**Chapter One**

1. **Introduction**

KIOT Online ticket reservation system is basically made from providing the customers, anywhere anytime services for booking set in movie hall and together. Information about the movies online. The user can easily be able to know about the movie released and then make choice.

It is a web-based system. The system is to provide an alternate and convenient way of for customer to buy movie tickets. It is an automatic system. Staff can use the system to insert and delete data (example film description, time table) which will update the webpage. Also, staff can check the statistic information from the system. The staff doesn’t need to do anything with the order once it is received through the system.

* 1. **Background**

Describe the project area you are working on online cinema reservation system (online movie ticket reservation system) for KIoT.

* It is believed that the application of computer technology in any activity would go a long way in making that activity easier.
* To study online movie ticket reservation system helps the system faster, and more convenient and you do not have to go to station twice.
* To save our time and money.
  1. **statement of the problem**

**The major problem of KIoT online cinema reservation system**

* Late seats reservation by customers.
* You must go station twice.
* Customer must come physically to buy ticket. This is boring, cost and time-consuming process.
* The existing system only provide text-based interfaces. Since the system is implemented in manual, so the response is very low.
* It wastes a lot of time and money for the customer and company.
  1. **Objectives of the system** 
     1. **General objective**
* The main aim of this study is to design and implement an online movies ticket reservation system.
  + 1. **Specific Objectives**

To achieve the specified general objective, the following specific objectives are stated:

* Determining how to build the proposed system and show the solution domain of this system.
* Determining what to build and understand the problem domain of the system that guide for transforming user requirement into system requirements.
* Understand Manual process and the efficiency of the existing system
* Identify functional and nonfunctional requirements.
* Implementation of the project using programming language according to the design specification.
* Finally implement and test the new system.
  1. **Scope and Limitation**
     1. **The scope of the project**

The scope of this project is to develop web-based cinema reservation system for KIoT.

* Enable customers view their cinema information easily and quickly.
* Generate report.
* Manage cinema related information.
* Assignment of technicians.
  + 1. **The limitation of the project**
* It is restricted only for English language.
* No ticket printing system.
* No ticket cancelation possible.
* The cinema hall only reserve for Cinema and DSTV purposes.
* The system depends on electric power and network connection.
  1. **Methodology**
     1. **Requirement Gathering Methods**
* **Interview**: - it is the primary technique used to elicit the necessary information from the manager of KIoT cinema who takes the responsibility to manage KIoT cinema and the ticket seller. The ticket seller gives us valuable information about the overall activity they perform, concerning on how they sell the ticket and the show times of the movies.
* **Observation**: - a site visiting was mad to support the interview done with aim to understand ticket reservation system. The team members have observed physically by going to the place. Also, the team have seen that there was no any well-developed computerized and online system in the KIoT cinema and also information about the cinema and the service that the KIoT cinema provides were not available easily. The team also observes that ticket reservation is only by face to face.
  + 1. **Tools used in the project**

In the analysis, design and implementation of the project, the team uses the following front end and back end software which are used for developing the documentation.

* Software tool
* Microsoft word and Microsoft power point.
* For document preparation and
* For Presentation
* Enterprise architect
* For Reversing UML Diagrams in chapter five
* For Preparation of Class Diagram in chapter four
* For Preparation of Sequence Diagram In chapter three
* For Preparation of Activity Diagram In chapter three
* NetBeans IDE 8.2
* For Generating Source code in chapter Five
* Hardware tools
* PC (personal computer)
* Flash disk
* Printer (only for printing)
* Network cables
* A4 size paper
  1. **Significance of the project**

The significance of this project is primarily to improve the manual system into computerized system. The main purpose of online ticket booking system is to provide another way for the customer to buy cinema ticket.

* The newly designed system is faster, more convenient and you don’t have to go station twice.
* Reduce the loss of documents & human resource.
* System gives fast service to the customer.
* Online ticket reservation system is always a safe option unless the website insecure.
* It saves a lot of time and money for the customer and company.
* The customer can buy the ticket anywhere and anytime.
* The company reduces money for paying employs.
* Administer they can manage the cinema hall easily and safely.
* Staff reduce the work load of staff.
  1. **Beneficiary of the project**

Our project benefits both for customer and for company.

* KIOT Student and
* KIOT Cinema Campus.
  1. **Organization of the project**

The researcher of the project has been arranged in order, chapter one contains the introduction, theoretical background, significance of the study, organization of the research, the second chapter shows the detail analysis of the existing and the new system. This chapter deeply deals with discussions of the new proposed system using a UML (Unified Modeling Language) diagrams. A detailed description of system functionalities and use case documentation, Sequence diagram for the interaction between the actor and system, activity diagrams, and class diagram are also part of this new proposed system. The Fivith chapter shows the implementation of the project

**Chapter Two**

**Requirement Analysis**

* 1. **Functional Requirement**

A functional requirement specifies what the system should do to perform some tasks. Functional requirement describes the functionality of service provided by the new system.

