Software Requirements Specification (SRS)
Pedestrian Collision Avoidance System (PCAS)

Authors: Cole Coughlin, Josie Beach, Andrew Gardner, Brandon Lam, Jiayue Chai
Customer: Mr. Chris Capalidi, Project Manager, Dataspeed Inc.
Instructor: Dr. Betty H.C. Cheng, Michigan State University

1 Introduction

According to the Insurance Institute for Highway Safety (IIHS), pedestrian deaths resulting from motor vehicle collisions account for 17% of crash fatalities [1]. More than half of these fatal collisions resulted in the evening and night time, when there is less visibility [1]. Pedestrians can be unpredictable and sometimes hard to see. In cases where the driver is going too fast or not paying close enough attention to the walkways it can be hard to stop in time to avoid a collision. To ensure the safety of pedestrians and drivers from collisions, the Pedestrian Collision Avoidance System (PCAS) is necessary to stop the vehicle without assistance from the driver in the event of a potential collision.

This document details the requirements, functionality, elements, and uses of the Pedestrian Collision Avoidance System (PCAS) that assists automobile drivers in stopping the vehicle to avoid collisions with pedestrians. In section 1, the purpose of the document and the scope of the software to be developed for PCAS are delineated.

1.1 Purpose

The purpose of this document is to describe PCAS such that it is easily understandable in detailing both the customer’s requirements and the implementation of the system. The document is designed to give a clear understanding of the system to developers of the system in designing and developing PCAS. The document is also designed to give a clear understanding of the characteristics and functionality of PCAS to the stakeholders.

1.2 Scope

PCAS is an embedded system in an automotive vehicle whose primary function is to prevent collisions and damages with a pedestrian while the vehicle is moving forward. The secondary objective of PCAS is to minimize lost time. PCAS detects pedestrians in the path of the vehicle and applies the brake if necessary to avoid a collision with the pedestrian. The driver is warned when there is a potential collision by the flashing of warning lights on the dashboard and the speakers emitting a beeping sound. Every 100ms the sensors give feedback to the system on the direction, speed, and position of the pedestrian. The system uses this information with the speed of the vehicle to determine if a collision is imminent via the Pedestrian Collision Avoidance Algorithm (PCAA or PCA Algo). If a collision is determined by the PCAA, the brakes will be applied to avoid the collision. The system also has a fail safe mode that will activate if the
Fail Safe Sensor detects there is an error in the system. With fail safe mode activated, the brake by wire system will not respond as quickly to the PCAS with the response time increasing from 200ms to 900ms. As a result, PCAS and the PCAA must adapt to avoid all collisions.

1.3 Definitions, acronyms, and abbreviations

- **Lost Time**: Time difference (in seconds) between system on and system off to reach a common point beyond the pedestrian with controlled vehicle back again at steady state velocity [2].
- **Pedestrian**: A pedestrian is a person or animal that is in the road.
- **Pedestrian Collision Avoidance Algorithm (PCAA)**: The PCAA is the algorithm used by the PCAS to determine if the vehicle will collide with the pedestrian using the vehicle’s speed and the information given by the sensor (position, speed, direction of pedestrian).
- **Pedestrian Collision Avoidance System (PCAS)**: The PCAS is the entire embedded system responsible for avoiding collisions with pedestrians.

1.4 Organization

The rest of this document is organized in the following order: Section 2, which describes the system, Section 3, which defines the requirements of the system, Section 4, which provides diagrams that illustrate the functionality of the system, and Section 5, which includes the proof of concept prototype of PCAS.

2 Overall Description

In this section, the PCAS is described in terms of its perspective and functionality. The expectations of the user and user interactions are also described. Lastly, the constraints, assumptions and apportioned requirements are explained in this section.

2.1 Product Perspective

The role of the PCAS is to mitigate death, injuries and damages from pedestrian collisions by ensuring the vehicle avoids them completely and automatically. The system is designed to apply the brake when the driver cannot react in time to a pedestrian in the path of the vehicle. This could be due to the driver not being able to see the pedestrian or the pedestrian not being able to see the vehicle, for example. The PCAS is only applicable in cases where the pedestrian is in front of the vehicle.

