# **System Design Document**

# **Car-Nerd**

# 

# 

# 

# 

# **Dynamic Developers**

**4601 Mid Rivers Mall Dr**

**Cottleville, MO 63376**

# 

# 

# 

# 

# **June 27, 2020**

# 

**Table of Contents**

Introduction ……...…………………………………………………………………………………...... 3

Purpose ...……………....…………………………………………………………………………...…. 3

System Overview ……………..…………………………………………………………………..….. 3

Design Constraints ………...……………………………………………………………………..…... 3

Roles and Responsibilities………...…………………….……………………………………….…... 4

Project References ……..……………………………………………………………………..…….... 5

System Architecture ………….…………………………………..…………………………………... 6

Database Design ……….…………………………………………………………………………….. 7

Hardware and Software Detailed Design ……...…………………………………………………... 7

System Security and Integrity Controls ………...………………………………………………….. 8

# 

# **Introduction**

The purpose of this document is to detail how the proposed Car-Nerd desktop application will be constructed. The intention of this application is to provide users with an effective way to maintain information for their personal vehicles, such as oil change reminders, mileage tracking, maintenance log, and the ability to access important vehicle documents. This document will translate what specifications will be required and used by developers to create this application. The system architecture will be identified along with identifying hardware and software requirements, as well as user interface components.

# **Purpose**

This System Design Document chronicles the requirements needed in order to define the architecture and system design of this application and give the development team guidance on how to create the system. This document also gives a detailed description of the system architecture, database design, and system security.

The intent of this desktop application is to provide a stress free solution for vehicle owners to maintain vehicle information. The application will track activities such as mileage, oil changes, maintenance log, and service reminders. Additionally, users will be able to select a button that will open to important vehicle documents such as insurance information, car registration, and digital copy of a driver’s license.

# **System Overview**

Vehicle owners have had the time consuming tasks of having to keep track of vehicle documents and service records. For owners with more than one vehicle, these tasks take even more time and can require quite a bit of organization. The proposed solution will improve the organization and management of personal vehicles.

Car-Nerd must be designed to be supportive and effective for users, and provide a foundation for future growth. The application must be user friendly so that users which are not as tech savvy will not have any issues when using the application. The user interface must be clear and concise which will add to the overall user experience. The application should be compatible with most operating systems and will allow for users to name their vehicles to assist with vehicle tracking and to provide a more personal experience.

Project Goals include:

1. Ability to add a vehicle to the application and give vehicle a personalized name
2. Ability to add year, make, and model to vehicle profile
3. Ability to schedule service reminders
4. Track oil changes
5. Track mileage
6. Track details for multiple vehicles in one central location
7. Ability to charge a user a monthly or yearly subscription fee for the application service
8. Payment and billing, which is to be handled through a third party (such as PayPal)

**Design Constraints**

The team has identified a few limitations that will have an impact on the design process. To date, the limitations are as follows:

1. Finding the time to create Car-Nerd is a challenge due to the team members having multiple priorities to deal with and being in separate locations working remotely. Schedule conflicts also play a role in this limitation.
2. With Car-Nerd only being deployed as a desktop application, users will not have access to it when they are away from their computer which impacts the overall design appeal of the application. Additionally, access to the internet is required, so there is no offline functionality.
3. Some team members are not versed in some of the applications being used to create and maintain Car-Nerd, such as GitHub and Java. This impacts the design process as the team has to learn these applications while at the same time, contributing to the design process.

# 

# **Roles and Responsibilities**

The team as a whole is responsible for the overall success of Car-Nerd. The illustration below shows the roles and responsibilities for Dynamic Developers. Contact information is provided for each team member should issues or concerns arise regarding Car-Nerd.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Phone** | **Email** |
| Alyssa Pettit | Developer - UI and Architecture, Info/Settings Tab | (314) 635 8688 | ap248489@my.stchas.edu |
| David Tippy | Developer - UI and Architecture, Reminders Tab | N/A | dt238919@my.stchas.edu |
| Kelli Ennis | Project Manager | (786) 280 8660 | ke242067@my.stchas.edu |
| Nicholas O'Connell | Developer - UI and Architecture, Main Page Tab | (636) 385 8568 | no239791@my.stchas.edu |
| Stephen Wood | Developer - UI and Architecture, Log Tab | N/A | sw229914@my.stchas.edu |
| William Waugh | Developer - UI and Architecture, Mileage Module | (951) 475 4505 | ww239648@my.stchas.edu |

# 

# 

# 

# **Project References**

Car-Nerd was developed using several references. First, the initial concept of Car-Nerd was based on these other apps:

* DrivrLog
* Car Minder

These apps, in general, have similar functionality to Car-Nerd, but are each lacking in full functionality that we at Dynamic Developer Services plan to provide our users of Car-Nerd.

