Team Big Data

U-tification

Business Requirement Document

Date: 10/2/2022

Team Leader: Joseph Armas

Team Members: Joshua Gherman

Rhoy Oviedo

Frank Curry

Ghabrille Ampo

David DeGirolamo

Git Repository: https://github.com/JosephArmas/cecs-491A-Team-Big-Data

Version History

Current Version: Version 2

- Summary
    - Removed statement about stakeholders.
- Requirements
    - General
        - Added General Requirements
        - Added descriptions to help with laymen.
        - Adjusted format of each feature.
    - General Success
        - Added: Users are able to view all pin types.
    - User types
        - Added new user type: service
    - Litter Map
        - Adjusted Functional Requirements
            - Added zoom and pan controls
        - Adjusted non-functional requirements
            - Functional Requirement to show parks and streets.
            - Non-Functional Requirement to show street names.
            - System scalability of 10,000 users.
            - Maintain acceptable input lag.
    - Pinning
        - Added
            - NFR:The system shall be scalable to support up to 50,000 pins without the need of additional servers.
        - Deleted
            - Functional: Users must be able to view all pins.
    - Pinning (Admin)
        - Updated modify or delete functionality.
    - Alerts

- ■ Added mark as read as a functionality
- ○ Reputation
    - ■ Changed some non-functional requirements to be functional requirements.
    - ■ Added new non-functional requirements.
- ○ Event Creation
    - ■ Added new non-functional requirements.
- ○ Event Joining
    - ■ Added new non-functional requirements.
- ○ Service Creation
    - ■ Added
        - ● Functional: Service providers can select what types of pins they'd provide services for.
        - ● Limited to 10,000 service providers
        - ● Non functional requirement limit of 1 service per user.
- ○ Service Requests
    - ■ Added new NFR and FR requirements
- ○ Services (Admin)
    - ■ Modified Functional Requirements
- ○ Picture Upload
    - ■ Added new NFR

Previous Version: Version 1

Table of Contents

Summary

U-tification aims to provide a litter map of litter, abandoned items, and vandalism that will have a management system in order to map places of high amounts of rubbish, track valid users, and increase awareness between places of garbage and local communities. In order to achieve our goal of beautification across California, we need to have a litter map that tells a cohesive story about local cleaning events and areas that are in need of cleaning services. Our goal is to have this website launched by May 10, 2023.

Requirements

Requirement
General Requirements

Description
Functional and Non-Functional Requirements that apply to the entire application.

Functional Requirements

1	A User must be Logged In.
2	Users exclusively have write access to their own profile or post.
3	Admins are able to access User data.
4	Users are able to view all pin types.
5	The system must keep the user location to the respective user.
6	The system must accept valid zip codes only from California.

Non-Functional Requirements

1	All Features will be available using a web browser: Chrome 104.x (64 bit).
2	All Features are expected to resolve issues within 5 minutes.
3	All Features should be available to 95% of California's residents every month during any time of day.

General Success Criteria	
1	All text uses a standardized font.
2	All text will use *en-us*.
3	All text will be *legible*.
4	Users do not directly communicate with *Backend*.
5	Data is both accurately displayed on the Client side as well as Server side.

General Failure Criteria	

1	Text flagged as invalid: Not standardized font, Not in en-us, Illegible.
	● Error Handling:
	    ○ No error message is displayed.
	    ○ The website will reload itself.

2	Data is not accurately recorded on Client and Server end.
	● Error Handling:
	    ○ Display message will be displayed: "Error has occurred. Reloading"
	    ○ The website will reload itself. Data on both sides will reset as if the event did not occur.

3	User communicates directly with the Backend.
	● Error Handling:
	    ○ Connection between User and Backend is severed immediately
	    ○ Backend is reset to state prior to User interaction.

Feature
User Types

Description
Ensures the user's account has the authorization to do the action they want. Each user has their own individual account and is able to make any changes with proper authorization. Users are able to login to their account with their proper credentials and exit with a logout button.

Admin	
1	Highest level user.
2	Able to perform actions that other users cannot.
3	Not bound by reputation.
4	Have read access to all users' information and interactions.
5	Not able to create pins, create services, request services, or upload pictures.
6	Able to ban and unban user accounts.
7	Able to recover user accounts.

Reputable	
1	Middle level user is able to perform all actions the lowest level can.
2	Authority to create events and request services.
3	Reputation must be greater than or equal to 4.2 rating.

Service	
1	Low level user. Able to perform all actions the lowest level can.
2	Able to add business to list of businesses
3	Able to receive notifications from user requests

Regular	
1	Lowest level user. Reputation is lower than 4.2 rating.

