CSC584: ENTERPRISE PROGRAMMING

REPORT: Web Application System

TITLE: Safe Neighbourhood Network System

PREPARED BY:

1	MUHAMMAD FAIZ BIN ABD RAZAK	2023911655
2	MOHAMMAD ADAM BIN RAJIAIN	2021410664
3	ANNA KHALILA BINTI KHAIRUDDIN	2023267298
4	SITI NUR UZMA BINTI MOHD NASIR	2023408076

PREPARED FOR:
MUHAMMAD RIDHWAN BIN MOHAMMAD RAZALI

TABLE OF CONTENT

1.0 INTRODUCTION

In recent years, the need for safer and more connected communities has become increasingly important. As urban areas grow and populations become more diverse, the challenge of maintaining effective communication between residents and local authorities becomes more complex. To address this challenge, the Safe Neighbourhood Network System was developed as a web-based platform aimed at enhancing community safety and public engagement.

The Safe Neighbourhood Network System serves as a centralized digital solution that facilitates two-way communication between residents and local governing bodies such as municipal councils, resident associations, or neighborhood committees. Through this system, residents can easily report issues or incidents occurring in their area ranging from criminal activity and suspicious behavior to infrastructure problems or environmental hazards. These reports are instantly accessible by the responsible authorities, enabling quicker response times and more efficient resource allocation.

Beyond incident reporting, the system includes a number of community-focused features. It allows users to create and participate in neighborhood events, ensuring a sense of involvement and togetherness among residents. The announcement module enables administrators to broadcast important updates, safety alerts, or community notices to all registered users, reducing the risk of miscommunication or delayed information. Additionally, a reporting module lets users submit concerns such as maintenance issues, complaints, or public safety hazards directly to the appropriate departments, fostering transparency and accountability.

Security and accessibility are key aspects of the system. Each user is assigned a role either as a resident or an admin with different permissions and levels of access. Admins can review, approve, and manage submitted reports and events, as well as control the visibility of announcements. Meanwhile, residents can track the status of their reports, receive feedback, and stay informed about activities happening in their neighborhood.

By digitalizing and streamlining communication within communities, the Safe Neighbourhood Network System not only improves the efficiency of handling public concerns but also strengthens trust and cooperation between residents and local authorities. Ultimately, the system aims to create a safer, more transparent, and more collaborative living environment where every individual plays a part in maintaining the peace and harmony of their surroundings.

This report details the development process, system features, technical implementation, and potential impact of the Safe Neighbourhood Network System. It outlines the technologies used, database structures, user interface design, and the roles and interactions of different system users. Through this project, we hope to demonstrate how simple, user-friendly digital platforms can contribute significantly to community resilience and proactive safety management.

2.0   OBJECTIVES

- Create an accessible, user-friendly platform
- Bridge the gap between community members and law enforcement
- Ensure prompt response to safety concerns

3.0 PROBLEM STATEMENT

In many residential communities, maintaining safety and fostering a sense of collective responsibility remain major challenges. Traditional methods of communication such as community notice boards, word-of-mouth updates, and periodic meetings are often inefficient, delayed, or inaccessible to all residents. As a result, important information regarding neighborhood incidents, safety hazards, or community events may not reach the relevant parties in time, potentially putting lives and property at risk.

Additionally, residents often lack a convenient and structured platform to report incidents or submit complaints directly to local authorities. This results in underreporting of important issues such as vandalism, theft, noise disturbances, or infrastructure problems. When issues are reported, there is frequently no system in place to track the status or receive feedback, leading to frustration and a lack of trust in the authorities' responsiveness.
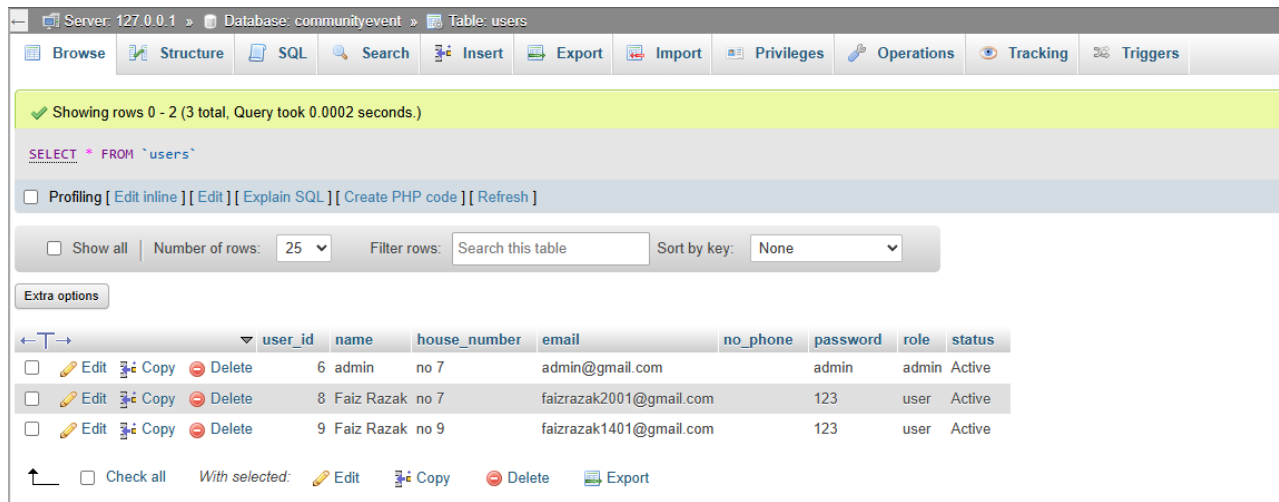
Moreover, authorities face difficulties in managing resident feedback, disseminating important announcements efficiently, and organizing community-related activities due to fragmented communication channels. These limitations hinder efforts to build safer, more cohesive neighborhoods where residents feel informed, involved, and empowered.

Therefore, there is a pressing need for a centralized, web-based platform that enables real-time communication between residents and authorities, facilitates structured reporting and feedback, and supports the organization of community events and announcements.

The Safe Neighbourhood Network System aims to solve these problems by providing a user-friendly digital solution where residents can actively participate in maintaining their community's safety, while authorities can efficiently manage reports, respond to concerns, and communicate updates. Through this system, we aim to improve public engagement, response time, and overall community trust in safety and governance mechanisms.