The PCAS is an embedded system that will interact with many other existing parts of the vehicle. The system will interact with the brake by wire system, the dashboard, the speaker system, the acceleration by wire system, and the front facing sensors on the vehicle (pedestrian detection sensor). The system warns the driver by flashing a light on the dashboard display. The speaker system warns the driver by emitting beeps. The system interacts with the brake by wire
system to decelerate the vehicle to avoid a collision. The driver may turn PCAS on and off via a button on the dashboard.

Below is a diagram of the architecture of PCAS. The system receives information from the Pedestrian Detection Sensor which is an input to the PCAA which, taking the vehicle speed into account, calculates the probability of a collision. The algorithm decides if the vehicle needs to slow down if there is a potential collision in which case the system engages the Vehicle Brake by Wire System, which reduces the speed of the vehicle.

![Figure 1: System Architecture for PCAS](image)

### 2.2 Product Functions

The PCAS gives the driver a sense of security in the event of a potential collision with a pedestrian by engaging the brake when the driver is unable to. The responsibility of the system is to prevent collisions by applying the brake by wire system when a pedestrian is detected in front of the vehicle. The system monitors the path in front of the vehicle for pedestrians. When a pedestrian is detected by the Pedestrian Detection Sensor, the PCAA runs a calculation determining if a collision is possible by analyzing the path between the vehicle and the pedestrian [2]. The PCAA assumes the width of the vehicle to be 2m and the pedestrian to be 0.5m wide in diameter when performing the collision calculation. The Pedestrian Detection Sensor can detect pedestrians from up to 35m away from the front of the vehicle in any visibility. The sensor sends packets of information of the pedestrian's speed, direction and position to the system every 100ms [2]. If the PCAA calculates that a collision is possible, the system will engage the brake by wire system to reduce the velocity of the car until a collision is no longer possible by the PCAA. The maximum that the brake by wire system can decelerate the vehicle is 0.7g, and the brake by wire system will reach the requested deceleration in 200ms [2]. When the system detects that a collision is no longer possible the system will engage the acceleration by wire system to accelerate the vehicle back to steady state at a rate of .25g [2]. The system includes a Fail Safe Sensor which scans the system for vulnerabilities or dysfunctions and engages the fail safe mode if any are detected. In fail safe mode, the response time for braking
increases from 200ms to 900ms. The system functions, including the PCAA, adjust to avoid collisions with a slower brake by wire system response time.

2.3 User Characteristics

The user (driver) is expected to be a legal driver. This includes somebody in driver’s training under legal supervision of a licensed adult of legal age to supervise. The user is expected to pay attention to road signs, surroundings and abide by all traffic laws while driving. The user is expected to have read the owner’s manual of the vehicle which includes a description of the PCAS and how it affects the user. The PCAS is not intended to be the only safety precaution for pedestrians; the user is still expected to drive with care.

2.4 Constraints

In this section, the constraints of PCAS are described. Hardware interface constraints of the PCAS include the following: the Pedestrian Detection Sensor must be unobstructed and functional, the Fail-Safe Sensor must be unobstructed and functional. The acceleration by wire system must also work properly. If these constraints are not met, the system will not perform properly. User interface constraints include the System On/Off button located on the dashboard screen. The System On/Off button must be functional and undamaged for the system to perform properly. The system’s safety-critical properties include the brake by wire system and the before-mentioned Pedestrian Detection Sensor and Fail-Safe Sensor. For the system to properly avoid collisions, the brake by wire system must be fully functional.

2.5 Assumptions and Dependencies

The PCAS depends on embedded systems of the vehicle to be fully functional. These hardware and software systems include the brake by wire system, the dashboard system, and the speaker system. The PCAS also depends on a fully functional Pedestrian Detection Sensor. The user is expected to interact with the system as described in the document.

2.6 Apportioning of Requirements

There are many aspects of the requirements of PCAS that are not in scope of this project that could be addressed in future versions. Examples of these aspects include the PCAA, wiring and hardware of PCAS, specifications of data transfer, and the exact details of cyber security protocols to mitigate cyber attacks. For this document these examples are expected to work as intended without the exact details on how they work.

3 Requirements
This section describes the requirements needed for the PCAS system to perform correctly.

1. The system will use the pedestrian sensor to detect pedestrians that are static or moving perpendicular to the vehicle within 35m of the front of the vehicle.

   1.1. The pedestrian sensor shall be implemented using infrared sensors, radar sensors, and a stereo camera. This sensor will be located on the front middle part of the vehicle.

   1.2. The pedestrian sensor will detect the pedestrian’s position, speed, and direction. It must send this information to the system every 100ms.

