**Product Requirements Specification**

**For**

**ARISE Flights**

**29th August 2019**

**Table of Contents**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Project Scope 1

a. References 3

**15 Overall Description 3**

a. Product Perspective 3

b. Product Features 3

c. User Classes and Characteristics 3

d. Operating Environment 4

e. Design and Implementation Constraints 4

f. User Documentation 4

g. Assumptions and Dependencies 4

**16 System Features 4**

**17 External Interface Requirements 10**

a. User Interfaces 10

b. Hardware Interfaces 10

c. Product Interfaces 10

d. Communications Interfaces 10

**18 Other Nonfunctional Requirements 10**

a. Performance Requirements 10

b. Safety Requirements 10

c. Security Requirements 10

d. Product Quality Attributes 10

**19 Other Requirements 10**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Details** | **Version** |
|  |  |  |  |
|  |  |  |  |

# **1.Introduction**

## **1.1Purpose**

This document presents a detailed explanation of the objectives, features, user interface and application of **Flight ReservationSystem** in real life. It will also describe how the system will perform and under which it must operate. In this document it will be also shown user interface. Both the stakeholders and the developers of the system can benefit from this document.

## **1.2Document Conventions**

The System name is highlighted all over the document with bolded letters as well as underlined as well. Irrespective of that there is no specific convention provide. Every requirement statement has it's own priority.

## **1.3 Intended Audience and Reading Suggestions**

This document is intended to provide a clear picture of the system for the users i.e passengers, admin. And the SRS document got divided into sections which are classified as the scope of the project, the overall description about the system, the system features, external interface requirements as well as the non functional requirements.

## **1.4Project Scope**

This system will help to book the flights. In this system, we will provide a website that can be used by the customers for flight reservation and check the availability of flights . Customers can also give feedback through this website. Customers can also make payment through debit or credit cards . This website provide facility to keep track of passengers and flight .

# **2. Overall Description**

The Overall Description section, of this document gives an overview of the functionality of the product . It describes the informal requirements and is used to establish a context for the technical requirements specification in the next heading.

The Product feature, Operating Environment, Design and Implementation Constraints, of this document is written primarily for the developers and describes the details of the functionality of the product. This system will be completely web-based, linking to **Flight Reservation System** and the remote web server from a standard web browser. An Internet connection is necessary to access the system.

## **2.1Product Perspective**

* Flight ReservationSystem is a self contained system. It is aimed towards reservation of flight tickets .

**Product Features**

* It is easy to use the product with all the information provided in the form of links.
* The user can access the portal with simple registration.
* Passengers can search flight , book tickets , check status , cancelation of tickets.
* Admin can check status of passengers , flight details .

## **User Classes and Characteristics**

The user classes will be customer and Administrator or admin.

## **Operating Environment**

## **Technologies to be used**

## Programming languages:

* **JAVA EE**: Java Enterprise Edition is a programming platform— part of the Java Platform-for developing and running distributed multi-tier architecture Java applications, based largely on modular product components running on an application server.
* **HTML**: Hyper Text Markup is the predominant markup language for web pages. It provides a means to describe the structure of text-based information in a document and to supplement that text with interactive forms, embedded images, and other objects.
* **Bootstrap**: A framework for HTML , CSS and JavaScript .
* **MySql :** It is an Oracle-based open source relational database management system(RDBMS) based on Structure Query Language

**Tools & Development Environment**:

* **Apache Tomcat 9.0.12 Server**: Apache Tomcat is a Servlet container developed by the Apache Product Foundation (ASF). Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run.
* **Eclipse IDE** is a modular, standards-based integrated development environment (IDE), written in the Java programming language. The NetBeans project consists of a full-featured open source IDE written in the Java programming language and a rich client application platform, which can be used as a generic framework to build any kind of application.

## **Design and Implementation Constraints**

* There is no maintainability of back up so availability will get affected.
* Limited to HTTP/HTTPS Protocols.
* No multilingual support

## **User Documentation**

* The user should be familiar with the Computer
* The user should be familiar with the Internet.

## **Assumptions and Dependencies**

* The details related to the , passengers , payment and service transaction provided online.
* Administrator is created in the system already.
* Roles and tasks are predefined.
* Roles and responsibilities are already established.

# **3.System Features**

**Registering New Users in the Database:-**

3.1.1 Description and Priority

This feature will enable the new user(only customer) so that the generation of profile for them may be done.

3.1.2 Stimulus/Response Sequences

This form will consist of basic fields such as Name, Username, E-mail Id , gender and mobile no . There are two buttons: Register and Reset. Register will submit the data to the database at the server tier, and as expected Reset will reset the input values of all the fields.

3.1.3 Functional Requirements

The most important requirement here is to input values in the database and store them there for future use. To implement the security and to ensure that no android is filling in the registration forms,the user has

to enter the id generated during registration . If any field is left to provide the data, the system will prompt the user by using the scripts and will not submit the data until corrections/data entries are made completely.

**Secure Login to the interface:-**

3.2.1 Description and Priority

This feature will enable the user to have a secure and simple login to the system. To avoid handling a large number of errors and exceptions this feature will enable the user to provide only a limited number of inputs having constraints upon them and if there are any errors the system will notify the user about them.

3.2.2 Stimulus/Response Sequences

It will consist of two basic fields Username and Password . There are two buttons: Login and Forgot password. Login will submit the entered data for approval followed by access, and Forgot password button will change the details of the user.