4.0 DATABASE MYSQL TABLE

4.1 USERS TABLE

CREATE TABLE users (
 user_id int(11) NOT NULL,
 name varchar(100) DEFAULT NULL,
 house_number varchar(50) DEFAULT NULL,
 email varchar(100) DEFAULT NULL,
 no_phone varchar(20) DEFAULT NULL,
 password varchar(255) DEFAULT NULL,
 role varchar(20) DEFAULT 'user'

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

4.2   REPORTS TABLE


```
Server: 127.0.0.1 » Database: communityevent » Table: reports

Browse    Structure    SQL    Search    Insert    Export    Import    Privileges    Operations    Tracking    Triggers

Showing rows 0 - 2 (3 total, Query took 0.0002 seconds.)

SELECT * FROM `reports`

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25    Filter rows: Search this table    Sort by key: None

Extra options
```

	report_id	description	category	timestamp	status	user_id
Edit Copy Delete	5	longkang tersubat	Maintenance	2025-07-24 12:27:55	In Progress	6
Edit Copy Delete	6	gotong-royong	Maintenance	2025-07-24 13:41:08	Resolved	8
Edit Copy Delete	7	kita lantik security guard	Security	2025-07-24 14:00:45	Resolved	8
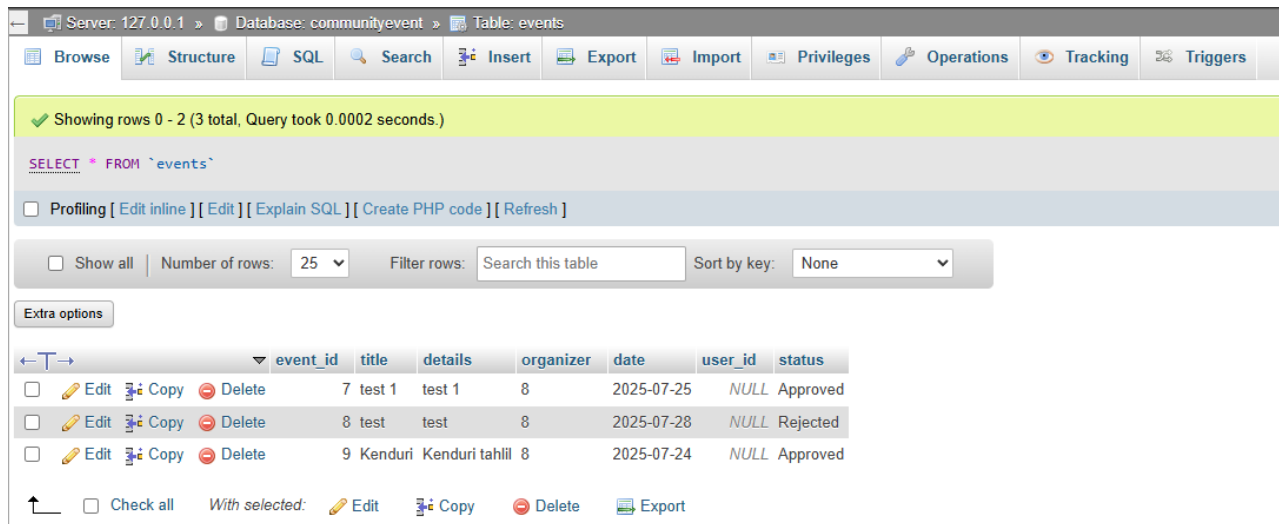
Check all   With selected: Edit   Copy   Delete   Export

CREATE TABLE reports (
  report_id int(11) NOT NULL,
  description text DEFAULT NULL,
  category varchar(50) DEFAULT NULL,
  timestamp datetime DEFAULT current_timestamp(),
  status varchar(20) DEFAULT 'pending',
  user_id int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

4.3   EVENTS TABLE

CREATE TABLE events (
  event_id int(11) NOT NULL,
  title varchar(100) DEFAULT NULL,
  details text DEFAULT NULL,
  organizer varchar(100) DEFAULT NULL,
  date date DEFAULT NULL,
  user_id int(11) DEFAULT NULL,
  status varchar(20) DEFAULT 'pending'

