Airline Application

Analysis and Design Document

Student: Prigoana Andrei

**Group: 30238**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 06/04/2017 | 1.0 |  | Prigoana Andrei |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

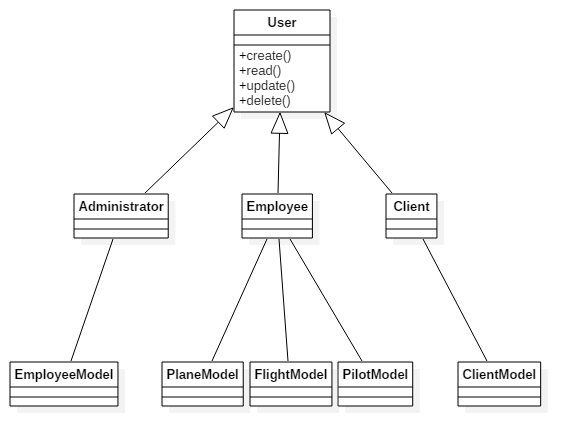
# Project Specification

The application will be written in Java Programming Language using Eclipse API. We will have three types of users, the administrator, the employee and the client.

* Administrator: or the supervisor of the employee it is responsible of giving access to the application for the employees, adding new employees, deleting them, update them (CRUD) or generate report to a specific employee.
* Employee: it is responsible with the information on the site, he can make CRUD on destinations list, on pilots list, on planes list and so on.
* Client: the client can simply make reservations or can directly buy the ticket online using it is credit card.

# Elaboration – Iteration 1.1

# Domain Model



# Architectural Design

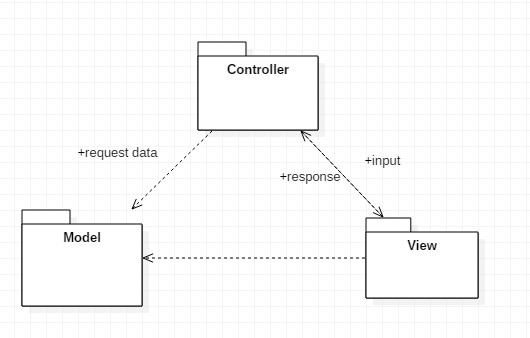
## Conceptual Architecture

Arhitectura folosita la aplicatie este MVC. Aceasta aplicatie este structurata in mai multe pachete: Model, View si Controller. Modelul si View-ul sunt conectate cu ajutorul Controller-ului, acestea neavand legaturi directe intre ele.

The application will be structured in four packages:

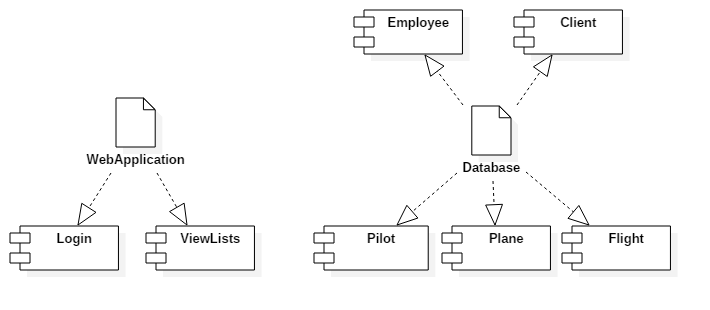
* Model: this package will create the objects;
* DataAcces: it is main purpose of this package is to communicate with the database;
* Controller: this package bounds the View and the dataAcces and Model package as well as the classes inside of them that implements the CRUD operations and not only;
* View: in this package we will have the user interface.