3.2.3 Functional Requirements

The most important function is to only grant access to users that are listed in the database. The customer will provide the information on who will be allowed access. To implement the security, the web page must check the database to see if the Username and Password are valid. If they are not, the user will receive an “Enter correct username and password ” as a response.

**Searching for Flight:-**

3.3.1 Description and Priority:

This feature will enable the user when he/she has successfully logged into the system or portal here to search for a flight according to their needs and requirements. To avoid handling a large number of errors and exceptions this feature will enable the user to enter only a limited number of inputs having constraints upon them and if there are any errors the system will notify the user about them.

3.3.2 Stimulus/Response Sequences

This will consists of basic fields such as from and to, date of journey. There is single button search flights.

3.3.3 Functional Requirements

The important function here is to suggest users a list of available flights according to the provided information. The user will then upon his sole discretion will select a suitable flight for him/her and will proceed to the payments . If no suggestions are found then the system will return no records.

# **4.External Interface Requirements**

# **4.1 User Interfaces**

The user interface is screen shown on the browser. The Home screen of the Web-Portal is where Customer can register and login. The portal screen acts as an interface to provide services to the user which are to be availed from the database.

# **4.2 Hardware Interfaces**

A minimum of 40GB of HDD, with Pentium IV processor, a minimum of 256MB of RAM so that a suitable OS (Windows XP ) may be installed, and a reliable internet connection is required for the client side/user side so that may be accessed easily.

## **4.3 Product Interface**

The system uses:

**JSP**: Java Server Pages. It is a technology that helps product developers serve dynamically generated web pages based on HTML, XML and other document types; uses java programming language.

**Servlet**: Java web-containers which holds actions to be performed; a Servlet a java programming language class used to extend the capabilities of servers that host applications access via a request response programming model.

**d. Communications Interfaces**

Internet connection and Browser are required in order for several functions to be executed such as booking tickets . The system uses the following browsers:-

## 5.Mozilla Firefox

## 6.Google Chrome

## **7.** Internet Explorer

# **5. Other Non-functional Requirements**

## **a. Performance Requirements**

Some Performance requirements identified is listed below

1. The database must be support more than 100 student, tutor and parents record.
2. Can support many user at the same time.
3. High speed internet .

## **b. Safety Requirements**

The details provided by the users should be authentic .

User should not share their login details with anyone .

## **c. Security Requirements**

Some of the factors that are identified to protect the product from accidental or malicious access, use, modification, destruction, or disclosure are described below. Specific requirements in this are could include the need to:

1. Keep specific log or history data sets.
2. Check data integrity for critical variables.
3. Communication needs to be restricted when the application is validating the user .
4. Providing Authentication.

## **d. Product Quality Attributes**

There are a number of attributes of product that can serve as requirements. It is important that required attributes should be specified so that their achievement can be objectively verified. The following terms provide a partial list of examples

**Portability**

Some of the attributes of product that relate to the ease of porting the product to other host machines and/or operating systems. This may include: Java is used to develop the product. So it is easiest to port the product in any environment.

**Maintainability**

The user will be able to reset all options and all stored user variables to default settings.

**Reliability**

Some of the attributes identified for the reliability is listed below:

1. All data storage for user variables will be committed to the database at the time of entry.

2. Data corruption is prevented by applying the possible backup procedures and techniques.

**Usability requirements**

Some of the usability requirements identified for this system are listed below:

* + - 1. A logical interface is essential to an easy to use system, speeding up common tasks.
      2. Error prevention is integral to the system and is provided in a number of formats from sanity checks to limiting free-text input.

**Availability:**

All cached data will be rebuilt during every startup. There is no recovery of user data if it is lost. Default values of system data will be assigned when necessary.

# **8.Other Requirements**

**Immediate Feedback:**

The System must try to answer all the queries of the user and it should provide immediate feedback after getting any request from the customer . The system must provide the illusion to the user that , they are in contact to administrator of the web-site.

**Increase the Quality of the Process:**

The system must provide variety of flights available for users at reasonable price.

**Make the Interface Simple as Possible:**

The System must provide the simple and easy interface for beginners and also provide facilities for technical peoples who are using the system. The interface must be simple as possible.

**Reduced Time:**

To perform any task time is one of the important factors to consider. If the system not utilize properly time, than the entire aim of system is fails and the system is fails to reach its goal. So time take to process all these activities should be less but the output should be effective.

**Appendix A: Glossary**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Database | Collection of all the information monitored by this system. |
| Administrator | Is a person responsible for maintaining one or many websites. The duties of the webmaster may include ensuring that the web servers, hardware and product are operating correctly, designing the website, generating and revising web pages, replying to user comments, and examining traffic through the site. |
| HTML | Hypertext Transfer Protocol is a transaction oriented client/server protocol between a web browser & a Web Server |
| HTTPS | Secure Hypertext Transfer Protocol is a HTTP over SSL (secure socket layer). |
| SRS(Product Requirements Specification) | A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document. |
| Stakeholder | Any person with an interest in the project who is not a developer. |
| User | Student or tutor or parent |
| IDE | An integrated development environment (also termed integrated design environment, integrated debugging environment or interactive development environment) is a [product application](http://en.wikipedia.org/wiki/Software_application) that provides comprehensive facilities to [computer programmers](http://en.wikipedia.org/wiki/Computer_programmer) for [product development](http://en.wikipedia.org/wiki/Software_development) |
| Email | Electronic mail, commonly known as email or e-mail, is a method of exchanging  digital messages from an author to one or more recipients |

# **Appendix B: Analysis Models**

Under the analysis model, we analyze the system to check the following:

1. Whether it meets the requirements that guided its design and development;
2. Works as expected; and
3. Can be implemented with the same characteristics.

