Abbreviated Project Plan

Description:
We are building a solution for Vaca3354, a vacation home rental business. In order to do so, we will be developing a website as well as a mobile application for users to access it.

Users will be able to:

-Schedule and manage rentals.
-Leave reviews.
-Learn about the surrounding area of their rental, i.e. attractions, night life.
-Communicate with the company through phone or live chat.

The administration of the company will be able to:

-Manage properties.
-Schedule maintenance on the platforms.
-Make edits to various properties such as the rent or the price of amenities.
-View details of reservations.
-Communicate with guests.

These features will make it so we have a fully-functioning and efficient platform for users to rent properties without the hassle. Our budget for this project is $2 million and we have 12 months to develop the website and mobile application. This budget will be divided into salaries for the team, server costs and costs for various tools required for the task.

Project Organization:
For our developers, we will have two separate teams: mobile and web. In these two teams, they will be further divided into front-end and back-end. We will also have a UX and UI team to help ensure the front-end is polished and up to our standard of design. These teams will most certainly be small due to the budget provided. Salaries for each member of the team will be rather high due to the competitive market. We are looking to have at least three front end and two back end developers for each web and mobile. As for UX and UI, we want to have two to three each. All this equates to fourteen to sixteen total members for the project. Each team will have a designated leader, a coordinator, a record keeper and a conflict manager.

Risk Analysis:
A major risk with this kind of project is a data breach. The likeliness of this occurring is dependent upon how well we set up our system up front. We are aiming at a 3-5% chance of this occurring. If one were to occur, this could lead to hackers stealing personal information from users such as addresses, credit card information, and phone numbers. This is something that can happen at any given moment unfortunately. In order to prevent this, we will need to have a breach preparedness plan to be ready for it. On top of this, we need to have a team that can handle it in case of it happening. Lastly, we need to know the state and federal requirements to avoid any fines if we don’t follow them.

A member of the team leaving is also a risk that can happen. The likelihood of this happening is higher than a data breach but still relatively low, probably closer to 10%. Whether it is due to personal reasons or not, this is something we must be prepared for. This can be done by providing proper working conditions and incentives to stick around. If something does happen that requires them to leave, i.e. a medical condition, we must have other options to replace them. We will have protocols for this sort of situation to try and find another option as soon as possible. If it is a valuable member, then this can lead to delays in the project as it will not be easy to find a replacement.

Hardware and software resource requirements:
We will need to have access to cloud servers so we can store information. This can be costly depending on how much space we will take up. Since we are expecting roughly ten to fifteen thousand visitors daily on the app and website with at least three to five thousand daily active users. Since we plan to use cloud servers, it will be a little cheaper in comparison to standard data servers. This can cost around $150 a month for a server with 200 GB and 6 GB of RAM.

Workstations will be required for the team to accomplish their tasks. The cost will vary, but for fourteen to sixteen members, we expect each workstation to cost at least $1000-1500 per member. This is a priority so this will all be taken care of at the start of the development process.

Lastly, we can look at security for the rental properties, we will need to invest in securities such as “Ring” for home security. This will cost around $150 per property and will add up as we continue to acquire more properties.

General Schedule:
-Project Start - January 1, 2020
-Server setup should be finished by the end of January
-Front end will take at least three months. By the start of Spring, we should be eighty percent done. If we run into problems, this should delay the process no later than the start of Summer, i.e. May or June.
-Back end should be taken care of by the start of Spring barring any setbacks. Both front end and back end should be in sync to maintain consistency.
-QA testing will begin after this and we will start Alpha testing through the Summer.
-Bug fixes and finalizing while reviewing with customers will happen at the end of the summer as we look to release by the end of the year.
-Advertising will begin in the Fall
-If everything goes according to plan, we will have the release by January 2021.

If there are delays, the project may get pushed to Spring 2021. The delays can happen at any moment if we lose a member or a bug occurs.