Feature
Litter Map

Description
Displays a map of hot spots for litter, abandoned items, and vandalism. This feature adds to the value of the project by being the source of any user's interaction and data logging.

Precondition
User type is either: Admin, Reputable, Service, or Regular.

Functional Requirements	
1	The system must allow a map of California to be displayed in order for the user to use it.
2	The system must allow the user the ability to move the map along the X-Axis and Y-Axis.
3	The system must allow the user to hold the mouse left click to pan across the map.
4	The system must allow the user to use touch controls to pan across the map.
5	The system must allow the user to zoom in as close as 1 mile on the map. As in the zoom can get as close as rendering one square mile of the map.
6	The system must allow the user to zoom out as far as 100 miles on the map. As in the zoom can get as far out as rendering 100 square miles of the map.
7	The system must allow the user to view all of the currently active pins.
8	The system must allow the user to filter the pins based on zip code as well as distance from the user's current position.
9	The system must have a map with streets and parks with names so that users can identify locations.

Non-Functional Requirements	
1	The system shall have the map load no longer than 10 seconds under any circumstances.
2	The system shall maintain at most 0.5 seconds of input lag with 10,000 visitors utilizing the feature.
3	The system shall have coordinates formatted as follows: 0.000000, 0.000000
4	The system shall be scalable enough to support 10,000 visitors.

Successful Operation
The user is able to load the map, move the map along the X-Axis and Y-Axis, adjust the zoom, filter pins, and view all pins. The Map maintains at most 0.5 seconds of input lag when 10,000 visitors are utilizing the feature.

Failed Operation

The map takes longer than 10 seconds to load.
● Error Handling:
   ○ The system should then inform the user that the map has timed out.
   ○ Then the system should Allow the map to be reloaded.
The map has slow response time i.e. user moves map the map moves more than 0.5 seconds later.
● Error Handling:
   ○ The system must add additional scale when overpopulated.
The map does not display street names or parks.
The map doesn't zoom.
The map doesn't pan.

Feature
Pinning

Description
An extension of the map feature where users are able to create pins on the map.

Precondition
A User is of type Regular, Service, or Reputable.   A User must be using the Map.

Functional Requirements	
1	The system must allow users to place a pin anywhere on the Litter Map.
2	The system must allow users the ability to remove any pin that their respective account has placed.
3	The system must allow users the ability to modify the details of any pin that their respective account has placed.
4	The system must allow the pin creator to add an image to the pin.
5	The system must allow the user to filter the pins by type: Litter pins, Group Event pins, Junk Item pins, Abandoned Item pins, and Vandalism pins.
6	The system must display the title and description of the event when the pin is accessed.
7	The system must have a way to let users exit the details of a pin.
8	The system must allow users to view the date, time, and location of created pins. Location details include zip code, global coordinates, and city.
9	The system must allow users to mark a pin as completed, removing it from the map.

10	The system must allow the user to view their cleaned or deleted pins in their User Profile.
11	The system must have the user confirm that they want to create a pin.
12	The system must log all pins created and deleted. The details included, the pin type, title, description, any images attached.

Non-Functional Requirements

1	The system shall allow a user to create a pin once every 3 minutes. Users are unrestricted by time of day.
2	The system shall be scalable to support up to 50,000 pins without the need of additional servers.
3	The system shall have pins of different types be shown in different colors. ● Litter pins shall have a blue color: #1369f2. ● Group Event pins shall have a yellow color: #f3f70f. ● Junk Item pins shall have a green color: #07b30d. ● Abandoned Item pins shall have a red color: #d11d17. ● Vandalism pins shall have a purple color: #930ff2.

Successful Operation

The Pinning Feature is considered successful if:

The Litter Map is viewable by a User.

Pins are shown with appropriate color.

Pins are displayed on the map.

Pins are shown on the map exactly where the pin was dropped.

There is a confirmation notification when creating a pin.

Failed Operation
The Pinning Feature will fail when encountering: Pins are displayed inappropriately.   - The User is notified of the error.   - The User is allowed to retry their pin request.   A map is not viewable by a User.   - The User is not notified.   - The map is automatically reloaded.   Pins are shown with an invalid color.   - The User is not notified.   - Invalid pins are reloaded to match data on the Server side.   Users are unable to post pins.   - The User is notified of the error.   - The website is automatically reloaded.   Users are able to post pins more than once every 3 minutes.   - The User is notified of an error.   - The most recent post by the User is removed.

Feature
Pinning (Admin)

Description
Part of the pinning feature that describes how Admins interact with the pins.