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

4.4   ANNOUNCEMENTS TABLE

Server: 127.0.0.1 » Database: communityevent » Table: announcements

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

✔ Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)

```
SELECT * FROM `announcements`
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 ▾ | Filter rows: Search this table

Extra options

	announcement_id	title	content	date_created	event_id	date_posted
☐ ✏ Edit 📋 Copy ⊖ Delete	4	test	test	2025-07-24 21:13:51	NULL	2025-07-24 13:13:51

☐ Check all With selected: ✏ Edit 📋 Copy ⊖ Delete Export

CREATE TABLE announcements (
 announcement_id int(11) NOT NULL,
 title varchar(100) DEFAULT NULL,
 content text DEFAULT NULL,
 date_created datetime DEFAULT current_timestamp(),
 event_id int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

**5.0 SESSION INTERFACE WITH DESCRIPTION**

    5.1   Login Page
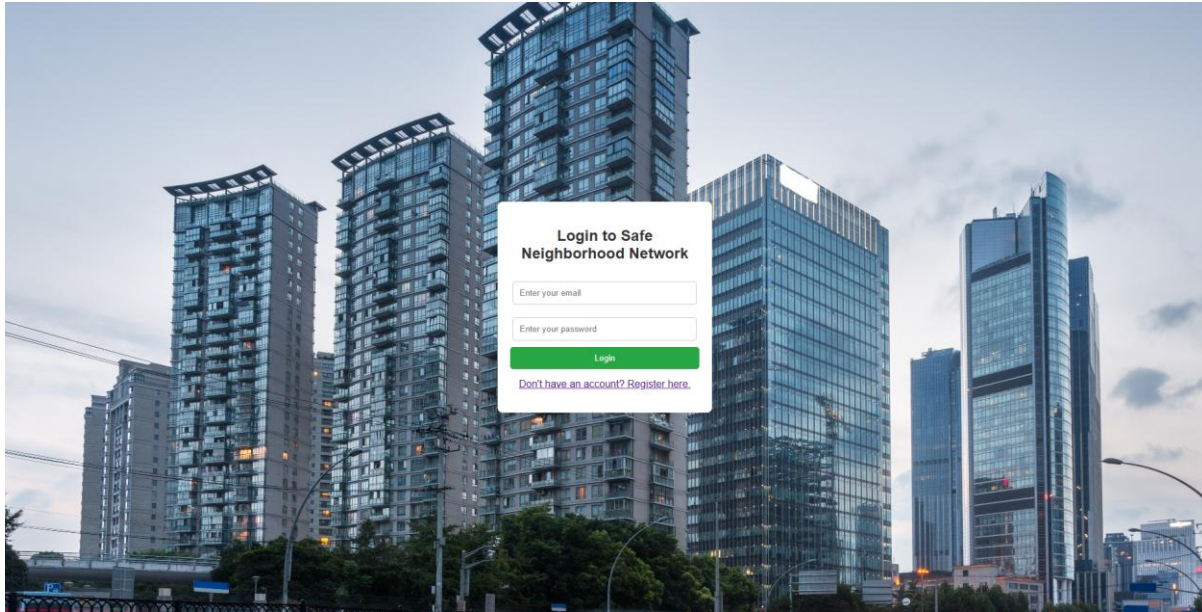
The image displays a login interface for the Safe Neighborhood Network, presented on top of a backdrop featuring modern urban architecture.

At the center of the image is a clean and minimalistic login box, which includes:

- **Title:** "Login to Safe Neighborhood Network"
- **Fields:**
    - "Enter your email"
    - "Enter your password"
- **Buttons:**
    - A prominent "Login" button
    - A registration link labeled "Don't have an account? Register here."

The overall design implies a secure and professional platform likely related to community safety or urban networking.

5.2   Register
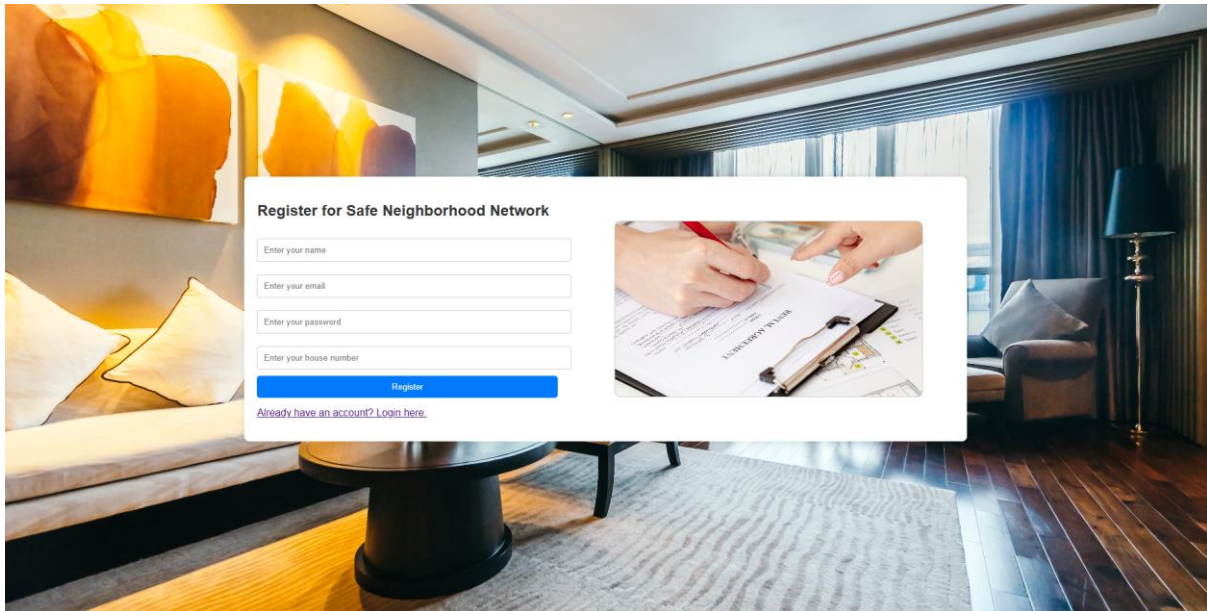
Overlaid on this cozy backdrop is a registration form, which includes the following components:

- **Title:** "Register for Safe Neighborhood Network"
- **Input Fields:**
  - Enter your name
  - Enter your email
  - Enter your password
  - Enter your house number
- **Actions:**
  - A prominent blue "Register" button
  - A login redirect link: "Already have an account? Login here."

Additionally, there is a small photo in the corner showing someone filling out a paper form with a red pen, reinforcing the theme of community participation and secure registration.

5.3 Admin

5.3.1 Dashboard
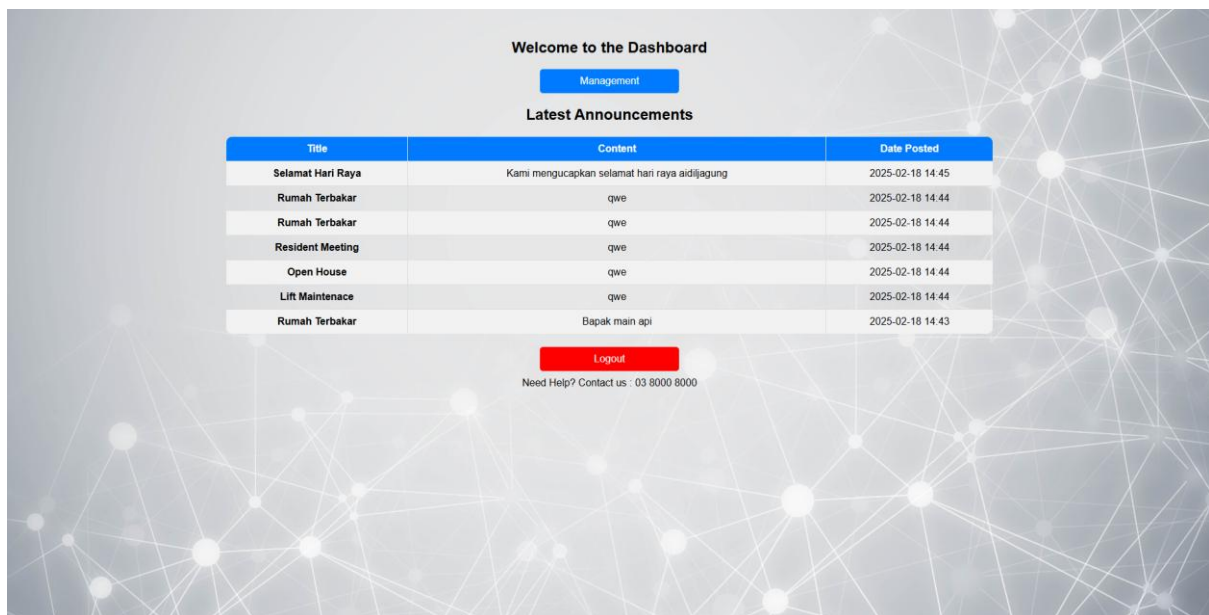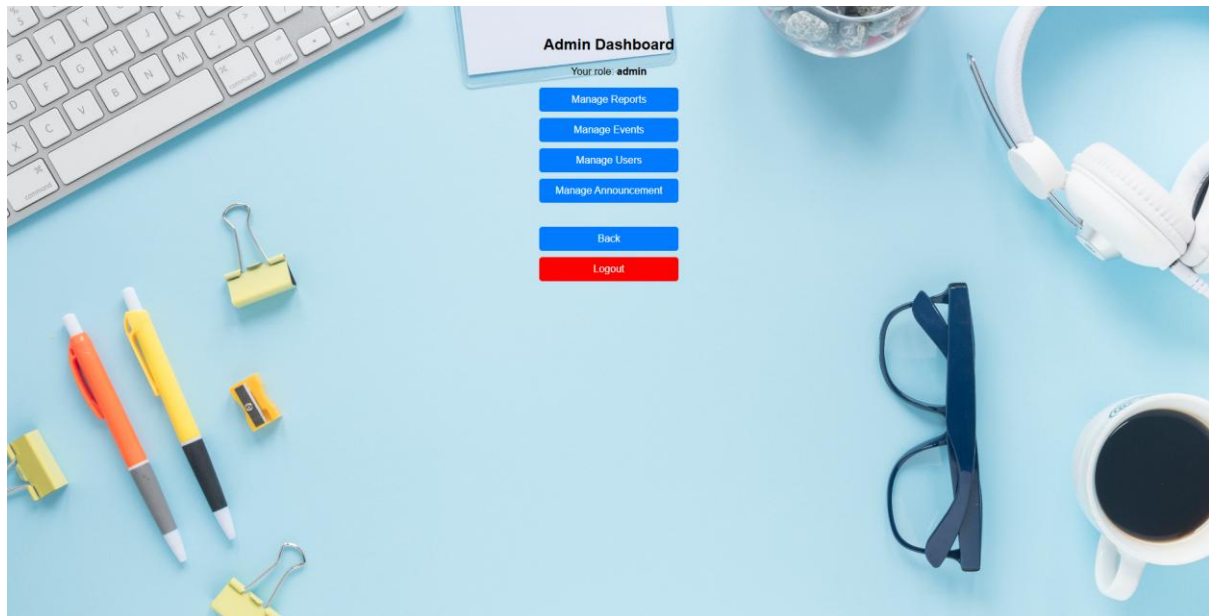
Safe Neighborhood Network – Admin Dashboard Interface
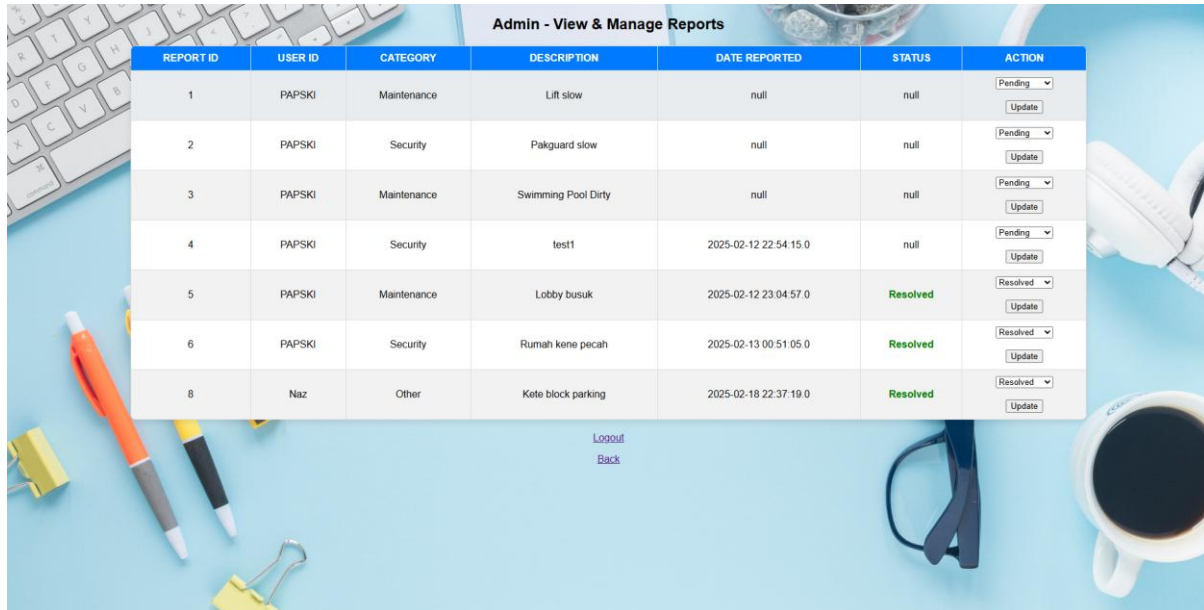
