**Final Project Report**

**Software System Design and Implementation**

ITCS - 6112

Spring 2017

**University of Charlotte Application System**

Team Members:

Susritha Baditha 800876039

Sharath Chandrika Mummoju 800965107

Puneeth Devabhaktuni 800953156

Aditya Viswanadha 800959537

Raghava Adarsh Mandarapu 800937296

Likhith Chinnam 800986749

Virinchi Ande 800970447

# **Table of Contents**

Abstract 3

Introduction 3

Technical Documentation 4

Sample runs and results 7

Conclusion 13

**ABSTRACT**

This is a university admission portal system where students can browse the site to know about the university, sign up for filling an admission form, sing in to the application portal, fill out the application form providing all the details.

The admin can login with his credentials and can see the students who applied to the university and can accept or reject students based on his profile. The number of seats left in each of the department is also shown as summary to admin. Based on the admin action (i.e. accepts or reject) an email will be sent to the respective user to let him know the decision taken. The user can also send an email to the support team using contact us page regarding their problems with the site or any other questions.

**INTRODUCTION**

In our project, there will be user and admin. Both of them will have different functionalities.

**USER:**

**User overview:** The users for this university application portal system are prospective students who want to apply to the universities for graduation in Computer Science, DSBA, and Information Technology.

**Actions of the users:**

1. User can access the home page upon launching the site where he can see the overview of the university.
2. Upon clicking on the about us page user can see more information about the university and its policies.
3. Upon clicking on the Contact us page user can send an email to the support team. The email can be anything regarding the problems with logging in or any specific information about the university.
4. Upon clicking on the login/signup page the user can log in to the site. If he is a new user he needs to sign up to the system by clicking in the signup link.
5. The user needs to provide email id, name and password and gender in order to signup for the university portal. (Passwords are also hashed in the backend database table to provide more security).
6. Upon successfully logging in to the system the user page will be shown where he can click on the application from to fill up.
7. In the application form the user can enter all his details like name, scores, department he is applying to, address etc. and submit the application form.

**ADMIN**

**Admin overview:** The admin for this system is part of admission department of the university who can see the profiles of the students who have applied and give them accepts or rejects based on the students profile and the number of seats left in each department.

**Actions of the Admin:**

1. Upon clicking on the login page the admin can login to the site with his username and password.
2. Upon logging in to the system the admin page will be displayed where he can click on a button to see the students who have applied to the university and their profiles in a tabular format.
3. The admin can click on the accept or reject link depending on the student profile and depending on the number of seats left in the specific department to which the student has applied(which is displayed in the table).
4. The number of seats left in each department will be displayed under the students table which will be updated after each accepts the admin gave to the students.

**METHODS USED AND TECHNICAL DOCUMENTATION**

We used incremental model for our application.

* Produce a rough initial version first with only some features
* Get feedback
* Refine and improve software; Each increment provides part of functionality
* + Rapid development and delivery of useful software to the customer
* + Strives to overcome Waterfall and Iterative models
* + Can accommodate changes

**Software Requirements:**

Programming Language: Java

Scripting Language: Java Script

We used JSP, CSS, Bootstrap to design the pages, Servlets for controllers, JDBC connection to connect to the Database.

**Tools and Environments:**

Tools used: JSP, CSS, Bootstrap, Servlets, Java Scripts, Java.

Environment used: NetBeans and MySQL Workbench.

