School Management System

Student Registration Design Document

# Purpose

The document illustrates the design of student registration module.

# For the Associates:

Associates should refer the following documents to understand the screen design and the high level design.

* CaseStudyScreenDesign.doc – Ilustrates the screen design the associates to follow for developing the application.
* CaseStudyMasterDesign.docx – Ilustrates the class hierarchy, flow sequence , packaging details and exception handling mechanism for developing the application.

# Database Design

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name | Column Name | Column Discription | Data Type | Data Size | Primary Key |
| FEES\_INFO | STANDARD | Column is used for storing standard | varchar2 | 3 | Yes |
|  | AGE | Column is used for storing the age | Number | 12 | No |
|  | STDFEES | Column is used for storing the Fees | Number | (8,2 ) |  |
|  | TRANSPORT FEE | Column used for storing the transport fee | Number | (8,2) |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name | Column Name | Column Discription | Data Type | Data Size | Primary Key | Foreign Key |
| STUDENT  \_INFO | STUDENT\_ID | Column is used for storing student register number | varchar2 | 5 | Yes |  |
|  | STUDENT\_NAME | Column is used for storing student name | varchar2 | 30 | No |  |
|  | DATE\_OF\_BIRTH | Column is used for storing the student’s date of birth. | date |  |  |  |
|  | GUARDIAN\_TYPE | Column is used for storing the guardian type | char  Values can be ‘G’ or ‘P’ |  |  |  |
|  | GUARDIAN\_NAME | Column is used for storing the guardian name | varchar2 | 30 |  |  |
|  | ADDRESS | Column is used for storing the student address | varchar2 | 50 |  |  |
|  | COUNTRY\_ID | Column is used for storing the id of the country. | varchar2 | 5 |  | Yes  COUNTRY\_INFO |
|  | CONTACT\_NO | Column is used for storing the student contact number | number | 10 |  |  |
|  | MAIL\_ID | Column is used for storing the student mail id | varchar2 | 30 |  |  |
|  | GENDER | Column is used for storing the student gender | char  Values can be ‘M’ or ‘F’ |  |  |  |
|  | STANDARD | Column is used for the storing the student standard | Varchar2 | 3 |  | Yes  FEES\_INFO |
|  | STANDARD\_CATEGORY | Column is used for the storing the student standard category | char  Values can be ‘P’ or ‘S’ or ‘H’ |  |  |  |
|  | TRANSPORT | Column is used for storing the transport facility selection | char  Values can be ‘Y’ or ‘N’ |  |  |  |
|  | NET\_FEES | Column is used for storing the fees payable by the student | number | (7,2) |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name | Column Name | Column Discription | Data Type | Data Size | Primary Key | Foreign key |
| COUNTRY\_INFO | COUNTRY\_ID | Column is used for storing countryid | varchar2 | 5 | Yes |  |
|  | COUNTRY\_NAME | Column is used for storing the storing the country name | varchar2 | 25 |  |  |
|  | STATE\_NAME | Column is used for storing the state name | varchar2 | 25 |  |  |

# Required Objects, Services, and Libraries

### Value Objects:

|  |  |
| --- | --- |
| **Value Object Name** | **Value Object Description** |
| StudentVO | This VO is used for transferring the student details from presentation to the back end. |
| FeesVO | This VO is used for retrieving the standard fee and transport fee details of a student for transfering it from DAO to BO |

