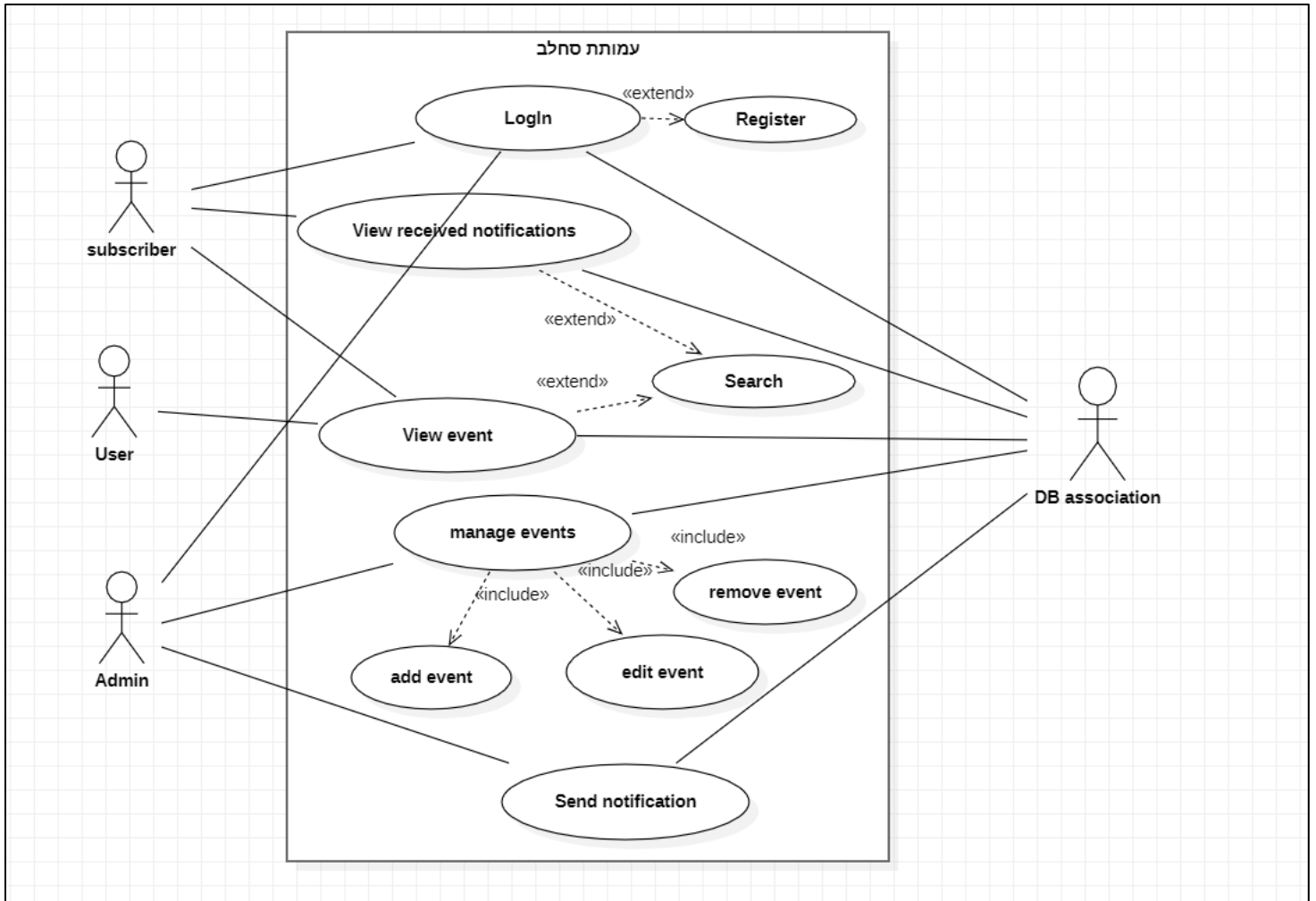


## Use Case UML:



## Use Case Templates:

- Use Case Name: Log-In
  - Brief Description: Allows users to authenticate and access the application.
  - Actor: Subscriber, Admin, DB Association.
  - Pre-conditions: User must be registered and on the login page, entering username and password.
  - Basic Flow:
    - i) User initiates this use case by entering the login page.
    - ii) User enters credentials (username and password).
    - iii) The system verifies credentials.
    - iv) Users are granted access to the application.
  - Alternate Flows: log-In by phone number.
  - Exception Flows: Invalid credentials; system prompts users to retry.
  - Post Condition: User successfully logged in.
- 
- Use Case Name: Register
  - Brief Description: Allows a new user to create an account.
  - Actor: Subscriber, Admin, DB Association.
  - Pre-conditions: User is not already registered.
  - Basic Flow:
    - i) User provides registration details.
    - ii) The system validates and stores user data.
    - iii) User receives confirmation of successful registration.
  - Alternate Flows: Email already in use; system prompts for a different email.
  - Exception Flows: Invalid input; system requests corrections.
  - Post Condition: User is successfully registered.
- 
- Use Case Name: View Received Notifications.
  - Brief Description: Allows users to check notifications.
  - Actor: Subscriber.

- Pre-conditions: User is logged in and on the home page, where a menu is active with "Notifications".
- Basic Flow:
  - i) User initiates this use case by selecting "Notifications".
  - ii) System displays received notifications.
- Alternate Flows: No new notifications; system displays an empty list.
- Exception Flows: System failure prevents notifications from loading.
- Post Condition: User views received notifications.

- Use Case Name: Search.
- Brief Description: Enables users to search for notifications or events.
- Actor: Subscriber, User.
- Pre-conditions: User is logged in and on the home page where there is a search bar.
- Basic Flow:
  - i) Users are logged in and on the home page where there is a search bar.