The following are some functionalities of the system.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  | | --- | | **R.ID** | | Priority | | Detail Description |
| Req 1 | moderate | | The system shall give application form for registration. |
| Req 2 | High | | |  |  | | --- | --- | |  | The system shall show information. | |
| Req 3 | High | | The system shall require login before allowing any functions for any Customers. |
| Req 4 | moderate | | The system shall give application form for reserving seats. |
| Req 5 | High | | The system shall allow to enter the account number. |
| Req 6 | High | | The system shall require login before allowing any functions for the staff. |
| Req 7 | High | | The system shall allow the staff to add movie schedule. |
| Req 8 | moderate | | The system shall allow the staffs to see seat information. |
| Req 9 | Low | | The system shall allow the staffs to see seat information. |
| Req 10 | moderate | | The system shall allow the staff to update movies. |
| Req 11 | High | | The system shall allow the staff to add movie adverts. |
| Req 12 | High | | The system shall allow the admin to login. |
| Req 13 | High | | The system shall allow the admin to add staff. |
| Req 14 | Low | | The system shall allow the admin to delete staff. |
| Req 15 | high | The system shall allow the admin to view report. | |

**2.2 Non-Functional requirement**

|  |  |  |
| --- | --- | --- |
| **Req ID** | **Requirement** | Category |
| **R1** | All pages should be supported with English languages. | Usability |
| **R2** | The system shall have simple user interface. | Usability |
| **R3** | The system shall support one user per minute. | Performance |
| **R4** | The system shall run on Linux/windows. | Portability |
| **R5** | The system shall not fail at all. | Reliability |
| **R6** | The system shall contain strong password to hack. | Security |

**2.3 Actors of the system**

Actor classes are used to model and represent roles for "users" of a system, including human users and other systems. Based on this we identify the following user of our system:

* **Customer:** - are peoples who buys a ticket and want to see movies, register, login, and search movie, reserve ticket, and view seat information
* **Staffs:** - login, search movie, manage movie, manage movie schedule, advert movie, and view seat information.
* **Administrator:** -A cinema administrator will be a team leader, will include: - login, view report, manage staff, manage account, delete, update, create accounts and change password.

**2.4 System use case diagram**

Use case diagram is a diagram that shows use case, actors and their relationships. Use case represents the interaction between the user and the system. Use case diagram depicts a collocation of use cases, actors, there associations, a system boundary box (optional), and packages (optional).

**A use case** describes a sequence of action that provides a measurable value to an actor and draw as a horizontal ellipse.

**An actor:** is a person, organization, or external system that plays a role in one or more interactions with the system and draw as stickman figure.

**System boundary:** indicates the scope of the system project. Anything within the box represent functionalities in side in scope.

Relationship between actors and use cases exists whenever an actor is involved with an interaction described by a use case and modeled as a line connecting use cases and actors.

**Use Case Diagram**



**Fig.01 Use Case Diagram (uc01)**

**2.5 Use case documentation**

These is step by step description of the action performed by each use case. It should contain

preconditions, post condition, main course of action, and alternate course of action as it is shown in the following table:-

|  |  |  |
| --- | --- | --- |
| Section | Purpose | |
| Actor | Customer | |
| Use case No: | Uc01 | |
| Use Case Name | **Register** | |
| Description | The system allows the customer registered. | |
| Precondition | The customer wishes to register. | |
| Post conditions | The user will use the website. | |
| Basic course of action | User Action | System Response |
| 1. Click registration button.  3. The user enter required information and Click register button. | 2. The system display registration form.  4. Validates the user information.  5. The system informs the user that they successfully registered. |
| Alternate Course of Action | If the user enters invalid password and username.  5.1 The system display try again error message.  5.2 go back to step3. | |

Table UC01 Register

|  |  |  |
| --- | --- | --- |
| Section | Purpose | |
| Actor | Customer, Staff, Admin | |
| Use case No: | Uc02 | |
| Use Case Name | **Login** | |
| Description | The system allows the staffs, customers and the admin to login to the system. | |
| Precondition | The staffs, customer and admin must have user name and password to login. | |
| Post conditions | The staffs, Admin and Customers use their own pages. | |
| Basic course of action | User Action | System Response |
| 1. The staffs, customer and admin click login button.  3.The staffs, customer, and admin enter his/her username and password.  4. Then click login button. | 2. The system display login form.  4. Validates the user information.  5.The system validates the entered information and display main page. |
| Alternate Course of Action | If the staff, Admin and/or Customer enters invalid user name or password.  5.1 The system displays try again error message.  5.2 the staff, Admin and/or Customer reenter the correct information. | |

