CAR RENTAL PROJECT

TC Table of Contents

1. Introduction

1.1 What the Problem is

A new up and coming car rental service is wishing to have a user interface that will allow their customers to view the models, descriptions and prices of different cars available. The user has the ability to register and log in to the Web App and track their rental plan. The Web App will be responsive, allowing for the customer to view it on any device, from tablets to mobile phones and desktop computers. The administrator will also be able to login through the same form but have the ability to add/remove new car rentals, change prices, and so on. Potential customers should be able to view all the cars available to rent even without logging in as well as rent without having an account, though the option is provided upon checkout.

1.2 Goals for the Project

This software, called 'Ghost Rental Data', will allow for the company to access their database securely and safely in a user-friendly online environment. Allowing for them to change car information with ease. The software will be in sync with the both the Web App, allowing for real-time up-to-date services for their customers.

Both registered and non-registered users will be able to search car rentals by price, model, seating and any other potential searches. They will also be able to select and pay for the service. The consumer would choose on checkout if they will come to pick up the car of if they

want the car to be brought to them. The company would instantly get that service demand through their 24/7 car rental support built directly into the software and either reserve the car for pickup or send out a pickup truck carrying the new rental car to the desired location upon time request.

There should be a Web App version for the software to connect with for those who wish for a quick car rental servicer. The Web App will be responsive to any device using it. This allows for consumers to access the service from any sort of hardware device: tablet, computer, mobile devices, and so on. The software itself will be available on all computer platforms that are running any aspect of Linux, Windows 7, Windows 8, and Mac operating systems.

The software will also be easily available to Windows 10 when it comes out later in the year. Besides, computers, the software has a minimal version for tablets for those working for the company to easily navigate through customer orders when they are on the road or simply away from the computer. Tablets are not required for full use the software however, in case the company has a budget that does not allow for them. The compatibility will still be available whenever they wish to provide their employees with them nonetheless.

1.3 Stakeholders

Several different types of stakeholders can be noted when it comes to our software.
The most obvious are those that requested for this software: the CEO and company board members. We made note to critical details on what the company wants, how they want it, and

how it should benefit both them and their clients. We plan to keep them in the know 24/7, keeping them informed on what is being planned for their new software. This allows for the CEO and board members to be aware of what they are getting from their new software.

Any local or district managers are also influenced by the software's stability. How easy the software is for the employees in turn represents how well the manager is able to guide them through it.

 The employees will also be affected by the use of the software, it is not only their obligation to learn the software, but for us to make it easily learnable. In this case there will be days dedicated to teaching the employees how to run the software for an easy navigational experience. Their input will also go into the testing process, making sure everything is to their liking and runs efficiently and effectively.

Customers are also vital stakeholders, and with a mobile/web car rental service, one can find the convenience of obtaining a car with circumstances ranging from their current vehicle breaking down to needing a means of transport during vacation. This puts an extreme time saving benefit of the customer having to search around to even find a car rental building. With the car rental service, they will help you locate the store as well as even take the car to you, if the customer so chooses.

1.4 Motivation for the Project

With this company in need of a better system, we felt it was our obligation to help them in their time of need. To develop such a system that would not only ease the burden on the company's customers, but the company itself. Our team has an immense amount of knowledge when it comes to problem solving, programming, and communication. Not only would we strive to give the car rental service everything they desired, but we will continue to make sure the software is at its very best and beyond. Each one of us will always and will continue to give 100% and more to making the transition a breeze for the car rental service.

Our developer and team leader Jon brings to the project mass amount of web programming. His expertise will coincide with our visual designer and project manager expert, ▮▮▮▮, whom has experience with databases, web programming, as well as mobile programming. Also, having multiple heads working on the programming abilities helps in ensuring no errors are implemented and every single detail is put into place. Everything is wrapped up together with our project relations expert and co-manager Nick. With his experience with planning and monitoring the team, he ensures that no step is missed right from the design of the software to testing and software management.

