BRD
MotoMoto

The New Panelists
2/5/2022

Blake Del Rey
Isabel Guzman
Jacob Sunia
James Austin Jr.
Justin Calma
Kayla Chu
Naeun Yu

Team Leader: James Austin Jr.

Executive Summary

Document Summary

This document contains our business objectives as well as limitations and how we plan to address them. Our delivery outline is scheduled to reach and complete every milestone and retrospective by the end of the semester: December 10th. We also outline every feature and their respective functional and non-functional requirements.

Business Objectives

- Provide a consolidated platform of car builds, parts, events, and interaction.
- Offer part compatibility features to lessen user's time spent.
- Assist users in price analyses and finding popular parts.
- Help users connect with other users interested in similar products or events.

Constraints

- Resources : Budget and time will be limited because of the clients mandated due date in May of 2022
- We limit the availability to California, U.S. as our current region.
- Our application will default to the measurements provided on affiliate programs, Imperial System of measurement and the SI metric system.
- The site will support the American English language.
- MotoMoto specializes in car parts, not whole vehicles nor the purchase of such.
- This product may have information and network latency delay due to third party maintenance.
- Since there are only 5 team members, we will be constrained on outside help we can receive because of the group limit.

Scope

Project Scope

MotoMoto provides link resources on the web application and does not manufacture, sell, or ship parts listed on the website. The product will ensure compatibility with the future version of Google Chrome by the end of product development, May of 2022. This product will include sources to external vendors for potential payment where information may be used by such parties for research purposes.

In-Scope
- Initial System
    - Single-page-application web application
- Initial Browser Support
    - Google Chrome version 94.0.4606.71 or later
- Data to be collected from the user during registration process
    - Name
    - Email address
    - Username
    - Password
- User Registration
    - MotoMoto user account
    - MotoMoto Event user account
- Login and Logout
- Community Board
- Car Builder
- Part Flagging
- Event List
- Part Price Analysis
- Notification System

Page Speed

1) Google's Pagespeed Insights will be used to measure the performance of each individual web page in this web application. A score greater than 80 is acceptable.

Unique Features

Community Board

Abstract:
The Community Board will contain prevalent information to the user. This will allow users to post, subscribe, and explore car builds and user accounts of different topics.

Conditions
- ○ Preconditions: A user must be registered for an account
- ○ Postconditions: A user is able to subscribe to an account, post, and ~~parts~~ while also having the capabilities to upvote, create a post, and/or comment on a post.

Functional Requirements
- ○ Registered Users may select from a drop down menu of available community boards to view the posts within them.
  - ■ Each community board will be selectable from the dropdown where users can scroll and select a community.
  - ■ If there are 100 community boards, the user must be able to select at least one of the 100 by scrolling down through the page itself.
    - ● More community boards may be created by the system's programmers and will not be limited to initially released communities.
  - ■ Logged in users will able to post to a selected community board where a post will be limited to 1-1500 characters
    - ● Titles will be limited from 15-75 characters to ensure that posts can be identified easily.
    - ● Users will be able to post images pertaining to their post and/or users will be able to post their saved vehicle builds.
    - ● Users will not be able to add comments to their post during the creation of their post and only add comments once the post is posted onto a community board
  - ■ Upvoting will be represented by a button where all registered users will be able to click to upvote.
    - ● A new view will update within the display, not allowing for multiple up-vote clicks on one profile post ~~or parts list~~.
- ○ All logged in users are able to comment and upvote on posts
  - ■ If 100 comments are written on a post, all users must be able to see all 100 comments through scrolling in the comments section.
  - ■ If 100 upvotes are given to a post. The post must show that it has 100 upvotes.

- ● Number of upvotes will be tracked next to the upvote button to ensure that users know how many people liked/enjoyed a post ~~or vehicle part~~.
  - ○ Users may subscribe to other profiles to see new posts released by the subscribed profiles
    - ■ Both user and event profiles may be subscribed to.
      - ● Users will get in-app notifications on new profiles posts
    - ■ Specific community boards may be entered to subscribe to posts not seen before.
  - ○ Registered users will be able to post multiple of comments to a post
    - ■ A registered user will not be able to post a message if a comments' length is less than 1 character or more than 1000 characters.
      - ● Users will no longer be able to type if they enter the limit of characters
      - ● Users will not be allowed to post if they don't type.
    - ■ Each user will be able to post multiple times to either comment on a post or ask questions to the original user who created the post.
      - ● This allows for users to engage in an active community and share ideas and thoughts.
    - ■ System failures from this feature must not result in the system going offline

