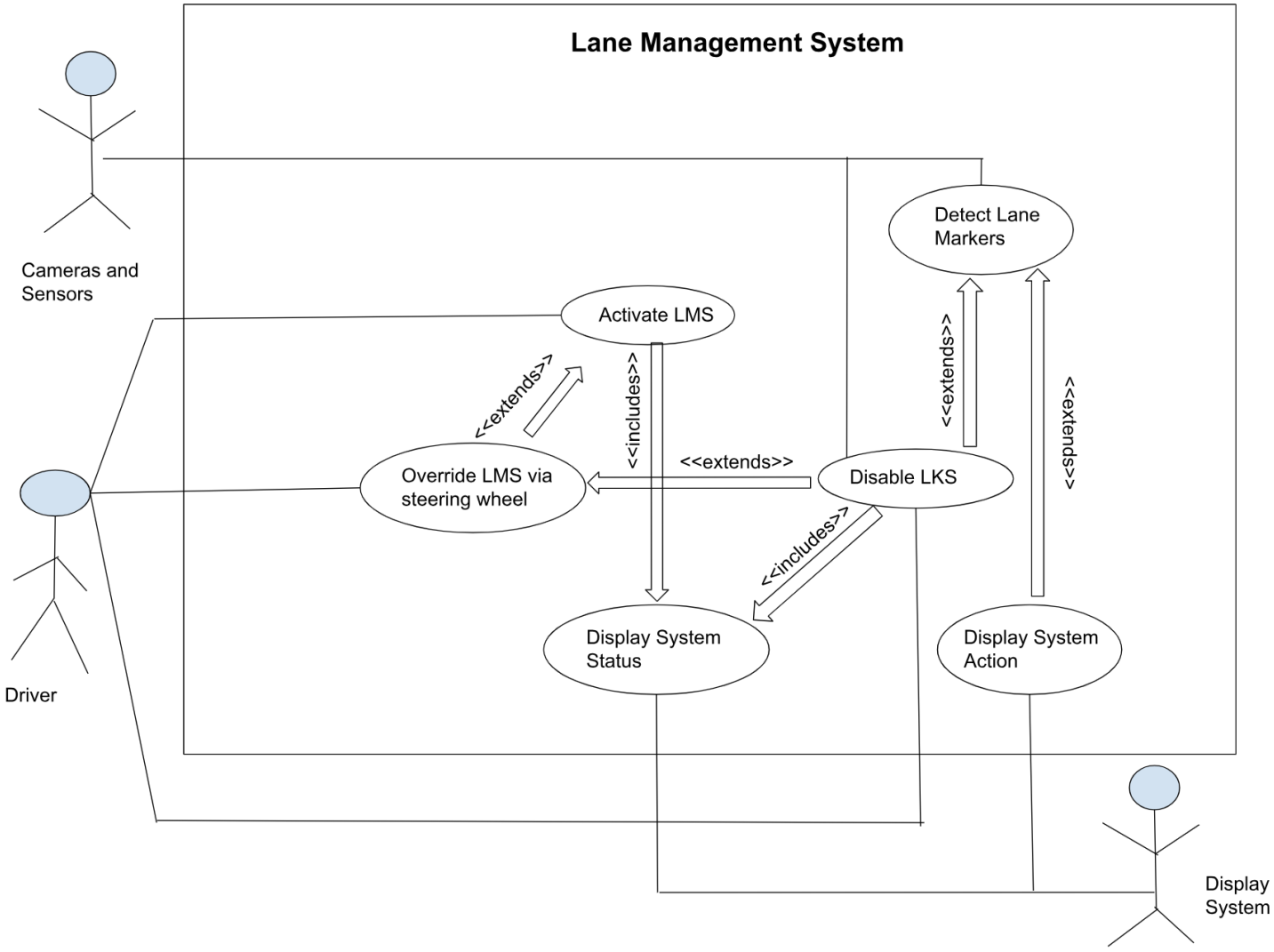
Don Truong, Evan Stanislaw, Kaijia Zhan, Katelyn Hurst

CSE435 Software Engineering

Intermediate Project Assignment (due Nov 1, 2023):

USE CASE DIAGRAM



Use case description:

Use Case:	***Name: Activate LMS***
Actors:	***Name of actors interacting with current use case*** **Driver, Cameras & Sensors**
Description:	***Description of the goals, objective, services provided by use case*** **The LMS shall activate at a speed of 35 miles per hour** **While active, the system shall override gently to ensure the vehicle stays within the lanes detected by the sensors**
Type:	***Primary (essential) or secondary*** **Primary**
Includes:	***Name of use cases that the current use case includes (common goal shared by 2 or more use cases)*** **Display system status**
Extends:	***Name of use cases that the current use case extends (the name of the normal scenario use case that the current use case extends)***
Cross-refs:	***Number of requirement that the current use case addresses*** **1a, 1b, 1c, 3a, 3b,**
Use cases:	***Enumerate use cases that are dependent upon the current one.*** **Detect Lane Markers**