Detailed Schedule with Dates and People:

- Requirements gathering, project definition, risk analysis and project plan: Completed
    - People: backend and frontend teams as well as members of management
- System models and architecture design created: Feb 15, 2020
    - People: backend and frontend teams
- Specify the objects and interfaces used within the system: Feb 28, 2020
    - People: back and front end teams
- Prototypes of system created: Feb 28, 2020
    - People: back and front end teams
- QA test cases planned and criteria gathered: March 15, 2020
    - People: front and back end team leads
- Specify components of software to develop from scratch and those to reuse/purchase off the shelf: March 15, 2020
    - People: back and front end teams

- Software Components:
    - UI of app: March 30, 2020
        - People: front end team
    - UI of website: March 30, 2020
        - People: front end team
    - User account capability: April 15, 2020
        - People: front end team
    - Administrator account capability: April 15, 2020
        - People: back end team
    - Rental home profile design and capabilities: April 30, 2020
        - People: front end team
    - Communication capabilities between administrators and users: April 30, 2020
        - People: front end team
    - Database: May 15, 2020
        - People: back end team
    - Security: May 15, 2020
        - People: back end team
- Documentation: should be done concurrently with above components
    - People: record keepers of both coding teams
- Unit testing: should be done concurrently with above components
    - People: both teams
- Component and system testing: May 30, 2020
    - People: both teams
- QA testing: May 30, 2020
    - People: both teams
- Alpha testing: June 15, 2020
    - People: users and management
- Final changes after testing: October 15, 2020
    - People: both teams
- Advertising methods identified: November 1, 2020
    - People: management
- Advertisements created and released: December 1, 2020
    - People: management
- Release system: January 2021
    - People: management
- Project assessment: January 2021
    - People: teams and management
- Maintenance: January 2021 and onward
    - Bug fixes
    - Add new functionality
    - System updates
    - People: both teams

Monitoring and Reporting Mechanisms:

The reports that should be generated include:

- Progress reports completed by teams each week
- Evaluations of resource allocation by administrators bimonthly
- Evaluations of schedule progress each week including:

    - What was completed
    - What was scheduled to be completed but wasn’t
    - When the new expected date for completion is
    - How that will affect the rest of the schedule
- Evaluations of quality by administration every month
- Evaluations of the budget bimonthly including:
    - Cost estimate
    - How much it actually cost
    - Explanation of discrepancy
    - Affect on overall budget

Monitoring mechanisms:

- Screen monitoring of employees
- One member of management present at all times

Interview Questions

What features would you like on your website?
Would you like an app for mobile devices?
Would you like an online live chat feature?
Do you prefer a lighter or darker theme to the site/app?
Do you want a maps application built from scratch or implement an existing version such as Google Maps?
Would you like to grab reviews from other sites?
Do you have any plans to expand into other similar types of markets in future, like booking experiences while they are renting?
Should we incorporate international languages to expand the market?
Would you like to include the ability to book a flight alongside booking a room?
Does the company have a general maintenance check on a schedule, and if so do you want to show a general unavailability cycle to the customer?
Is the information about properties stored in an external system or is that our responsibility?
Do you want a two step verification process?
How many properties do you currently have?
What is your rate of growth?

User Stories/Scenarios

As the data management team we need to be able to add a new property to the database within 7 business days of the company acquiring the property so the company can rent it out in a timely manner.

As the data management team we need to be able to back up all of our critical systems and important personal data on the databases so that I can reboot the entire system in a short timeframe if something brings the system down or erases important data.

As a website manager I want to be able to add a new posting to rent out a property within 3 business days of the property being added to the database so that I can rent out new properties as soon as possible.

As a website manager I want to be able to collect and store information related to user and data traffic and usage of the website so that I have accurate information about the traffic to the website and can show my superiors about the approximate impact of different ad campaigns.

As the data security team I want to be notified of attempts to bypass security so that I can handle them in a timely manner and they don’t have time to breach the system.

As the data security team I want all of the information transfer should be encrypted so that I can be sure that passwords can’t be easily read through network communications.

As a guest I want to be able to view property information based on numerous different factors so that I can find the best fit.

As a guest I want to be able to easily compare properties so that I can find the one I like the best.

As a guest I want to be able to post images of the property so that other people can see them.

As a guest I want to be able to save certain rentals so I can go back and review them at a later date.

As a guest I want to be able to pay for my rental in a safe and secure manner through the company’s website so that I can pay immediately.

As a guest I want to be able to see activities and events going on within a 10 mile radius of where I am staying so I can plan out my vacation more easily.

As a guest I want to be able to request a maintenance order and it be fulfilled within 48 hours of sending it so that my stay is more enjoyable.

