**Movie World System**

**Taha Mohamed Alzein (269055)**

**Muhammad Nadeem (266704)**

**Fadi Atia Dasus (266265)**

**Oskars Arajs (266534)**

**Balkis Ibrahim (260092)**

**Supervisors**

**Ole Ilsgaard Hougaard**

**Christian Flinker Sandbeck**

**ICT Engineering, VIA University College, Horsens**

**3rd Semester**

*Abstract*

The purpose of this project

**Table of content**

[1.Introduction 4](#_Toc532567444)

[2 User Stories and Requirements 5](#_Toc532567445)

[2.1 User Stories 5](#_Toc532567446)

[2.2 Functional Requirements 5](#_Toc532567447)

[2.2 Non-Functional Requirements 5](#_Toc532567448)

[3 Analysis 6](#_Toc532567449)

[3.5 Conceptual diagram 6](#_Toc532567450)

[4 Design 7](#_Toc532567451)

[4.2 Class Diagram 7](#_Toc532567452)

[5 Implementation 8](#_Toc532567453)

[6 Testing 8](#_Toc532567454)

[7 Conclusion 8](#_Toc532567455)

[8 References 8](#_Toc532567456)

**List of Figures**

# 1.Introduction

# 2 User Stories and Requirements

The purpose of this chapter is to draw the user stories, that can help to create a simplified requirement. The requirements are one of the major tasks in designing systems development (Alsaleh.S and Haroon.H., 2016). Requirements are very critical phase for functioning system. So first we start writing the User stories then requirements. Requirements can be functional and non-functional as well.

## 2.1 User Stories

As the name suggests. it describes how the User require the software that is particularly helpful for customer. Admin should be the one who need the functioning software and customers can visit the software.

1. As an Admin I want to be able to add movie to the system, so that the Customer can see it.
2. As an Admin I want to be able to remove movie from the system, so that I can delete old movies.
3. As a Customer I want to be able to select a movie, so that I can check all the details about it.
4. As a Customer I want to be able to book a ticket for a certain movie, so that I can reserve a place for that movie.
5. As a Customer I want to be able to stream movie, so that I can watch movie at home.
6. As a Customer I want to be able to pay using electronic payment method for my reservation or streaming, so that I can have a receipt.

## 2.2 Functional Requirements

Based on the information from user stories the following functional requirements have been created, and non-functional requirements are provided by university,

1. The system must allow the Admin to be able to add movie.
2. The system must allow the Admin to be able to remove movie.
3. The system must allow the Customer to be able to see a list of movies.
4. The system must allow the Customer to be able to get movie information.
5. The system must allow the Customer to be able to book a cinema ticket for the selected movie.
6. The system must allow the Customer to be able to purchase a ticket electronically for cinema.
7. The system must allow the Customer to be able to purchase a subscription electronically for streaming.
8. The system must allow the Customer to be able to get a receipt for his payment. through email.
9. The system must allow the Customer to be able to stream movie.