1.5 Process Flow Preview

For our process flow, we plan on taking the iterative route, as we find communication essential throughout the development process. In order to plan all aspects of the project in detail, we feel that contacting the car rental service and having those in charge be in the same room when

the planning is taking place. It is our way of discussing requirements and develop important notes that will help in constructing the overall feel and idea. The modeling process in its own right we feel is not a start to finish process. There will be times that we may have to go back to certain portions within the modeling activity to ensure a sufficient model. We wish to make sure that if we miss anything, we do not figure that out in the construction stage.

Lastly we find the construction framework activity to require communication with the car rental service as well. It would allow for us to get small tweaks out of the way as well as have a kind of testing process on the interaction between the constructed elements and those that will use the software directly. We feel that allowing for the users to view how the software is made will ease the transition during the deployment stage.

Analysis and Design

2.1 Plan for Requirements Engineering

Inception Task:

The goal for the beginning is to identify the business case created by the stakeholders. We want to get a grasp on the market these rental cars are for, analysis how often the software will be used and to ensure the final product can handle all of the rental company's customers. These are just a few questions we asked the stakeholders. With this in mind, we took into consideration that the CEO and company employees may have different viewpoints on who the targets consumers are, as employees are more likely to have a direct contact with them and the CEO's response may be expectations. To get a basic understanding of the project, here were some more questions we asked:

> *What are the basic functions? (What do you want the website and software to do? - What tasks/problems is the product supposed to accomplish?)*
>
> *What sort of customers are you targeting with this new website? (Who is going to use it?)*
>
> *Can you show us the environment where this software will be implemented?*
>
> *Will there be any issues or constraints that may affect the planning and construction?*
>
> *Are there any other people you suggest we ask these questions to?*
>
> *Is there anything else you want to add?*

Elicitation Task:

Our goal at this stage is to identify the problem, propose solutions, and talk amongst each other on the many different approaches. Meetings are scheduled with the software engineering team and the stakeholders in order to get a more refined understanding. The plan is to get a grounded idea of what the objectives for the system are, what should be accomplished, and how the overall system fits into the car rental business. Overall, lists will be created to understand who the stakeholders that took part were, descriptions of the technical environment, usage scenarios and a list of requirements were created at this time.

Elaboration Task:

Information gathered from the inception and elicitation stage are grouped together and refined during this stage. A model is conducted that clearly portrays the numerous conditions of the software function and behavior. Scenarios were created to describe and aid in understanding how the customer will interact with the website and how the employees will interact with the software. Any attributes are to be defined as well as how each function interacts with one another.

Negotiation Task:

With any conflict that would come up in need of a resolution, the team and the stakeholders would discuss them through to come up with a resolution. If there are too many requirements asked by the stakeholders, we would have them rank each by their importance. Anything that turns out to be the bottom requirements by all, if not most of the stakeholders may have to be omitted to save time and money.

Specification Task:

During this task, we plan to create a software requirements specification template. In this template we will note down the overall purpose of the project and the intended audience. Descriptions regarding the product features, user classes/characteristics, operating environment and design will be included. Also included are safety and security requirements, quality attributes, and what interfaces are to be used with this software. Along with this written document, a 'mockup' or 'prototype' will be created to get a visual outlook on the project.

Validation Task:

During this stage, any requirements stated are to be ensure that they are clearly defined. No miss-interpretation should be present and any that exist should be resolved. All and any sources or stakeholders used in the project planning should be proved legitimate and their input is 100% valid for use. All the requirements should be congruent with the overall objectives and can be easily understood. Any hard-to-understand phrases should be rewritten and discussed over with again with the stakeholders and team members.

Requirements Management:

Any changes that may occur throughout the project stages should be handled with clarity and care. Any potential changes would be looked over, discussed and determined if the time allotted for the construction of the project can allow for such a change - that is if it is agreed upon by the stakeholders and software engineering team. Requirements Management will

occur throughout the project process flow as changes or alterations can occur under any circumstances.

