Analysis and Design Document

Student:

**Group:**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <dd/mmm/yy> | <x.x> | <details> | <name> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

*[Present the project specification]*

# Elaboration – Iteration 1.1

# Domain Model

*[Define the domain model and create the conceptual class diagrams]*

# Architectural Design

## Conceptual Architecture

*[Define the system’s conceptual architecture; use an architectural style and pattern - highlight its use and motivate your choice.]*

## Package Design

*[Create a package diagram]*

## Component and Deployment Diagrams

*[Create the 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

Grading/pass or not pass on Project and Iteration 2 will be done on following criteria:

* you need to refine Iteration 1 and Iteration 1.2
  + present what you have done in first iteration and what evolution was **0.5 p**
* you will have more users (admin, regular user) who login in your application (reflected also in Iteration 2) **0.5 p**
  + sequence diagrams (show who are the users) or somewhere else, I need to see it
* you will present minimum 3 design patterns for the final project (Creational, Structural, Behavioral) (reflected also in Iteration 2)
  + the design patterns will be shown also in the deliverable in the class diagrams (Iteration 1.2 and 2) **2p**
  + I want explanations here: Why these? Class diagrams? What could you use instead? **0.5p**
  + *GRASP and SOLID – I want to see that this is included in the documentation, how did you treat it? Have you thought about it?* ***0.5 p***
* you will present the architectural pattern that you used in your application (reflected also in Iteration 2))
  + I want explanations here: Why ? What could you use instead? **1p**
* *[Refine the architectural design:*
* *conceptual architecture****, 0.5 p***
* *package design (consider package design principles,* ***0.5p***
* *component and deployment diagrams. Motivate the changes that have been made.]* ***1p***
* as discussed the projects will be more complex than the Assignments (architecture, design, frameworks used) (reflected also in Iteration 2) **1p**
  + you will present the frameworks used: Hibernate, Spring or others ??? (reflected also in Iteration 2)
  + frameworks? What do you use? Why?
* report generation (motivate your choice)
* Integration Testing, Unit Testing? **1p**
* Future improvements
* projects with only 2 or 3 classes in the class diagram will not be accepted! Will be more complex than that! (reflected also in Iteration 2) -> FAIL
* project with only 2 or 3 tables in the database model diagram will not be accepted! Will be more complex than that! (reflected also in Iteration 2) (present relationship types in your database model: OneToOne, OneToMany, ManyToMany and motivate your choice) -> FAIL
* 1p oficiu