Name:	Deepanshu Aggarwal & Kunal Bhatia
UID No.:	2021300002 & 2021300010
Class & Division:	COMPS A (A)
Experiment No.:	2
Aim:	To design the UML diagram – Use Case for the given problem statement

Requirements: UML Diagram Software (Star UML).

Theory:
Use Case Diagram:

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

Purpose of Use Case Diagrams
The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirements, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system.

Following are the purposes of a use case diagram given below:

     1. It gathers the system's needs.

     2. It depicts the external view of the system.

     3. It recognizes the internal as well as external factors that influence the system.

     4. It represents the interaction between the actors.

Use Case Diagram:

Table 1: List of Actors

Actor	Description
Guest:	A Guest User is an unregistered visitor who can explore the platform's public content but cannot access advanced features or contribute to discussions. They have limited interaction capabilities and are encouraged to register for an account to unlock full functionality.
Registered User:	"Registered Users" refers to users who have made an account on "Secure Spot," giving them access to features like reporting events, subscribing to updates, and engaging in conversations. They are crucial users since they actively contribute to the platform and interact with the community.
Admin:	System administrators known as "Admins" have complete control over "Secure Spot." They have the power to control user accounts, set up system preferences, and produce crime reports. They are essential to platform governance and system administration as a whole.

Table 2: List of Use-Cases

#	Use Case	Description
UC1	Register	This use case allows a guest user to create a new account by providing a valid email address and password

UC2	Login	Registered Users securely log into their accounts by entering their email and password credentials.
UC3	Manage profile	Users can update their profile information, including their name, profile, contact details, ensuring their account information is up to date
UC4	Report crime incident	Users can report incidents by providing details such as location, date, time, and description. They can also upload images or documents as evidence.
UC5	View Profile	Users can check their and other user profiles.
UC6	Subscribe to get Notifications	Users can subscribe to receive real-time crime updates and notifications for their selected locations.
UC7	Get crime analysis	This use case enables logged-in users to obtain crime reports pertaining to current incidents and crimes that have happened close to their location. Users can use real-time crime statistics to make informed decisions and keep up with local safety concerns.
UC8	Read Crime Report	Registered users can view in-depth details about individual crimes that have been reported to the "Secure Spot" platform. For a thorough knowledge of each reported crime, users can access incident data, location, date, and user comments.
UC9	Community Interactions	Users can engage in discussions by commenting on posts, sharing opinions, and interacting with other community members.
UC10	Delete crime report	User and Admin can delete crime report if there is something wrong with the post.

Use Case Scenarios

Table3: US1 - Register

Use Case:	Register
Goal:	Add new User
Actor:	Guest User
Pre-condition:	The User shouldn't already exist
Post-condition:	New User must be saved in the Database
Mainline Scenario:	• User selects "Register" -> leads to a page which asks for two fields: email Password (which must meet security requirements, such as a minimum length and complexity criteria). • System validates the email format and password complexity. • User submits the form.
Alternative Flows:	• The system checks if the email is unique; if not, it prompts for a different email. • If unique, System securely stores the user's email address and password in the database, associating them with the new account. • User receives a confirmation email and verifies their email address. • The user can now log in and access their Cashflow account.

Table4: US2 - Login

Use Case:	Login
Goal:	Login User
Actor:	Registered User, Admin
Pre-condition:	User / Admin should have registered.

Post-condition:	No Changes
Mainline Scenario:	- User opens the SecureSpot web application on a browser. - User selects the "Log In" or "Sign In" option to access their account. - System presents a login form, requesting the following information: Email address & Password. - Users enter their registered email address and password into the respective fields. - User clicks the "Log In" button to submit the login form
Alternative Flows:	- System validates the provided email and password: - It checks if the email address is in the correct format. - It verifies the password against the stored hash in the database. - If the email and password combination is valid, the system grants access to the user's account. - Users are redirected to their dashboard or the main interface of the Cashflow application, where they can start managing their personal finances. - If the provided email or password is incorrect, the system displays an error message, prompting the user to re-enter their credentials. - User can make additional attempts to log in or use the "Forgot Password" feature to reset their password if needed

Table5: US3 – Manage Profile

Use Case:	Manage Profile
Goal:	Edit User Profile
Actor:	Registered User
Pre-condition:	User should be Logged In.
Post-condition:	Changes in the Database
Mainline Scenario:	- User logs into their "Secure Spot" account. - User navigates to the "Profile" or "Account Settings" section of the application.

