College of Engineering

CS Capstone Design Document

December 1, 2017

Axolotl

Prepared for

Kevin McGrath ______________ Signature ______ Date

Prepared by

Group 15

Team Wombat

Victor Li ______________ Signature ______ Date

Ryan Crane ______________ Signature ______ Date

Nicholas Wong ______________ Signature ______ Date

Abstract

This document is a software design description (SDD) that discusses the design of the Axolotl system with respect to navigation, media, cameras, and data logging functions as well as user interface and background operations. The document will cover our project stakeholder's design concerns and requested functionality, the viewpoints associated with these concerns and requests, and the design views and testing methods developed from and governed by those viewpoints. Rationale will be offered to explain the choices made in design views.

CONTENTS

1 Introduction

This software design description concerns the Axolotl system, an infotainment and black box data logging system that is based on the NVIDIA Jetson TX2 hardware. Our stakeholder is Instructor Kevin McGrath of Oregon State University, who is our project client. The Axolotl system can be divided into individual components, including: navigation, media, data logging, camera, user interface, and supplemental functionality. This SDD will first list all stakeholder concerns with respect to each component. We will then associate viewpoints to the concerns listed; afterwards, we will develop design views that are governed by these viewpoints. An individual viewpoint may address one or more concerns; viewpoints will be directly used to form a design view. Design rationale will be given stating the rationale behind the design view of each component, and testing solutions will be offered in design overlays.

2 Design Stakeholders and Their Concerns

2.1 Navigation

2.1.1

Routing and mapping functionalities of the navigation system must be able to work without an internet connection.

2.1.2

Users must be able to enter a street address as a destination and navigate to destination.

2.1.3

Users must be able to cancel their route if one is active.

2.1.4

Navigation must utilize a GPS to determine vehicle location.

2.2 Media

2.2.1

FM radio playback will be supported and must also display RDS information to the screen.

2.2.2

The Axolotl Media menu will allow for the playing of music through USB devices.

2.2.3

The Media menu will allow the connection to a network drive to download media online.

2.2.4

Users will be able to play music on the Axolotl through handheld devices via Bluetooth or Auxiliary connections.

2.2.5

Users will be able to select media from an onboard storage medium.

2.3 Data Logging

2.3.1

The user interface will allow users to reset the black box and delete all of the data logged.

2.3.2

Any data logged must be transferable to a removable storage medium.

2.3.3

The system must require user authentication in order to gain access to the specified download and reset functionality.

2.3.4

Users should be able to change their authentication credentials for the black box system..

2.3.5

The data logging system must log vehicle data.

2.3.6

The data logging system will only log data that is relevant to the act of driving and driver well-being.

2.4 Cameras

2.4.1

The Axolotl will support viewing and logging for both dashboard mounted cameras and a backup camera.

2.4.2

Backup cameras will connect to the main Axolotl system wirelessly.

2.5 User Interface

2.5.1

The user interface will utilize a touchscreen to receive input from the user.

2.5.2

The user interface will automatically display the backup camera feed when the car is in reverse.

2.6 Supplemental and Background Functionality

2.6.1

The Axolotl should boot up using power from a UPS and switch over to vehicle power when the vehicle is running.

2.6.2

The Axolotl should go into a low power state when the vehicle is turned off. After 48 hours, the system must power itself down.

2.6.3

The Axolotl will allow the user to include supplemental turn signals throughout the vehicle.

3 DESIGN VIEWPOINTS

3.1 Navigation

3.1.1 Interface

Concerns: 2.1.2, 2.1.3

Elements: Open Source Qt, NavIt

Analytical Methods: The navigation system must be able to display routing-critical information to the screen and allow for touch-based address/destination entry, route cancellation, and interaction with the map.

Viewpoint Source: Victor Li

3.1.2 Routing and Mapping

Concerns: 2.1.2

Elements: NavIt

Analytical Methods: The navigation system must be able to accept a U.S. address as the destination and successfully guide the user to that destination with turn-by-turn instructions.

Viewpoint Source: Victor Li

3.1.3 Offline Data Storage

Concerns: 2.1.1

Elements: NavIt, OpenStreetMaps

Analytical Methods: The navigation system must store offline map data for all 50 U.S. states and use this data for routing and mapping.

Viewpoint Source: Victor Li

3.1.4 GPS

Concerns: 2.1.4

Elements: NavIt, GPSD, NEO M8U

Analytical Methods: GPS should be used to determine vehicle location for navigation. The navigation system should be able to compensate for the loss of GPS signal.

