A MAJOR-PROJECT REPORT

ON

***“TEACHER-STUDENT CONNECT”***

Submitted to

CHANDIGARH ENGINEERING COLLEGE

BY

RYTHEM BANSAL (1502972)

*UNDER THE GUIDANCE OF PROF.* ***VASUNDHRA BAGDE***

SCHOOL OF COMPUTER ENGINEERING

CHANDIGARH ENGINEERING COLLEGE

LANDRAN,MOHALI.

**CERTIFICATE**

This is to certify that the project report entitled ***“Teacher-Student Connect”* by Rythem Bansal (1502972)** of **Chandigarh Engineering College, Landran (Mohali)** affiliated to **Punjab Technical University** during the academic year 2019, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (Computer Science and Engineering) is a bonafide record of the work carried out under our guidance and supervision at     **Cognizant Technology Solutions CDC, Pune.**

.

                                                                 Shilpa Mahajani

                                                                  (Project Manager)

 Cognizant Technology Solutions

                                                                          CDC, Pune

**ACKNOWLEDGEMENT**

We feel immense pleasure and feel privileged in expressing our deepest and most sincere gratitude to our supervisor **Vasundhra Bagde**, for his excellent guidance throughout our project work. His kindness, dedication, hard work and attention to detail have been a great inspiration to us. Our heartfelt thanks to you sir for the unlimited support and patience shown to us. We would particularly like to thank him for all her help in patiently and carefully correcting all our manuscripts. We acknowledge the support received from Cognizant Technical Solutions CDC, Pune. We are also very thankful to **Mrs. Shilpa** and **Mr. Mohit** for their support and suggestions during our course of the project work in the final year of our undergraduate course.

**Rythem Bansal**

**(1502972)**

ABSTRACT

**Cognizant** is an American multinational corporation that provides IT services, including digital, technology, consulting, and operations services. It is headquartered in Teaneck, New Jersey,United States of America. Cognizant is included in the NASDAQ-100 and the S&P 500 indices. It is also one of the fastest growing Fortune 500 companies. It was founded as an in-house technology unit of DUN & BRADSTREEIN in 1994, and started serving external clients in 1996.

Cognizant had a period of fast growth during the 2000s, becoming a Fortune 500 company in 2011. In 2015, the FORTUNE Magazine named it as the world's fourth most admired IT Services company. In 2017, Cognizant was named in Fortune’s Future 50 list.

Cognizant provides information technology, information security, consulting, ITO and BPO services. These include business & technology consulting, system integration, application development & maintenance, IT infrastructure services, analytics, business intelligence data , warehousing, customer relationship management, supply chain management, engineering & manufacturing solutions, enterprise resource planning, research and development, outsourcing, and testing solutions.

**Table of Contents**

[**1.0 Introduction 5**](#_Toc6323339)

[**1.1 Purpose of this Project 5**](#_Toc6323340)

[**1.2 Project Overview 5**](#_Toc6323341)

[**1.2.1 Objectives 5**](#_Toc6323342)

[**1.3 Intended Audience 6**](#_Toc6323343)

[**1.3.1 Business Case 6**](#_Toc6323344)

[**1.4 Technologies Recommended 6**](#_Toc6323345)

[**1.5 Hardware and Software Requirements 7**](#_Toc6323346)

[**1.6 Project Scope 7**](#_Toc6323347)

[**2.0 Process Architecture 8**](#_Toc6323348)

[**3.0 High Level Business Requirements 9**](#_Toc6323349)

[**3.1 Detailed Business Requirements 9**](#_Toc6323350)

[**3.1.1 Functional Requirements 9**](#_Toc6323351)

[**3.2 References 14**](#_Toc6323352)

[**3.2.1 Table 1.0 14**](#_Toc6323353)

[**3.2.2 Table 2.0 14**](#_Toc6323354)

[**3.2.3 Table 3.0 14**](#_Toc6323355)

[**3.2.4 Table 4.0 15**](#_Toc6323356)

[**3.2.5 Table 5.0 15**](#_Toc6323357)

[**3.2.6 Table 6.0 15**](#_Toc6323358)

[**3.2.7 Table 7.0 15**](#_Toc6323359)

[**3.2.8 Table 8.0 16**](#_Toc6323360)

[**3.3 Use case ‘User Registration’ 16**](#_Toc6323361)

[**3.3.1Use case attributes 16**](#_Toc6323362)

[**3.3.2 Business Rules 16**](#_Toc6323363)

[**3.3.3 UI Requirements 17**](#_Toc6323364)

[**3.3.4 UI Field Validations 18**](#_Toc6323365)

[**3.4 Use case ‘User Login’ 18**](#_Toc6323366)

[**3.4.1 Use case attributes 18**](#_Toc6323367)

[**3.4.2Business Rules 19**](#_Toc6323368)

[**3.4.3 UI Requirements 19**](#_Toc6323369)

[**3.5 Use case ‘Resource List’ 20**](#_Toc6323370)

[**3.5.1 Use case attributes 20**](#_Toc6323371)

[**3.6Use case ‘Document’ 21**](#_Toc6323372)

[**3.6.1Use case attributes 21**](#_Toc6323373)

[**3.6.2 Business Rules 22**](#_Toc6323374)

[**3.6.3 UI Requirements 22**](#_Toc6323375)

[**3.6.4 UI Field Validations 23**](#_Toc6323376)

[**3.7 Use case ‘Teacher’ 23**](#_Toc6323377)

[**3.7.1 Use case attributes 23**](#_Toc6323378)

[**3.7.2 Business Rules 24**](#_Toc6323379)

[**3.7.3 UI Requirements 24**](#_Toc6323380)

[**3.8 Use case ‘Student’ 25**](#_Toc6323381)

[**3.8.1 Use case attributes 25**](#_Toc6323382)

[**3.8.2 UI Requirements 26**](#_Toc6323383)

[**4.0 Database Design 27**](#_Toc6323384)

[**4.1 Tables Structure 27**](#_Toc6323385)

**Chapter 1**

# 1.0 Introduction

## 1.1 Purpose of this Project

This Project is aimed at:

* Providing the necessary inputs to the detailed requirements gathering phase and further on for the SDLC processes.
* This project also serves to establish the traceability between the Business Objectives and the requirements identified in the proposed solution and how they satisfy the stated objectives.
* This Project is about the teacher and student login in which there would be two user using the web.
* The purpose of this document is to systematically capture the requirements of the project and the system to be developed. The document also captures the Functional requirements and serves as an input for the scope of project.
* This project is about Teacher-Student Connect where the student and teacher would login to resource list and update or download the documents.

## 1.2 Project Overview

Creating an online Repository that contains a collection of learning and teaching materials flowing from Teachers and Students on a regular basis.

