Secure Backup Software System
Analysis and Design Document

Student: Ciucescu Vlad Andrei
Group: 30432

Revision History

Date	Version	Description	Author
04/Apr/18	1.0	First version of the project analysis and design document	Ciucescu Vlad
19/Apr/18	1.1	Added domain model, updated conceptual architecture, package diagram, deployment diagram, component diagram	Ciucescu Vlad

Table of Contents

I. Project Specification

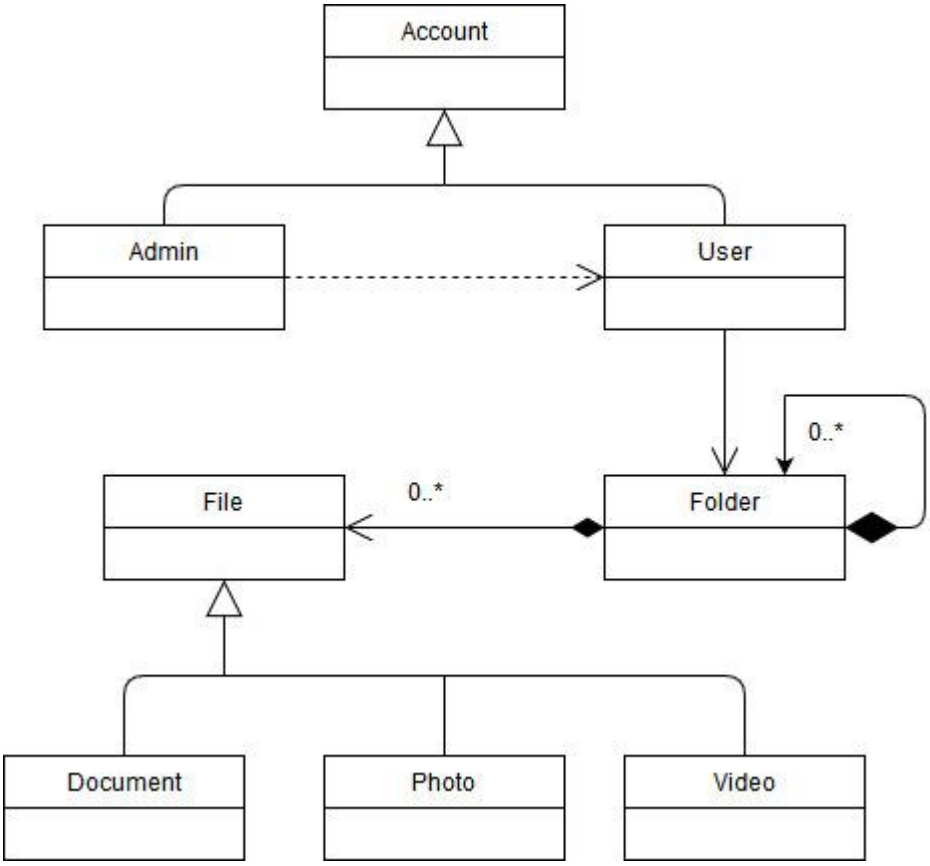
A software backup system should be implemented, which allows users to store files, documents, images and videos through a windows desktop application in a secure manner. A separate folder should be created for each user for them to store their files in. This folder should be only accessible to authorized users. All the details of the users, user activity and files will be stored in a RDBMS.

There will exist two types of users:

- Administrator – has the responsibility of managing user accounts (creating new accounts, deleting accounts at the request of users, blocking a user etc.)
- Regular User – can use the software to store and retrieve files. Can access their account after an administrator has created it and can request an administrator to close their account.

