**Dream Destination Getaways**

Software Requirements Specification Document

Project by:

Erika Valle-Baird

Madison Bilko

Tamara Greer

Mohammed Khoshkhoiyazdi

Domeniko Kodra

Collin Cavanaugh

Table of Contents

[1.0 Introduction 2](#_Toc106455062)

[1.1 Purpose of Document 2](#_Toc106455063)

[1.2 Project Summary 2](#_Toc106455064)

[1.3 Background 3](#_Toc106455065)

[1.4 Project Scope 3](#_Toc106455066)

[1.5 System Purpose 4](#_Toc106455067)

[1.5.1 Users 4](#_Toc106455068)

[1.5.2 Location 5](#_Toc106455069)

[1.5.3 Responsibilities 5](#_Toc106455070)

[1.5.4 Need 6](#_Toc106455071)

[1.6 Overview of Document 6](#_Toc106455072)

[2.0 Functional Objectives 7](#_Toc106455073)

[2.1 High Priority 7](#_Toc106455074)

[2.2 Medium Priority 8](#_Toc106455075)

[2.3 Low Priority 8](#_Toc106455076)

[3.0 Non-Functional Objectives 9](#_Toc106455077)

[3.1 Reliability 9](#_Toc106455078)

[3.2 Data Integrity 9](#_Toc106455079)

[3.3 Security 9](#_Toc106455080)

[3.4 Usability 10](#_Toc106455081)

[3.7 Interfaces 11](#_Toc106455082)

[4.0 Deployment Diagram 12](#_Toc106455083)

[5.0 The Context Model 13](#_Toc106455084)

[5.1 Goal Statement 13](#_Toc106455085)

[5.2 Context Diagram 14](#_Toc106455086)

[6.0 The Use Case Model 16](#_Toc106455087)

[6.1 System Use Case Diagram 16](#_Toc106455088)

[6.2 Use Case Descriptions 17](#_Toc106455089)

[6.2.1 Login 17](#_Toc106455090)

[6.2.2 Account Registration 17](#_Toc106455091)

[6.2.3 Flight Search 17](#_Toc106455092)

[6.2.4 Purchase 17](#_Toc106455093)

[6.2.5 Update Account Information 18](#_Toc106455094)

[6.2.6 Contact Us 18](#_Toc106455095)

[6.2.7 Help 18](#_Toc106455096)

[6.2.8 User Management 18](#_Toc106455097)

[6.2.9 Itinerary 18](#_Toc106455098)

[6.2.10 Account Logout 18](#_Toc106455099)

[6.3 Use Cases 19](#_Toc106455100)

[7.0 The Class Model 36](#_Toc106455101)

[8.0 Appendix and Glossary 38](#_Toc106455102)

# 1.0 Introduction

## 1.1 Purpose of Document

This is a Software Requirements Specification document for a new web-based open destination vacation-planning tool called Dream Destinations Getaways. Dream Destinations Getaways will allow users to purchase airline tickets at the lowest possible rate, along with supplying hotel and car reservations based on the user’s needs. Users must know when they are departing and arriving back from their vacation but can be indifferent about the destination. This system will allow users to register for an account, login to their account, search for tickets, purchase tickets, update their account information, contact us, get help, and logout. Laid out in this document are descriptions of the non-functional requirements including scope, objectives, and goal of this vacation-planning tool. In addition to this, functional requirements are presented via modeling with a Use-Case diagram, followed by specifications for each use case. To present the system’s interactions with external entities, a context diagram depicts the interactions between the two. To visualize both the physical hardware and software of Dream Destinations Getaways, a deployment diagram is included. The final diagram of this document consists of a class model, which depicts the static structure of objects within the system and then defines the entire system. Using an object-oriented language, this specified document aids in advising the design and implementation of the aforementioned system.

## 1.2 Project Summary

**Project Name:** Dream Destinations Getaways

**Project Manager: Erika Valle-Baird, Madison Bilko**

**Project Analysts:** Collin Cavanaugh

Tamara Greer

Mohammed Khoshkhoiyazdi

Domeniko Kodra

## 1.3 Background

Dream Destination Getaways sells flight tickets for different purposes, vacations at the lowest possible rates. The purpose of this app is to make a one stop shop to show the user the cheapest rates for all their needs. Its customers include people looking to find a vacation destination with specific dates in mind but can be indifferent to the location of travel. Dream Destination Getaways will give all the users the opportunity to find the lowest price of the flight's tickets, the lowest price of the hotel and car rental. Web based applications are very useful nowadays. Every one of us gets most of the service that we need from the web. This application is going to give the opportunity to the users to find the cheapest prices of flight tickets, hotels and car rental on the market on one app.