To perform these analyses of the model, the following testing is to be implemented:-

**Unit testing**: Unit testing, also known as component testing, refers to tests that verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.

**Integration testing**: Integration testing is any type of product testing that seeks to verify the interfaces between components against a product design. Product components may be integrated in an iterative way or all together ("big bang").

Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested product components corresponding to elements of the architectural design are integrated and tested until the product works as a system.

**System Testing**: system testing is done to ensure whether the system meet all the requirements stated in the SRS.

System testing is performed on the entire system in the context of a [Functional Requirement](http://en.wikipedia.org/wiki/Functional_requirements) Specification(s) (FRS) and/or a [System Requirement](http://en.wikipedia.org/wiki/Requirements_analysis) Specification (SRS). System testing tests not only the design, but also the behavior and even the believed expectations of the customer. It is also intended to test up to and beyond the bounds defined in the product/hardware requirements specification(s).

The system testing is categorized into three:

#### ***Alpha testing***

Alpha testing is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf product as a form of internal acceptance testing, before the product goes to beta testing.

**Beta testing**

Beta testing comes after alpha testing and can be considered a form of external [user acceptance testing](http://en.wikipedia.org/wiki/User_acceptance_testing). Versions of the product, known as [beta versions](http://en.wikipedia.org/wiki/Beta_version), are released to a limited audience outside of the programming team. The product is released to groups of people so that further testing can ensure the product has few faults or [bugs](http://en.wikipedia.org/wiki/Computer_bug). Sometimes, beta versions are made available to the open public to increase the [feedback](http://en.wikipedia.org/wiki/Feedback" \l "In_organizations) field to a maximal number of future users.

**Acceptance testing**

Acceptance testing performed by the customer, often in their lab environment on their own hardware, is known as [user acceptance testing](http://en.wikipedia.org/wiki/User_acceptance_testing) (UAT). Acceptance testing may be performed as part of the hand-off process between any two phases of development.

# **Appendix C: Issues List**

*The problems that might occur with the product product are:*

* It might occur that a customer searches for a flight but the flight is not available.
* No transaction is flawless. Transaction can break at multiple points during the process.
* The portal is dependent on web services. Though we make every effort to ensure that services are provided on time but there is no guarantee The problems that might occur are:
  + Some web pages are lost . The user is looking for a specific Web page but try as they might, they can't find it.
  + Web pages load slow or incorrectly. The user found the Web page he wanted but it took forever to load or things are jumping around on the page while loading.
  + **JavaScript Errors.**

Forms are completely broken. After clicking submit button, an error might occur.

* Big security vulnerability. Someone to steal your login information and hack into your account
* Broken Registration Process
* Site won't load. Websites are supposed to work fine whether you type in the "www" or not. But an error might occur.

**Class Design Specification**

**DOCUMENT VERSION HISTORY**

|  |  |  |
| --- | --- | --- |
| **Document Version** | **Date Issued** | **Comments** |
| 1.0 | 02-04-2012 | First version of the Web Portal |

# **1Table of Contents**

[Table of Contents 2](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=en-IN&rs=en-IN&hid=%2F8D097WWQUiwFSSXPQ6Law.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F58E6599A058132F1!2500&wdnd=1&wdprevioussession=ffda394e-0d55-497e-ba72-4308d86ab4bd&wdnewandopenct=1568116672972&wdo=7&wdpreviouscorrelation=5820268e-9df5-4873-948a-9b59aea6b4a5&wde=docx&sc=host=&qt=Folders&mscc=1&wdp=0&uih=OneDrive&wdorigin=Unknown&jsapi=1&newsession=1&corrid=159c6200-4886-41c7-86c8-3c2d903ace8e&usid=159c6200-4886-41c7-86c8-3c2d903ace8e&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlushFallback" \l "__RefHeading___Toc219268173)

[Class Design](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=en-IN&rs=en-IN&hid=%2F8D097WWQUiwFSSXPQ6Law.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F58E6599A058132F1!2500&wdnd=1&wdprevioussession=ffda394e-0d55-497e-ba72-4308d86ab4bd&wdnewandopenct=1568116672972&wdo=7&wdpreviouscorrelation=5820268e-9df5-4873-948a-9b59aea6b4a5&wde=docx&sc=host=&qt=Folders&mscc=1&wdp=0&uih=OneDrive&wdorigin=Unknown&jsapi=1&newsession=1&corrid=159c6200-4886-41c7-86c8-3c2d903ace8e&usid=159c6200-4886-41c7-86c8-3c2d903ace8e&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlushFallback" \l "__RefHeading___Toc219268174)

[1](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=en-IN&rs=en-IN&hid=%2F8D097WWQUiwFSSXPQ6Law.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F58E6599A058132F1!2500&wdnd=1&wdprevioussession=ffda394e-0d55-497e-ba72-4308d86ab4bd&wdnewandopenct=1568116672972&wdo=7&wdpreviouscorrelation=5820268e-9df5-4873-948a-9b59aea6b4a5&wde=docx&sc=host=&qt=Folders&mscc=1&wdp=0&uih=OneDrive&wdorigin=Unknown&jsapi=1&newsession=1&corrid=159c6200-4886-41c7-86c8-3c2d903ace8e&usid=159c6200-4886-41c7-86c8-3c2d903ace8e&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlushFallback" \l "__RefHeading___Toc219268175) Introduction 3