  - ii) Users enter search criteria.
  - iii) System processes and returns matching results.
- Alternate Flows: No matching results found; system notifies the user.
- Exception Flows: System error prevents search.
- Post Condition: User retrieves search results.

- Use Case Name: View Event
- Brief Description: Allows users to view event details.
- Actor: Subscriber, User.
- Pre-conditions: User is logged in and on the home page, where a menu is active with "Events".
- Basic Flow:
  - i) User initiates this use case by selecting "Events".
  - ii) System displays All events with details.
- Alternate Flows: There are no events; the system notifies the user.

- Exception Flows: System error prevents event display.
  - Post Condition: User successfully views event details.
- 
- Use Case Name: Manage Events.
  - Brief Description: Allows the admin to manage events, including adding, editing, and removing them.
  - Actor: Admin.
  - Pre-conditions: The admin is logged in and on the home page, where a menu labeled "Manage Events" is available.
  - Basic Flow:
    - i) The admin initiates the use case by selecting "Manage Events."
  - Alternate Flows: No changes are made.
  - Exception Flows: A system error prevents the admin from accessing "Manage Events".
  - Post Condition: The admin successfully accesses the event management section.
- 
- Use Case Name: Add Event.
  - Brief Description: Allows the admin to create a new event.
  - Actor: Admin.
  - Pre-conditions: The admin is logged in and has accessed "Manage Events," where a menu presents three options: Edit Event, Add Event, and Delete Event.
  - Basic Flow:
    - ii) The admin initiates the use case by selecting "Add Event".
    - iii) The system displays the required fields for event creation, including event name, event date, event description, event type.
    - iv) The admin fills in the required details.
    - v) The system validates the input and saves the event.
  - Alternate Flows: If required details are missing, the system prompts the admin to complete the information.
  - Exception Flows: A system failure prevents event creation.

