<Airport Application>

Analysis and Design Document

Student:Baidoc Vlad

**Group:30641**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <03/04/2017>> | <x.x> | <details> | <Baidoc Vlad> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

I. Project Specification 4

II. Elaboration – Iteration 1.1 4

1. Domain Model 4

2. Architectural Design 4

2.1 Conceptual Architecture 4

2.2 Package Design 4

2.3 Component and Deployment Diagrams 4

III. Elaboration – Iteration 1.2 4

1. Design Model 4

1.1 Dynamic Behavior 4

1.2 Class Design 4

2. Data Model 4

3. Unit Testing 4

IV. Elaboration – Iteration 2 4

1. Architectural Design Refinement 4

2. Design Model Refinement 4

V. Construction and Transition 5

1. System Testing 5

2. Future improvements 5

VI. Bibliography 5

# Project Specification

Aplicatie mea se numeste “Airport Application” si are ca scop gestionarea si controlarea activitatii intr-un aerport.

Vor fi 2 tipuri de utilizatori care se pot loga fiecare pe contul personal .

Tipuri:

-user

-administator

Un user normal poate sa se logheze pe contul personal , daca nu are inca un cont il poate crea. Dupa logare poate sa vada zborurile de pe aeroport , destinatie , ora de placare/aterizare si poate sa isi reserve bilet la un anumit zbor.

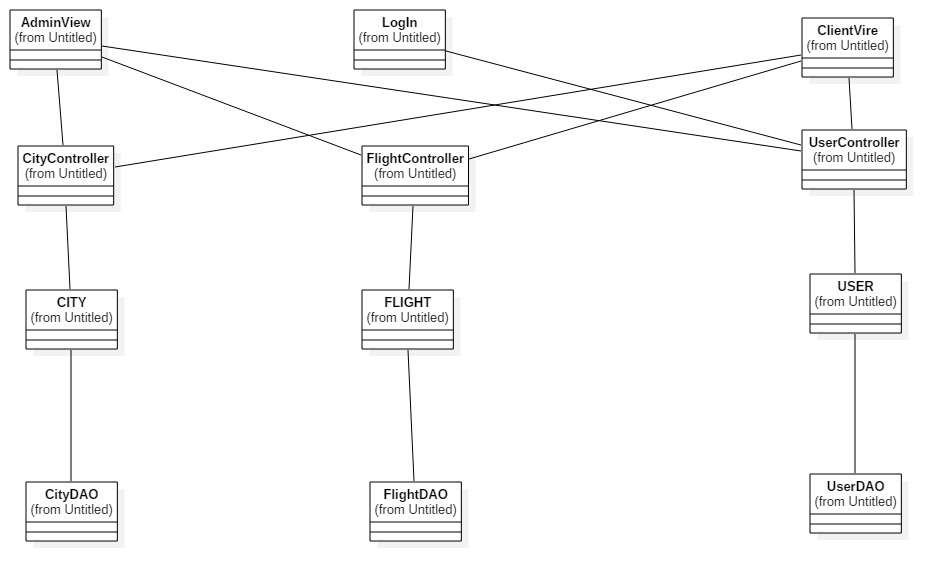
Un administrator are urmatoarele functii pe care le poate executa:

-CRUD pe user

-CURD pe zboruri

# Elaboration – Iteration 1.1

# Domain Model



# Architectural Design

## Conceptual Architecture

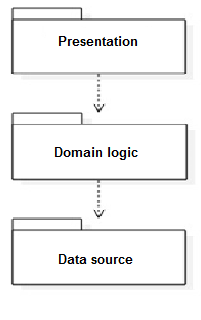
Will have 3 packages: Presentation logic, Domain logic, Data source logic

In Presentation logic will have the window application for every user and login window.

In Domain logic will have the implementation of the DB classes.