Viewpoint Source: Victor Li

3.2 Media

3.2.1 Interface

Concerns: 2.2.1, 2.2.2, 2.2.3, 2.2.4, 2.2.5

Elements: Kodi

Analytical Methods: The media menu of the Axolotl should allow the user to select a medium to play media from. Each selection should introduce similar menus which display the information of the current media selection.

Viewpoint Source: Nick Wong

3.2.2 Remote Access

Concerns: 2.2.3, 2.2.5

Elements: TCP/IP

Analytical Methods: The media menu should allow for the downloading of media from network drives. This media can then be selected from the onboard storage option.

Viewpoint Source: Nick Wong

3.3 Data Logging

3.3.1 Graphical User Interface

Concerns: 2.3.1, 2.3.2, 2.3.3, 2.3.4

Elements: Open Source Qt

Analytical Methods: The data logging system should offer a menu for users to manage the system and its data.

Viewpoint Source: Victor Li

3.3.2 Physical Interface

Concerns: 2.3.5, 2.3.6

Elements: OBDII, OBDLink MX Bluetooth

Analytical Methods: The data logging system should interface with vehicle computers and collect vehicle data.

Viewpoint Source: Victor Li

3.4 Cameras

3.4.1 Interaction

Concerns: 2.4.1

Elements: Open Source Qt

Analytical Methods: Cameras will be viewable on the touchscreen, and conditionally recorded to the data log.

Viewpoint Source: Ryan Crane

3.4.2 Dependency

Concerns: 2.4.2

Elements: Raspberry Pi

Analytical Methods: Backup camera must communicate with main Axolotl unit wirelessly.

Viewpoint Source: Ryan Crane

3.5 User Interface

3.5.1 Interface

Concerns: 2.5.1

Elements: Open Source Qt

Analytical Methods: The user interface must allow the user to access the media, navigation, and logging features provided by the Axolotl.

Viewpoint Source: Ryan Crane

3.5.2 Information

Concerns: 2.5.2

Elements: Open Source Qt

Analytical Methods: The screen must automatically display the backup camera feed when the car is shifted to reverse and return to normal display when the car shifts out of reverse.

Viewpoint Source: Ryan Crane

3.6 Supplemental and Background Functionality

3.6.1 UPS Power

Concerns: 2.6.1

Elements: PowerStream UPS

Analytical Methods: A power source independent from vehicle power must be used to supply the Axolotl with power during bootup and standby/shutdown sequences. The system must be able to remain active in a low power state for 48 hours after the ignition is returned to the off position.

Viewpoint Source: Victor Li

3.6.2 Turn Signals

Concerns: 2.6.2

Elements: Raspberry Pi, Bluetooth

Analytical Methods: There should be additional turn signals which are connected to the Axolotl wirelessly. These turn signals should light up when the user decides to signal to turn in a direction.

Viewpoint Source: Nick Wong

4 Design Views

4.1 Navigation

4.1.1 Routing, Mapping, Turn-by-Turn Directions

The Axolotl must be able to navigate drivers to a specific destination in the United States. To do this, the system will need to offer mapping, routing, and navigate-to-destination functionality with turn-by-turn directions. These features will be provided by the open-source NavIt navigation software, which offers all of these capabilities in a ready-to-install package for Linux. The NavIt software will be encapsulated within a graphical wrapper within our Axolotl system in order to integrate it with our system.

4.1.2 Offline Map Data

The Axolotl must be able to navigate without using online sources for map and street data. To implement this, we will download OpenStreetMap data for the United States in the .osm file format and store it locally on internal storage. The NavIt navigation software we are using will then be pointed to this offline data store and utilize it to provide routing, mapping, and turn-by-turn directions; the system will not attempt to query an online map database for the same data.

4.1.3 Location Accuracy

Navigation requires a near-to exact understanding of vehicle location in order to provide proper route guidance and turn-by-turn instructions by voice. The primary location method we will use is GPS by way of a GPS receiver connected to the Axolotl, supplemented by a dead reckoning system built into the GPS unit. Navigation will rely on the GPS receiver to supply true GPS data so long as GPS signal is available; the system will automatically resort to dead reckoning of the GPS signal is lost.