	• System presents the user's current profile information, including their name, profile picture, about, skills, location etc. • User selects the option to edit their profile. • System displays an editable form containing the user's existing profile information. • User updates the desired fields, such as: Name, Profile picture (with an option to upload a new one) .Contact details (e.g., phone number, address) • User reviews the changes and ensures accuracy. • User submits the updated profile information
Alternative Flows:	• After selecting the option to edit their profile in step 4, the user decides not to make any changes and cancels the editing process. • Upon cancellation, the system acknowledges the user's decision not to edit their profile. • The system navigates the user back to the view displaying their existing profile information without any changes. • The use case concludes without any updates made to the user's profile information. • In this alternative scenario, the user chooses to cancel the profile editing process, and as a result, the database remains unchanged, retaining the user's existing profile information.

Table6: US4 – Report Crime

Use Case:	Report Crime
Goal:	Report a crime.
Actor:	Registered User
Pre-condition:	The user is logged into their "SecureSpot" account.
Post-condition:	• The user's incident report is successfully added to the "SecureSpot" database. • Other users can access and view the reported incident and related details. • The user receives a confirmation message indicating the successful submission of the incident report.

Mainline Scenario:	• The user logs into their "SecureSpot" account and navigates to the "Report Crime" section of the platform. • The system presents a reporting form with fields for entering incident details. • The user begins filling out the form, including the following information: • Type of Crime (e.g., theft, vandalism, assault) • Date and Time of the Incident • Location (address or map-based selection) • A Description of the Incident • The option to upload images or documents as evidence • The user reviews the entered information for accuracy and completeness. • Upon confirming the details, the user submits the incident report to the system. • The system processes the report, adding it to the database of reported incidents for community visibility.
Alternative Flows:	• User Cancels Incident Report: After reviewing the entered information in step 4, the user decides to cancel the incident report and return to the platform's main interface. • The use case concludes without the user submitting the incident report. • No incident report is added to the database, and the user returns to the main platform interface.

Table7: US5 – View Profile

Use Case:	View Profile
Goal:	This use case allows registered users to view the profile of another user on the "SecureSpot" platform, providing insights into their community involvement and contributions.
Actor:	Registered User & Admin
Pre-condition:	The user must be on "SecureSpot" platform.
Post-condition:	• The user has successfully viewed the selected user's profile, gaining insights into their community involvement and contributions.
Mainline Scenario:	• The user logs into their "SecureSpot" account and navigates to the platform's "Community" or "User Profiles" section. • The user initiates a search for a specific user profile by

	entering the username or other identifying information.
	• The system displays search results that match the user's criteria, including user profiles.
	• The user selects a specific user profile from the search results.
	• The system presents the selected user's profile, which includes information such as: User's Username, Profile Picture, User's Contributions, Bio or Description etc.
	• The user can browse the selected user's profile to gain insights into their activity and contributions within the "SecureSpot" community.
Alternative Flows:	• If there are no search results that match the user's criteria in step 3, the system notifies the user that no profiles were found and suggests refining the search criteria.
	• After viewing the selected user's profile in step 5, the user can navigate back to the search results or continue exploring other profiles.
	• If the use is interested in connecting with the viewed user, they can choose to follow, send a message, or initiate a conversation from the user's profile.
	• The use case concludes after the user has finished viewing the selected user's profile.

Table8: US6 – Subscribe Notifications.

Use Case:	Subscribe Notifications
Goal:	This use case enables registered users to sign up to receive notifications about criminal occurrences and updates occurring in their neighborhood or other areas of interest.
Actor:	Registered User
Pre-condition:	The user is logged into their "Secure Spot" account.
Post-condition:	• The user's subscription preferences are successfully saved to their profile. • The user will receive notifications based on the selected subscription categories and preferred delivery method.
Mainline Scenario:	• The user logs into their "Secure Spot" account and navigates to the footer section of the platform. • The user clicks on the Subscribe button in the footer Section to get regular crime alerts in his area.