An Interactive UI platform that allows the Teachers and Students to digitally connect.

### 1.2.1 Objectives

Below are the objectives that shall be fulfilled post the execution of this project:

* Teachers and Students can view/engage in the courses from anywhere and anytime.
* Discovery, and retrieval of the learning materials from the repository along with other reference materials.
* Reuse of learning content.
* Cost effective and cost efficient solution of online knowledge base by hosting the repository on Cloud.
* Digital platform for the connect between Teachers and Students.

## 1.3 Intended Audience

* Interns/Project Team
* Mentors and SME’s

## 1.3.1 Business Case

* The effectiveness of learning through academic institutions greatly depend on the theoretical coverage offered by the facilitators, above and beyond the details provided in the reference materials prescribed in the curriculum. To quote the success of the institution greatly depends on the additional enablement opportunities that are presented to those consuming the education offered in these house of knowledge.
* The predominant objective is to enable the knowledge consumers with their day to day enablement needs and provide an online Repository that contains a collection of learning and teaching materials flowing from Teachers and Students on a regular basis.
* An Interactive UI platform that allows the Teachers and Students to digitally connect that are aligned with the curriculum prescribed which would result in a 'deep driven' knowledge being gained by the learners or knowledge seekers.

## 1.4 Technologies Recommended

|  |  |
| --- | --- |
| Front End | Java (HTML5, CSS3, JavaScript,) |
| Middleware | Java (Spring, Spring MVC, Hibernate MVC, WEB API2) |

## 1.5 Hardware and Software Requirements

|  |  |  |
| --- | --- | --- |
| Technology | Hardware | Software |
| Java | Desktop PC with 8GB RAM | 1. Node.js 10.15.1  2. Angular 5.0  3. Visual Studio Code 1.30  4. Eclipse IDE for Java EE Developers (Oxygen)  5. Maven 3.6.0  6. Tomcat 9  7. MySQL Community Server 8.0  8. MySQL Workbench 8.0.14  9. Putty  10. WinSCP 5.9.4  11. Oracle 11g express version |

## 1.6 Project Scope

Creating an online Repository that contains a collection of learning and teaching materials flowing from Teachers and Students on a regular basis.

An Interactive UI platform that allows the Teachers and Students to digitally connect.

**Chapter 2**

# 2.0 Process Architecture

Sign

-

In/

Registrati

on Page

Registration Page:

Procure user

information/

create

new user

and

allow

Sign

-

In

Resource List

Selection

Display

the table with

the

selected details

available in the

Repository

Submit New Resource

Page:

Allow user to upload

new documents

Archive Documents

Move documents to

Archive folder

Log Out

Docu

ments

Stude

nts

Teach

ers

**Chapter 3**

# 3.0 High Level Business Requirements

Note - Primary focus is to complete developing the critical requirements (highlighted in Yellow) and then to proceed with the remaining requirements.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **S.No.** | | **Busine**  **Requirements**  **ID** | |  | **Short Description** | | | | | | | |  | **Description in detail** | | | | | | | | | | | | | | **Interacting**  **Business**  **Processes** |
| 1 |  |  | REQ-1 |  |  | | | | New Use | r | | |  |  | | | | Ability of the portal to | | | | | | |  | | |  |
|  |  | | | Registration | | |  | |  | | | procure user data and | | | | | | | |
|  | | | create new user profiles | | | | | | | | |  | |
| 2 | |  | REQ-2 |  |  | | | | User login |  | | |  |  | | | | Ability of the portal to | | | | | | |  | | |  |
|  |  |  | | authenticate and authorize | | | | | | | | | | |  |
|  | | | the registered user | | | | |  | | |
| 3 |  |  | REQ-3 |  |  | | Resource List | | | | |  |  |  | Displays the complete list o | | | | | | | | | | | | f  e |  |
|  |  | | | | | selected resources availabl | | | | | | | | | | | |
|  | | | | | in the repository | | |  | | | |
| 4 |  |  | REQ-4 |  |  | Advanced Search | | | | | | |  |  | Allows the user to search fo | | | | | | | | | | | | r |  |
|  |  | | | | | | |  | | | | | | specific |  | | | | |
| Documents/Teacher/Student | | | | | | | | | | | |
| 5 | |  | REQ-5 |  |  | | | Submit new | | |  | |  |  | | | Allows the Teachers o | | | | | | | | r | | |  |
|  |  | resource |  | Students to upload new | | | | | | | | |  | |
|  |  | documents to the repository | | | | | | | | | | | |  |
| 6 | | REQ-6 | |  | Archive Document | | | | | | | | s | Allows the Teachers to move the documents to archive folder | | | | | | | | | | | | | |  |
| 7 | | REQ-7 | |  | Help | | | | | | | |  | Page to assist the users to report technical issues | | | | | | | | | | | | | |  |

# 

# 3.1 Detailed Business Requirements

## 3.1.1 Functional Requirements

The functional requirements are projected below, for each of the high level requirements called out in the earlier section

