SW Engineering CSC648-848-05 Spring 2023

BucketLyst

Team 1

Marie Shimizu[1], Arielle Riray[2], Aaron Kuo[3], Tommy Truong,
Kenneth Gee, Rabin Karki, Francis Quang

[1] Team Lead mshimizu4@sfsu.edu
[2] Front End Lead
[3] Back End Lead
Github Master

Milestone 1

Version	Date	Description Of Changes
2.0	5/24/2023	Update based on feedback
1.0	3/2/2023	First Draft

Table of Contents

Executive Summary

BucketLyst is a product designed to help users create and share lists of their favorite places to visit. People create lists of their favorite places to visit all the time, whether in the form of a locally saved notes app or a Google Maps list. This means that there is already a user base, and we aim to elevate their experience through a user-friendly interface that helps users organize, manage, and manipulate their lists easily. Whether you're planning a trip, just want to keep track of your favorite spots, or looking for a place to explore, BucketLyst is the perfect tool.

One of BucketLyst's main features is its ability to organize and manipulate lists. For example, users can use the diff checker to compare differences between their and their following lists and see how they've changed over time. They can also use tags to categorize their favorite places by type (e.g., restaurants, museums, parks) or by location (e.g., New York City, Paris, Tokyo). Moreover, users can filter and search for spots by tags, categories, cities, radius from a user's current location, etc. This allows users to find the perfect spot they are looking for!

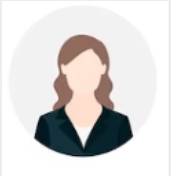
Another key feature of BucketLyst is allowing users to see public lists created by other users. Users can choose to make their lists public to everyone or share them with friends, making it easy to collaborate and plan trips together. Additionally, BucketLyst integrates with Google Maps, so users can import data from Google Maps, making it easy to get started with the app.

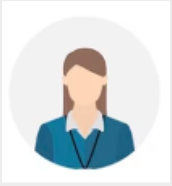
With these two main features combined, BucketLyst makes it easy to find the best spot, whether it has been recommended by best friends or favorite influencers. The app's emphasis on community-driven recommendations, sharing, and filtering sets it apart from other review sites or travel guides. This can be a great way to discover new places and get inspiration for your own travel plans from someone you trust.

Currently, there are several competitors that offer lists in conjunction with map integration. Google Maps is the primary competitor because of the size of their market capture. However, where Google Maps fails is in the lack of advertising for this useful and demanded functionality. Most users have little knowledge of creating lists on Google Maps. With proper advertising and deployment, we can capitalize on this and provide a service that addresses the lack of functionality from our competitors.

Main Use Cases

a. Persona Table

Name/ Occupation	Characteristics, Skills,	Pain points	Goals
Peter (Engineer) 	· Easygoing, spontaneous, and adventurous · Loves to explore good food and bars · Trusts his friends' recommendations · Easily forgets things · Not patient	· He forgets his friends' recommended places easily. · Usually, he goes places without plans. So when he asks his friends for their recommended places, they take a long time to respond.	· To view his options when and as needed. · To create a list easily so he can remember his places. · To quickly narrow down his options.
Alice (Student) 	· Detail-oriented, organized, and analytical · Busy with her studies and part-time jobs · Has a straightforward personality · Enjoys going to cafes and studying there · Likes to go to places where her favorite influencer goes · Not very tech-savvy	· She finds that creating lists on her iOS note is hard to navigate and share · The Tiktok or Instagram saved function can become messy and it's hard to make a list of the places her influencer goes	· To narrow down her options quickly · To easily share her lists with friends (Currently, she takes screenshots of her notes) · To have a seamless flow and not switch between three or four apps to get directions
Mary Jane (Digital Nomad) 	· A traveling enthusiast · Enjoys discovering new places and sharing her experiences with others · Tech-savvy	· Mary Jane has limited time in the city and wants to quickly find and organize her list of must-visit places.	· She wants to share her list with others to help them discover new places.

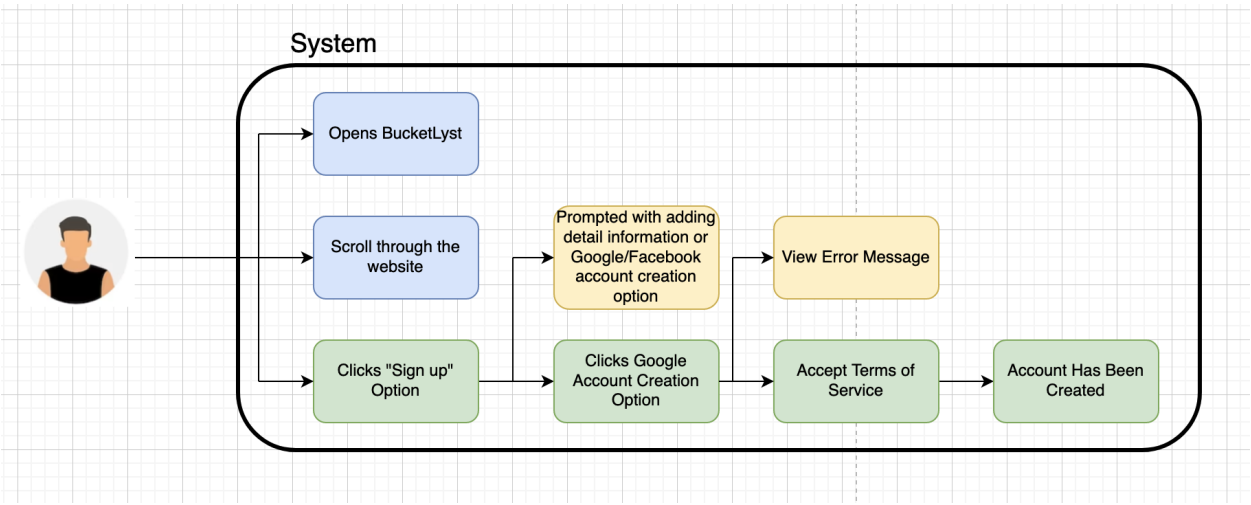
Drew (Influencer) 	· Loves to eat and share his honest reviews on his YouTube channel. · Very friendly and loves to share his favorite things with others. · Is a Google Maps power user but is looking for a social media outlet to share his favorite places on a centralized platform.	· He is looking for an app that is a central location for all things lists and locations. He can post his lists in his YouTube descriptions, but it would be better for his viewers if all his lists were in one place to view.	· Create a list in this app with his own reviews, notes, and labels so that his followers can find, sort, and follow his lists.
Lisa (Recruiter) 	· Outgoing, open to trying new places to eat · Resourceful in using social media · Excellent communication skills, detail-oriented, interpersonal skills, and time management	· Lisa works as a recruiter and maintains a list of recommended workplaces located in different cities. · With many workplaces, she needs to quickly find places to update or remove them from her list.	· Needs to be able to search within her existing list of companies to quickly find and update or remove places that are no longer relevant.
Olivia (A small business owner) 	· Talkative, enjoys engaging and having conversations with customers. · Observant, remembers certain information well for reference at a later time. · Uses resources to her advantage. · High risk-taker in terms of investment for high reward.	· Her business slowly began to operate at a net loss due to initially not having enough customers. · While being a high risk taker can yield high reward, it can also lead to high losses.	· Extend the reach and online presence of her restaurant business to be seen by more local people for a chance to increase her customers and profits.
Brandon (Traveler) 	· Has a well-paying remote job and isn't tied down to anyone or any place. · Because he can afford it, he travels to a lot of places around the world often. · Loves experiencing and learning about other cultures. · Very adventurous and takes lots of photos for himself to look back on.	· Forgetful at times, such as when obtaining contact information. · Usually waits for someone else to make the first move in initiating long-term connections. · Travels everywhere but always does so alone. Despite the thrills, he feels as if something is missing.	· He longs to find a meaningful long-term connection with others. · Wants to find a way to share the places he travels to with others and discover more places to visit. · Wants to befriend others who share common interests regarding traveling.