The image presents an administrative dashboard for the Safe Neighborhood Network, designed to provide centralized control and streamlined management functionality. The interface is framed within a workspace environment that suggests clarity, order, and purpose.

Interface Design:

- **Dashboard Title:** Clearly marked as "Admin Dashboard"
- **User Role:** Displayed prominently as "Your role: admin"
- **Navigation Buttons:**
    - **Manage Reports** – for overseeing submitted safety reports

- o **Manage Events** – to coordinate neighborhood gatherings or alerts
- o **Manage Users** – to handle member access and permissions
- o **Manage Announcements** – for broadcasting important community updates
- o **Back** – returns to the previous screen
- o **Logout** – securely exits the administrator session

5.3.2   Report Management

Safe Neighborhood Network – Admin Report Management Interface

The image depicts the **"View & Manage Reports"** section of the Safe Neighborhood Network
Admin Dashboard, which facilitates oversight of user-submitted issues related to neighborhood
safety, maintenance, and general concerns.

Report Table Overview:

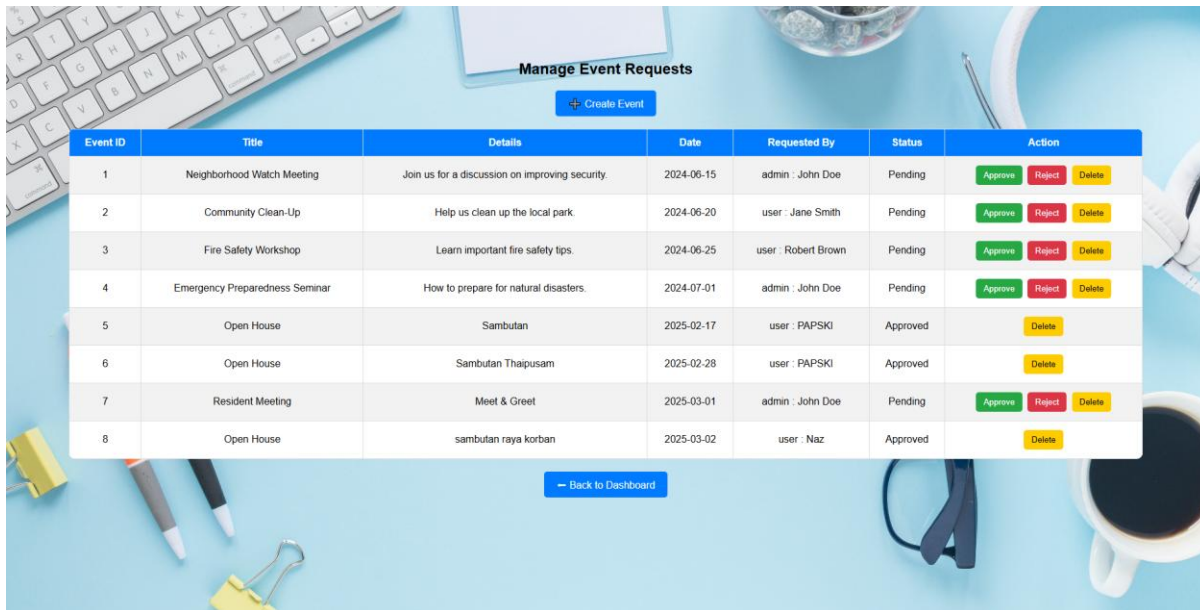The central feature of the interface is a tabulated layout comprising the following columns:

- **Report ID** – Unique identifier for each submission
- **User ID** – The account that submitted the report
- **Category** – Classification of issue (Maintenance, Security, Other)
- **Description** – Brief notes detailing the concern
- **Date Reported** – Timestamp indicating when the report was filed
- **Status** – Current state of resolution (Pending, Resolved, or blank)
- **Action** – Includes an "Update" option for modifying report status

Administrative Controls:

- Action links "**Logout**" and "**Back**" positioned below the table to allow seamless navigation
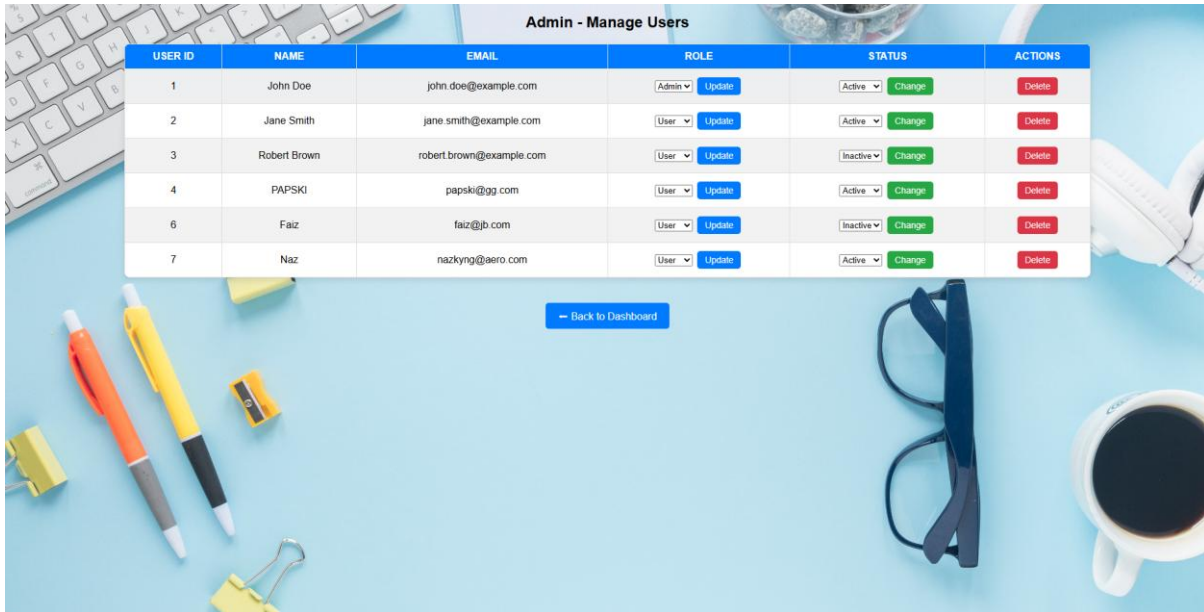- "Update" buttons beside each entry support direct status changes by the administrator

Safe Neighborhood Network – Event Management Dashboard

This image showcases the **"View & Manage Events"** section of the Safe Neighborhood Network Admin Dashboard, tailored for overseeing community event requests and coordination tasks.

Interface Layout:

- A structured **tabular display** organizes submitted event requests into distinct columns:
    - Event ID
    - Title
    - Details
    - Date
    - Requested By
    - Status
    - Action
- Administrative **action buttons** are available for each entry, offering options to:
    - Approve
    - Reject
    - Delete
- System-level controls include:
    - A button to **Create New Event**
    - A **Back** navigation option to return to the main dashboard

Safe Neighborhood Network – User Management Interface

The image presents the **"View & Manage Users"** section of the Safe Neighborhood Network Admin Dashboard, tailored for administrative control over community user accounts.

Interface Structure:

- The layout features a **table-based design** with clearly labeled columns for key user attributes, including:
    - User identification
    - Personal information
    - Role assignments
    - Account status
    - Administrative actions
- Each row represents an individual user entry with management options positioned for easy access.

Core Functions:

- **Role Management:** Allows administrators to modify user roles within the system.
- **Status Control:** Supports updating the activity or membership status of users.
- **Account Removal:** Provides the ability to delete user records as needed.
- **Navigation Buttons:** Includes controls for returning to the main dashboard or logging out.

Purpose:

The design streamlines user oversight by presenting essential controls in a structured and accessible format. This facilitates quick decision-making and ensures smooth administration of user data and account privileges.

5.3.5   Announcement Management

Safe Neighborhood Network – Announcement Management Interface

This interface represents the **"View & Manage Announcements"** section within the Safe Neighborhood Network Admin Dashboard, designed to facilitate streamlined creation, monitoring, and control of public communications.

Core Functional Components:

- **Announcement Creation Panel:**
    - Input fields for announcement title and content
    - A prominent "Post Announcement" button to publish the message
- **Announcements Table:**
    - Structured in columns for system tracking and display, including:
        - ID
        - Title
        - Content
        - Date
        - Action options
- **Administrative Tools:**
    - A "Delete" button associated with each announcement entry for removal
    - System navigation options for returning to prior screens or logging out

Functional Purpose:

This module supports timely and organized dissemination of information across the neighborhood network. The layout encourages clarity and ease of use, empowering administrators to manage announcements with precision and efficiency.

5.4.1   User Dashboard
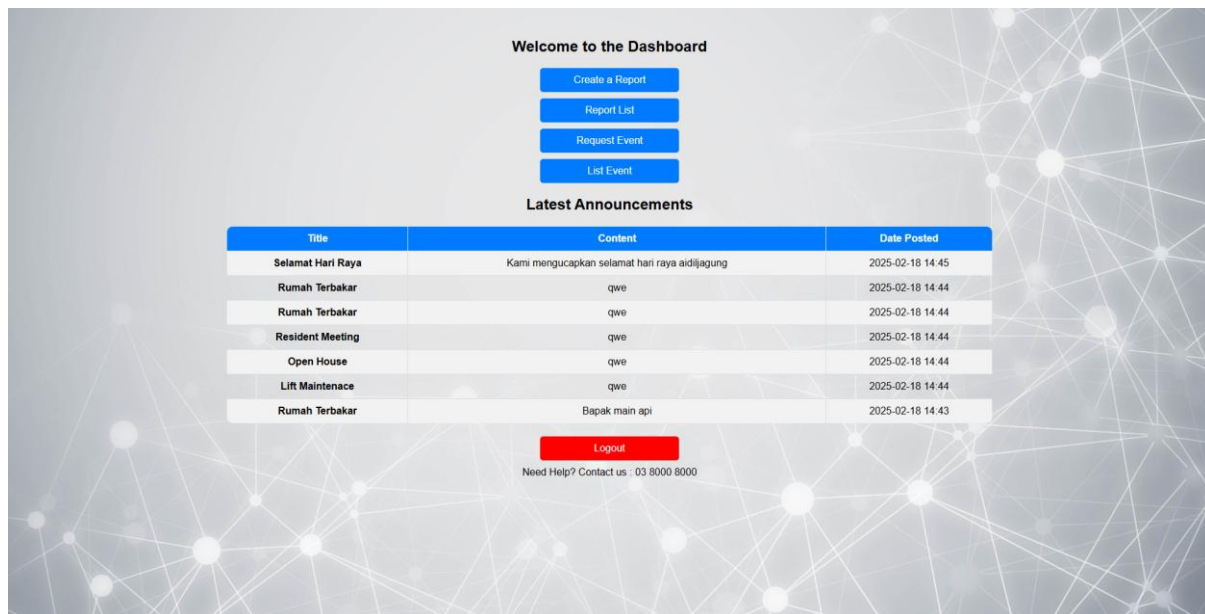
Safe Neighborhood Network – User Dashboard Interface

This screen serves as the central hub for registered neighborhood users, offering intuitive access to essential community features. The interface prioritizes clarity, usability, and quick interaction.
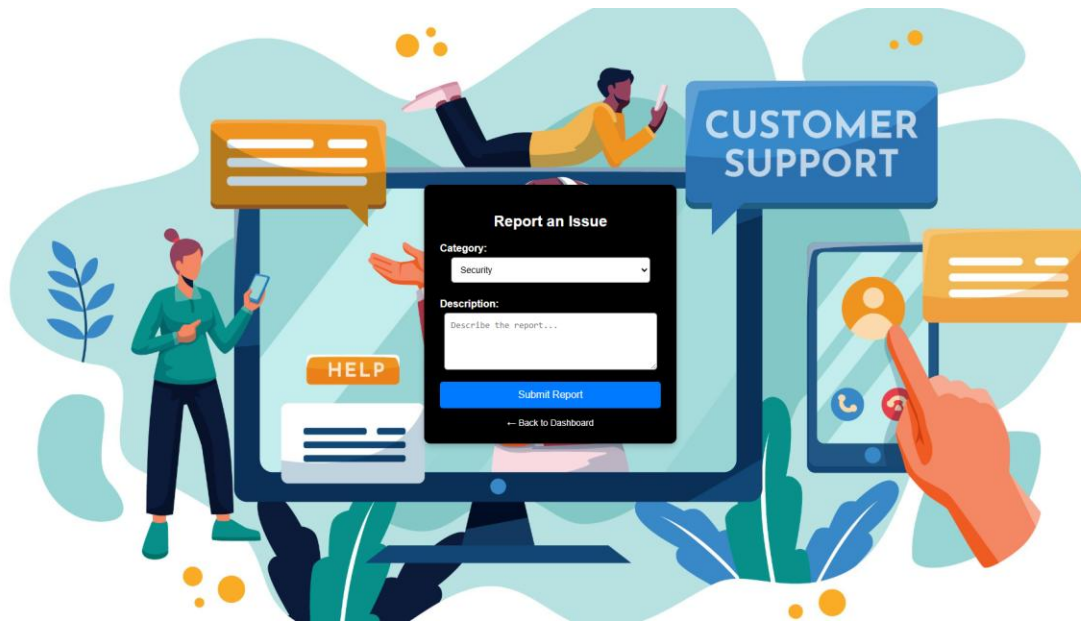
Core Functional Modules:

- **Create a Report:** Enables users to submit incident reports regarding local concerns or safety issues.
- **Report List:** Provides a personalized overview of previously submitted reports, allowing users to review status or details.
- **Request Event:** Allows residents to propose or schedule community gatherings, initiatives, or facility bookings.
- **List Event:** Displays a catalog of upcoming and past events in which the user has interest or involvement.

Auxiliary Functions:

- **Latest Announcements Panel:** Summarizes public messages relevant to all users, fostering community awareness.
- **Logout Option:** Easily accessible to ensure secure session termination.
- **Help Contact Message:** Offers support guidance and directs users to assistance channels.

Safe Neighborhood Network – User Create Report Interface

This module enables users to submit service-related problems or inquiries directly to neighborhood support, designed with user convenience and operational efficiency in mind.
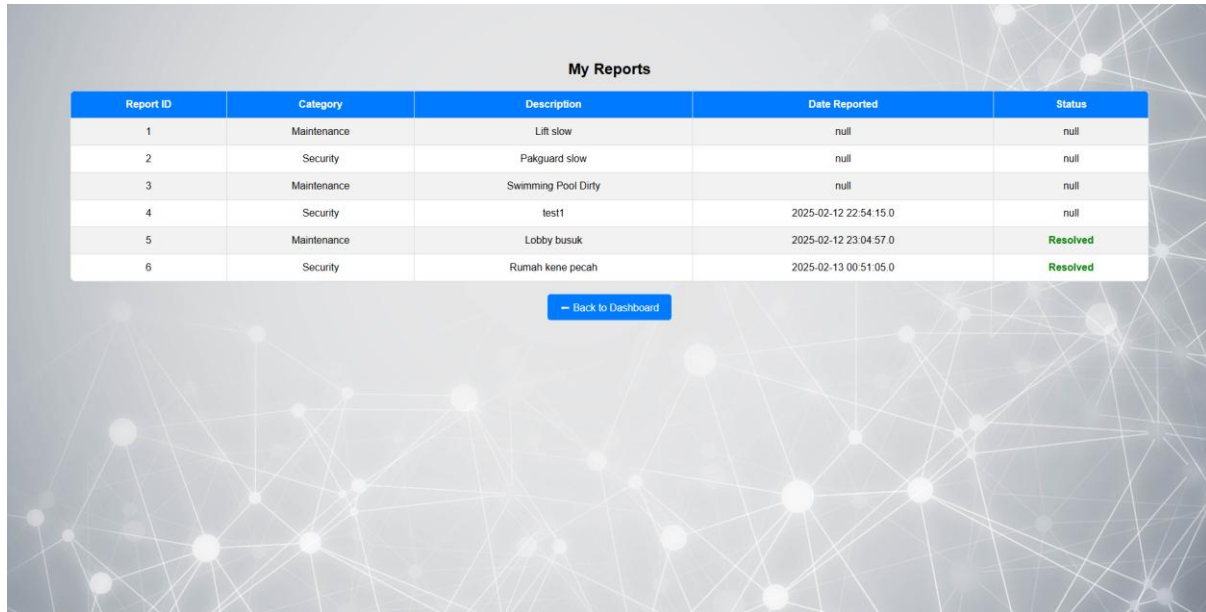
Functional Components:

- **Category Dropdown Field:** Allows users to select the type of issue being reported (e.g., maintenance, safety, amenities).
- **Description Text Area:** A multiline input field for detailed explanation of the user's concern.
- **Submit Button:** Triggers submission of the filled-out form to administrative backend services for processing.

Support-Oriented Visual Cues:

Though not referenced in description output, the layout reflects the system's commitment to accessible and responsive customer support, guiding the user through each step of the reporting process with intuitive form controls.
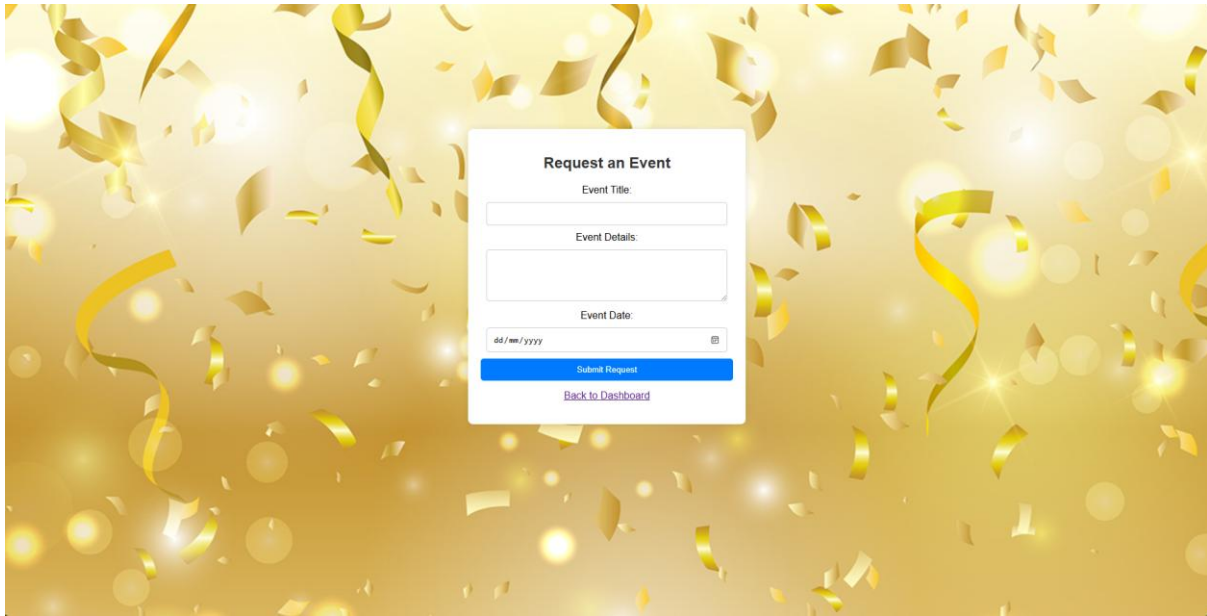
Safe Neighborhood Network – User Reports Interface

This module provides users with a personalized record of service issues they've submitted, offering transparency and tracking through a structured report table.

Core Functional Elements:

- **Report Table Columns:**
  - **Report ID:** Unique identifier assigned to each submitted issue
  - **Category:** Classification of the report type (e.g., Security, Maintenance)
  - **Description:** Brief explanation provided by the user regarding the issue
  - **Date Reported:** Timestamp indicating when the issue was submitted
  - **Status:** Reflects current resolution stage (e.g., Pending, Resolved)
- **Navigation Control:**
  - **Back to Dashboard Button:** Allows users to return to the main dashboard interface seamlessly

5.4.4   Request Create Event
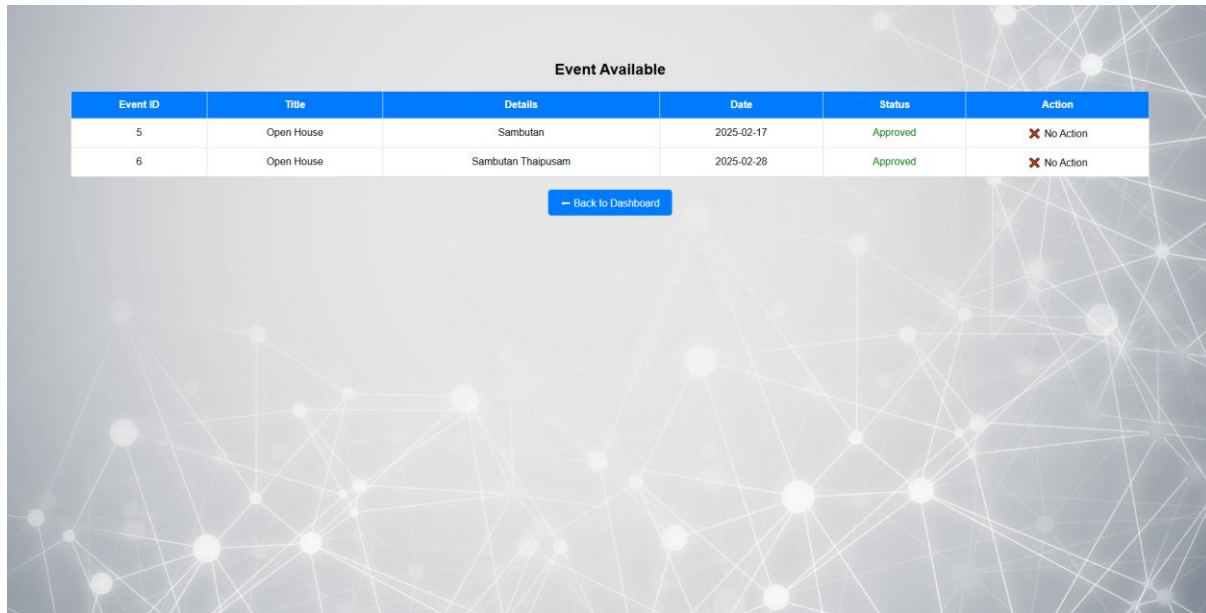
Safe Neighborhood Network – Request Create Event Interface

This module provides residents with a structured pathway to propose and request community-based activities. It supports transparent coordination between users and administrators while maintaining simplicity of input.

Functional Components:

- **Event Title Field:** A single-line input for users to specify the name or theme of the proposed event.
- **Event Details Text Area:** A larger text box for users to describe the event's purpose, logistics, or any special notes.
- **Event Date Picker:** Allows users to select the desired date of the event through a user-friendly calendar interface.
- **Submit Request Button:** Finalizes the request submission, triggering backend processing or administrative review.
- **Return Navigation Link:** Redirects users back to the main dashboard interface for continued interaction.

Safe Neighborhood Network – List Events Interface

This module presents users with a structured record of community events, offering a clear view into participation history and administrative status.

Functional Features:

- **Event Table Display:**
    - **Event ID:** A numeric identifier for backend tracking
    - **Title:** The name or theme of each scheduled community event
    - **Details:** Supplementary event information (e.g., celebration type, location highlights)
    - **Date:** Scheduled date of the event