Alternative Flows:	• While Scrolling down, user decides not to subscribe for notifications and abort the process.

Table9: US7 – Generate Crime-Analysis

Use Case:	Generate Crime-Analysis
Goal:	This use case enables registered users with an interest in crime analysis to access and analyze crime data on the "Secure Spot" platform.
Actor:	Registered User
Pre-condition:	The user is logged into their "Secure Spot" account.
Post-condition:	• The user has successfully conducted a crime data analysis using the available tools and parameters. • The user can gain insights, generate reports, or visualize crime trends based on their analysis.

Mainline Scenario:	• The user logs into their "Secure Spot" account and navigates to the "Crime Analysis" or "Analytics" section of the platform.
	• The system presents the user with a range of analytical tools and options for crime data analysis, such as:
	
	    1.) Crime Type Filters
	    2.) Time Frame Selection (e.g., daily, weekly, monthly)
	    3.) Geographic Mapping and Visualization
	    4.) Statistical Reporting
	• The user selects specific crime data parameters and filters based on their analysis requirements.
	• The system retrieves and processes the crime data based on the selected parameters.
	• The user is presented with analytical results, which may include Graphs and Charts depicting crime trends.

Alternative Flows:	• After reviewing the initial analytical results in step 5, the user may choose to adjust the analysis parameters and filters for a more refined analysis.

Table10: US8 – Generate Crime-Report

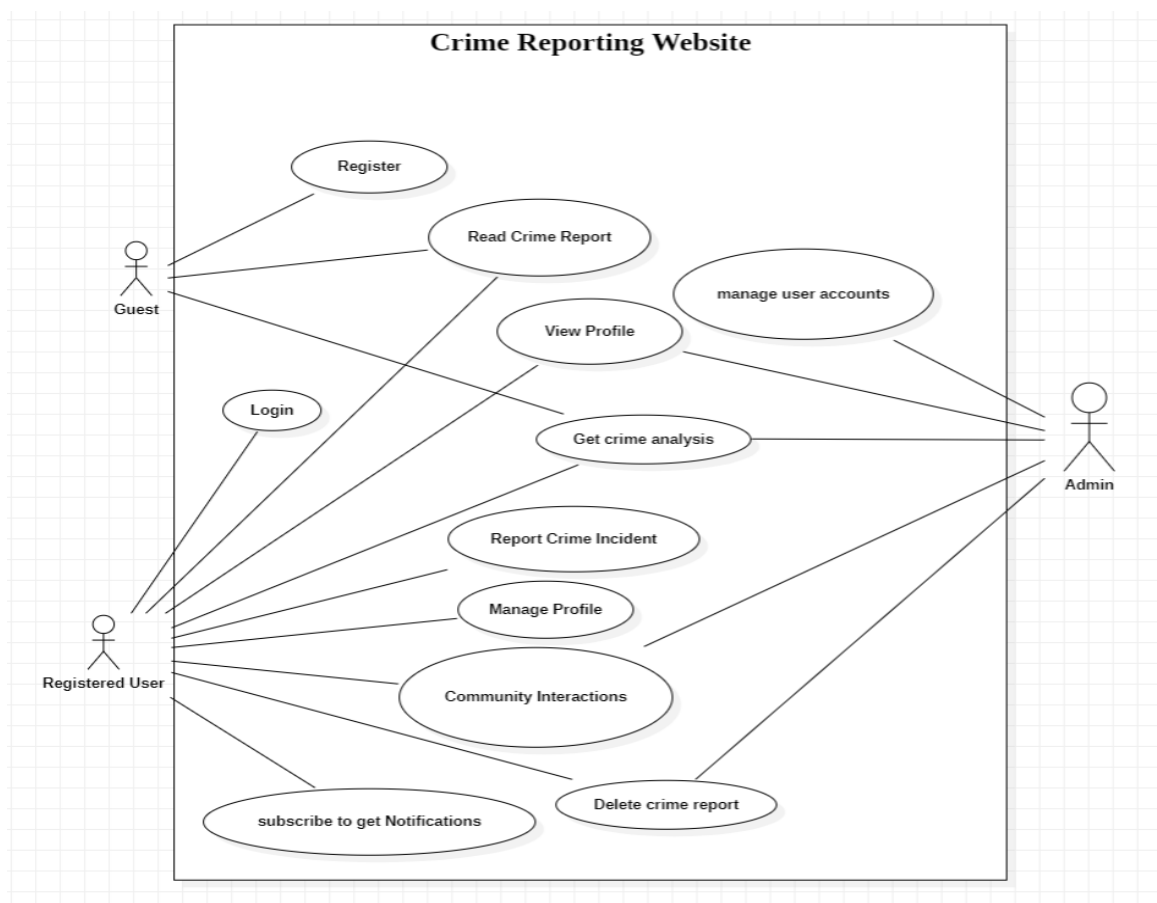| Use Case:	Generate Crime-Report
Goal:	This use case enables the users to get the information about a particular crime and know how and where the incident happened.
Actor:	Registered User
Pre-condition:	The user must be on the "SecureSpot" platform.
Post-condition:	• The user has successfully read detailed information about the selected crime report. • Users can engage in discussions or explore other crime reports as desired.