II. Elaboration – Iteration 1.1

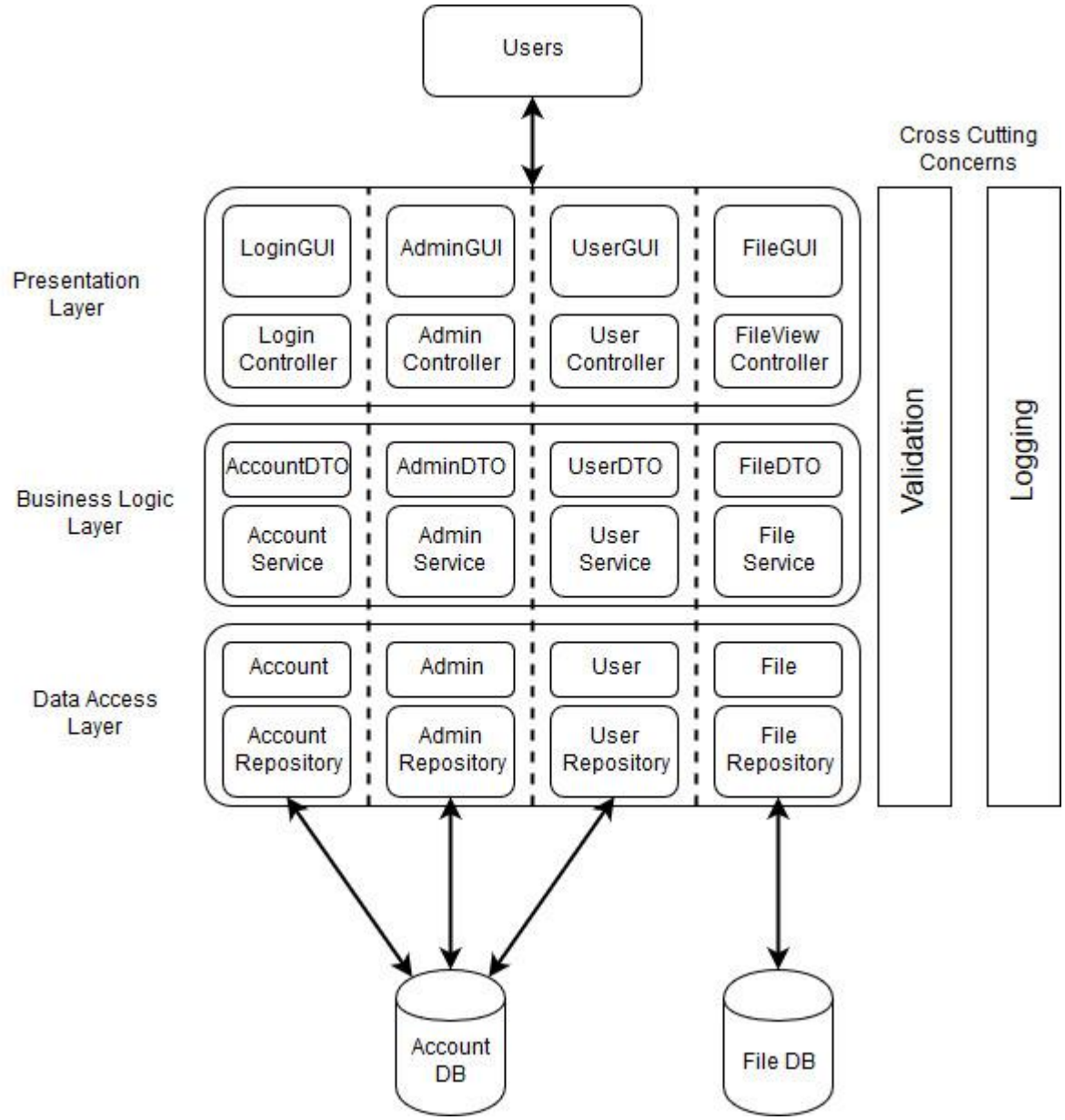
1. Domain Model



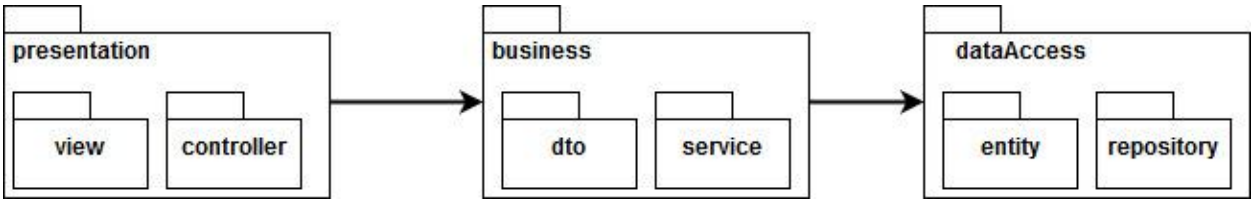
2. Architectural Design

2.1 Conceptual Architecture

The proposed architecture of the application is the layers architecture. This aids in separating the responsibilities, because each layer is designed to perform a specific role within the application. As such, each layer will be cohesive and easily replaceable, should the need arise. This system will be composed of three layers: data access layer, business logic layer and presentation layer.