# Technical Specifications



## Register Student screen: This screen is used by admin to enter the student details into the system.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Data Element** | **Control type** | **Default Values** | **Editable Field** | **Restrictions** | **Mandatory/**  **Non Mandatory** | **Data Type** | **Data Size** |
| Student Name | Textbox | No | Yes | Only Characters | Mandatory | Varchar | 30 |
| DOB | Textbox with date picker | No | No |  | Mandatory |  |  |
| Guardian Type | RadioButton | No | No | Parent/Guardian | Mandatory |  |  |
| Guardian Name | Textbox | No | Yes | Only Characters | Mandatory | Varchar | 30 |
| Country | Dropdownbox | No | No | Country names should be populated from the country\_info table | Mandatory |  |  |
| State | Dropdown box | No | No | Based on the above country selection ,the state names should be populated from the country\_info table | Mandatory |  |  |
| Address | TextArea | No | Yes | AlphaNumeric | Mandatory | Varchar | 50 |
| Contact No. | Textbox | No | Yes | Only Numbers | Mandatory | Number | 10 |
| Mail Id | Textbox | No | Yes | Should be in a valid Email Format for example abc@xyz.com | Mandatory | Varchar | 30 |
| Date of Joining | Textbox with date picker | No | No |  | Mandatory |  |  |
| Gender | Radio Button | No | No | Male/Female | Mandatory |  |  |
| Standard | Dropdownbox | No | No | Should be populated from the fees\_info table | Mandatory | Varchar | 3 |
| Transport | Radio Button | No | No | Yes/No | Mandatory |  |  |
| Feeviewer | Link | No | No | By clicking the link, pop up window should be displayed with the fee details retrieved from the fees\_info table based on the standard | Mandatory |  |  |
| Register | Submit Button | No | No | By Clicking the button, all the field contents will be validated and stores all the details into the system | Mandatory |  |  |
| Reset | Button | No | No | By Clicking the button, the contents in all the fields will be cleared. | Mandatory |  |  |

On successful registration the following message should be displayed **“Student Registered Successfully”**

## Application Components Specification:

Refer to the sequence and class diagram in the master design document for details on the components hierarchy, how they structured and integrated.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Component Name** | **Method Name** | **Method input** | **Method Output** | **Description** | **Algorithm** |
| SchoolRegistrationDelegate | registerStudent | StudentVO | boolean | This method invokes the facades and throws School and SchoolBusiness Exception. | Not applicable |
| ServiceLocator | Locate | Façade name | Façade Object | Method used for locating the appropriate façade based on the input name and return the object. Throws SchoolException when any runtime error occurs. | Not applicable. |
| SchoolRegistrationFacade | registerStudent | StudentVO | Boolean | Method invokes the appropriate BO’s and returns the value to the screens. Handles exceptions , logs it and throws it back to the presentation tier. | Not applicable. |
| RegistrationBO | registerStudent | StudentVO | Boolean | Method perform validations , calculates the fees and invokes the DAO to store the student details in the system. Throws School and SchoolBusinessException. | Refer to section 6.2  for fee calculation details and the validations to be done. |
| StudentDAO | retrieveTotalRegistrations |  | Int | This method should return the total numbers registrations in the system from student\_info table . This is used for the generation of the new student registration numbers by the system. | Refer to section 6.2.1  for retrieve the maximum register number existing in the system from Student\_info table. |
| StudentDAO | insertStudent | StudentVO | Boolean | Loads the student\_info table with the details in studentVO | NA |
| StudentDAO | retrieveFees | FeesVO | Float | Retrieves the standard fee and transport fee from the fees\_info table based on the given requirement and Calculate the net fees | Refer to section 6.2  For Calculating the Net Fees |



# Algorithms



### Validations

* If(student name contains characters other than alphabets) throw business exception with error code 501
* If(Email Id does not contains @ and .) throw business exception with error code 502
* Based on the given DOB, student\_age should be calculated and then student\_age will be compared with the standard\_age of the selected standard given in the database. If the student\_age is less than standard \_age,throw business exception with error code 503
* If DOB greater than sysdate throw business exception with error code 504
* If DOJ lesser than sysdate throw business exception with error code 505
* If(Contact no does contain alphabets and >10 digits) throw business exception with error code 506.

### Fee Calculation

**Step 1 :** Get the standard fees for the standard from the fees\_info table using the retrieveFees method of the DAO.

**Step 2 :** Calculate the net fee as per the conditions below

1. If the student is using the transport, then retrieve the transport fee from the fees\_info table using the retrieveFees method of DAO.
2. If the student is not using the transport, then assign zero to the transport fee.

**Step 3:** Calculate the net fees

net fees=standardfees+transport fee

### Student Registration

**Step 1** : Perform all the business validation

**Step 2** :Generate the register number for the student

Calculate the register number as given below

1. Get the total number of registrations in the system
2. Generate the register number as one more than the total number as provided below

Register Number = Total Registrations+1

**Step 3:**Identify the standard Category(P/S/H) based on the student Standard.

**Step 4:** Calculate the net fees

**Step 5:** Load all the student details into the student\_info table.