## Package Design

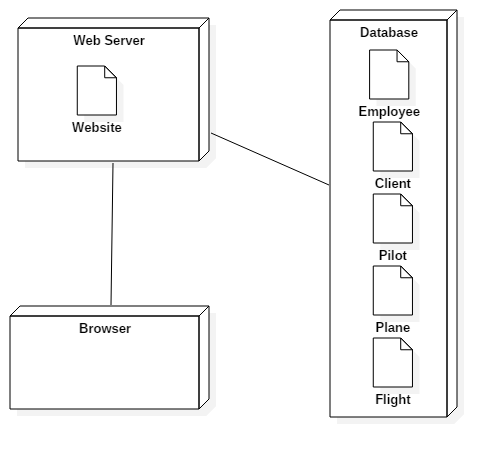


## Component and Deployment Diagrams

### Component Diagram



### Deployment Diagram



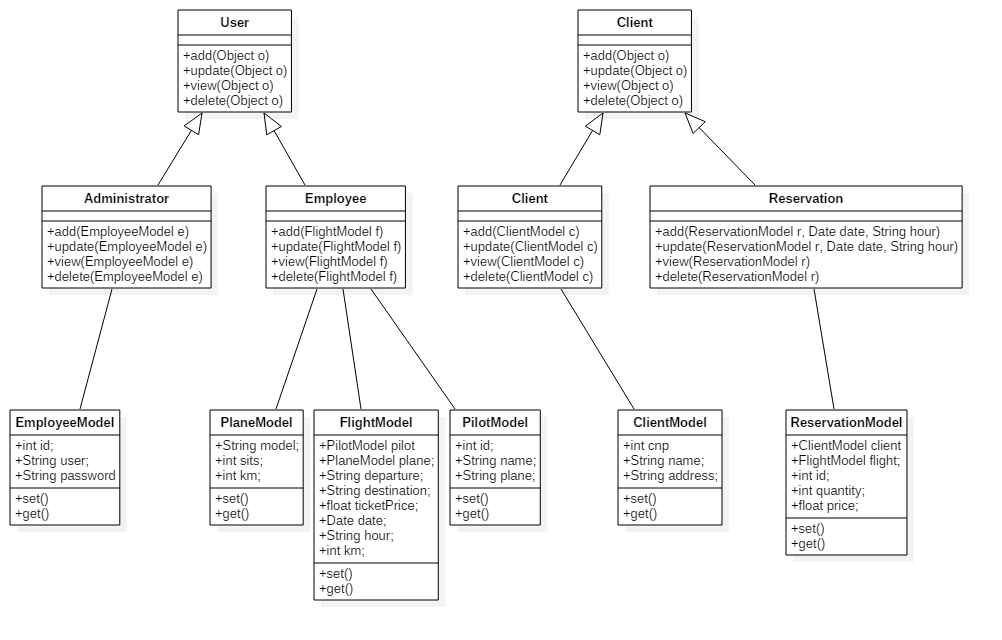
# Elaboration – Iteration 1.2

# Design Model

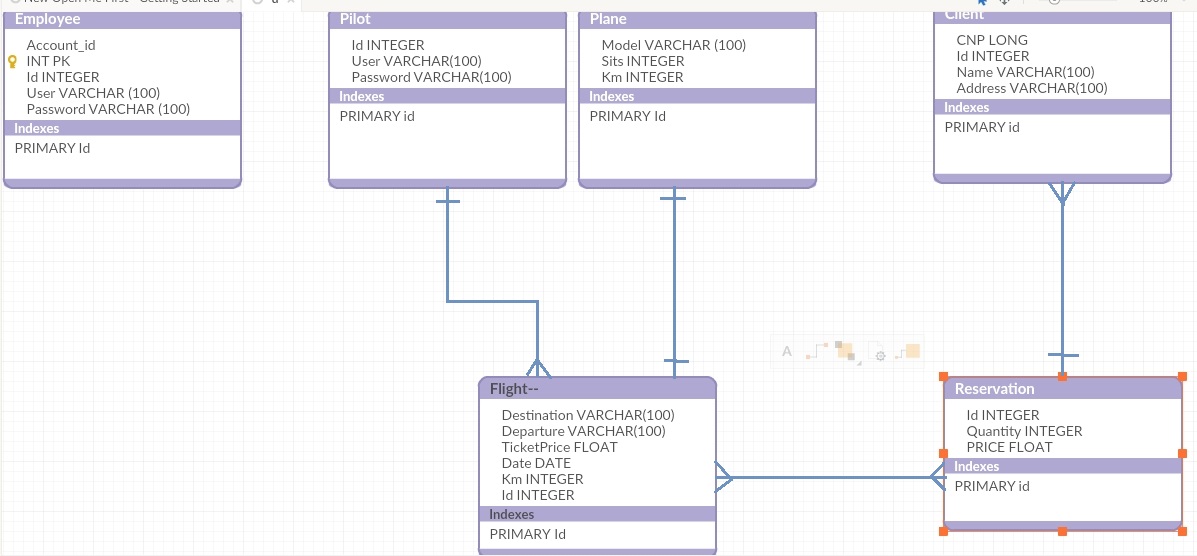
## Dynamic Behavior

## D:\An III, Sem 2\PS\Diagrams\SequenceDiagramCreateEmpl.pngD:\An III, Sem 2\PS\Diagrams\ObjectDiagram1.png

## Class Design



# Data Model



# 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

# Future improvements

The application can be improved by implementing the client server architecture or by adding the multi layer design pattern. In this way the application will be more advanced when we refer to performance.

# Bibliography

http://www.oracle.com/technetwork/articles/javase/index-142890.html