[2](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=en-IN&rs=en-IN&hid=%2F8D097WWQUiwFSSXPQ6Law.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F58E6599A058132F1!2500&wdnd=1&wdprevioussession=ffda394e-0d55-497e-ba72-4308d86ab4bd&wdnewandopenct=1568116672972&wdo=7&wdpreviouscorrelation=5820268e-9df5-4873-948a-9b59aea6b4a5&wde=docx&sc=host=&qt=Folders&mscc=1&wdp=0&uih=OneDrive&wdorigin=Unknown&jsapi=1&newsession=1&corrid=159c6200-4886-41c7-86c8-3c2d903ace8e&usid=159c6200-4886-41c7-86c8-3c2d903ace8e&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlushFallback" \l "__RefHeading___Toc219268176) Conventions followed 3

[3](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=en-IN&rs=en-IN&hid=%2F8D097WWQUiwFSSXPQ6Law.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F58E6599A058132F1!2500&wdnd=1&wdprevioussession=ffda394e-0d55-497e-ba72-4308d86ab4bd&wdnewandopenct=1568116672972&wdo=7&wdpreviouscorrelation=5820268e-9df5-4873-948a-9b59aea6b4a5&wde=docx&sc=host=&qt=Folders&mscc=1&wdp=0&uih=OneDrive&wdorigin=Unknown&jsapi=1&newsession=1&corrid=159c6200-4886-41c7-86c8-3c2d903ace8e&usid=159c6200-4886-41c7-86c8-3c2d903ace8e&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlushFallback" \l "__RefHeading___Toc219268177) Class Diagrams 4

## 

# **2. Introduction**

The class diagram is the main building block of [object oriented](http://en.wikipedia.org/wiki/Object_oriented) modelling. It is used both for general [conceptual modelling](http://en.wikipedia.org/wiki/Conceptual_model) of the systematics of the application, and for detailed modelling translating the models into [programming code](http://en.wikipedia.org/wiki/Programming_code). Class diagrams can also be used for [data modeling](http://en.wikipedia.org/wiki/Data_modeling). The classes in a class diagram represent both the main objects and or interactions in the application and the objects to be programmed. In the class diagram these classes are represented with boxes which contain three parts:

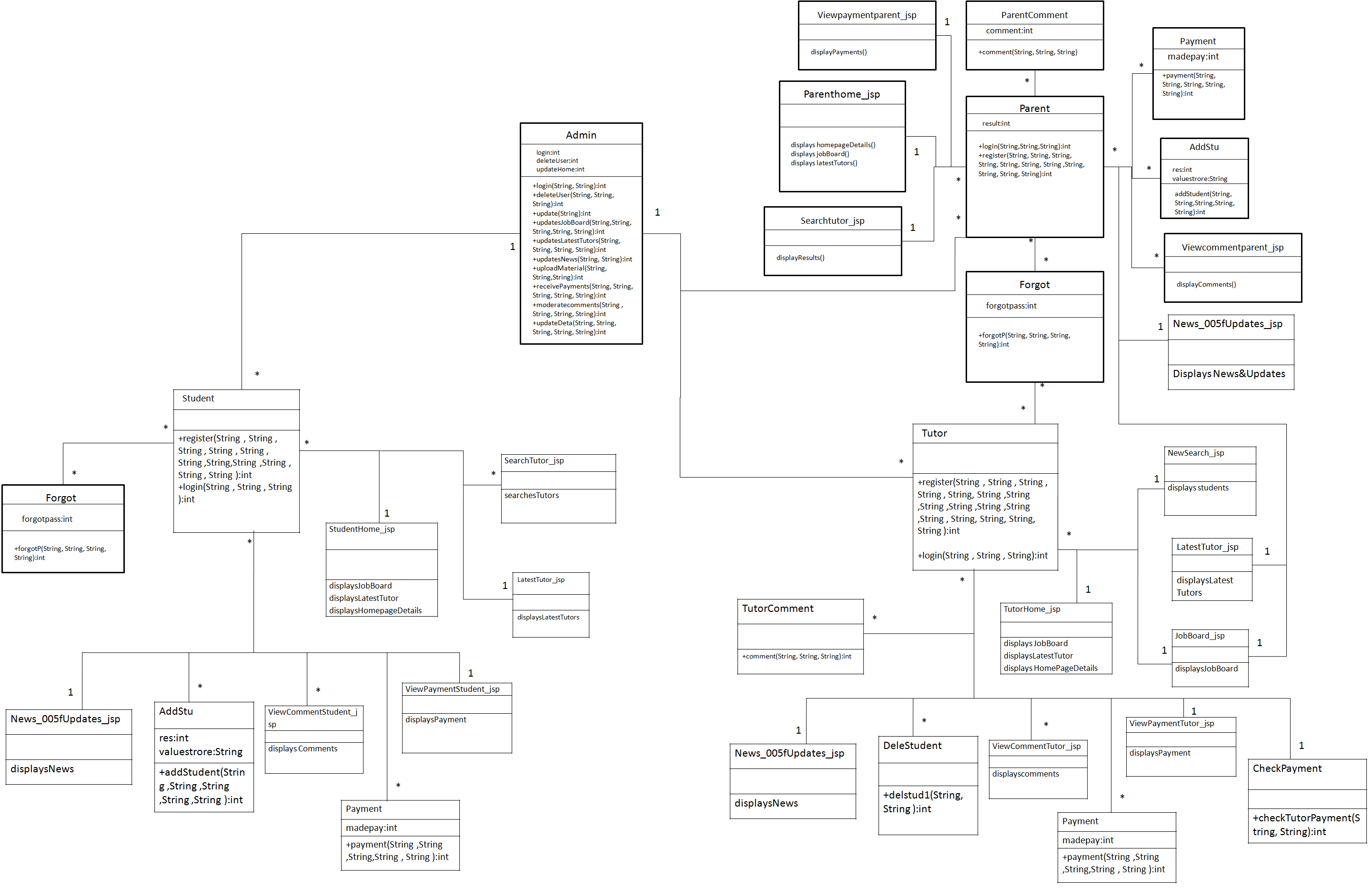
A class with three sections:

* The upper part holds the name of the class
* The middle part contains the attributes of the class
* The bottom part gives the methods or operations the class can take or undertake

# **3Conventions followed**

Hungarian Convention

# **4Class Diagrams**



# **5Classes**

## **5.5Admin**

**Class Description :**

**The Administrator class holds those attributes and functions which may be used for the functioning of the system. The administrator manages tutor, parent and student. The administrator uploads online materials and moderates comments.**

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **deleteUser(String, String, String)** | **int** | This function deletes the existing user from the database |
| **2** | **login(String, String)** | int | checks for the authentic username and password and provides the login |
| **3** | **updateDeta(String, String, String, String, String)** | int | the administrator make changes in the user details, qualifications, etc |
| **4** | **receivePayments(String, String, String, String, String)** | int | enables to receive payments from the user |
| **5** | **uploadMaterials(String, String, String)** | int | the administrator uploads materials which can be accessed online or downloaded and can be used by user for learning |
| **6** | **update(String)** | int | the administrator keeps the website updated by updating the home page details |
| **7** | **moderatecomments(String, String, String, String)** | int | the administrator views the comments and moderates them. |
| **8** | **updatesJobBoard(String, String, String, String, String)** | int | the administrator updates the latest jobs available and is very useful for the tutors who want good money and students |
| **9** | **updatesLatestTutors(String, String, String, String)** | int | the administrator updates the latest tutors i.e. give the new tutors the chance to establish themselves and get good monetary value |
| **10** | **updateNews(String, String)** | int | the administrator keeps its user updated with the world by giving them regular updates about the new technologies and subjects |

## **5.6Student**

**Class Description :**

**The Student class contains the attributes and related functionality for the students**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **register(String, String, String, String, String, String, String, String, String, String, String)** | int | registers the new student in the portal |
| **2** | **login(String, String,String)** | int | authenticates the student to login |

### **5.6.1Forgot**

**Class Description :**

**The Forgot class enables the student user to reset his username and password if he/she has forgot it.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **forgot(String, String, String, String)** | int | This function enables the student to reset his password.  updates the username, password to the new one |

### **5.6.2StudentHome\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displaysJobBoard** | displays the JobBoard |  |
| **2** | **displayLatestTutor** | displays the Latest Tutors |  |
| **3** | **displaysHomePageDetails** | displays the homepage details |  |

### **5.6.3SearchTutor\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **searchesTutors** | displays the list of tutor’s on the searched query |  |

### **5.6.4LatestTutor\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayLatestTutors** | displays the latest tutors added in the portal |  |

### **5.6.5News\_005fUpdates\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayNews** | displays the latest news subjects and their links |  |

### **5.6.6AddStu**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **addStudent(String, String, String, String, String)** | int | enables the student to get added in the batch of the particular tutor |

### **5.6.7Viewcommentstudent\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayComments** | displays the comments done by the tutor on the student |  |

### **5.6.8Payment**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **payment(String, String, String, String, String)** | int | enables the student to make payments to study under a particular tutor, or download a material |

### **5.6.9ViewpaymentStudent\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayPayment** | displays the payment done by the student for various reasons |  |

## **5.7Parent**

**Class Description :**

**The Parent class contains the attributes and related functionality for the parent**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **register(String, String, String, String, String, String, String, String, String, String, String)** | int | registers the new parent in the portal |
| **2** | **login(String, String,String)** | int | authenticates the parent to login |

### **5.7.1Forgot**

**Class Description :**

**The Forgot class enables the parent user to reset his username and password if he/she has forgot it.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **forgot(String, String, String, String)** | int | This function enables the parent to reset his password. |

### **5.7.2ParentHome\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displaysJobBoard** | displays the JobBoard |  |
| **2** | **displayLatestTutor** | displays the Latest Tutors |  |
| **3** | **displays HomePageDetails** | displays the homepage details |  |

### **5.7.3SearchTutor\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displaysResults** | displays the List of tutors on the searched query |  |

### **5.7.4LatestTutor\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayLatestTutors** | displays the latest tutors added in the portal |  |

### **5.7.5News\_005fUpdates\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayNews** | displays the latest news subjects and their links |  |

### **5.7.6AddStu**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **addStudent(String, String, String, String, String):int** | enables the parent’s ward to get added in the batch of the particular tutor |  |

### **5.7.7Viewcommentparent\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayComments()** | displays the comments done by the tutor on the parent’s ward |  |

### **5.7.8Payment**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **payment(String, String, String, String, String)** | int | enables the student to make payments to study under a particular tutor, or download a material |

### **5.7.9ViewpaymentParent\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayPayment** | displays the payment done by the parent for various reasons |  |

### **5.7.10ParentComment**

**Class Description :**

**This class enables the parent to comment on the performance of the particular Tutor.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **comment(String, String, String)** | int | enables the parent to comment on the particular tutor |