4.2 Data Logging

4.2.1 Vehicle Connection Method

The data logging functionality of the Axolotl requires intimate knowledge of vehicle data from the wide array of sensors installed in a modern vehicle. Modern vehicles sold in the United States collect this data and make it available via an OBDII port for reading. We will use a Bluetooth OBDII connector to interface with the OBDII port and then have the Axolotl connect to the OBDII connector via a Bluetooth connection, using OBDII software to sample the OBDII vehicle data.

4.2.2 Data Formatting

OBDII and AHRS will need to be formatted in a readable manner for the black box portion of the Axolotl. All data will be formatted in the .csv file format. Each column of the .csv file represents a data field; each data snapshot will add one new entry for each data field and thus will add one row to the .csv file. The OBDII data fields selected for data logging include: calculated engine load, engine coolant temperature, engine speed, vehicle speed, throttle position, relative throttle position, run time since engine start, fuel tank level, ambient air temperature, and absolute barometric pressure. The AHRS data fields selected for data logging include vehicle roll, pitch, and yaw. The units utilized for each OBDII data field will be defined by the SAE J1979 standard for OBDII PIDs; the AHRS data fields will be measured in degrees.

4.2.3 Data Storage

Logged data will need to be stored on some sort of internal storage medium on the Axolotl itself. A 1TB hard disk drive connected to the Axolotl via a USB port will be used to provide data storage. This hard disk drive will be formatted with an ext4 file system and will have one directory labeled *black_box* that will hold the .csv data log and the .mp4 files from the dashcam system.

4.2.4 User Interface

Users must be able to manage the data logging system. Specifically, users must be able to wipe all of the data logged or download the black box data to a removable storage medium, and must only be able to access these functions with authentication. A graphical user interface will be supplied in order to allow for this functionality. The interface will prompt the user for a password, entered with a touch-based QWERTY keyboard. Passwords cannot be entered while the vehicle is in motion. The default password used for the Axolotl system is *ltoloxa*. Successful authentication will grant access to two options: change black box password and wipe all data. Leaving this interface and returning will require users to enter their password again.

4.3 Media

4.3.1 Media Player

Upon selection of the media menu, a new interface will be brought up to allow users to select mediums through which to play their media. The selections include USB through a flash drive or phone, Auxiliary, Bluetooth, FM, or onboard HDD media. The media player will be integrated using Kodi software. The Kodi software will allow the users to browse media as well as play from multiple media formats.

4.3.2 Network Media Download

Users will also have the option to download media from a network drive directly onto the onboard HDD when there is an internet connection available. This will be set up through a TCP/IP P2P network connection. This requires the user to go through an initial setup to connect to their desired computer. Upon selecting this option first, the user will have a choice to setup a new computer to connect with. A menu will then be brought up, guiding the user through connecting to their computer over a network. Once the initial setup is complete, a new computer will be added to a list of existing devices that the user can then choose from.

4.4 Cameras

4.4.1 Camera Feed

The Axolotl will be able to integrate footage from a dashcam into its black box log, as well as utilize a backup camera as a standard infotainment system would. The backup camera feed will be displayed automatically when needed and hidden otherwise. Footage from dashboard mounted cameras will be recorded and archived in the black box data log.

4.5 User Interface

4.5.1 Touchscreen

The Axolotl will need to display information to its user and be able to receive input so that they can utilize the functions it offers. The Axolotl will use a ten inch touchscreen for this purpose. The screen will be used to display maps and routing information as well as track information from the various music sources it will support. The touchscreen will also allow the user to navigate between menus. To build our user interface we will use a tool called Open Source Qt. Open Source Qt is an open source toolkit for making graphical user interfaces that we can use to design a fast familiar interfaces that users can learn to use quickly.

4.5.2 Camera integration

The user interface will be superseded by the backup camera feed when the car is shifted to reverse. In general Infotainment systems like the Axolotl automatically display backup camera feeds while the car is in reverse, and return to the normal menus when the car leaves reverse.

4.6 Supplemental and Background Functionality

4.6.1 UPS Power

The Axolotl requires power for booting up on vehicle startup and remaining in standby mode for 48 hours after the vehicle has been turned off.

4.6.2 Supplementary Turn Signals

The Axolotl will include additional turn signals which will attach onto the outside of a vehicle. These turn signals will be controlled by a Raspberry Pi which will communicate with the Axolotl wirelessly via bluetooth. Once the Axolotl receives information on the turn signals, it will send a signal to the respective turn signal allowing it to flash.