2.2 Functional Requirements

Hardware Requirements:

The software should be ran on any sort of desktop or laptop environment, regardless of the operating system. The software also has the potential of running on tablets, but with a more simplified version. Essential input/output devices are keyboards, mouse, and printers; nothing else is required but can be recommended if desired.

Website Interface - Primary Tasks:

- View all available rental cars
    - Connects with the database through JavaScript to call all car objects and display them accordingly
- Search for desired car by model, seating capacity, and cost
    - A search bar will be implemented on the website that will search the site based on the input
- Select their desired rental car
    - A button will be displayed under the rental car previews that will allow for the selection
- Allow for registration
    - registration display form

- Allow the customer to log in

    - log in display form

- Allow administrators to change what cars are available for rent

    - Administrators will log in through the log in display form such as customers

        would, but would have a different looking interface to allow for them to add and

        remove options.

Website Interface - Secondary Tasks:

- Allow for rent cancellation by customer

    - Only for users who are registered, there were be a cancel button on the display

        page that shows all rents present by customer

- Allow the user select and provide a payment type

    - Display form - with options for typical card information - type, number, code on

        back, month, year etc.

- Authenticate any user logging in

    - communicates with the database to verify the inputted username and password

        is correct

- Authenticate any form of payment

    - communicates with verifiers to confirm the input information on the card is

        correct

- Calculate tax rates, customer totals

- algorithms implemented within the code that will calculate all totals 'behind the scenes'

- Send an email to verify a registration

  ○ Once a customer registers and click the complete registration button, an email will be sent to the email the customer provided, provided a link in the email that will allow for a verification that this is the said-customer.

- Display a transaction summary / email a summary

  ○ after customer submits their payment and it is confirmed client-side, a transaction summary will display on the website and a copy will be emailed to the customer

- Select date of desired delivery (& choose delivery destination), or date of planned pick-up at nearest store location

  ○ Calendar options will be displayed that will allow the user to select directly on the calendars - if they want it as soon as possible, there will be a checkbox for that option

- Select length of rental and planned drop-off date

  ○ Calendar options as well

- Store customer information in the database

  ○ When registering, the customer will provide their information in the form - this information will be sent securely sent and stored in the database

Company-side Software - Primary Tasks:

- Allow company workers to open and view customer information

    - navigation option that will allow for search of customer by name or rental car

    - implementation of search bar

- Track customer payments

    - displays with the customer's information

- Keep and display available rental car records (for damages purposes and such)

    - Searching and viewing much like with customers

Company-side Software - Secondary Tasks:

- Print invoice

    - if print option is selected, a printable version of the page being viewed by the

      worker will display with a confirm print button to selected printer

- Calculate total of all transactions

    - algorithms behind the scenes

- Allow for rent cancellation by company worker

    - a button will cancel a rent request by a customer

2.3 Non Functional Requirements

Performance Requirements:

- Ability to maintain mass amount of customers on the website at once without crashing

- Speedy performance / transmission of data

- Send any emails immediately

- Being logged in should allow for customers to quickly make payments without reentering information, and allow for any potential registered perks the company may have

- Have a quick recovery time if anything were to go wrong

- Display accurately and efficiently on all devices (responsive view)

Security Requirements:

- Secure any transmissions of private information between the customer and the company

- Prevent any potential threats such as SQL injections through the forms or search boxes.

- Prevent third party users at administration level

- Verify website security certificates (that lock in the address bar)

- Prevent false information from being used as payment

- Prevent false email inputs from being used when registering

Quality Attributes:

- Maintain a user friendly environment that is visually appealing

- Easy to see and use navigation

- Maintain readable content

- Searching cars should be accessible to people who are and are not logged in

- Selecting and making a payment should be available to customers who are and are not logged in

Screenshot Mockups:

Company Name   Register   Login   ≡

Price

Model   Seating Capacity

Description

Rent this Vechicle

2.4 Use Cases

Use Case #1: *Checking Car Status*

Primary Actor: Car Renter

Goal in Context: To see if car rental is on route to Car Renter

Preconditions: Car Renter has made a rental in the past and is nearing the time that the car

should be delivered to the car renter

Trigger: The Car Renter knows the current information of where his/her car is at.