## **5.8Tutor**

**Class Description :**

**The Tutor class contains the attributes and related functionality for the tutor**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **register(String, String, String, String, String, String, String, String, String, String, String, String, String, String, String)** | int | registers the new tutor in the portal |
| **2** | **login(String, String,String)** | int | authenticates the tutor to login to his account |

### **5.8.1Forgot**

**Class Description :**

**The Forgot class enables the tutor user to reset his username and password if he/she has forgot it.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **forgot(String, String, String, String)** | int | This function enables the tutor to reset his password. |

### **5.8.2TutorHome\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displaysJobBoard** | displays the JobBoard |  |
| **2** | **displayLatestTutor** | displays the Latest Tutors |  |
| **3** | **displays HomePageDetails** | displays the homepage details |  |

### **5.8.3NewSearch\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displaysstudents** | displays the List of students in the batch of the tutor |  |

### **5.8.4LatestTutor\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayLatestTutors** | displays the latest tutors added in the portal |  |

### **5.8.5News\_005fUpdates\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayNews** | displays the latest news subjects and their links |  |

### **5.8.6DeleSudent**

**Class Description :**

**This class enables the tutor to delete the student from the batch.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **delstud1(String, String)** | int | enables the tutor to delete the student from the Batch |

### **5.8.7ViewcommentTutor\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayComments()** | displays the comments done by the tutor on the parent’s ward |  |

### **5.8.8Payment**

**Class Description :**

**This class enables the tutor to store payments done by the tutor.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **payment(String, String, String, String, String)** | int | enables the tutor to make payments to take classes. |

### **5.8.9ViewpaymentTutor\_jsp**

**Class Description :**

**This class holds the business logic to display the values stored in the database.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **displayPayment** | displays the payment done by the tutor for various reasons |  |

### **5.8.10TutorComment**

**Class Description :**

**This class enables the tutor to comment on the performance of the particular student.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **comment(String, String, String)** | int | enables the tutor to comment on the particular student |

### **5.8.11CheckPayment**

**Class Description :**

**This class enables the tutor to check for payments made by the Tutor.**

Methods

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Function** | **OutPut Parameters** | **Description** |
| **1** | **checkTutorPayment(String, String)** | int | enables the tutor to check for the payments made |