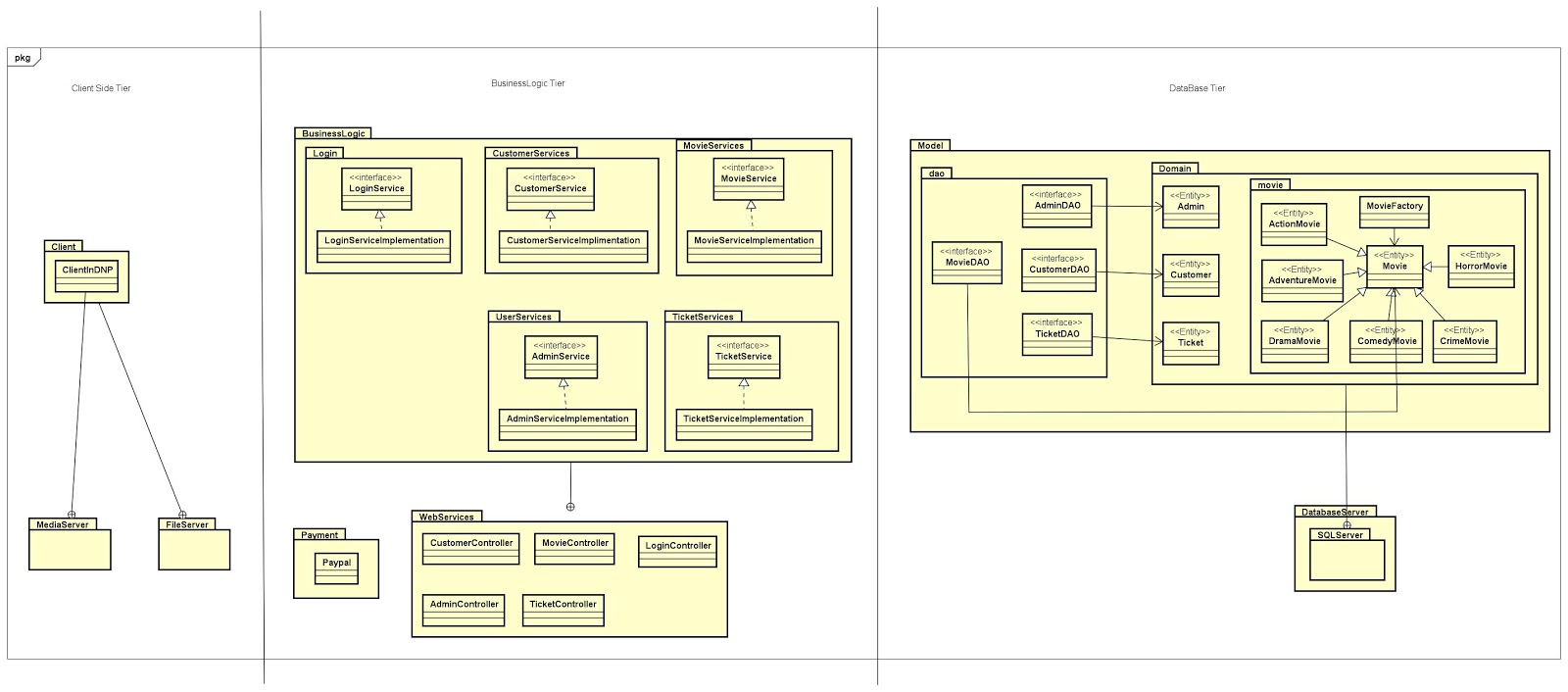
## 2.2 Non-Functional Requirements

1. The system must be implemented as a heterogeneous system (Server in Java, Client in .NET).
2. The system must be 3-tier architecture.
3. The system must use web services
4. The system must use a protocol for sockets and RMI
5. The system must include GUI for two clients (Customer and Admin)

# 3 Analysis

## 3.5 Conceptual diagram

Conceptual Diagram demonstrates how all the classes in their packages interact with each other. As we have a three-tier architecture system so conceptual diagram is also representing how tiers are interacting. The first tier is represented by database tier which has data access objects to access the database. Model package has four database access objects i.e. Admin, Customer, Movie and Ticket DAO. Second Tier is Business Logic Tier, where all business logic operations are implemented. Business rules will determine how the data should be stored, changed and create. Web Services are also implemented in Business Logic. Third tier should be representation of data that is our client-side tier. In that tier, how movie system is represented graphically. Three tier architecture system will be discussed in detail in class diagram.



