UNIVERSITY OF WOLLONGONG AUSTRALIA

Bachelor of Computer Science

(Digital System Security)

CSCI321 – Final Year Project

System Architecture Document

Project Particulars

Supervisor	Dr Ta Nguyen Binh Duong
Project Group	SS18/1F
Project Title	Two Factor Authentication

Project Team's Particulars

Student Number	Student Name	Email Address
5363536	Koh Hong Wei	hwkoh003@mymail.sim.edu.sg
5710923	Chua Han Ming Adler	hmachua002@mymail.sim.edu.sg
5711356	Ong Wei Hao	whong012@mymail.sim.edu.sg

A secure file locking application

Table of Contents

1. Introduction

The System Architecture Document(SAD) aims to document the architecture of the go2FA system.
It includes:

- go2FA's system general description
- Logical architecture of the software, layers and top-level components
- Physical architecture of the hardware
- Technical choices made

2. Software Architecture

2.1 Architecture Overview

The system works in a windows computer environment. The user is required to have his mobile phone together with his computer to lock or unlock the files on the computer. The above consideration is taken into account when developing the system. The architecture takes these functionalities into consideration

- User authentication
- Create Account
- Account Recovery
- Lock File
- Unlock Files
- User Account Management

2.2 Architecture Design Pattern

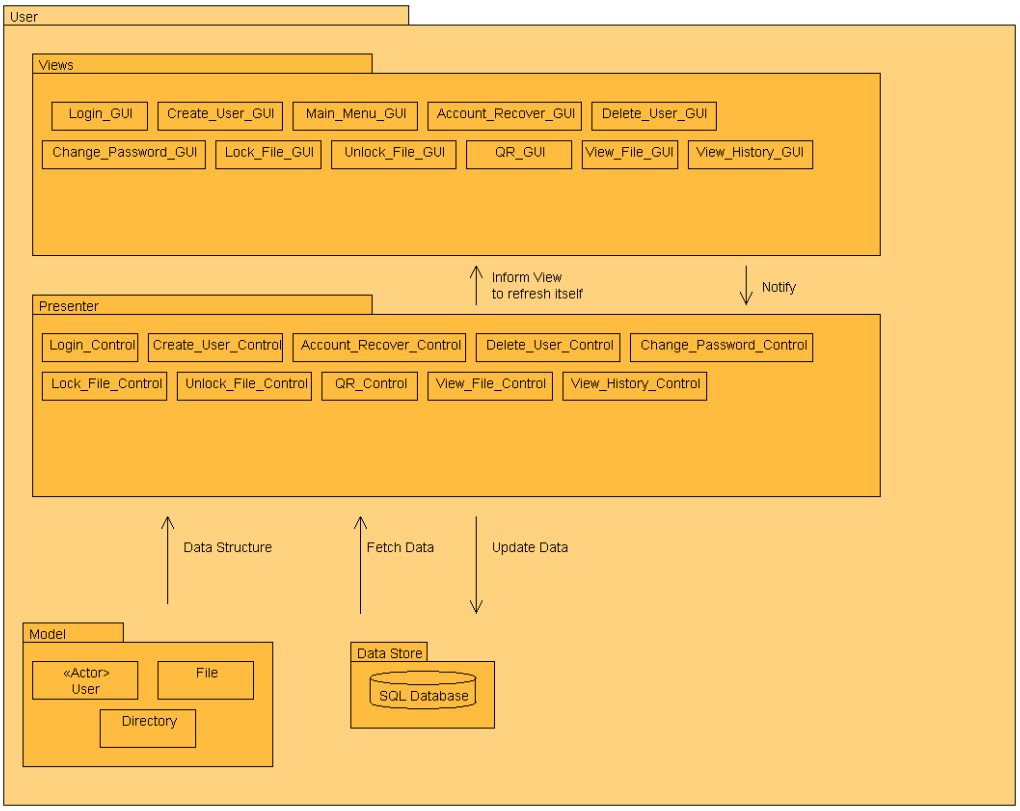
After much considerations, we have decided to adopt Model View Pattern for our system.

2.2.1 Why MVP pattern?

MVP is an architectural pattern derived from the Model View Controller (MVC). In the MVC, the controller is tied tightly to the APIs thus resulting in difficulties when conducting unit tests. In MVC, the views and controllers are tightly coupled together. This causes many avalanche effects when updating the views as the controls need to be updated too.

2.3 Logical Architecture Overview

2.3.1 Layer Diagram



View

This layer is a passive interface that displays information to the user. It relays user commands to the presenter to act upon the request.

Presenter

This layer acts upon the model and view layers. It's main role in the system is to retrieve data from repositories and process information to display to the users.

Model

This layer is an interface defining the data to be displayed. It does all the necessary backend operations relating to the business logic.

2.3.2 Logical Design Qualities

Adopting the MVP architectural pattern, we aim to achieve the following qualities:

- High Cohesion
    - Each modules in our system has functions and elements that are strongly related only to fulfil one specific task.
- Low Coupling

    - Modules are designed to be loosely coupled so that changes in one module will not result in the need of change in other modules.
- Modularity
    - We aim to separate the functions of the program into independent modules such that each modules only contain everything necessary to execute one and only one aspect of the program.

2.4 Physical Architecture

2.4.1 Physical Architecture Overview

2.4.2 Deployment Diagram

3. System Architecture Qualities

3.1 Availability
System should be available to intended users whenever they wish to lock or unlock a file

3.2 Confidentiality
Locked files can only be unlocked by the user who locks it.

3.3 Integrity
Locked files cannot be accessed by unauthorized users.

3.4 Extensibility
Adding new features to our system will not affect the current functionalities of the system.

3.5 Usability
Our system aims to adopt the KISS model, keeping it simple and straightforward to avoid user complications.

3.6 Compatibility
Our proposed system will be able to run on various versions of Windows and Android.

4. Appendix A: Glossary

5. References
[1] https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93presenter