**Team Big Data**

# U-tification

# Business Requirement Document

**Date: 9/30/2022**

**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 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. The requirements needed for our system to work would be our inclusion of a litter map, records of all data received from Users, the ability to manage events and contact between local cleaning services. The only stakeholders involved in this project would be our client, Vatanak Vong, and the development team, Team Big Data. 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

## General Success Criteria

- All text uses a standardized font.
- All text will use *en-us.*
- All text will be *legible.*
- All Features will be available using a web browser: Chrome 104.x (64 bit).
- A User must be Logged In.
- Users do not directly communicate with *Backend*.
- An Admin is able to access User data.
- Data is both accurately displayed on the Client side as well as Server side.
- Users exclusively have write access to their own profile or post.

## General Failure Criteria

- Text flagged as invalid: Not standardized font, Not in en-us, Illegible.
    - Error Handling:
        - No error message is displayed.
        - The website will reload itself.
- 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.
- User communicates directly with the Backend.

## User Types

- Admin
    - Highest level user.
    - Able to perform actions that other users cannot.
    - Not bound by reputation.
    - Have read access to all users' information and interactions.
    - Not able to create pins, create services, request services, or upload pictures.
    - Able to ban and unban user accounts.
    - Able to recover user accounts.
- Reputable
    - Middle level user is able to perform all actions the lowest level can.
    - Authority to create events and request services.
    - Reputation must be greater than or equal to 4.2 rating.
- Regular
    - Lowest level user.
    - Reputation is lower than 4.2 rating.

# Map

Precondition:

- User type is either: Admin, Reputable, or Regular.

Functional Requirements:

- The system must allow a map of California to be displayed.
- The system must allow the user the ability to move the map along the X-Axis and Y-Axis.
- The system must allow the user to view all of the currently active pins.
- The system must allow the user to filter the pins based on zip code as well as distance from the user's current position.
- The system must allow the user to adjust the zoom of the map.

Non-Functional Requirements:

- The map shall only have 75 pins appear on the map at any given time.
- The system shall have the map load no longer than 20 seconds.
- The system shall have the map reload if an error occurs, informing the user.
- The system shall have a map with streets and street names.
- The system shall have a map displaying parks.
- The system shall keep the user location to the respective user.
- The system shall maintain optimal performance with 10,000 visitors.
- The system shall have coordinates formatted as follows: 0.000000, 0.000000

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.

Failed Operation:

- The map takes longer than 20 seconds to load.
    - Inform the user that the map has timed out.
    - 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.
- The number of pins on screen exceeds the limit (75).
- The map does not display street names or parks.

# Pinning

Precondition:

- User type is either: Reputable or Regular.
- User is using the map.

Functional Requirements:

- The system must allow users to place a pin anywhere on the map.
- The system must allow users the ability to remove any pin that their respective account has placed.
- The system must allow users the ability to modify the details of any pin that their respective account has placed.
- The system must allow users to view all pins that have been placed.
- The system must allow the pin creator to implement an image.
- 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.
- The system must allow the user to see how many pins their account has placed and only theirs.
- The system must display the title and description of the event when the pin is clicked.
- The system must have a way to exit the details of a pin.
- The system must allow users to view the date, time, and location of created pins. Location details include zip code, global coordinates, and city.
- The system must allow users to mark a pin as completed, removing it from the map.
- The system must allow the user to view cleaned or deleted pins.
- The system must have the user confirm that they want to create a pin.
- The system must log all pins created and deleted.

Non-Functional Requirements:

- The system shall allow a user to create a pin once every 3 minutes.
- 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.

Success:

- A 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.

Failure:

- Pins are displayed inappropriately.
- A map is not viewable by a User.
- Pins are shown with an invalid color.
- Users are unable to post pins.
- Users are able to post pins more than once every 3 minutes.

## Pinning (Admin)

Precondition:

- User type is an Admin.

Functional Requirements:

- The system must allow Admin read access to all User’s pin data.
- The system must allow Admin the ability to modify or delete User’s pin data.

Non-Functional Requirements:

- The system shall take no longer than 7 seconds to delete User’s data.

Success:

- User’s data is removed from the Server side.

Failure:

- 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.

## Alerts

Precondition:

- User type is either: Reputable, or Regular.

Functional Requirements:

- The system allows users to accept notifications from entered events.
- The system allows users to accept notifications from pin drops by zip code.
- 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.
- The system allows users to ping other users when creating a pin, the ping only goes to those accepting notifications.
- The system must have a method for the user where they can view a list of notifications.

Non-Functional Requirements:

- The system shall accept valid zip codes only from California.
- All alert functions with the exception of Admin alerts are off by default.

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 receives a notification for a pin they did not accept notifications for.
- The pin creator’s notification never actually deploys.
- The list of notifications does not show the correct notifications.
- The user receives a notification for the wrong pin type.

## Alerts (Admin)

Precondition:

- User type is Admin.