**Airline Reservation System Use Case Diagram**

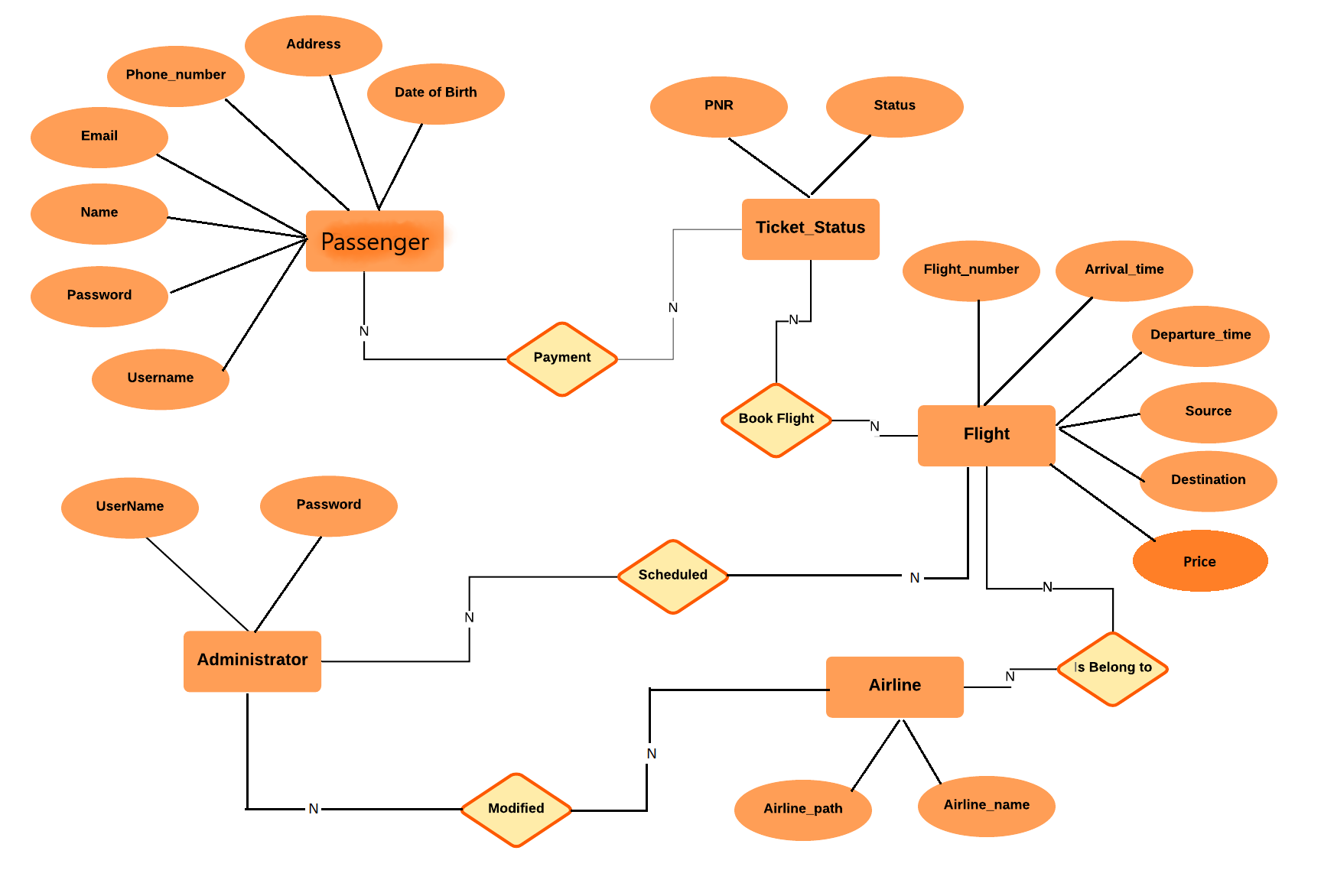


**There are two Actors in Use Case Diagram:**

* **User/Passenger:** The User/Passenger is a class. It has some attributes like Username, Password, Name, Phone\_number, Email, Address, Date of Birth. These attributes are use to perform the following operation like Search Ticket, Choose Flight, Book/Cancel Ticket, Review Order, Proceed to checkout, Make Payment.
* **Administrator:** The Administrator is a class. it has some attribute like Username, Password. These attribute use to perform the following operation like Manage passenger, Manage Reservation, Review Order, Manage Airlines.

**Airline Reservation System ER-Diagram**

This­­­­­ ER(entity Diagram) represents the model of Airline Reservation System Entity.The main entity of Entity of Airline Reservation System are Passenger,Administrator,Ticket\_Status,Flight,Airline.



**Airline Reservation System entity and their Attributes:**

1. **Administrator:** The Attributes of Administrator are Username and Password.
2. **Passenger:** The Attributes of Passenger are Ussername, Password, Name, Email, Phone\_number, Address, Date of Birth.
3. **Flight:** The Attributes of Flight are Flight\_number, Arrival\_time, Departure\_time, Source, Destination, Price.
4. **Airline:** The Attributes of Airline are Airline\_name, Airline \_path.
5. **Ticket\_Status:** The Attributes of Ticket\_Status are PNR and Status.

**Information Gathering**

Project Name : AIRLINE RESERVATION SYSTEM

**Submitter Name** : SHANTANU VERMA, RASHI BHARGAVA, SAURABH KUMAR SINGH, SIMRAN DHIMAAN, RAVI AGGRAWAL

**Date Submitted**: 29/08/2019

**ABOUT AIRLINE RESERVATION SYSTEM:**

For the people who love travelling, want to explore and want to take a flight to their destination, we are here. **Arise Airline reservation system** is a place where you can find too many flights to their particular destinations that too of your choice. You can book any flight according to your will and have a peaceful and happy journey to your place.

**DIFFERENT FUNCTIONALITIES OF AIRLINE RESERVATION SYSTEM:**

* Providing the best flight that too according to your preference and choice.
* Helping you to take the flight that suits your comfortable zone that too at suitable prices.
* Minimum cancellation charges with refunding price within 24 hours.
* Affordable prices for flight bookings.
* Minimum delays and cancellation of flights.

**ANALYSIS OF EXISTING AIRLINE RESERVATION PROCESS:**

1. Processing the flight details you want to travel in and take.
2. Booking the flight of your choice.

**CHALLENGES IN EXISTING AIRLINE RESERVATION PROCESS:**

The existing Airline Reservation System provides the online flight bookings and facilities but fails to provide minimum cancellation charges and provides maximum delays and cancellation of flights.

# Introduction

Airline reservation systems were first introduced in the late 1950s as relatively simple standalone systems to control flight inventory, maintain flight schedules, seat assignments and aircraft loading. The modern airline reservation system is comprehensive suite of products to provide a system that assists with a variety of airline management tasks and service customer needs from the time of initial reservation through completion of the flight.

One of the most common mode of travel is travelling by air. Customers who wish to travel by air nowadays have a wide variety of airlines and a range of timings to choose from. Nowadays competition is so fierce between airlines that there are lot of discounts and a lot of luxuries given to customers that will give an edge to that particular airline.

The World Wide Web has become tremendously popular over the last four years and currently most of the airlines have made provision for online reservation of their flights. The Internet has become a major resource for people looking for making reservations online without the hassle of meeting travel agents. My Project intends to serve the purposes. It intends to check all the available airline databases and return a string the results, which can help them in their travel plans.

The objective of this project is to create an airline reservation system where a traveler can request all flight information as per their journey dates. They can get information regarding time, cost, etc all at the same time and place. When the customer calls the Counter Assistant for his/her travel needs, the counter assistant will enter the customers details (flight requirements) in the system. The system displays all the available airlines, schedules and prices. This system would help their airline to better serve its customers by catering to their needs. The site would use a Database to hold this information as well as the latest pricing and availability information for the airlines

## Purpose of Project

This document presents a detailed explanation of the objectives, features, user interface and application of **Flight Reservation System** in real life. It will also describe how the system will perform and under which it must operate. In this document it will be also shown user interface. Both the stakeholders and the developers of the system can benefit from this document.

The main purposes of this project are:-

* It is easy to use the product with all the information provided in the form of links.
* The user can access the portal with simple registration.
* Passengers can search flight , book tickets , check status , cancelation of tickets.

Admin can check status of passengers , flight details

**Benefits of the Project:**

The system includes the students, parents, tutors. Benefits to each of these are described below.

**Benefits of the Project:**

The system includes the client, admin . Benefits to each of these are described below.

