Vikram Ramesh (Bronco ID: 017319549)

CS 5800 - Advanced Software Engineering

Homework 2 - CPP Maps (OOA & OOD)

Part 1 - Use Case (OOA)

Use Case 1: Navigation (Get directions from a starting location to a destination location)

Title: Navigation
Primary Actor: User
Success Scenario:

1. User enters a starting location.
2. User enters a destination location.
3. System calculates the best route.
4. System displays navigation instructions.
5. System re-routes dynamically based on traffic or roadblocks.
6. User follows directions to reach the destination.

Extensions:

3A. No valid route available due to traffic or roadblocks.

Scenarios:

1. System failure: Who takes care of system failures during navigation?
2. Analytics: Who performs analytics on navigation patterns (e.g., to improve route suggestions)?

Preconditions:

1. The user has a device with GPS and location services turned on.
2. Starting and destination locations are valid and accessible.

Postconditions:

1. The user reaches the destination or reroutes successfully.

Secondary Actors:

1. **GPS System**: Provides real-time location data.
2. **Traffic Data System**: Feeds traffic updates into the navigation system.
3. **Database**: Stores geographic and map data, providing routes.
4. **System admin**: Updates and maintains database.

Stakeholders:

1. Users who need to navigate from one point to another.
2. Businesses relying on location-based services.

Scope: Navigation function in CPP Maps application.
Priority: High
Owner: CPP Maps Developer (Vikram Ramesh)

Use Case 2: Search (Search for a place by name such as Nordstrom, Disneyland...etc)

Title: Search

Primary Actor: User

Success Scenario:

1. User enters the name of the place (e.g., Nordstrom, Disneyland).
2. System matches the entered name with stored locations.
3. System displays a list of matching results.
4. User selects the desired location..
5. System shows the location on the map with a location pointer.

Extensions:

2A. No matches found. System suggests alternate search options.

2B. System provides auto-complete suggestions as the user types.

Scenarios:

1. System failure: What happens if a search fails due to a backend issue?
2. Analytics: Who monitors search patterns to improve search results?
3. Security: Who manages security to prevent data breaches during searches?

Preconditions:

1. The system has the latest database of locations.
2. The user has an active internet connection.

Postconditions:

1. The user finds and selects the desired place.

Secondary Actors:

1. **Search Server**: Processes user search queries.
2. **Database**: Retrieves matching places.
3. **System admin**: Updates and maintains database.

Stakeholders:

1. Users searching for places to visit or get directions to.
2. Businesses and landmarks included in the database.

Scope: Search function in the CPP Maps application.

Priority: High

Owner: CPP Maps Developer (Vikram Ramesh)

Use Case 3: Explore (Search for a place by category such as Restaurant, Gym, University...etc)

Title: Explore

Primary Actor: User

Success Scenario:

1. User selects a category (e.g., Restaurant, Gym, University).
2. System retrieves places matching the selected category.
3. System displays the places on the map.

4. User selects a place to view details (e.g., ratings, reviews).
5. System shows place details.

Extensions:

2A. No places found in the selected category.

Scenarios:

1. System failure: Who is responsible if the system fails to retrieve the category data?

Preconditions:

1. The system has categorized locations available.
2. The user is in a location that has the selected category.

Postconditions:

1. The user explores and selects their interested place.

Secondary Actors:

1. **System admin**: Updates categories and maintains database.
2. **Database**: Retrieves categorized locations.

Stakeholders:

1. Users looking for recommendations based on category.
2. Businesses listed under various categories.

Scope: Explore function in the CPP Maps application.

Priority: High

Owner: CPP Maps Developer (Vikram Ramesh)

Use Case 4: Share Location (Send location to a friend or family using messaging app on phone)

Title: Share Location

Primary Actor: User

Success Scenario:

1. User selects a location on the map.
2. User selects the "Share" option.
3. System offers options to share using messaging apps on the phone.
4. User selects a messaging app and a contact.
5. System sends the location to the selected contact.

Extensions:

3A. User doesn't have a messaging app installed.