Use Case:	***Disable LKS***
Actors:	***Driver, Cameras & Sensors***
Description:	***The driver shall disable LKS via the UI or by applying a torque on the steering wheel.*** ***LKS shall be disabled when cameras & sensors can't detect lane***

	markers. ***LKS shall be disabled when the vehicle is traveling under 35 mph.***
Type:	***Primary***
Includes:	**Display system status**
Extends:	***Detect lane markers*** ***Override LMS via steering wheel***
Cross-refs:	**1d, 1e, 2f**
Use cases:	**Display *system* status**

Use Case:	***Override LMS via steering wheel***
Actors:	***Driver***
Description:	***The system shall detect the torque applying on the steering wheel to gently steer the car back into the center of the lane*** ***When the driver applies torque to the wheel, then the LKS will disable itself***
Type:	***Primary***
Includes:	
Extends:	***Activate LMS***
Cross-refs:	**1b, 1c, 1e, 2e**
Use cases:	**Disable LKS, *Activate LMS***

Use Case:	***Detect lane markers***
Actors:	***Cameras & Sensors***
Description:	**Cameras & Sensors provide the information of lane markers for the system in a frequently basis** **The information provided by the sensors shall be used to predict the given path of the vehicle**

Type:	***Primary***
Includes:	
Extends:	
Cross-refs:	**2a, 3c, 3g, 3d, 3b, Secondary Requirement 3**
Use cases:	***Disable LKS, Activate LMS***

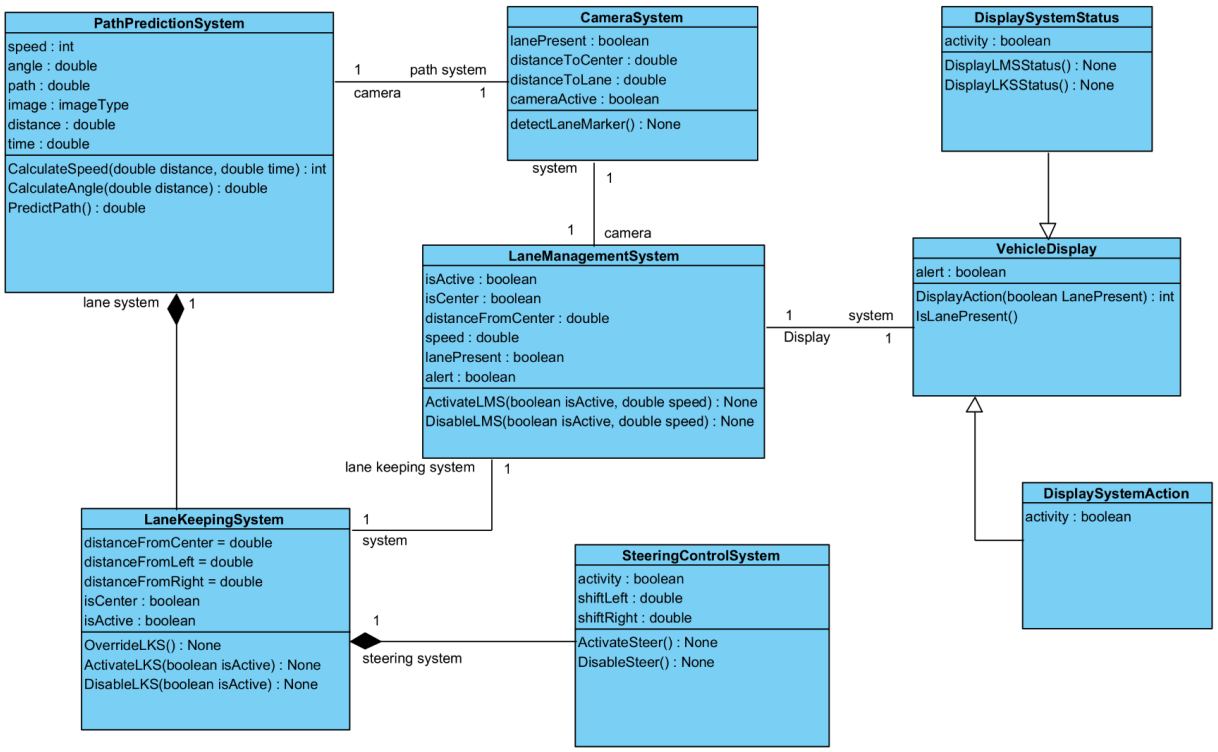
Use Case:	***Display system status***
Actors:	***Display System***
Description:	**There should be a light on to display if the system is on.** **LKS on then everything has to be on.** **LKS can be off and departure warnings can still be present.**
Type:	***Secondary***
Includes:	
Extends:	
Cross-refs:	**Secondary Requirement 3**
Use cases:	