Non-Functional Requirements
  - ○ Community boards will be updated upon new posts in them.
    - ■ Users will be able to scroll down to view events where all events will be on the same page.
    - ■ Users will be able to view posts based on their screen size and will be able to view more if the resolution of their computer is higher (Pixels Per Inch must be higher for users to view more content at once).
    - ■ If the applications community board is unreachable due to host issues.
      - ● Users will be refreshed with a new instance of the page where errors will be noted and handled by the host.
  - ○ Subscriptions made in these community boards will show new posts or updates on a user's main feed
    - ■ If a user is subscribed to community board posts a new post must appear in their subscriber's feed once the system picks up new activity and posts.
      - ● New posts will be updated onto a community board within 30 seconds to at most 5 minutes.
      - ● If the post is not viewable to all users after 30 seconds of the post request, the system will display the error message to the user and the system failure notification will be sent to the admin.
  - ○ Users are limited to one upvote per post

- - - A user will not be able to upvote a post or product more than once to limit spamming.
  - ○ Users can only subscribe to one profile at a time and is not limited to a number of subscriptions
  - ○ Notifications will be set in-app for users to be notified on specific profiles
    - ■ If users enable notifications for a subscribed profile, and the profile makes 100 posts, the user will receive 100 notifications from that profile.
  - ○ All user posts can upload images and vehicle builds
    - ■ All uploaded images will be converted to a jpeg image and must not exceed 1MB as a jpeg image
      - ● Users will be allowed to upload up to 5 images per post at a time as long as each image does not exceed 1MB.
    - ■ A user's post to a community board will only accept one vehicle build per post.
      - ● Vehicle uploads should not exceed 30 seconds as our goal will be to conduct business logic as quickly as possible to ensure quality user experience.

Community Board: Main Feed

Abstract:
This section will contain information according to user subscriptions. Main feed will be updated upon new subscription posts. The feed will present user accounts and car builds of which users have subscribed to.

Conditions
- - ■ Available subscriptions are for user accounts and parts for the purpose of receiving notifications and updates to their feed.
  - ○ Preconditions: Users must be logged into a valid registered account
  - ○ Postconditions: Users will be presented with subscribed content which will be posted in their main feed.

Functional Requirements
- - ○ The main feed will present user account posts and car builds posts of which users are subscribed to
    - ■ If no subscriptions were made (i.e. new user) then the feed will contain posts of highest popularity based on upvotes
      - ● Popularity is determined by at least 100 upvotes
    - ■ User accounts can be subscribed to by selecting the profile and selecting the subscribe button
      - ● Subscribed accounts will allow a user to access in-app notifications for a profile to keep them up to date with the latest released content (see more from notifications).

- - ~~Parts can be subscribed to by selecting them in the parts lists of posts and selecting the subscribe button~~
  - Users will be able to place one single upvote on a post and be able to comment on their favorite car builds in the application.
  - Posts can be selected to reveal more information
    - Posts will have descriptions up to 1500 characters
  - Other user profiles may be accessed while viewing an account by clicking on.
  - System failures from this feature must not result in the system going offline

Non-Functional Requirements
- Main feed will be updated upon new subscription posts
  - Updates will happen in a rough timeline from 3 to 5 minutes once a subscription has taken place to ensure that users are confirmed with their choice of subscription.
  - With minimum resolution (800x600 dpi) on a user's monitor a user should be able to view at least 10 posts at a time and may be more due to resolution changes done to a users monitor.
- Each comment posted on a car build or user posts can range from a minimum of 1 character to 1000 characters
  - Users will be able to comment multiple times if they exceed the 1000 character limit
  - Post comments will not be limited and users are free to post as long as they meet the requirements of posting.
- If a user is subscribed to 100 accounts, recent posts from those 100 accounts will appear in the user's main feed.
  - Recent posts averaging from as early as 1 week to as far as 3 months.

Car Builder

Abstract:
 The Car Builder allows a user to select their vehicle by VIN number or stock vehicle, consisting of all original manufacturer parts, and customize it with OEM or Aftermarket parts.

Conditions
- Preconditions: All users will be able to modify and discard car builds.
- Postconditions: Registered Users will be able to save their vehicle build based on login or will be discarded after the user leaves the car builder. All non-registered users car builds discard after exiting car builder.

Functional Requirements
- User should be able to choose the car year, make, and model OR type in the VIN of their car to generate the car they want
  - Year consists of four digits (####)

- ● Our goal will be to ensure that all vehicles registered after Jan 1, 1950 will be applicable to car builder but we will not guarantee that all parts will have available modifiable parts
    - ■ When car make is chosen, there will be another drop down menu with all the models for that make and year
    - ■ If the user choose to type in the VIN:
        - ● VIN must be 17 characters long
    - ■ If a vehicle does not exist within the modifiable list a refresh will take the user back to the beginning of the car builder.
  - ○ The car is then saved to a logged in profile, with a maximum of 10 saved cars
  - ○ Users can then choose from a list of OEM or Aftermarket parts to add to their virtual car.
    - ■ List of available parts will vary based on availability from third party vendors.
    - ■ When a main component is selected in the builders parts list, there will be a number of categories (which may vary between parts) that will appear.
        - ● Each sub category is selectable to provide specific parts, where the number of parts may vary based on brand and price
  - ○ System failures from this feature must not result in the system going offline
  - ○ Part title (name of a brand specific part) will be clickable to compare prices to other vendors or be comparable on a price standpoint with other branded parts (see part price analysis for more details).
  - ○ The application will retrieve information on these parts from the following sources:

Source Name	Website Hyperlinks	OEM	Aftermarket	Status
Web Scraping	-	✓	✓	Temporary
NHTSA (Gov)	NHTSA	✓		Awaiting Approval
Amazon Services LLC Associates Program	Amazon	✓	✓	Awaiting Approval

Autozone Affiliate Program	Autozone	✓	✓	Awaiting Approval
Pepboys Affiliate Marketing Program	Pep Boys	✓	✓	Awaiting Approval

Non-Functional Requirements
- The application's relational database management system holds every stock car available in the United States.
- Will display 5 makes at a time from a drop down menu and the user can scroll through every car make
- The application's relational database management system will ensure through the server side that all VIN transactions match the car a user owns.
- The application's relational database management system will ensure that all vendor OEM parts and aftermarket parts can be shared with users during vehicle builds or if users want to view a specific part.
- The application's service database holds various aftermarket parts used for different modifications.

Part Flagging

Abstract:
Flags are indicators of modification compatibility and only appear as suggestions to the user in the Car Builder customizing session or car build posts.

Conditions
- Preconditions: All users will be able to use part flagging
- Postconditions: Users will be able to add parts to their respective vehicle builds based on their choice but will be notified on compatible and non-recommended parts.

Functional Requirements
- Users see flags are tagged onto modifications relative to the selected vehicle and have labels: Compatible or Non-Recommended
- When users put incompatible parts into their online build, they are notified with a warning that their part is not recommended for their selected vehicle
- When users view car build posts, flags may appear in the parts lists set by the posting user

- - Users can select flags, when selected they will present a review set by the specific user in view and a rating based on other users' upvotes
  - Users can place new flags on parts they believe to be incompatible.
  - The flag database will update upon every new flag set given that a user provides the necessary information
  - Flags' credibility are improved by part assembly in vehicle posts, user reviews, and ratings
  - System failures from this feature must not result in the system going offline

Non-Functional Requirements
- Flags are tagged onto modifications relative to the selected vehicle and have labels: Compatible or Non-Recommended
  - The Compatible label means that a part, OEM or Aftermarket, is manufactured to be installed with the selected car
  - The Non-Recommended label means that a part is considered potentially incompatible which we do not recommend purchasing
  - This label includes interactable user vehicle posts that contain the part, user reviews, and ratings
  - Parts will show as non-recommended only after receiving 100 incompatibility flags from users.

Event List

Abstract:
This section contains all events being hosted in a specified area. Users can select events to view details about them or to register to attend them.

Conditions
- Preconditions: All users will be able to view events
- Postconditions: Only registered users will be able to register to attend car events

Functional Requirements
- When users select the post it will expand details and reveal a register button for which a logged in user can confirm or cancel their intention of attending an event
  - Address, City, State, and Zip code will be provided
- When users access the Event Feed they will be prompted to input a zip code to discover events in that area OR users can search for specific events by event title or event account.
- Users may input filters: most popular based on upvotes, car type, new or old based on time of post, radial distance relative to the zip code, and date intervals
- Users can make event accounts to create an event
- System failures from this feature must not result in the system going offline

Non-Functional Requirements

- Event list will be updated once in the beginning of the week and once in the middle of the week
- Each event post requires the following: event title, location, time, and date
  - Event title has a maximum 100 characters
  - Date will appear ##/##/####
  - Time will show for whatever time zone the user is in
- All uploaded images will be converted to a jpeg image and must not exceed 1MB as a jpeg image
- An Event Account is limited to one event post per day
- Activity includes all users that have registered for the event and when a new registered user appears

Part Price Analysis

Abstract:

The Part Price Analysis will be a page that gives detailed price points (in USD) and a price graph of a brand specific part where comparisons are made from vendors who sell the same part or those parts that are wanting to be compared against.

Conditions

- Preconditions: All users will be able to use the part price analysis without a validated profile
- Postconditions: Users will be able to either compare prices with supported vendors or can compare the cheapest prices of a specific type of car part with other brand specific parts.

Functional Requirements

- Users can browse a large assortment of car parts where brand specific parts are listed with all other parts.
- When users choose a car part they are immediately notified if it is or is not compatible with their current vehicle
- Users may subscribe to a part to be notified of part price updates
- Users can compare different brand specific parts
- Users can pick the link to where they'd like to buy from on the part price page
- Users will be able to hover over a vendors line on the graph to see price points of previous or current prices of the item itself
- Part price page will have a table where each vendor's price is listed and a compare button to allow comparison of prices with other brand specific parts of the same type.

- ○ Every vendor will have selectable icons that will link a user to the part to purchase (links will be embedded with vendors icons)
  - ■ Part prices will be sorted on the table from lowest to highest where linked icons will be available for users next to prices
- ○ Updates on new part prices will be updated accordingly based on our affiliate program data sources
- ○ System failures from this feature must not result in the system going offline

Non-Functional Requirements
- ○ Prices can be ordered by lowest price to highest price or ratings
  - ■ Prices listed will be determined by the current lowest price of a brand specific part (prices will be listed by USD)
  - ■ Prices will be formatted as USD $####.##
    - ● Prices will be listed in the basic USD pricing system. All digit slots will be filled where every penny be shown for each vendor.
  - ■ Ratings will determined by upvotes
- ○ Specific Part titles will be able to be selected by the user to retrieve a full in depth view of a brand specific part.
  - ■ Titles will not be restricted due to information fetching from the outsourced vendors which our application will be flexible towards each vendor.
- ○ All vendor icons of a specific part will have the ability to be selected by the user to be sent to a vendors website where the brand specific part is sold (restrictions and separate accounts may be necessary to purchase)
  - ■ Vendor icons do not exceed 1MB, and will be scaled to fit the needs of the price table.
- ○ Product price page will contain a line graph that has all vendors and their price of the brand specific part over the past 6 months, thus will be updated every week to ensure that part price steals/discounts, missed out on are shown.

Notification System

Abstract :

The Notification System will notify users on any subscription activity inside and outside the application depending on the type of notification.

Conditions
- ○ Preconditions: Users must be logged into a valid registered account
- ○ Postconditions: Logged in users will be able to receive notifications based on their subscriptions to accounts, posts, two-factor authenticator, or vehicle parts.

Functional Requirements
- ○ Out-app notifications

- - - Deliver to user's email inbox
    - Notification activities will include:
      - New events posted by a subscribed event account
      - Upcoming registered events within one and a half to the date of event
    - Email notifications will be delivered Monday and Wednesday every 3PM PST containing all the updates that happened in the previous 24 hours.
      - All event accounts will be aware of sudden last minute changes and no notifications will be sent to users after messages are delivered.
    - Unsubscribe option is available: the users may opt-out from receiving notification emails
  - In-app notifications
    - Push an instant notification to the users account inbox when they are logged in the application upon new subscription activity.
    - All updates on new vehicle builds will be posted in-app
    - In-app notifications will be sent to a virtual inbox within the application and will consist of all subscription activity:
      - Price drops of a subscribed part.
      - New posts or events from a subscribed account.
      - Upcoming registered events based on date.
      - Updates to a registered event details.
  - Users have the option to unsubscribe to any notification they want
  - In-app notifications will be sent to a virtual inbox within the application and will consist of all subscription activity
    - An in-app notification will contain:
      - A message allowing the user to understand the reason for notifying.
      - A notification summary of each subscribed activity or accounts.
      - Every summary will present the type of activity, user profile name, and date of occurrence.
      - When a summary is selected the user will be redirected to the origin of the notification (e.g. a part price drop will redirect to the part in Part Price Analysis).

Non-Functional Requirements
  - Out-app notifications will be sent to a user's email inbox that was used during sign-up and will only consist of registered vehicle events and two factor notifications.
    - Every out-app notification summary will present the type of event, date of occurrence, and location of an event.

- If changes to the event were made, the email message will present the new post and a link redirecting to the specific post
  - A summary will be in the range of 1-1000 characters based on the message that will be sent to the user.
- If an event has already passed, then users will not receive any new information on a former event
  - Notification of an event will stop the day it starts to ensure that users will no longer receive notifications once an event has past
- Two factor notifications will present a random generated 5 digit code where users can validate themselves within the program.

Encrypted Direct Messages

Abstract:
Direct messages is a page that allows users to chat and connect with each other privately

Conditions
- Preconditions:
  - Users must have an account before they will be able to send messages
- Postconditions:.
  - Users will be able to safely send and view messages

Functional Requirements
- Users will be able to send direct messages to other users
  - If the users have never messaged each other before, a request will first be given to the receiver asking if they will like to view the message and open a chat with the user
- Users will be able to access previously opened chats
- Users will be able to delete chats
- Users will have the ability to block other users and if the blocked users attempts to send a message, the message will not successfully go through

Non-Functional Requirements
- There is a character limit of 1000
  - If the character limit is exceeded, the user will not be able to continue typing
  - empty messages will not be sent
- When a message is sent, a notification will appear on the receiver end informing them of the message
- All direct messages will be written in ASCII characters

Personalized Recommendation

Abstract:

The system will show recommendations on the users main feed page based on their preference

Conditions

- ○ Preconditions:
    - ■ User must have an account
    - ■ User has to interact with the application before their feed can be personalized
- ○ Postconditions:
    - ■ Posts will show up on the user's main page based on their preference

Functional Requirements

- ○ If the user dislikes a recommendation, they can choose to have it removed from their feed
- ○ When the user first makes their account, the feed will be filled with the newest post
- ○ The system will store the users action so that the system will know what to show on the users feed
- ○ Favorited/Followed users posts will be directed to the personalized recommendation

Non-Functional Requirements

- ○ Users will only be able to select one single vehicle brand for favorite

Event Account Verification (Ratings and Reviews on Event Accounts)

Abstract:

The system will give users the ability to rate and review the event that they have attended and the event poster. Furthermore, this lets users know who is trusted to conduct business with and users who are not trusted.

Conditions

- ○ Preconditions: Users must be logged into a valid registered account and there must be an event that completed within 48 hours
- ○ Postconditions: Completed event postings and event accounts will be reviewed by event attendees. Ratings and reviews will be visible to logged in users.

Functional Requirements

- ○ Event attendees will be able to rate and review the event holder and the event they have attended.

- - Users rate from a 1-5 star scale.
    - A user can rate and review any event account only once, but it can be changed at any time.
    - A review is limited to 4 images and 40 MB per image
    - The review is limited to 1000 UTF-8 characters.
  - Only validated logged in users are able to view all ratings and reviews that others have posted.
    - Ratings and reviews regarding an event will viewable on the event listing
    - Ratings and reviews regarding the event poster will be viewable on the event poster's profile
  - Event accounts that have an overall rating below 2.5 stars will be suspended from holding events until they wait 2 weeks or they appeal with an admin

Non-Functional Requirements
- - The system is considered fast if the user is able to post their rating and review within 3 seconds
  - The system will generate only one rating and review per product or user
  - The system will ensure that none of the user's personal sensitive information is posted, unless the users themselves wrote it in the description.

The error rate of users attempting to follow another user is within 10%

User Profile

Abstract:
The user profile allows the users to have their own unique views and shows their own posts and the posts that they have upvoted to revisit the things that a user enjoys

Conditions
- - Preconditions: All users must have an account
  - Postconditions: Users will be able to logout and edit their profile through the user profile

Functional Requirements
- - Users will be able to go to posts by clicking on the post link provided
  - Users will be able to modify their description of their account to be displayed
  - Users will be able to edit their preferences, username, and profile image through their profile edits
  - Upvoted posts will display all posts upvoted by a user where the post is not created by the user

Non-Functional Requirements
- - Users can have a description of 160 characters

- User Posts will display only one page at a time where 5 posts are per page display
- Users can only have one profile image where the image does not exceed 1 megabyte.

Note Dashboard

Abstract:
The dashboard allows the users to write notes down in a dedicated space within the website

Conditions
- Preconditions: All users must have an account
- Postconditions: Users will be able to see their notes on the dashboard

Functional Requirements
- Users will be able to create a note by clicking an add button
  - the button will be disabled if the user exceeds 100 notes
- Users will be able to delete a note by pressing a delete button
- Users must press the save button in order to save the notes (no auto-save)
- Users can order their notes by the date made, or in alphabetical order
  - default is by date

Non-Functional Requirements
- Max lag time per note would be 5 seconds
- Users will only be able to have 100 notes at one time
- All notes will be written in ASCII characters
- There is a character limit of 5000
  - If the character limit is exceeded, the user will not be able to continue typing
- There is a 30 character limite for the title
- Empty titles will note be sent

Meeting Point Directions

Abstract:
Meeting Point Directions will help the value of MotoMoto because users will have access to a map showing street directions from a location of their choosing to the event location

Conditions
- Preconditions: All users will be able to view events that have been posted and posted events must contain a valid location.
- Postconditions: All users will be able to view a map showing street directions from a location of their choosing to the event location.

Functional Requirements

- If there are available event listings, all users will have access to a map showing street directions from a location of their choosing to the event location.
  - With the users' consent, their current location will be used as the origin point to connect to the preferred meeting location as the default.
    - The origin location to begin directions from can be changed by the user.
- The user will be able to choose from two different travel modes
  - Driving / Walking
- Maps and geolocation services will display multiple possible street directions to the specified event location
- The users inputted geolocation origin will not be saved within the system when they determine street locations to the event location.

Non-Functional Requirements

- The system would be considered fast if it is able to retrieve the user's location and load up directions for the user within 10 seconds
- The system must ask the user for permission to use their current location in order to be compliant with using location services
- The error rate of users attempting to search with an invalid location is within 10%
- Unless permission was allowed, the system will not be able to get the user's location and not allow the user to interact with the feature

Expansion Points

- The estimated time from origin to destination will be displayed for each possible street direction route
- Users will be given the ability to add waypoints to possible routes
- Users will be given two additional travel modes to select
  - Bicycling / Transit (i.e Bus, Railway, Train, Tram)

Security

Authentication

Abstract:
Authentication will represent as a mechanism for identifying a valid user of the web application.

Conditions:

- Preconditions: User must not already have an active authenticated session with the system on the chrome browser, otherwise authentication is not possible. User

must be on login view or attempting to access to a protected resource as defined in Authorization

- Postconditions: System will be able to grant access to a specific user and is able to authenticate access.

Functional Requirements

- The user must provide a valid security credentials whenever attempting to authenticate with the system
- Valid security credentials consist of a valid username and valid time-based one time password (OTP)
- For each failed attempt, the account undergoing authentication and the IP address that initiated the authentication request will be recorded.
- System failures from this feature must not result in the system going offline

Nonfunctional Requirements

- OTP Username and Passwords will consist of the following characters:
  - a-z
  - 0-9
  - @!
- OTP is defined in NIST SP 800-63b section 5.1.4.1
  - OTP is changed upon every successful use
  - OTP expires every 2 minutes
  - OTP must be at minimum 8 characters
    - Valid characters will consist of the following:
    - a. a-z
    - b. A-Z
    - c. 0-9
- A maximum of 5 failed authentication attempts within 24 hours for the same account before account is disabled
  - 24 hour timer begins on the first failed authentication. If the account was not disabled after 24 hours, then the fail count resets to 0.
  - Account is locked until a valid account recovery is performed by the user or by system admin. Upon successful account recovery, the failed count resets to 0.

Authorization

Abstract :

A mechanism for restricting access to protected resources (e.g. functionalities, data, and views) to only valid users

Conditions

- Preconditions:

- ■ User must be authenticated to enforce user-specific restrictions
- ■ User account must be active
  - ○ Postconditions: User accounts will be able to access protected functionalities within authorization which includes :
    - ■ Reading
    - ■ Writing
    - ■ Viewing

Functional Requirements
- ○ By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of user's identity
- ○ The system must prevent unauthorized users from viewing, modifying or deleting any protected data (scalar or aggregate data)
- ○ The system must prevent unauthorized users from executing any protected functionality
- ○ The system must prevent unauthorized users from viewing or interacting with any protected views
- ○ Any user access modifications should be active upon the next successful authentication by user
- ○ System failures from this feature must not result in the system going offline

Non-Functional Requirements
- ○ The operation and timestamp of each unauthorized access will be recorded by the system
  - ■ Timestamps will be recorded as follows:
    - ● MM/DD/YYYY
    - ● HH:MM

Logout

Abstract

Will be used as a mechanism for ending a registered user's active authenticated session.

Conditions
- ○ Preconditions:
  - ■ User must have an active authenticated session on their google chrome browser, otherwise the user is unable to perform the operation
  - ■ User must be on view with Logout option on their respective views
- ○ Postconditions:
  - ■ User performs a logout request and the active session ends.
  - ■ The user is automatically navigated to the default home view of the system with the default culture settings.

Functional Requirements
- ○ The user will be navigated to the home view of the system upon successful completion
- ○ System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- ○ The current active session on the device will end within 5 seconds upon invocation

User Administration

Account Creation

Abstract:

A mechanism for creating new user accounts within the system

Conditions
- ○ Preconditions:
  - ■ User must not have an active authenticated session
  - ■ User must be on account creation view
- ○ Postconditions:
  - ■ User registers with a valid email and valid passphrase. A system message displays "Email confirmation pending".
  - ■ The user receives a confirmation email within 15 seconds upon invocation of the system message.
  - ■ The user completes email confirmation within 24 hours.
  - ■ Users are notified of their username. A system message displays "Account created successfully"

Functional Requirements

- ○ System administrators cannot be created using Account Creation feature
- ○ All user accounts must be stored in a persistent data store
- ○ The user provides a valid email address that belongs to the user
- ○ The user provides a secret passphrase for requesting OTP
- ○ A unique URL link is provided for every account creation request
- ○ The URL link is valid for only 24 hours from date of account creation request

Nonfunctional Requirements
- ○ The OTP will be a passphrase of 8 characters which will be represented with:
  - ■ blank space
  - ■ a-z
  - ■ A-Z
  - ■ 0-9
  - ■ .,@!
- ○ The user is given a system-wide unique username upon validation of email
- ○ Valid characters will consist of the following:
  - ■ a-z
  - ■ 0-9
  - ■ .,@!

Account Recovery

Abstract:
A mechanism for regaining access to an active or disabled account

Conditions
- ○ Preconditions:
  - ■ User must not have an active authenticated session on the device, otherwise the user is unable to perform the operation
  - ■ User must be on account recovery view
- ○ Postconditions:
  - ■ User provides valid username and associated email. User receives a recovery email within 15 seconds. User completes the recovery request within 24 hours.
  - ■ A system message displays "Account recovery completed successfully" within 5 seconds of completion.

Functional Requirements
- ○ The user must provide a valid username and associated email to receive a system email containing a recovery URL link

Nonfunctional Requirements
- ○ The user must receive recovery email within 15 seconds upon invocation

- ○ Recovery URL link must be navigated to and completed within 24 hours, otherwise a new recovery request must be started
- ○ A maximum of 5 recovery requests can be made per account per calendar month.
- ○ Recovery request limit resets every 00:00:00AM local time on the 1st of the month.
- ○ System failures from this feature must not result in the system going offline

Account Deletion

Abstract:
A mechanism for deleting a user account
Conditions
- ○ Preconditions:
  - ■ User must have an active authenticated session
  - ■ User must be on account deletion view
  - ■ User has permission to delete account
- ○ Postconditions:
- ○ User chooses to delete account and confirms action
- ○ All PII data and user account data is permanently deleted from the system
- ○ A system message displays "Account deletion successful"
- ○ Upon acknowledgment of the system message, the user is automatically navigated to the home view with default language and culture settings.

Functional Requirements
- ○ All personal identifiable information (PII) along with the user account data is permanently deleted from the system
- ○ Account deletion is irreversible
- ○ Only a system administrator account can delete another system administrator account
- ○ System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- ○ Please Refer to Account creation on specific characters that the system will take input of

User Management

Abstract:
A mechanism for administration of any user account

Conditions
- ○ Preconditions:
  - ■ User must have an active authenticated session

- ■ User must be on user management view
- ■ User must be a system administrator
- ○ Postconditions:
  - ■ User is able to perform any single UM operation within 5 seconds upon invocation. A system message displays "UM operation was successful"
  - ■ Users are able to perform less than 10K UM operations in bulk within 60 seconds. A system message displays "Bulk UM operation was successful"
  - ■ User is able to perform 10K UM operations in bulk within 60 seconds. A system message displays "Bulk UM operation was successful"

Functional Requirements
- ○ All operations are applied to a persistent data store
- ○ Only system administrator have access to the User Management view
- ○ The system administrator will have access to view and modify all accounts and their associated user profile data within the system
- ○ Single Operation
  - ■ Create Account
  - ■ Update Account
  - ■ Delete Account
  - ■ Disable Account
  - ■ Enable Account
- ○ Bulk Operation
  - ■ Multiple operations (e.g. all the same or mixed) within the same request
  - ■ Requests can be made through an uploaded file extract
- ○ All single and bulk operations must be able to affect any user account/profile attribute within the system
- ○ Only a system administrator account can create other system administrator accounts
- ○ The system must have at least one system administrator account with total system access at all times
- ○ System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- ○ Single Operation should be completed within 5 seconds upon invocation
- ○ Bulk operations have a maximum of 10K operations per request
  - ■ File extract cannot be greater than 2GB in size
  - ■ Operation should be completed within 60 seconds

System Observability

Usage Analysis Dashboard

Abstract:
A visualization mechanism for gaining insight on user behavior within system

Conditions

- ○ Preconditions:
    - Persistent data store must be active
    - Persistent data store must accessible by the system
    - User must have an active authenticated session on the device
    - User must be on Usage Analysis Dashboard view
    - User must be a system administrator
- ○ Postconditions:
    - User is able to navigate to the view. The view loads within 15 seconds. All KPIs automatically refreshes data within 60 seconds.

Functional Requirements
- ○ All data must be fetched from a persistent data store
- ○ Persistent data store must be active at all times
- ○ Persistent data store must accessible by the applications system
- ○
- ○ System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- ○ Key Performance Indicators (KPIs):
    - ■ The top 5 most visited view of all time (bar chart)
    - ■ The top 5 average duration per view of all time (bar chart)
    - ■ The number of logins per day within the span of 3 months (trend chart)
    - ■ The number of registrations per day within the span of 3 month (trend chart)
    - ■ Two application specific feature metric
- ○ All KPI data must be automatically refreshed in intervals of 60 seconds
- ○ The view must load within 15 seconds upon completion of navigation.

Logging

Abstract:
An internal mechanism for tracking all events of the system for auditing
Conditions

- Preconditions:
  - Persistent data store must be active
  - Persistent data store must accessible by the system
  - Persistent data store must have storage capacity for log entry
- Postconditions:
  - Data Log is recorded onto the relational database management

Functional Requirements
- All log entries must be immutable
- All log entries must be saved to a persistent data store
- All log entries must contain a UTC timestamp, log level, user performing operation, a category and a description/message
- Valid Log Levels
  - Info - for tracking flow of system
  - Debug - for tracking key information crucial to maintainers of the system
  - Warning - for tracking events that may lead to system failures
  - Error - for tracking system errors
    - Valid Categories
    - View
    - Business
    - Server
    - Data
    - Data Store
- The logging process must not block any user from performing any interaction with the system
- Data store must always be active as long as the application is running on the web
- System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- The logging process must complete within 5 seconds upon invocation

Archiving

Abstract:
An internal mechanism for offloading log entries to preserve system resources

Conditions
- Preconditions:
  - Persistent data store must be active
  - Persistent data store must accessible by the system
  - Archival destination location must have storage capacity

- ○ Postconditions:
    - ■ New archival data is stored for later usage if the application calls upon it.

Functional Requirements
- ○ Archival process must consolidate and compress entries being archived
- ○ Archival process must offload entries to another location
- ○ Archival process must remove offloaded entries from the system after successful archival
- ○ System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- ○ Archival process must execute every 00:00:00AM (local time) on 1st of the month
- ○ Archival process must only offload log entries that are older than 30 days
- ○ Archival process must complete within 60 seconds upon invocation

Universal Requirements

User Privacy

Abstract:
A mechanism to inform and protect user data from being used without consent

Functional Requirements
- ○ EULA per GDPR or California Consumer Privacy Act (CCPA) / California Privacy Rights Act (CPRA)
- ○ Opt-out of user data collection and selling of user data
- ○ Explanation of use of data
- ○ Deletion of user data / user account

UI/UX

Abstract:
The goal will be to provide an intuitive interface for users to interact with the application

Conditions
- ○ Preconditions:
    - ■ All features that requires user interaction within the system

Functional Requirements
- ○ All text must be in the selected language and culture setting
- ○ All formats align with the selected unit of measurement
- ○ All views must not require assistance from another human understand how to interact with the view

- All system messages to the user must be displayed in the default culture settings for non-Authenticated users and the selected culture settings for Authenticated users
- System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- All system messages must appear within 5 seconds of the resolution of an operation

Error Handling

Abstract:
A mechanism to prevent system failures from making the system go offline to any user

Functional Requirements
- System failures from this feature must not result in the system going offline

Nonfunctional Requirements
- Allowed System Failures:
    - Web Server loses internet access
    - Cloud/Host Provider outage