3B. System suggests other sharing methods like email, social media, etc.,

Scenarios:

1. Security checks: Who ensures the secure sharing of locations over messaging services?
2. System Failure: Who is responsible if the messaging service fails?

Preconditions:

1. The user has a messaging app installed and internet access.
2. The location is a valid address.

Postconditions:

1. The contact receives the location information.

Secondary Actors:

1. **Messaging App providers**: Whatsapp, Messages, etc.,
2. **Messaging API:** Handles the communication between the map app and messaging services.
3. **Database**: Stores location data.
4. **System Admin:** Monitors data sharing and secures transmission.

Stakeholders:

1. Users sharing locations with others.
2. Messaging app providers.

Scope: Share location function in the CPP Maps application.

Priority: Low

Owner: CPP Maps Developer (Vikram Ramesh)

Use Case 5: Share Rideshare (Send location to rideshare app such as Uber or Lyft)

Title: Share Rideshare

Primary Actor: User

Success Scenario:

1. User selects a location on the map.
2. User selects the "Share Rideshare" option.
3. System gives options to share rideshare apps like Uber or Lyft.
4. User selects a rideshare app.
5. Rideshare app uses the shared location as the destination for the ride.

Extensions:

3A. The rideshare app is not installed on the user's phone.

3B. System suggests downloading the app from the app store.

Scenarios:

1. System failure: Who resolves if rideshare API communication fails?

Preconditions:

1. The user has an installed rideshare app and internet access.
2. The location is a valid address.

Postconditions:

1. The user successfully shares the location with the rideshare app.

Secondary Actors:

1. **Rideshare service providers:** Uber, Lyft, etc.,
2. **Rideshare API**: Communicates with rideshare services
3. **Database**: Stores shared location data.
4. **System Admin:** Ensures correct pick-up/drop-off points are shared.

Stakeholders:

1. Users booking rides.
2. Rideshare service providers.

Scope: Share Rideshare function in the CPP Maps application.

Priority: Medium

Owner: CPP Maps Developer (Vikram Ramesh)

Part 2 - (OOA)

Tasks:

a) Highlight all the nouns to determine possible potential classes.
b) Highlight all the verbs and verb phrases to determine possible methods and relationships.
c) Create a rough graph showing how all the classes are connected.

Solution:

a, b)

Use Case 1: Navigation

The user enters a starting location and a destination location. The system calculates the best route using the GPS system and traffic data system. The system displays navigation instructions and dynamically re-routes if there are traffic or roadblocks. The user follows the directions to reach the destination.

Use Case 2: Search

The user enters the name of the place they are looking for in the search bar. The system matches the entered name with stored locations in the database and displays a list of matching results. The user selects a location, and the system shows the location on the map with a location pointer.

Use Case 3: Explore

The user selects a category (e.g., restaurant, gym, university) in the system. The system retrieves places matching the selected category from the database and shows them on the map. The user selects a place and views the details (e.g., ratings, reviews).

Use Case 4: Share Location

The user selects a location on the map and clicks the share button. The system offers messaging apps as sharing options. The user selects a messaging app and a contact, and the system sends the location data to the selected contact.

