**Iteration 01**

**Subject:** Software Engineering (CS-3009)  
**Project Name:** Smart Home Security & Monitoring System  
**Group:** 18-B  
**Group Members:**

* Fahad Jameel (i210394)
* Shaffin Imam (i212963)
* Ahmed Javeed (i221067)

**Deliverable:** Assignment 2 – Iteration-1 (RE Phase)  
**Submission Date:** March 22th, 2025



Table of Contents

[**1. Introduction** 4](#_Toc193478528)

[**1.1 Purpose** 4](#_Toc193478529)

[**1.2 Document Conventions** 4](#_Toc193478530)

[**1.3 Intended Audience and Reading Suggestions** 4](#_Toc193478531)

[**1.4 Product Scope** 4](#_Toc193478532)

[**2. Overall Description** 5](#_Toc193478533)

[**2.1 Product Perspective** 5](#_Toc193478534)

[**2.2 Product Functions** 5](#_Toc193478535)

[**2.3 User Classes and Characteristics** 5](#_Toc193478536)

[**2.4 Operating Environment** 5](#_Toc193478537)

[**2.5 Design and Implementation Constraints** 6](#_Toc193478538)

[**2.6 User Documentation** 6](#_Toc193478539)

[**2.7 Assumptions and Dependencies** 6](#_Toc193478540)

[**3. External Interface Requirements** 6](#_Toc193478541)

[**3.1 User Interfaces** 6](#_Toc193478542)

[**3.2 Hardware Interfaces** 6](#_Toc193478543)

[**3.3 Software Interfaces** 6](#_Toc193478544)

[**3.4 Communications Interfaces** 6](#_Toc193478545)

[**4. System Features** 7](#_Toc193478546)

[**Feature 1: User Registration and Login** 7](#_Toc193478547)

[**Feature 2: Remote Arm/Disarm** 7](#_Toc193478548)

[**Feature 3: Live Camera Feed and Notifications** 7](#_Toc193478549)

[**5. Other Nonfunctional Requirements** 8](#_Toc193478550)

[**5.1 Performance Requirements** 8](#_Toc193478551)

[**5.2 Safety Requirements** 8](#_Toc193478552)

[**5.3 Security Requirements** 8](#_Toc193478553)

[**5.4 Software Quality Attributes** 8](#_Toc193478554)

[**5.5 Business Rules** 8](#_Toc193478555)

[**6. Sprint Backlog (Iteration-1)** 9](#_Toc193478556)

[**6.1 Module Selected for Sprint** 9](#_Toc193478557)

[**6.2 Selected User Stories for Sprint 1** 9](#_Toc193478558)

[**6.3 Sub User Stories and Tasks** 9](#_Toc193478559)

[**6.4 Trello Board Snapshots** 10](#_Toc193478560)

[**6.5 Burn-Down Chart** 10](#_Toc193478561)

[**6.6 GitHub Repository** 10](#_Toc193478562)

[**7. Requirements** 10](#_Toc193478563)

[**7.1 Functional Requirements** 10](#_Toc193478564)

[**7.2 Non-functional Requirements** 10](#_Toc193478565)

[**8. Design** 11](#_Toc193478566)

[**8.1 Activity Diagrams** 11](#_Toc193478567)

[**8.2 Use Case Diagram** 13](#_Toc193478568)

[**8.3 Sequence Diagrams** 14](#_Toc193478569)

[**8.4 Class Diagram** 16](#_Toc193478571)

[**9. Appendices** 18](#_Toc193478572)

[**Appendix A: Glossary** 18](#_Toc193478573)

[**Appendix C: To Be Determined (TBD) List** 18](#_Toc193478574)

**1. Introduction**

**1.1 Purpose**

This Software Requirements Specification (SRS) document specifies the requirements for the Smart Home Security & Monitoring System. It details both the functional and non-functional requirements and serves as the reference for the design, development, and validation of the system for Iteration-1 (Requirements Engineering Phase).

**1.2 Document Conventions**

* **Priority Levels:** High (H), Medium (M), Low (L)
* **Requirement IDs:** Prefixed with “REQ-” followed by a unique number.
* **Diagrams:** Provided as separate image files and referenced here.

**1.3 Intended Audience and Reading Suggestions**

This document is intended for:

* **Developers:** To understand the detailed functional requirements and design constraints.
* **Testers:** To base test cases on the defined functional requirements.
* **Project Managers & Stakeholders:** To review system scope, priorities, and progress.  
  Readers are suggested to first review the Introduction and Overall Description before delving into the detailed requirements and design sections.

**1.4 Product Scope**

The Smart Home Security & Monitoring System provides homeowners with a flexible, cost-effective way to secure and monitor their homes. It integrates various sensors, cameras, and smart devices, enabling real-time alerts, remote control, and detailed activity logs. The system is designed to be user-friendly and easily integrated with existing alarm services.

**2. Overall Description**

**2.1 Product Perspective**

The product is a new, standalone system that integrates with IoT sensors (motion, door/window, cameras) and third-party alarm services. It will complement existing home automation systems and is designed with modularity in mind for future expansion.

**2.2 Product Functions**

* **User Management:** Registration, login, and profile management
* **Sensor Configuration:** Adding sensors (motion, door/window) and configuring device settings
* **Real-Time Monitoring:** Live camera feeds, push notifications, and activity logs
* **Remote Control:** Arm/disarm system remotely, schedule device operations
* **Integration:** Connect with third-party alarm services and smart devices

**2.3 User Classes and Characteristics**

* **Homeowners:** Primary users with full access to system functionalities
* **Family Members/Guests:** Limited access as assigned by the homeowner
* **System Administrators:** Responsible for maintenance, updates, and security oversight

**2.4 Operating Environment**

* **Hardware:** IoT sensors, smart cameras, smart locks
* **Software:** Mobile application (iOS/Android), web dashboard, backend server
* **Network:** Internet and local network connectivity for sensor communication

**2.5 Design and Implementation Constraints**

* **Integration:** Must seamlessly interface with third-party alarm systems.
* **Hardware:** Limited memory and power on IoT devices require optimized code.
* **Security:** Strong authentication and encryption protocols are mandatory.

**2.6 User Documentation**

* User Manual (online and PDF format)
* Quick-start Guide and Video Tutorials
* FAQs and Troubleshooting Guide on the website

**2.7 Assumptions and Dependencies**

* Availability of reliable internet connectivity
* Third-party APIs for alarm service integration are stable and documented
* Homeowners have basic technical literacy to manage mobile apps

**3. External Interface Requirements**

**3.1 User Interfaces**

The system includes:

* **Mobile App:** Simplified UI for on-the-go monitoring and control
* **Web Dashboard:** Detailed interface for configuration and system status review
* **Interactive Alerts:** Push notifications and SMS/email alerts for immediate updates

**3.2 Hardware Interfaces**

* **Sensors and Cameras:** Communication via Wi-Fi/Zigbee protocols
* **Smart Devices:** Integration with smart locks and lighting systems via proprietary APIs

**3.3 Software Interfaces**

* **Third-Party Alarm Systems:** RESTful API integration
* **Database:** SQL/NoSQL backend for storing user profiles, sensor data, and logs
* **Cloud Services:** For real-time alerts and video storage

**3.4 Communications Interfaces**

* **Network Protocols:** HTTP/HTTPS for web services, MQTT for sensor data transmission
* **Data Formats:** JSON for data exchange between devices and server

**4. System Features**

For Iteration-1, the focus is on key features that provide a foundation for the system. The following selected user stories from the project backlog will be implemented:

**Feature 1: User Registration and Login**

* **Description & Priority:**  
  Allow new users to create an account and existing users to securely log in (High).
* **Sub User Stories:**
  + **User Registration:** As a new homeowner, I want to register my account to access the system.
  + **User Login:** As a returning user, I want to log in to manage my home security.
* **Functional Requirements:**
  + REQ-101: The system shall allow a new user to register with a valid email and password.
  + REQ-102: The system shall validate credentials during login and display an error message on failure.

**Feature 2: Remote Arm/Disarm**

* **Description & Priority:**  
  Enable users to arm or disarm their security system remotely (High).
* **Sub User Stories:**
  + **Arm/Disarm System:** As a homeowner, I want to control the system remotely for added flexibility.
* **Functional Requirements:**
  + REQ-201: The system shall allow remote arming/disarming via the mobile app.
  + REQ-202: The system shall confirm the action and update the system status in real time.

**Feature 3: Live Camera Feed and Notifications**

* **Description & Priority:**  
  Provide real-time video streaming and alerts in case of anomalies (Medium).
* **Sub User Stories:**
  + **Live Camera Feed:** As a homeowner, I want to view a live camera feed for instant monitoring.
  + **Push Notifications:** As a homeowner, I want to receive immediate alerts for unusual activities.
* **Functional Requirements:**
  + REQ-301: The system shall stream live video from cameras to the mobile app.
  + REQ-302: The system shall send push notifications for predefined events.

*Additional features and sub-stories like sensor configuration, activity logs, can be incorporated in future iterations.*

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

* The system shall process user requests within 2 seconds under normal operating conditions.
* Live video streaming latency should not exceed 3 seconds.

**5.2 Safety Requirements**

* The system shall include fail-safe mechanisms to avoid false alarms in emergency scenarios.
* Regular system health checks must be performed to ensure device functionality.

**5.3 Security Requirements**

* User data must be stored using encryption.
* All communications between devices and servers must be secured via HTTPS.
* Multi-factor authentication (MFA) is recommended for user login.

**5.4 Software Quality Attributes**

* **Usability:** Intuitive UI for both novice and experienced users.
* **Maintainability:** Modular design for easier updates and bug fixes.
* **Reliability:** The system must have 99.9% uptime.
* **Scalability:** Must support an increasing number of devices and users without performance degradation.

**5.5 Business Rules**

* Only registered homeowners and authorized users can access sensitive features (e.g., live feed, remote control).
* System alerts and actions must comply with local security and privacy regulations.

**6. Sprint Backlog (Iteration-1)**

**6.1 Module Selected for Sprint**

**Module Name:** User Management and Security Control

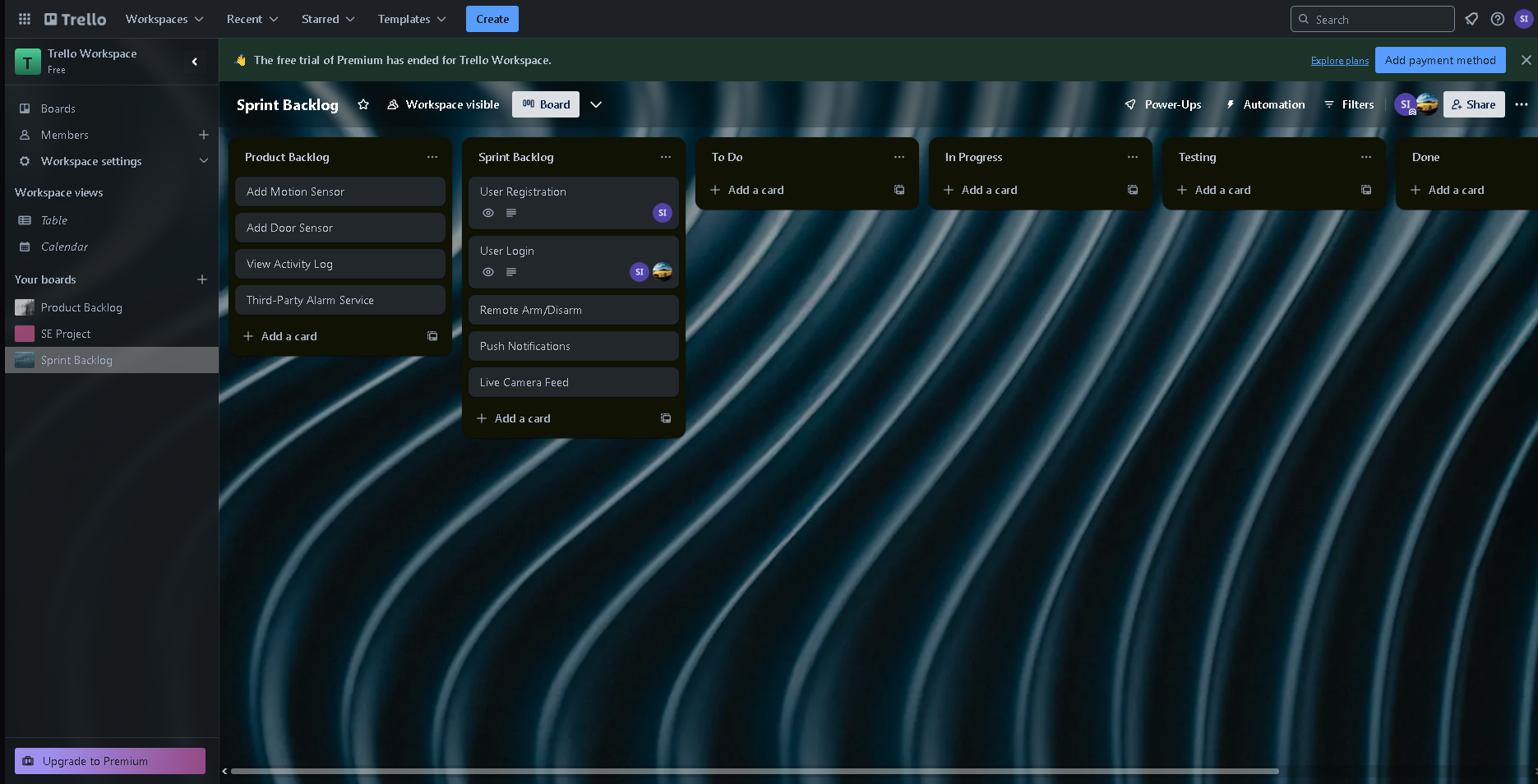
**6.2 Selected User Stories for Sprint 1**

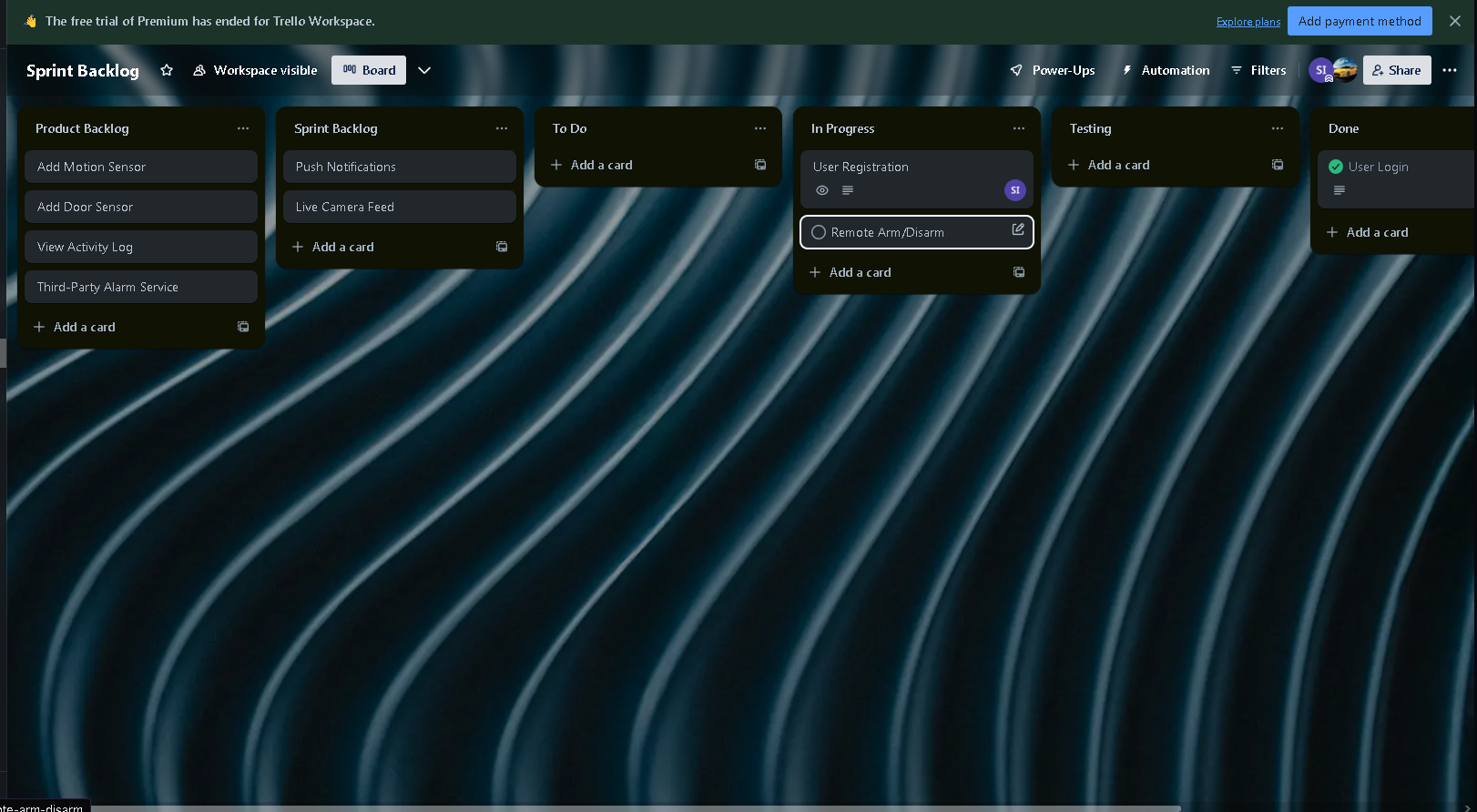
* **User Registration & Login:**
  + Implement registration and login functionality (REQ-101, REQ-102).
* **Remote Arm/Disarm:**
  + Enable users to arm/disarm the system remotely (REQ-201, REQ-202).
* **Live Camera Feed & Notifications:**
  + Integrate live streaming and push notification functionality (REQ-301, REQ-302).

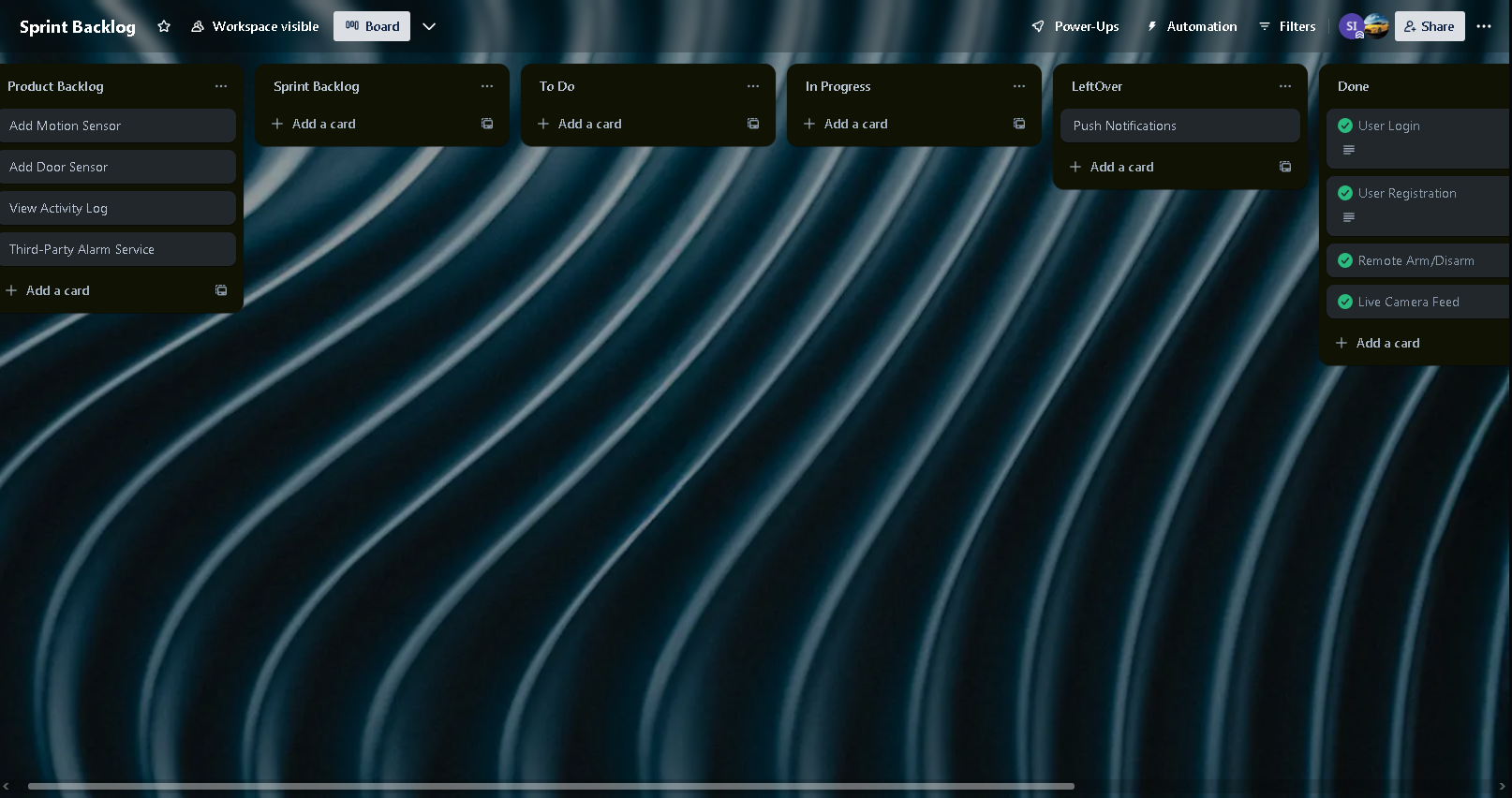
**6.3 Sub User Stories and Tasks**

* **User Registration:**
  + Task 1: Design registration UI screen.
  + Task 2: Develop backend registration API.
  + Task 3: Implement input validation and error handling.
* **User Login:**
  + Task 1: Design login screen.
  + Task 2: Develop login API with session management.
* **Remote Arm/Disarm:**
  + Task 1: Create UI controls for arming/disarming.
  + Task 2: Integrate system status update.
* **Live Camera Feed & Notifications:**
  + Task 1: Develop live streaming interface.
  + Task 2: Configure push notifications and event triggers.

**6.4 Trello Board Snapshots**

**

**

**

**6.6 GitHub Repository**

Link: <https://github.com/Fahad-Jameel/SmartHomeSecurity>

**7. Requirements**

The detailed requirements for Iteration-1 are captured below.

**7.1 Functional Requirements**

* **Registration and Login:**
  + REQ-101: User registration with email and password.
  + REQ-102: Secure login with error messaging.
* **Remote Control:**
  + REQ-201: Remote arming/disarming functionality via the app.
  + REQ-202: Real-time status updates after control actions.
* **Live Streaming and Alerts:**
  + REQ-301: Live streaming of camera feeds.
  + REQ-302: Push notifications for system events.

**7.2 Non-functional Requirements**

* **Performance:** Fast response time (≤ 2 seconds) for user actions.
* **Security:** End-to-end encryption for all data transmissions.
* **Reliability:** System uptime of 99.9% with robust error handling.
* **Usability:** User-friendly interface with minimal learning curve.

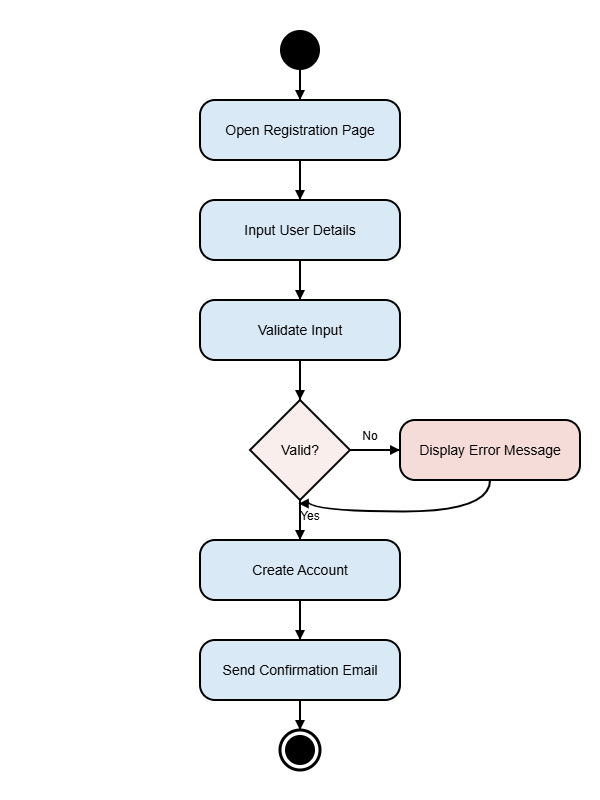
**8. Design**

This section includes a description of the system diagrams.

**8.1 Activity Diagrams**

**Processes Covered:**

* **User Registration Process:**



* **Remote Arm/Disarm Process:**

A diagram of a system

AI-generated content may be incorrect.

* **Live Camera Feed Process:**

A diagram of a system

AI-generated content may be incorrect.

**8.2 Use Case Diagram**

The diagram represents the interactions between the following actors and the system:

* Homeowner (Primary actor)
* Guest/Family Member (Secondary actor with limited access)
* System Administrator (For maintenance and security monitoring)

A diagram of a smart home security system

AI-generated content may be incorrect.

**8.3 Sequence Diagrams**

**Sequence diagrams for key activities:**

* **Registration/Login Sequence:**

A diagram of a software application

AI-generated content may be incorrect.

* **Remote Arm/Disarm Sequence:**

A diagram of a software project

AI-generated content may be incorrect.

* **Live Feed Sequence:**

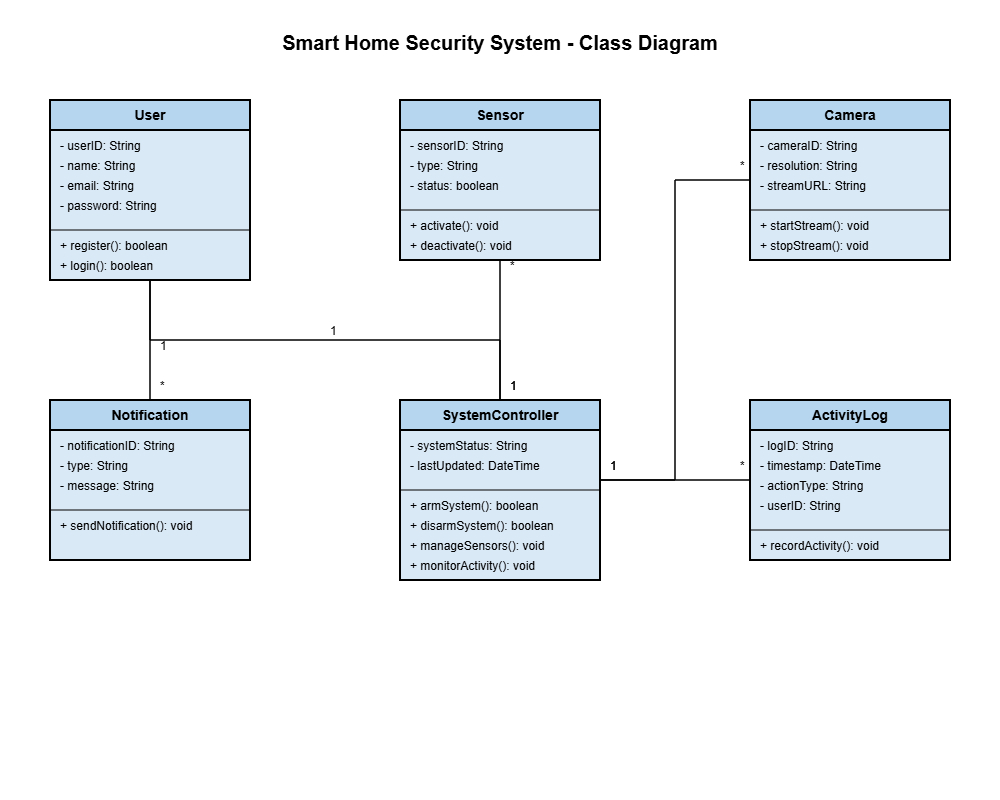
A diagram of a computer process

AI-generated content may be incorrect.

**8.4 Class Diagram**

A class diagram includes the following classes:

* **User:** Attributes (userID, name, email, password), Methods (register(), login(), updateProfile())
* **Sensor:** Attributes (sensorID, type, status), Methods (activate(), deactivate(), updateStatus())
* **Camera:** Attributes (cameraID, resolution, streamURL), Methods (startStream(), stopStream())
* **Notification:** Attributes (notificationID, type, message), Methods (sendNotification())
* **SystemController:** Manages user sessions, sensor status, and device integrations.

**

**9. Appendices**

**Appendix A: Glossary**

* **IoT:** Internet of Things
* **MFA:** Multi-Factor Authentication
* **API:** Application Programming Interface

**Appendix C: To Be Determined (TBD) List**

* TBD-1: Finalize integration details with third-party alarm services.
* TBD-2: Performance testing criteria for live video streaming.