## 1.4 Project Scope

The scope of this project is a web-based system capable of allowing customers to directly access and purchase travel at discounted rates. This easy-to-use application will use a database to synchronize data and store user information. Account holders of this system will have access to searching for low-fare airline flights, the option to add hotel reservation, car rentals, and discounted packages containing both. Customers do not have to know where they are going to receive flight information, although they can input city, country, or continent if they so desire.

## 1.5 System Purpose

### 1.5.1 Users

Customers:

* At the time of completion of the system, customers will be able to
* Find the tutorial for site navigation
* Create/login/logout/update their account
* Search for flights based on departure and arrival time
* Search for flights based on optional destinations
* Purchase flights
* Add a return flight, car rental, hotel, or package to their reservation
* Request help from an administrator

Administration:

* After completion of this system, the administration is able to
* View website traffic
* View purchases made
* View and respond to help requests
* Review and edit user accounts
* Cancel reservations
* Lock/unlock accounts based on suspicious activity
* Delete a user’s account

### 1.5.2 Location

This website will be available to all customers who have access to the internet. Preferably, the user will only use the website on laptops or large tablets due to size constraints with smaller tablets and phones.

For those working on developing the website, they will be required to have secure access to the website due to sensitive information they may access.

### 1.5.3 Responsibilities

The primary responsibilities of the system:

* Generate lowest possible airfare tickets given departure and arrival dates
* Allow for the purchase of airline tickets, car rental, and hotel rooms
* Create an account with specific required information
* Generate various package deals that include hotel and car rentals cheaper than individual purchases
* Log all user activities and allow management to analyze user behavior
* Synchronize flight, hotel, and car rental information from a variety of travel websites
* Connect to external credit institution to process credit card payment

Other desired features of the system:

* Timely page response
* User interface should be attractive and easy to navigate
* Allow the user to sort their results

### 1.5.4 Need

This website is needed to service customers all around the world to purchase flight tickets. This system allows the customer to have access to all the flights available based on the customer’s date. This website is indifferent to a user’s age, race, or language the service will be provided in and allow them to purchase their flights.

## 1.6 Overview of Document

**Section 2: Functional Objectives**

To create a working system, each objective is defined with criteria given for expectations of the system, basic functionalities, input to the system, operation performed, and expected output. Priorities for such objectives are given, with high priority being the most pertinent objectives to fulfill, while low priority are the least significant objectives for the system.

**Section 3: Non-Functional Objectives**

Measurable categories that describe quality constraints imperative to the system. These objectives are non-behavioral requirements made by the project.

**Section 4: Deployment Diagram**

A graphical representation to depict the hardware and software execution environment. This presents the execution architecture of the system and illustrates its deployment.

**Section 5: The Context Model**

A graphical representation to illustrate how external entities interact with the system. This is a high-level view of the system, depicting only a broad generalize image and not internal details.

**Section 6: The Use Case Model**

All potential interfaces of a user are encapsulated here. This is presented as both a graphical model and comprehensive explanations of each use case. Included are the different types of users and their specific interactions within the system.

**Section 7: The Class Model**

A diagram depicting the different classes, attributes, operations, and relationships of the system. This illustrates a static view of the vacation planning tool application.

**Section 8: Appendix**

The appendix will include a glossary for words that are frequently used in the document. The appendix will also include any outside resources that was used in the creation of this document.

# 2.0 Functional Objectives

## 2.1 High Priority

* The system shall allow for the purchase of a plane ticket, hotel reservation, car rental, or a package of all three.
* Customers should be able to create a unique username and password.
* Customers should be able to log in to the system with the right login information.
* The system should be able to recognize the correct set of login information.
* Customers should be able to log out of the system.
* Customers should be able to contact administration for assistance.
* The system shall provide customers with data security.
* The system should provide secure payment method.

## 2.2 Medium Priority

* The website provides a search facility that will allow the customer to input the dates for the flight and a destination, which can be left open if they are unsure. It allows the user to choose how many passengers are traveling. The website will also include the following:
* Departure From: the user inputs the location they will be leaving from
* Destination: enters the destination, either through the destination’s country or continent, or completely left open
* Dates: user chooses dates of departure and arrival of the flights
* Passenger: allows user to choose number of passengers traveling
* The system provides help and instructions, allowing customers to better navigate the website and how to purchase a ticket.

## 2.3 Low Priority

The system shall allow items in the user’s cart to be stored for 2 hours before clearing them.

The system shall allow the user to put the price range for the given destination or given trip and notify the user the given price range is satisfied.

The system shall translate web pages for all countries that access and use it.

The system shall give different ways the user can sort and view the information given to them

# 3.0 Non-Functional Objectives

## 3.1 Reliability

* System will provide results within a reasonable amount of time.
* System will not crash when site traffic is high.
* System will process payment within a reasonable amount of time.
* System will both, smoothly and promptly, perform functions such as registering for an account, logging into an account, searching for tickets, purchasing tickets, and updating account information.
* System will keep data information secure

## 3.2 Data Integrity

* The system will compress and decompress any images on the website to fit the user’s device.
* The system will flag and lock user accounts when any suspicious activity is noted.
* The system will be backed up by admin every eight hours to ensure synchronization, accurate, and reliable information.