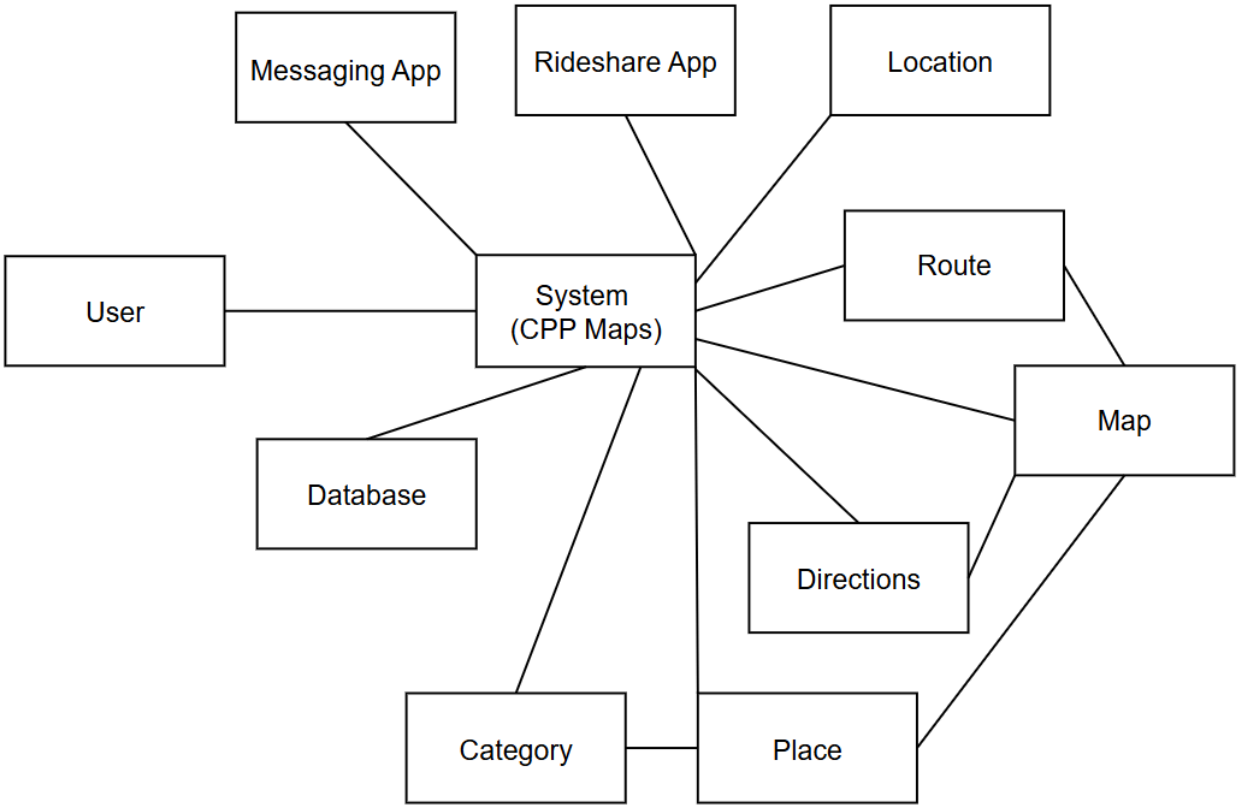
Use Case 5: Share Rideshare

The user selects a location on the map and clicks the share rideshare button. The system offers rideshare apps (e.g., Uber, Lyft) as sharing options. The user selects a rideshare app, and the system sends the location data to the rideshare app for pickup/drop-off details.

Potential Classes (eliminating unnecessary ones):

- **User**: interacts with the system.
- **System**: performs all operations (routes, shares, retrieves).
- **Location**: represents starting, destination, or places.
- **Route**: defines the route for navigation.
- **Database**: stores places, routes, and categories.
- **Messaging App**: facilitates sharing locations.
- **Rideshare App**: facilitates sharing rideshare destinations.
- **Map**: displays locations, routes, places.
- **Place**: represents a searchable entity.
- **Category**: groups places by type.
- **Directions:** turn right, left, u-turn, etc.,

c) Rough graph showing how all the classes are connected



Part 3 - (OOD)

a. CRC Cards

System	
Manages user interaction	User
Retrieves data from the database	Location
Displays data to the user	Route
Calculates routes for navigation	Map
Sends location data to messaging or rideshare apps	Database
Provides turn-by-turn navigation	Messaging App
	Rideshare App
	Directions
	Place
	Category

User	
Enters starting and destination locations	System
Searches for places by name or category	
Shares locations with contacts or rideshare services	
Selects places to view details	
Follows navigation directions	

Vikram Ramesh (Bronco ID: 017319549)

Location	
Holds starting and destination points for navigation	System
Represents places that can be searched or shared	Database
	Map

Route	
Defines the route for navigation	System
Updates the route based on traffic data and roadblocks	Map
Provides alternative routes in case of roadblocks	Database
	Directions

Map	
Displays locations, places, and routes to the user	System
Shows places based on categories or search results	Location
Visualizes routes and directions on the map	Place
	Route
	Directions