|  |  |  |
| --- | --- | --- |
| **Value** | **Rating** | **Description** |
| 1 | Critical | This requirement is critical to the success of the project. The project will not be possible without this requirement. |
| 2 | High | This requirement is high priority, but the project can be implemented at a bare minimum without this requirement. |
| 3 | Medium | This requirement is somewhat important, as it provides some value but the project can proceed without it. |
| 4 | Low | This is a low priority requirement, or a “nice to have” feature, if time and cost allow it. |
| 5 | Future | This requirement is out of scope for this project, and has been included here for a possible future release. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Req. #** | **Rationale Categorization** |  | **Business Requirement** | **Req**  **.**  **Typ e \*** | **Priority \*\*** | **Originator**  **\*\*\*** | **BR Traced to Business**  **Requirement / Use case ID** | **Remarks** |
| 1.01 | New User  Registration |  | Clicking on the “Register” button should open the  Registration form. | UI | Critical | NA | REQ-1 |  |
| 1.02 | New User  Registration |  | User to fill the basic attributes:  First Name  Last Name  Age  Gender  Contact Number  Category  User Id  Password Picture | UI | Critical | NA | REQ-1 | Please refer  Table 1.0 under References section |
| 1.03 | New User  Registration |  | Clicking ‘Submit’ button should  validate the datatype  constraints for each  field | F | Critical | NA | REQ-1 |  |
| 1.04 | New User  Registration |  | User failing to provide information for the mandatory  fields be presented with an alert  message – ‘Please update the  highlighted  mandatory field(s).’ Also, highlight the missed out field in  red | E | Medium | NA | REQ-1 |  |
| 1.05 | New User  Registration |  | Post-successful field validation, save the  information in the database | F | Critical | NA | REQ-1 |  |
| 1.06 | New User  Registration |  | Upon saving the information in the  database, display  the message “New user created successfully” | E | Medium | NA | REQ-1 |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Req. #** | **Rationale Categorization** | **Business Requirement** | **Req**  **.**  **Typ e \*** | **Priority \*\*** | **Originator**  **\*\*\*** | **BR Traced to Business**  **Requirement / Use case ID** | **Remarks** |
| 2.01 | User login | Clicking “SignIn” link should allow the  user to enter the User ID and  Password and click  “Login” button | F | Critical | NA | REQ-2 |  |
| 2.02 | User login | Portal to validate the entered user  credentials against  the database and allow login for registered users. | F | Critical | NA | REQ-2 |  |
| 3.01 | Resource List | Clickling on  “Resource List” link should display the  below choices:  Document  Teacher  Student | F | Critical | NA | REQ-3 |  |
| 3.02 | Resource List | Portal to allow the user to select the  required choice from the above | UI | Critical | NA | REQ-3 |  |
| 3.03 | Resource List | Selecting  “Document” should display table format of the entire list of  documents that are  stored in the the repository | F | Critical | NA | REQ-3 | Please refer  Table 2.0 under References section |
| 3.04 | Resource List | Allow user to choose sorting order: Sort by Year  (Newest First)  Sort by Year (Oldest First)  Sort by Title (A to Z)  Sort by Title (Z to A)  Sort by Author (A to  Z)  Sort by Author (Z to A)  Sort by Subject (A to  Z)  Sort by Subject (Z to  A) | UI | Critical | NA | REQ-3 |  |
| 3.05 | Resource List | Selecting a particular document from the resultant  table should allow  the user to preview  the document in a new tab | F | Critical | NA | REQ-3 |  |
| 3.06 | Resource List | Selecting “Teacher” should display table format of the  complete list of  Teachers, registered in the the portal | F | Critical | NA | REQ-3 | Please refer  Table 3.0 under References section |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Req. #** | **Rationale Categorization** | **Business Requirement** | **Req**  **.**  **Typ e \*** | **Priority \*\*** | **Originator**  **\*\*\*** | **BR Traced to Business**  **Requirement / Use case ID** | **Remarks** |
| 3.07 | Resource List | Allow user to choose sorting order:  Sort by Name (A to  Z)  Sort by Name (Z to  A)  Sort by  Specialization (A to Z)  Sort by  Specialization (Z to  A) | UI | Critical | NA | REQ-3 |  |
| 3.08 | Resource List | Clicking on a particular Teacher  details from the resultant table  should allow the  user to preview the details in a new tab | F | Critical | NA | REQ-3 |  |
| 3.09 | Resource List | Clicking on the  “Assigned classes, schedule” hyperlink  should open a new tab with the  schedule in table format | F | Critical | NA | REQ-3 | Please refer  Table 4.0 under References section |
| 3.10 | Resource List | Selecting “Student” should display table format of the  complete list of  Students registered in the the portal | F | Critical | NA | REQ-3 | Please refer  Table 5.0 under References section |
| 3.11 | Resource List | Allow user to choose sorting order:  Sort by Name (A to  Z)  Sort by Name (Z to  A)  Sort by Batch (A to  Z)  Sort by Batch (Z to  A) | UI | Critical | NA | REQ-3 |  |
| 3.12 | Resource List | Selecting a particular Student  details from the resultant table  should allow the  user to preview the details in a new tab | F | Critical | NA | REQ-3 |  |
| 4.01 | Advanced Search | Clicking “Advanced Search” should display a form with below fields:  Keyword Search  Title  Author  Subject Year | UI | Critical | NA | REQ-4 | The search criteria should have logical operators like  “AND” / “OR”    Please refer  Table 5.0 under References section |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Req. #** | **Rationale Categorization** | **Business Requirement** | **Req**  **.**  **Typ e \*** | **Priority \*\*** | **Originator**  **\*\*\*** | **BR Traced to Business**  **Requirement / Use case ID** | **Remarks** |
| 4.02 | Advanced Search | The display form should have buttons  to “Submit Query” or  “Reset” the fields | UI | Critical | NA | REQ-4 |  |
| 4.03 | Advanced Search | Clicking on “Submit Query” button should display the  required documents and allow user to download the file | UI | Critical | NA | REQ-4 |  |
| 5.01 | Submit new Resource | Clicking “Submit new Resource”  should display a form with below fields:  Subject/Topic  Title  Author  Subject  Year  Upload/Attach file | UI | Critical | NA | REQ-5 | Please refer  Table 6.0 under References section |
| 5.02 | Submit new Resource | Clicking on “Submit” button should  perform backend  validation of the user  profile and upload the document | F | Critical | NA | REQ-5 |  |
| 6.01 | Archive  Document | Portal capability to allow Teachers to archive documents | F | Medium | NA | REQ-6 | Access to Archive documents should only be provided for Teachers. |
| 7.01 | Help | Page that displays the basic details text about how to search documents | F | Low | NA | REQ-6 |  |
| 7.02 | Help | Portal to display a form that allows user to report the  technical issues  through Email:  Issue  Description  Send Button | F | Low | NA | REQ-6 | Please refer  Table 7.0 under References section |
| 7.03 | Help | Portal to display the  Contact number to report issues | F | Low | NA | REQ-6 |  |

# 3.2 References

## 3.2.1 Table 1.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field**  **Name** | **Field Type** | **Data Type** | **Mandatory** | **Possible Values** |
| First Name | Text(50) | Alphabetic | Yes |  |
| Last Name | Text(50) | Alphabetic | Yes |  |
| Age | Numeric(2) | Numeric | Yes |  |
| Gender | Drop Down | NA |  | Male, Female |
| Contact Number | Text(10) | Numeric | No |  |
| Category | Drop Down | NA | Yes | Teacher/Student |
| UserId | Text(15) | Alphanumeric | Yes | UserId |
| Password | Text(15) | Alphanumeric | Yes | Password |
| Picture | Button | gif, jpeg, png | Yes |  |

## 3.2.2 Table 2.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field**  **Name** | **Field Type** | **Data Type** | **Mandato ry** | **Possible Values** |
| Image of the Cover page | Image | gif, jpeg, png | Yes |  |
| Title | Text(100) | AlphaNumeric | Yes |  |
| Author | Text(50) | Alphabetic | Yes |  |
| Subject | Text(50) | Alphabetic | Yes |  |
| Year | Numeric(4) | Numeric | Yes |  |
| File | Document | Word,Excel,PDF, etc | Yes |  |

