BUSINESS REQUIREMENT DOCUMENT

MEet and You

Brent Nishioka (Leader)
Gideon Essel
Joshua Ramos
Raymond Guevara
Vivian Dinh
Team Pentaskilled
September 28th, 2021

Table of Contents

1. Introduction
- 1.1. Objectives
- 1.2. Background
- 1.3. Project Scope
  - 1.3.1. In Scope
  - 1.3.2. Out of Scope

2. Core Feature Requirements
- 2.1. Data Store Access
  - 2.1.1. Functional
  - 2.1.2. Non-Functional
  - 2.1.3. Pass/Fail Requirements
- 2.2. Documentation
  - 2.2.1. Functional
  - 2.2.2. Non-Functional
  - 2.2.3. Pass/Fail Requirements
- 2.3. Error Handling
  - 2.3.1. Functional
  - 2.3.2. Non-Functional
  - 2.3.3. Pass/Fail Requirements
- 2.4. Logging / Archiving
  - 2.4.1. Functional
  - 2.4.2. Non-Functional
  - 2.4.3. Pass/Fail Requirements
- 2.5. Login / Logout
  - 2.5.1. Functional
  - 2.5.2. Non-Functional
  - 2.5.3. Pass/Fail Requirements
- 2.6. Network Communication
  - 2.6.1. Functional
  - 2.6.2. Non-Functional
  - 2.6.3. Pass/Fail Requirements
- 2.7. Registration
  - 2.7.1. Functional
  - 2.7.2. Non-Functional
  - 2.7.3. Pass/Fail Requirements
- 2.8. Usage Analysis Dashboard
  - 2.8.1. Functional
  - 2.8.2. Non-Functional
  - 2.8.3. Pass/Fail Requirements

2.9. User Access Control
2.9.1. Functional
2.9.2. Non-Functional
2.9.3. Pass/Fail Requirements
2.10. User Management
2.10.1. Functional
2.10.2. Non-Functional
2.10.3. Pass/Fail Requirements

3. App-specific Feature Requirements
3.1. Duplicating Itineraries
3.1.1. Functional
3.1.2. Non-Functional
3.2. Hyperlink Sharing
3.2.1. Functional
3.2.2. Non-Functional
3.3. Localized Weather Forecast
3.3.1. Functional
3.3.2. Non-Functional
3.4. Memory Album
3.4.1. Functional
3.4.2. Non-Functional
3.5. Rating System
3.5.1. Functional
3.5.2. Non-Functional
3.6. Suggestions
3.6.1. Functional
3.6.2. Non-Functional
3.7. Text Notification
3.7.1. Functional
3.7.2. Non-Functional
3.8. User Profile Dashboard
3.8.1. Functional
3.8.2. Non-Functional

4. Glossary

1. Introduction

1.1. Objectives

- The goal is to create a web app that gives users event recommendations in a city. These recommendations will allow users to create an itinerary.
- Users can register with our web app, allowing the user to save and share itineraries.
- Users can browse through their itineraries with information such as itinerary rating and event cost, if applicable.
- Support search functions to display event details for an itinerary.

1.2. Background

Initially, our application allowed the user to have a shareable calendar that displayed our itinerary. This did not add value to our product and we have since updated our system to include features that help users determine which events to choose. These features include the weather of the day, the price associated with an event, and user ratings.

1.3. Project Scope

1.3.1. In Scope

- Create a web application for people who live in the United States and who also understand American English.
- Allow users to create an account with the site in order to access more privileges (e.g., saving, sharing, duplicating itineraries)
- The web application will be compatible with Chrome v94.0.4606.61.
- Allow users to be redirected to an event details page to reselect or reschedule an event.
- A succinct and holistic overview of an event (e.g., name, description, and address) will be shown to the user on the event details page.
- Allow an itinerary to obtain event cost details to help the user select events within their budget.

- Allow an itinerary to obtain weather details to help the user select weather-appropriate events and clothing.
- A dashboard centralizing information related to the user's account such as profile picture and account settings.
- Create an all itinerary section to show every itinerary that the user owns or has been shared with.
- Allow users to access a subset of itineraries that have been favorited and recently accessed.
- Allow registered accounts to save, edit, and share itineraries.
- Allow registered accounts to share itineraries via a hyperlink.
- Allow registered accounts to rate and review an itinerary or event.
- Allow users to upload images to the web server for an itinerary.
- Registered users also have the ability to duplicate itineraries.

1.3.2. Out of Scope

- Multiple language support.
- Multiple browsers support.
- Support for countries that are not the U.S.

2. Core Feature Requirements

2.1. Data Store Access

2.1.1. Functional

- Create
  Description**:** Support the ability to create and store application-specific user data.