As a guest I want to be able to back out of rental or leave early so that I’m not forced to go if I change my mind or I don’t want to stay as long.

As a guest, I want to be able to refute bills charged to me if I believe they are incorrect so that I am not taken advantage of.

As a guest I want to be able to leave reviews about my experience so that if I have a complaint or praise other people can take it into account when booking with this company.

As a guest I want to be able to add more people to the rental property so that I can have more people stay with me.

As a guest I want to be able to report issues with amenities and facilities so that the company can resolve them and I can enjoy the amenities.

As a guest I want to be able to contact the company through phone call or live chat for help at any time so that I can receive help 24/7.

As a guest I want to be able to receive confirmation of my rental with details of my rental by email so that I can ensure my reservation was made and I can look back on the details at a later date.

As an administrator I want to be able to require a username and password so that the system is more secure.

As an administrator I want to be able to update the database with reservations immediately after they are made or canceled so that I can avoid double booking.

As an administrator I want to be able to add, remove and edit properties manually so that if there is an issue with the system it can be done manually.

As an administrator I want to be able to view all of the details of all of the guests reservations so that if there is a problem we have access to the proper information to fix the problem.

As an administrator I want to be able to set certain users separate permissions to access data so that I can be sure that there is minimal risk that employees have access to and can edit information that they don’t need access to for their job.

As the company, I want to be able to charge a fee if a guest backs out of/shortens a rental so that I am compensated for reserving the rental for them.

As the company, I want to be able to bill guests if they cause damage to the property so that I am not losing money if they damage something.

As the company I want to be able to see a report of both positive and negative reviews so that I can offer consolation for people who had a negative experience and thank those who had a positive one.

As the company I want to be able to have issues with facilities and amenities fixed within 24 hours of them being reported so that I won’t lose customer satisfaction.

As the company I want to be alerted if housekeeping has scheduling issues so that they can be resolved in a timely manner.

As the company I want to be able to communicate with the guests quickly and efficiently about issues and resolutions so that we can make their stay as enjoyable as possible.

As the company I want to be alerted about data breaches immediately so that I can communicate that information to guests and offer them consolation.

As the company I want to be notified of inclement weather near a rented property so that if it subsequently damages it, we can complete repairs in a timely manner.

As the company I want to prompt the guests to leave a review after their stay so that we can gain valuable feedback on how to improve our business.

As the company I want to be able to have facilities signed up for insurance and properly secured so that I can be covered by insurance and be able to not be held accountable for damages done to my properties by tenants.

As the development team I want to be able to manually undergo maintenance if needed so that it can be done without needing a middle man.

As the development team I want to be able to have a test server with usable and valid data so that I can make sure that there are not many bugs that were overlooked and to make sure that rolling out the changes won’t bring down the real system.

<u>Requirement Specification</u>

User Requirements

Functional Requirements:

User end

- Option to:
  - Select check in time
  - Select check out time
  - Select destination
  - Rate
  - Post review
  - View reviews
  - Pay for rental
  - Save rental for future viewing
  - Select number of adults/children
  - See rentals based on suitability (preferences like kid friendly, pet friendly, wheelchair accessible, etc.)
  - Send a message to company with questions about rental
  - Request a maintenance order

- Viewable:
  - Images of rental location
  - Map of area
  - Availability
  - Number of bed/baths
  - Cost per night
  - Amenities (WiFi, complimentary breakfast)
  - Policies

Administrative end

1. Require a username and password for entrance
2. Option to add new properties
3. Accept a maintenance request
4. Change rates
5. Upload photos
6. Update description, policies, amenities, availability, and other information about the properties. This ability should be restricted to only certain stakeholders with proper administrative rights

7. Show Current description, policies, amenities, availability, and other information about the properties. This ability should be restricted so only the appropriate information is available to each stakeholder.
8. Respond to messages about rentals
9. Insures property
10. Schedule when housecleaning goes to clean property
11. View complaints submitted by guests
12. View all reservation details
13. Automatically update availability when a rental is confirmed
14. Prohibit double booking

Non-Functional Requirements:

- Reliability
  - User interface needs to be accurate and up to date within 12 hours of any changes
- Performance – for example Response Time, Throughput, Utilization, Static Volumetric.
  - Search time for results should not take more than 1 second
  - Hovering to highlight/activate drop down menus should be instantaneous. Within half a second
  - General web load time should not take more than 3 seconds depending on internet connection for user
- Scalability
  - If functionality is expanded in the future such as adding the ability to book flights and ability for 3rd party renters to list, then the website/application should be able to be easily expanded to handle these
  - The website should be able to handle approx 10,000 to 15,000 users per day
- Availability
  - The website/application should be available everyday with weekly maintenance lasting no longer than 15-20 minutes on Sundays at 3 AM EST.
  - It should be compatible with various web browsers such as Internet Explorer, Google Chrome, Mozilla Firefox and mobile devices including Android and iOS.
- Security
  - In two step verification, user should receive text or email to verify within a minute.
  - Each time there is an unsuccessful login, it will be recorded. After 5 failed tries, the account will be locked and a notification will be sent to the user in case of a breach.
  - Should have ability to reset password in case user forgets password.
  - The website will have SSL encryption to protect the user.
- Regulatory
  - A privacy policy must be included to indicate what we will collect in terms of data.
  - The website will be accessible to anyone who needs it.
  - The website must have SSL encryption to protect the user.
  - The website will have a copyright.
- Manageability
  - The website/application should be organized and maintainable
- Data Integrity
  - Users of the product should be broken up into at least two fundamental groups, guest and staff. Staff should then be broken up into team managers group and staff group.
  - Guests should have no access to any of the other guests, and no access to any of the administrative information for the rentals.

    - Staff should have the personal information available to them restricted to the job specification that they have.
- Usability:
    - Both interfaces (user and admin) should be intuitive and easy to use with little to no explanation needed.
    - Admin should be able to quickly and efficiently make changes to the information that is viewable on the user interface.

System Requirements

- Types of Users:
    - General: Users who are not logged in are general viewers, and can still access the website. They can view the rentals and its description, its images and map, its availability, its reviews, its amenities, its policies, and its price, but they cannot rent a property, submit reviews, save rentals for future viewing, or chat with administrators about the property. If they try to do these things, they will be prompted to log in. General users can also filter out rental results by attributes like pet friendliess, kid friendliness, or wheelchair accessibility, requiring these attributes to be a part of each listing in the Rental Listings Database.
    - Logged-In Users (Customers): Customers can do all the things General users can do (listed above), but additionally can rent a property, can save a rental for future viewing, can submit a review (full review or just a rating) or complaint but only for a property they have previously rented, can chat with administrators about any property, can edit or cancel their property rentals, and can send in a maintenance request only for the property they are currently renting. When a customer rents out a property, a new entry is added to the Customer Rentals Database, which includes their Name, Phone Number, Address, Payment Information (passed forward to be handled by an external payment system), Property Rented, Days and Times Rented For, Number of Adult Guests, and Number of Children Guests, Once a Customer successfully rents a property, the availability of the property is updated in the Rental Listings Database. When a customer edits or cancels their rental, their corresponding entry in the Customer Rentals Database is updated, and the availability of the rental property is updated in the Rental Listings Database.
    - Administrators: Administrators can add new properties for rental and change the attributes of a rental property (described in detail in Rental Bookings below), can chat with Customer Users about various rentals, can reply to Customer User complaints about a rental property, and can schedule times for maintenance requests sent by Customer Users.
    - Both Customer Users and Administrator Users' account attributes will exist in the User Accounts Database. Both types of accounts will have an attribute indicating the type of account, either Administrator or Customer. Administrator Accounts will have attributes such as Name, Phone Number, Address, and Clearance/Rank, and the functionality described above. Customer Accounts will have attributes such as Name, Address, Phone Number, Payment Methods, Saved Rentals, Previous Rentals, Reviews and Ratings Submitted, and Complaints Sent and Addressed, and will have the functionality described above.
- Rental booking:
    - Administrators have the ability to create a new rental property listings. A listing has attributes such as the description of a listing (things like location and number of beds/baths), the images of a listing, the price of a listing, the availability of a listing, the