Place	
Represents a place that can be searched or viewed	System
Displays details such as ratings, reviews, and contact info	Database
	Map
	Category

Category	
Groups places into specific categories	System
Filters places based on the user's selection	Place
	Database

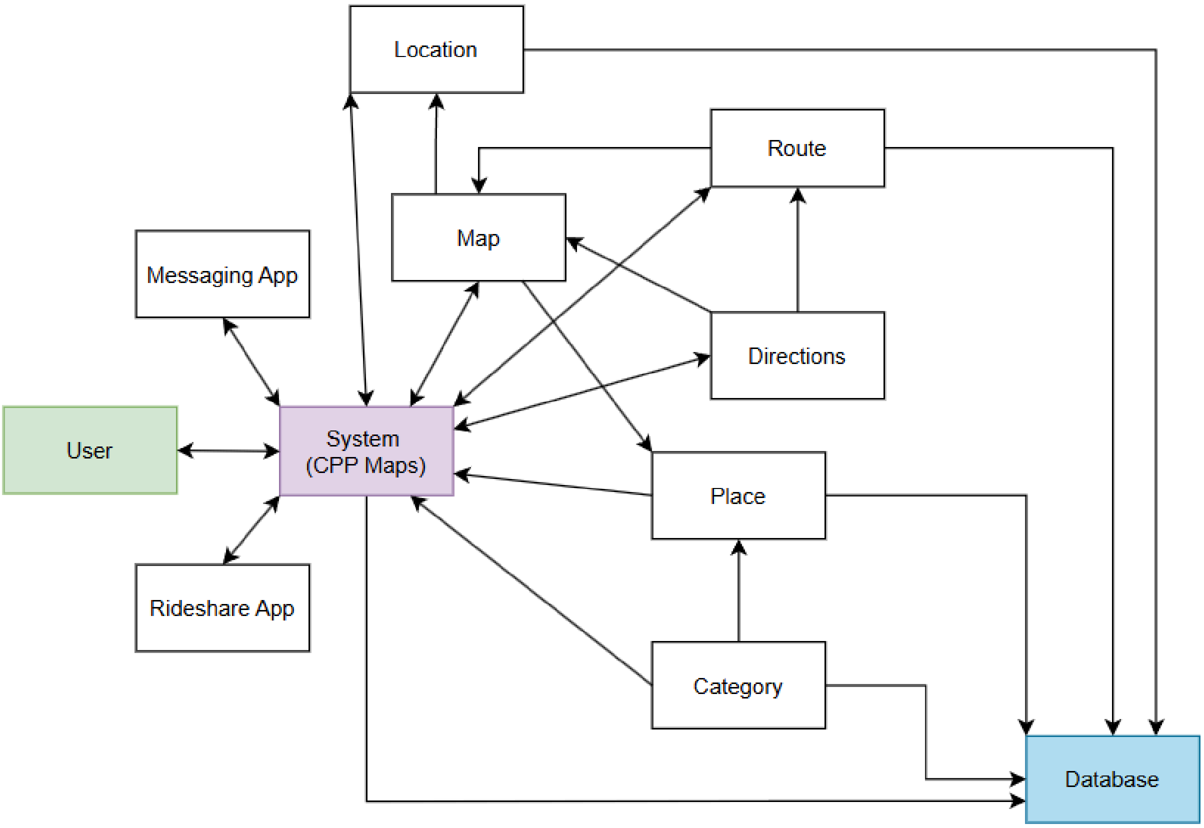
Database	
Stores data for locations, places, routes, and categories	System
Provides the system with data for searches and navigation	Location
Updates data in real-time for accurate navigation details	Place
	Route
	Category

Messaging App	
Facilitates sharing of location with contacts	System
Sends location to selected contact via messaging services	

Rideshare App	
Facilitates sharing of location with rideshare services	System
Sends the selected location as the destination for the ride	

Directions	
Provides turn-by-turn navigation for the user	System
Updates directions in real-time based on traffic conditions	Route
Provides re-routing options when necessary	Map

b. Final graph connecting all the class names with their collaborators


Location
Route
Map
Messaging App
Directions
User
System (CPP Maps)
Place
Rideshare App
Category
Database