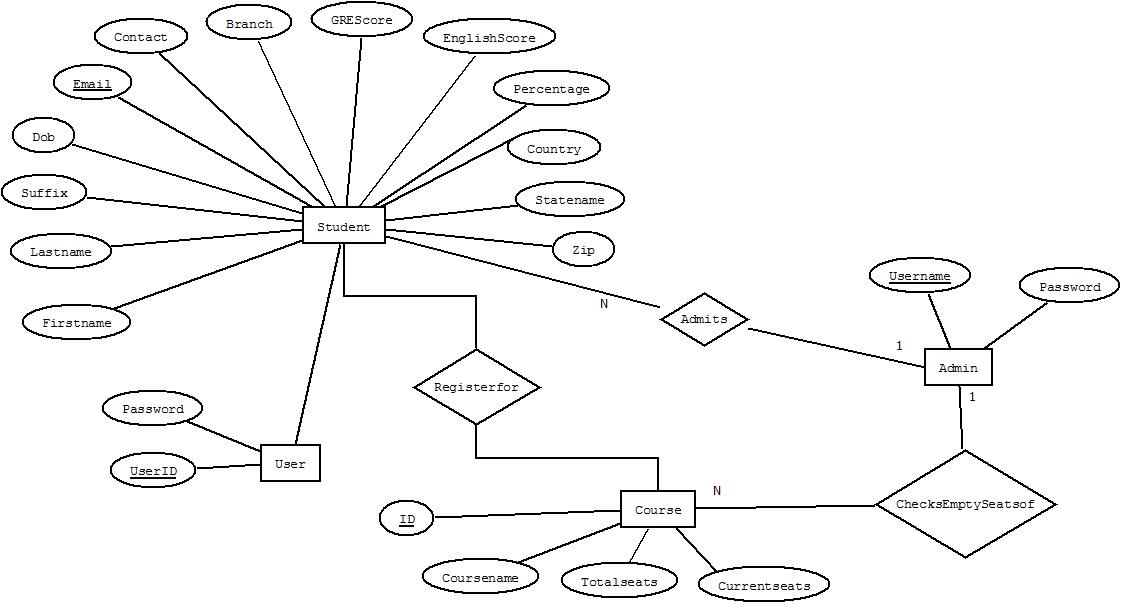
**Database System:**

There is a total of 4 tables in the our ‘College’ Database.

Tables:

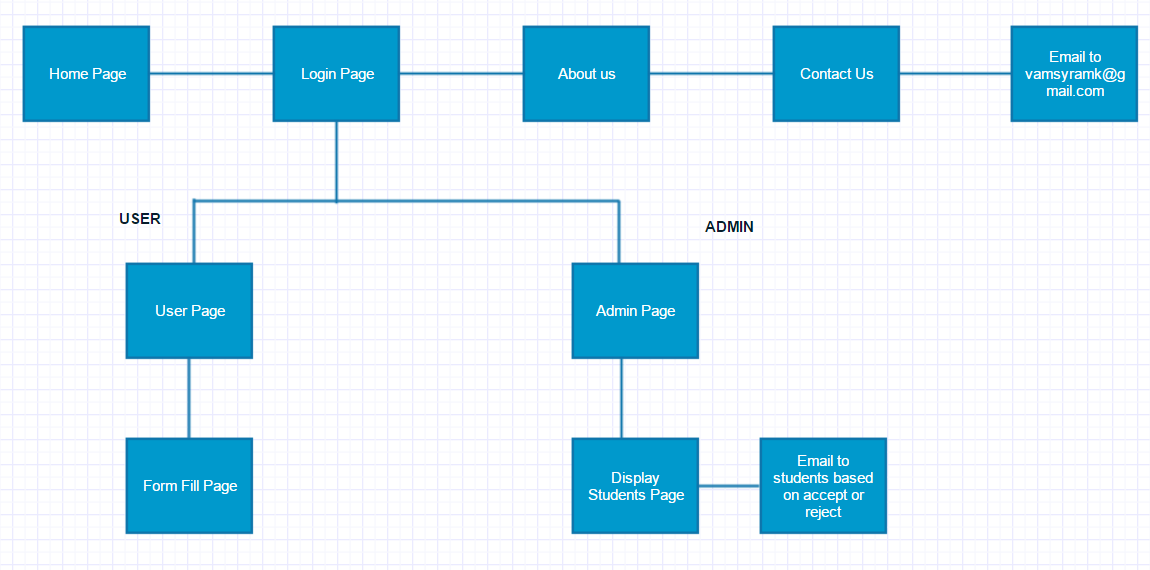
1. Admin
2. Course
3. Admin
4. User

**ER Diagram:**

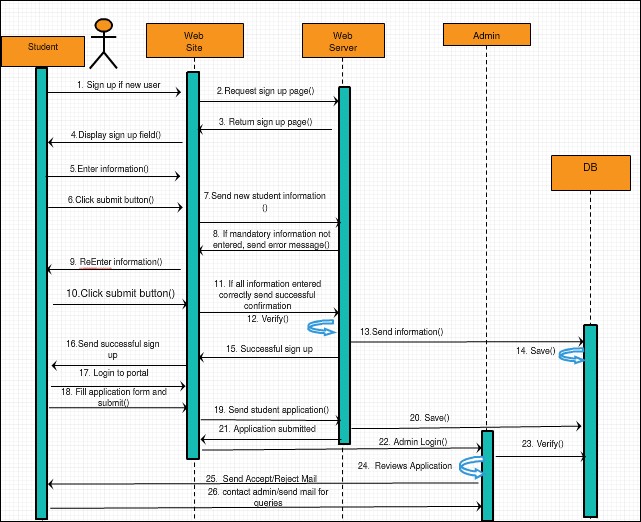


**SAMPLE RUNS AND RESULTS**

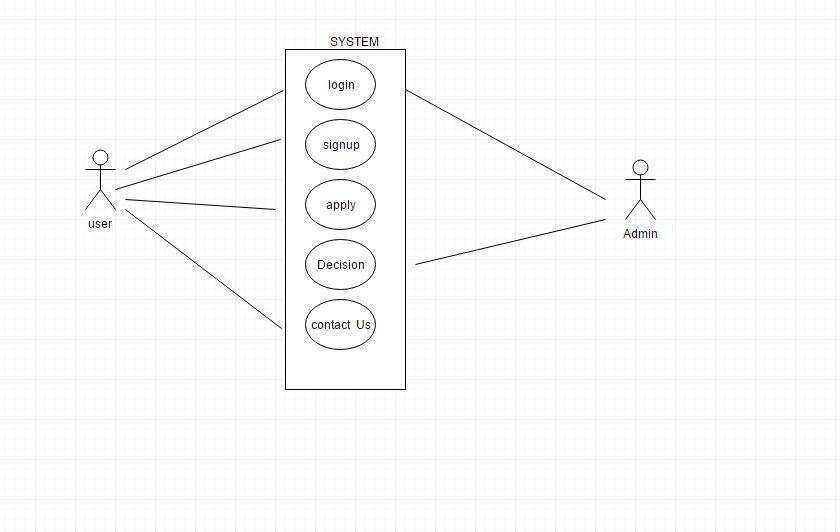
**Site Map:**

****

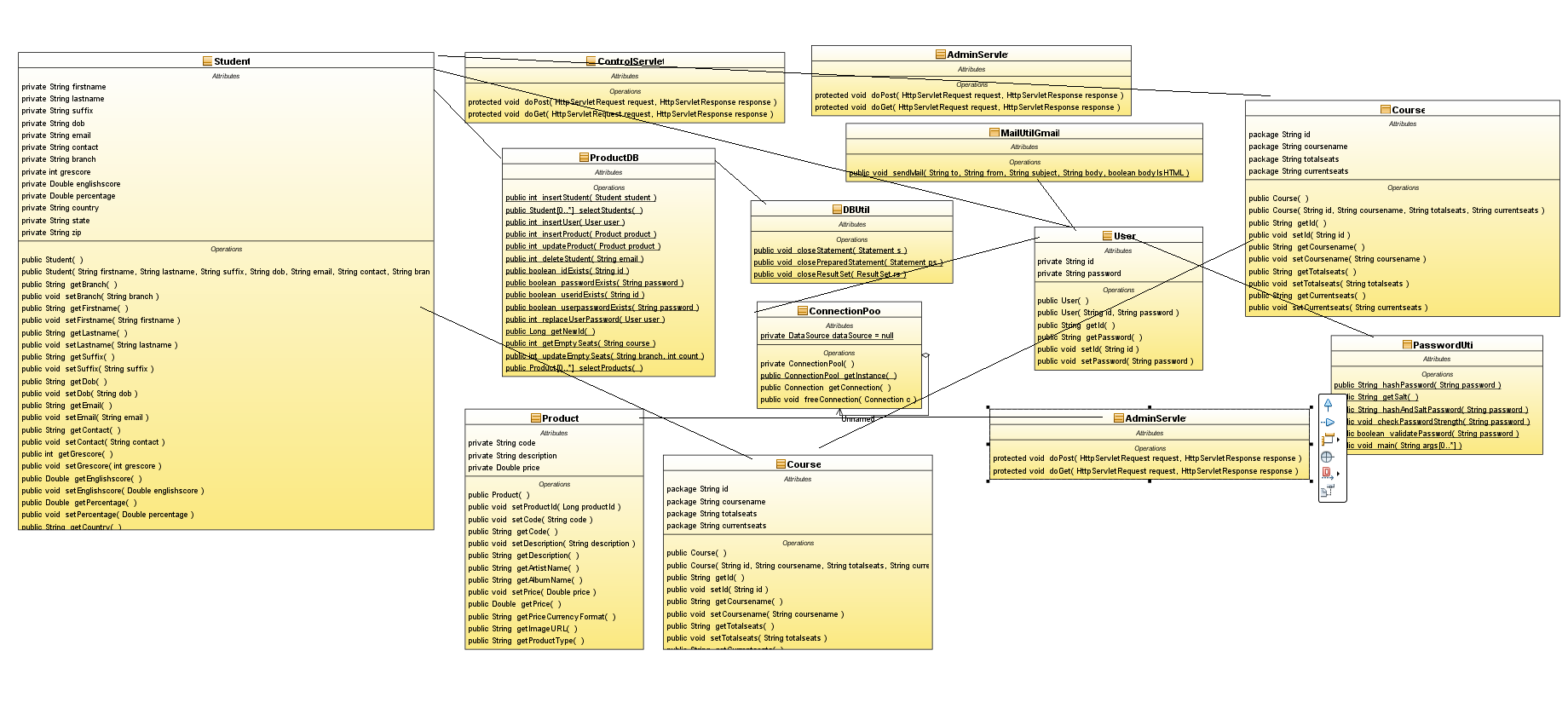
**Sequence Diagram:**

****

**Use Case Diagram:**



**Class Diagram:**



**Users and Access:**

**Admin:** can give admits or rejects to the students.

**User:** Student can fill the application form and it will be submitted to the admin.

**Page Design:**

**1. Home Page:**

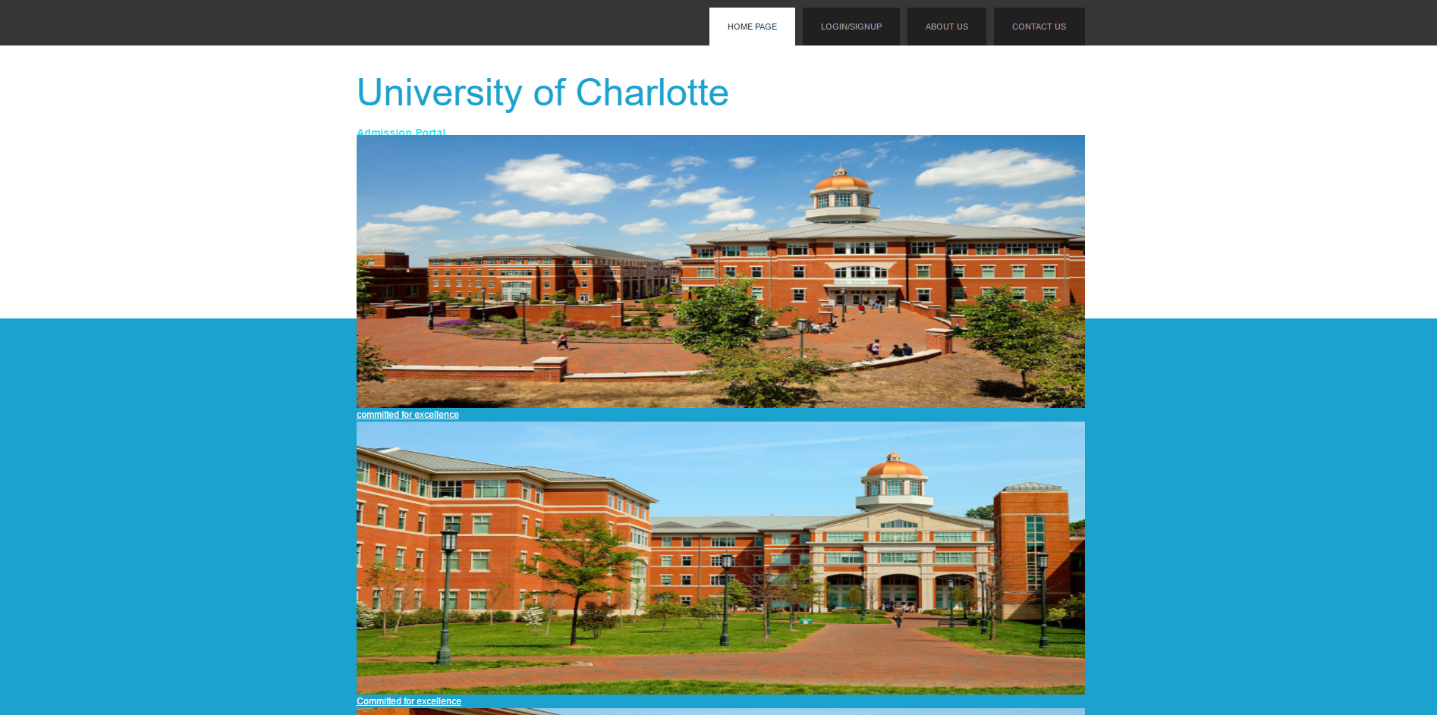
**Name of the page:** Home.jsp

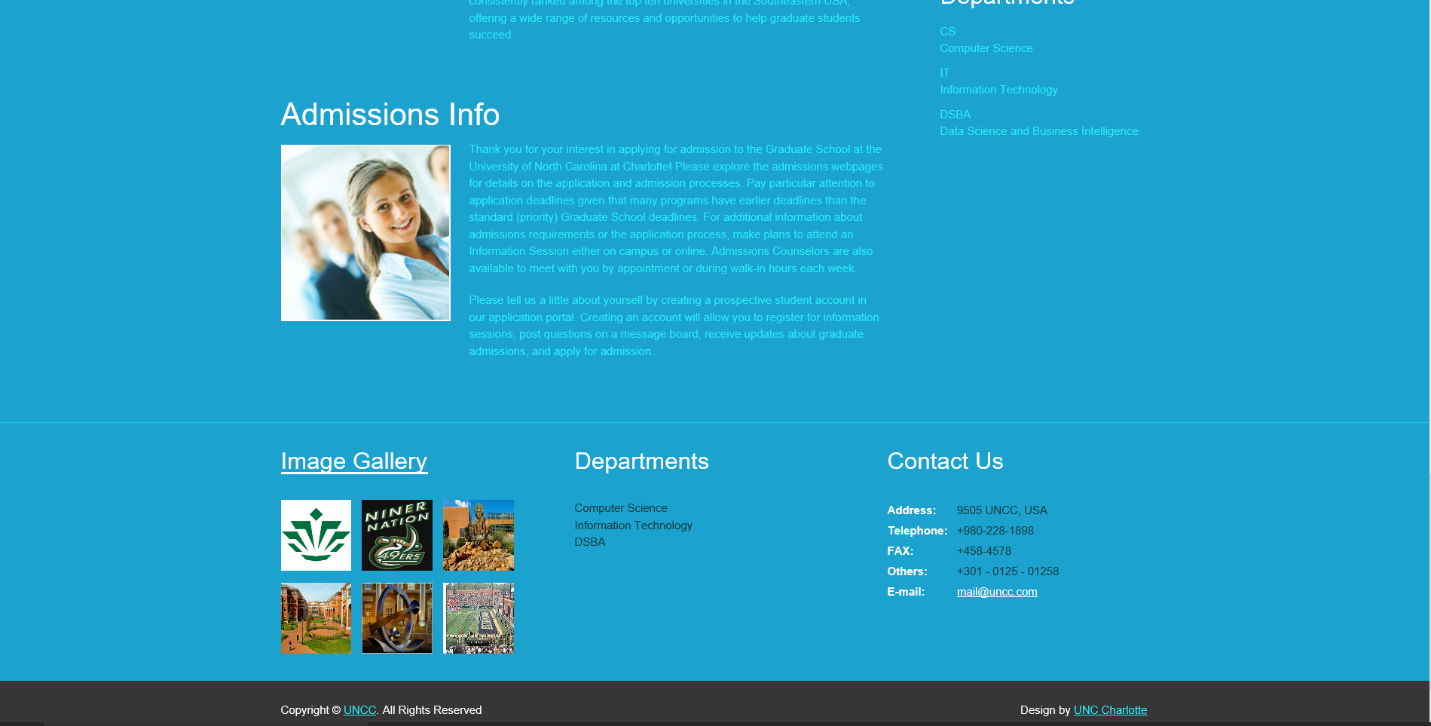
**Purpose of the page:** Starting page

**Audience of the page:** user and admin

**Buttons/Hyperlinks:** Image Gallery, Login, about us, Contact us,

**Special notes:** This is the landing page of the application.





**2. Login Page:**

**Name of the page:** login.jsp

**Purpose of the page:** login for admin and user

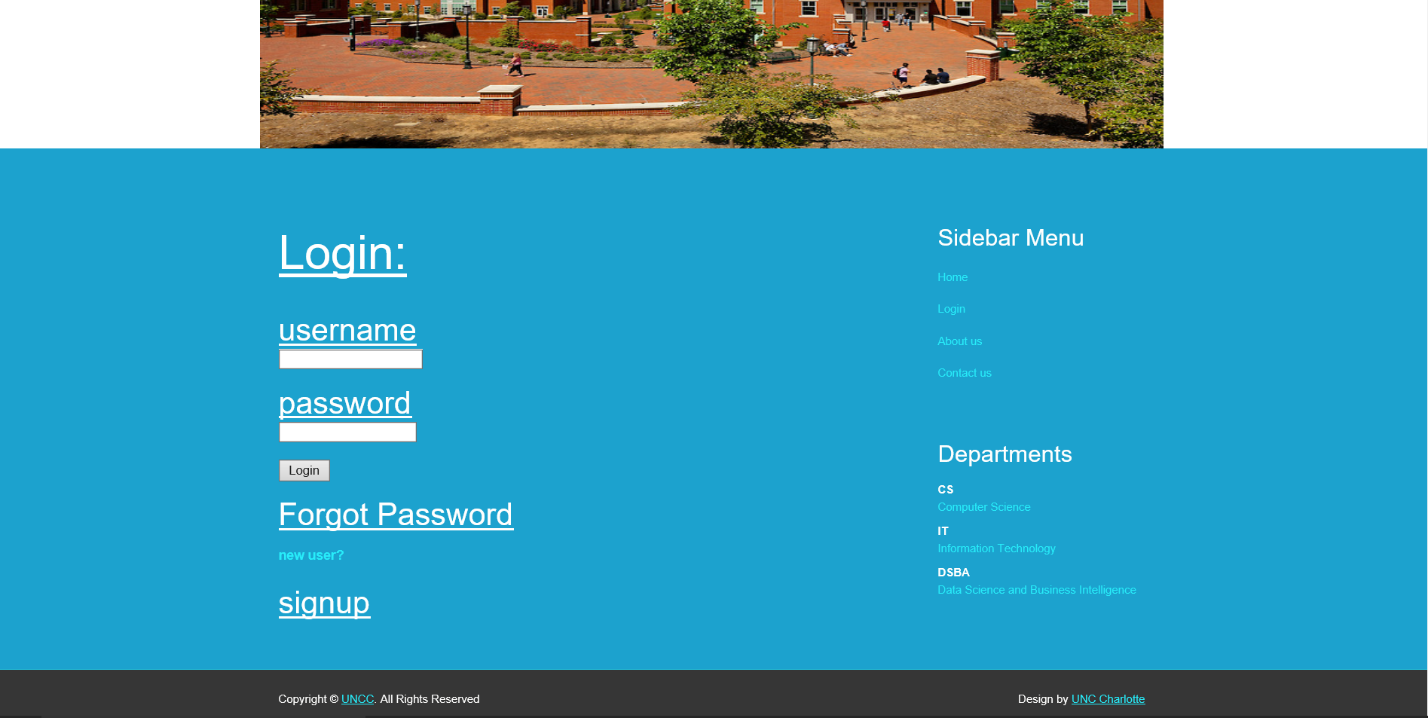
**Audience of the page:** user and admin

**Data fields:** username, password

**Validations:** admin and user will have different landing pages upon login, user name and password should match the database details.

**Buttons/Hyperlinks:** Signup, Forgot Password, Login button

**Special notes:** The user can login or signup or click on the forgot password

****

**3. About US Page:**

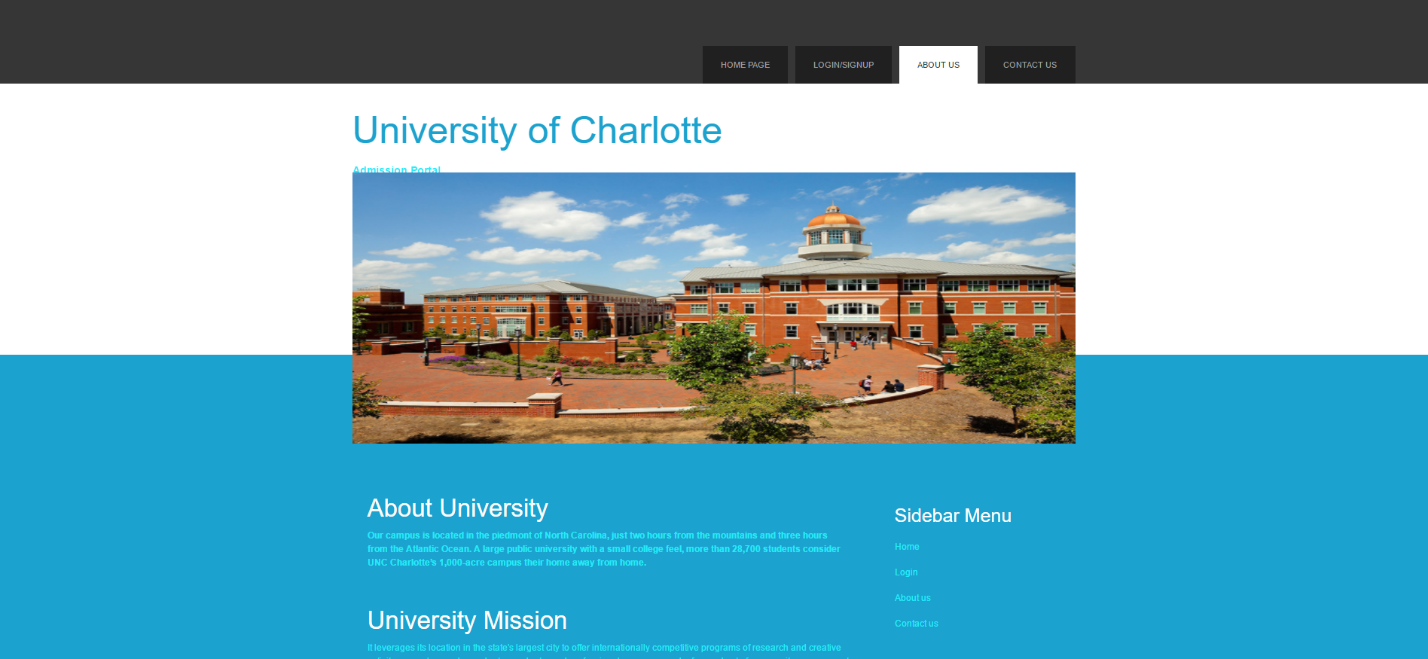
**Name of the page:** about.html

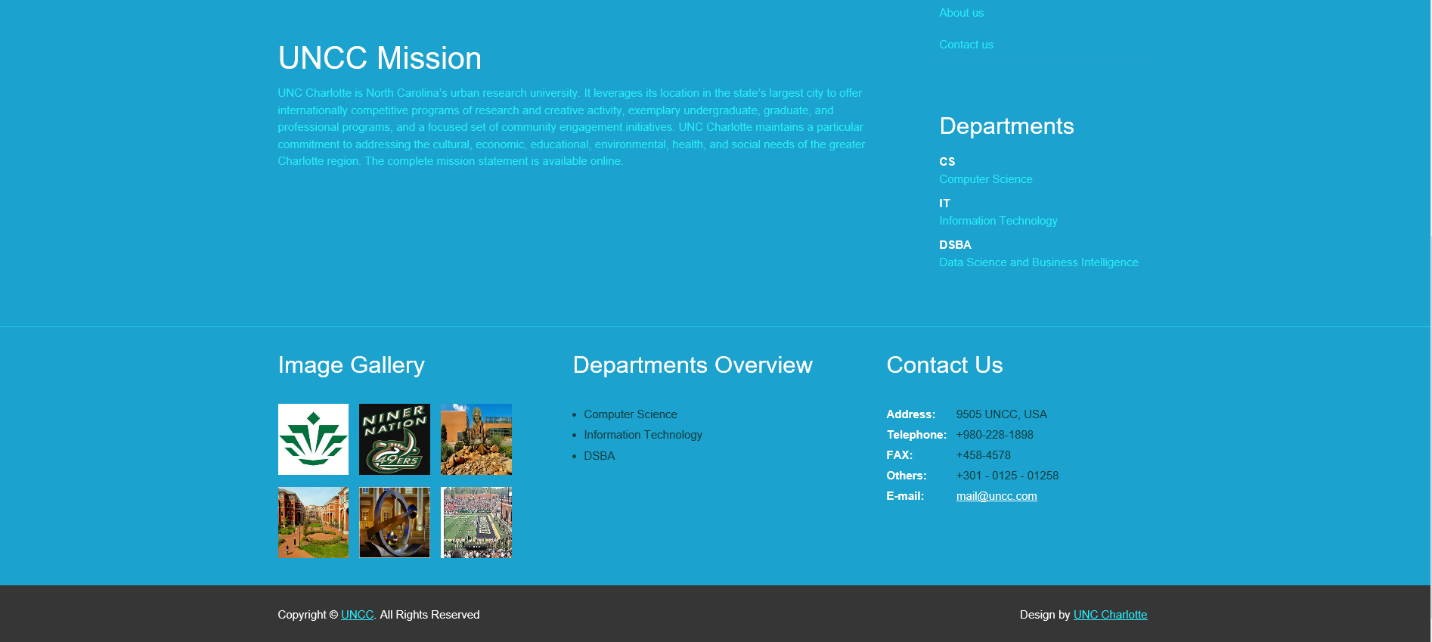
**Purpose of the page:** Starting page

**Audience of the page:** user and admin

**Buttons/Hyperlinks:** Image Gallery, Login, About us, Contact us,

**Special notes:** This is the landing page of the application.



****

**4. Contact Us Page:**

**Name of the page:** contact.html

**Purpose of the page:** Users can send a mail to the support team if he have any doubts regarding the site, or if he have any problems with logging in.

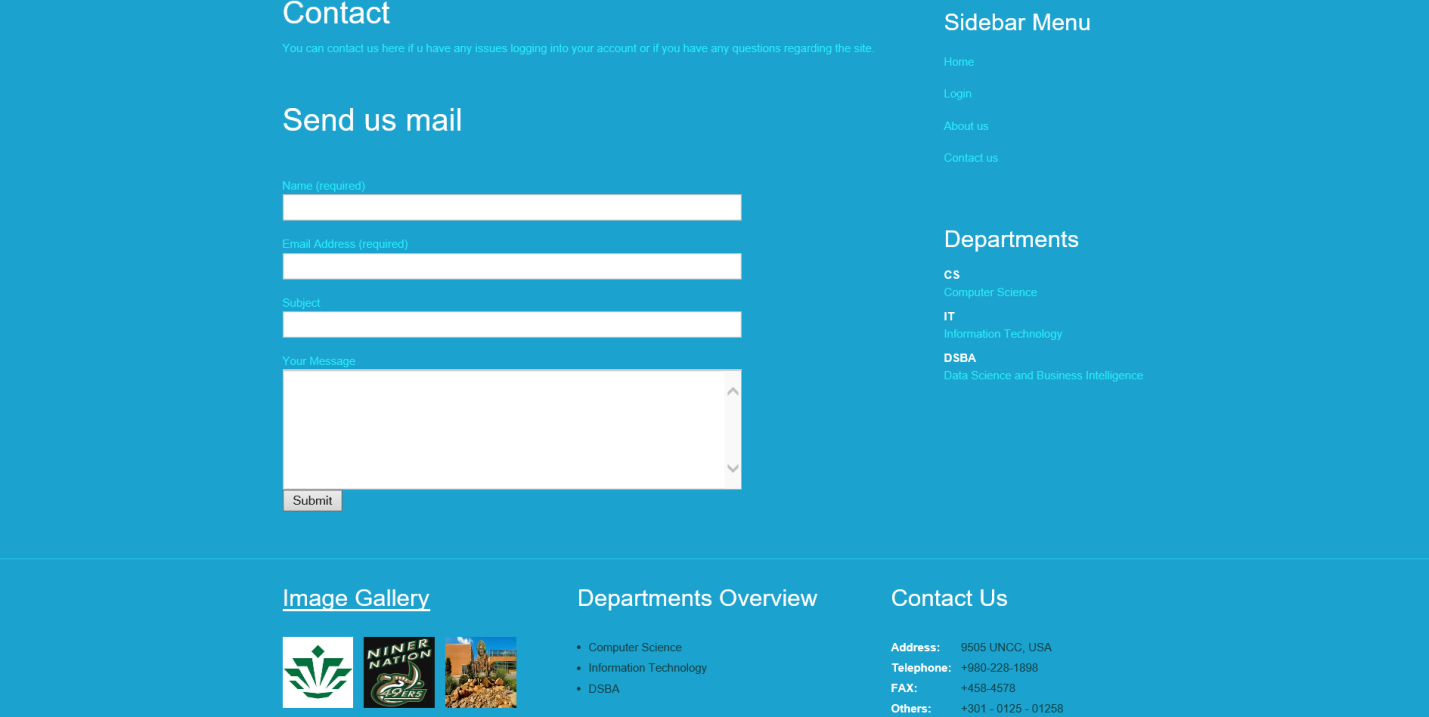
**Audience of the page:** user and admin