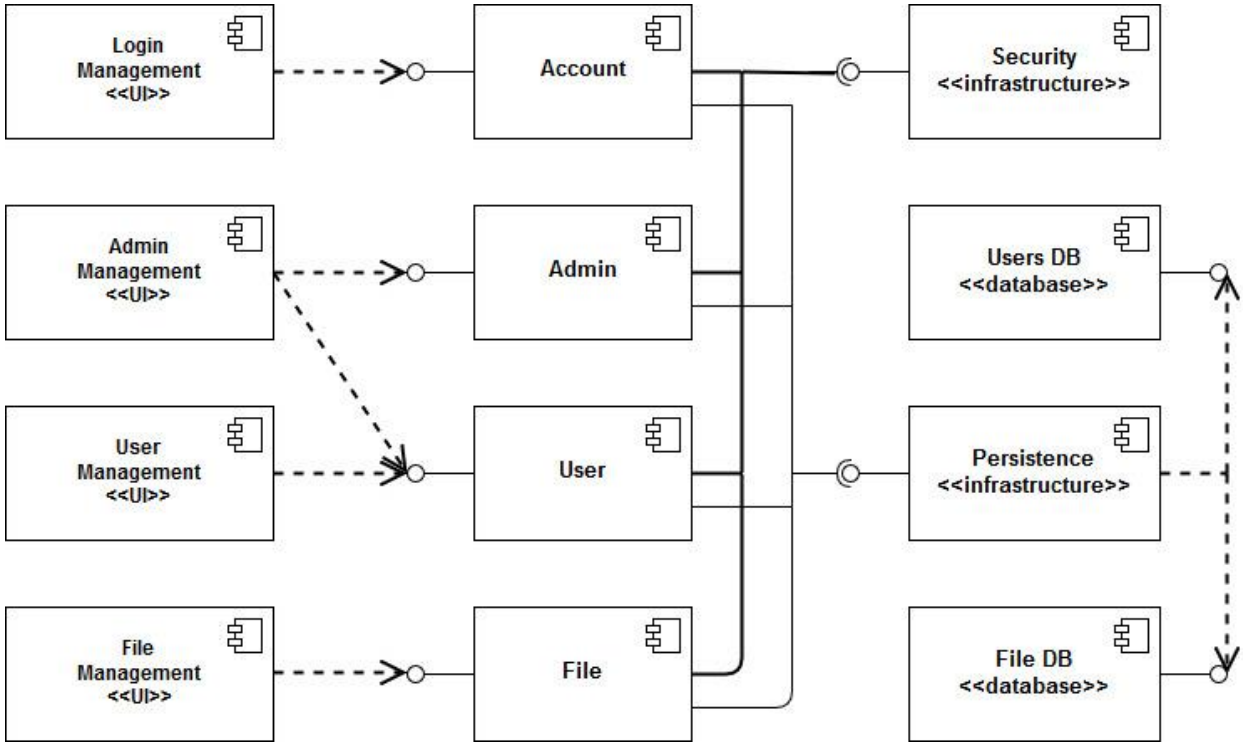


2.2 Package Design

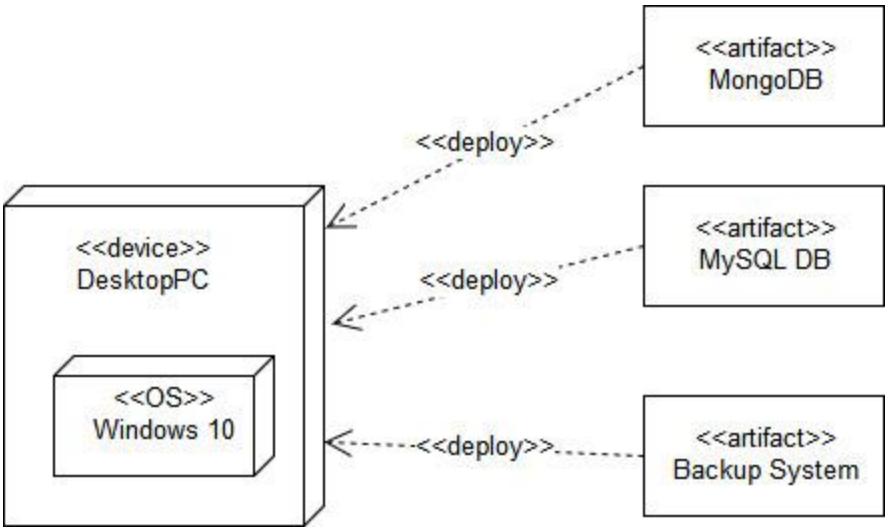


2.3 Component and Deployment Diagrams

Component Diagram



Deployment Diagram



III. Elaboration – Iteration 1.2

1. Design Model

1.1 Dynamic Behavior

[Create the interaction diagrams (1 sequence, 1 communication diagrams) for 2 relevant scenarios]

1.2 Class Design

[Create the UML class diagram; apply GoF patterns and motivate your choice]

2. Data Model

[Create the data model for the system.]

3. Unit Testing

[Present the used testing methods and the associated test case scenarios.]

IV. Elaboration – Iteration 2

1. Architectural Design Refinement

[Refine the architectural design: conceptual architecture, package design (consider package design principles), component and deployment diagrams. Motivate the changes that have been made.]

2. Design Model Refinement

[Refine the UML class diagram by applying class design principles and GRASP; motivate your choices. Deliver the updated class diagrams.]

V. Construction and Transition

1. System Testing

[Describe how you applied integration testing and present the associated test case scenarios.]

2. Future improvements

[Present future improvements for the system]

VI. Bibliography