b. Main Use Cases Stories

Use case 1: Signing up

Actor: Peter

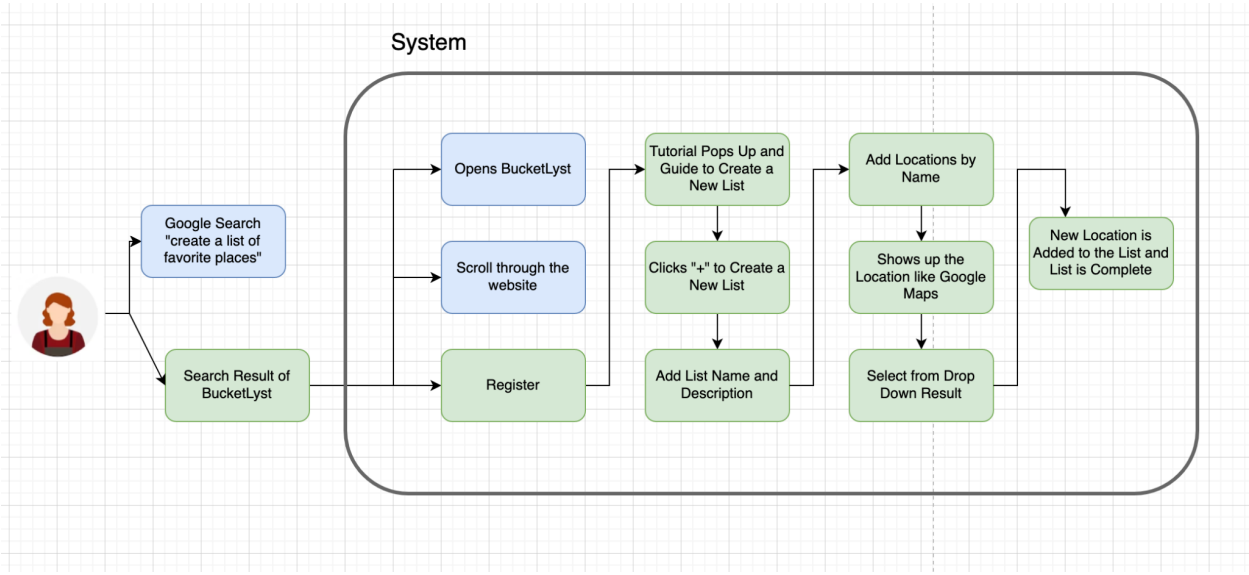
Peter loves to ask for recommendations on bars and restaurants from his friends, but he forgets their names too easily. His friend shows him BucketLyst, where he can create an account and have access to his friends' lists on the app when needed. His friend shares the URL, and he opens the website. He scrolls through the website and finds the "Sign Up" option at the top, which allows him to create an account and see his friends' lists. After clicking the "Sign Up" button, he is prompted to fill in details, including his first name, last name, and other information. He also has the option to create an account with Google or Facebook. Since he uses Gmail, he chooses to sign up using his Google account. However, he sees an error message because he has not yet agreed to the terms of service. After agreeing to the terms of service, he tries again, and the sign-up is successful.



Use case 2: Create a list

Actor: Alice

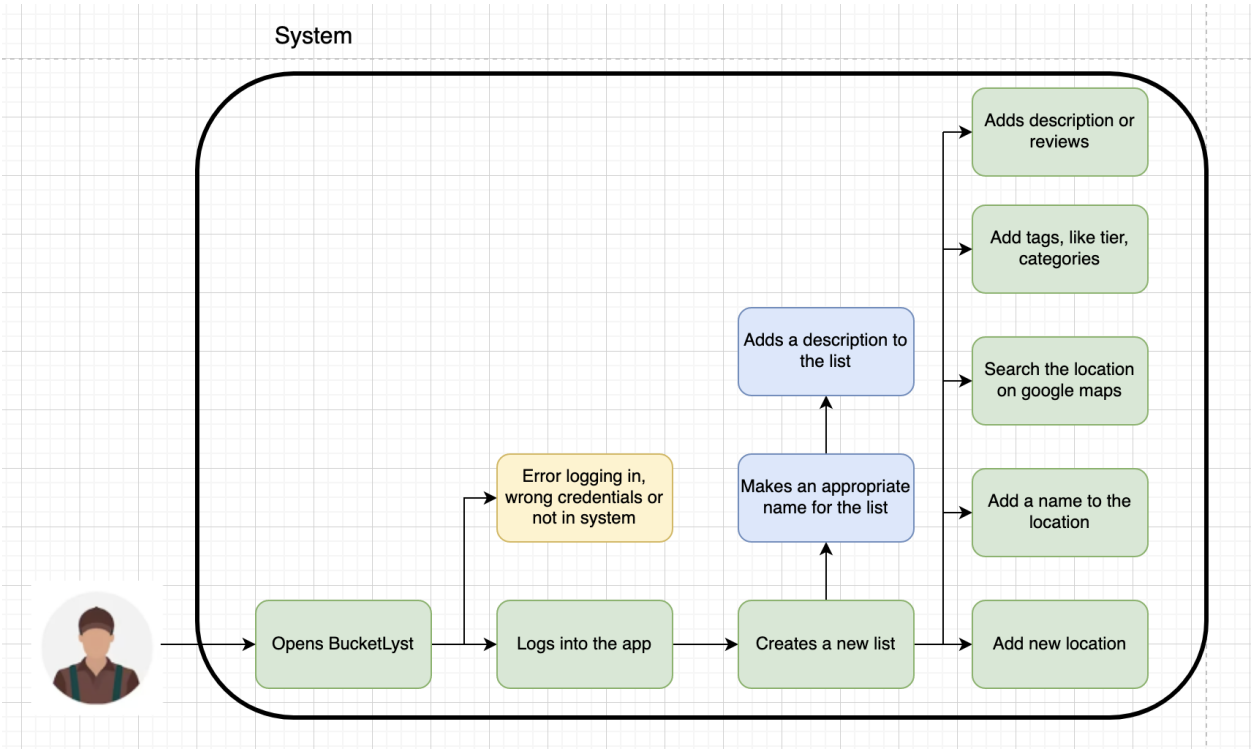
Alice wants to create a list to store her favorite cafes and the places her favorite influencer frequents more efficiently. She feels her iOS notes get too messy and unorganized. When she searches on Google for "how to create a list of favorite places", the second search result shows her BucketLyst with a short description: "Organize your favorite places with ease using our app. Create and customize lists, share them with friends, and filter and search through saved locations. Never forget a great spot again!" She is intrigued, and with only a few steps, she already has an account. Now she can create a list. BucketLyst pops up a tutorial to guide her in creating her first list. Alice follows the arrow and goes to "Lists," where it is empty, and the arrow appears again, pointing to the "+" sign at the top. She clicks it and is prompted to enter the name and description of the list. After creating the list, she is guided to add locations to the list, where she can search for the name and it will show the result, just like on Google Maps. She selects the result, and it auto-populates the rest of the information like the address and store hours. Now she has a list with her favorite cafes!



