allow the user to pause or cancel a maneuver at any time via brake input, HMI, or app. Holding the brake for more than10 seconds must fully terminate the session.
27. The system must disable automatic parking when towing is detected.
28. The system must verify user authentication and phone proximity (12 ft and near key fob) before allowing remote operation.
29. The system may use AI object detection but must request user confirmation when obstacle classification confidence is below 95%.

Non-functional Requirements:

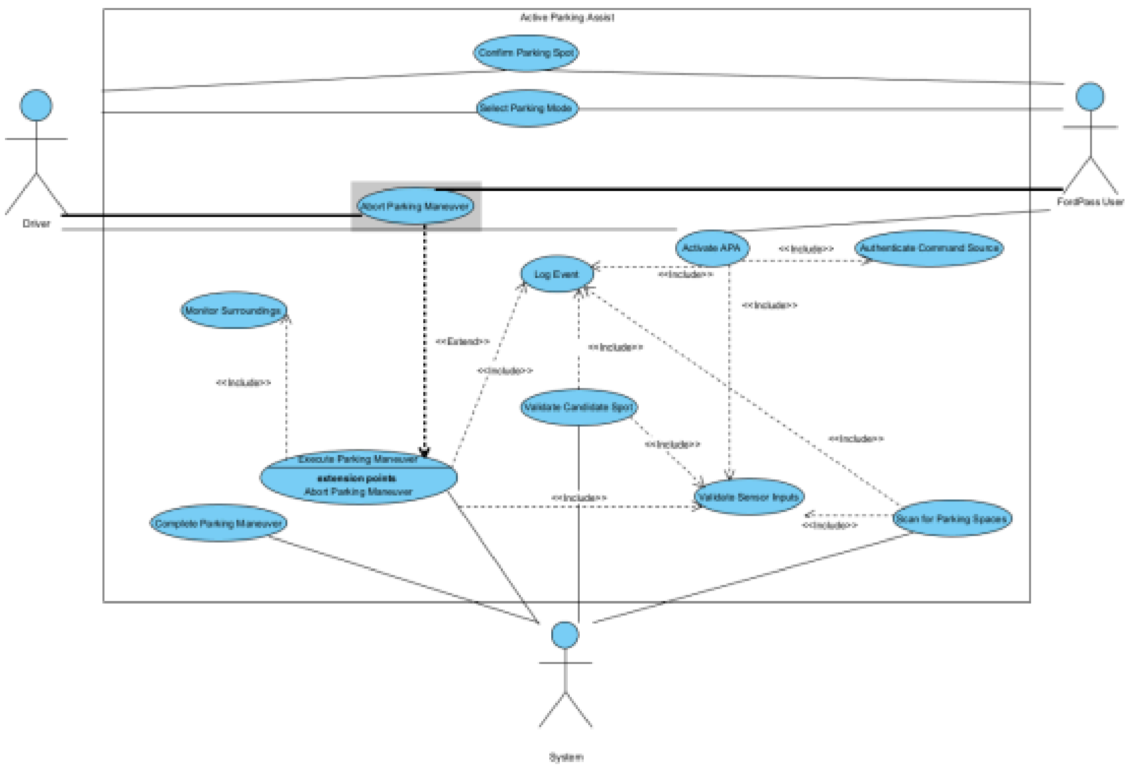
1. The vehicle's speed must never exceed 5 mph during any parking maneuver.
2. A typical parking event should complete within a "reasonable time" (need to ask client).
3. HMI updates (spot found, prompts, warnings, etc.) should appear with low latency.
4. All parking maneuvers (parallel and perpendicular) must complete within 30 seconds under nominal conditions.
5. The system should maintain performance under moderate weather interference (light rain, fog, or low light).

Invariant Requirements:

1. The car must never make contact with any objects or pedestrians.
2. If any required sensor or safety-critical actuator fails, the system must enter a safe state (controlled stop and user notification).
3. The system must only shift gears when vehicle speed is below a safe threshold and braking is applied.
4. If any obstacle is detected within stopping distance, the system must apply braking to avoid contact.
5. The system must verify that cameras and radar are functional before engaging.
6. Radar and camera data should be cross-checked for consistency with significant mismatches preventing activation.
7. The system must immediately stop the maneuver if any sensor reports invalid or conflicting data.
8. The system must never move without confirmed and authenticated user intent.
9. The system must disable itself if environmental conditions prevent safe operation.
10. The system must not operate when towing is detected or if vehicle load conditions change the geometry beyond the parking algorithm's assumptions.
11. If the driver takes manual control (steering or brake), the system must disengage safely and return control to the user.
12. In case of lost connection during remote operation, the system must stop the vehicle immediately.

4 Modeling Requirements

Use Case Diagram and Descriptions:



Use Case:	Activate APA (HMI/App)
Actors:	Driver (using vehicle HMI), FordPass User (using phone app)
Description:	The user activates the Active Park Assist system either through the in-vehicle HMI or the FordPass app. The system checks that activation conditions are met (vehicle state, sensors functioning, and user authentication). Once validated, the system becomes active and is ready for the user to select the parking mode.
Type:	Primary
Includes:	Authenticate Command Source, Validate Sensor Inputs, Log Event
Extends:	N/A

Cross-refs:	FR1 (activation through HMI) , FR12 (remote activation through app) , FR13 (user authentication) , FR14 (sensor input validation), FR2, FR10, FR22, FR28 — app activation, logging, NFC one-time session, phone proximity/key-fob for remote. NFR3 — HMI/app messages should appear with low latency. IR8, IR9, IR10, IR12 — never move without authenticated user intent; disable if unsafe conditions; no operation when towing; stop on lost remote connection. IR5 & IR6 (verify sensors before engaging)
Use Cases	Must be completed before Select Parking Mode and Scan for Parking Spaces.

Use Case:	Select Parking Mode
Actors:	Driver
Description:	After activating APA, the driver chooses the type of parking maneuver (parallel or perpendicular) through the HMI. The system records the selection and adjusts its scanning and parking logic based on the chosen mode.
Type:	Primary
Includes:	Log Event

Extends:	N/A
Cross-refs:	FR2 (select parallel or perpendicular mode) FR25 — apply 20% safety margin to min spot dimensions (ties mode to geometry thresholds).
Use Cases	Must occur before Scan for Parking Spaces.

Use Case:	Scan for Parking Spaces
Actors:	Driver (primary); FordPass User (remote case)

Description:	After the driver selects a mode, the system scans for candidate parking spaces using front/rear cameras and side ultrasonics while the vehicle creeps along. Results feed the next step (spot validation).
Type:	Primary
Includes:	Validate Sensor Inputs; Log Event
Extends:	N/A
Cross-refs:	FR3 (scan with cameras/ultrasonics) FR24 — suspend scanning if sensors/weather/lighting are degraded; notify user. NFR1, NFR5 (speed ≤ 5 mph during maneuver updates with low frequency) IR5–IR6 (verify cameras/radar functional; cross-check) IR7, IR9, IR12 — stop if inputs conflict/invalid; disable if conditions prevent safe operation; stop if remote link is lost.
Use Cases	Must be before Validate Candidate Spot and Confirm Parking Spot.

Use Case:	Validate Candidate Spot
Actors:	System
Description:	For each detected space, the system checks basic geometry and makes sure it is right. For parallel parking it verifies the spot is large enough, and confirms that a collision-free path exists before asking the driver to confirm.
Type:	Secondary
Includes:	Validate Sensor Inputs; Log Event
Extends:	N/A
Cross-refs:	FR4 (parallel spot ≥ 1.2× vehicle length) FR19 (do not choose a trajectory that intersects obstacles) FR25 — enforce the 20% safety margin (parallel ≥ 1.2× length; perpendicular ≥ 1.2× width).
Use Cases	Before confirming parking spot

Use Case:	Confirm Parking Spot

Actors:	Driver (using HMI), FordPass User (using phone app)
Description:	After a suitable parking space is identified and shown to the user, the system asks for confirmation. The user must approve the selected spot either through theHMI or through the FordPass app.The system will not begin the parking maneuver until confirmation is received.
Type:	Primary
Includes:	Log Event
Extends:	N/A
Cross-refs:	FR5 (user must confirm selected spot before maneuvering) FR12 (remote APA control via app) IR8 — system must not move without confirmed/authenticated intent.
Use Cases	Must be done before execute parking maneuver

Use Case:	Execute Parking Maneuver
Actors:	System; Driver (may slow vehicle with brake); FordPass User (in remote mode)
Description:	After the user confirms the selected spot, the system begins the parking maneuver. The system controls steering, gear changes, and vehicle movement to follow a calculated path into the parking space. During vehicle parking, the driver may reduce speed using the brake pedal, while the system maintains a safe low speed throughout the maneuver.

Type:	Primary
Includes:	Monitor Surroundings & Camera View; Validate Sensor Inputs; Log Event
Extends:	Obstacle Detected – Emergency Stop; System Failure Detected – Fail-Safe Stop; Abort Parking Maneuver

Cross-refs:	FR6 (system controls gear, steering, braking to follow trajectory) FR7 (continuous monitoring during maneuver) FR8, FR15, FR17, FR18, FR20, FR21, FR23, FR24, FR26, FR27, FR29 - FR19 (no intentional trajectory into obstacles) NFR1,NFR2 (max speed ≤ 5 mph) IR1 & IR4 (avoid contact and brake if obstacle detected) IR3 (only shift gears when safe) IR4, IR7, IR8, IR9, IR10, IR11, IR12
Use Cases	Leads to Complete Parking Maneuver if successful.

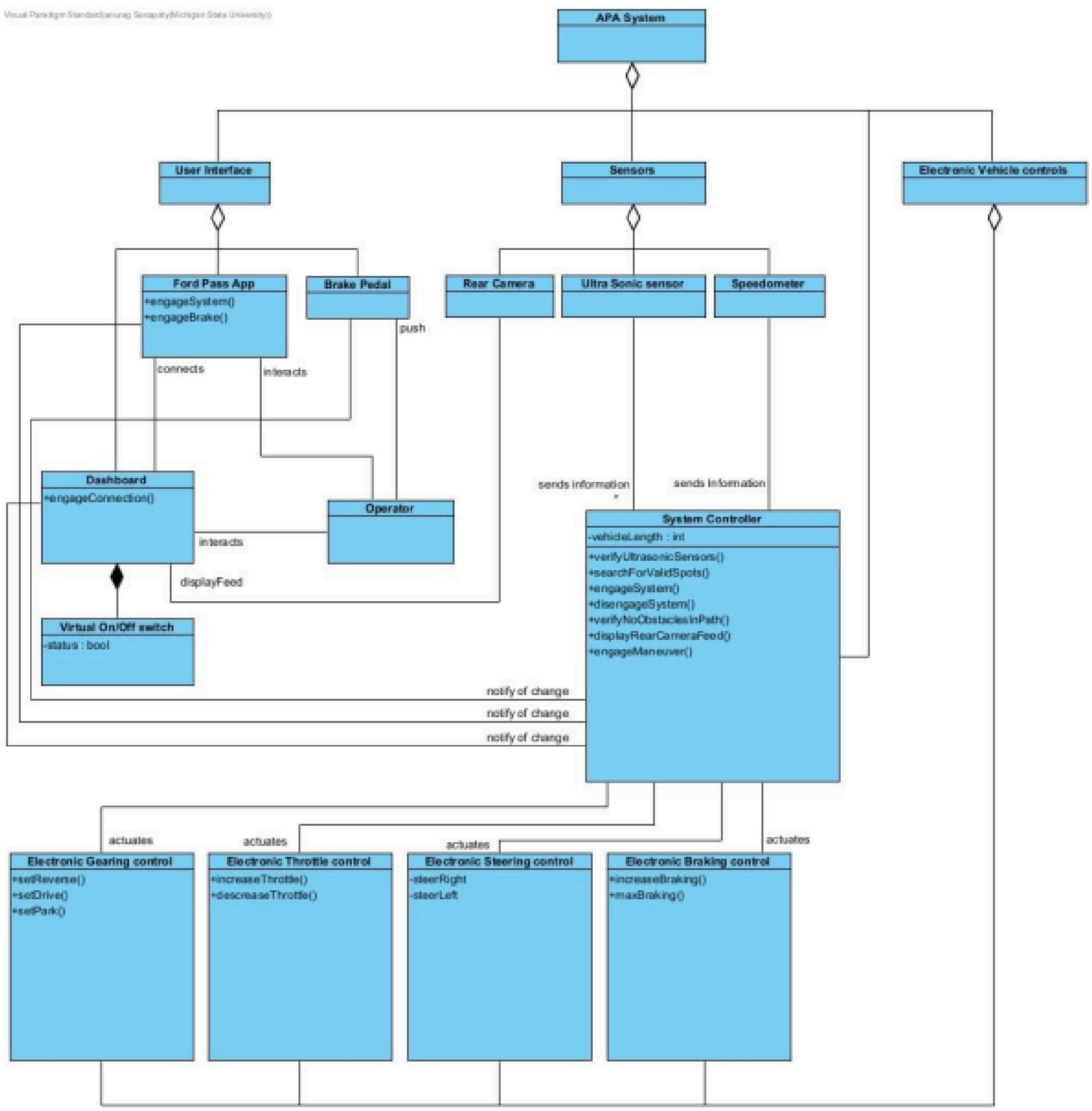
Use Case:	Abort Parking Maneuver
Actors:	Driver; FordPass User; System
Description:	At any point during the parking maneuver, the driver or FordPass user can cancel the operation. The system also stops the maneuver automatically if it detects unsafe conditions. When an abort occurs, the system brings the vehicle to a controlled stop and returns control back to the user.
Type:	Primary
Includes:	Log Event
Extends:	Execute Parking Maneuver
Cross-refs:	FR11 (user can abort at any time via brake/HMI/app) FR16 (system safely deactivates if driver takes control) FR26 — holding brake > 10 s fully terminates the session. IR11 — disengage immediately on manual control.
Use Cases	Occurs during Execute Parking Maneuver.

Use Case:	Complete Parking Maneuver
Actors:	System, Driver; FordPass User

Description:	Once the vehicle is fully positioned within the selected parking space, the system finishes the maneuver. It shifts the vehicle into Park, notifies the user that parking is complete, and then deactivates APA.
Type:	Primary

Includes:	Log Event
Extends:	N/A
Cross-refs:	FR9 (system must notify driver when parking is complete) FR10 — log completion/failure status and any fault codes (diagnostics). NFR2 – overall timing
Use Cases	Marks the end of the parking process.

Domain Model:



Data Dictionary:

Element Name	Description
APA System	A Class to represent the system as a whole
Relationships	Contains the UI, Electronic vehicle controls, sensors and the System controller

Element Name	Description
Sensors	A Class to aggregate all the sensors that input data into the system
Relationships	An aggregation of all the ultrasonic sensors, the rear camera and the speedometer and is part of the APA System class

Element Name	Description
Ultrasonic Sensors	A Class to represent ultrasonic sensors that report how close objects are
Relationships	Sends information to the System controller and is part of the sensors class

Element Name	Description
Speedometer	Measures the speed of the vehicle
Relationships	Sends information to the System controller and is part of the sensors class

Element Name	Description

Rear Camera		A Class to represent the rear camera, used only as a visual aid to the user, not by the system controller
Relationships	Sends information to the Dashboard (Video feed) and is part of the sensors class	

Element Name		Description
User interface		A Class to aggregate all the ways the operator interacts with the system
Relationships	An aggregation of the Brake pedal, dashboard and the Ford Pass app	

Element Name		Description
Brake pedal		A Class to represent the brake pedal the operator can use to pause/stop the system
Relationships	Sends information to the System controller and is part of the user interface class, is also pushed by the operator	

Element Name		Description
Ford Pass App		Allows for remote control of the system. Connects using NFCS and the dashboard
Operations		
	engageSystem	Starts the parking maneuver
	engageBrake	Virtually acts as if the user was pressing the brake

Relationships	Sends information to the System controller and is part of the user interface class. Connects via the dashboard and is used by the operator

Element Name		Description
Dashboard		A Class to represent the car's dashboard, used to connect to the Ford Pass app, display the virtual on/off button and display the rear camera feed
Operations		
	engageConnection	Uses the NFCS chip to establish a one time encrypted connection with the ford pass app
Relationships	Sends information to the System controller and is part of the user interface class. Owns the virtual on/off switch and receives information from the rear camera	

Element Name		Description
Virtual On/Off switch		A Class to represent the virtual button the operator uses to turn the system on
Attributes		
	Status: bool	Was it pressed
Relationships	Sends information to the Dashboard and is owned by the dashboard class.	

Element Name	Description
Electronic vehicle controls	A Class to aggregate all ways the system controller can control the

		car
Relationships	An aggregation of all the Electronic Gearing, Throttle, steering and braking control classes. Is part of the APA system	

Element Name		Description
Electronic Gearing control		A Class to represent the Gear system
Operations		
	setReverse	Sets the car to reverse, allowing us to back up
	setDrive	Sets the car to drive, allowing us to drive forward
	setPark	Sets the car to park, allowing us to end the maneuver
Relationships	Is actuated by the System controller and is part of the Electronic vehicle controls class	

Element Name		Description
Electronic Throttle control		A Class to represent the virtual Accelerator pedal
Operations		
	increaseThrottle	Simulates pushing down on the accelerator
	decreaseThrottle	Simulates easing off the accelerator
Relationships	Is actuated by the System controller and is part of the Electronic vehicle controls class	

Element Name	Description

Electronic Steering control		A Class to represent the virtual steering wheel
Operations		
	steerRight	Turn the car to the right
	steerLeft	Turn the car to the left
Relationships	Is actuated by the System controller and is part of the Electronic vehicle controls class	

Element Name		**Description**
Electronic Braking control		A Class to represent the virtual Brake pedal
Operations		
	increaseBraking	Slows the car smoothly
	maxBraking	In case of emergency brings the car to a full stop as quickly as possible
Relationships	Is actuated by the System controller and is part of the Electronic vehicle controls class	

Element Name		**Description**
System Controller		A Class to represent the System controller that uses information provide to actuate the necessary systems and complete/stop the maneuver quickly and safely
Attributes		
	vehicleLength : int	Vehicle length, is variable from car to car and is used to determine if a

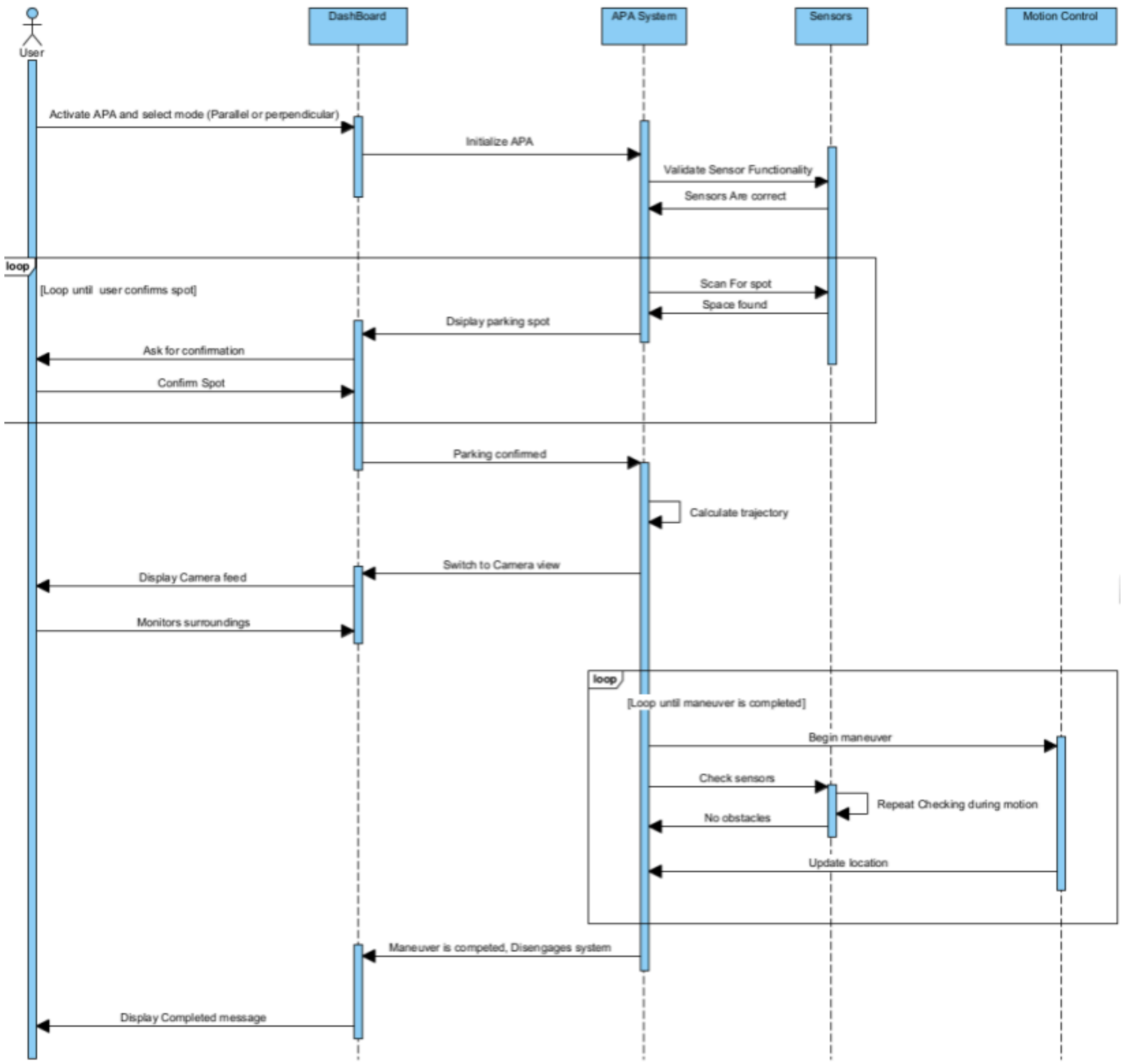
		spot is large enough, 1.2x this
Operations		
	verifyUltrasonicSensors	Cross reference data inputs from all the sensors to ensure a minimum amount, required to safely complete the maneuver are functioning
	searchForValidSpots	Check the sensor data to determine if the parking spot detected is large enough
	engageSystem	Start searching for and detecting parking spots
	disengageSystem	Return manual control at either the end of a successful maneuver or in case the system detects an obstacle
	verifyNoObstaclesInPath	Ensure that the ultra-sonics do not detect obstacles in the desired vehicle path
	displayRearCameraFeed	Engage the dashboard to show the feed from the rear camera when backing up to ensure operator is aware
	engageManeuver	Begins the parking Maneuver, ensured to stay under the 8 kilometers/h (5 miles/h) speed limit and actuates the Garing, Throttle, steering and braking systems to complete the parking maneuver

Relationships	Is a part of the APA system and receives sensor inputs from the Ultrasonics and speedometer. It receives Inputs from the User through the Dashboard, brake pedal and FordPass app. It actuates the Electronic Gearing, Throttle, Steering and braking control systems

<u>Sequence and State Diagrams:</u>

Diagrams for the regular use case including the case in which a spot is invalid

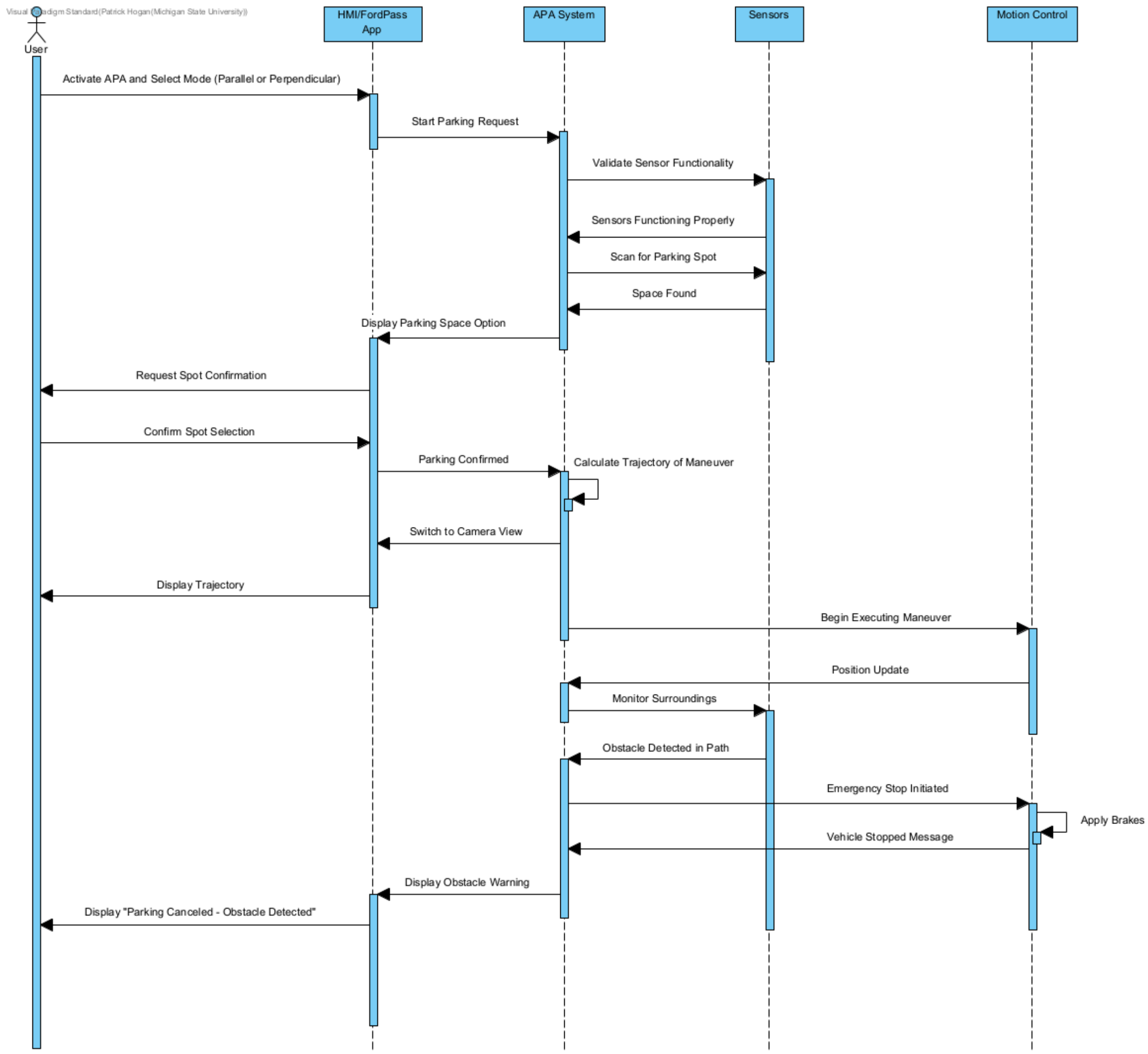
Sequence Diagram:



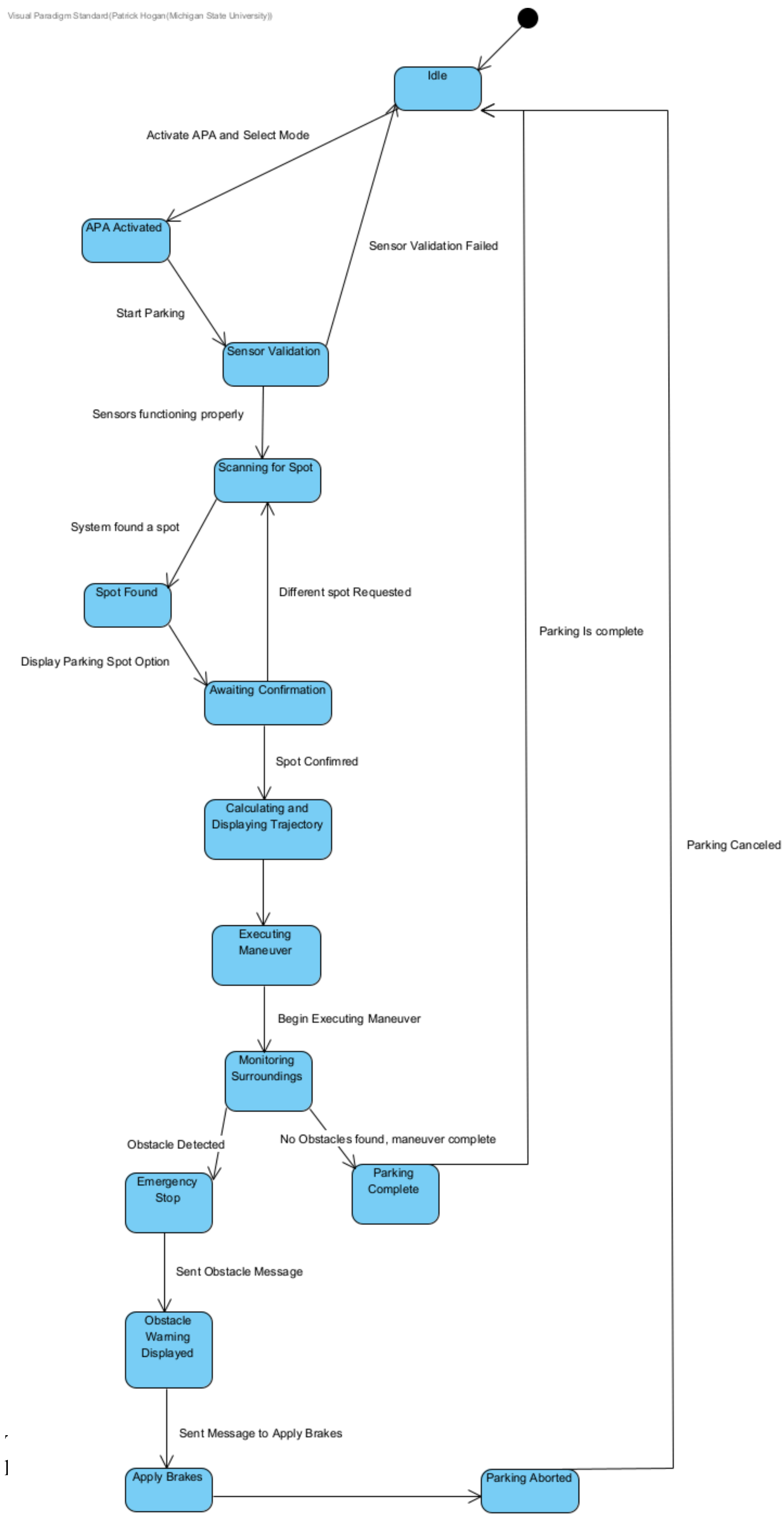
State Diagram is the same as the next set of diagrams.

Diagrams for when the system detects an obstacle in its path

Sequence:



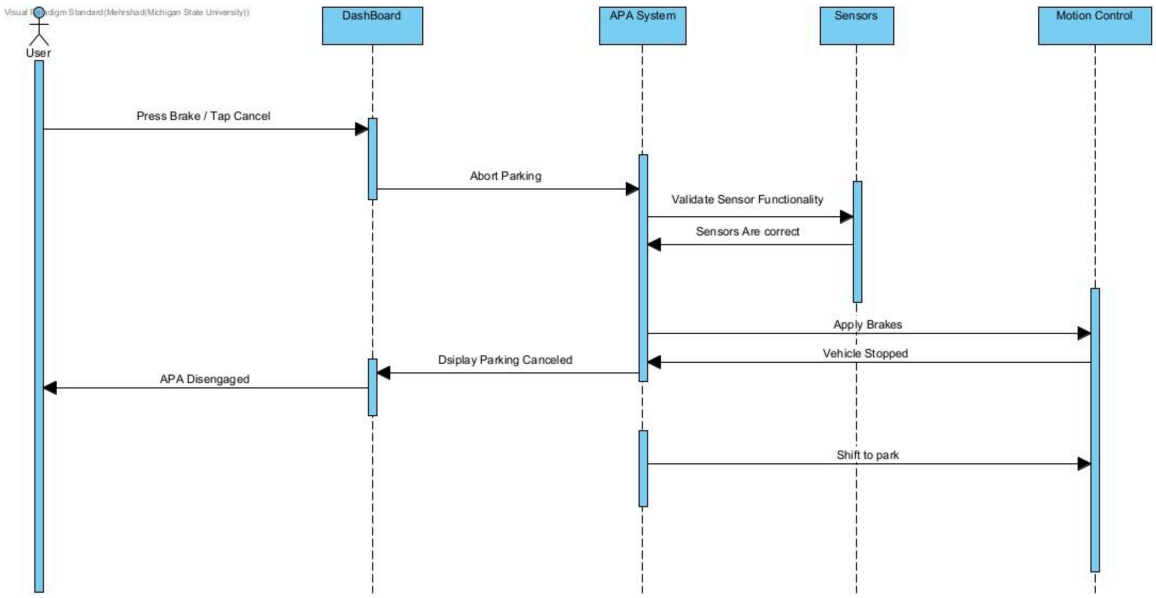
State on next page:


Visual Paradigm Standard(Patrick Hogan(Michigan State University))
Idle
Activate APA and Select Mode
APA Activated
Sensor Validation Failed
Start Parking
Sensor Validation
Sensors functioning properly
Scanning for Spot
System found a spot
Spot Found
Different spot Requested
Parking Is complete
Display Parking Spot Option
Awaiting Confirmation
Spot Confimred
Calculating and Displaying Trajectory
Parking Canceled
Executing Maneuver
Begin Executing Maneuver
Monitoring Surroundings
Obstacle Detected
No Obstacles found, maneuver complete
Emergency Stop
Parking Complete
Sent Obstacle Message
Obstacle Warning Displayed
Sent Message to Apply Brakes
Apply Brakes
Parking Aborted

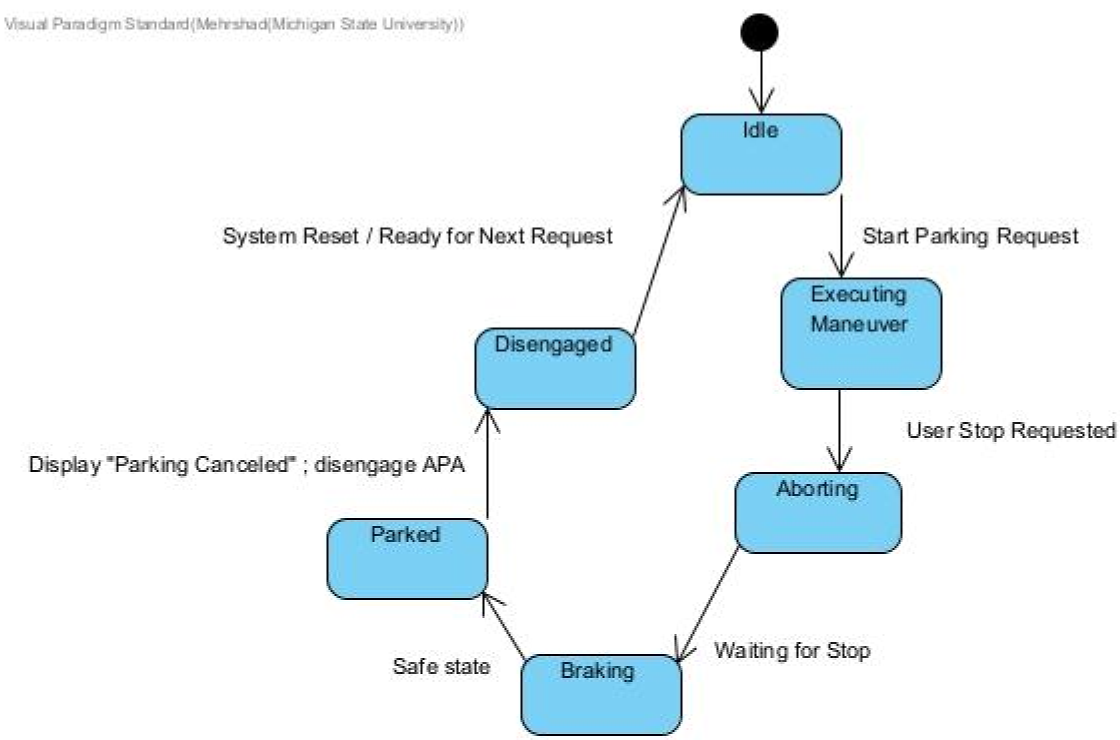
ıformation)

Diagrams for aborting the APA system

Sequence:

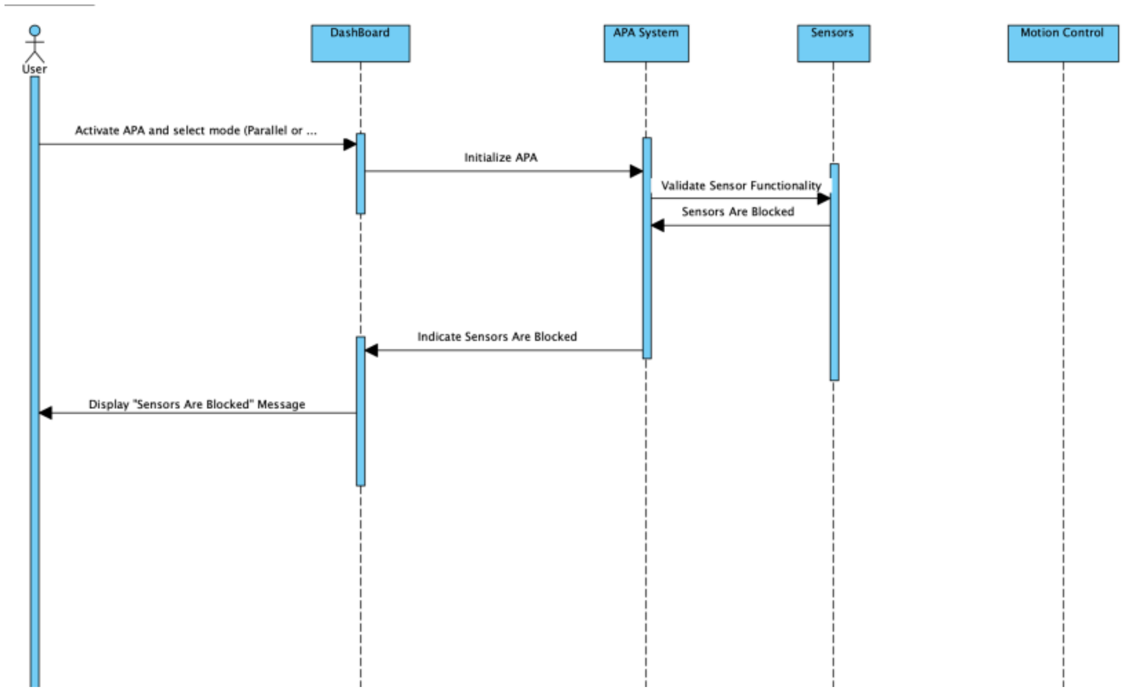


State:

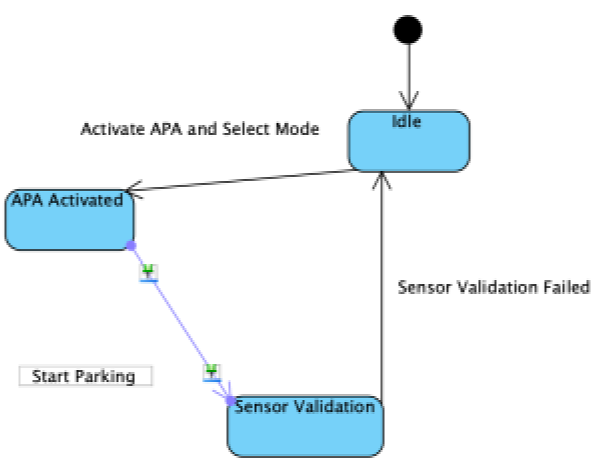


Diagrams for the case in which sensors are blocked and the system cannot run

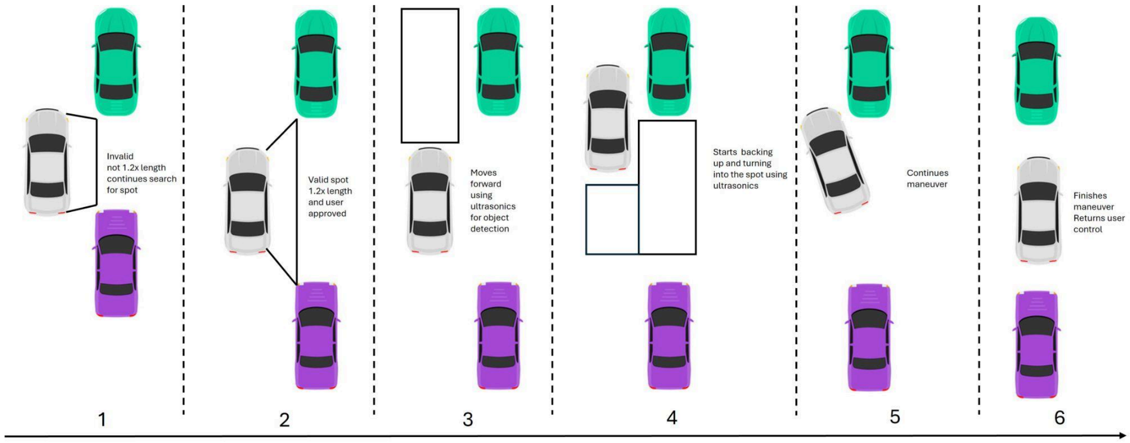
Sequence:



State:



5 Prototype



^This is prototype v0, check prototype on website for v1. **This section is not final! We will have our real prototype on this doc for SRS v2. We are working on a time crunch.**

https://alsteada.github.io/CSE435APA2/

5.1 How to Run Prototype

[Insert Prototype v1 instructions here]

5.2 Sample Scenarios

[See Prototype v1]

6 References

None at this time.

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.