The system will not allow admin to tamper with reports generated about user activity.

## 3.3 Security

* The system shall provide high security to protect online transactions made by customers.
* The system will keep data information gathered from customers both safe and protected.
* The system will keep data information gathered from administration both safe and protected.
* The system will keep information gathered from financial institutions safe and protected.
* The system should block access to user and administration who fail login attempts after three failed attempts.
* The system should lock accounts of user and administration if suspicious activity is noted.
* The system should authenticate any user attempting to access it.
* The system should be accessed only by a registered account username and password.

## 3.4 Usability

* Customers who have access to an Internet connection can use this website.
* The button’s size and font are designed in a way that makes the website usable for different user demographics.
* Site will be organized in a way that makes it easy to navigate and understand.

3.5 Performance

* System will give accurate, updated rates and flights.
* System will accurately display hotel and car reservations based on the user’s needs.
* System will display accurate pricing.

3.6 Online User Documentation and Help

The account access shall be customized to the type of user allowances.

The site contains documentation to each specific functionality to allow users to get informational help via “Help” links that are easily accessible.

The system shall maintain documentation of user activities to be viewed by admin.

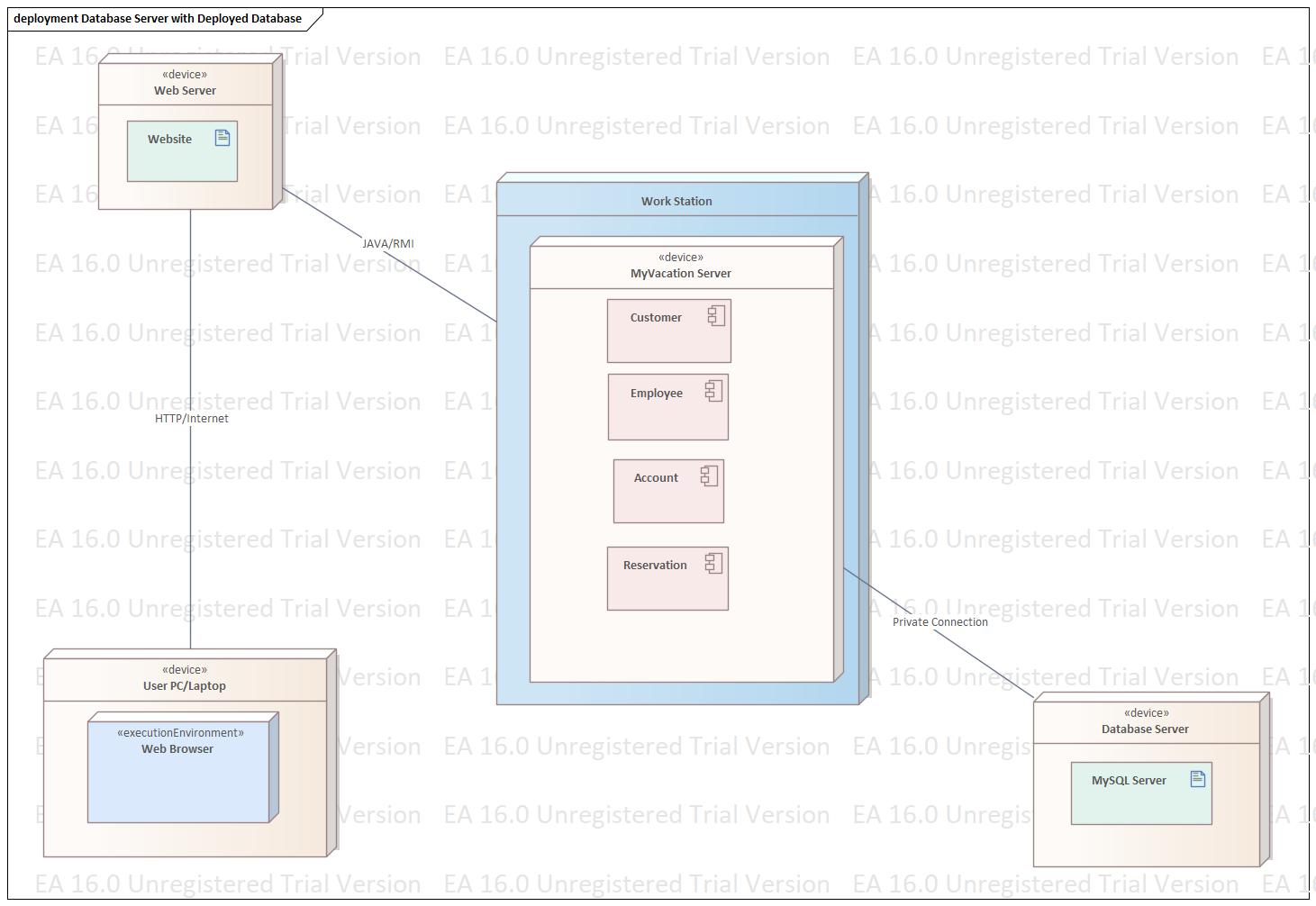
3.7 Supportability

* The system website shall be viewable from internet explorer 4.0 or later, Google Chrome, and FireFox.
* The website should be able to update the new flights and data without any major issues.