## 3.2.3 Table 3.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Data Type** | **Mandatory** | **Possible Values** |
| Image of the Teacher | Image | gif, jpeg, png | Yes |  |
| Name | Text(100) | Alphabetic | Yes |  |
| Specialization | Text(100) | Alphabetic | Yes |  |
| Assigned classes, schedule | Hyperlink | AlphaNumeric | Yes |  |
| Number of students | Numeric(4) | Numeric | Yes |  |
| Planned Leaves | AlphaNum eric(4) | AlphaNumeric | No |  |

## 3.2.4 Table 4.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Data Type** | **Mandatory** | **Possible Values** |
| Date | AlphaNum eric(10) | AlphaNumeric | Yes |  |
| Time | AlphaNum eric(10) | AlphaNumeric | Yes |  |
| Batch | Text(10) | Numeric | Yes |  |
| Subject | Text(10) | Alphabetic | Yes |  |
| Notes | Hyperlink | Alphabetic | Yes |  |

## 3.2.5 Table 5.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field**  **Name** | **Field Type** | **Data Type** | **Mandatory** | **Possible Values** |
| Image of the  Student | Image | gif, jpeg, png | Yes |  |
| Name | Text(100) | Alphabetic | Yes |  |
| Specializat ion | Text(100) | Alphabetic | Yes |  |
| Batch | Numeric(8) | Numeric | Yes |  |
| No. of documents uploaded | Numeric(4) | Hyperlink | Yes |  |

## 3.2.6 Table 6.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field**  **Name** | **Field Type** | **Data Type** | **Mandatory** | **Possible Values** |
| Keyword Search | Text(50) | Alphabetic | No |  |
| Title | Text(50) | Alphabetic | No |  |
| Author | Drop Down | NA | No |  |
| Subject | Drop Down | NA |  | Subject Names |
| Year | Drop Down | Numeric | No | 2000 to 2019 |

## 3.2.7 Table 7.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Data Type** | **Mandatory** | **Possible Values** |
| Subject/Topic | Text(50) | Alphabetic | Yes |  |
| Title | Text(50) | Alphabetic | Yes |  |
| Author | Drop Down | NA | Yes |  |
| Subject | Drop Down | NA | Yes | Subject Names |
| Year | Drop Down | Numeric | Yes | 2000 to 2019 |

## 3.2.8 Table 8.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Data Type** | **Mandatory** | **Possible Values** |
| Issue | Text(100) | Alphabetic | Yes |  |
| Description | Text(300) | Alphabetic | Yes |  |

## **3.3 Use case ‘User Registration’**

### 3.3.1Use case attributes

**Use Case Description:**

This use case deals with the capture of user details. The ‘user’ here shall be the operator of the system and will be keying in the user details.

**Scope:**

* User registration

**Actors:**

* User – the operator

**Trigger:**

Click ‘Registeration’ button in the home page

**Pre-Condition:**

User being able to access the homepage & get redirected to the ‘ new User Registration form’ page upon click of ‘Registeration’ link on the login page.

**Post Condition:**

User is in the registration page & submit details

**Flow of Events:**

User at homepage 🡪 Click ‘Registration’ link 🡪 User is in the registration page 🡪 User details are submitted and added onto the database

**Primary Scenario:**

A new user – is able to click ‘Registeration ’ link and able to provide his details and get registered in the system.

