REPORT ONE Parts 1-3

GROUP 3

Brett J. Davis

Thomas Kerley

Jasmine Irvin

Joshua Puetz

CONTRIBUTIONS

Brett J. Davis 25%

Thomas Kerley 25%

Jasmine Irvin 25%

Joshua Puetz: 25%

Table of Contents

[1 PART 1 1](#_Toc138582408)

[1.1 Problem Statement 1](#_Toc138582409)

[1.2 Decomposition 4](#_Toc138582410)

[1.3 Glossary of terms 5](#_Toc138582411)

[1.4 Goals, Requirements, and Analysis 6](#_Toc138582412)

[1.4.1 Business Goals 6](#_Toc138582413)

[1.4.2 Enumerated Functional Requirements 8](#_Toc138582414)

[1.4.3 Enumerated Nonfunctional Requirements 8](#_Toc138582415)

[1.4.4 User Interface Requirements 10](#_Toc138582416)

[2 PART 2 11](#_Toc138582417)

[2.1 Use Cases 11](#_Toc138582418)

[2.1.1 Stakeholders 11](#_Toc138582419)

[2.1.2 Actors and Goals 11](#_Toc138582420)

[2.1.3 Use Cases 12](#_Toc138582421)

[2.1.4 System Sequence Diagrams 17](#_Toc138582422)

[2.2 User Interface Specification 18](#_Toc138582423)

[2.2.1 Preliminary Design 18](#_Toc138582424)

[2.2.2 User Effort Estimation 21](#_Toc138582425)

[3 PART 3 - System Architecture 23](#_Toc138582426)

[3.1 Identifying Subsystems (UML Package Diagram) 23](#_Toc138582427)

[3.2 Architecture Styles 25](#_Toc138582428)

[3.3 Mapping Subsystems to Hardware 26](#_Toc138582429)

[3.4 Connectors and Network Protocols 27](#_Toc138582430)

[3.5 Global Control Flow 27](#_Toc138582431)

[3.6 Hardware Requirements 27](#_Toc138582432)

[4 PART 3 – PROJECT SIZE ESTIMATION 29](#_Toc138582433)

[5 PART 3 - PROJECT MANAGEMENT (PLAN OF WORK) 30](#_Toc138582434)

[6 PART 3 - Reference list 32](#_Toc138582435)

# PART 1

## Problem Statement

A large number of cars in a parking lot

Description automatically generated with low confidenceAs a car dealership, our vehicles frequently move around due to test drives, vehicle previews, maintenance operations, cleanings, sales, and an assortment of other reasons. Given that the pandemic has increased demand for both new and used vehicles, turnaround times have materially decreased, making it significantly more challenging to track where a car is located in our various parking and sales lots. We need a system to track how long a vehicle is in our lot, which parking spaces have a higher turnover rate, and a means by which to optimize vehicle movements.



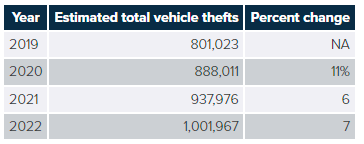
Currently, the only means by which we have to track automobile inventory is to physically place a tag with the vehicle’s VIN (Vehicle Identification Number) on each set of corresponding keys. While this allows us to know which vehicles are in stock, it provides us with no information regarding the vehicle’s location. Consequently, our technicians and salespeople are burdened with developing their own means to track cars. These ad hoc and often esoteric tracking solutions are difficult to teach to new crew members, which in turn leads to many errors and delays. Furthermore, the current system is fairly cumbersome as rapid changes cannot be made to it when we try to perform optimizations. Due to this manual, analog process, we do not have any data on how our vehicles attract customers or if there are theft issues due to positioning.

A picture containing screenshot

Description automatically generatedOverall, we need a system that allows our technicians to find our vehicles quickly and efficiently. By decreasing the time spent searching for a certain vehicle, employees can more effectively spend their time doing repairs, cleaning, or show casing various vehicles to customers. Additionally, we believe that faster turnaround times in the garage would increase customer satisfaction, and thereby net our company greater business in the form of repeat customers seeking vehicle maintenance. In addition to finding where a given car is, knowing which spaces are available to park new inventory would expedite the parking and stocking process. Furthermore, our ideal system would prioritize which parking spaces to fill based on certain vehicle characteristics, such as placing those in need of maintenance closer to the garage in order to reduce travel times. In conjunction, the system should also be able to detect if there is a car in a parking space by a sensor to confirm that a vehicle was parked in the parking space. Ideal this confirmation should include a timer or a camera to ensure the accuracy that the correct car was placed in the correct parking space.

We would prefer that when a delivery is scheduled, the prospective vehicles would be added to a list that would let us know if we have sufficient spaces to store all of the new inventory across our various lots. The system should further add the new vehicles to our inventory list and assign optimal parking spaces to each new car. Additionally, the ideal system would have sorting capabilities that would tell us deliveries by date, in addition to vehicle locations. Combined with the existing inventory of vehicles, these can be shown to a customer on a web page, enabling the online customer to perform some self research on our inventory. The newly arriving vehicles should be earmarked as “Coming Soon” in some fashion. This web page might include a means to contact us to schedule a test drive or allow a reservation on a vehicle they desire to purchase after visiting our dealership.

