Pet Hotel

Analysis and Design Document

Student: Campean Casiana Stefana

**Group: 30238**

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

Our animals are like family to us, but unfortunately we can’t bring them everywhere we go and we can’t let them home alone for a long time when we go somewhere far. So this project aims to solve this problem by implementing an hotel application for pets, where you can leave your animal there while you are away from home for a longer period of time.

This application will be used by every pet owner who wants to bring their animal at the hotel, and will also be used by the hotel manager and staff.

The client can choose from different services based on the type of their animal and they will get a report of the activities of their animal when they take their pet home.

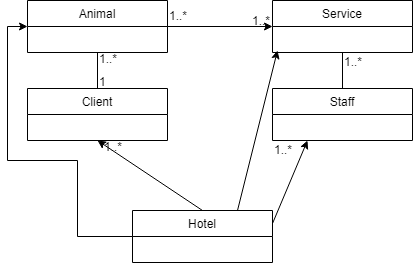
The hotel staff will consist of different employees, like a vet, pet groomer, pet trainer, and other types of employees that are in charge of taking care of the animals, like playing with them, giving them food and putting them to sleep.

The manager user will be in charge of handling with staff, clients and animals, validate registrations and giving reports to the clients about theirs animal activity.

# Elaboration – Iteration 1.1

# Domain Model

The domain for this project is the hotel. From the project specification we can determine some important classes, like Client, Animal, Service, staff and Hotel, and we can make the conceptual class diagram.



# Architectural Design

## Conceptual Architecture

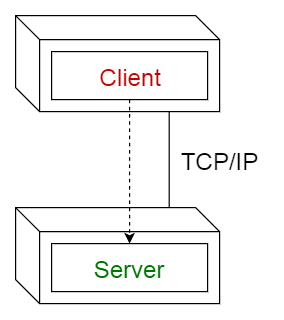
**Client Server Architecture**

Client/server architecture is a computing model in which the server hosts, delivers and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or internet connection. This system shares computing resources.

Client/server architecture is also known as a networking computing model or client/server network because all the requests and services are delivered over a network.

This pattern consists of two parties; a server and multiple clients. The server component will provide services to multiple client components. Clients request services from the server and the server provides relevant services to those clients. Furthermore, the server continues to listen to client requests.

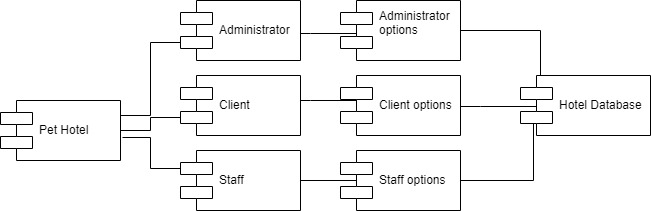
I choose this architectural pattern because it was the best fit for this application, because it is a web application and I have notifications and it is easier to do this in a client server application.