## 3.7 Interfaces

* GUI will be user-friendly, organized, and easy to navigate, with a constant and precise format.
* HTML/CSS/JavaScript will be used to achieve optimal site.

# 4.0 Deployment Diagram



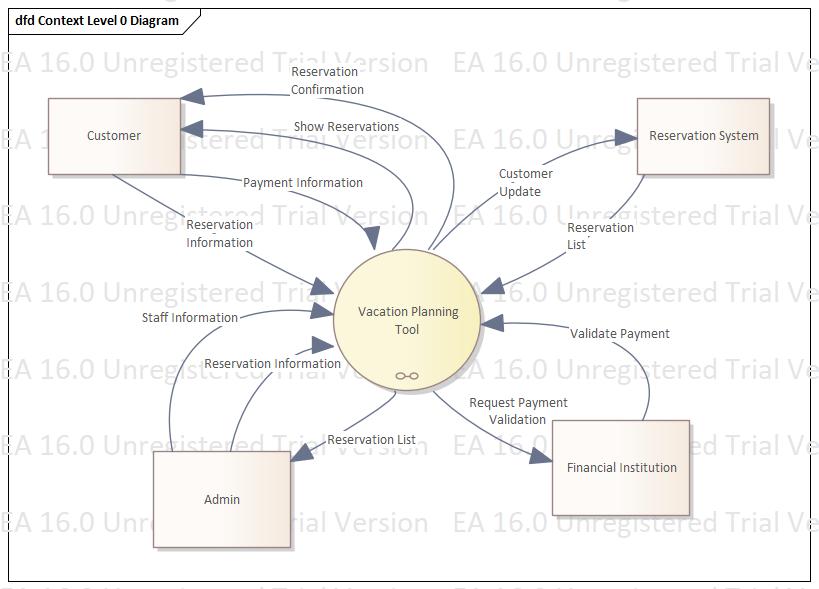
To illustrate how software will interact with the hardware in the system, an illustration of the static system is depicted above, using a deployment diagram. It begins with the user deploying their web browser on their personal PC/laptop. This web browser is an execution environment from which software executes on, executing on the user’s PC/laptop, a physical hardware device. This environment is dependent upon the user due to varying information for each user. To make a reservation, their device must first connect to a web server, followed by the Dream Destination Getaway server. An internet/HTTP connection allows this personal device to connect to the web server. This server is identical to the user’s personal device, a physical hardware device. Deployed inside that device is the Dream Destinations Getaway website. The web server connects to the website’s workstation via JAVA/RMI. This station houses the Dream Destination Getaway server, a physical hardware device that houses various components that make up the system. A customer, employee, account, and reservation encompass the components of the system. To obtain flight, hotel, car rental, and package information, the workstation connects to a database server via a private secure connection.

# 5.0 The Context Model

## 5.1 Goal Statement

The goal of the context model is to show how external entities will interact with the website. Depicting the Vacation Planning Tool as a singular entity, this model demonstrates the relationships external entities have with the system. None of the inner workings within the site are displayed here, allowing those unfamiliar with computer programming to understand how the Vacation Planning Tool interacts with other entities and systems.

## 5.2 Context Diagram



**System Externals**

**Customer**

A customer is any user of the system who is not an employee and who does not have an admin account. A customer may search for flights, car rentals, hotels, and packages via reservation. A customer may submit payment information to complete a reservation order. A customer who provides departure and return dates in a flight query will receive the results of such a query. A customer who provides valid payment information will receive a reservation confirmation upon completion.

**Admin**

An admin is any user of the system who is an employee and who holds an admin account. An admin is informed about reservation information. The admin may update employee data and reservations.