Use Case 3: Add Details on One Location

Actor: Drew

Drew is an adept user of Google Maps and is familiar with its lists feature. He's interested in migrating to BucketLyst because he just saw an ad for it and wants to use it as a central hub for all his lists, including his own reviews and labels for his followers to see. He logs into BucketLyst directly as he already has an account, and creates a new list by clicking on the appropriate button. He then adds an appropriate name and description for the list. Now, he can add a new location to the list by selecting the corresponding button and filling out the required fields, such as the name and the location search. Although the description and tags are optional, he fills them out for the convenience of his followers.



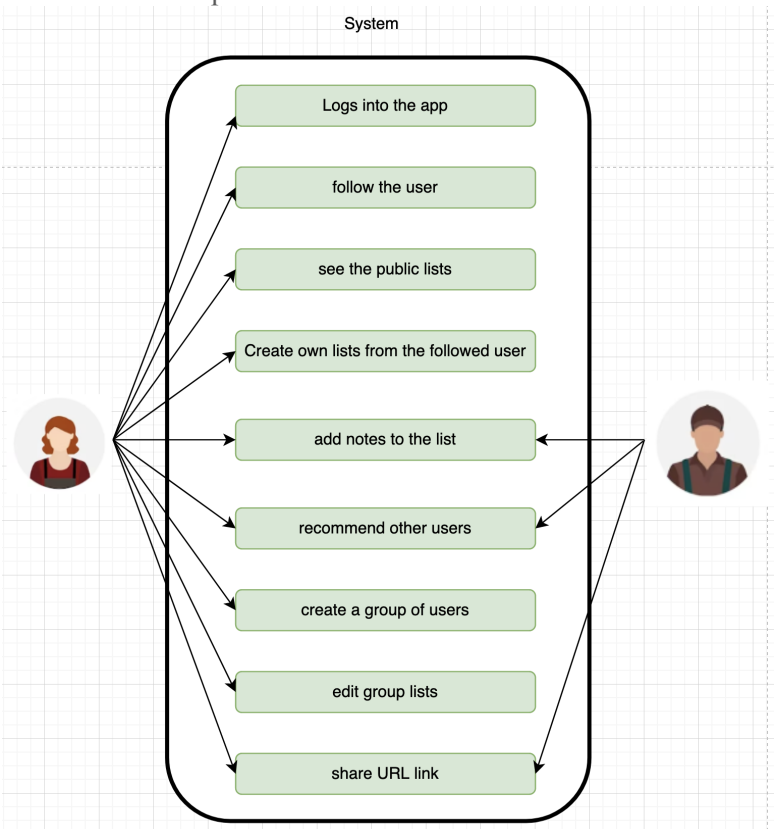
Use Case 4: Share a list

Actor: Drew, Alice

Drew is a renowned YouTuber who predominantly uploads content related to traveling and food reviews on his channel. He is very transparent with his fans and audience, and he shares his favorite things with them on YouTube. Since joining BucketLyst a couple of weeks ago, Drew cannot stop talking about the platform's amazing features. He was impressed by the idea of creating a list of favorite things in multiple categories. He created his own list of favorite things in his account. Moreover, he gained over 500 followers within a short period, and his followers provided constructive feedback on his lists through comments.

Alice has been Drew's biggest fan since high school, and she never misses any of his videos. She is an organized individual who loves traveling to different places and trying out new cuisines. In Drew's recent YouTube video, he talked about BucketLyst's interactive features, such as sending follow requests, commenting on lists, creating personalized lists, and creating groups to maintain lists among users. Drew is very social and aims to connect people through BucketLyst user profiles by sharing his favorite lists with others. As a result, he decided to provide his link in the description section of his YouTube channel.

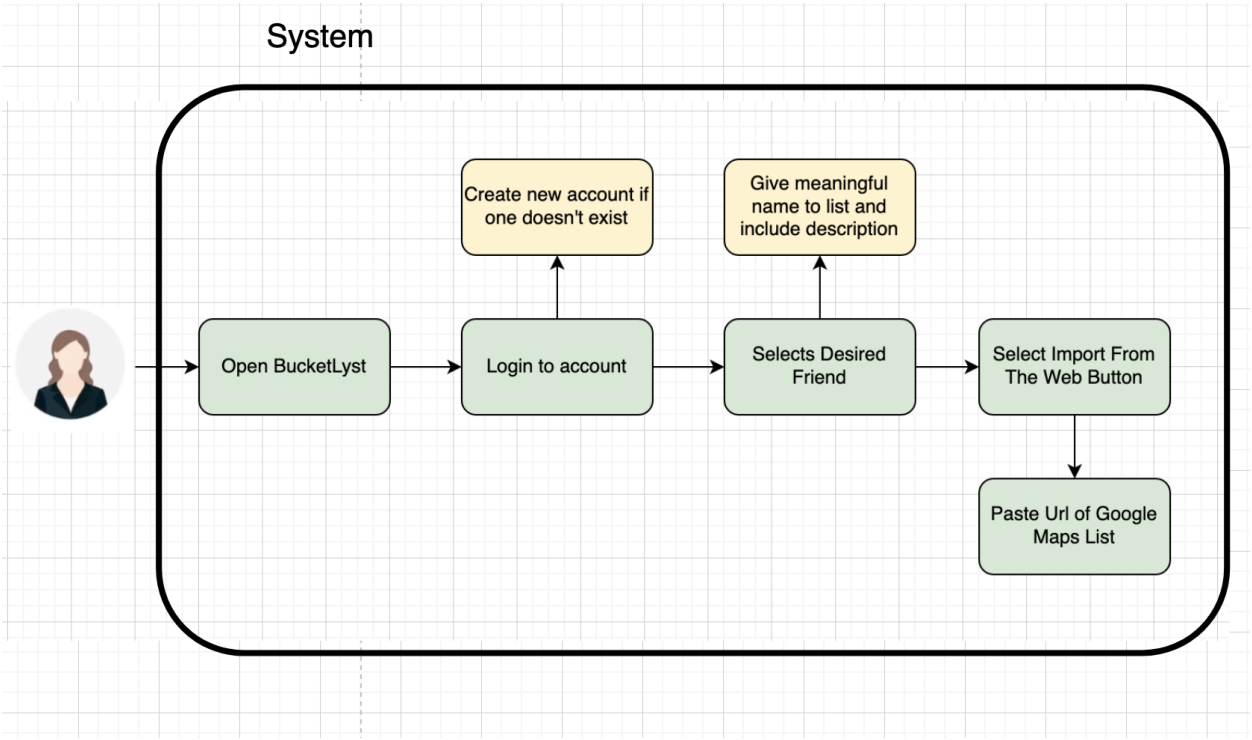
When Alice watched the video, she paused it in the middle and immediately followed her idol through the link from the description section. Since Drew is an influencer, he always makes his list public to his fans. When she visited the link, she signed into the app first before being directed to Drew's profile. Alice got excited about some of his lists, such as his favorite hiking places in SF. She followed him on BucketLyst and created her own list based on Drew's list. She also decided to share some of her lists with her group of friends, allowing others to edit her list with feedback in notes or descriptions.