5 Design Elements

5.1 All

5.1.1 NVIDIA Jetson TX2

Type: System

Purpose: The NVIDIA Jetson TX2 will be used as the computing system hardware that will run all of the software developed for all Axolotl subsystems.

5.2 Navigation

5.2.1 NEO M8U

Type: System

Purpose: This is the GPS module that the Axolotl will utilize to drive the navigation system.

5.2.2 NavIt

Type: Subprogram

Purpose: NavIt will be used to provide all navigation functionality including offline routing, offline mapping, navigate-to-destination, and turn-by-turn directions. The subprogram will also handle user input for address entry and route cancellation.

5.2.3 OpenStreetMaps

Type: System

Purpose: OpenStreetMaps will be used to provide the street and map data used for routing and mapping.

5.2.4 Open Source Qt

Type: Framework

Purpose: Open Source Qt will be used to create the Axolotl user interface in addition to the Axolotl graphical wrapper for the NavIt system.

5.3 Data Logging

5.3.1 OBDII

Type: Port

Purpose: OBDII will be used to read data from an automotive OBDII port for data logging.

5.3.2 OBDLink MX Bluetooth

Type: Connector

Purpose: The OBDLink MX Bluetooth will collect and aggregate OBDII data, making that data available to the Axolotl via a Bluetooth connection.

5.3.3 pyOBD

Type: Subprogram

Purpose: pyOBD will be used to interpret vehicle data from an OBDII connector and translate it into readable text that can then be formatted and logged.

5.3.4 OBDSim

Type: Subprogram

Purpose: OBDSim will be used to simulate OBDII data in order to test the data logging system for correct functionality.

5.3.5 Open Source Qt

Type: Framework

Purpose: Open Source Qt will be used to create a graphical user interface that allows for user interaction with the black box system.

5.4 Media

5.4.1 Grove I2C FM Receiver

Type: Component

Purpose: This will be used to receive FM radio signals, including Radio Data System data.

5.4.2 Kodi Media Player

Type: System

Purpose: This media player will be used to allow for the playing of multiple file formats.

5.4.3 I2C

Type: Port/Interface

Purpose: This serial computer bus standard will be used to connect the Grove I2C FM receiver with the Jetson TX2 computer module.

5.4.4 TCP

Type: System

Purpose: TCP will be used to download music files from a host computer over a network connection in order to implement networked music library functionality.

5.5 Cameras

5.5.1 Raspberry Pi

Type: System

Purpose: A Raspberry Pi will be used to control the backup camera and wirelessly stream its video feed to the Axolotl.

5.6 User Interface

5.6.1 Open Source Qt

Type: Framework

Purpose: Open Source Qt is an open source toolkit for creating graphical user interfaces. Open Source Qt will be used to create the Axolotl user interface.

5.7 Supplemental and Background Functionality

5.7.1 Raspberry Pi

Type: System

Purpose: A Raspberry Pi will be used to communicate with the Axolotl, and control light signals outside of the car.

5.7.2 PowerStream UPS

Type: Component

Purpose: The PowerStream PST-DCUPS-12A-N13.1-P10 is a UPS that will be used to supply power to the Axolotl during vehicle startup or vehicle shutdown.

5.7.3 Bluetooth

Type: System

Purpose: The Raspberry Pi will communicate wirelessly with the Axolotl using Bluetooth.

6 Design Rationale

6.1 Navigation

OpenStreetMap data in the .osm file format was chosen for offline street and map data as the OpenStreetMap project is the most extensive open-source database for street and map data. The NEO M8U was chosen as the best GPS module because it provides built in support for dead reckoning and logging applications without sacrificing anything in precision or speed. The use of the dead reckoning system on the chosen GPS receiver supplements the offline map database with the ability to compensate for the loss of GPS signal, therefore giving our system the additional ability to calculate vehicle location offline. The NavIt software was chosen for navigation based on its proven use in a variety of applications ranging from portable handheld navigation devices to Linux-based car computers. The NavIt system was built with offline capability as one of its core features and provides the functionality of a full navigation suite comparable to that offered by a car manufacturer.

6.2 Media

The decision to use Kodi software is because of the supported media formats it plays. It is an open source software that has a large library of documentation for the process of integrating the media player into a system. The decision to allow the downloading of media through a network connection was derived from the possibility of a user wanting to be able to access a library they have already compiled. The user simply needs an internet connection, and any libraries on connected devices will be available.