**Benefit to Client:**

With availability of flights the client may find it a flexible mode to reserve and cancel the flights in a better way, as he has guidance of the flights. The client can easily access the site as its user friendly. Easy reservation and cancellation provided.

**Benefit to Admin:**

Admin can easily control the information.

# project And Product Overview

The main purpose of this software is to reduce the manual errors involved in the Airline Reservation Process and make it convenient for the customer to book the flights else when they require such that they can utilize this software to make reservation, modify reservations or cancel a particular reservation.

The name of the Software is “AIRLINE RESERVATION SYSTEM” .This software provides options for viewing different flights available with different timings for a particular date and provides customer with the facilities to book a ticket, modify or cancel a particular reservation but it does not provide the customers with the details of cost of the ticket and it does not allow the customer to modify a particular part of his reservation and he/she can modify all details.

The Arise Airlines can be divided into four modules, namely:

**Module 1: Client**

This module is responsible for maintaining the overall system based on defined business rules and it perform the following functions:

1. It can delete user.
2. It can modify the user details.
3. It can update website details
4. Respond to e-mail request
5. It can receive payments(online).

**Module 2: Admin**

Admin is responsible for the whole reservation system and its modification providing the necessary information.

# Scope

## Objectives

1. The new user must register with the portal before availing any services from it.
2. For each registration maintaining the record in the database for the new user.
3. For each new registration a profile is to be created.
4. The payment could be done in two ways by offline payment (sending draft) or through online payment system (NEFT Fund Transfer).
5. To provide the correct details of the particular flight the user wants to take.

## High-Level Requirements

The following table presents the requirements that the project’s product, service or result must meet in order for the project objectives to be satisfied.

| Requirement | I Requirement Description |
| --- | --- |
| Registration | To Provide complete access to the users i.e.to become a valid user. |
| Transaction | Details of valid transaction and card details are retrieved from the  Bank. |
| Availability Of Flights | Flight will be available for limited passengers. |
| Network Connectivity | The user should have fast internet connection and should be  supported by an advanced browser. |
| Availability Of Online  Material | Only Valid member can access online material. Whenever he/she  wants to access he should login along with the transaction id. |
| Searching For Flights | Only a registered user can search for Flights. |

## Business Rules and Special consideratioNs

* To book the flight on time to avoid any delay.
* Cancellation charges will be refunded within 24 hours.
* Cancellation of any flight will be done 4 hours before the flight takes off.

## Major Deliverables/Milestones

[Provide a list of the major deliverables/milestones that will be completed by the end of this project. A deliverable is any unique and verifiable product, result or capability to perform a service that must be produced in order to complete a process, phase or project. A milestone is a key performance indicator that is typically reported to executives to indicate the project’s progress.]

| Major Deliverable | I Deliverable Description |
| --- | --- |
|  |  |
|  |  |
|  |  |

# 

## 

## Duration Timeline

[Provide an estimate of the project duration (e.g.18 months).You may provide a high-level timeline for the project if information is available at this time. This time estimate will be further refined in the Planning Phase of the project. If applicable, also state the expected life of the product. An example of a high-level timeline is provided below.]

System Development

Completed

Developed Prototype

Requirements Analysis

Completed

Project Plan Completed

System Development

Completed

Developed Prototype

Requirements Analysis

Completed

Project Plan Completed

20/08

22/08

24/08

28/08

# project understanding approval

The undersigned acknowledge they have reviewed the **Project Understanding Document** forthe **AIRLINE RESERVATION SYSTEM**  project.Changes to this document will be coordinated with and approved by the undersigned or their designated representatives.

[List the individuals whose signatures are desired. Examples of such individuals are Business Steward, Project Manager or Project Sponsor. Add additional lines for signature as necessary. Although signatures are desired, they are not always required to move forward with the practices outlined within this document.]

|  |  |  |  |
| --- | --- | --- | --- |
| Signature: |  | Date: |  |
| Print Name: | SHANTANU VERMA |  |  |
| Title: |  |  |  |
| Role: |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Signature: |  | Date: |  |
| Print Name: | RASHI BHARGAVA |  |  |
| Title: |  |  |  |
| Role: |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Signature: |  | Date: |  |
| Print Name: | SIMRAN DHIMAN |  |  |
| Title: |  |  |  |
| Role: |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Signature: |  | Date: |  |
| Print Name: | SAURABH KUMAR SINGH |  |  |
| Title: |  |  |  |
| Role: |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Signature: |  | Date: |  |
| Print Name: | RAVI AGGRAWAL |  |  |
| Title: |  |  |  |
| Role: |  |  |  |

APPENDIX A: REFERENCES

[Insert the name, version number, description, and physical location of any documents referenced in this document. Add rows to the table as necessary.]

The following table summarizes the documents referenced in this document.

|  |  |  |
| --- | --- | --- |
| **Document Name and Version** | **Description** | **Location** |
| *<Document Name and Version Number>* | *[Provide description of the document]* | *<URL or Network path where document is located>* |

APPENDIX B: KEY TERMS

The following table provides definitions for terms relevant to this document.

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **Credit/Debit Card** | means any credit/debit card issued on the account to the holder and must be a valid card. |
| **VISA/MasterCard** | types of cards issued by the banks which may be used for making payments. |
| **Transaction** | Transfer of monetary value from one bank account to another. |
| **Web Portal** | website bringing together information from diverse sources in a unified way. |
| **Authentic** | valid things, refers to the truthfulness of origins, attributions and commitment. |
| **Discretion** | depends upon the understanding of the user judgment power |
| **Contemporary** | relating to the current world |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |