

**SOEN-343 Software Architecture and Design – Lab Section: WK-X**

**Project – Phase 1 Report**

**Presented to**

**The Department of Gina School of Engineering and Computer Science**

**Concordia University**

**by**

**Joseph Pagliuca                      ID: 40092947**

**Tanzir Hoque                        ID: 40210275**

**Mohamed Nemroud                ID: 40153847**

**Anh-Tuan Nguyen                 ID: 40177349**

**Kevin Phan                          ID: 40097439**

**Sabari Krishna Orakkan          ID: 40079144**

**Concordia University**

**February 12<sup>th</sup>, 2024**

## **Table of Contents**

<b>1.1 Problem definition</b>	<b>3</b>
1.1.1 Problem Statement	3
1.1.2 Product Position Statement	3
1.2 Product Overview	4
1.2.1 Product Perspective	4
1.2.2 Assumptions and Dependencies	4
<b>2. Technology Used</b>	<b>5</b>
2.1 Control Version System	5
2.2 Team Collaboration	5
2.3 Monitoring and Verification	5
2.4 Designing and Modeling Work	5
2.5 Development Framework	5
2.6 Coding	5
<b>3. Context Diagram</b>	<b>7</b>
<b>4. Domain Model</b>	<b>8</b>
<b>5. References</b>	<b>9</b>

## 1.1 Problem definition

### 1.1.1 Problem Statement

The smart home simulator addresses the challenge of managing complex smart home systems. It creates a tool to facilitate the programming, testing, and interaction of various smart home devices in a virtual environment.

The problem of	Difficulties in managing several smart home systems due to programming complexities
Affects	Homeowners, students, researchers, practitioners, and anyone interested in smart home technology
The impact of which is	Ineffective use of smart home systems and underutilization of resources due to lack of modularity
A successful solution would be	A user-friendly stimulator that simplifies understanding and management of smart home systems. An effective way to utilize all resources and enhance user education

### 1.1.2 Product Position Statement

For	Homeowners, Researchers, and specialists in the field of smart home technology
Who	For a comprehensive understanding and control over smart home systems
The SmartScape	IoT/Home Automation
That	Provides proper modularity towards the individual components of a smart-home system and provides ease of use. The smart-home system can also enhance the security of a household.
Unlike	Other apps which have a separate system for each individual component.
Our product	Our app provides direct and easy access to all components in a single application, while keeping modularity and providing security to a household.

## 1.2 Product Overview

### 1.2.1 Product Perspective

The Smart-Home System is a product that relies on third-party pre-installed smart devices in a household and provides a singular interface that allows for easy access to each individual smart component in a house. The system is independent of any other third party product, but instead allows for communication between said products and the software.

### 1.2.2 Assumptions and Dependencies

Assumptions	Dependencies
Each component has a communication interface	An Internet connection or Bluetooth
A device that can install the Smart-Home System	Windows/Mac/Linux/Android/iOS
Each house has a different layout	A template of the house's layout

## 2. Technology Used

### 2.1 Control Version System

**GitHub** is one of the most well-known developer platforms and will be our choice of version control for this team project. It provides a seamless platform easy for collaboration and allows team members to work concurrently on the same project without overwriting each other's changes. Additionally, **GitHub** has branching and merging capabilities making the workflow management more efficient, easing the development process while reducing conflicts. Lastly, all changes made are tracked and displayed on the platform [1].

### 2.2 Team Collaboration

**Discord** is our choice of platform to hold weekly team meetings during the development phase of this team project due to its versatile features and its easy accessibility. It delivers a user-friendly interface and diverse ways of communication channels via text, voice and video allowing real-time interaction regardless of the different time zones. **Discord** also has customizable roles and permissions systems for efficient organization and team management. Overall, this platform provides a dynamic hub for team collaboration and engagement [2].

### 2.3 Monitoring and Verification

**JUnit 5** will be used as our testing framework since it is best used with systems coded with Java. We decided to use JUnit 5 as it provides a simple and intuitive way to test our code by having us create the tests manually for each function our program has. This will allow us to manually see which function is causing any problems if any arise [3].

**Jenkins** will be used for automating the build, test, deployment processes, enabling continuous integration and continuous delivery (CI/CD) to streamline and accelerate the development lifecycle. It is chosen for its extensive plugin ecosystem, flexibility, and wide adoption, making it highly customizable and compatible with several development tools and environments [4].

### 2.4 Designing and Modeling Work

**Draw.io** is a well-known free diagram software that offers numerous templates and symbols that will be used extensively in this project in order to make flowcharts, UML diagrams and other visual artifacts as necessary throughout the duration of this project. It is used in this project as it is simple and intuitive to use while still offering powerful tools for diagramming [5].

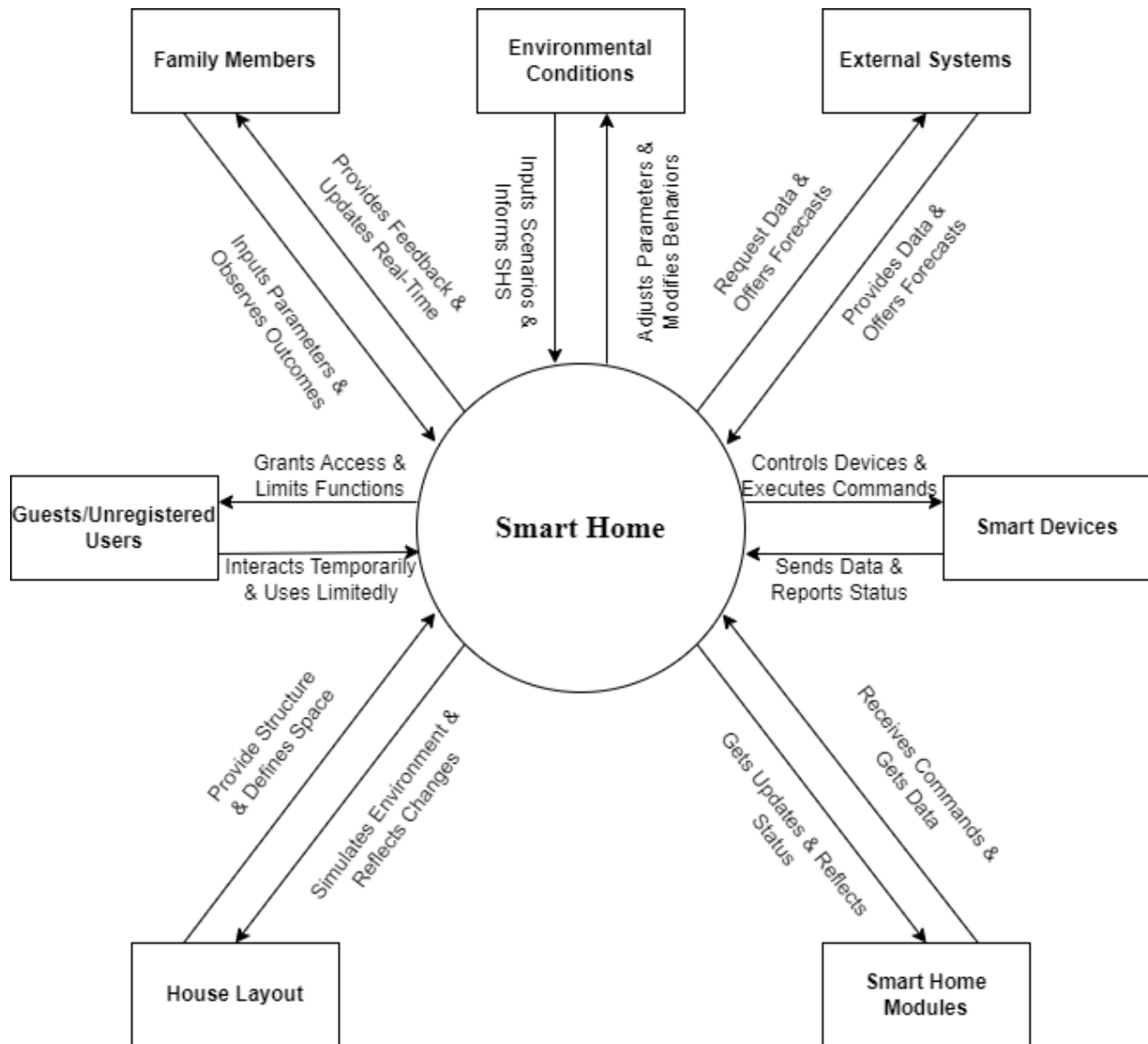
### 2.5 Development Framework

The development framework is set to be in Java **Swing** as it provides a comprehensive set of tools for creating user interfaces. Swing also allows developers to create cross-platform integrations allowing further modularity for further scope of the project [6].

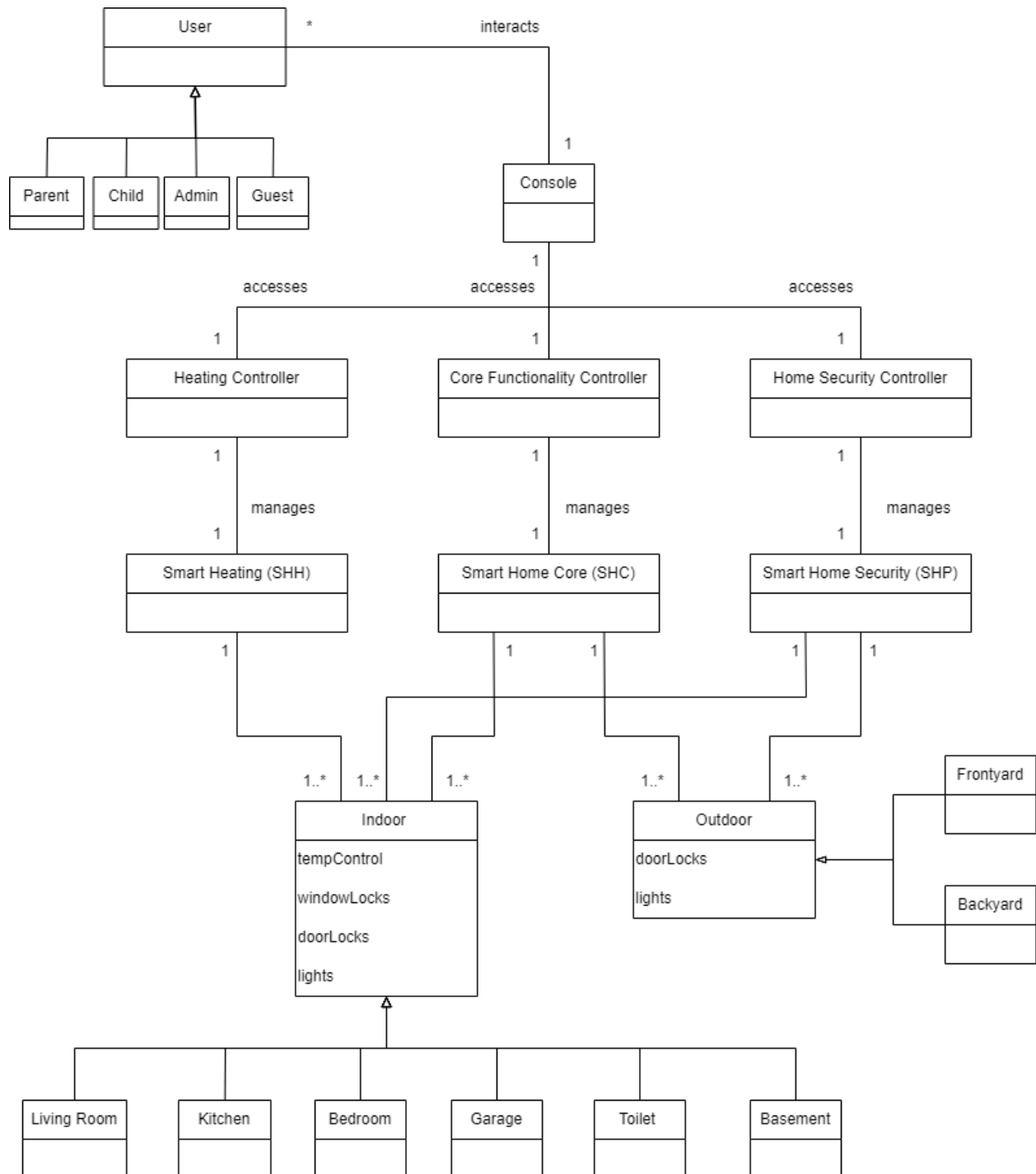
### 2.6 Coding

**Java** is the coding language that was chosen for its ease of use. It is a strongly typed Object-oriented language that has a good balance between performance and abstraction making it better suited for developing our application. Java has many great features such as a multitude of inbuilt classes, memory-safety, and good portability [7].

### 3. Context Diagram



## 4. Domain Model



## 5. References

- [1] “Let’s build from here,” GitHub, <https://github.com/> (accessed Feb. 11, 2024).
- [2] Discord, <https://discord.com/> (accessed Feb. 11, 2024).
- [3] “JUnit,” JUnit 5, <https://junit.org/junit5/> (accessed Feb. 11, 2024).
- [4] Jenkins, <https://www.jenkins.io/> (accessed Feb. 11, 2024).
- [5] “drawio,” draw.io, <https://www.drawio.com/> (accessed Feb. 11, 2024).
- [6] “Trail: Creating a GUI with swing,” Trail: Creating a GUI With Swing (The Java™ Tutorials), <https://docs.oracle.com/javase/tutorial/uiswing/> (accessed Feb. 11, 2024).
- [7] “Java,” Java.com, <https://www.java.com/en/> (accessed Feb. 11, 2024).