Functional Requirements:

- The system allows Admins to send notifications to users.

Non-Functional Requirements:

- 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.

## Reputation

Precondition:

- User type is either: Regular or Reputable

Functional Requirements:

- The system must allow users to have a reputation rating from zero to five as a float type.
- The system must allow all users to view user reputation.
- The system must allow for reports to impact their reputation, either negatively or positively.
- The system must have a way to report other users, not the user's own profile.
- The system must allow for the user to view reports submitted by the respective user.
- The system must keep the reputation from being altered on the browser.
- The system must allow users to gain reputation by creating or marking pins as completed.
- The system must log all changes to the reputation.

Non-Functional Requirements:

- The system shall display the rating as a ratio out of five "stars" along with the float rating.
- The system shall recognize a reputable user as a user with a 4.2 rating.
- The system shall start every user with a rating of 2.0.
- The system shall adjust the users rating by sum score divided by the number of reviews. (sum score / number of reviews)

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.

## Reputation (Admin)

Precondition:

- User type is Admin

Functional Requirements:

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

Non-Functional Requirements:

- The system shall process modification requests instantly.
- 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 delete
    - Admin is notified about the failed operation.
    - Admin is able to retry their request.
- User does not receive a notification
    - User is not informed

# Event Creation

Precondition:

- User type is either: Reputable.
- User must be viewing the map.

Functional Requirements:

- The system must allow users to create events by placing a pin on the map.
- The system must allow the pin to have a title and a description.
- The system must allow the pin creator to edit the title and description
- The system must allow the pin creator to toggle the display of attendance for an event.
- The system must allow the event to be logged.

Non-Functional Requirements:

- The system shall allow the title of pins a character limit of 30.
- The system shall allow the description on pins a character word limit of 150.
- The system shall place the pin on the map as soon as the event is created.
- The system shall allow a user to post an event once every 7 days.

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.

## Event Joining

Precondition:

- User type is either: Reputable or Regular.
- User must be viewing a Group Event pin.

Functional Requirements:

- The system must allow users to join an event.
- The system must allow users to exit from an event.
- The system must allow the user to view their joined events.
- The system must log the users joining and leaving events.

Non-Functional Requirements:

- The system shall inform the user that they have joined the event.
- The system shall take no longer than 7 seconds to process a user joining 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.

## Service Creation

Precondition:

- User type is either: Reputable or Regular.

Functional Requirements:

- The system must allow users the ability to add their services to the application.
- The system must allow the user to enter their company name, description, phone number, and website when adding their services.
- The system must allow the user to enter a website link for their services.
- The system must allow the user who created a service, the ability to remove the service.
- The system must allow the service provider an option to limit services based upon distance from location or zip code.

Non-Functional Requirements:

- The system shall accept a company name of 30 characters as a valid entry.
- The system shall accept a description with a 200 word limit as a valid entry.
- 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.
- 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".

Successful Operation:

- The User successfully registers their service to the system.
- The User is able to remove their service from the system.

Failed Operation:

- The service provider 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.

## Service Requests

Precondition:

- User type is either: Reputable or Regular.
- The user must have a created, active pin.

Functional Requirements:

- The system must allow users to view a list of cleaning services.
- The system must allow users to request a service for their created pin.
- The system must allow the user that requested a service to cancel said service.
- The system must allow for the service provider to deny a request.
- The system must allow the service provider to view their requests.
- The system must allow the service to be recommended or not recommended by the user who requested the service.

Non-Functional Requirements:

- 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.
- The system shall take no longer than 7 seconds to create a list of cleaning services.
- The system shall inform the user that the request has been sent.
- The system shall inform the service provider that request is pending.

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.

## Services (Admin)

Precondition:

- User type is Admin

Functional Requirements:

- The system allows Admins to see which service a user recommended.
- The system allows Admins to see the number of recommended services by a user.
- The system allows Admins to modify or delete a service.

Non-Functional Requirements:

- 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.

## Picture Upload

Precondition:

- User type is either: Reputable, or Regular.
- User is viewing their created pin.

Functional Requirements:

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

Non-Functional Requirements:

- The system shall consider only photos of type .Jpeg and .png as valid.
- The system shall consider only photos that don't exceed 9 megabytes as valid.
- The system shall display a pop-up notification displaying "Invalid Photo Type" when a photo with an invalid type is uploaded.
- The system shall display a pop-up notification displaying "Photo size is too big" when a photo over 9 megabytes is uploaded.
- The system shall display both invalid notifications when both criteria on photos have been flagged as invalid.
- 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.

## Picture Upload (Admin)

Precondition:

- User type is Admin.

Functional Requirements:

- The system must allow Admin to delete User’s uploaded images.

Non-Functional Requirements:

- The system shall display an uploaded image to Admin after at most 7 seconds.
- 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.

# Glossary

***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

***Service Provider -*** A user that has registered themselves in the system as a service.

***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].

# References

[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].