EPI-USE Labs

Securitree- Recruiting Exercise

[Company name] | [Company address]

Overview

Timothy Cloete

[Year]

Contents

[Exercise Overview 2](#_Toc74922265)

[Problem Overview 2](#_Toc74922266)

[Technical Requirements 2](#_Toc74922267)

[Core Requirements 2](#_Toc74922268)

[Application Screens 2](#_Toc74922269)

[Overview of my proposed application system. 3](#_Toc74922270)

[Technical Requirements 3](#_Toc74922271)

[Core Requirements 3](#_Toc74922272)

[Application Screens 3](#_Toc74922273)

[Instructions on how to run the program. 3](#_Toc74922274)

[How does the application works? 3](#_Toc74922275)

# Exercise Overview

## Problem Overview

Super Secure Systems (S3) is a well-established security firm specialising in physical security and access control management for enterprise-level clients. Their products range from simple door locks to high-end camera systems that feature motion tracking, facial and license plate recognition, and pattern analysis and matching.

As part of their continuous drive to secure, contain, and protect their client’s physical assets, S3 has approached your development start-up to assist with the implementation of a new application called SecuriTree.

SecuriTree is an access control management application that provides a visual tree view of the security and access control units installed in a client’s S3 system. This application will allow authorised security operatives to monitor and manage each physical security and access control unit (areas, doors, elevators, floors, etc.), at a client’s premises from one central location.

Therefore, I need to develop an application that will fulfil the needs of SecuriTree.

## Technical Requirements

Allowed programming languages:

* ABAP
* Java
* JavaScript/TypeScript
* Python
* C++
* C#

User Interface Examples:

* Command-line text interface
* Desktop application interface
* Web page interface
* AR/VR application interface
* Etc.

## Core Requirements

The application must provide the following core functionality:

* View Security Entity Hierarchy
* Door Management
* Secure User Access Management System
* Persistent Secure Data Storage

## Application Screens

The application must include the following screens:

* Login Screen
* Home Screen
* View Hierarchy
* Manage Doors
* Log out.

# Overview of my proposed application system.

## Technical Requirements

For programming language, I decided on using:

C#.

For user interface, I decided on using:

Command-line text interface.

## Core Requirements

My program consists out of the following core functionalities:

* View Security Entity Hierarchy
* Door Management
* Secure User Access Management System
* Persistent Secure Data Storage

## Application Screens

My application includes the following screens:

* Login Screen
* Home Screen
* View Hierarchy
* Manage Doors
* Log out.

# Instructions on how to run the program.

Simply the user will need to install Microsoft Visual Studios.

GitHub will be required to gain access to the program. The program can be downloaded and open using Visual studios

# How does the application works?

With main program, the user will be welcomed to the Securitree dashboard where the program will request the user for their username and password. The program will make use of the validate method to validate the username and password provided by the user. The program will notify the user whether the information provided are incorrect and if incorrect the program will request the user to enter again their user login details.

When the user provided the correct information, the user will be taken to the menu screen. Menu screen consists out of view hierarchy, manage doors and log out.

View hierarchy will show the user the security entity hierarchy screen. User will be able to return to the menu by pressing enter on the keyboard.

Manage doors section will give the user an option to lock or unlock a door. Program will request to user for the door id to change the door’s status depend on which one the user chooses.