Use Case:	***Display system action***
Actors:	**System**
Description:	**There shall be warning light in the panel to let the driver know when the system is about to intervene** **The warning light shall display the direction of the system's action** **The system shall warn the driver when the vehicle is about to cross the lane**
Type:	***Primary***
Includes:	

Don Truong, Evan Stanislaw, Kaijia Zhan, Katelyn Hurst

Extends:	***Detect lane markers***
Cross-refs:	**2b,2c, 2d, 2g, 3g, Secondary Requirement 1**
Use cases:	***Activate LMS***

LMS DOMAIN MODEL



Data dictionary:

Class	SteeringControlSystem	
Name	***Description (responsibilities)*** The system shall help steer the bar back into the center	
	Export control (public: yes/no) No	
	Relationships	Associations: LaneKeepingSystem
		Aggregations:
		Generalization:
	List of attributes and their primitive types activity = boolean shiftLeft = double shiftRight = double	
	List of operations (include parameters and results) ActivateSteer() return None DisableSteer() return None	

Class	Camera System
Name	***Description (responsibilities)*** The camera system shall detect the lanes present and send data to the path prediction system.

	Export control (public: yes/no) No	
	Relationships	Associations: LaneManagementSystem, PathPredictionSystem
		Aggregations:
		Generalization:
	List of attributes and their primitive types lanePresent = boolean distanceToCenter = double distanceToLane = double cameraActive = boolean	
	List of operations (include parameters and results) detectLaneMarker() return None	

Class	Path Prediction System	
Name	***Description (responsibilities)***	
	Export control (public: yes/no) no	
	Relationships	Associations: CameraSystem
		Aggregations: LaneKeepingSystem

		Generalization:
	List of attributes and their primitive types speed = int angle = double path = double image = imageType distance = double time = double	
	List of operations (include parameters and results) CalculateSpeed(distance, time) return int CalculateAngle() return double PredictPath() return double	

Class	VehicleDisplay	
Name	***Description (responsibilities)*** The vehicle display alerts the driver of the current system's status	
	Export control (public: yes/no) yes	
	Relationships	Associations: LaneManagementSystem
		Aggregations:
		Generalization: DisplaySystemStatus, DisplaySystemStatus

	List of attributes and their primitive types alert = boolean
	List of operations (include parameters and results) DisplayAction(boolean LanePresent) return int IsLanePresent() return boolean

Class	Display System Status	
Name	***Description (responsibilities)*** It displays whether the system is active or not	
	Export control (public: yes/no) no	
	Relationships	Associations:
		Aggregations:
		Generalization:
	List of attributes and their primitive types activity = boolean	
	List of operations (include parameters and results) DisplayLMSStatus() return None DisplayLKSStatus() return None	

Class	LaneKeepingSystem	
Name	***Description (responsibilities)*** Keep the car in the center of the lane	
	Export control (public: yes/no) no	
	Relationships	Associations: LaneManagementSystem
		Aggregations: PathPredictionSystem, SteeringControl
		Generalization:
	List of attributes and their primitive types distanceFromCenter = double distanceFromRight = double distanceFromLeft = double isCenter = boolean isActive = boolean	
	List of operations (include parameters and results) OverrideLKS() return None ActivateLKS(boolean isActive) return None DisableLKS(boolean isActive) return None	

Class	LaneManagementSystem

<table>
<tr><td rowspan="5">Name</td><td colspan="2">Description (responsibilities) Manages the overall system to ensure the car is in the center through the usage of camera path prediction system, LKS, and vehicle display</td></tr>
<tr><td colspan="2">Export control (public: yes/no) yes</td></tr>
<tr><td rowspan="3">Relationships</td><td>Associations: CameraSystem, VehicleDisplay, LaneKeepingSystem</td></tr>
<tr><td>Aggregations:</td></tr>
<tr><td>Generalization:</td></tr>
<tr><td colspan="2">List of attributes and their primitive types
isActive = boolean
distanceFromCenter = double
isCenter = boolean
speed = double
lanePresent = boolean
distanceToCenter = double
alert = boolean</td></tr>
<tr><td colspan="2">List of operations (include parameters and results)
ActivateLMS(boolean isActive, double speed) return None
DisableLMS(boolean isActive, double speed) return None</td></tr>
<tr><td></td><td colspan="2"></td></tr>
</table>