    - amenities of a listing, and the policies of a listing. This listing will be entered into the Rental Listings Database as a new entry.
    - Administrators have the ability to modify the attributes of a listing described above. Any modifications made will update the listing's corresponding existing entry in the Rental Listings Database.
    - Customer Users have the ability to rent properties. Once they are successfully rented, the rental details will be added to the Customer Rentals Database, which will include Rental Property Name, Rental Property Address, Rented By (Customer's Name), Rent Date and Time, and Number of Guests (Adult/Children). The Rental Property's entry in the Rental Listings Database will have its availability updated. If the Customer changes or cancels their rental, this will modify/remove the corresponding entry into the Customer Rentals Database, and will update the availability of the property in the Rental Listings Database.
- Creating Reviews:
    - Customer Users have the ability to submit reviews on rentals. They may only do this on properties they have previously rented, which are stored in the User Accounts Database. The reviews can be solely ratings (1 - 5 star ratings), or can be a rating with a written review. Once the review is submitted, it will be stored in the User Accounts Database, and will also be stored in the Rental Listings Database.
    - Customer Users can also submit complaints about rental properties. Once submitted, these complaints are stored in the rental property's corresponding entry in the Rental Listings Database, and can be viewed and addressed by Administrators.
- Databases and their Attributes:
    - Rental Listings:
        - Description: This Database is a list of rental properties. It can be modified by Administrators when they add, change, or remove rental properties in the system, and the Availability of a property changes when a customer rents a property, or cancels or edits their rental. Administrators can address complaints and maintenance requests stored here.
        - Attributes: Property Name, Price of Renting, Availability, Location, Number of Beds, Number of Baths, Images, Amenities, Nearby Locations, Weather, Policies, Reviews and Ratings, Complaints, Pet Friendliness, Kid Friendliness, Wheelchair Accessibility, Maintenance Requests
    - Customer Rentals:
        - Description: This database is a list of properties that have been rented out. A customer renting a property will add an entry to this database, and a customer changing their rental will change this entry. If a customer cancels their rental or they are no longer renting the property, the entry will be removed from the database after a year. This information will be viewable by both administrators and customers.
        - Attributes: Name of Renter, Phone Number of Renter, Address of Renter, Payment Information of Renter, Property Rented (an item from the Rental Listings database), Rent Date and Time, Rent Price, Number of Adult Guests, Number of Children Guests
    - User Accounts:
        - Description: The accounts of all users of the system are stored here.
        - General Attributes: Username, Encrypted Password, Email Address, Name, Home Address, Phone Number, Type of Account (Administrator or Customer), Messages Sent/Received

- Attributes Specific to Administrators: Clearance/Rank/Position, User and Data Traffic
- Attributes Specific to Customers: Previous Rentals (item from Customer Rentals database), Current Rental (item from Customer Rentals Database), Payment Methods, Saved Rentals, Reviews and Ratings Submitted (also stored in Rental Listings database), and Complaints Sent and Addressed

High level architecture design:

Design overview:

When it comes to designing a home rental service, we need to maintain a simple and user friendly aesthetic. Due to this, it will allow the users to easily access and use the service. On top of this, developers will be able to monitor and make fixes thanks to a simpler look. In terms of risks, we have to look at developers potentially leaving, the rapid advancement of technology, data breaches as well as a change in requirements. We consider our schedule to be achievable due to our in depth breakdown and plan for the project.

Architecture:

Our major systems in this project are rental booking and creating reviews for users. For the administrators, they have the ability to create and modify new listings as well as access the user's information. The servers will contain the data acquired through users and the various listings. We can also use a cloud-based storage.

Under the booking system, we will have systems to set a booking time, select a destination, and set a number of occupants. For the rating system, we have the ability to leave a rating, write a full review as well as the ability to ask questions about the rental. There will also be systems to view the property, amenities, policies and the total costs.

In terms of the administrative end, we will have the add a property system. Under that will be systems to change the prices, add photos or amenities included. Other subsystems include a messaging system to communicate with rental owners, viewing rental details and scheduling a maintenance for the platform.