2. When the pedestrian sensor detects a pedestrian in the path of the vehicle, the system will use the Pedestrian Collision Avoidance Algorithm (PCAA) to determine if the pedestrian will collide with the vehicle at the vehicle’s current speed and the pedestrian’s current speed, direction, and position.

   2.1. The PCAA will calculate the path of the vehicle and the pedestrian. In this calculation, the vehicle is assumed to be 2m wide and the pedestrian is assumed to be 0.5m wide in diameter.

   2.2. If the PCAA calculates there will be a collision between the pedestrian and the vehicle, the system shall warn the driver by flashing a warning light on the dashboard and emitting a beeping noise over the speaker system.

   2.3 If the PCAA calculates there will be a collision between the pedestrian and the vehicle, the system shall engage the brake by wire system to decelerate the vehicle. The maximum deceleration of the brake by wire system is 0.7g.

3. When the system no longer detects a pedestrian in the path of the vehicle, the system will accelerate back to steady state speed at 0.25g using the acceleration by wire system.

4. The system shall include a fail-operational mode in which the system increases the response time to reach requested deceleration from 200ms to 900ms.

   4.1. The system should adjust to maintain zero collisions in trade for increased lost time during fail-operational mode.

5. The system shall prioritize avoiding accidents over minimizing lost time.

6. The system shall mitigate “lost time” for all non-collision scenarios by providing as little slowdown as possible.

7. The system shall have a PCAS power on and PCAS power off setting accessible by a button on the dashboard screen.
8. With functional and unobstructed sensors, the system shall avoid all collisions in the specified scenarios in any weather conditions and time of day.

3.2 Cybersecurity Requirements

1. The system shall prevent, detect, and mitigate hackers from accessing the total system through isolated system parts.

2. The system shall use digital certificates to authenticate and encrypt communication between the system and devices in the vehicle.

3. The system shall constantly check for malicious code that is running against the original code.

4. The system shall use memory protection units (MPUs) to prevent cascading failures and exploits.

5. The system shall minimize the use of external communication hardware.

With vehicles becoming more technological comes more opportunities for threats to occur via components such as communication based hardware and software running on an electronic control unit. These pathways that allow for cyberattacks are known as threat vectors and as mentioned, one threat vector for the PCAS system is communication based hardware. For communication based hardware, a communication channel allows hackers to access a vehicle remotely. This can allow threat actors such as interruption in DOS attacks and fabrication such as fake input data. DOS attacks would lead to interrupted services, which in turn can cause a potential collision with a pedestrian. Fake input data would deliberately mess with the PCAS’ ability to detect a pedestrian while also offering the ability to cause harm to not just a potential pedestrian, but to the passengers and other vehicles on the road. Brakes and the acceleration by wire system can be messed with if the system uses the false data input, driving the vehicle out of control. To minimize this risk, the use of external communication hardware should be minimized. If communication is necessary, then digital certificates should be used in order to authenticate and encrypt communication between the system and devices in the vehicle.

Another threat vector for the system is software running on an electronic control unit. As vehicles become more technological, many of the vehicle’s operations become software based, which leads to threat actors such as interception in monitoring/copying of data and modification with a virus to change/remove data. Typically physical access is required for these threat actors, which can be done through repair shops or mutual connections with the owner of the vehicle. From there, malicious code can be injected into the system leading to vehicle tracking and false error reports. To prevent this, the system should continuously check for malicious code that is running against the original code. In addition, the total system will consist of isolated system
parts that use memory protection units to prevent a complete takeover. If a part of a system becomes compromised, the system resorts to fail-safe mode.

4 Modeling Requirements

Section 4 includes models with descriptions that specifies how the PCAS system works in detail. These models describe how key elements interact with each other while including example scenarios that depict the PCAS services in action.

4.1 Use Case Diagram

The use case diagram below describes the use cases in the PCAS system. The actors include the driver, dashboard, fail safe sensor, sound system, acceleration by wire system, pedestrian dashboard sensor and brake by wire system. Each actor plays a role in the system that interacts with the use cases and other actors in the system. The use cases are shown in the diagram as circles and are connected to actors by straight lines and connected to other use cases with a dotted line. The include on the dotted lines show that the use case includes the other use case it is pointing to. Under the use case diagram there are tables that help describe each use case more in depth.
Figure 2: Use case diagram for PCAS
**Use Case:** Alert Driver

**Actors:** Sound System, Dashboard

**Description:** Warning lights on the dashboard flash and beeping sounds over the speaker system. The warning occurs when a collision is imminent and Avoid Collision is initiated.