**Reservation System**

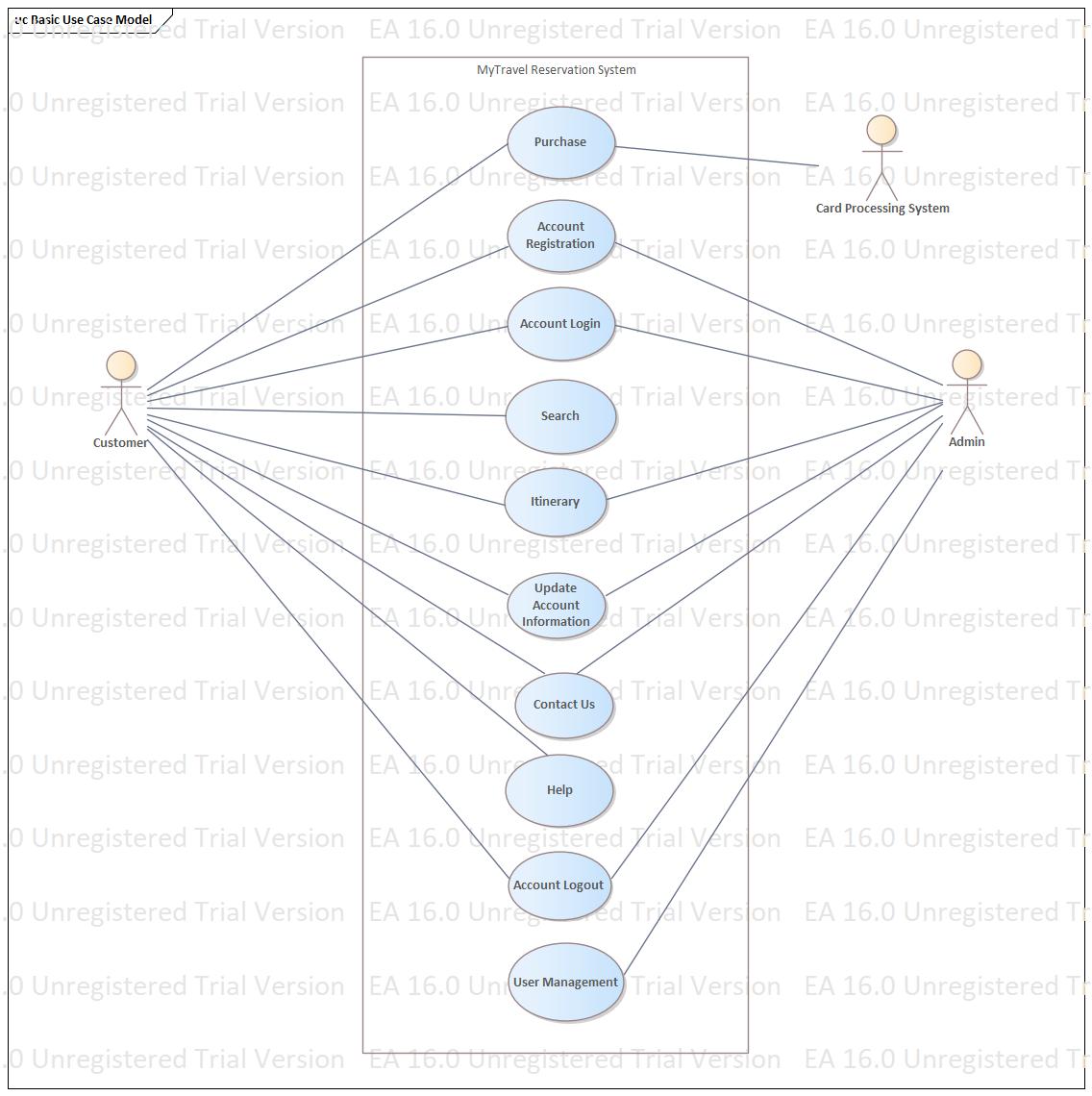
The reservation system is a database that contains customer and reservation information. The reservation system is informed about customer updates so that it can provide reservation lists specific to the customer’s need.

**Financial Institution**

An institution responsible for all financial transactions acting as a sales channel. The financial institution is informed about any payment requests and returns responses to those requests.

# 6.0 The Use Case Model

## 6.1 System Use Case Diagram



## 6.2 Use Case Descriptions

Functionalities provided by the system to users and administration are described below.

### 

### 6.2.1 Login

Description: This function allows users to login to the system. If user is not yet registered, the website shall allow the user the ability to enroll. Both the username and password will be examined to confirm user identity.

### 6.2.2 Account Registration

Description: This function allows users to register for an account with the system. The system will check the username given by the user is not a duplicate name in the current systems database. The system will require necessary user data to create an account including the user’s name, address, credit card information, and billing address.

### 6.2.3 Flight Search

Description: This function allows a customer to search for available flights given a departure date, return date, and departure location. The arrival location of the destination does not have to be specified. The system shall display current flights, departure time, arrival time, and cost of each flight on the specified dates. The system shall cross over to the shopping cart to aid the user in their purchase.

### 6.2.4 Purchase

Description: This function allows customers to purchase airline tickets, hotel reservations, car rental reservations, discounted packages, and processes payments. If a purchase cannot be validated, a request for new credit card information will be displayed.