  Business Rules:
  - Duplicate accounts are not allowed (e.g., the user can only have one account associated with a specific email).
  - The system will not add new data if the storage capacity has been reached (5 GB).

- Delete
  Description**:** Support the ability to delete existing data (itineraries, accounts, etc.) from the database.

  Business Rules:
  - The user will be notified if requested data to delete does not exist in the database.

- Update
  Description**:** Support the ability to modify existing users data (e.g., change username and password).

  Business Rules**:**
  - Changes to the record must follow the uniqueness constraints (not changing username to already existing username).

- Read
  Description**:** Support the ability to read existing data from the database.

  Business Rules**:**
  - The user will be notified if the requested data to read does not exist in the database.

2.1.2. Non-Functional

- The time it takes to execute an action for data store access (create, delete, update, or read) is less than 4 seconds.

2.1.3. Pass/Fail Requirements

Pass

- Database objects will be created if uniqueness constraints are met (e.g., the username is unique and not present in the database).
- Database objects will be retrieved if the key to identifying the object is present in the server.
- Database objects will be deleted if the key to identifying the object is present in the server.
- Database objects will be updated if the key to identifying the object is present in the server.

Fail

- If the storage capacity has been reached then the create function will fail.
- If requested data does not exist then the delete, update, read functions will fail.
- An unregistered account is able to execute an action for datastore access.

2.2. Documentation

2.2.1. Functional

- Developer Docs
  Description**:** Contains technical information relevant to software developers (e.g. code functions, use cases, etc.).

  Business Rules**:**
  - The document is made available to all users.
  - Sections of the document (e.g. code) should be copy-paste easy.

- FAQ
  Description: Answers the most common questions asked by users of the web application.

  Business Rules:
  - Registered & non-registered accounts can access the FAQ.

- The FAQ is tailored for non-technical users.
- Only administrators have the privilege to edit the FAQ.

- User Manual
  Description: Document with instructions and examples on how to use the web app.

  Business Rules:
  - The manual excludes any technical documentation of the web app.
  - The manual includes all necessary information to use the web app.
  - The manual can be accessed by anyone.
  - Only administrators have the privilege to edit the user manual.
  - The user manual is tailored for non-technical users.

2.2.2. Non-Functional

- Font and Font size will be Times New Roman and 12-point respectively for all documentation created.

2.2.3. Pass/Fail Requirements

Pass

- Developer Docs provide a transparent explanation of all the technicalities associated with the web app.
- FAQ helps non-technical users with questions when they use the web app.
- User Manual provides an overview to non-technical users on how to use the web app.

Fail

- Developer Docs fail if it is not comprehensive for software developers.
- FAQ fails if it is not comprehensible for non-technical users.
- User Manual fails if it is not comprehensive for non-technical users.

2.3. Error Handling

2.3.1. Functional

- Client-side Error Handling
  Description: Detecting and mitigating the errors generated by the client.

  Business Rules:
  - All client-side exceptions will result in a user-friendly message when applicable. This message will display when any of the following occur:
    - Server Request Timeouts
    - Invalid Server Request
    - Server Error
    - Invalid User Input
    - Unauthorized Access
    - Required Contact Administrator
  - Exceptions should not cause the system to halt.
  - Quick fixes will be provided as a part of the error handling.

- Server-side Error Handling
  Description: Detecting and mitigating the errors generated by the server.

  Business Rules:
  - All server-side errors will result in a user-friendly message when applicable. This message will display when any of the following occur:
    - Invalid Server Request
    - Server Error
    - Unauthorized Access
    - Required Contact Administrator
  - Only fatal errors (e.g., server/network shutdown) will cause the system to halt.

2.3.2. Non-functional

- The software system must detect the type of error with 100% accuracy.

2.3.3. Pass/Fail Requirements

Pass

- The correct error message as well as possible solution(s) is shown to the client when an error occurs on their end.
- The correct error message is shown on the server side and corrected shortly after.

Fail

- The system halts and does not address the error appropriately.
- The user does not receive contextual information of the error.
- The system does not detect the error.
- The system detects the wrong error type.

2.4. Logging/Archiving

2.4.1. Functional

- Archiving
  Description: The systematic behavior describing the log archiving process.

  Business Rules:
  - All logs older than one month are backed up.
  - All logs older than one year are archived and then removed from the system.
  - If an archive attempt fails, then try again after 3 hours.
  - If an archive attempt fails more than 3 times, then the system administrator is notified.

- Error Logging
  Description: The systematic recording of errors that occurred in the web application.

  Business Rules:
  - One hundred or more failed error log attempts will result in the system administrator being notified.
  - Information regarding the error logged for every user, this includes:
    - Error date & time of the error
    - Error description
    - Error location
    - Which user encountered the error
    - How the user encountered the error

- Only administrators have the privilege to delete error logs.
- Error logs will be machine-parsable with high human-readability.