Table UC02 Login

|  |  |  |
| --- | --- | --- |
| Section | Purpose | |
| Actor | Staff | |
| Use case No: | Uc03 | |
| Use Case Name | **Advert movie** | |
| Description | The system allows the staff to advert movie when they watch. | |
| Precondition | The staff must be authorized to advert movie news. | |
| Post conditions | The movie to be adverted will be added. | |
| Basic course of action | User Action | System Response |
| 1. Click advert movie link  3.The staff click “advert movie” button.  5. the staff insert movie and information and then click “add” Button. | 2. the system display advert movie page  4. The system display advert movie form.  6. The system will check the validity and display successfully message. |
| Alternate Course of Action | If the Staff enters invalid information  5.1 the system displays error message  5.2 go back to step 5 | |

Table UC03 Advert Movie

|  |  |  |
| --- | --- | --- |
| Section | Purpose | |
| Actor | Customer, Staff, Admin | |
| Use case No: | Uc02 | |
| Use Case Name | **Login** | |
| Description | The system allows the staffs, customers and the admin to login to the system. | |
| Precondition | The staffs, customer and admin must have user name and password to login. | |
| Post conditions | The staffs, Admin and Customers use their own pages. | |
| Basic course of action | User Action | System Response |
| 1. The staffs, customer and admin click login button.  3.The staffs, customer, and admin enter his/her username and password.  4. Then click login button. | 2. The system display login form.  4. Validates the user information.  5.The system validates the entered information and display main page. |
| Alternate Course of Action | If the staff, Admin and/or Customer enters invalid user name or password.  5.1 The system displays try again error message.  5.2 the staff, Admin and/or Customer reenter the correct information. | |

Table UC03 Advert movie

|  |  |  |
| --- | --- | --- |
| Description | The system allows the staffs, customers and the admin to login to the system. | |
| Precondition | The staffs, customer and admin must have user name and password to login. | |
| Post conditions | The staffs, Admin and Customers use their own pages. | |
| Basic course of action | User Action | System Response |
| 1. The staffs, customer and admin click login button.  3.The staffs, customer, and admin enter his/her username and password.  4. Then click login button. | 2. The system display login form.  4. Validates the user information.  5.The system validates the entered information and display main page. |
| Alternate Course of Action | If the staff, Admin and/or Customer enters invalid user name or password.  5.1 The system displays try again error message.  5.2 the staff, Admin and/or Customer reenter the correct information. | |

Table UC03 Advert movie

**Chapter Three**

**High Level Design**

**3.1 Sequence Diagram**

Sequence diagram will be prepared for each use case to show how different objects interact with each other to achieve the functionality of the use case. A sequence diagram models how the classes of the objects interact with each other over time as the system runs.

**Registration**



**Fig.02 Register Sequence Diagram (uc01)**

**Login**



**Fig.03 Login Sequence Diagram (uc02)**

**Reserve Ticket**



**Fig.04 Reserve Ticket Sequence Diagram (uc03)**

**3.2 Activity Diagram**

An Activity diagram is essentially a flowchart, showing flow of control from activity to activity it involves

* Modeling the sequential (and possibly concurrent) steps in a computational process.
* Modeling the flow of an object as it moves from state to state at different point in the flow of control.

****

**Fig.05 Register Activity Diagram (uc01)**

****

**Fig.06 Reserve Ticket Activity Diagram (uc02)**

**Chapter Four**

**Low Level Design**

**4.1 Class modeling diagram**

Class diagram provide an over view of target system by describing the object and classes inside the system and the relationship between them. It provides a wide variety of usage; from modeling the domain specific data structure to detailed design of the target system. With the share model facility, you can re-use the class model in the interaction diagram for modeling the detailed design of the dynamic behavior. This diagram can be derived from one part of modeling is class responsibility collaboration (CRC).



**Chapter Five**

**Implementation**

**5.1 implementation**

Implementation is the phase where objectives of physical operations of the system turned into reality therefore real working model. The crucial phase in the system development life cycle is the successful implementation of the new system design. The process of converting as new system into an operational one is known as system implementation. This includes all those activities that takes place to convert from an old system to a new system. To implement the project we use Java language.

**5.2 Coding**

The first phase of implementation is coding. Coding is the process whereby the design specification created by the designers is tuned to working computer code by the programmer. The code is made simple in such a way that another programmer can easily understand and work on that in future.