    - **Status:** Indicates approval or scheduling stage (e.g., Approved, Pending)
    - **Action:** Field reserved for possible user actions or administrative review notes
- **Navigation Control:**
    - **Return Button:** A clearly labeled option to navigate back to the main dashboard

Purpose:

This interface supports transparency and user engagement by displaying the status and scope of events users have requested or are involved in, reinforcing community collaboration and administrative responsiveness.

6.0 CONCLUSION

The Safe Neighborhood Network platform demonstrates a well-structured application of web development technologies geared toward clarity, modularity, and user engagement. Each module from safety reports and event scheduling to administrative dashboard functions—is purposefully designed with discrete workflows and minimal interdependencies, allowing for smooth operation, documentation, and future scalability. The interface components emphasize clean navigation, consistent terminology, and straightforward user flows, making them suitable for technical specification and structured review.

Key technologies driving the platform include JavaServer Pages (JSP), HTML, Servlets, and the web.xml configuration file. JSP facilitates the generation of dynamic web content by embedding Java code directly within HTML, supporting real-time data availability and personalized user interactions. HTML forms the structural backbone of the user interface, providing semantic clarity and layout precision. Servlets are employed to handle backend processing and business logic, orchestrating request-response cycles and enabling secure, validated user actions. The web.xml descriptor governs deployment logic, servlet mappings, and security rules ensuring predictable system behavior and streamlined configuration management.

The platform is further strengthened by its content-agnostic design philosophy, where interfaces are documented based on structural and functional attributes rather than visual style or sample data. This approach supports clear documentation, reusable component descriptions, and rapid onboarding of technical reviewers or future developers. It also simplifies role-specific user experience definitions, allowing technical writers to differentiate between resident-facing and admin-facing modules without relying on contextual datasets.