****

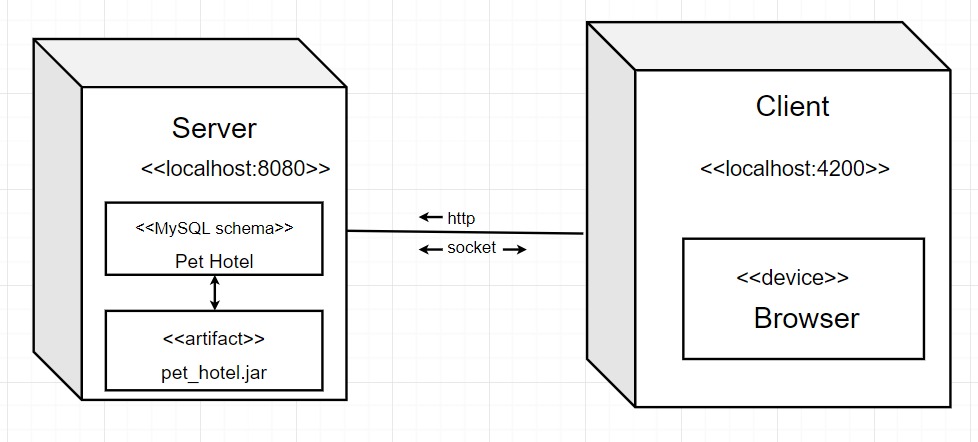
## Package Design

## Component and Deployment Diagrams

***Component Diagram***

****

***Deployment Diagram***

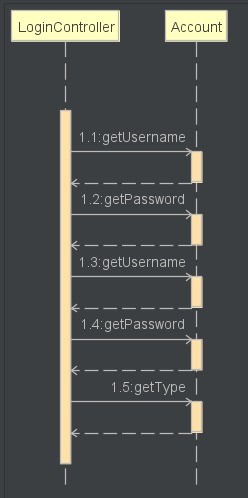


# Elaboration – Iteration 1.2

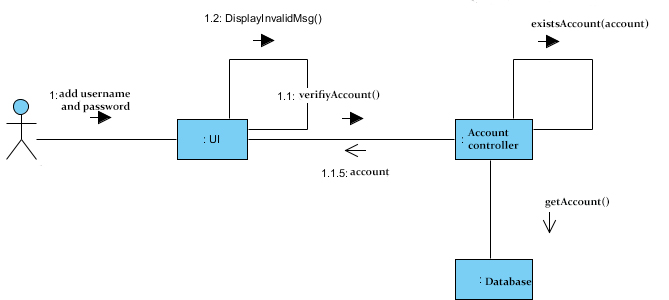
# Design Model

## Dynamic Behavior

**Sequence diagram**



**Communication diagram**



## Class Design

**UML Diagram**

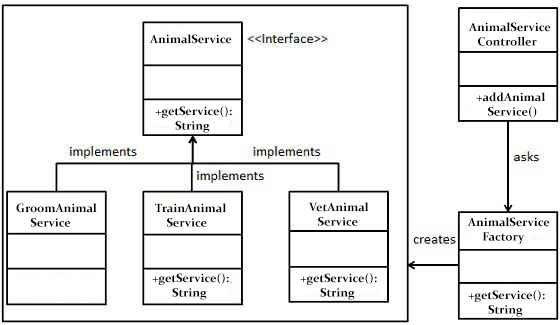
**GOF Patterns**

**Factory Design Pattern**

The design pattern required for this project is Factory Design Pattern, a creation pattern.

Factory pattern is one of the most used design patterns in Java. This type of design pattern comes under creational pattern as this pattern provides one of the best ways to create an object.

In Factory pattern, we create object without exposing the creation logic to the client and refer to newly created object using a common interface.



# Data Model

The data model is represented as Java classes and as database tables.

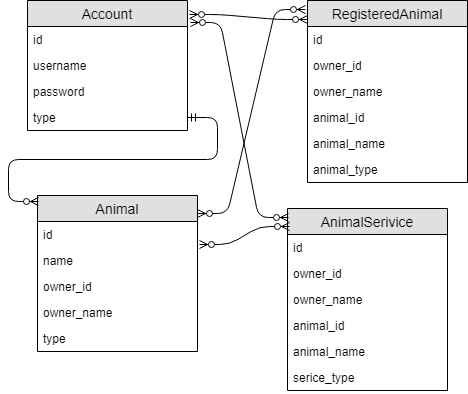
In this project, the data model is represented by these classes in Java/tables in DB:

-Account

-Animal

-RegisteredAnimal

-AnimalService



# Unit Testing

In order to test the project, I implemented Junit test for the class AccountService, testing getAccounts, deleteAccount and existsAccount.



C:\Users\User\AppData\Local\Microsoft\Windows\INetCache\Content.Word\exists.png

# 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

# Future improvements

I believe that my application can have a lot of improvements. For example, this hotel may contain more services, staff can view animals by type, and there should be more notifications for the client.

# Bibliography

[1] Hibernate Tutorial: <https://www.youtube.com/playlist?list=PLEAQNNR8IlB7fNkRsUgzrR346i-UqE5CG>

[2]: Angular tutorial:

<https://angular.io/tutorial>

[3] Building a Web Application with Spring Boot and Angular

<https://www.baeldung.com/spring-boot-angular-web>

[4] WebSocket with Spring boot and Angular

<https://www.stackextend.com/angular/websocket-with-spring-boot-and-angular/>

[5] Angular 7 + Spring Boot Login Authentication

<https://www.javainuse.com/spring/ang7-login>

<https://www.javainuse.com/spring/ang7-basic>