Use Case 5: Import a List from Google Maps

Actor: Mary Jane

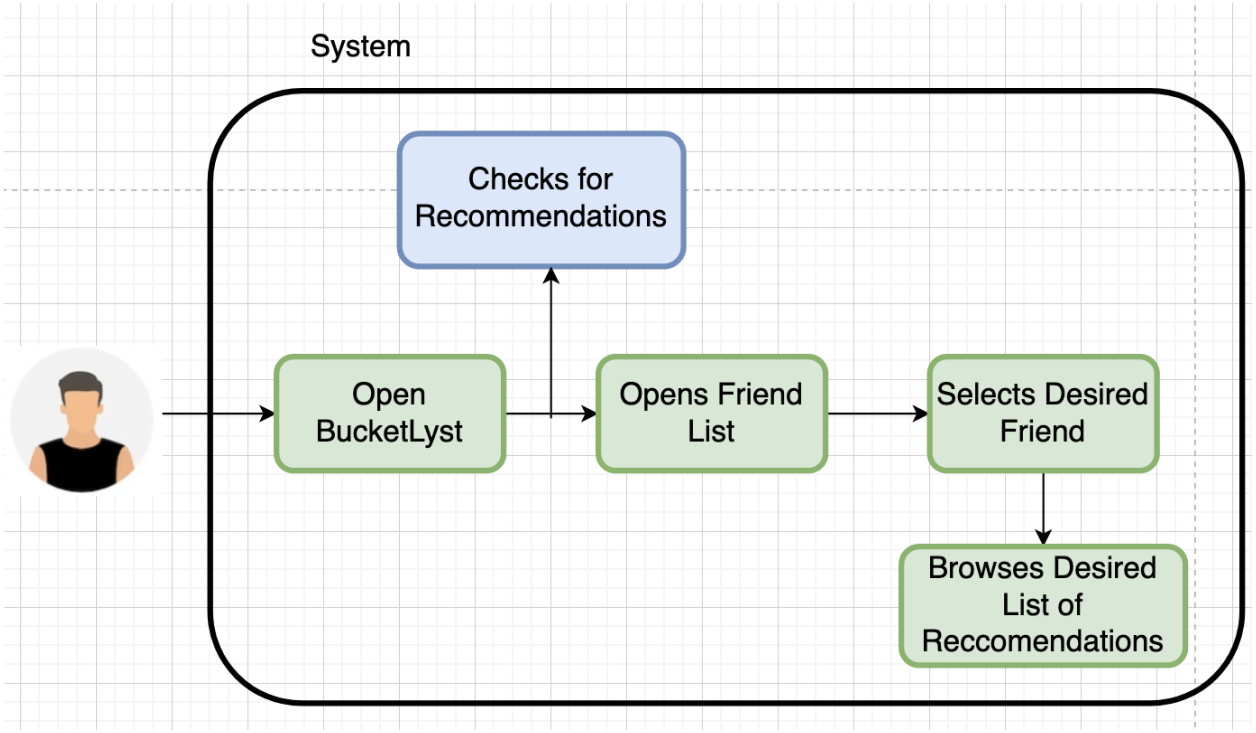
Mary Jane is a frequent traveler and digital nomad who creates extensive lists on Google Maps for every country or city she visits. However, she finds it challenging to navigate through her Google Maps list as it is one big list. She has to scroll repeatedly to find each item, making it tedious to plan out her trip. Her best friend, who shares the same name, Mary, tells her about BucketLyst and how the startup created an engaging and useful web app for solving this kind of issues. After learning about BucketLyst's powerful sorting and filtering tools, Mary Jane decides to import her lists from Google Maps. She navigates to BucketLyst and creates a new account. After logging in, she creates a new list with a meaningful name and clicks on the option to import from the web. Mary Jane then enters the URL of her Google Maps list, and now she has the Google Map list on BucketLyst with additional features to manipulate it!



Use Case 6: Exploring New Places from Friends' List

Actor: Peter

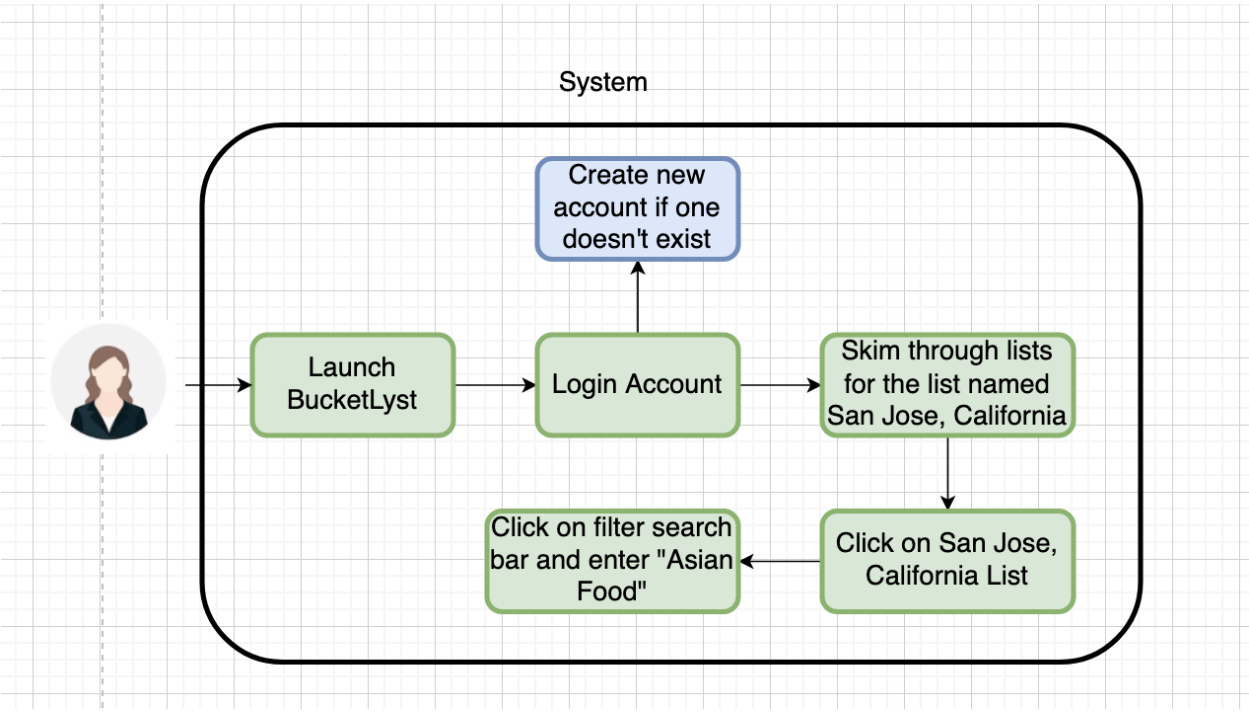
Peter is an adventurous and spontaneous person who seizes any opportunity to go out and explore. He loves discovering new places and trying out new restaurants with good food. One day, as Peter is searching for new places to try, he comes across an advertisement for BucketLyst and decides to give it a try. He tells his friends about BucketLyst and discovers that he can see their recommendations on the platform. Usually, Peter's friends are busy with work, so they may not be able to respond promptly to his requests for recommendations. However, with BucketLyst, he no longer has to wait for his friends to respond. He clicks on his friend's list and starts browsing through the recommendations that they have created.