### 6.2.5 Update Account Information

Description: This function allows a user to edit account information such as name, address, and card information.

### 6.2.6 Contact Us

Description: This function allows a user to contact staff if any questions or concerns about their reservations arise.

### 6.2.7 Help

Description: This function provides customers with information pertaining to how to use each functionality in the website. This includes how to reserve air travel, a hotel room, a car rental, and information about discounted packages we offer.

### 6.2.8 User Management

Description: This function provides admin with the ability to lock a user’s account if suspicious behavior arises. This includes deleting accounts if policies are not followed.

### 6.2.9 Itinerary

Description: This function provides the customer with reservation details, specifies the number of passengers, displays each passenger’s details, any reservation updates, and allows the user to submit cancellation requests.

### 6.2.10 Account Logout

Description: This function allows users to log out of the system.

## 6.3 Use Cases

Notes:

The main actors of the system include:

Admin

These actors are employees responsible for locking accounts when a potential security threat is suspected, deleting accounts if users neglect policy and procedures. Admin can also analyze user behavior, edit reservations, and are alerted when a customer signifies, they need help via the contact us function.

Customer

Person who uses the web-based application to search, make, cancel, and update flight, car rentals, and hotel reservations.

Card Processing System

This actor is a system that validates, and processes payments made by customers of the system.

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | **User Login** | |
| **Actors:** | Customer, Admin | |
| **Summary:** | Customer | Admin |
| In order for a customer to view lowest price ticket information, purchase tickets, edit account information, or view their reservations, a user must login. | An employee must login to the system to edit account information, generate reports, or perform user management. |
| **Entry Conditions:** | * The user is registered and has a username and password. * The user’s account is not locked or admin must be contacted for user verification. | |
| **Flow of Events:** | 1. The use case starts when a user indicates that he wants to login. 2. The system requests the username and password. 3. The user enters his username and password. 4. The system verifies the username and password against all registered users. 5. The system starts a login session and displays a welcome message based on the user's preferences. | |
| **Alternative Flow:** | 4.a:  if username is invalid, the use case goes back to step 2.  4.b:  if the password is invalid the system requests that the user re-enter the password. When the user enters another password the use case continues with step 4 using the original username and new password.  4.c:  If three incorrect inputs for password is signed, user’s account is locked until account and behavior are verified, signaling an admin to release it. | |
| **Exit Conditions:** | * The user can now obtain flight data and perform functions. * The user remains logged in until the “Logout” button is specified. | |
| **Special Requirements:** | * Depart from location, departure date, and return date are required to get flight data. * Cancel button is pressed returning the user to the main screen without being logged in. * The user is registered. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | **Account Registration** | |
| **Actors:** | Customer, Admin | |
| **Summary:** | Customer | Admin |
| In the interest of obtaining flight tickets, a new user must register a username and password. | An employee of the system must be allowed access to generate reports, analyze user behavior, perform user management, view and edit customer information, and assist customers. |
| **Entry Conditions:** | The user does not have an account. | |
| **Flow of Events:** | Customer | Admin |
| 1. The use case starts when a user indicates that he or she wants to register. 2. The system requests a username and password. 3. The user enters a username and password. 4. The system checks that the username would not be a duplicate of any existing registered usernames. 5. The system requests a name, mailing address, billing address, credit card type, credit card number, and expiration date. All of these items are required for registration. 6. The system starts a login session and displays a welcome message based on username. | * + - 1. The use case starts when a user indicates that he or she wants to register.       2. The system requests a username and password.       3. The user enters a username and password.       4. The system checks that the username would not be a duplicate of any existing registered usernames.       5. The system requests a name, mailing address, email, date of birth, phone number, and tax information.       6. The system starts a login session and displays a welcome message based on username. |
| **Exit Conditions:** | Customer | Admin |
| The user is now registered and can access the system.  The user can now obtain flight ticket information and update their account information. | The user can now obtain flight ticket information and update their account information.  The user is now registered and can access the system. |
| **Special Requirements:** | The user must not be banned from the system.  The user is not already a registered user of the system. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **Flight Search** |
| **Actors:** | Customer |
| **Summary:** | This use case allows a registered customer to search for available flights. |
| **Entry Conditions:** | * The user is registered. * The user is logged in to their account. |
| **Flow of Events:** | 1. The user navigates to the webpage and selects the button “Search” that allows them to search flights. 2. The user inputs departure and arrival dates, along with destination for their vacation and any additional passengers and their ages. 3. A list of flights that aligns with the user’s input is then displayed to the user to browse through. |
| **Alternative Flows:** | 2a. The user does not enter a destination location, only departure and arrival dates.  2b.There are no flights matching user’s specifications, returns to step 1.  3a. A list of flights that depart and arrive within the time, as designated by the user, is displayed for the user to browse and choose from. |
| **Exit Conditions:** | * Lowest fare flights are displayed to screen. * The user has the option to add a specific flight to their cart. |
| **Special Requirements:** | * Depart and return locations are spelled correctly. * The user is registered. * Both departure and return date are in the future. * Return date is after departure date. |