Scenario:

1. Car Renter: Logs onto Car Rental service site (Enters Username/ Password)

2. Car Renter: Selects "My Car Rental" on website.

3. Car Renter: Selects "Car Status" on website.

4. Car Renter: Observes status of car to know current information on car.

Exceptions:

1. Car Rental service Username/Password incorrect: Car Renter is sent to main page to re-enter credentials

2. Password is not recognized: E-mail is sent to Car Renters current E-mail and prompt to enter new password

3. "Car Status" is unavailable: Car Renter is prompt to call Car Rental Service for further details on Car Status.

Priority: Essential, must be implemented

When available: First increment

Frequency of use: Couple times per day/ week

Channel to actor: Car Rental Website

Secondary Actors: Car Rental Customer Service

Channels to Secondary Actors:

1. Customer Service: Phone line / Online chat support

2. Automated E-mail: Sent to Car Renters E-mail for status request

Open Issues:

1. Should there be an option to opt out of the Car Rental on the day of the Car Rental date?

2. Should the Car Renter have an option to change what car he/she wants after confirming what car they decided on(i.e.: going from a stock model to luxury)?

3. How much time does the Car Renter have on their account before an amount of time has passed before the system automatically logs them out due to inactivity?

Use Case #2: *Change Car Rental Date*

Primary Actor: Car Renter

Goal in Context: To change date of when Car Renter wants car

Preconditions: Car Renter has made a rental and wishes to change date of car rental

Trigger: Car Renter has switch the date of when Car is being rented on

Scenario:

1. Car Renter Logs onto Car Rental service site (Enters Username/ Password)

2. Car Renter Selects "My Car Rental" on website.

3. Car Renter Selects "Car Rental" on website.

4. Car Renter Selects "Change Date" on website.

5. Car Renter is prompt on website to select new date for car rental

6. Car Renter selects new date for car rental and accepts the new date

7. Car Renter is prompt to enter "password" sent to his/her E-mail to validate the switch

8. Car Renter enters "password" and sees the new date for the car rental

Exceptions:

1. Car Renter denied request, Car Rental date is too close to actual date of the rental, cannot make change.

2. Car Renter submits wrong "password"; is prompt to re-enter otherwise the date will not be changed

3. Car Renter encounters error on accepting new date; is prompt to call Customer service and talk to representative about issue.

4. Car Renter enters wrong date and accepts changes; is prompt to call Customer service and talk to representative about issue

Priority: Essential, must be implemented

When available: First increment

Frequency of use: Once per quarter to Registered Members

Channel to actor: Car Rental Website

Secondary Actors: Car Rental Customer Service

Channels to Secondary Actors:

1. Customer Service: Phone line

Open Issues:

1. How many times can a Car Rental member request this service in a given period of time?

2. Should the Car Rental member be charged extra for a convenience fee for using this service?

3. How much time does the Car Rental member have before this option is void to him/her?

2.5 Models

Sequence Diagram

One of these three
must be derived

3. Project Plan
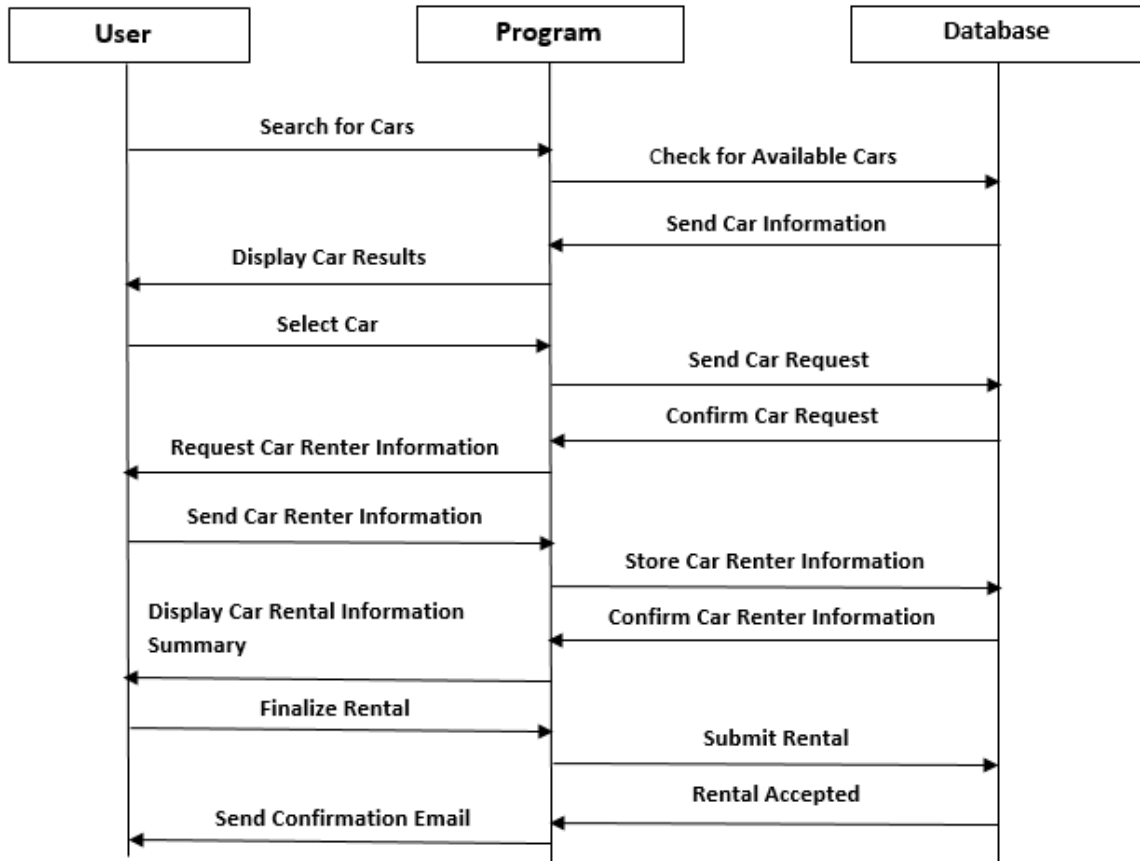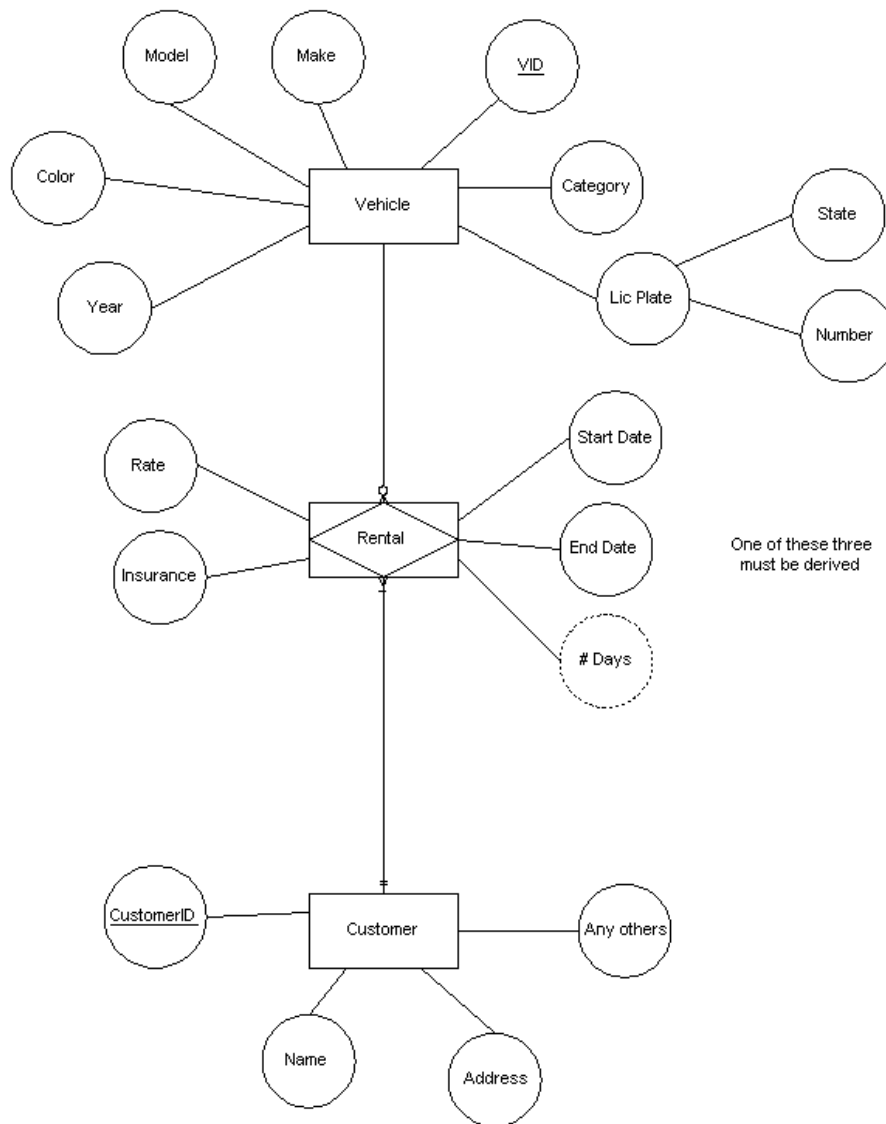
3.1 Task Descriptions

Stakeholder Meetings

The stakeholders and the software engineering team conducts meetings in order to get the full grasp of the problem at hand, obtaining any and all information needed. Conflicts and negotiations would be conducted during this time and any time throughout the project process flow.

Design Models and Mockups

Designing the models and mockups help to ensure clarity in view of the project as well as how it works. Stakeholders are to sit through this process as drawings are created.

Database Creation

A database is created using the models to provide storage for customer information, vehicle information, and payment reports. Testing of the database is ensured at this point.

Employee Software Creation

The software that is to be used by the employees will be designed using Visual Basic using the guide of the mockups, requirements, and models. The software will act as a simple and easy to

understand user interface to 'browse' and among other functions the database - the

information stored including customer information, vehicle information, and payment reports.

Website Creation

The web site will be designed using Ruby on Rails programming language using the guide of the

mockups, requirements, and models. The website will be implemented using high-quality

design techniques. It should allow for users to immediately see the rental cars that are

available. They can also have the ability to search for a certain vehicle they desire. Once they

select they can register or log in to store any private information they are about to give to the

company in order to obtain their vehicle, although registering is not required.

Testing

Testing will be implemented on both the website and software. Test cases may be used to

guide and understand the basic actions of both customers and employees. Any bugs or errors

that occur will be identified and resolved.

Finalization and Reports

All testing and function processes are finalized at this stage. Reports will be created to ensure

all information and functionality is clear in order to make the user manual and to help ensure

employees can use the software with ease.

3.2 Task Assignment

Assignments were distributed evenly among the group assigned to the project. All three worked together in the project planning, sharing in the opportunity of any models and analyzing all specifications made by the proprietor.

Katie was given the objectives of building the interface of the web page using Ruby on Rails, PHP, and JavaScript. She created the interaction between the web page and the database that will be held by the company. The software will also connect with the database.

Nick created the use cases that will be used as a guide for the testing process. He also talked to the stakeholders to gather information for the project planning process. His work coincided with Katie in order to get all fields from his diagrams implemented into the database.

