Car Selling Application

Analysis and Design Document

Student: Abrudan Darius

**Group: 30233**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 05/04/17 | 1.0 |  | Abrudan Darius |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

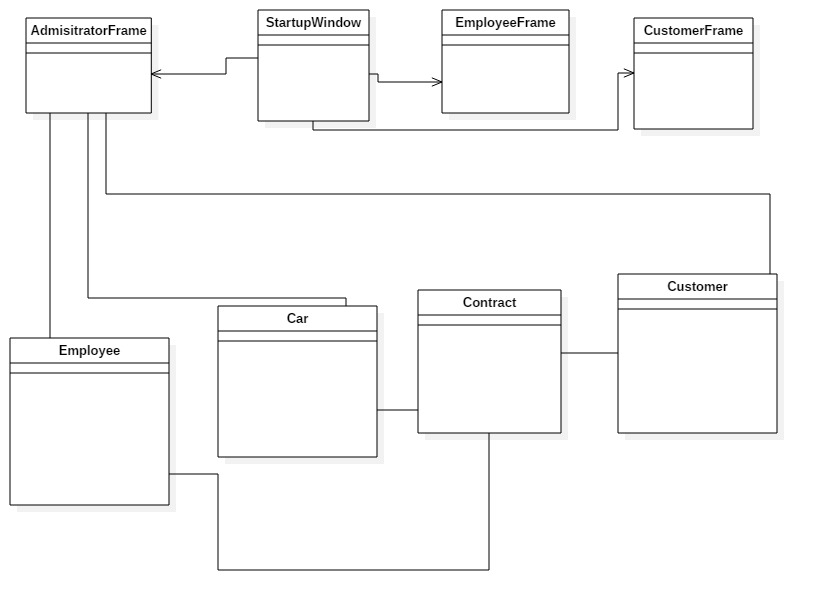
# Project Specification

Design and implement a client-server application for selling cars. The application will be used by customers, selling company’s employees and an administrator. A customer can search and view the available cars for a specific mark and also has the possibility of configure some additional parts for a car. In order to buy a car, a customer must fill a contract with its personal data, contract created by the employee. The administrator can make CRUD operations (create/update/delete) on cars, employees and customers.

# Elaboration – Iteration 1.1

# Domain Model

The domain model will look like the preliminary class diagram shown below:

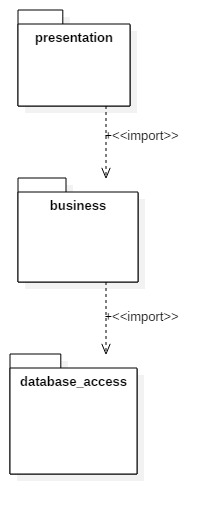


# Architectural Design

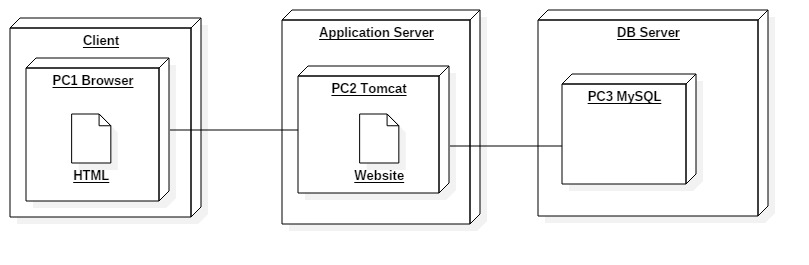
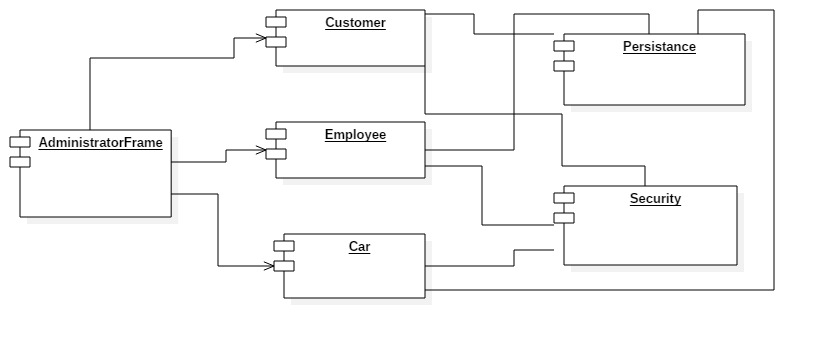
## Conceptual Architecture

As an architectural pattern I choose to implement the client-server pattern for my car selling application and also the layered architectural pattern. I will have 3 major layers: presentation layer which will contain the GUI for the log in and the possibility to choose what kind of operation you want to do, depends on what type of user you logged in, business layer representing the operations and functionalities that the application provides and a database access layer for accessing and modifying the DB.

## 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