According to the National Insurance Crime Bureau (2023), car thefts increased by 7% in 2022 reaching a total 1,001,967 stolen vehicles within the United States of America. Netwatch North America (Netwatch, 2023) claims that approximately forty percent of all automotive thefts occur at dealerships, making this a prime concern for our business. We note, however, that if an automobile is reported stolen within the first twenty four hours, there is a thirty four percent chance of recovery (Brenckle and Stroisch, 2022). Given this, it is of utmost importance that we have a clear catalog of all of our inventory so that we may identify any theft as quickly as possible and thereby increase our chance of recovering the car and minimize any potential losses.



Though less of an economic burden than the loss of an entire vehicle, we have noted that thieves are increasingly targeting cars’ catalytic converters. CarFax’s Patrick Olsen (2022) reports that 153,000 catalytic converters were stolen in the United States in 2022 alone. These parts are expensive to replace and repair due to the precious metals within the pollution controlling device. Russo (2023) states “average estimates for replacing the catalytic converters in our sample vehicles ranged from $933 to $4,414” nationwide. The asset protection teams need to know when a vehicle was parked and in which spaces in order to find out where, if any, auto thefts or vehicle damages occur. This would allow us to find solutions to any high crime areas in our lot. The system should collect a log of when a reported incident occurs and we can view a list of incidents by space.

A person lying on the ground next to a car

Description automatically generated with medium confidence

We would further like for the system to track odometer changes from the time a vehicle arrives in our lot until it is sold so that we may better understand our vehicles’ movements and account for them. We believe that we could use this data to estimate and project our fuel needs. Additionally, if our salespeople could use the system to easily access the computer records and acquire the odometer readings for the vehicle sale documents without physically walking to and from the car, we could expedite closings. Furthermore, knowing how often a car gets test driven before it gets sold could give us insight into how much value we might lose on a vehicle before it is sold.

This system will need a login system to prevent unauthorized access. There should be an administrator access level who can access, maintain, or troubleshoot the system. A salesperson might step away from a computer spontaneously to assist a customer, and we want our data to be secure during those times. As such, the computer will need to automatically lock, as well as a prompt activated to log in. Technicians will also need a login to mA person and person giving each other a high five

Description automatically generatedanage vehicles for test driving, storage, and other relocation-based activities. In addition, there needs to be a means of packaging the vehicle information to easily integrate with our existing sales software. The vehicle information should include the odometer readings, the date the vehicles arrived on our lot, when the vehicle was sold, and if it was related to any theft or damage while on our lot.

This system shall be novel in its execution in the following ways: it will provide a parking automation system to a dealership instead of a garage; the system will track inventory of the dealership; assist to deter theft and quickly recover from the incidents that do occur; provide an online portal to customers of inventory; and provide sales data to improve profitability of the dealership over time. Many dealerships do not have an automation system in place to perform these tasks.

## Decomposition

What the dealership needs from the tracking system is the following:

1. An interface/web page that takes in vehicle data
   1. time of acquirement
   2. odometer
   3. year
   4. make
   5. model
   6. size
2. Which spaces are available for a given size of car
   1. which of these are most efficient
   2. deliverables future space conflict detection and allocation
3. This tells the technicians where to park the car via a web page
   1. a sensor detects that a car was parked in it’s corresponding space
   2. the sensor tracks when the space was occupied and shares it with the database.
4. An interface/web page to input any theft or damage reports of a given vehicle
   1. this includes noting if/when the police report was made
5. Login pages at each terminal to increase security across the dealership.
   1. This authentication has at least three levels of access: Admin, Salesperson, and Technician
6. A web page that shows inventory to a potential customer
   1. This might be a fourth level of authentication
   2. This should also include deliverables
7. A data package to allow software designed by others to easily integrate the new data.

## Glossary of terms

* Administrator – A user who has the highest access level.
* Dealership – The primary customer that the software is being provided to.
* Deliverables – new vehicles in route to the dealership.
* Display parking space – a unique parking space(s) that show cases a certain vehicle.
* Lot – a collection of parking spaces.
* Occupied space – A parking space that contains a vehicle.
* Odometer reading – The number of miles a vehicle has been driven
* Parking space – A location where a car can be parked.
* Parking space number – a unique number assigned to a parking space.
* Parking space sensor – a device in each parking spot that can determine the presence of a car.
* Salesperson(s) – A user who often accesses the system. Salespersons may not need sales metrics. They will need parking information and vehicles history/information.
* Technician(s) – Users who often access the system.
* Vehicle – the object of importance that the data collection is focused on. It can be assigned a parking space.

## Goals, Requirements, and Analysis

### Business Goals

The business goals of implementing a comprehensive tracking system for the car dealership are aimed at improving operational efficiency, customer satisfaction, and overall profitability. By addressing the challenges posed by the increased demand for vehicles and the need to track their location within various parking and sales lots, the dealership aims to streamline its operations and optimize vehicle movements. The primary objective is to provide a system that enables quick and efficient vehicle tracking, allowing technicians to locate vehicles promptly, reducing search time, and enabling them to focus on repairs, maintenance, and showcasing vehicles to customers.

The system is expected to enhance customer satisfaction by facilitating faster turn-around times in the garage, thereby fostering repeat business and increasing customer loyalty. Additionally, the system should optimize parking space allocation, prioritize vehicles based on specific characteristics, and provide real-time information on available parking spaces to expedite the parking and stocking process.