Use Case 7: Search within a List with Criteria

Actor: Mary Jane

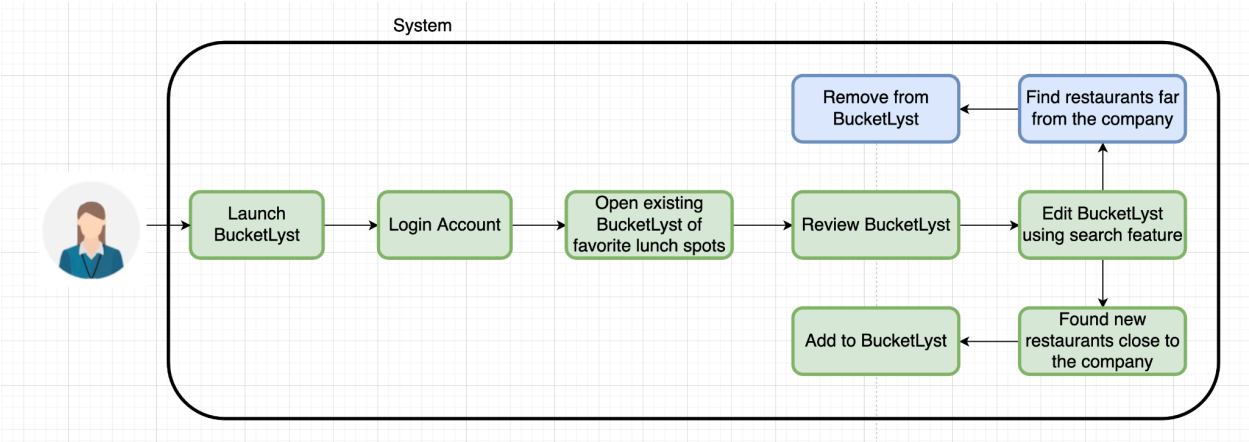
Mary Jane is on her way to Rio de Janeiro for a month to work remotely after the holidays. Before her trip, she had friends send her many recommendations that she accumulated over time. Her Google Maps list has over 90 locations! Before arriving, Mary Jane develops a strong craving for "Asian food," so she begins by importing her huge Google Maps list into BucketLyst. After importing her list, Mary Jane becomes elated at the idea of being able to search through her lists for some Asian food. She opens up BucketLyst and logs into her account. Once she is in, she sifts through her lists to find the one she labeled "San Jose, California." Once inside her list, she clicks on the filter search bar and enters "Asian food." Her original list of 99 spots is now reduced to 10 of her listed restaurants.



Use Case 8: Edit an Existing List

Actor: Lisa

Lisa is a recruiter who has a list of favorite lunch spots she likes to go to during her break. Recently, Lisa's company has been growing and becoming increasingly busy. Because of this, her main focus is time efficiency, so Lisa wants to update her list of favorite lunch spots by finding the ones closest to her office. She remembers some places that took too much time during her break, so she will use the search function to find those specific restaurants or if she does not remember, then she will filter by distance from location. If a restaurant is too far away, she will remove it from her BucketLyst. After removing some, she realizes the list is too small, and she wants to broaden her variety of choices. So, she plans to add new restaurants close to her company to her list.

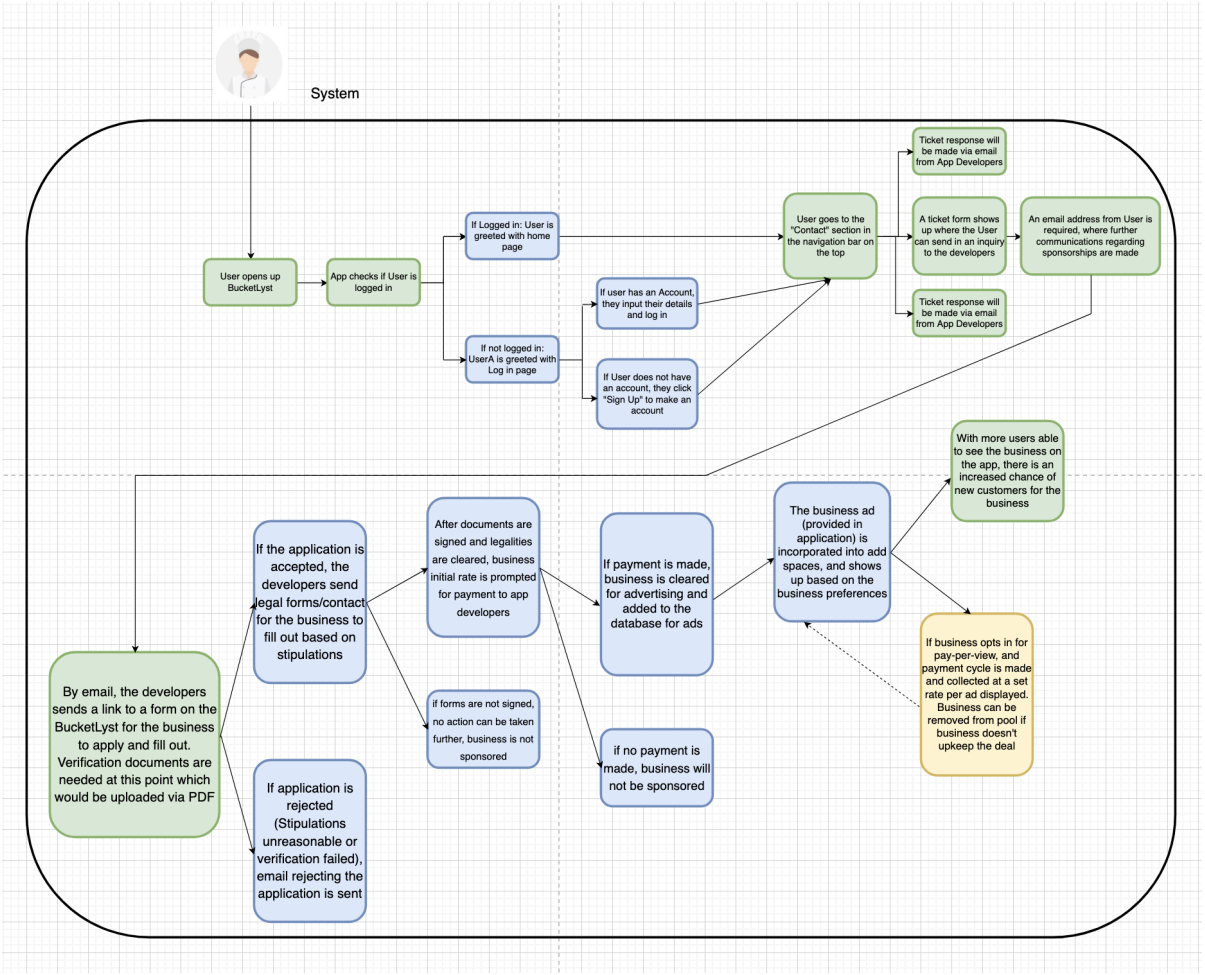


Use Case 9: Boost a Business Presence

Actor: Olivia

Olivia is the owner of a restaurant. One day, she walks in during the lunch hour and surveys the restaurant. Although the restaurant isn't completely full, there is a sizable number of patrons. However, Olivia feels that the remaining empty tables could be filled. She approaches the remaining customers and strikes up a conversation, asking how they found the restaurant. The customers show Olivia that they found the restaurant on the BucketLyst app, after being recommended it by their friends. Olivia is delighted to hear this and notices that there are sponsored results for other nearby businesses at the top of the results page. After business hours, Olivia thinks about her conversation with the customers and decides to get in contact with the BucketLyst app developers through the contact form on their app. After some negotiation via direct messages, verification through identification details, filling in legal forms, and signing a contract, Olivia and the app developers strike a deal to sponsor her restaurant. The sponsorship allows the restaurant to appear at the top of the search results page randomly for users within a 10-mile radius.