- Post Condition: The event is successfully added.
- 
- Use Case Name: Edit Event.
  - Brief Description: Allows the admin to modify existing event details.
  - Actor: Admin.
  - Pre-conditions: Pre-conditions: The admin is logged in and has accessed "Manage Events," where a menu presents three options: Edit Event, Add Event, and Delete Event.
  - Basic Flow:
    - i) The admin initiates the use case by selecting "Edit Event".
    - ii) The system displays the current event details: Event name, Event date, Event description, Event type.
    - iii) The admin modifies the necessary information.
    - iv) The system validates and saves the updates.
  - Alternate Flows: No modifications are made.
  - Exception Flows: A system failure prevents event updates.
  - Post Condition: The event details are successfully updated.
- 
- Use Case Name: Remove Event.
  - Brief Description: Allows the admin to delete an existing event.
  - Actor: Admin.
  - Pre-conditions: The admin is logged in and has accessed "Manage Events," where a menu presents three options: Edit Event, Add Event, and Delete Event.
  - Basic Flow:
    - i) The admin initiates the use case by selecting "Delete Event".
    - ii) The system displays a list of all events.
    - iii) The admin selects the specific event to delete.
    - iv) The system prompts confirmation.
    - v) The admin confirms the deletion.
    - vi) The admin confirms the deletion.

- vii) The system removes the event from the database.
  - Alternate Flows: No events are available for deletion.
  - Exception Flows: A system failure prevents event deletion.
  - Post Condition: The event is successfully removed.
- 
- Use Case Name: Send Notification
  - Brief Description: Allows the admin to send notifications to users.
  - Actor: Admin.
  - Pre-conditions: The admin is logged in and on the home page, where a menu labeled "Send Notification" is available.
  - Basic Flow:
    - i) The admin initiates the use case by selecting "Send Notification".
    - ii) The system displays a list of all subscribers.
    - iii) The admin selects the specific subscriber to send the notification to.
    - iv) The admin composes the notification.
    - v) The system sends the notification to the selected subscriber.
  - Alternate Flows: No subscribers are available to receive notifications.
  - Exception Flows: A system failure prevents the notification from being sent.
  - Post Condition: The notification is successfully sent.

## Requirements:

- Functional Requirements:
  - i) Notifications:
    - Send mobile notifications for:
      - New messages from the association.
      - Updates on events or donations.
    - All the previous Notifications should be saved and can be accessed by users.
  - ii) Administrative Tools:

- Admins can create, update, and delete events, donations, and notifications.

iii) Platform Compatibility:

- Available on android and iOS devices.

- Non-Functional Requirements:

i) Performance:

- The system shall handle concurrent access by up to 5000 users without performance degradation.

ii) Response Time:

- The system should be loaded within 10 seconds.

iii) Portability:

- The user experience and design should remain consistent across Android and iOS.

## Risk Management template:

				Priorities			Plan to reduce impact
Sort by priority	Description of risk	Owner	Possible impact	Prob (%)	Impact (L, M, H, VH)	Risk code (color per table above)	Mitigation plan / Contingency plan
1	Project management leaves the project.	Rana	Project delays, lack of leadership.	0 - 25%	VH	3	Have a backup project manager who is familiar with the project and ready to take over responsibilities.
2	Failure to Complete the Project on Time.	Project Team.	Delayed delivery, missed deadlines.	25 - 50%	VH	2	Implement Project Planning Techniques such as: WBS (Work Breakdown Structure), Critical Path Analysis, and Gantt Charts to stay on track.
3	Data Privacy Breach.	Yousif	Unauthorized data access, legal implications	50 - 75%	Medium	3	Minimize sensitive data collection. Only request essential information like phone numbers; avoid storing financial details (e.g., credit card, bank info).



4	Underestimation of Application Complexity.	Project team	Unexpected delays, increased workload	50 - 75%	M	3	Conduct technical feasibility analysis before development, allocate buffer time, and follow an agile methodology for better flexibility.
5	Changes in Application Requirements.	Client & Yousif	Scope creep, increased workload.	0 - 25%	H	4	Evaluate impact on deadlines. If major changes arise, communicate with the client to adjust expectations based on available time.
6	Development Tools Do Not Meet Expectations.	Project Team	Delays due to tool limitations.	25 - 50%	M	4	Test and validate tools before committing them. Maintain alternative solutions in case the primary tools fail.