Internally, there were several meeting minutes and reference guides used to keep our trajectory on point.

* Notes 6\_8.docx: discusses beginnings of feature highlights.
* Notes 6\_10.docx: Hashes out more feature details, and how we may execute these features.
* Notes 6\_15.docx: Discussed beginning of UI designs.
* Car-Nerd UI.jpg: depicts a rough illustration of what the UI should look like - June 15, 2020.
* Notes 6\_17.docx: Created the start of a privacy statement, redefined the phases, and start the division of work.
* Notes 6\_20.docx: GitHub discussions.
* Notes 6\_22.docx - Further GitHub discussions.
* Notes 6\_23.docx - documentation creation.

# 

# 

# 

# 

# **System Architecture**

**Hardware:**

Dynamic Developers is a small team based company where due to the global climate, team members are working remotely. Therefore, hardware components and setup range between the group, however, all developer setups are able to successfully and effectively work on system design, including UI work and the coding of the application.

* Developer Computers consisting of:  
  - 8GHz Server Suite  
  - RAM: 8 GB  
  - 6x 80GB 15,000 RPM Hard Drive  
  - 6x Power Cord + Charger  
  - 1 TB SAN Storage Device  
  - 6x Dell P3000 Workstations  
  - 6x Viable Wifi Routers

**Software:**Dynamic Developers’ application works with various components in conjunction to store and display user inputted data. The software creates a database in which input can be entered and modified by users to make information easily accessible by the user. The components the team currently has is as follows:

* Car Tab: Ability to add a vehicle to the app and/or select a previously added vehicle.
* Info Tab: Users can specify a name, the make, model, and year for the vehicle they have added.
* Log and Reminder Tab: Stores reminders, and gives ability to log maintenance service data.
  + Time tracking
  + Notes
  + Displays events added to log
* Mileage Tab: Holds and calculates the mileage of the users vehicle.
  + Mileage at last oil change
  + Current mileage
  + Date of last oil change
* Settings Tab: Provides users with logout function.
* Google API: Gives user google login access to store and hold data.

# **Database Design**

The database that is featured within Car-Nerd assists with the integrity of the user entered data. This data assists the user when inputting their vehicle information that they have the proper information for their car. The database is a static list of all year, make and models for the last 25 years that are currently available in the US.

The user will first input a year, then based on the year entered, any car manufacturers that have vehicles available within that year will populate. Any manufacturers that do not have any vehicles manufactured within that year will not be selectable. Once the manufacturer has been selected, the available models will populate, excluding any options that are not available for that year. Any cars above and beyond 25 years must be manually added to the application due to the inability to track imports.

# **Hardware and Software Detailed Design**

**Hardware:**

The Car-Nerd application does not require any specialized hardware, and the developers do not plan to use any specialized hardware to complete the project at this time. The application is being designed to run on standard Windows machines.

**Software:**

The Car-Nerd product will be written in Java, using Swing libraries to create GUI functionality. The core functionality of the program will be to keep track of the documents associated with an individual’s automobiles, as we at Dynamic Developers feel that there isn’t currently an off-the-shelf application that serves this purpose to a satisfactory degree. The implementation of this system will be done exclusively in Java, until such time as the Dynamic Developers team deems other tools necessary for timely completion of the project.

# **System Security and Integrity Controls**

Internal Security for our application will be more user based since the data they enter will be stored locally and only on their machine in the form of a text file.

The user will have the ability to retrieve their data from the database at their choosing should they need to, since data will be stored on their local machine.

Login data from logging into the application will be handled through the use of the Google API and no data will be stored by Dynamic Developers in any instance. Additionally, the ability to check user data files will be convenient for the user as they will just have to check the install folder for the text file.