Overall, the BucketLyst app developers get paid for the sponsorship, users see verified recommendations for restaurants, and the restaurant receives a boost in customers. It's a mutually beneficial relationship where both the business and the app developers benefit.



Use Case 10: Gaining a follow on BucketLyst

Actor: Brandon (Traveler)

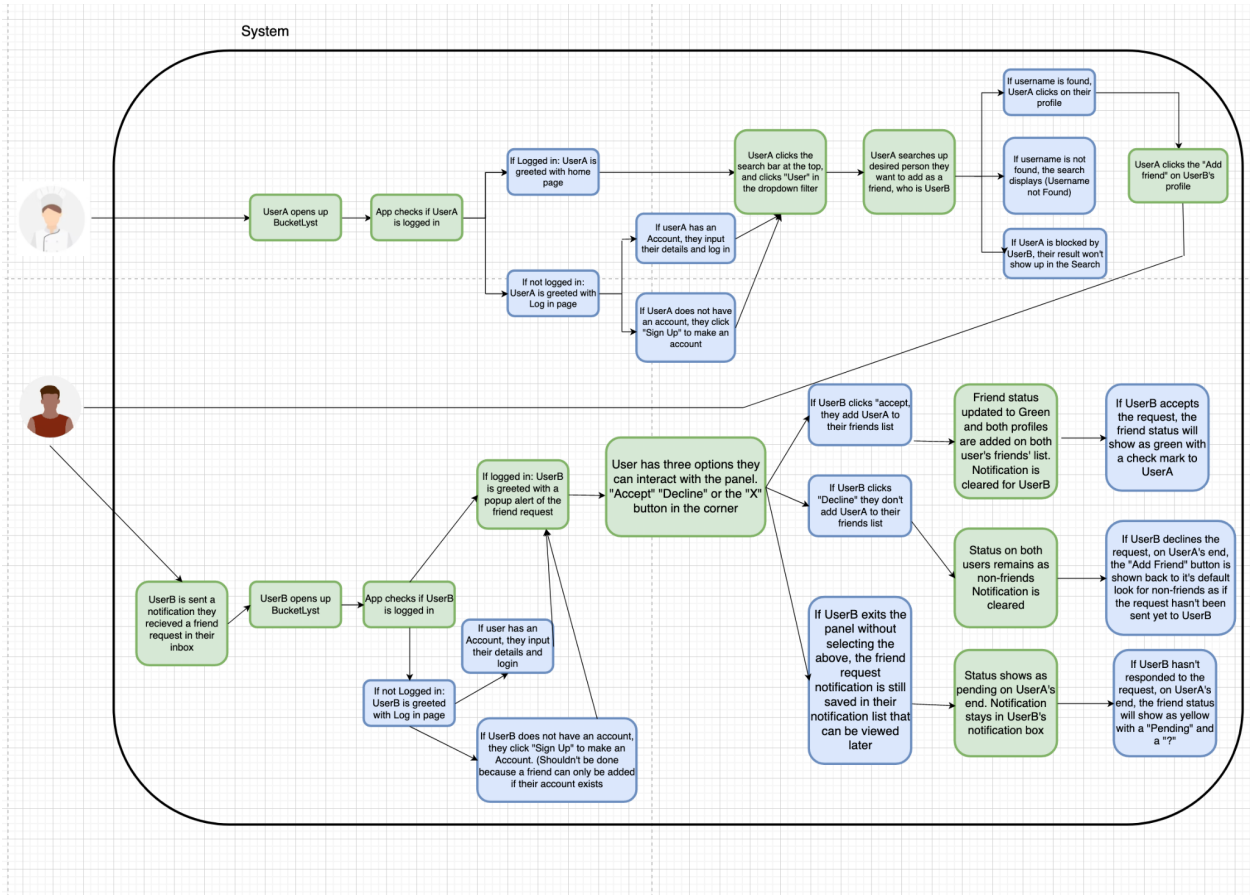
Enter Brandon, a well-off traveler living what some may call the "Dream-Life." Due to his high-paying remote job, he is free to venture throughout the world wherever he pleases at any time of the year (granted the place has Wi-Fi). Despite all the talks he'd have with the locals of the places he'd visit, they were only one-time conversations, and then he'd go off and never see them again. He'd learn the stories of hundreds of people but never got to tell his own. They were all bundled up in the camera roll of his phone, only there for his eyes to see. Everyone else seemed to have a home, but Brandon felt the opposite. He felt lost.

With all the places he has visited, it's quite overwhelming trying to recall which one was his favorite. Thankfully, he comes across the BucketLyst app, and he begins using it. Finally, a place where he can lay out the history/stories of his travels! As he's building up his list with places, he visits a beach in the San Francisco Bay Area. There on the beach, he meets a woman and has a well-meaning conversation that goes a bit deeper than he expected. She was interested in his travels, which was a pleasant surprise since he hasn't met anyone else who has done so. Then Brandon proceeds to show her his profile on the BucketLyst app and the list of places he has visited. It becomes very apparent to Brandon that he very much enjoyed her company. The woman looked to share the same sentiment as well. Unfortunately, they had to soon part ways.

Later that night, Brandon reflects on their encounter and realizes he never got her contact info or even her name. He finds himself feeling a deep sense of regret and sadness that he'd most likely never see her again, just like those hundreds of others he met on his travels. In the morning, Brandon wakes up to an unusual notification. Someone by the name of "Sandra" followed his profile on the app. He sees her profile picture, and that's when it hits him. It was her! The woman on the beach! Brandon had shown her his profile on the BucketLyst app, and because of that action, she was able to find him by searching up his name username in the search bar, visiting his profile, and sending him a friend request. Right now, it would be seen as "Pending" on her screen with a yellow-colored button. Brandon clicks the notification and is greeted by a prompt of "Accept" or "Decline." He clicks the green "Accept" button, and the app opens a page to her profile. There among the list of places were all sorts of stops in the Bay Area. However, at the top of her favorites list was one he had never been to before, "Sutra Baths."

Later that day, Brandon thinks about the place he saw on Sandra's list and decides to go for a visit at sunset. When he arrives, he's greeted by a beautiful sight of the rocks and ocean. Then from behind him, he is greeted by that unmistakable voice from the beach.

"I knew that putting this place on my list would lead to us meeting again."


System
UserA opens up BucketLyst
App checks if UserA is logged in
If Logged in: UserA is greeted with home page
If not logged in: UserA is greeted with Log in page
If userA has an Account, they input their details and log in
If UserA does not have an account, they click "Sign Up" to make an account
UserA clicks the search bar at the top, and clicks "User" in the dropdown filter
UserA searches up desired person they want to add as a friend, who is UserB
If username is found, UserA clicks on their profile
If username is not found, the search displays (Username not Found)
If UserA is blocked by UserB, their result won't show up in the Search
UserA clicks the "Add friend" on UserB's profile
UserB is sent a notification they recieved a friend request in their inbox
UserB opens up BucketLyst
App checks if UserB is logged in
If logged in: UserB is greeted with a popup alert of the friend request
If user has an Account, they input their details and login
If not Logged in: UserB is greeted with Log in page
If UserB does not have an account, they click "Sign Up" to make an Account. (Shouldn't be done because a friend can only be added if their account exists
User has three options they can interact with the panel. "Accept" "Decline" or the "X" button in the corner
If UserB clicks "accept, they add UserA to their friends list
If UserB clicks "Decline" they don't add UserA to their friends list
If UserB exits the panel without selecting the above, the friend request notification is still saved in their notification list that can be viewed later
Friend status updated to Green and both profiles are added on both user's friends' list. Notification is cleared for UserB
Status on both users remains as non-friends Notification is cleared
Status shows as pending on UserA's end. Notification stays in UserB's notification box
If UserB accepts the request, the friend status will show as green with a check mark to UserA
If UserB declines the request, on UserA's end, the "Add Friend" button is shown back to it's default look for non-friends as if the request hasn't been sent yet to UserB
If UserB hasn't responded to the request, on UserA's end, the friend status will show as yellow with a "Pending" and a "?"