* Use Java Programming language for the implementation of the system.

**1, Sample code for Person**

public class Person {

private String E\_mail;

public String F\_Name;

private int ID;

public String L\_Name;

private int Phone\_number;

public char Sex;

public Person(){

}

public void finalize() throws Throwable {

}

public String getE\_mail(){

return "";

}

public int getID(){

return 0;

}

public int getPhone\_number(){

return 0;

}

public void setE\_mail(String mail){

}

public void setID(int id){

}

public void setPhone\_number(float phone){

}

}

**2, Sample code for Customer**

public class Customer extends Person {

public Customer(){

}

public void finalize() throws Throwable {

super.finalize();

}

public String getE\_mail(){

return "";

}

public int getID(){

return 0;

}

public int getPhone\_number(){

return 0;

}

public void Login(){

}

public void Register(){

}

public void Resrev\_Tikect(){

}

public void Search\_Movie(){

}

**public void setE\_mail(String mail){**

**}**

**public void setID(int id){**

**}**

**public void setPhone\_number(float phone){**

**}**

**public void View\_Seat\_Infromatcion(){**

**}**

**}**

**5.3 Testing**

Testing is a final phase of implementation. Testing is a process to show the correctness of the program. Testing is checking of the system workability in an attempt to discover errors and avoiding such errors from the system. Some examples of testing.

**5.3.1 Unit Testing**

Each module is tested individually in an attempt to discover any errors in its code. In unit testing each module (roughly a section of code that performs a single function) is tested individually to discover any errors that may exist in the modules code.

1, **Sample Test for Payment**

package customer\_tool;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

public class Customer\_ToolTest {

Customer\_Tool b;

public Customer\_ToolTest() {

}

@Before

public void setUp() {

b=new Customer\_Tool();

}

@After

public void tearDown() {

b=null;

}

@Test

public void testMain() {

String c=b.Payment();

assertEquals("Payment",c);

}}

**2, Sample Testing for View Report**

package admin;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

public class AdminTest {

Admin report;

public AdminTest() {

}

@Before

public void setUp() throws Exception{

report=new Admin();

}

@After

public void tearDown() {

report=null;

}

@Test

public void testReport() {

}

@Test

public void testMain()throws Exception {

String view=report.Report();

assertEquals("View Report",view);

}}

**Chapter Six**

**Conclusion and Recommendation**

**6.1 Conclusion**

In analyzing of the existing manual system, we found that services in “cinema reservation system” are not enough to serve and a lot of problem. This manual system is highly time consuming. So that the existing system creates a lot of work load on employee. These lead to service delay, job and service dissatisfaction and inefficient, and user need to find resources from the office.

So, after we have completed this project, we are sure that the existing problems would answer. The” WEB BASED ONLINE CINEMA RESERVATION SYSTEM” minimizes the problem in the company.

In this project we try to gather different information about KIoT online cinema reservation system how to work their management activities. Such kinds of data gathering procedures help us to web based online cinema reservation system by using all software development life cycle like requirement gathering, requirement specification, system design, implementation, and configuration of the system.

The proposed system made computerized to reduce human errors/weakness and to increase the efficiency. The main focus of this project is to lessen human efforts. The maintenance of the records is made efficient, as all the records are stored in the database, and also data can be retrieved easily. The editing is also made simpler.

Our main aim of the project is:

* Reduce the number of employees.
* Minimize the time required to perform task.
* Minimize the work load of employees.
* Provide enough security.
* Increase employee satisfaction.

**5.2 Recommendation**

Finally, the team would recommend that further work should done on the system in order to make the system perform better for organization who like to use web based online cinema reservation system.

While doing this system the team members has faced different challenges. But by the cooperation of all the group members and the advisor the team is now able to reach to the final result. I.e. all the group members strongly fight these challenges and take the turn to the front.

So now all the group members strongly recommend the department that for the coming students, it has to provide them with better service than the present in better hard ware, guaranteed software’s, giving orientations how to proceed, offering guest to provide them with more experienced work, support morally, manually, forming good relation with students, giving students description of each phases and so on. So that it will get what it expects from its students and satisfy with them.

**5.2 Reference**

We have searched and looked for various and different information’s from different positions, books and websites.

From these references of information some of them are listed below

* **Software Engineering A practitioner’s approach by Roger S Pressman**
* **Encyclopedia of Software Engineering by LaPlante LaPlante**
* [**www.Google.com**](http://www.Google.com)
* **Machine Learning Applications In Software Engineering (Series on Software Engineering and Knowledge Engineering)” by Du Zhang and Jeffrey J P Tsai**