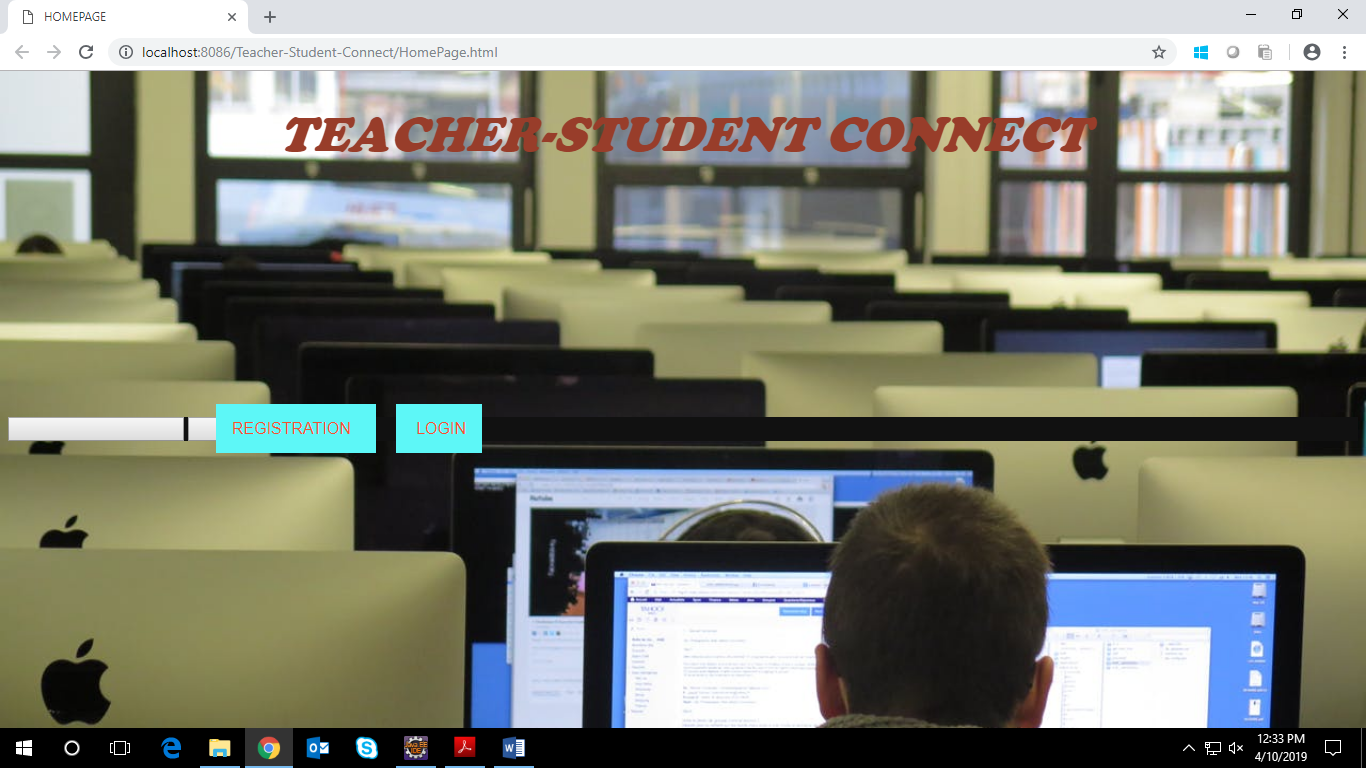
### 3.3.2 Business Rules

Business rules should be defined using the following attributes: -

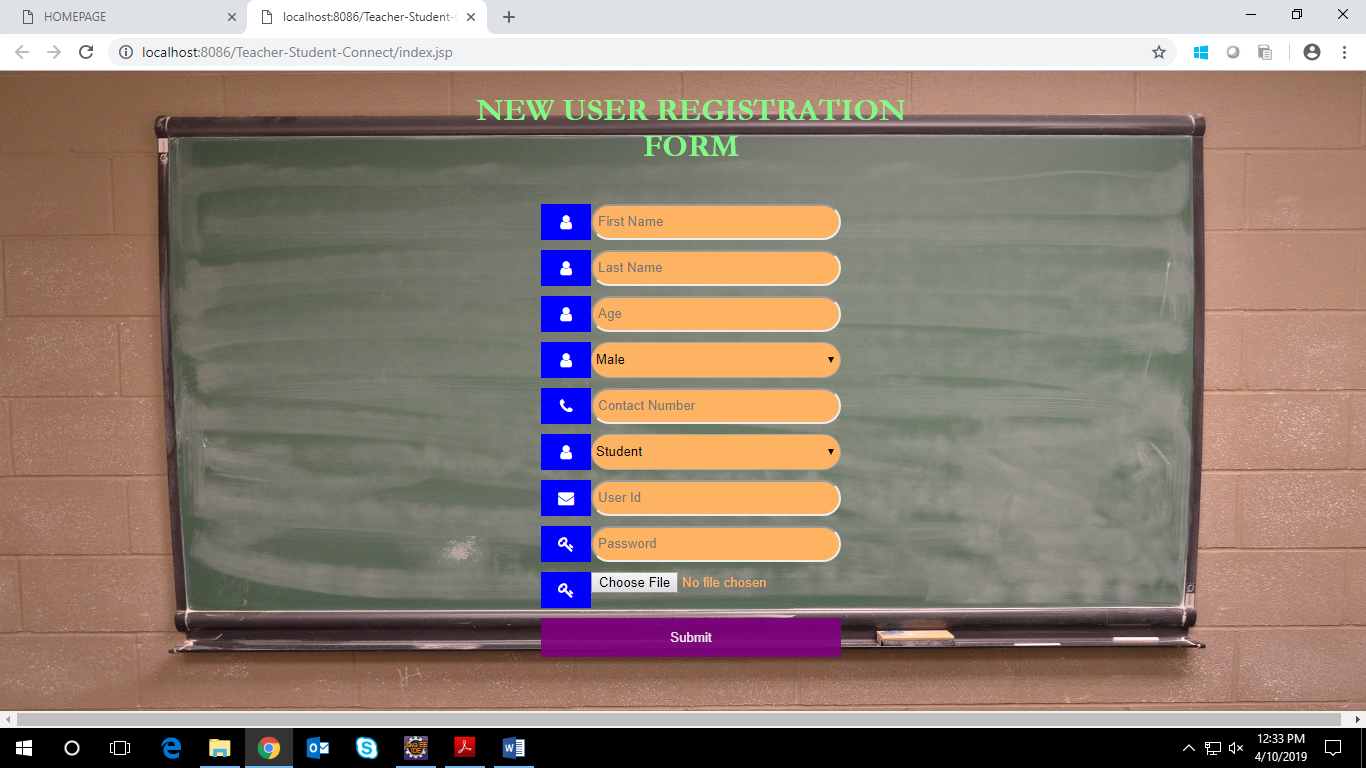
* When the user clicks on the register link, it should re-direct to registration form.
  + - User needs to fill some of the basic attributes/fields as mentioned below in requirement: First Name, Last Name, Age, Gender, Contact Number, Category, userID, Password, Picture.
* Clicking ‘submit’ should validate the datatype constraints for each field
* Post-successful field level validation, save the information in the database
* Upon saving the information in the database, display the message ‘New User Created successfully ’.

### 3.3.3 UI Requirements

Here is a prototype on how the homepage should look like.



Here is a prototype on how the ‘New User Registration Form’ page should look like.



### 3.3.4 UI Field Validations

Please refer to the below requirements for field level validations:

* All fields are mandatory.

Password should have maximum 15 alphanumeric, space & can contain special characters (ex.!,@,#,%,\*,& etc.)

* Contact Number must be of 10 digits’ length.
* Age should be 2 digit numeric value.
* First Name, Last Name should contain only alphabets.
* Gender should be chosen from the dropdown options.

## **3.4 Use case ‘User Login’**

### 3.4.1 Use case attributes

**Use Case Description:**

This use case deals with the authentication of the user credentials. The ‘user’ here shall be the operator of the system and will be keying in the user information into the system.

**Scope:**

* User credentials authentication

**Actors:**

User – the operator

**Trigger:**

Click ‘Login’ link, after keying in ‘UserID’ & ‘Password’ field.

**Pre-Condition:**

User being able to access the homepage

**Post Condition:**

User is in the Resource List page.

**Flow of Events:**

User at homepage 🡪 Click ‘Login’ link 🡪 User is in the login page 🡪 User will be directed to resource list page.

**Primary Scenario:**

A registered user – is able click ‘Login’ link, after keying in ‘UserID’ & ‘Password’ field and get his credentials authenticated with the existing database entry.

**Alternative Scenario:**

A registered user – is able click ‘Login’ link, after keying in ‘UserID’ & ‘Password’ field and unable to get his credentials authenticated. The user is presented with relevant error messages: Invalid username or password.

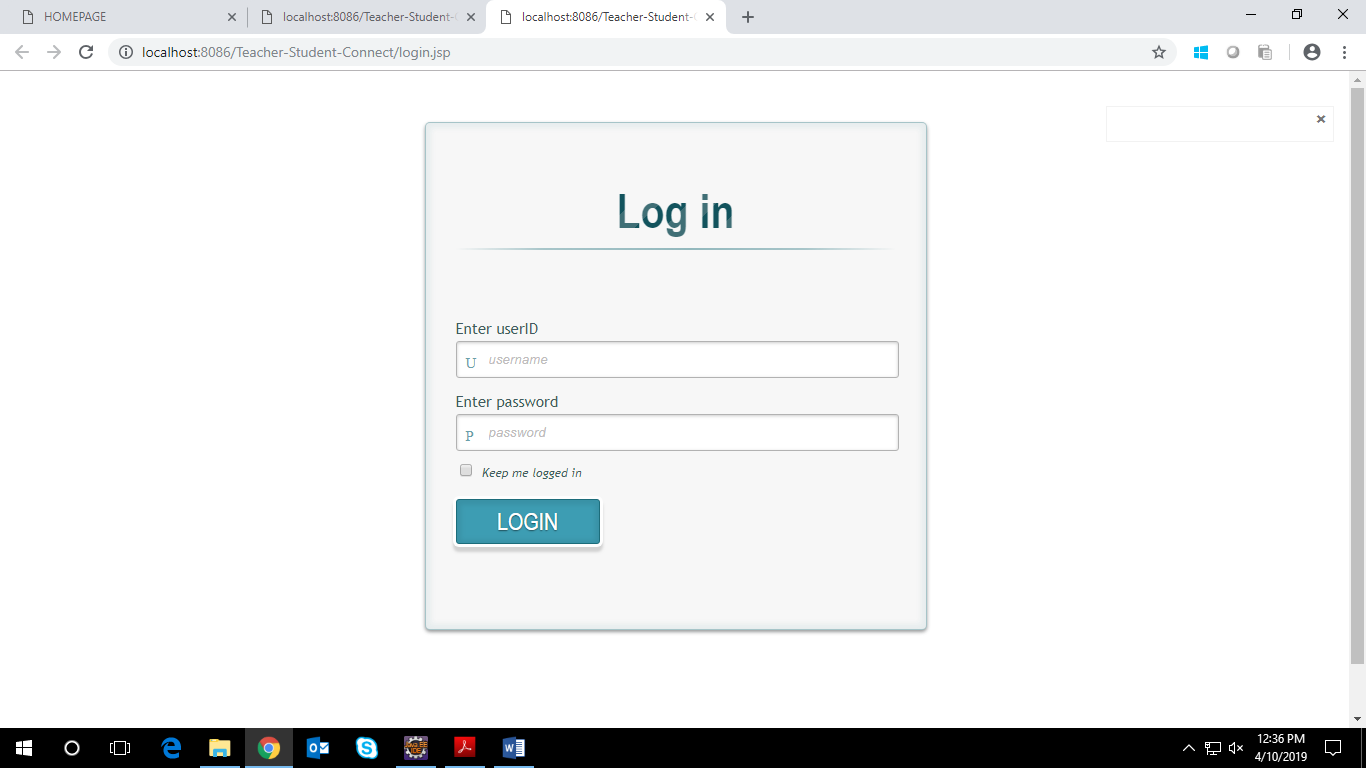
### 3.4.2Business Rules

Business rules should be defined using the following attributes:-

* A registered user – is able click ‘Login’ link, after keying in ‘UserID’’ & ‘Password’ field and get his credentials authenticated with the existing database entry.

### 3.4.3 UI Requirements

Here is a prototype on how the Billing System Login should look like.



## **3.5 Use case ‘Resource List’**

### 3.5.1 Use case attributes

**Use Case Description:**

This use case deals with the authentication of the Resource list.The ‘user’ here shall be the operator of the system and will be keying in the resource list information.

**Scope:**

* Resource List

**Actors:**

* User – the operator

**Trigger:**

user should get redirected to the ‘resource list’ page upon clicking on the ‘Registration or login’ link.

**Pre-Condition:**

User being able to access the login page

**Post Condition:**

Allow access to resource list for registered users.

**Flow of Events:**

User at login page 🡪 resource list page is displayed -> Key in ‘teacher’ ,’student’,’document’

**Primary Scenario:**

A registered user – is able click ‘Login’ link, after keying in ‘UserID’ & ‘Password’ field and get his credentials authenticated with the existing database entry.and after successful login he will be able to access the resource list which includes teachers,students,documents



## **3.6Use case ‘Document’**

### 3.6.1Use case attributes

**Use Case Description:**

This use case deals with the documents that are uploaded on the database.the suser after login can download the documents.

**Scope:**

* Documents

**Actors:**

* User – the operator

**Trigger:**

Click ‘document’ and preview any given file.

**Pre-Condition:**

After clicking on the ’Login ’ button in the login page user will be directed to resource page.it contains fields like:-teacher,student,document. And other fields like AboutUs,Help,logout.

**Post Condition:**

User is displayed a document page which contains a table of documents that are uploaded on the documents table

**Flow of Events:**

User at homepage 🡪 Click ‘login’ link 🡪 User at loginpage 🡪 Click ‘Login’ button ->User at Resource List page 🡪 Click the button corresponding to any given resource list🡪 user selected the document link. 🡪user will move to documents page -> user can view a table of documents or upload any document

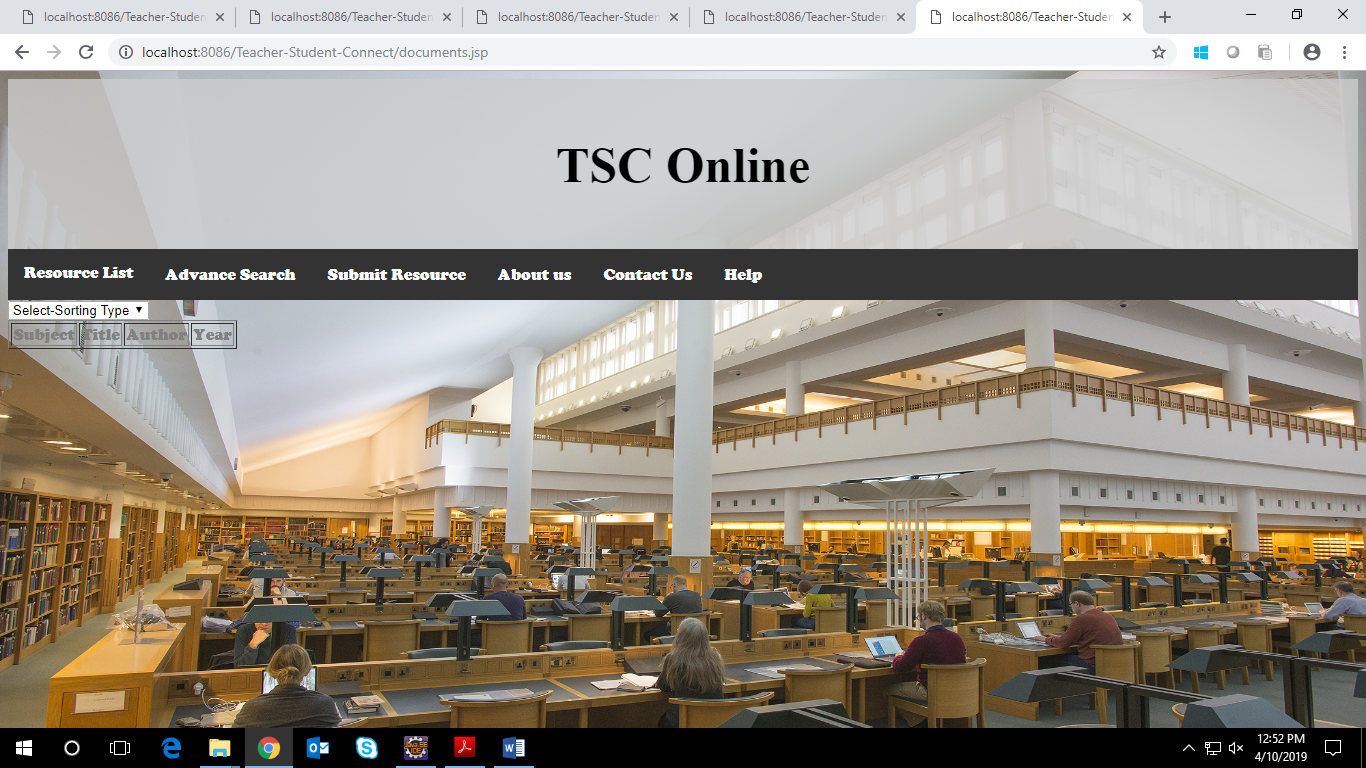
### 3.6.2 Business Rules

Business rules should be defined using the following attributes: -

* When the user clicks on any given Bill type say Electricity button, it is redirected to the ‘Pending Bills’ page where the user is provided an option to add new bill of previously selected type.
  + - User needs to fill some of the basic attributes/fields as mentioned below in bill form requirement: Username, PAN card No., Amount to be paid, Card Holder Name, CVV Number, Credit Card Number;
* Clicking ‘Save and Exit’ should validate the datatype constraints for each field
* Post-successful field level validation, save the information in the database
* Upon saving the information in the database, display the message ‘Bill Form Data Saved Successfully’.

### 3.6.3 UI Requirements

Here is a prototype on how the ‘Document” page should look like.



### 3.6.4 UI Field Validations

Please refer to the below requirements for field level validations:

* All fields are mandatory.
* PAN Number must be of 15 digits’ length
* Amount to be paid should contain only numeric digits.
* Credit Card Holder name should consist of alphabets only.
* Credit Card CVV field should contain only numerical digits of length 3.
* Credit card Number should be of exactly 12 digits.

## **3.7 Use case ‘Teacher’**

### 3.7.1 Use case attributes

**Use Case Description:**

This use case deals with the ability of the system to provide information about Teachers. The ‘user’ here shall be the operator of the system and will be keying in the resource list.

**Scope: Teacher**

**Actors:**

* User – the operator

**Trigger:**

Click any given teacher and details of the teacher will be displayed

**Pre-Condition:**

After clicking on the ’Login ’ button in the login page user will be directed to resource page.it contains fields like:-teacher,student,document. And other fields like AboutUs,Help,logout.

**Post Condition:**

User is displayed a Teacher page which contains a list of all the teachers

**Flow of Events:**

User at homepage 🡪 Click ‘login’ link 🡪 User at loginpage 🡪 Click ‘Login’ button ->User at Resource List page 🡪 Click the button corresponding to any given resource list like teacher🡪 user selected the particular teacher. 🡪user will move to details of selected teacher profile 🡪user can sort the teachers on basis of name and qualification etc.

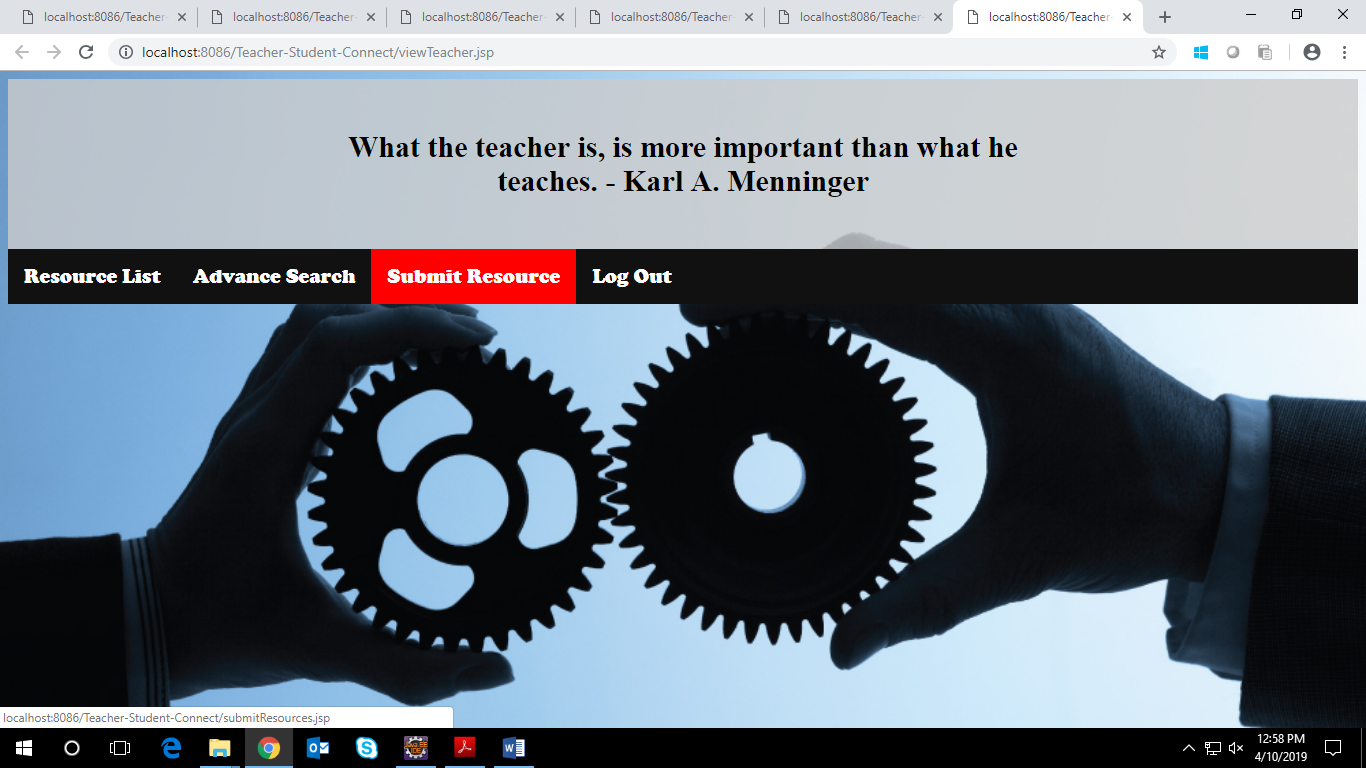
### 3.7.2 Business Rules

Business rules should be defined using the following attributes: -

* When the user clicks on the any given bill type button, it should re-direct to the pending Bill Form.
* User can view the details of any bill having pending status and is provided an option to modify or delete any of the pending bills listed against him.
* Clicking ‘delete’ should delete that particular bill from the pending list against the current user.
* Clicking ‘view’ against any pending bill should display the details of the corresponding pending bill against the user and should provide the user a link to go for Bill Payment.