**Type:** Primary

**Includes:** N/A

**Extends:** N/A

**Cross-refs:** 2.2

**Use cases:** Avoid Collision

---

**Use Case:** Engage Brakes

**Actors:** Brake By Wire System

**Description:** The system engages the brakes of the vehicle to decelerate the vehicle. The system engages the brake while Avoid Collision is active.

**Type:** Primary

**Includes:** N/A

**Extends:** N/A

**Cross-refs:** 2.3

**Use cases:** Avoid Collision

---

**Table 1:** Use case description for *Alert Driver*

**Table 2:** Use case description for *Engage Brakes*
### Use Case: Avoid Collision

**Actors:** Pedestrian Detection Sensor

**Description:** The system takes necessary action to avoid the collision if a collision is calculated probable by Pedestrian Collision Avoidance Algorithm (PCAA). If PCAA determines that the system must take action in order to avoid a pedestrian, Avoid Collision interacts with Alert Driver and Engage Brakes.

**Type:** Primary

**Includes:** Alert Driver, Engage Brakes, Pedestrian Collision Avoidance Algorithm

**Extends:** N/A

**Cross-refs:** 1, 1.2, 2.2, 2.3, 3

**Use cases:** Alert Driver, Engage Brakes, Pedestrian Collision Avoidance Algorithm

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Avoid Collision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors</td>
<td>Pedestrian Detection Sensor</td>
</tr>
<tr>
<td>Description</td>
<td>The system takes necessary action to avoid the collision if a collision is calculated probable by Pedestrian Collision Avoidance Algorithm (PCAA). If PCAA determines that the system must take action in order to avoid a pedestrian, Avoid Collision interacts with Alert Driver and Engage Brakes.</td>
</tr>
<tr>
<td>Type</td>
<td>Primary</td>
</tr>
<tr>
<td>Includes</td>
<td>Alert Driver, Engage Brakes, Pedestrian Collision Avoidance Algorithm</td>
</tr>
<tr>
<td>Extends</td>
<td>N/A</td>
</tr>
<tr>
<td>Cross-refs</td>
<td>1, 1.2, 2.2, 2.3, 3</td>
</tr>
<tr>
<td>Use cases</td>
<td>Alert Driver, Engage Brakes, Pedestrian Collision Avoidance Algorithm</td>
</tr>
</tbody>
</table>

**Table 3:** Use case description for *Avoid Collision*

### Use Case: Engage Fail Safe Mode

**Actors:** Fail Safe Sensor

**Description:** Primary Use case invoked by the Fail Safe Sensor. The objective of this use case is to increase the response time for a requested deceleration. As a result the collision calculations will have to be adjusted to maintain the system priorities.

**Type:** Primary

**Includes:** Adjust Pedestrian Collision Avoidance Algorithm

**Extends:** N/A

**Cross-refs:** 4, 4.1

**Use cases:** Adjust Pedestrian Collision Avoidance Algorithm

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Engage Fail Safe Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors</td>
<td>Fail Safe Sensor</td>
</tr>
<tr>
<td>Description</td>
<td>Primary Use case invoked by the Fail Safe Sensor. The objective of this use case is to increase the response time for a requested deceleration. As a result the collision calculations will have to be adjusted to maintain the system priorities.</td>
</tr>
<tr>
<td>Type</td>
<td>Primary</td>
</tr>
<tr>
<td>Includes</td>
<td>Adjust Pedestrian Collision Avoidance Algorithm</td>
</tr>
<tr>
<td>Extends</td>
<td>N/A</td>
</tr>
<tr>
<td>Cross-refs</td>
<td>4, 4.1</td>
</tr>
<tr>
<td>Use cases</td>
<td>Adjust Pedestrian Collision Avoidance Algorithm</td>
</tr>
</tbody>
</table>

**Table 4:** Use case description for *Engage Fail Safe Mode*
### Use Case: Accelerate to Steady State

**Actors:** Acceleration By Wire System  
**Description:** When the sensor no longer detects any pedestrians in the way, the system will accelerate back to steady state.  
**Type:** Primary  
**Includes:** Pedestrian Collision Avoidance Algorithm  
**Extends:** N/A  
**Cross-refs:** 3  
**Use cases:** Pedestrian Collision Avoidance Algorithm

*Table 5: Use case description for *Accelerate to Steady State*

### Use Case: Turn On System

**Actors:** Driver  
**Description:** The driver is able to turn on the system through a dashboard screen.  
**Type:** Primary, Essential  
**Includes:** N/A  
**Extends:** N/A  
**Cross-refs:** 7  
**Use cases:** N/A

*Table 6: Use case description for *Turn On System*
<table>
<thead>
<tr>
<th>Use Case:</th>
<th>Disable System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors:</td>
<td>Driver</td>
</tr>
<tr>
<td>Description:</td>
<td>The driver is able to turn off the system through a dashboard screen. When the system is disabled Avoid Collision will not initiate.</td>
</tr>
<tr>
<td>Type:</td>
<td>Primary, Essential</td>
</tr>
<tr>
<td>Includes:</td>
<td>N/A</td>
</tr>
<tr>
<td>Extends:</td>
<td>N/A</td>
</tr>
<tr>
<td>Cross-refs:</td>
<td>7</td>
</tr>
<tr>
<td>Use cases:</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Table 7:** Use case description for *Disable System*

<table>
<thead>
<tr>
<th>Use Case:</th>
<th>Adjust Pedestrian Collision Avoidance Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors:</td>
<td>N/A</td>
</tr>
<tr>
<td>Description:</td>
<td>The calculations done for collision probability are adjusted to maintain zero collision despite Engage Fail Safe Mode changing requested deceleration from 200ms to 900ms.</td>
</tr>
<tr>
<td>Type:</td>
<td>Secondary</td>
</tr>
<tr>
<td>Includes:</td>
<td>N/A</td>
</tr>
<tr>
<td>Extends:</td>
<td>N/A</td>
</tr>
<tr>
<td>Cross-refs:</td>
<td>4.1</td>
</tr>
<tr>
<td>Use cases:</td>
<td>Engage Fail Safe Mode</td>
</tr>
</tbody>
</table>

**Table 8:** Use case description for *Adjust Pedestrian Collision Avoidance Algorithm*
### Use Case: Pedestrian Collision Avoidance Algorithm

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Pedestrian Collision Avoidance Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors:</td>
<td>N/A</td>
</tr>
<tr>
<td>Description:</td>
<td>The system uses the information received from the Pedestrian Detection Sensor (the speed, direction, and position of the pedestrian) to determine the probability that the vehicle would collide with the pedestrian at the vehicle’s current direction and speed. If the probability of collision is not 0, the system will Avoid Collision. If the vehicle is not at steady state velocity and the probability of collision is 0, the vehicle will Accelerate to Steady State.</td>
</tr>
<tr>
<td>Type:</td>
<td>Secondary</td>
</tr>
<tr>
<td>Includes:</td>
<td>N/A</td>
</tr>
<tr>
<td>Extends:</td>
<td>N/A</td>
</tr>
<tr>
<td>Cross-refs:</td>
<td>2, 2.1, 2.2, 2.3</td>
</tr>
<tr>
<td>Use cases:</td>
<td>Avoid Collision, Accelerate to Steady State</td>
</tr>
</tbody>
</table>

**Table 9: Use case description for Pedestrian Collision Avoidance Algorithm**

### 4.2 Domain Model and Data Dictionary

The domain model for the PCAS system in addition to the associated data dictionary are described here. To describe what the key elements are of the system and how they interact between each other, a domain model is used, which utilizes class diagram notation. For a more comprehensive outlook on the system, a data dictionary is used to provide information on each class, which includes an extensive overview of the relationships, attributes, and operations.
4.2.2 Data Dictionary

Tables 10 to 18 provide a comprehensive description of the classes provided in the domain model in addition to further examining the relationships, attributes, operations of the class. A class can have three types of relationships: association, aggregation, and generalization. An association is when one class interacts with another class and provides the ability for the other class to perform an action. An aggregation is when one class is owned by another class or the owner class being a part of the class being owned. A generalization relationship is when a class inherits traits from other classes, which can also be described as a parent-child relationship.
### PCAS

**Description:** The overarching automated system, whose main goal is to prevent pedestrian collisions. The system interacts with a sensor that detects pedestrians, which leads to the appropriate actions taken to avoid collisions. A toggle switch is also a part of the system in which it allows the system to be turned on and off.

**Export Control:** Private

**Associations:** PedestrianDetectionSensor

**Aggregations:** Has ToggleSwitch as a part

**Generalizations:** None

**Attributes:**

**Operations:** switchOn(), userDisable()

**Table 10:** Data dictionary entry for **PCAS**

---

### PedestrianDetectionSensor

**Description:** When PCAS is on, PedestrianDetectionSensor is responsible for detecting objects and gathers information about them including their position, direction of movement, and/or speed. Information is then sent to PCAA to determine if a collision can potentially happen.

**Export Control:** Private

**Associations:** PCAS, Pedestrian and PCAA

**Aggregations:** None

**Generalizations:** None

**Attributes:**

**Operations:** engage(), receivePedestrianData(pedestrian), disable()

**Table 11:** Data dictionary entry for **PedestrianDetectionSensor**
Class

### Pedestrian

**Description:** A pedestrian that comes into the radius of the PedestrianDetectionSensor. The pedestrian has attributes such as position, direction, and speed, which are gathered by the PedestrianDetectionSensor.

**Export Control:** Private

**Attributes:**
- position : vector<>  
- direction : float  
- speed : float

**Operations:**

Table 12: Data dictionary entry for Pedestrian

### PCAA

**Description:** PCAA is responsible for determining whether a collision will occur with a pedestrian detected by PedestrianDetectionSensor. If a pedestrian is detected, the vehicle may have to speed up or slow down. To achieve this, PCAA will activate BrakeByWire or control AccelerateByWire in order to adjust the speed of the vehicle according to the situation. PCAA also sends an Alert to the Driver if a pedestrian is close to collision.

**Export Control:** Private

**Attributes:**

**Operations:** engage(), analyzePedestrianData(pedestrianData), disable()

Table 13: Data dictionary entry for PCAA
<table>
<thead>
<tr>
<th>Class</th>
<th>Description: The brake by wire system of the vehicle in which it responds to requests by PCAA to slow down the vehicle to stop a potential collision with a pedestrian. This is done by applying brake torque via electro-mechanical actuators at all four wheels.</th>
<th>Export Control: Private</th>
</tr>
</thead>
<tbody>
<tr>
<td>BrakeByWire</td>
<td>Relationships:</td>
<td>Associations: PCAA</td>
</tr>
<tr>
<td></td>
<td>Generalizations: None</td>
<td>Attributes:</td>
</tr>
<tr>
<td></td>
<td>Operations: receiveAlgorithmDecision(brake)</td>
<td>Table 14: Data dictionary entry for BrakeByWire</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>Description: The acceleration system of the vehicle in which it responds to requests by PCAA to speed up the vehicle to stop a potential collision with a pedestrian.</th>
<th>Export Control: Private</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccelerateByWire</td>
<td>Relationships:</td>
<td>Associations: PCAA</td>
</tr>
<tr>
<td></td>
<td>Generalizations: None</td>
<td>Attributes:</td>
</tr>
<tr>
<td></td>
<td>Operations: receiveAlgorithmDecision(accelerate)</td>
<td>Table 15: Data dictionary entry for AccelerateByWire</td>
</tr>
</tbody>
</table>
### Alert

**Description:** An alert notification consisting of warning lights on the dashboard flash and beeping sounds over the speaker system. The alert notifies Driver if a potential collision will occur from the information provided by PCAA.

**Export Control:** Private

<table>
<thead>
<tr>
<th>Relationships</th>
<th>Associations: PCAA and Driver</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregations:</td>
<td>Is part of PCAA</td>
</tr>
<tr>
<td>Generalizations:</td>
<td>None</td>
</tr>
</tbody>
</table>

**Attributes:**

**Operations:** receiveAlgorithmDecision(brake)

**Table 16:** Data dictionary entry for Alert

### Driver

**Description:** The Driver is the operator of the vehicle and interacts with PCAS via ToggleSwitch to turn the system on or off. In addition to this, Alert notifies Driver of a potential collision with a pedestrian.

**Export Control:** Private

<table>
<thead>
<tr>
<th>Relationships</th>
<th>Associations: Alert and ToggleSwitch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregations:</td>
<td>None</td>
</tr>
<tr>
<td>Generalizations:</td>
<td>None</td>
</tr>
</tbody>
</table>

**Attributes:**

**Operations:**

**Table 17:** Data dictionary entry for Driver
### Table 18: Data dictionary entry for ToggleSwitch

<table>
<thead>
<tr>
<th>Class</th>
<th>Description: A switch that allows Driver, the operator of the vehicle, to turn PCAS on or off.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Export Control: Private</td>
<td></td>
</tr>
<tr>
<td>Associations: Driver</td>
<td></td>
</tr>
<tr>
<td>Aggregations: Is a part of PCAS</td>
<td></td>
</tr>
<tr>
<td>Generalizations: None</td>
<td></td>
</tr>
<tr>
<td>Attributes:</td>
<td></td>
</tr>
<tr>
<td>● setting: boolean</td>
<td></td>
</tr>
<tr>
<td>Operations: receiveSystemOn(), receiveSystemOff()</td>
<td></td>
</tr>
</tbody>
</table>

4.3 Sequence Diagrams

In this section, several sequence diagrams are included. These diagrams are intended to display the systems classes and objects as well as how they interact for a given scenario. The scenarios included for the display interactions for basic and exceptional flow for the system. For each object, its lifeline is represented as a dashed vertical line with messages being displayed by a horizontal line between interacting entities, which points in the direction of the receiving entity.

4.3.1 No Pedestrian, Normal Use

Figure 4 shows the scenario in which the PCAS system is activated and no pedestrian is detected in use. This is a base case for the system and is initiated when the car starts. The figure below shows the sequence of events following the activation of the system with no pedestrians involved.
4.3.2 Static Pedestrian is Detected

This scenario captures the interactions between the system when a static pedestrian is detected by the pedestrian sensor. No action is needed to avert the pedestrian.

4.3.3 Pedestrian is Moving then Stops

Figure 6 exhibits the interactions between a pedestrian and the system for a scenario where the pedestrian is moving at first, then becomes static. The system take action to avert the moving pedestrian, then once the pedestrian becomes static, the algorithm accelerates to the normal speed.
Figure 6: This scenario shows the interaction between the pedestrian, the sensors, and the algorithm. Additionally, interactions between the algorithm and the systems that are used when the algorithm decides that action must be taken are displayed. In this scenario, once the algorithm calls analyzePedestrianData(pedestrianData) since the pedestrian is in motion the algorithm messages the alert and braking objects that braking is needed. While a pedestrian is detected the algorithm will analyze the data. Once the pedestrian stops, the passenger is clear and the system accelerates back to normal speed.
4.3.4 Pedestrian is Static then Moving

Figure 7 exhibits the interactions between a pedestrian and the system for a scenario where the pedestrian is static at first, but then begins to move. The system takes action once the pedestrian begins to move.

![Diagram showing pedestrian interactions](image)

**Figure 7:** A scenario similar to 4.3.3 where the sensor is receiving pedestrian data. However the pedestrian is static at first and the system takes no action until the pedestrian begins to move. The alert and brake receive the decision from the algorithm.

4.3.5 User Disables System

The figure below shows how the system shuts down after the driver has manually disables the PCAS system. The system receives the users disable command from the toggle switch and disables itself.
Figure 8: Sequence Diagram Similar to figure four. The driver manually attempts to disengage the system via the toggle switch. The system receives this from the switch and disables itself.

4.4 State Diagrams

This section introduces the state diagrams, which are used to depict what states an object can be in while showing its behavior. The beginning state of a diagram is a black dot with an arrow leading to a state. States of a system are blue boxes and the arrow between two states represents how one state transitions into the next.

4.4.1 PCAS System

Figure 9 shows the state diagram for the PCAS system. The vehicle initially starts Off, which can be turned on via ToggleButton. Once PCAS is on, it moves to a state of scanning the surroundings of the vehicle with PedestrianDetectionSensor. If an object is detected, then the system moves to Analyze Object. If the object is not a pedestrian, it returns back to the state Scan Surroundings. However, if the object is a pedestrian, the system moves to Pedestrian Collision Avoidance Algorithm, which is done using PCAA. If the probability of a collision is 0, then the state returns back to Scan Surroundings, otherwise the system moves to Braking state in which the BrakeByWire system initiates, slowing down the car. The system returns to the Scan Surroundings state and begins braking, which may lead to Vehicle Stopped. If so, then the vehicle can accelerate back to cruise control speed using AccelerateByWire once the pedestrian has moved out of the vehicle’s path to return back to Scan Surroundings.
4.4.2 Pedestrian Detection Sensor

Figure 10 shows the state diagram for the pedestrian detection sensor. The sensor begins in an off state and when it receives an on signal from PCAS, the sensor enters a state of Obtaining Data. During the Obtaining Data state, PedestrianDetectionSensor is continuously obtaining data from Object and if Object is a pedestrian, the data is sent to PCAA to determine if a collision may occur. Once PCAS sends a request to turn off the system, the PedestrianDetectionSensor turns off.
4.4.3 Brake by Wire System

Figure 11 shows the state diagram for the brake by wire system. The BrakeByWire class can either be Braking or Not Braking. The brake by wire system first starts off in a state of not braking. If the BrakeByWire class receives a signal to activate it by the PCAA class, then the brake by wire system moves to a state of continuous braking. Once the surroundings are clear, the Braking state moves back to Not Braking.

![State diagram for BrakeByWire](image)

Figure 11: State diagram for BrakeByWire

5 Prototype

We have developed the prototype of the Pedestrian Collision Avoidance System (PCAS). This section will demonstrate the system functionalities of the prototype.

5.1 How to Run Prototype

The prototype can be accessed in the Prototype section on PCAS2 team website ([https://cse.msu.edu/~beachjo3/](https://cse.msu.edu/~beachjo3/)). The prototype is developed with Lumion ([www.lumion.com](http://www.lumion.com)) and is demonstrated in videos.

The prototype consists of 4 components: pedestrian, driver, sensor, brake and acceleration by wire system. The sections below explain each component in detail.

5.1.1 Pedestrian

Pedestrians are an obstacle that the PCAS system is going to help the driver to avoid.

5.1.2 Driver

A driver is needed to interact with the system.

- Turn on the system
The driver is allowed to turn on the PCAS system through a dashboard screen. Once the system is turned on, the driver will be alerted when the system detects pedestrians.

- **Disable the system**

The driver is allowed to disable the PCAS system through a dashboard screen. Once the system is disabled, the system will stop detecting pedestrians or alerting the driver.

### 5.1.3 Sensor

Sensors on the side of the vehicle are to detect pedestrians. This section describes the component of the sensor in detail.

- **Pedestrian detection**

  Detects pedestrians and calculates collision probability.

- **Fail Safe Mode**

  Invoked by Fail Safe Sensor to increase the response time for a requested deceleration.

### 5.1.4 Brake and Acceleration by Wire System

Wire system helps accelerate and decelerate the vehicle according to the collision probability. When the pedestrian enters the collision zone of the vehicle, the Wire System decelerates the vehicle to avoid collision. When the pedestrian is no longer detected, the Wire System accelerates the vehicle back to steady state.

### 5.2 Sample Scenarios

This section demonstrates 5 scenarios of the PCAS prototype. A description of each respective scenario is provided. Figures 12 and 13 capture examples of the prototype for the scenarios with pedestrian involvement [4].

1. **No pedestrian, no effect**
   a. There is no pedestrian present in front of the vehicle, the vehicle's velocity remains the same and there is no action from the system.

2. **Pedestrian moves in front of the vehicle and then stops.**
   a. The system warns the driver, the system decelerates to 0 velocity before reaching the pedestrian
Figure 12: The image shows the pedestrian after they have stopped in front of the vehicle. The system avoids hitting the pedestrian and comes to a complete stop.

3. Pedestrian is in front of the vehicle and moves out of the way
   a. The system warns Driver of a pedestrian, the system engages the brake by wire system to keep velocity of 0 while the pedestrian is in front of the car. The pedestrian moves out of the way, the system engages acceleration by wire system to accelerate to steady state.
Figure 13: The pedestrian has moved from the path of the vehicle and the PCAS accelerates back to a steady state.

4. Fail Safe Mode Engaged
   a. The fail safe sensor is activated and alerts the system. The prototype adjusts its collision calculations in order to increase the response time for the system. Once the prototype has engaged failsafe mode its response time to reach a requested deceleration increases from 200ms to 900ms.

5. Cyber Attack Prevented
   A 3rd party has launched a cyber attack on the PCAS prototype. The prototype uses the following to prevent and mitigate the attack.
   a. Digital certificates to encrypt information and minimized communication hardware in order to prevent 3rd party access to communication.
   b. The system will check for malicious code on top of using memory protection units to prevent and mitigate 3rd party exploits.

6 References


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.