Main Data Items and Entries

1. User: A person who uses BucketLyst to find, save and share their favorite places.

2. List: A collection of places that a user has saved to organize and keep track of their favorite places

3. Followers List: A collection of followers (also users) a user a has

4. Following List: A collection of accounts a user (also users) a follows

5. Block List: A collection of blocked users if a user does not wish to interact with them

6. Location: A place such as a restaurant, hotel, attraction, or bar that a user can add to their list

7. Location Data Object (LDO): the object which contains the geogtaphic data of the location such as google map links

8. Notification: A message that alerts a user of activity related to their account, such as a new comment or follower.

9. Business User: a business user who wish to promote their business on BucketLyst

Functional Requirements

1. Regular User
    a. User should create their username
    b. User should enter their email
    c. User should create a password
    d. User can create a list
    e. User can manage a list
    f. User can own multiple lists
    g. User can name a list
    h. User can add tags
    i. User can edit lists
    j. User can edit accessible location data fields
    k. User can edit user profile
    l. User can edit notes on the lists
    m. User can edit their account details
    n. User can add their followers
    o. User can add additional informations to profile
    p. User can add notes on the lists
    q. User can add a location
    r. User can delete lists
    s. User can delete notes on the lists
    t. User can mute their followings
    u. User can delete their account
    v. User can delete location from their list
    w. User can share their list with others with link
    x. User can set their lists to private
    y. User can set their list to public so followers can view
    z. Users can own shared lists as a group so that everyone can edit them
    aa. User can copy other’s list to make it their own list
    bb. User can search for locations by tags
    cc. User can search for locations by categories
    dd. User can search for locations by radius from the user’s current location
    ee. User can search for locations by open now
    ff. User can search other users on the platform
    gg. User can follow other users
    hh. Users can see other’s profiles
    ii. User can send a follower request to other users
    jj. User can accept follower request
    kk. User can deny follower request
    ll. User can remove lists

mm. User can remove followers
nn. User can remove their following list
oo. User can sign out
pp. User can temporarily disable their account
qq. User can mute their followings
rr. User can manage privacy setting such as making user profile public or private
ss. Users can view recommendations from other users.
tt. User can view recommendations from algorithm (Based on their lists)
uu. Users can rate a place on a scale of 1-5 stars. (Tier)
vv. Users can post a comment on a place to share their thoughts or feedback on each location
ww. Users can be notified if there is any updates or new activities
xx. Users can view notifications
yy. Users can turn off notifications
zz. Users can turn on notifications
aaa. User can be notified via email
bbb. Users get notified for "trending among your followings" spots
ccc. Users can report abuse, such as inappropriate comments or content.
ddd. User can explore lists through the recommendations
eee. Users can turn off notifications
fff. Users can view a map that displays the location of their saved places and recommendations.
ggg. Users can import their existing lists from Google Maps
hhh. User can compare their list with selected list and see the difference

2. Business User
    a. Business should enter their company name
    b. Business should enter their email
    c. Business should create a password
    d. Business can sign out
    e. Business can temporarily disable their account
    f. Business can mute their followings
    g. Business can manage privacy setting such as making user profile public or private
    h. Business can search other users on the platform
    i. Business can be notified if there is any updates or new activities
    j. Business can view notifications
    k. Business can turn off notifications
    l. Business can turn on notifications
    m. Business can be notified via email
    n. Businesses can promote their business on the platform

3. Location:
    a. Location contains google map link that allow user to jump to the location on google map seamlessly
    b. Location has “Navigation” that allow users to open google map navigation

Nonfunctional Requirements

1. Performance (assuming latency[1] is less than 1000 ms)
    a. Page loading time: less than 5000 ms
    b. Search bar loading time: less than 10000 ms
    c. Log-in verification time: less than 5000 ms
    d. List loading time: less than 10000 ms

2. Expected load
    a. No more than 100 active users
    b. No more than 10000 unique location entries
    c. No more than 1000 lists created
    d. No more than 100 locations in a given list

3. Fault tolerance
    a. Server downtime: no more than 5 business days

4. Security requirements
    a. HTTPS/2 protocol
        i. Unexpired SSL/TLS certificate
    b. User information won't be willingly distributed
    c. Minimum 8 character alphanumeric password with at least one special character
    d. Account lock after certain number of login attempts

5. Availability
    a. During working hours on business days
    b. The system should be available 24/7, with minimal downtime for maintenance and upgrades.

6. Storage
    a. No more than 10000 GB on server storage
    b. No more than 10000 GB on database storage
    c. No more than 10 GB of client RAM
    d. No more than 10 GB of client storage
    e. The file size of images or media uploaded by users shall not exceed 2 MB.

7. User Privacy
    a. Application will not collect more data than what the user explicitly provided

   b. Application will collect user data including name, email, and list of saved places. The data shall be used only for improving the user experience and shall not be shared with any third parties without explicit user consent.

8. Compatibility
   a. The platform should be compatible with all major web browsers, including Chrome, Firefox, and Safari, as well as mobile device browsers running iOS and Android.

9. Accessibility
   a. The platform should be accessible to users with disabilities

10. Usability
    a. The platform should be easy to use and intuitive, with a simple and clear user interface that is accessible to all users regardless of technical expertise.
    b. Users with basic computer skills, after 15 minutes of training, shall complete the task of creating and sharing a list within 5 minutes with no more than 1 error.

11. Scalability:
    a. The platform should be designed to scale easily as the user base grows, with the ability to add new features and functionality without affecting the performance of the system.

[1]: Latency is defined as the time from just before sending the request to just after receiving the first part of the response.

Competitive Analysis

a. Strength and Weaknesses Table

	Strengths	Weaknesses	Pricing	Social Media	Onboarding experience
Google Maps	+ Can directly see location on map + Can create and see ratings + Can create list and add locations + Can share lists	- Cannot manipulate or sort list - Cannot compare between users	Free	All popular avenues	Intuitive, but no guide or tutorial
Apple Maps	+ Can directly see location on map + Can create and see ratings	- Cannot make lists	Free	All popular avenues	Intuitive, but no guide or tutorial
Yelp	+ Can directly see location on map + Can create and see ratings + Can create lists	- Cannot manipulate lists - Cannot share to specific people - Does not have all the spots like google maps	Free	All popular avenues	Intuitive, but no guide or tutorial
Note apps (ie: iOS notes, notepad)	+ Can add location easily	- Cannot easily share list (software specific) - Cannot view location easily - Cannot manipulate or sort list - Cannot compare between users - Does not have map view	Free	None	Intuitive, but no guide or tutorial
Trip Advisor	+ Can directly see location on map + Can create and see ratings	- Cannot make lists	Free	All popular avenues	Intuitive, but no guide or tutorial
Expedia	+ Can search by location easily + Recommend new places for you	- Cannot make lists	Free	All popular avenues	Intuitive, but no guide or tutorial

b. Feature Analysis Table

Features	Google Maps	Apple Maps	Yelp	Locally stored notes (ie: iOS notes, notepad)	TripAdvisor	Expedia	BucketLyst
View location on map	++	++	++	-	+	+	++
Create lists	+	-	++	+	-	-	++
Search in the lists	-	-	+	+	-	-	++
View friends' lists	+	-	+	-	-	-	++
Share lists to specific users in app	++	-	-	+	-	-	++
Customizability of list	-	-	+	+	-	-	++
Reviews	++	++	++	-	++	+	++
Recommendations	+	-	+	-	++	++	++
Diff checker	-	-	-	-	-	-	++