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | **Purchase** | |
| **Actors:** | Customer, Card Processing System | |
| **Summary:** | Customer | Card Processing System |
| This use case allows a registered customer to purchase tickets for a flight, along with hotel reservations and car rentals. | A third-party processing system validates and processes transactions. |
| **Entry Conditions:** | * The user has logged in. * The customer has completed a search for a flight. * The customer has flight and additional reservations in their shopping cart. * The ticket has not yet been purchased. * The customer has a valid form of payment. | |
| **Flow of Events:** | Customer | Card Processing System |
| 1. This use case starts when a customer indicates he wants to place an order for the current reservation being displayed. 2. The system displays the customer's information: name, street, city, zip, phone, email for confirmation. 3. The customer may add or change any of the information. 4. The system stores any changes. If the zip code has changed, the system modifies the customer's location. 5. The system displays a screen with input fields for information about any additional passengers including their name, date of birth, and address. 6. The system provides option for hotel reservation, car rental, and package options that are discounted when purchasing both together. 7. The customer adds or declines any additional options. 8. The system displays itinerary information consisting of flights, passengers, hotel reservation, car rental, and package information. 9. The customer selects purchase option. 10. The system displays option for credit card or bank checking account payment. 11. The user selects which type of payment they are going to use. 12. The system displays a screen with input fields for payment information based on the user’s choice. 13. Customer selects submit button sending the information to the card processing system. 14. Payment is successful and reservation is now made. | 1. This use case starts when the third- party processing system receives a request for payment validation from the Dream Destination Getaways server. 2. The card processing system validates the payment. 3. The card processing system returns a message to the system based on the result of the payment. |
| **Alternative Flow:** | Customer | Card Processing System |
| 14a. If the selected payment method could not be validated, go to step 10 to get another payment option.  14b. The purchase did not successfully complete due to discrepancy in account holder information. The customer goes back to step 10 and is asked for another form of payment. After 3 attempts the customer’s account is locked and flagged for suspicious activity  14c. Payment type is fraudulent, account is immediately locked and flagged for admin to investigate.  14d. The purchase did not successfully complete due to lack of funds, customer is asked for another form of payment. | 2a. Selected Payment could not be validated by the card processing system.  2b. The payment could not be validated due to discrepancy in account holder information.  3a. The method of payment returned a flag due to suspicious activity.  3b. The account holder does not have the necessary funds to complete the transaction.  3c. The payment could not complete due to network connection errors. |
| **Exit Conditions:** | * The reservation is successfully made. * Reservation details now available to customer in itinerary. * Reservation is now available to admin in user management. | |
| **Special Requirements:** | * Cancel button is pressed returning the customer to the main screen without purchase. * Timed out of session, not completing purchase. * The credit card used cannot be reported as stolen. * The customer must have the funds in their account to complete purchase. * The user is registered. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **Update Account Information** |
| **Actors:** | **Customer, Admin** |
| **Summary:** | A user can edit their account information**.** |
| **Entry Conditions:** | * The user has an existing account. * The user is logged in to the system. |
| **Flow of Events:** | * + - 1. The “Edit Account” button has been selected by the user.       2. The system displays current account information, allowing for input from the user.       3. The user selects “Done”, submitting the changes. |
| **Alternative Flow:** | 3a. The input by the user included a character that was not recognized as allowable input for that given field, returns user to step 2. |
| **Exit Conditions:** | The account has updated information per the user input. |
| **Special Requirements:** | * The user is registered. |

|  |  |
| --- | --- |
| **Use Case Name:** | **Contact Us** |
| **Actors:** | **Customer** |
| **Summary:** | The user desires to contact the business with questions about their reservation, concerns, or complaints. |
| **Entry Conditions:** | * The user has a registered account. * The user is logged in to the system. |
| **Flow of Events:** | 1. The user has an issue with the website that the “Help” site cannot assist them with. 2. The user navigates to the “Contact Us” page. 3. A list of ways to contact administrator appears for the user to use. |
| **Exit Conditions:** | * The user is able to reach out to the company with their question, concern, or complaint. |
| **Special Requirements:** | * The various options to contact company are continuously up to date. * The user is registered. |

**Help**

|  |  |
| --- | --- |
| **Use Case Name:** | **Help** |
| **Actors:** | **Customer** |
| **Summary:** | Guidelines are presented to the customer describing the functionality and how to use each of the functions. Information pertaining to reservations are also presented. |
| **Entry Conditions:** | None |
| **Flow of Events:** | * The “Help” button has been selected. * Headers for each function in the website along with headersfor each type of reservation is displayed to the user’s screen. * The user can select which function or topic they desire. |
| **Exit Conditions:** | * The user understands how to use each of the functions on the website. * The user understands how to reserve and cancel their reservations. |
| **Special Requirements:** | None. |

|  |  |
| --- | --- |
| **Use Case Name:** | **User Management** |
| **Actors:** | **Admin** |
| **Summary:** | This use case allows admin to edit customer information, access customer information and reports, lock and unlock accounts due to suspicious behavior, and allows admin to delete a user account if any polices are violated. |
| **Entry Conditions:** | * The user is logged in. * The user’s account is an admin account. |
| **Flow of Events:** | 1. The use case starts when an admin selects the “User Management” button. 2. Admin is alerted to any requests for reservation cancellations. 3. Admin issues a cancellation request and refunds the customer. 4. Admin is alerted that suspicious behavior has been encountered. 5. Admin selects generate report button to analyze user behavior. |
| **Alternative Flow:** | 3a. If a customer submits the request at least 72 hours before their scheduled reservation, admin can allow their request and return their funds.  3b. If a customer submits their request less than 72 hours before their scheduled reservation, admin can deny their request and keep their payment.  4a. If a customer was locked out of their account due to too many username or password attempts, admin completes a user verification. If the user can be verified, the account is unlocked.  4b. If a customer is locked out due to suspicious fraudulent account activity, a warning is placed on their account and cannot receive another warning without their account being deleted by admin.  4c. If a customer violates website or company policy, their account is deleted in the server database by admin. |
| **Exit Conditions:** | * Alerts have been resolved. * Any requested reports have been sent to printing device or saved remotely. |
| **Special Requirements:** | * The user is registered. * Cancel button is pressed returning the admin to the main screen without saving editing of user accounts. |

|  |  |
| --- | --- |
| **Use Case Name:** | **Itinerary** |
| **Actors:** | **Customer** |
| **Summary:** | This use case allows a customer to view and cancel their flight, hotel, car rental, and package information, along with the ability to view previous transactions. |
| **Entry Conditions:** | * The user is logged in. * The customer has a reservation or has had a previous reservation in the past. |
| **Flow of Events:** | 1. Selects the “Itinerary” button. 2. System displays most recent reservation information including flight, hotel, car rentals, and package details. 3. System displays a screen with buttons “Cancel a Reservation”, “View Previous Transactions”, and “Get Help”. |
| **Exit Conditions:** | * Customer indicates they are finished by logging out or by selecting the “Main Menu” or “Cancel” buttons. |
| **Special Requirements:** | * The user is registered. * Cancel button is pressed returning the customer to the main menu screen without saving editing of account or cancellation requests. |

|  |  |
| --- | --- |
| **Use Case Name:** | **Account Logout** |
| **Actors:** | **Customer, Admin** |
| **Summary:** | The user is able to log out of their account. |
| **Entry Conditions:** | The user is logged in to their account. |
| **Flow of Events:** | **The user selects the “Logout” button on their menu.** |
| **Exit Conditions:** | The user is no longer logged into their account. |
| **Special Requirements:** | The user is registered. |

# 7.0 The Class Model

The main classes of the Vacation Planning Tool include:

* **Admin**: An employee that holds an admin account. This type of account houses pertinent data and allows them to manipulate the system and access data.
* **AddOns**: This encompasses hotel, car rentals, and package data.
* **CarRental**: Each car rental will include arrival date, departure date, cost, car type, and contain a reservation id specific to its rental.
* **Customer**: A customer holds an account that houses pertinent data and allows a customer to access the system for reservations and information.
* **Flight**: Each flight will have arrival and departure dates, arrival and departure locations, and a flight number.
* **Hotel**: Each hotel will include arrival date, departure date, cost, type of hotel, star rating, and contain a reservation id specific to its reservation.
* **Itinerary**: Each itinerary includes arrival date, departure date, departure location, arrival location, group size, and an id number for that specific instance.
* **Package**: Each package will include arrival date, departure date, cost, and a reservation id specific to its reservation.
* **Passenger**: Each passenger on the itinerary has their own date of birth, name, and passenger id number.
* **Payment**: Each payment received for a reservation includes the payment type.
* **Reservation**: Each reservation includes the total cost, customer id number, reservation date, and unique reservation id number.

# 8.0 Appendix and Glossary

**Account**: A single account in the system. There are two types, a customer account and an admin account. No user can have multiple accounts.

**Itinerary**: A record of any future and past reservations made by the customer.

**Package**: When purchasing flight tickets, a user is able to add amenities such as a hotel and car rental, to the flight ticket. This trio of items, a package, will be sold for a cheaper price compared to individual purchase.

**Reservation**: A placeholder for the user at a hotel or at car rental. It ensures that a room or car will be available for the user during their vacation.

**User**: A person who will be interacting with the website. This may be used interchangeably with customer.