By having a clear catalog of inventory, the dealership aims to mitigate the risk of theft, improve security, and increase the chances of recovering stolen vehicles promptly. Furthermore, the system should track odometer changes, enabling better understanding of vehicle movements and fuel needs estimation. The intent is to improve the overall profitability of the dealership by leveraging data-driven insights and providing a secure, efficient, and user-friendly system for dealership staff. Additionally, the dealership aspires to enhance the online customer experience by offering a web page that showcases inventory, allows self-research, and facilitates test drive scheduling or vehicle reservations.

By achieving these business goals, the dealership aims to position itself as a technologically advanced and customer-centric automotive provider, driving growth and success in a competitive market.

A picture containing text, diagram, plan, parallel

Description automatically generated

1. Improve inventory tracking: Develop a system to track the location and duration of vehicles in the dealership's parking and sales lots, allowing technicians and salespeople to quickly locate vehicles and reduce errors and delays in the tracking process.
2. Optimize vehicle movements: Implement a means to optimize vehicle movements within the lots, including prioritizing parking spaces based on vehicle characteristics and reducing travel times for vehicles in need of maintenance.
3. Enhance customer satisfaction: Decrease the time spent searching for vehicles, leading to faster turnaround times in the garage and increased customer satisfaction. Provide a web page to showcase inventory and enable online customers to perform self-research, schedule test drives, and make reservations.
4. Improve security and minimize losses: Deter theft and quickly recover stolen vehicles by having a clear catalog of inventory, tracking incidents of theft or damage, and identifying high crime areas. Increase the chances of recovering stolen vehicles by promptly reporting thefts and monitoring the parking spaces.
5. Streamline processes and increase profitability: Track odometer changes, estimate fuel needs, and gather sales data to better understand vehicle movements and improve profitability over time. Expedite sales by providing computer records and odometer readings to salespeople without physically accessing vehicles.

### Enumerated Functional Requirements

|  |  |  |
| --- | --- | --- |
| Requirements | Priority | Description |
| REQ-1 | Priority: High | Develop an interface/web page to input vehicle data, including time of acquirement, odometer reading, year, make, model, and size. |
| REQ-2 | Priority: High | Provide information on available parking spaces for a given size of car and identify the most efficient spaces for parking. |
| REQ-3 | Priority: High | Enable technicians to access a web page that suggests where to park vehicles and use parking space sensors to detect and record occupancy. |
| REQ-4 | Priority: Medium | Design an interface/web page for reporting theft or damage of vehicles, including information about police reports. |
| REQ-5 | Priority: High | Implement login pages with three levels of access (Admin, Salesperson, Technician) to ensure security and prevent unauthorized access. |
| REQ-6 | Priority: High | Create a web page to showcase inventory to potential customers, including information about deliverables. |
| REQ-7 | Priority: High | Provide a data package for easy integration with existing sales software, including odometer readings, acquisition date, sale date, and incidents on the lot. |

### Enumerated Nonfunctional Requirements

1. REQ-8: Functionality - Priority Weight: High:
   * Capability: The system should have a comprehensive set of features to track and manage inventory effectively, including vehicle data input, parking space optimization, theft or damage reporting, and inventory showcase.
   * Reusability: The system should be compatible with existing sales software, ensuring seamless integration and data synchronization.
   * Security: The system should incorporate robust security measures to protect data and prevent unauthorized access or exploitation.
2. REQ-9: Usability (UX) - Priority Weight: High:
   * Human Factors: The system should have an intuitive and user-friendly interface, considering the needs and capabilities of different user roles (Admin, Salesperson, Technician).
   * Aesthetics: The user interface should be visually appealing and enhance the overall user experience.
   * Consistency: The system should maintain consistency in design, navigation, and terminology throughout the user interface.
   * Documentation: Clear and comprehensive documentation should be provided to guide users in operating the system.
   * Responsiveness: The system should respond promptly to user actions, providing a smooth and interactive user experience.
3. REQ-10: Reliability - Priority Weight: High:
   * Availability: The system should have a high level of availability, minimizing downtime and ensuring continuous access to the inventory tracking and management features.
   * Predictability: The system should be stable and predictable, with consistent behavior across different scenarios and user interactions.
   * Accuracy: The system should minimize errors in tracking and reporting vehicle information, ensuring the reliability and integrity of data.
4. REQ-11: Performance - Priority Weight: Medium:
   * Speed: The system should respond quickly to user inputs, minimizing latency and providing a responsive user interface.
   * Efficiency: The system should utilize system resources efficiently, optimizing performance and minimizing resource consumption (e.g., CPU, memory, storage).
   * Throughput: The system should handle concurrent requests and transactions effectively, maintaining high throughput and accommodating multiple users simultaneously.
   * Capacity: The system should be able to handle a growing inventory volume and increasing user load without significant degradation in performance.
   * Scalability: The system should be scalable to support future growth, allowing for easy expansion and accommodating additional features or modules.
5. REQ-12: Supportability - Priority Weight: Medium:
   * Testability: The system should be designed to facilitate testing and debugging processes, ensuring easy identification and resolution of issues.
   * Flexibility: The system should be modular and easily modifiable, allowing for future enhancements, modifications, and customization.
   * Installability: The system should have clear installation instructions and be easily deployable on different hardware and software configurations.
   * Localizability: The system should be adaptable to different languages, cultures, and regional requirements, enabling localization for global use.

### User Interface Requirements

### 

The user interface requirements for the system primarily focus on the backend functionality, as it serves as a comprehensive tracking and management system for the car dealership. While the initial implementation of the system may not have a front-end-facing section, there is a future goal to provide a front-end interface to enhance customer interactions and enable online inventory browsing.

The user interface design, although currently back-end-oriented, should consider the potential front-end expansion, ensuring scalability and adaptability to accommodate the addition of a customer-facing web page or portal. The backend user interface should prioritize functionality, ease of use for dealership staff, and efficient data management, laying the foundation for a seamless integration with a future frontend interface.

1. Vehicle Data Input Interface:
   * The interface should have a form to input vehicle data fields (time of acquirement, odometer reading, year, make, model, and size).
   * It should provide clear labels and instructions for each field.
   * The interface should include validation checks to ensure accurate data entry.
2. Available Parking Spaces Interface:
   * The interface should display a visual representation of parking spaces and indicate their availability.
   * Different sizes of cars should be distinguishable on the interface.
   * The interface should prioritize and highlight the most efficient parking spaces for a given car size.
3. Technician Parking Guidance Interface:
   * The interface should provide clear instructions and suggestions to technicians on where to park vehicles.
   * It should include a real-time occupancy display to confirm if a parking space is occupied.
   * The interface should provide a notification or alert when a parking space becomes available.
4. Theft or Damage Reporting Interface:
   * The interface should have a form to report theft or damage incidents, including relevant details and information.
   * It should include fields for recording police report information if applicable.
   * The interface should provide clear instructions and options for reporting incidents.
5. Inventory Showcase Web Page Interface:
   * The web page should have a visually appealing layout showcasing vehicles available for customers.
   * It should provide search and filtering options based on vehicle characteristics.
   * The interface should include interactive elements to schedule test drives or make reservations.
6. Data Package Integration:
   * The system should provide an easily integratable data package compatible with existing sales software.
   * The package should include a well-documented API or data format for seamless integration.
   * Integration should allow for smooth synchronization of vehicle data and sales information.

# PART 2

## Use Cases

### Stakeholders

* + 1. Dealerships
    2. Salespersons
    3. Customers
    4. Inventory Managers
    5. System Architects
    6. IT Managers
    7. Project Managers
    8. Testing and Quality Assurance Engineers
    9. Developers
    10. End Users
    11. Business Analysts
    12. Car Manufacturers
    13. Shipping Companies
    14. Lot Managers

### Actors and Goals

|  |  |
| --- | --- |
| **Actor** | **Actor’s Goal (what the actor intends to accomplish)** |
| Inventory Manager | To enter vehicle information |
| Inventory Manager | To update vehicle status |
| Inventory Manager | To manage inventory |
| Inventory Manager | To manage parking lots |
| Salespersons | To query inventory |
| Salespersons | To locate vehicle location |
| Sales manager | To perform vehicle sales |
| Customers | To query inventory |
| Customers | To locate vehicle location |
| Sensor Device | To determine vehicle location |

### Use Cases

#### Casual Description

Use Case No. 1: An inventory manager is responsible for a new shipment of vehicles arriving at the dealership. The inventory manager queries the system to determine the number of available parking spaces. The inventory manager uses the system to determine the optimal parking locations for each of the vehicles. The inventory manager assures that all parking locations remain full to optimize dealership inventory.

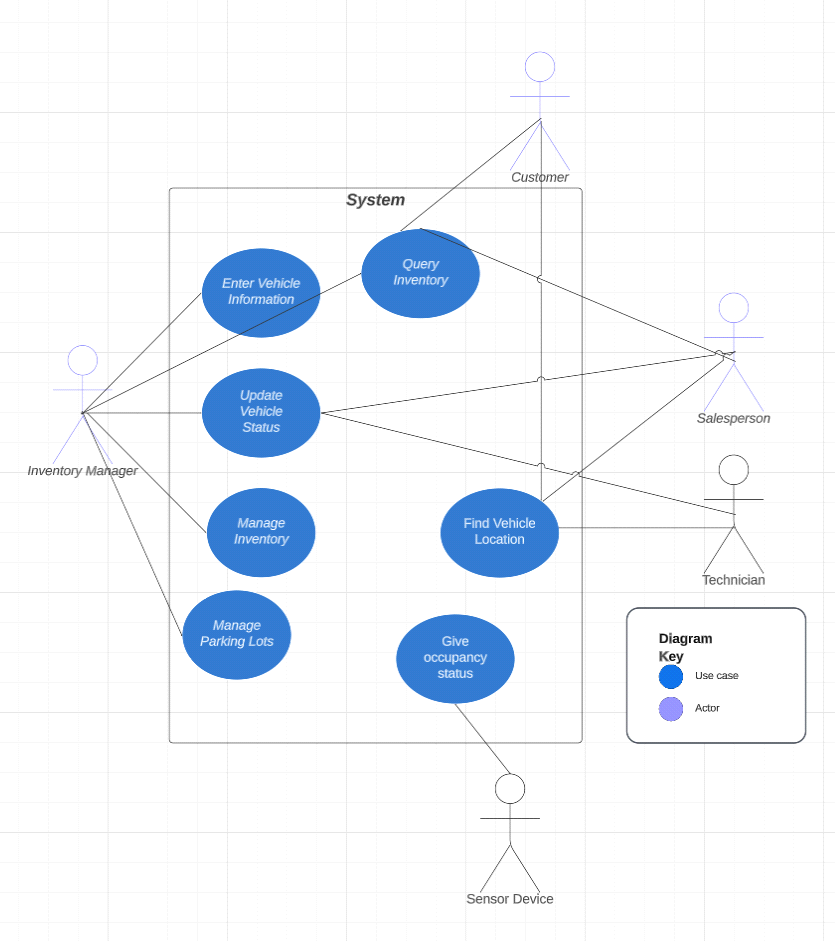
Use Case No. 2: A customer arrives at a dealership to test drive a vehicle advertised online. The salesperson uses the system to locate the physical location of the vehicle in the parking lot. The salesperson dispatches a technician to quickly retrieve the vehicle for a test drive.

Use Case No. 3: A technician is asked to repair a vehicle at the dealership. The technician uses the system to locate the vehicle in the parking lot.

Use Case No. 4: A customer locates a vehicle on a website. The website provides the customer with the location of the vehicle in a parking lot. The customer goes directly to the vehicle’s location on the lot. The car automatically unlocks and allows the customer to perform a test drive.

Use Case No. 5: An inventory manager is alerted that a vehicle is missing from inventory. The inventory manage contacts local law enforcement to report the vehicle stolen.

#### Use Case Diagram



#### Traceability Matrix

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Req’t | PW | UC1 | UC2 | UC3 | UC4 | UC5 |
| REQ1 | 5 | X |  |  |  | X |
| REQ2 | 5 | X | X |  |  |  |
| REQ3 | 5 |  |  | X |  |  |
| REQ4 | 3 | X |  |  |  | X |
| REQ5 | 5 | X |  | X |  | X |
| REQ6 | 5 |  | X |  | X |  |
| REQ7 | 5 |  |  | X |  | X |
| Total PW | | 18 | 10 | 15 | 5 | 20 |
| Max PW | | 5 | 5 | 5 | 5 | 5 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case # | Test Case | Test Steps | Test Data | Expected Result |
| 1 | Verify Login | * Go to Login Page * Enter UserId * Enter Password * Click Login | Id = {valid name} password = {valid password} | Successful Login |
| 2 | Move Car | * Select “move car” option. * Enter car information * Look at cars informatin * Select “move” | Car ID = {license plate no} | Respond that move command was successful and updates status of the car in our DB |
| 3 | Finalize Car Movement | * Select Search car * Select “parked” * Input updated location | Car ID = {license plate no}  NewLocation = {new location ex. A3| | Respond that park command was successful and updates status of the car in our DB |
| 4 | Query Database | * Select “Search Car” * Input car’s ID | Car ID = {license plate no} | Returns a web page for the specific car. If no car is found, returns error |
| 5 | Query Database | * Select Search by parking space * 2) Input parking space identifier | ParkingSpace = {parking space number} | Returns a web page with parking space’s status and current car occupying space.  If no space is found, returns error |
| 6 | Edit Inventory | * Click on car or parking space from search result * Click edit * Make changes to object | Car ID = {valid license plate no}  ParkingSpace = {valid parking space number} | Changes made by user are reflected in the database. If changes aren’t valid, an error is thrown and describes what the user did wrong |
| 7 | Customer Database Query | * User goes to dealership website * User clicks on link that takes them to our customer inventory view | Get Request | Shows specially made customer for the dealership that the customer was looking for. |

#### Fully-Dressed Description

1. Technician

This is the most common use case for our system and is the one undertaken by the technician. Our software will mainly have a linear flow to it in terms of updating the system but there will be some event-driven interactions. Ideally, there is always a pre-check before moving a car. A user will input a qualified identifying characteristic of the car, such as VIN number or license plate number and cross reference our system with the car's current location to verify the car is where we think it is. If not, the user can report the inconsistency using our system and it will be logged. A pre-check also allows us to implement logic to detect anomalous behavior. The pre-check allows us to send a notification to our system that a car is being moved. With this notification, we can send alerts when a sensor value changes without the notification being sent to our system so the matter can be investigated.

After the pre-check is conducted, the user can then move the car to the desired location or location chosen by a recommendation algorithm. After a user parks the car, they update the car with the new location that also notifies the server that the car is parked and to change the isBeingMoved property of the car in our system back to false. This way, we can ensure accurate information for the car’s location and anytime a sensor state changes we can check our system for the car parked at that location and see if it is expected behavior.

2. Salesperson

a) The salesperson would like to see which part of the parking lot cars are selling most frequently from. Salesperson sorts spaces by the highest sales value.

b) The Salesperson would like to know which cars are test driven the most often and examine trends.

c) Salesperson would like to know the location of a car so he can quickly find it for a customer or confirm it is still available.

3. Admin/Manager

a) Hired a new employee and adds them to our system so they can login and do their job duties.

b) found a discrepancy in the system, such as, a car being logged in the wrong location and changes it to the correct value.

c) receives notifications about anomalous behavior and acts upon it.

d) Admin needs to change the role of someone in the system (technician to salesperson, for example)  
 e) Admin needs to remove someone's access from the system.

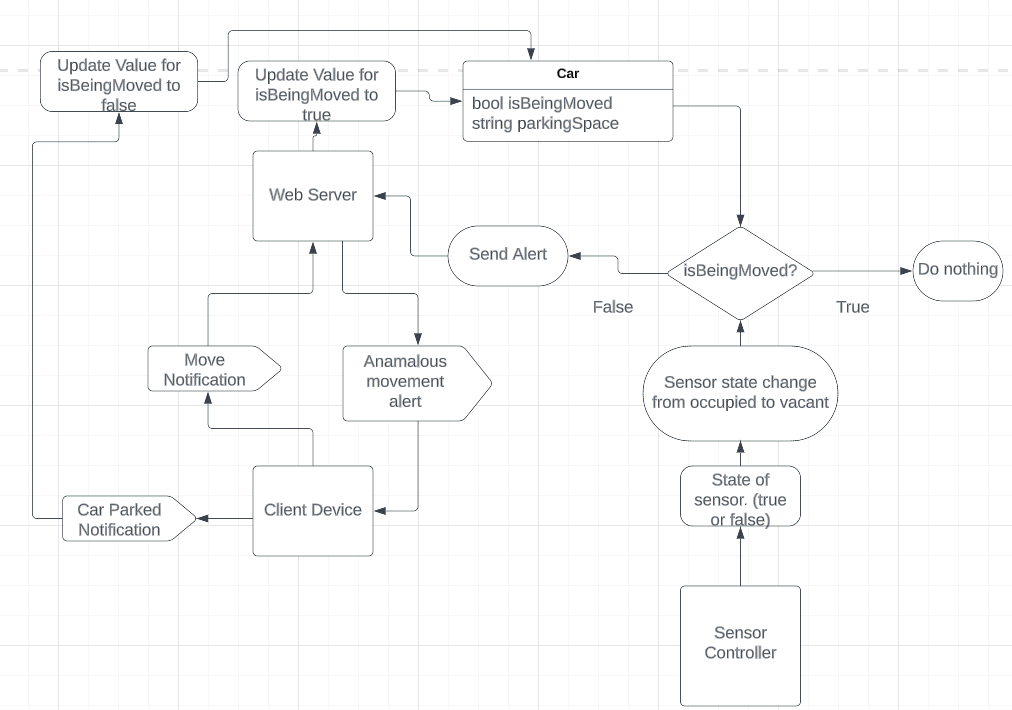
### System Sequence Diagrams

Adding a Vehicle

A picture containing text, diagram, line, parallel

Description automatically generated

Moving a Car



## User Interface Specification

### Preliminary Design

Below is a preliminary webpage designs to add a vehicle to a database:

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

Below is a design to add a vehicle to a particular lot.

A screenshot of a car

Description automatically generated with medium confidence

Below is a webpage to locate a vehicle that may be displayed on a webpage:

A screenshot of a car

Description automatically generated with medium confidence

In response, the system may display an image of a parking lot with the location of the vehicle highlighted:

A picture containing screenshot, line, circuit

Description automatically generated

Alert if Vehicle Missing from Lot:

A screenshot of a car

Description automatically generated with medium confidence

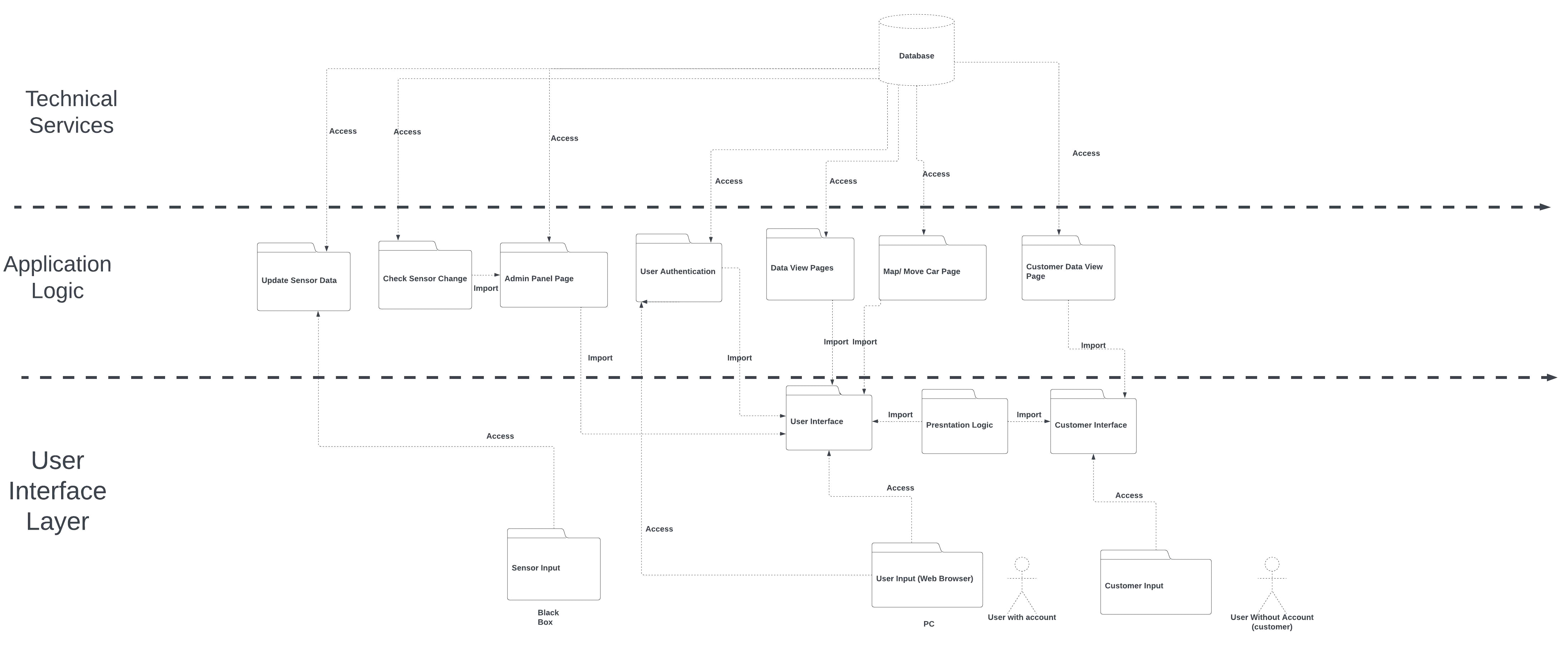
### User Effort Estimation

* + - 1. Navigation (3 total clicks)
  + Click to page
  + Click add vehicle
  + Click Ok

1. Data Entry (21 clicks)
   * Add Vehicle Details (20 boxes, 20 clicks)
   * Select Lot (1 click)
2. Locate Vehicle (1 click)

# PART 3 - System Architecture

## Identifying Subsystems (UML Package Diagram)



**User Interface Layer**

*User interface* – Web page that imports the logic for displaying views to the user based on their role.

*Presentation Logic* – Dictates how the data is displayed to the end user.

*Customer interface* – Displays view for unauthenticated user to search for a dealership and view availability.

**Application Logic**

*Update Sensor Data –* Queries the sensor server on intervals and updates the database with new values.

*Check Sensor Change –* Queries the database for current sensor data on a time interval staggered with the time interval for updating sensor data. It compares the sensor values to the values on the last check and identifies changes from “on” (occupied) to “off”(vacant). When a change from on to off is detected, it checks whether the is Being Moved property for the car in that space is set to true. If true, do nothing, if false, that means the car has moved without the move being registered and sends a notification to admins to investigate further.

*Admin Panel Page –* Page that provides an interface for administrative users to view events, CRUD users/cars/spaces, edit dealership information.

*User Authentication –* Receives log in credentials from the user, checks it against the DB, and grants or denies access based on the results. Passes the users role to the view controller so it can render the appropriate view for the user’s role in the system.

*Data View Pages –* This will be an interface for admins and salespersons to query the database to view information about the cars at the dealership and to input sales data about the cars.

*Map/ Move Car Page –* This page will display a map of the dealership and show empty spaces quantity/location. It will also allow the user to search for cars, input current odometer readings, and initiate/finalize car movements.

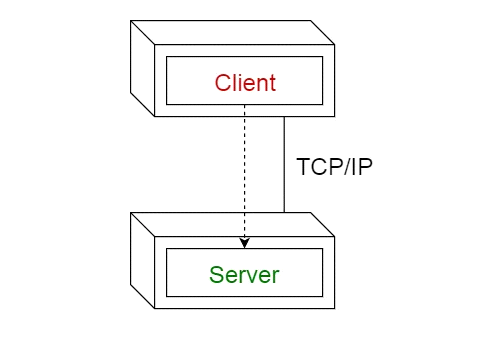
*Customer Data View Page –* This page will be a custom view for unauthenticated user to search for a particular dealership(site), view a showcase of its inventory, and check availability of certain cars.

**Technical Services**

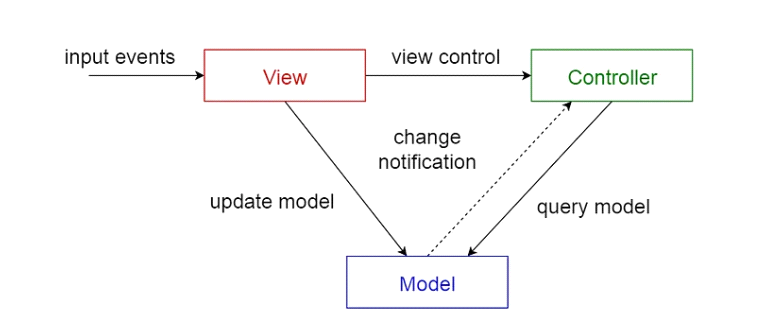
*Database –* This will be a relational database.

## Architecture Styles

Our system will be based on a **client-server** architecture utilizing HTTP communication protocols. Where the clients are the users on a PC and the sensor controller and the server is our web server on the internet.



While treating the web server like a black box, the system as a whole operates on a client-server architecture. If we peek inside the web server, we will see that this subsystem operates with a **MVC (Model – View – Controller)** architecture.



This architecture is ideal for the web server as we will have multiple users who will have a different view of the data based on their role. It separates internal representations of information from the ways information is presented to, and accepted from, the user. It decouples components and allows efficient code reuse.

**Models**

Car – Will represent a car with all relevant attributes.

Space – Will represent a space with all relevant attributesMap – Will represent the map of our dealership. Contains cars and spaces.

User – Authenticated users with defined roles and credentials.

View Model – This will dictate what a user sees based on their role and what page they are viewing.

## Mapping Subsystems to Hardware

**Sensors**

Being that a car dealership is a large outdoor area. The optimal way in terms of cost and maintainability for an interconnected sensor system would be to have them communicate using HTTP protocols over an 802.11 Wi-Fi network. While wireless communication over such a large area poses its own challenges, the challenges that need to be overcome pale in comparison to wired installation. Since there are hundreds of spaces in a car dealership. There would need to be a vast network of underground cabling enclosed in conduit strong enough to resist the weight of a vehicle rolling over it routinely. This would need to be installed by trained electricians, construction workers, and engineers due to the system's complexity. This would turn into a very costly endeavor and would dramatically increase the cost of maintaining the system after installation. With Wi-Fi connected devices there are still technical limitations. Interference can be an issue so building a system that can detect false positives and other faulty behavior by the data transmitted to the system will need to be implemented. Many car dealerships may not have Wi-Fi coverage in all parts of their facility that they would like tracked. This can be solved by adding Wi-Fi extenders to increase overall coverage of their system. While these limitations can be a pain, they are minor in comparison to the problems that arise from a wired installation.

In both cases, there will need to be a centralized controller on the site that can receive data from the sensors and transmit it via the internet to our webservers. The controller itself will need to be a software installed on a device with two main properties.

A) Enough computing power to process the potentially large number of sensors in the lot

B) Connectivity to the network that the sensors broadcast on and connectivity to the internet.

This could be accomplished with software on a desktop PC or for bigger installations, a server onsite running the software on a VM or a physical device. This will be a third- party solution and will be treated like a black box with “on” and “off” inputs for each sensor.

**Web Server**

This will be a remote server on the internet provided by a web hosting provider such as Heroku.

**Database**

This will be a relational database running on a remote server on the internet.

## Connectors and Network Protocols

**Connectors**

Because our system functions on the internet, all connections in our system will be using ethernet and RJ-45 or a Wi-Fi antenna in the case of sensor communication.

**Protocols**

Since our software is going to reside on a web server, we will use the HTTP protocol to transmit data in and out of our software. In particular, we will be using the TCP/IP protocol. This protocol is ideal because it is using a three-way handshake between the sender and receiver to verify that the HTTP packets arrive in order and no data is missing. We will use this as opposed to UDP because it provides much more reliability with the data. UDP is best suited for video or voice applications because of its speed. We do not transmit voice and video in our system so we will use TCP instead as data reliability is extremely important in every level of our system.

## Global Control Flow

Execution orderness: In general, the system is a procedure-driven process that executes in a linear fashion. The system responds to queries from users regarding the location of vehicles in parking lots. Users generate the actions in the same order.

Time dependency: Our system is a real time system, meaning the system returns the locations of vehicles in real time. The system should update automatically whenever there is a change detected by a sensor.

## Hardware Requirements

As shown above, the system will utilize a client-server model.

**Client-Side**

On the client-side, users may access the server from any web-enabled device. These devices may include commercially available smart phones and tablets using the Apple iOS and Andriod operating systems.

User devices may further include desktop and laptop computers having the following specs:

* Memory/RAM: 8-64 GB RAM
* Processor: Multi-core models
* Storage: At least 256 GB hard drive, 7200 RPM, preferably SSD (fastest), HDD also good... buy as much as you can afford, you can always add external hard drives. Avoid SATA drive if possible.
* Graphics Card: Depends on video editing software. For example, Premiere Pro and Davinci Resolve do well with both AMD and NVIDIA. (Minimum 2GB memory)
* Operating System: Windows 10, MacOS, Linux (Your OS will likely determine the editing software you go with)
* Nice Size Screen: 19-21 inch minimum

**Server-Side**

Hardware specifications for the serve may include:

Processor: 4 CPUs with 3.0 GHz processor

Memory: 16 GB RAM

Storage: 1 TB hard drive (minimum 7.2KRPM)

Operating System: Windows Server, Linux, macOS Server.

# PART 3 – PROJECT SIZE ESTIMATION

A picture containing text, screenshot, parallel, font

Description automatically generated

# PART 3 - PROJECT MANAGEMENT (PLAN OF WORK)

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

# PART 3 - Reference list

Brenckle, J. and Stroisch, C. (2022). *Chevrolet and Ford Full Size Pick-Ups Most Stolen Vehicles For Second Year in a Row | National Insurance Crime Bureau*. [online] www.nicb.org. Available at: <https://www.nicb.org/news/news-releases/chevrolet-and-ford-full-size-pick-ups-most-stolen-vehicles-second-year-row> [Accessed 15 Jun. 2023].

National Insurance Crime Bureau (2023). *Archived Tables*. [online] Insurance Information Institute. Available at: <https://www.iii.org/table-archive/21263> [Accessed 15 Jun. 2023].

Netwatch (2023). *Crime Affecting Car Dealerships Is on the Rise*. [online] Netwatch North America. Available at: <https://netwatchusa.com/industry-solutions/automotive/> [Accessed 16 Jun. 2023].

Olsen, P. (2022). *Catalytic Converter Theft: 10 Most Targeted Vehicles*. [online] CARFAX. Available at: <https://www.carfax.com/blog/catalytic-converter-theft> [Accessed 15 Jun. 2023].

Russo, D. (2023). *Cost to Replace a Catalytic Converter (2023) | ConsumerAffairs*. [online] www.consumeraffairs.com. Available at: <https://www.consumeraffairs.com/automotive/cost-to-replace-a-catalytic-converter.html> [Accessed 15 Jun. 2023].