6.3 Data Logging

The decision to utilize OBDII for reading most vehicle data is due to the prevalence of the port in vehicles sold in the United States, thus using the OBDII port will maximize the compatibility. A password was chosen as the authentication solution as it is one of the most secure solutions and would only require a text entry field and QWERTY touch keyboard to implement.

The range of data fields selected for data logging were chosen based on their relevance to driving and driver well-being. The OBDII port defines a variety of vehicle data parameters; most of this data is not useful for driver information nor accident investigation. Some of these irrelevant data fields include: fuel rail pressure, mass air flow sensor flow rate, and time since trouble codes cleared. Irrelevant fields will be excluded from the .csv black box log. The same approach is taken with the data from the AHRS system.

The decision to use a hard disk drive as the storage medium for data logging functionality stems from its favorable form factor, relatively low price, and its proven reliability. The USB connection method also makes the hard drive easily removable, enabling users to connect their drive to a computer and gain access to their black box data log.

6.4 Cameras

Design decisions surrounding camera integration is driven by the need for wireless communication and flexibility for the system to log footage captured by the cameras. External tools will need to be used in order to manage wireless video streaming which will influence project design.

6.5 User Interface

User interface design is driven by the importance of usability and aesthetics in the project. Open Source Qt is a popular tool for developing graphical user interfaces for Linux applications. Our application will run on Linux since it is distributed with the Jetson TX2. Open Source Qt is a toolkit with excellent support for automotive user interfaces.

6.6 Supplemental and Background Functionality

The decision to include supplemental turn signals is to ensure safety in situations where vehicle lights are obscured and difficult to see by other drivers. This may arise if the vehicle running the Axolotl system is towing a trailer or is very long. A Raspberry Pi was chosen to manage these lights as it is an inexpensive and compact solution. The decision to use the PowerStream UPS was made based on its ability to detect vehicle alternator status and then switch to vehicle power when the alternator is on or switch to internal power if the alternator is shut off.

7 Design Overlays

7.1 Navigation

Our navigation system will be tested using functional testing. With the Axolotl system installed in a vehicle, the navigation system will be set to navigate to a predetermined destination. The driver will attempt to drive to the destination, obeying any and all traffic laws. This will be tested against a smartphone running Google Maps with a destination set to the same street address as the predetermined destination entered into the Axolotl. Implementation of navigation will be judged as successful if both systems end at the correct address.

7.2 Data Logging

Data logging will be tested using functional testing. The testing method is a simulation test; OBDSim will be used to generate a set of fake OBDII data that will be logged by the Axolotl. Both data sets will be compared; if data fields that share the same timestamp are equal, then the data logging functionality has been correctly implemented. The same method will be done using a custom-built program for the AHRS system.

8 Appendices

8.1 Appendix I: Glossary of Terms

- NVIDIA Jetson TX2: A versatile, efficient, and high-performance computer made by NVIDIA to be used in robots, drones, and smart cameras.
- OBDII: On-Board Diagnostics II, a standardized connector installed in all automobiles sold in the United States since January 1st, 1996. Devices connected via an automotive OBDII port can read vehicle sensor data on the fly.
- LIDAR: Light Detection and Ranging, a method of using laser pulses to determine the 3D properties of a faraway object.
- AHRS: Attitude, Heading, and Reference System, a system used in modern aircraft to determine and display roll, pitch, and yaw.
- Dead Reckoning: A method of calculating the current location of a person or an object by using a previously known location and then using known or estimated speed, directional heading, and elapsed time.
- Infotainment: A portmanteau of ”information” and ”entertainment”. When we use the term ”infotainment”, we are referring to the center console touchscreen that gives drivers access to information and media in modern cars.
- RDS: Radio Data System, a method of transmitting the current track information of an FM radio broadcast.
- UPS: Uninterruptible Power Supply, an auxiliary power source that enables a device to function for a limited time if its main power source is unavailable.
- USB: Universal Serial Bus, a commonplace connection method for a multitude of computer peripherals.
- MP3: A common digital audio coding format that uses lossy compression. MP3 files are given the .mp3 file extension.
- MP4: A common digital video container format. MP4 files are given the .mp4 extension.
- .csv: Comma-Separated Values, a file format to store tabular data in plain text.
- TCP: Transmission Control Protocol is a method of establishing a method of communicating with devices through a network.