### 3.7.3 UI Requirements

Here is a prototype of Teacher page



## **3.8 Use case ‘Student’**

### 3.8.1 Use case attributes

**Use Case Description:**

This use case deals with the ability of the system to provide information about Teachers. The ‘user’ here shall be the operator of the system and will be keying in the resource list.

**Scope: Student**

**Actors:**

* User – the operator

**Trigger:**

Click any given Student and details of the student will be displayed

**Pre-Condition:**

After clicking on the ’Login ’ button in the login page user will be directed to resource page.it contains fields like:-teacher,student,document. And other fields like AboutUs,Help,logout.

**Post Condition:**

User is displayed a Student page which contains a list of all the Students

**Flow of Events:**

User at homepage 🡪 Click ‘login’ link 🡪 User at loginpage 🡪 Click ‘Login’ button ->User at Resource List page 🡪 Click the button corresponding to any given resource list like Student🡪 user selected the particular Student. 🡪user will move to details of selected student profile 🡪user can sort the students on basis of name etc.

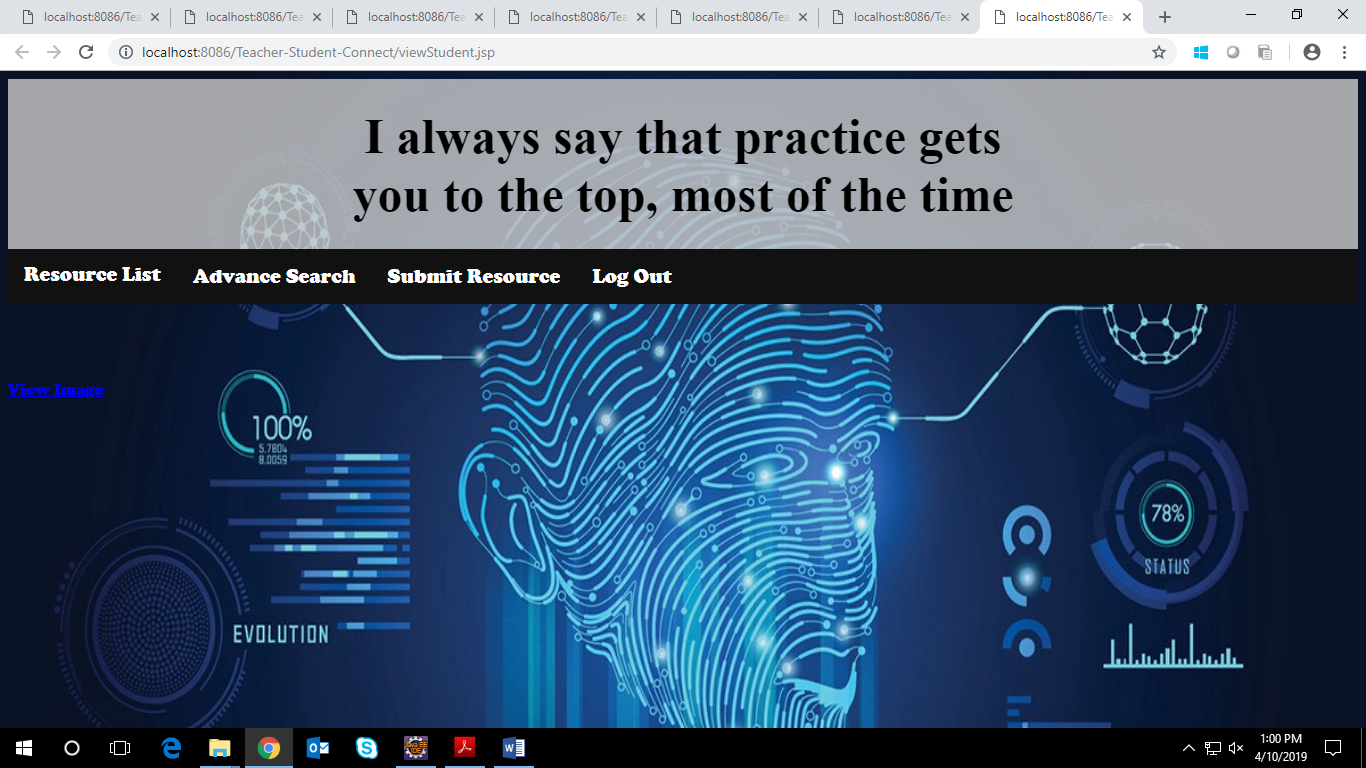
**Business Rules :**

Business rules should be defined using the following attributes: -

* + - User needs to click on any of the bill types and should be able to see all the bills pending against him of the selected bill type.
* Upon clicking the view button , user should be redirected to bill form page where user can choose the payment vendor and select on pay bill button to pay the bill.
* Post-successful payment ,the corresponding pending entry is deleted from the table.
* Thereafter we redirect User to ’Record Your Bill’ page.

### 3.8.2 UI Requirements

Here is a prototype of Student Page



# 4.0 Database Design

## **4.1 Tables Structure**

**4.1.1 New User Registration:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| firstName | varchar | 50 | N |
| lastName | varchar | 50 | N |
| age | int | 2 | N |
| gender | varchar | 6 | N |
| Contact number | varchar | 10 | N |
| category | varchar | 15 | N |
| UserID | varchar | 10 | N |
| password | varchar | 15 | N |
| Picture | varchar | 20 | N |

**4.1.2 User Login:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| User Id | varchar | 50 | N |
| password | varchar | 15 | N |

**4.1.3 Resource List:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| document | varchar | 50 | N |
| Teacher | varchar | 50 | N |
| Student | varchar | 50 | N |

**4.1.4 Teacher:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| Image of the Teacher | blob | 50 | N |
| Name | varchar | 10 | N |
| Specialization | varchar | 15 | N |
| Assigned classes, schedule | varchar | 50 | N |
| Number of students | int | 10 | N |

**4.1.5 Student:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| Image of the Student | blob | 50 | N |
| Name | varchar | 10 | N |
| Specialization | varchar | 15 | N |
| Batch | varchar | 50 | N |
| Number of documents uploaded | int | 10 | N |

**4.1.6 Document:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| Image of the cover page | blob | 50 | N |
| title | varchar | 10 | N |
| Author | varchar | 15 | N |
| subject | varchar | 50 | N |
| year | int | 10 | N |
| File | Word,excel | 20 | N |