Precondition
A User is of type: Admin.

Functional Requirements

1	The system must allow Admin read access to all User's pin data.
2	The system must allow Admin the ability to modify or delete User's pin data in the case that a user's pin is inappropriate. In that it contains profanity, slurs, or crude images

Non-Functional Requirements

1	The system shall take no longer than 7 seconds to delete User's data.

Successful Operation
User's data is removed from the Server side. Admin is able to view a User's pin data.

Failed Operation
User's profile does not reflect Server side upon removal.   - Server side takes priority and will have its data reflected on the User Profile.   - Admin is notified if their delete process went through.

Feature
Alerts

Description
Notifications that inform users of updates to pins.

Precondition
A User is of type: Regular, Service, or Reputable.

Functional Requirements	
1	The system allows users to accept notifications from entered events.
2	The system allows users to accept notifications from pin drops by zip code.
3	The system allows users to accept notifications from pin drops by type: Litter pins, Group Event pins, Junk Item pins, Abandoned Item pins, and Vandalism pins.
4	The system allows users to ping other users when creating a pin, the ping only goes to those accepting notifications.
5	The system must have a method for the user where they can view a list of notifications.
6	The system must have Service Users receive notifications from User requests.
7	The system must allow the user to mark a notification as read.

Non-Functional Requirements	
1	All alert functions with the exception of Admin alerts are off by default.
2	The list of notifications marked as read will automatically be removed from the list 7 days after being marked.

3	The list of notifications must display all the notifications in no greater than 2 seconds.
4	The system shall hold up to 100 notifications before they are automatically marked as read by the system. Only 1 notification will be marked as read, the oldest notification.
5	The notification list must be available to the users 95 percent of the time during each week PST.

Successful Operation

The user receives a notification about a pin they accepted notifications for and is able to view it.

The user does not receive a notification for a pin they are not accepting notifications from.

Failed Operation

The user never receives the notification that they were meant to receive.

- The User is not notified of an error.
- The system shall automatically try to resend the notification.

The user receives a notification for a pin they did not accept notifications for.

- The User is notified of an error.
- The notification is deleted from the User's profile.

The pin creator's notification never actually deploys.

- The User is notified of an error.
- The system shall automatically redeploy the notification.

The list of notifications does not show the correct notifications.

- The User is notified of an error.
- The list of notifications automatically is updated with information from Server side.

The user receives a notification for the wrong pin type.

Feature
Alerts (Admin)

Description
Part of the Alerts feature that defines how an Admin is able to control alerts.

Precondition
User type is Admin.

Functional Requirements	
1	The system allows Admins to send notifications to Users.

Non-Functional Requirements	
1	The notifications are the same as the notifications of a pin, but distributed to all applicable users.

Successful Operation
The Admin notification is sent to all applicable users and received.

Failed Operation
The Admin notification is not sent to the correct users, or never sent at all.

Feature

Reputation

Description
A feature of the system that gives all users a rating that is adjusted by interacting with pins and other users.

Precondition
A User is of type: Regular, Service, or Reputable.

Functional Requirements

1	The system must allow users to have a reputation rating from zero to five as a decimal value. Example: 3.0
2	The system must allow all users to view User reputation.
3	The system must allow for reports to impact their reputation, either negatively or positively.
4	The system must have a way to report other users, not the user's own profile.
5	The system must allow for the user to view reports submitted by the respective user.
6	The system must keep the reputation from being altered on the browser.
7	The system must allow users to gain reputation by creating or marking pins as completed.
8	The system must log all changes to the reputation.
9	The system shall recognize a reputable user as a user with a 4.2 rating.
10	The system shall start every user with a rating of 2.0.
11	The system shall adjust the users rating by sum score divided by the number of reviews. (sum score / number of reviews)

Non-Functional Requirements	
1	The system shall display the rating as a ratio out of five "stars" along with the decimal rating.
2	The system shall retrieve other user reputation no later than 0.5 after requested.
3	The system shall notify a User when they change from Regular user to a Reputable user.
4	The system shall support up to 5,000 reports on a user's reputation to calculate rating.

Successful Operation

A user successfully reports another user positively, the other user's reputation increases.

A user successfully reports another user negatively, the other user's reputation decreases.

A user is able to see the reputation of other users.

Failed Operation

The user is unable to view the reputation of other users.

The user's report does not impact the other user's reputation.

The user's report is not received by the server.

The user's "stars" do not match the rating.

A Regular user is not changed into a Reputable user upon reaching 4.2 star rating.

- The User is not notified of the error.
- The system will automatically update the User's type.

Feature

Reputation (Admin)

Description

An extension of the Reputation feature that defines how the Admin user type is able to control user reputation.

Precondition

User type is Admin

Functional Requirements

1	The system must allow Admins to reset the rating of other users.
2	The system must allow Admins the ability to modify or delete user feedback.

Non-Functional Requirements

1	The system shall process modification requests within 7 seconds.
2	Users are notified about Admin modifications to their profile.

Successful Operation

Admin's actions of resetting ratings of users is validated.
Admin's ability to modify or delete user feedback is validated.
Users receive a notification about Admin activity on their account.

Failed Operation

User's ratings are not reset/ User's feedback is not modified or deleted.
- Admin is notified about the failed operation.
- Admin is able to retry their request.

User does not receive a notification
- User is not informed

Feature

Event Creation

Description
A derivative of the events feature, it defines how reputable users are able to create events.

Precondition
A User is of type: Reputable.   A User must be viewing the map.

Functional Requirements	
1	The system must allow users to create events by placing a pin on the map.
2	The system must allow the pin to have a title and a description so users can understand the purpose of the event.
3	The system must allow the pin creator to edit the title and description in case any typos or details have changed.
4	The system must allow the pin creator to toggle the display of attendance for an event.
5	The system must allow the event to be logged.

Non-Functional Requirements	
1	The system shall allow the title of pins a character limit of 30.
2	The system shall allow the description on pins a character word limit of 150.
3	The system shall place the pin on the map as soon as the event is created.
4	The system shall allow a user to post an event once every 7 days.
5	The feature must create an event 100 percent of the time.
6	The availability of the feature must be available 100 percent of the time that the application is functional.

Successful Operation
The Event pin is registered on the map.
The Event pin accurately displays title and description.

Failed Operation
The event is created but the pin does not appear on the map.
The event is created but lacks a title and description.
The event is edited by another user that is not the event creator.
The toggle for attendance does not update (turn on/ turn off).
A Regular type user is able to create an event.

Feature
Event Joining

Description
Defines how a user can join an event pin.

Precondition
A User is of type: Regular, Service, or Reputable. A User must be viewing a Group Event pin.

Functional Requirements	
1	The system must allow users to join an event.
2	The system must allow users to exit from an event.
3	The system must allow the user to view their joined events in their User profile.
4	The system must log the users joining and leaving events.

Non-Functional Requirements	
1	The system shall inform the user that they have joined the event.
2	The system shall take no longer than 7 seconds to process a user joining an event.
3	The system shall allow up to 100 users to join an event.

Successful Operation
A User is recorded as having joined the event client and Server side.

Failed Operation
The system took longer than 7 seconds to process a user joining an event.   ● Inform the user that the process has timed out.   ● Allow the user to attempt the join process again.   The system does not log joining or leaving events.   The system allows more than 100 users to join an event.   ● Users that join after the initial 100 are notified of their removal from the event.   ● The system removes users from being logged as participating in the event.

Feature
Service Creation

Description
Defines how a user is able to create a service to assist with clean-up

Precondition
A User is of type: Regular, Service, or Reputable.

Functional Requirements	
1	The system must allow users the ability to add their services to the application.
2	The system must allow the user to enter their company name, description, phone number, and website when adding their services.
3	The system must allow the user who created a service, the ability to remove the service.
4	The system must allow the Service user an option to limit services based upon distance from location or zip code.
5	The system must allow the Service user to select the types of pins they would provide services for.
6	The system must allow only 10,000 Service users.

Non-Functional Requirements	
1	The system shall accept a company name of 30 characters as a valid entry.
2	The system shall accept a description with a 200 word limit as a valid entry.
3	The system shall accept a phone number of 10 digits as a valid entry. ● All valid phone numbers will have a country code of +1.
4	The system shall accept any input as a valid link to a website. ● If the line is left empty, the system shall pass the website with "N/A".
5	The system shall have only one service per User account.

Successful Operation
The User successfully registers their service to the system.
The User is able to remove their service from the system.
The system accurately recorded what types of pins a Service user opted for.

Failed Operation
The Service user removes their service from the system but it stays in the list of services.
The user adds their service to the system but it does not appear in the list of services.
The service is available outside the distance stated by the user.
The service's entered company name is longer than 30 characters.
The service's entered description is longer than 200 words.
The Service User's account is not updated correctly, resulting in the User still being viewed as Regular by the system.

Feature
Service Requests

Description
Defines how users are able to request a service for their pin.

Precondition
A User is of type Reputable, Service, or Regular.
A User must have a created, active pin.

Functional Requirements

1	The system must allow users to view a list of cleaning services.
2	The system must allow users to request a service for their created pin.
3	The system must allow the user that requested a service to cancel said service.
4	The system must allow for the Service user to deny a request.
5	The system must allow the Service user to view their requests.
6	The system must allow the service to be recommended or not recommended by the user who requested the service.
7	The system shall inform the Service user that request is pending.

Non-Functional Requirements	
1	The system shall display a list of cleaning services within the limit of zip code or specified distance. ● Users may specify a distance up to 20 miles away to find cleaning services.
2	The system shall take no longer than 7 seconds to create a list of cleaning services.
3	The system shall inform the user that the request has been sent.
4	The system should send notifications no later than 0.4 seconds after a request is received by the server.

Successful Operation
The user is able to order a cleaning service to their pin.

Failed Operation
The service was unable to be processed.   ● Inform the user that the request was not processed. The system takes longer than 7 seconds to create a list of services. The system displays services further than specified by the user or more than 20 miles away.

Feature
Services (Admin)

Description
Defines how Admins are able to interact with service features.

Precondition
User type is Admin

Functional Requirements	
1	The system allows Admins to see which service a user recommended.
2	The system allows Admins to see the number of recommended services by a user.
3	The system allows Admins to modify or delete a service in case of the service no longer being used, the service is using malpractice, or service details contain vulgar content.

Non-Functional Requirements	
1	The system shall allow an Admin to sort services.

Successful Operation
Admin is able to modify a service.
Admin is able to delete a service
Admin is able to see the recommendation of a service.

Failed Operation
Admin is unable to delete a service.
Admin is unable to successfully modify a service.
Admin is unable to see the recommendation of a service.

Feature
Picture Upload

Description
Picture Upload allows users to upload photos to their pins, as well as store the images on their account.

Precondition
A User is of type: Regular, Service, or Reputable.
User is viewing their created pin.

Functional Requirements	
1	The system must allow users to upload an image on a pin.
2	The system must allow users to look at all the pictures they have uploaded to the system.
3	The system must allow the user to control read and write access.
4	The system must allow the user only to upload to their respective account.

Non-Functional Requirements	
1	The system shall consider only photos of type .Jpeg and .png as valid.
2	The system shall consider only photos that don't exceed 9 megabytes as valid.
3	The system shall display a pop-up notification displaying "Invalid Photo Type" when a photo with an invalid type is uploaded.
4	The system shall display a pop-up notification displaying "Photo size is too big" when a photo over 9 megabytes is uploaded.
5	The system shall display both invalid notifications when both criteria on photos have been flagged as invalid.
6	The system shall work optimally with more than 100 images.

Successful Operation
The user's image is uploaded to the pin and can be viewed by other users.

Failed Operation
A user is able to upload an image to a pin that is not theirs.
A user uploads an image that is not in .jpeg or .png format.
A user uploads an image larger than 9 megabytes.
A user is unable to access their images.

Feature
Picture Upload (Admin)

Description
Defines how Admins are able to interact with user's uploaded images.

Precondition
User type is Admin.

Functional Requirements	
1	The system must allow Admin to delete User's uploaded images in case of crude content.

Non-Functional Requirements	
1	The system shall display an uploaded image to Admin after at most 7 seconds.
2	The system shall work optimally with more than 100 images.

Successful Operation
Admin is able to delete a user's image.

Failed Operation
Admin is unable to delete images.
Image takes more than 7 seconds to load.
System slows due to too many images.

<u>Glossary</u>

Backend - Part of the computer system that deals with storing data. Synonym: Server Side

En-us - This annotation refers to the English language using United States colloquialism.

Legible - The text displayed can be easily read. Not jumbled

Zip code - In this document zip code refers to valid zip codes in California. Valid zip codes for California range from 90001 to 96162 [1].

<u>References</u>

[1]   "Zip-codes.com," *Listing of all Zip Codes in the state of California*. [Online]. Available: https://www.zip-codes.com/state/ca.asp. [Accessed: 29-Sep-2022].

[2]   *A guide to functional requirements (with examples)*. [Online]. Available: https://www.nuclino.com/articles/functional-requirements. [Accessed: 29-Sep-2022].

[3]   J. Weller, "Download free BRD templates," *Smartsheet*, 18-Nov-2021. [Online]. Available: https://www.smartsheet.com/content/business-requirement-document-templates. [Accessed: 01-Oct-2022].

[4]   "Non-functional requirements: Examples, types, how to approach," *AltexSoft*, 12-Feb-2020. [Online]. Available: https://www.altexsoft.com/blog/non-functional-requirements/. [Accessed: 01-Oct-2022].