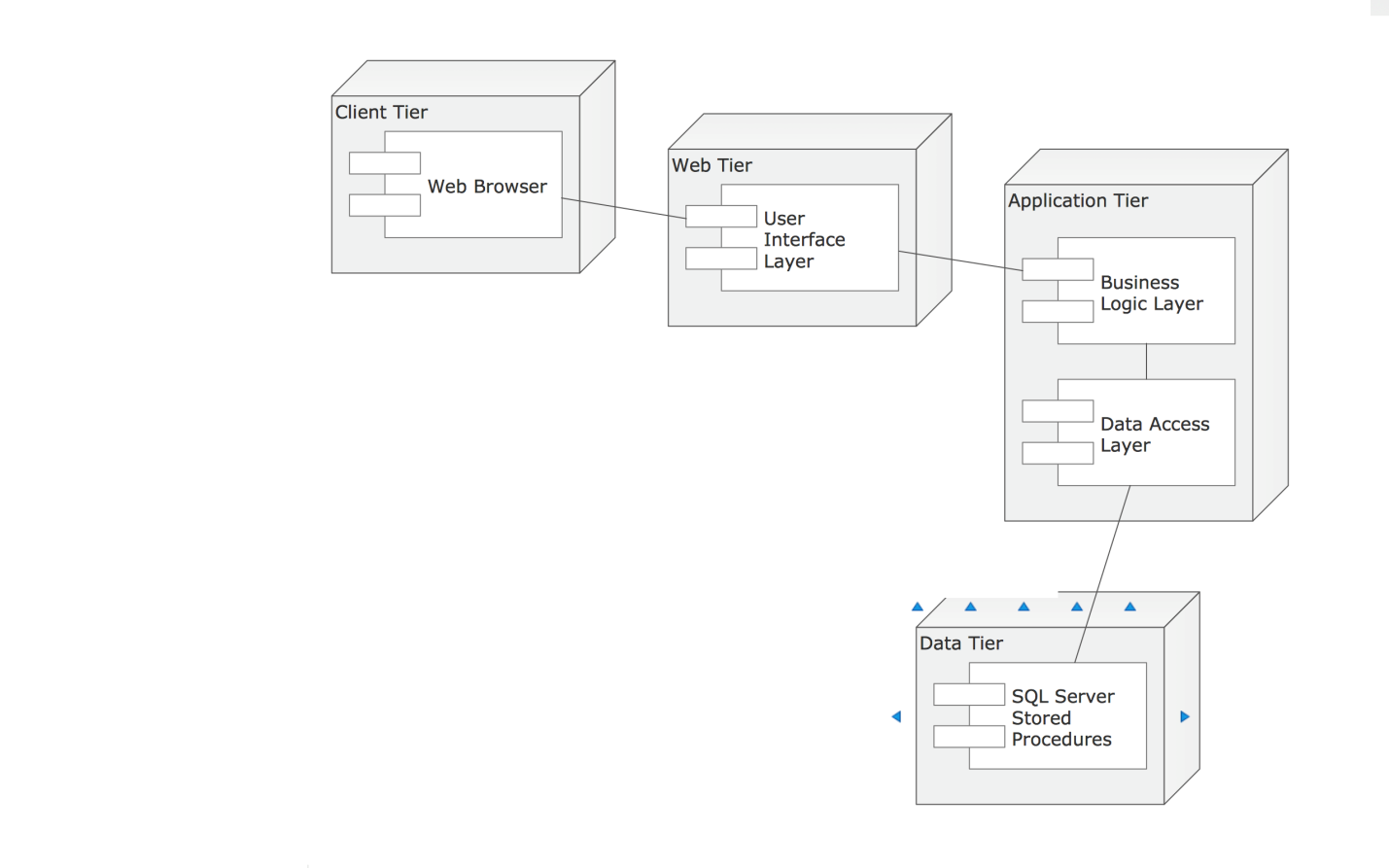
In Data source logic will have the DB connection and the Gateway for every class from Domain logic package.

## Package Design



## Component and Deployment Diagrams

# 



# Elaboration – Iteration 1.2

# Design Model

## Dynamic Behavior

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

## Class Design

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

# Data Model

*[Create the data model for the system.]*

# Unit Testing

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

# Elaboration – Iteration 2

# 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.]*

# Design Model Refinement

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

# Construction and Transition

# System Testing

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

# Future improvements

*[Present future improvements for the system]*

# Bibliography