- Telemetry
  Description: The systematic recording of user and usage data and storing it over a network.

  Business Rules:
  - One hundred or more failed telemetry log attempts will result in the system administrator being notified.
  - Telemetry information will be logged for every user unless opted out, this includes:
    - Date and time of user login
    - Date and time of user logout
    - Date and time of user page visit
    - Date and time of user functionality execution
    - IP address & location

- Malicious Attacks
  Description: The systematic recording of server requests to identify request patterns with malicious intent.

  Business Rules:
  - All server requests, made by any user, are logged to detect denial-of-service attacks.

2.4.2. Non-Functional

- Record of the error-log should be stored into the database at most 5 seconds after the error is detected.
- Error date & time of the errors is 100% accurate.

2.4.3. Pass/Fail Requirements

Pass

- Personal user data is recorded unless the user opts out.
- Error and telemetry data is properly logged and archived.

Fail

- Malicious, error, and telemetry logs are not recorded and archived.
- A system log is not archived at the proper time.
- The system logs do not provide enough contextual information.

2.5. Login / Logout

2.5.1. Functional

- Login
  Description: Starting a session with a corresponding username and password that is associated with an account.

  Business Rules:
  - Credentials (username and password) are required to access the system
  - Users have a username separate from their email that they can use to login.
  - Email addresses can also be used to login.
  - Passwords should be a minimum of 9 characters.
  - Passwords should have a maximum of 50 characters.
  - Passwords shall not be stored as text.
  - Passwords shall be ran through an encryption algorithm, to ensure unrealistic time for brute force
  - Locked accounts require administrator privileges to unlock.

- Logout
  Description: Ending a session to the web application for a given account.

  Business Rules:
  - Users are automatically logged out after 15-30 minutes of being idle. (Low-risk web application)
  - Users are flagged idle when they have not interacted with any page for 15-30 minutes.
  - Users have the ability to automatically logout of their current session.

- Password Reset
  Description: Users have the ability to change their existing password.

  Business Rules:
  - Users have the option to update their password if they are in an active session.

- Users have the option of answering security questions, or using either phone or email authentication in order to reset their password while not authenticated.
- Users cannot use previous passwords when resetting passwords.

2.5.2. Non-Functional

- The user should be able to login and start an active session within 3 seconds of entering their credentials.
- After validating their identity, resetting the password should take no more than 3 seconds.
- The user is able to logout in less than 3 seconds.

2.5.3. Pass/Fail Requirements

Pass

- User provides valid login credentials and begins an active session.
- User provides their current password during the active session to update the password accordingly.
- User provides valid two-factor authentication and resets the password accordingly.
- User ends the active session successfully.

Fail

- User provides invalid login credentials and can start a session.
- User provides invalid password during an active session and can update to a new password.
- User does not successfully validate their identity via security questions of phone/email authentication and resets their password.

2.6. Network Communication

2.6.1. Functional

-

2.6.2. Non-Functional

-

2.6.3. Pass/Fail Requirements

Pass

-

Fail

-

2.7. Registration

2.7.1. Functional

- User Registration
  Description: User must provide information in order to create an account within our system.

  Business Rules:
  - User must provide name, phone, email address, username, and password to create an account within our system.
  - Three security questions are required to be answered in order to register with our system.
  - Existing username cannot be used to register with a new account.

- Terms of Service (ToS)
  Description: Terms that the user must agree to in order to use our web app.

  Business Rule:
  - Users that do not agree with ToS can not register an account.
  - System administrators are the only ones that can modify the EULA.

2.7.2. Non-Functional

- It should take no more than 5 seconds to append the user's account information to our system once the user has registered.
- Registration interface should be user-friendly and straightforward, allowing users to successfully register with 90% accuracy.

2.7.3. Pass/Fail Requirements

Pass

- User accepts the EULA and an account is registered.
- User provides a username not in the system and an account is registered.
- User provides an email not in the system and an account is registered.

Fail

- User declines the EULA and an account is registered.
- User provides an existing username and an account is registered.
- User provides an existing email already within the system and an account is registered.
- A system administrator is unable to modify the EULA.

2.8. Usage Analysis Dashboard

2.8.1. Functional

- Bar Chart
  Description: Application data will be displayed to the system administrator using a bar chart.

  Business Rule:
  - Data should show a comparison of the total amount of clicks for event categories.

- Line Chart

2.8.2. Non-Functional

2.8.3. Pass/Fail Requirements

Pass

-

Fail

-

Notes:
Usage Analysis Dashboard:

"In my system, what are users using the most?"

- High level view (one screen), top metrics

- Tracks any metric that determines what the user is doing

App-specific metrics:
Locations visited

View time range of itineraries (what times the user typically makes plans)

Total itineraries made

Account date creation