Jon was given the tasks of creating the software for the company to use using Visual Basic. He was also given the objective to test all aspects of the both the web page and the software. His findings were reported to each person in the group and discussions were made on how to fix them.

Reports were created throughout the process by all three group members and gathered to accurately and sufficiently create this final report.

3.3 Deliverables and Milestones

We had four major Milestones in this project:

1. Completion of Requirements Gathering.

2. Completion of Design and code.

3. Completion of Testing.

4. Completion of Demonstration.

These milestones were all completed on schedule and yielded a Deliverable at the end of each.

Our four corresponding deliverables (respectively) in this project were as follows:

1. A completed list of all stakeholders needs to be met.

2. A finished and easily navigational GUI (Graphical user interface).

3. A clean "bill of health" for our software.

4. Satisfied stakeholders and customers after demonstration and launch of software.

3.4 Project Schedule

The first month of the project start date (April) was used mainly for requirements setting and

Intel gathering. This took (over the course of four weeks) roughly 37 man hours. Through the

last two weeks of April and part of May, we were able to begin designing, which took 52 hours.

Around the same time, testing began and continued throughout the remainder of the project.

This total time was about 70 hours. We then wrote out the user manual, and gave a final report

of the software. The time this took was about 40 hours. We then had a minor period for

demonstration and final adjustments which totaled 10 hours. We finished the project in late

July. The percentage breakdown was as follows:

Requirements: 37 hours - 17.71%

Design and code: 52 hours - 24.88%

Testing: 70 hours - 33.49%

Manual and Final Report: 40 hours - 19.14%

Demonstration and Adjustments: 10 hours - 4.78%

Total: 209 hours - 100%

4.   Testing

4.1 Features to be tested

We will start by using both static and dynamic testing strategies. The static strategies will

include reviewing the basics of the application whereas the dynamic testing is based on actual

code execution.

The features we tested were as follows:

- To ensure that the application itself ran (Dynamic)

- log-ins worked efficiently and consistently (Dynamic)

- accessing the web app from multiple platforms to ensure cross compatibility (Dynamic)

- checking page load time (Dynamic)

4.2 Test Cases

The following are examples of test cases we implemented:

- Dropdown fields should have first entry as blank or text like 'Select'

- Amount values should be displayed with correct currency symbols

- Graphical User Interface

- All fields on page (e.g. text box, radio options, dropdown lists) should be aligned

  properly

- Upon click of any input text field, mouse arrow pointer should get changed to cursor

- Check all pages for broken images

- Check all pages for broken links

4.3 Testing Schedule

The testing should begin right after the project itself begins. Keeping up on testing will ensure

that any mistakes are caught early and corrected immediately.

5. Conclusion

5.1 The Problem and Solution

The problem that undergo for Ghost Rental was a lack of real time car renting for their customers. They wanted a website that would allow for users to purchase their service with ease and admiration. Customers should have been able to rent and view vehicles with or without logging into the system. Requests should be transferred directly into the database and displayed on the software within the company building to allow for a proper response and review to commence.

The solution was to provide Ghost Rental with an user-friendly web application that would allow for customers to access and use on a wide range of devices: desktops, laptops, mobile devices, tablets. The website is designed to stay up to date by giving administrators the ability to change/add/remove any featured vehicles on the site. The website will verify and store any information the user may input when making a rental purchase request. This will appear in real time onto the software used by the employees. This software provides an easy-to-use interface to allow for simple access to rental requests and customer information.

5.2 The Team and the SE Process

The Software Engineering process we used was the spiral method. In this method, we start in the middle of the model, and spiral outward, allowing all departments working on the software to be an active part of every aspect of the engineering. Each individual department will be able to work and test during the concept development, system development, system enhancement, and system maintenance phases of development.

5.3 Engagement of Umbrella Activities

Four of the main Umbrella activities we used were as follows:

1. Software Project Management - Which was used to lead the project and ensure that the project was controlled, monitored, and on schedule.