Mainline Scenario:	• The user logs into their "SecureSpot" account and navigates to the "Crime Reports" section of the platform.
	• The system presents a list of crime reports, either as a general list or categorized by crime type, location, or date.
	• The user selects a specific crime report from the list that interests them.
	• The system displays a detailed view of the selected crime report, including information such as:
	    1.) Type of Crime (e.g., theft, vandalism)
	    2.) Date and Time of the Incident
	    3.) Location (including a map if available)
	    4.) A Description of the Incident
	• User Comments and Discussions related to the report.
	• Users can navigate back to the list of crime reports or continue exploring additional reports as needed.

Alternative Flows:	• If there are no crime reports that match the user's criteria in step 2, the system notifies the user that no reports were found and suggests refining the search criteria. • After reading the selected crime report in step 4, the user decides to engage in discussions or provide comments related to the report. • The use case concludes after the user has finished reading the selected crime report and any related discussions.

Table11: US9 – Post Comment

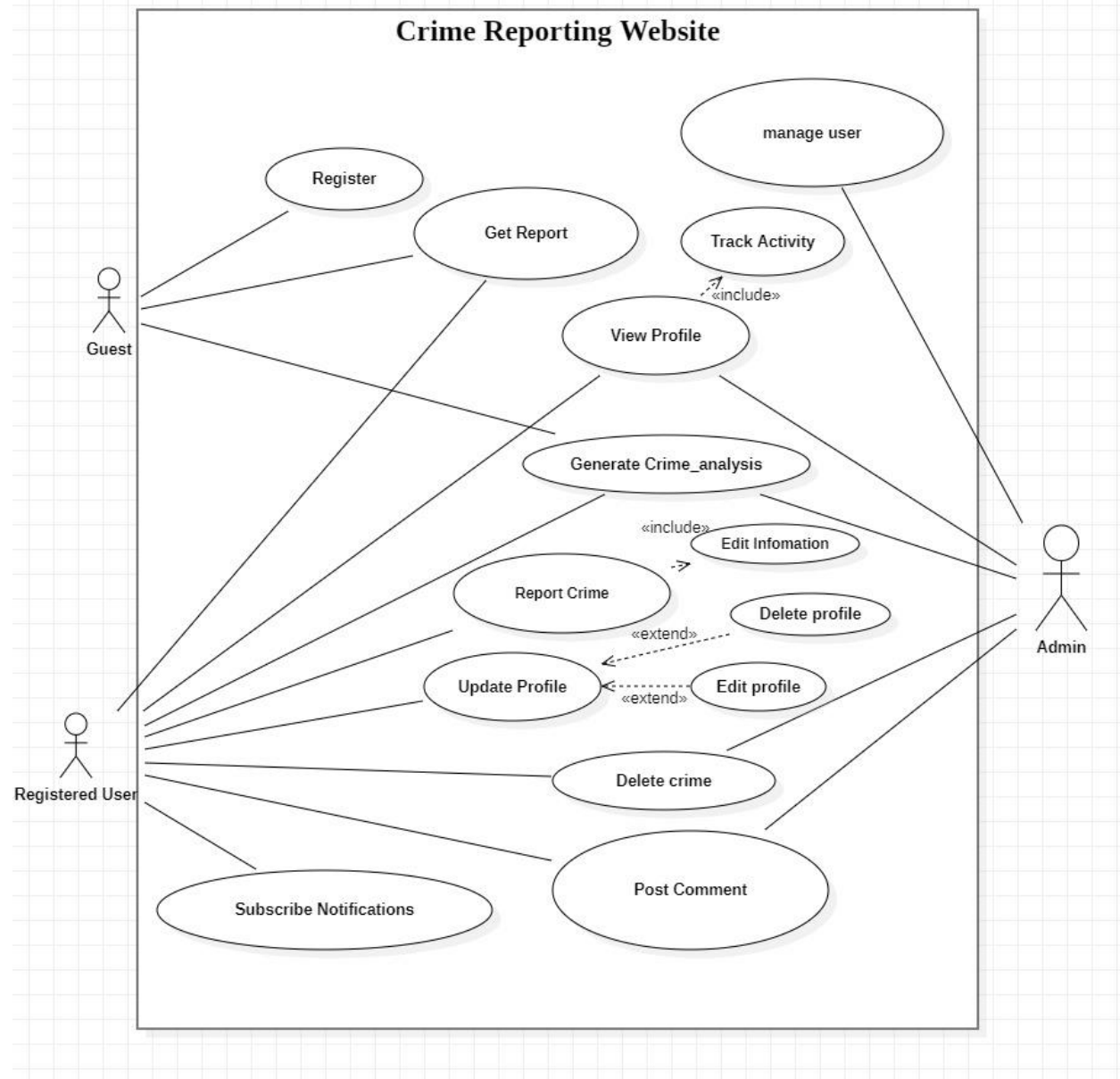| Use Case:	Post Comment
Goal:	This use case enables the users to share their views on a particular incident.
Actor:	Registered User
Pre-condition:	The user is logged into their "Secure Spot" account.
Post-condition:	• The user has successfully read detailed information about the selected crime report.
Mainline Scenario:	• The user logs into their "SecureSpot" account and navigates to a crime incident. • After coming on crime incident, he navigates to the bottom of the page and enters to community Interactions sections. • The user can actively participate in community interactions by simply sharing his views on the incident
Alternative Flows:	• After participating in a specific community interaction in step 5, the user decides to continue engaging in discussions or exploring other interactions. • The use case concludes after the user has finished engaging in community interactions.