Does not exist: - Exist: + Superior: ++

c. Competitive Analysis Summary

Currently, the market is composed of established companies that have integrated list creation into their service offerings, placing emphasis on their core competencies. Notably, Yelp and Google Maps have achieved prominence through their respective review system and location display features. In addition to our own core capabilities, we intend to enhance and adopt certain attributes of our competitors. Specifically, we recognize Yelp's Collections as a dominant name brand in the market, which we compete with through our own feature set: list manipulation, customized tags, and detailed descriptions. Our approach provides greater flexibility in sorting, organizing, and presenting content to our users' followers. Furthermore, we distinguish ourselves from Yelp Collections by incorporating Google Maps' ability to showcase all locations from a list on a map. We aim to harmonize two dominant services while preserving optimal user experience.

High-Level System Architecture and Technologies Used

1. Front end technologies
    a. React.js
        i. render website
    b. Next.js
        i. Server side rendering (SSR)
2. Back end technologies via AWS amplify CLI tool chain
    a. Amazon Route 53
        i. Manage DNS
    b. AWS Certificate Manager
        i. Manage SSL certificates
    c. Amazon CloudFront
        i. Content delivery network for fast global delivery of sites
    d. Amazon API Gateway
        i. Communication with front end (handling SSR)
    e. Amazon S3
        i. Stores front end assets
    f. Amazon Lambda
        i. Compute platform for back end
    g. Node.js
3. Database system
    a. AWS aurora - graphql api
4. Version Control
    a. Git
5. Code editor
    a. Visual Studio Code
    b. IntelliJ
6. API integration
    a. Google Map API

Checklist

	Task	Status	Notes
✅	Team found a time slot to meet outside of the class	Done	Every Friday at 7 PM
✅	Github master chosen	Done	Tommy
✅	Team decided and agreed together on using the listed SW tools and deployment server	Done	Techstack Approved
✅	Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing	Done	
✅	Team lead ensured that all team members read the final M1 and agree/understand it before submission	Done	
✅	Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)	Done	

Team Contributions

Name	Role	Contribution
Marie Shimizu	Team Lead	Checkpoint 1: - Created team Jira board (project management tool) and invited teammate - Collected the draft from everyone and put them into one using the outline Aaron made - Give feedback to the teammates on each chapters when needed - Created template on main use cases and persona table - Created table of contents - Main use cases 1 and 2, persona table Peter and Alice - Facilitate meetings according to agenda - Divide up and assign each chapter to the teammates - Proofread everyone's contribution and corrected grammar when needed - Proofread whole M1 document - Communicate with CTO and submit the finalized work - Check and give feedback to each tasks when needed Checkpoint 2: - Set up github private repo, and invited all the teammates - Organize project ideas and create polls to vote - Organize the decision making process on techstack - Email professor for techstack approval - Created AWS account share it with team - Divide up work and assign each work to the teammates - Filled out Github credential detail - Check and give feedback to each tasks when needed
Arielle Riray Score: 10/10	Frontend Lead	Checkpoint 1: - Created the 'About Me' Page templates prototype in pure HTML where group members can pull, insert their About Me data, and push back for me to use in later version - Created Use Case #9: Olivia: Business Owner Promotion - Created Use Cae #10: Brandon: Adding someone - Made 14 Functional Requirements (Removed if there's duplicate) - Made 11 Non-Functional Requirements (Removed if there's duplicate) Checkpoint 2: - Main Task: Created the 'About Me' Page for deployment - Used Node.js and create-react-app command to lay down groundwork for the app - Used React to create a simple About Me page (Un-Stylized for second prototype, but is in form of a table) - Tested it works locally, communicated with backend for stipulations in development and they helped successfully deploy it. - Revise the About Me Page to make it look more presentable (CSS added and React code reworked) - Helped revise the Main Data section and added on some data requirements regarding cardinality

		- Created two Use Case diagrams for #9 and #10 - Designed “about me” page - Created logo for the project - Leading frontend team
Francis Quang Score: 10/10	Backend Lead	Checkpoint 1: - Contribute to being scrum master. - Create issues from Tommy’s action items and assign to team members via jira - Edit functional requirements - Contribute name idea - Main use case #4: Lisa and her persona table - Demonstrated how to use Jira board with others Checkpoint 2: - Learned how to set up an AWS Amplify server with Aaron and Kenneth - Create IAM users for each teammate to login on AWS console - Leading backend team
Tommy Truong Score: 9/10	Github Master	Checkpoint 1: - Created weekly agenda for the team including notes and action items - Generated all zoom meetings as well as sharing recorded meetings with missing team mate - Consistently reminded the team to attend meetings or turn in assignments by Discord and text - Created an ALL LINKS tab listing out every useful link as well as coloring the individual tabs - Generated the most names for the app (Also team#1 name) - Github master is my duty but also a contribution - Provided a couple templates to help team members write their “about me” - Contributed to competitive analysis summary Checkpoint 2: - Fixed everyone’s diagrams to make everything cohesive
Kenneth Gee Score: 8/10	Backend Engineer	Checkpoint 1: - Created a draft version of checkpoint 1 - Finalized & contributed to non-functional requirements - Added main use case #6 with diagram Checkpoint 2: - Learned to setup AWS Amplify with Aaron and Francis
Rabin Karki Score: 10/10	Database Master	Checkpoint 1: - Actively participated in team’s meetings - Discussed about the project ideas and tech stack - Created a draft version of checkpoint 1 - Finalized the functional requirement section from the draft - Added one main use case with the diagram - Learned to use Jira Software

		Checkpoint 2: - Installed MySQL Workbench - Created a database instance on AWS using Amazon Aurora engine - Launched an EC2 instance to serve as a bastion host - Used SSH tunneling to securely connect to the Aurora Database instance from MySQL Workbench - Created a new security group called “BucketLystSecurity” for the instance which included inbound rules for SSH and MySQL traffic - Made a 11-page pdf file, and shared with the team on Discord about how to connect DB instances with the necessary steps and requirements.
Aaron Kuo Score: 10/10	Document Editor/ Frontend Engineer	Checkpoint 1: - Executive Summary - High-Level Systems Architecture and Technologies Used - Contributed to functional requirements - Contributed to non-functional requirements - Contributed to name and project idea discussion - Created M1 document outline - Created M1 document .md file on GitHub for contributions - Main Use Case #3 - Main Use Cases Persona table: Drew - Created a draft version of checkpoint 1 - Created both competitive analysis Strengths weakness table, feature table and competitive analysis summary Checkpoint 2: - Researched and proposed tech stack - Connected application repository to AWS Amplify - Researched project management tools - Initialized and demoed the Jira board - Setting up Amplify project - Organized meeting to corroborate backend & frontend