User interfaces:

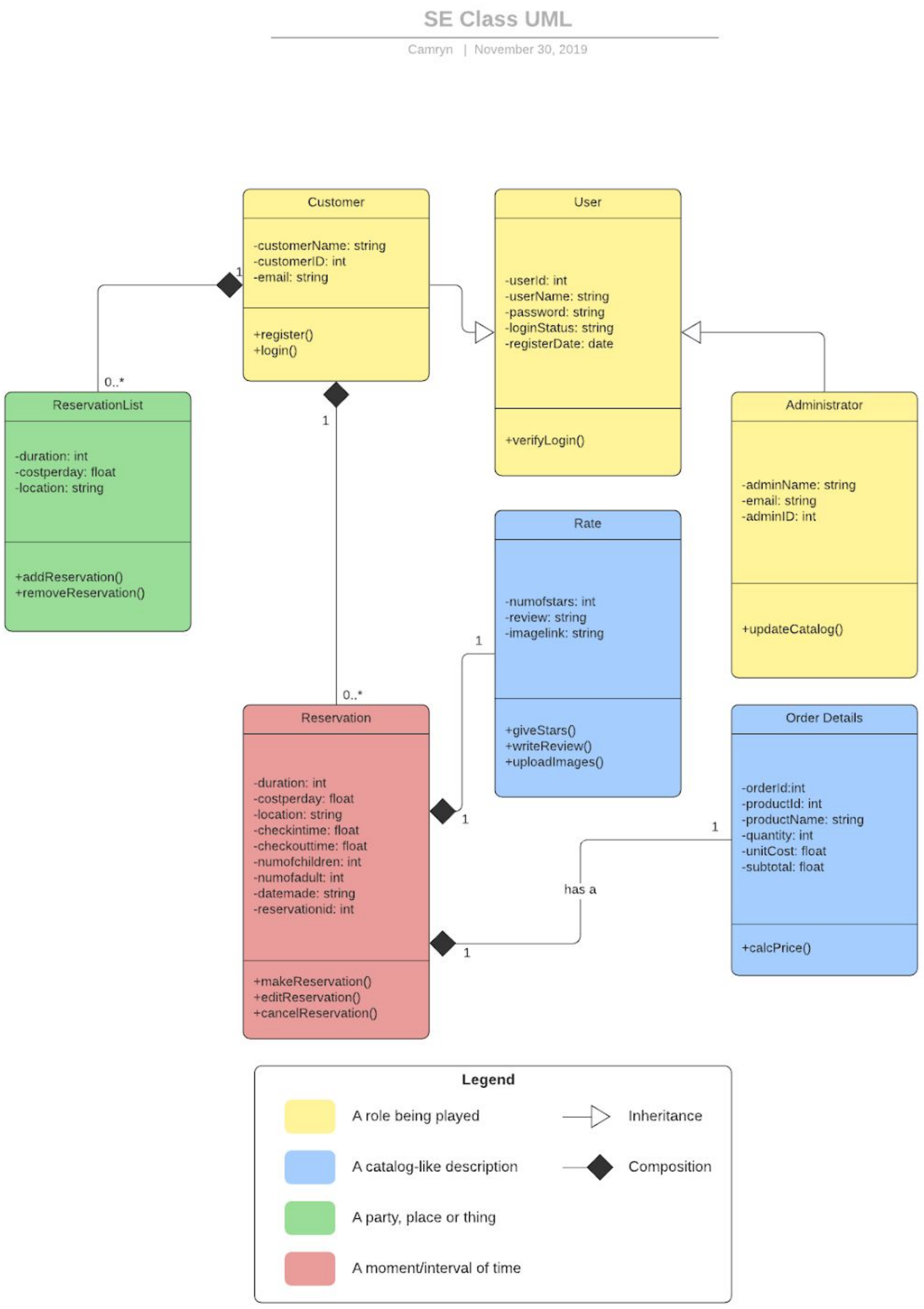
The menu on the site will feature multiple options for the user to access various parts of the website. We have the home tab which directs the user back to the home page, a book tab which allows the user to schedule a rental, an account tab where the user can login and manage their account information and a help tab which gives the user the ability to view frequently asked questions as well as reach out to the company for assistance if needed.

The home page features a simple view for booking or scheduling a rental property. It will have a few options for the user to get a quick and easy idea of how much it will cost. This will include the amount of time they want to rent it for as well as the number of occupants. Alongside this, there will be a slideshow of photos of various properties and the customers. On the book page, there will be an advanced view for the user to get in depth and will be able to see multiple properties after submitting some options as explained above. They can also access the reviews and feedback for different locations and properties. The account page contains the login information for the user and other personal information including addresses, credit cards, etc. Lastly, the help page contains a list of frequently asked questions as well as a live chat system.