Table12: US10 – Delete CrimeReport

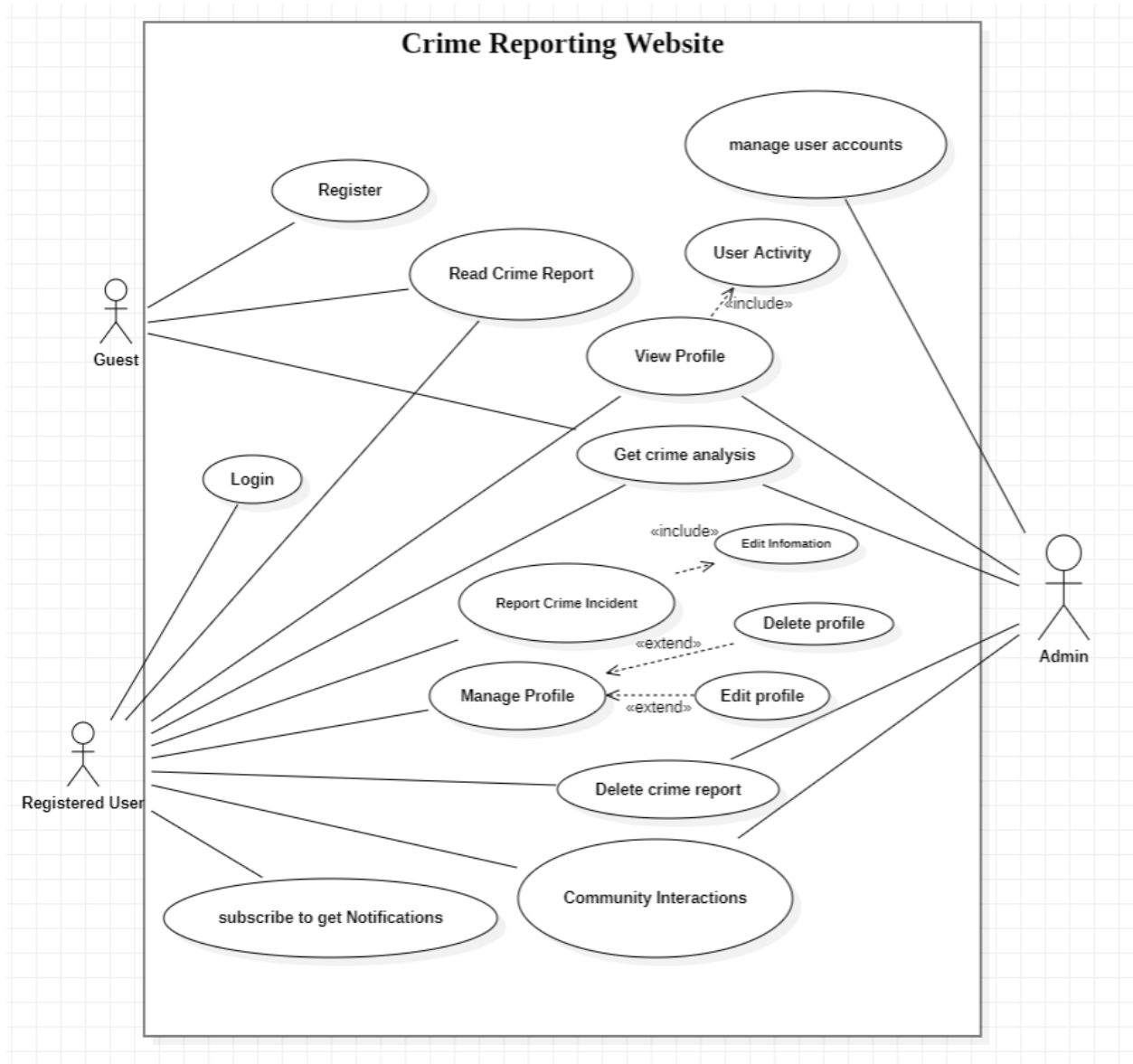| Use Case:	Delete CrimeReport
Goal:	This use case enables the users to share their views on a particular incident.
Actor:	Registered User or Admin
Pre-condition:	The user is logged into their "SecureSpot" account.
Post-condition:	• The user's selected crime report has been successfully deleted from the "SecureSpot" platform
Mainline Scenario:	• The user logs into their "SecureSpot" account and navigates to the "My Reports" or "My Incidents" section of the

	platform. • The system presents a list of crime reports authored by the user. • The user selects the specific crime report that they wish to delete. • The system displays the selected crime report with options for actions. • The user selects the "Delete" or "Remove" option associated with the crime report. • The system presents a confirmation dialog to ensure the user's intent to delete the report. • The user confirms their decision to delete the report. • The system removes the crime report from the platform, and it is no longer accessible to other users.
Alternative Flows:	• After selecting the "Delete" option in step 5, the user decides not to proceed with the deletion and cancels the action. • The use case concludes without the deletion of the crime report, and the report remains accessible on the platform.

Use Case Diagram Level-1:

Level 2:

Crime Reporting Website

- manage user
- Register
- Get Report
- Track Activity
- View Profile
- Generate Crime_analysis
- «include» Edit Infomation
- Report Crime
- Delete profile
- «extend» Update Profile
- Edit profile «extend»
- Delete crime
- Post Comment
- Subscribe Notifications

Actors: Guest, Registered User, Admin

Crime Reporting Website

Conclusion:

In this experiment, we created a use case diagram for our crime report website application. We understood the different actor and use cases in the web application. The UML diagram was created with 10 use cases and their associativity with the actors and relations with other use cases.