***Figure 3.4 Conceptual diagram***

## 

# 4 Design

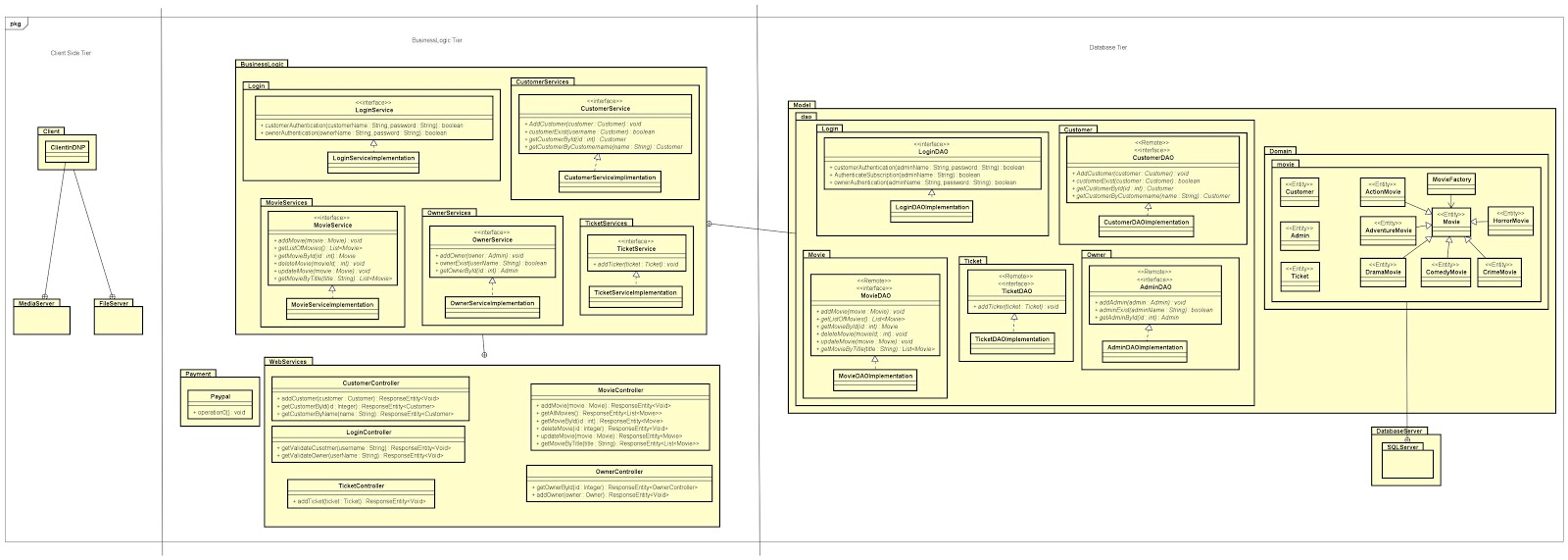
## 4.2 Class Diagram

Class diagram demonstrate, which operations needed for functioning software. In three tier architecture system first tier should be database tier, second should be business logic tier and third tier is a graphical representation (User Interface) tier.

**Database Tier:** Database tier consist of Database, Data transfer object and Data access object.  Database is Microsoft Server SQL for storing values. DTO are simple object that does not contain business logic but only the data. The behaviour of DTO is storing, serialization and deserialization of Data. So Movie, Admin, Customer and Ticket objects are DTO, that carries only the field, getter and setter. Data operation (CRUD) is provided by DAO without exposing data. DTA and DTO is created on JAVA. Hibernate is used for persistence (JPA).

**Business Logic Tier:** is a real-world business rules which exchange the information between user interface and database. For example, get movie by title is a business logic. How this logic will operate, as movies are stored in database so system call the DAO and present it on User Interface. There are communication mediums used between database and user interface. First is Remote invocation method that is a medium of communication between DAO and business logic. Second Web Services (HTTP) are used for communication between business logic and User Interface. Business Logic and restful web services are implemented by JAVA.

**Client-Side Tier** (User Interface Tier): This tier is a representational tier of system, how system look. As movie system has two client admin and customer. Admin GUI main responsibility is to add and remove movie, and Customer can watch movie at cinema or also stream movie at home. Client-Side tier is implemented on .NET.



***Figure 4.2 Class diagram***

# 5 Implementation

# 6 Testing

# 7 Conclusion

# 8 References

Guidelines Development and Summary, IT-SDJ2X-S18 Sesion Material,[Last accessed 05/04/2018] via link: <https://studienet.via.dk/Class/IT-SDJ2X-S18/Session%20Material/SDJ2-S18-19GuidelinesDeploymentAndSummary.pdf>

Ken Schwaber, Jeff Sutherland, 2011; The Scrum Guide, [Last accessed 10/04/2018 ] via link: <https://studienet.via.dk/Class/IT-SWE1X-S18/Session%20Material/Scrum_Guide.pdf>

Project report, 2017 (Appendix 3) VIA Engineering Guidelines [Last accessed 10/04/2018] via link: <https://studienet.via.dk/projects/Engineering__project_methodology/General/Guidelines/2017%20Project%20Report%20(Appendix%203)%20-%20VIA%20Engineering%20Guidelines.pdf>

Project description, 2017 (Appendix 1) VIA Engineering Guidelines [Last accessed 27/02/2018] via link:<https://studienet.via.dk/projects/Engineering__project_methodology/General/Guidelines/2017%20Project%20Description%20(Appendix%201)%20-%20VIA%20Engineering%20Guidelines.pdf>

Process report,2017 (Appendix 2) VIA Engineering Guidelines, [Last accessed 05/04/2018] via link:<https://studienet.via.dk/projects/Engineering__project_methodology/General/Guidelines/2017%20Process%20Report%20(Appendix%202)%20-%20VIA%20Engineering%20Guidelines.pdf>

Rohit Joshi, 2015; Java Design Patterns, [Last accessed 14/03/2018] via link; <http://enos.itcollege.ee/~jpoial/java/naited/Java-Design-Patterns.pdf>

Thomas Connolly, Carolyn Begg, 2015; Database Systems A Practical Approach to Design,Implementation and Management, [Last accessed 18/05/2018] via link;

<http://people.stfx.ca/x2011/x2011asx/5th%20Year/Database/Database%20Management%20Textbook.pdf>

9 Appendices

9.1 Appendix 1 Project Description

9.2 Appendix 2 Use case description

9.3 Appendix 3 USER GUIDE

9.4 Appendix 4 Diagrams

9.5 Appendix 5 Scenarios

9.6 Appendix 6 Testing