Requirements:

We are expecting ten to fifteen thousand daily active users on mobile and three to five thousand through the website. Due to this, the servers will cost around $150 a month for a cloud based server with 200 GB of storage and 6 GB of RAM. This should be enough to maintain a site of this nature for a smaller set of users. As it grows, we will have to invest in larger servers. Over the course of the first year, we expect to grow at a rate from roughly twenty thousand overall users to at least thirty thousand users due to the competitiveness of the market.

UML Design


SE Class UML
Camryn | November 30, 2019
Customer
-customerName: string
-customerID: int
-email: string
+register()
+login()
User
-userId: int
-userName: string
-password: string
-loginStatus: string
-registerDate: date
+verifyLogin()
ReservationList
-duration: int
-costperday: float
-location: string
+addReservation()
+removeReservation()
Administrator
-adminName: string
-email: string
-adminID: int
+updateCatalog()
Rate
-numofstars: int
-review: string
-imagelink: string
+giveStars()
+writeReview()
+uploadImages()
Reservation
-duration: int
-costperday: float
-location: string
-checkintime: float
-checkouttime: float
-numofchildren: int
-numofadult: int
-datemade: string
-reservationid: int
+makeReservation()
+editReservation()
+cancelReservation()
Order Details
-orderId:int
-productId: int
-productName: string
-quantity: int
-unitCost: float
-subtotal: float
+calcPrice()
1
0..*
1
0..*
1
1
1
1
has a
Legend
A role being played
A catalog-like description
A party, place or thing
A moment/interval of time
Inheritance
Composition

Testing Plan:

1. **Introduction:**

   A home rental service through web and mobile based applications. Users can view various property listings and schedule rentals. They can also leave reviews and feedback.

2. **Objectives and Tasks:**

   Objectives:
   - Define the various tasks in the application
   - Test the different functions and features it make sure they are all working

   Tasks:
   - Ensure users can successfully schedule rentals
   - Feedback must work
   - Maintenance scheduling
   - Check to see if current storage capacity and RAM are sufficient
   - Images appear to user
   - Adding properties
   - Modifying prices
   - Ensure encryption is working

3. **Testing Strategy:**

   We will have QA teams for monitoring and stress testing the service. This will be done by having multiple users attempting to use this site through alpha testing. We can invite other people to use the site and have as many people as possible on at a set time to test if the service can handle heavy traffic.

   Maintenance testing will also be important for this kind of service. We will conduct maintenance at the regular scheduled time as if the service was already released as well as impromptu maintenances to test how long it will take.

   A major risk with this service is data breaches or attacks to the server causing a crash. We will prepare for this by doing tests where we intentionally crash the service and see how fast we will be able to fix the problem.

4. **Hardware Requirements:**
   - Computers
   - Servers
   - Smart home technologies
     - Ring doorbell
     - Cameras
     - A/C systems that can be accessed through smart phone
     - Alarm system that can be accessed through smart phone

5. Test Schedule:

- Testing begins in May of 2020 after development process
- Stress test servers over the course of alpha testing through June 2020
- Schedule impromptu maintenances during May 30, 2020 through June 15, 2020
- Test the functionality and features of the service during May 30, 2020 through June 15, 2020

6. Control Procedures:

Problem Reporting:

If a problem occurs, we will have a logging system for testers to log the time, data and an explanation of the problem. At the end of each day, we will collect the logs and determine a plan to solve the problems in a timely manner.

Change Requests:

If there is a change that needs to be made, the project manager will make the final decision. The team will first write a brief report explaining the change and the reason for it. IIf the manager gives it the green light, the teams can work to modify the program. After, it will be tested and then given to the clients to ensure it is done to their liking.

7. Roles and Responsibilities:

We will have a front end and back end team to create this product. Both teams will be working to test the project as well as a QA team. The front end team will work primarily to test the front end portions of the product whereas the back end team will focus on the back end and test the servers as well as maintenance. The QA team will stress test also.

8. Tools:

1. **Selenium** for browser testing
2. **TestingWhiz** for mobile app, database and API testing
3. **Test Complete** for GUI testing
4. **Ranorex** for bug testing
5. **Telerik Test Studio** for performance testing