2. Formal technical Reviews - This activity was essentially implemented for peer review. Having new and fresh eyes to view code and ensure that everything met the requirements.

3. Reusability Management - This activity was used to help us create flexible and generic assets that may be reused for future projects or for this project in other regions. This would cut down on cost and help with consistency.

4. Risk Management - This activity was used to assess and identify potential risks with creating the software such as assuring that not too much money be spent in assets on the project.

5.4 The Stakeholders that Benefited

After release of the product, all of our active stakeholders benefited from the software. This list includes but is not limited to; the company shareholders, the customers, the project development team, upper management, line managers, consultants, and many more.

5.5 The Organization's Benefits

Our Organization benefitted greatly from the production of this software. Along with a hefty pay-day, we have built a report with this company and expect future projects to be assigned to our company. As well as a report with this business, word of our software will travel, and typically for a car rental agency, they deal a lot with high class businessmen. If these high class businessmen get to see our software in action, we will more than likely receive future job offers from other companies as well.

6.   **User Manual**

6.1 Software Description

The web application will allow for customers to view vehicles currently for rent either by browsing or searching directly. The customer can register to save any information they may input during the payment/checkout process or they can log in to use any previous information they provided with the company already.

The software the company employees will use will be a user interface to view, edit and modify the customer information, customer requests, payments, and vehicle information. There will be a notice section for the employee to be notified in real-time of any recent purchases and whether they need immediate action. They employee will also have the ability to search customer and vehicle information as well as print out any reports that may be necessary.

6.2 How to Use the Software

The web application designed for the customers of the car rental service should see the main page upon entering the URL into the address bar. There they can see the features and promoted vehicles up for rent as well as clearly identify the search options. Upon searching and clicking a vehicle, they would be directed to a page with more description on the vehicle consideration for rental. It is at this page the user can click the button to rent the vehicle and

have the option to register (to save the information they plan to input) or log in (use previous information used before with the company). The user will input card information which will be securely transferred and verified by the company and the processing will begin on the company's end.

The software when opened will be split into two main sections: the active real-time customer processing that the employee is assigned to, and the searching/browsing section. The real-time section will have alerts that the employee can click to view in detail on when customers that have successfully checked-out need their vehicles and how (whether it be pickup or delivery). The searching/browsing section is subdivided into three main sections: total payment reports, customer information, and vehicle information. When the section is clicked. The customer will be taken to a new 'page' with a more detailed report of what they are searching. Reports may not be modified, only printed and search for specific payments. Customer information and vehicle information can be modified however, with edit buttons next to each sub section (description, name, etc.). Customer payment will be properly hidden accept for the last four digits of the card used for payment.

6.3 Troubleshooting Common Problems

The Website

Problem: Loading incorrectly / Session has Timed Out

- Make sure your internet is working properly. You may need to restart your router. Refer to your router's user manual to accomplish this task.

- Clear your browser's cache and re-launch the browser.

Problem: Invalid Login

- The credentials used to log in was not found in the database, input the credentials again
  in case of mistype

- If invalid login persists, pursue the lost password option to obtain a new password. If
  error 'email not registered' Please attempt to register instead with the email.

Problem: Page Not Found

- Make sure you have entered the URL correctly.

- If the URL is correct, contact the company through telephone or email to report the
  missing link.

The Software

Problem: Customer requests for vehicles are being delayed by a significant amount

- Make sure the internet is working properly, most of the time the case is the internet
  connection either used by the company or the customer themselves.

- If the problem persists, contact someone of higher authority who can verify if the server
  is having down time issues. It may need to be waited out before the problem can
  continue. Emails will be automatically sent to customers within 15 minutes of the initial
  rental request about the delay if an employee does not respond to it in time.

Problem: Customer/Vehicle Not Found

- Upon searching through the software, this means that the customer or vehicle is not currently in the database or may have been removed. Review recent changes to the database by the company to see if any changes had occurred.

- Contact someone of higher status to verify that the customer or vehicle is not in the database if you suspect they should be when no recent changes come up.