**Data fields:** Name, email, subject, message

**Validations:** email should have the email format (with @ and .com)

**Buttons/Hyperlinks:** submit button to send an email

**Special notes:** Using this page users can send an email for support or any questions



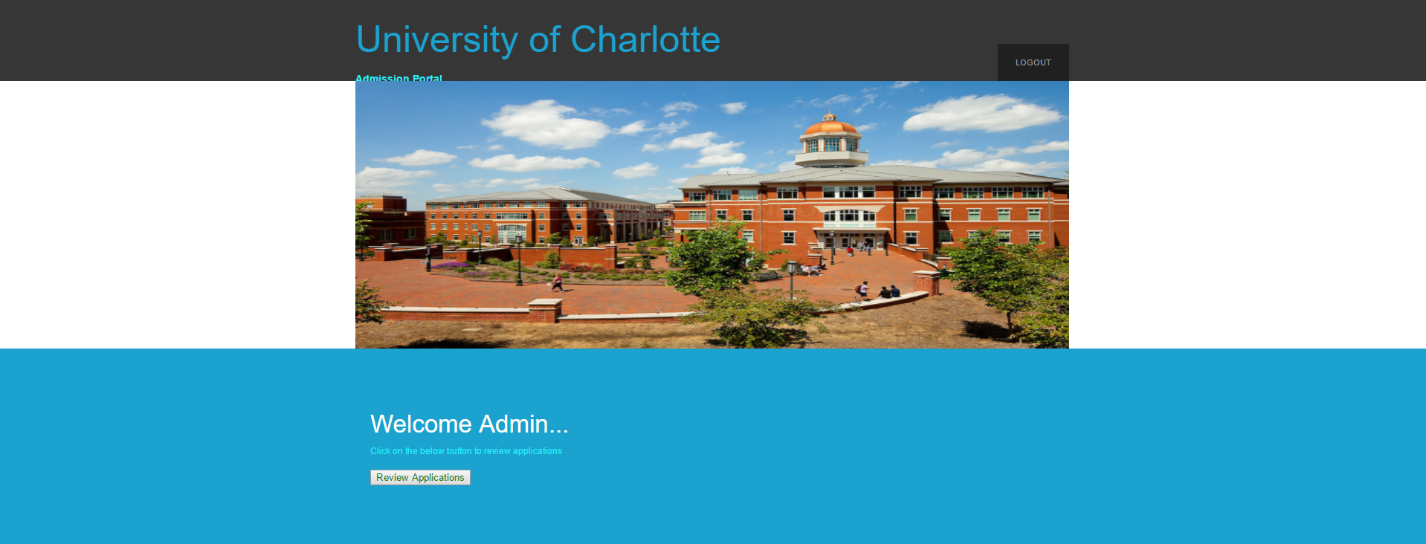
**5. Admin Page:**

**Name of the page:** admin.jsp

**Purpose of the page:** This will be loaded after the admin logs in where the admin can click a button to review the applications or logout.

**Audience of the page:** Admin

**Buttons/Hyperlinks:** review applications button



**6. Form Fill Page:**

**Name of the page:** cs.jsp

**Purpose of the page:** Page for student to fill his application form for the college

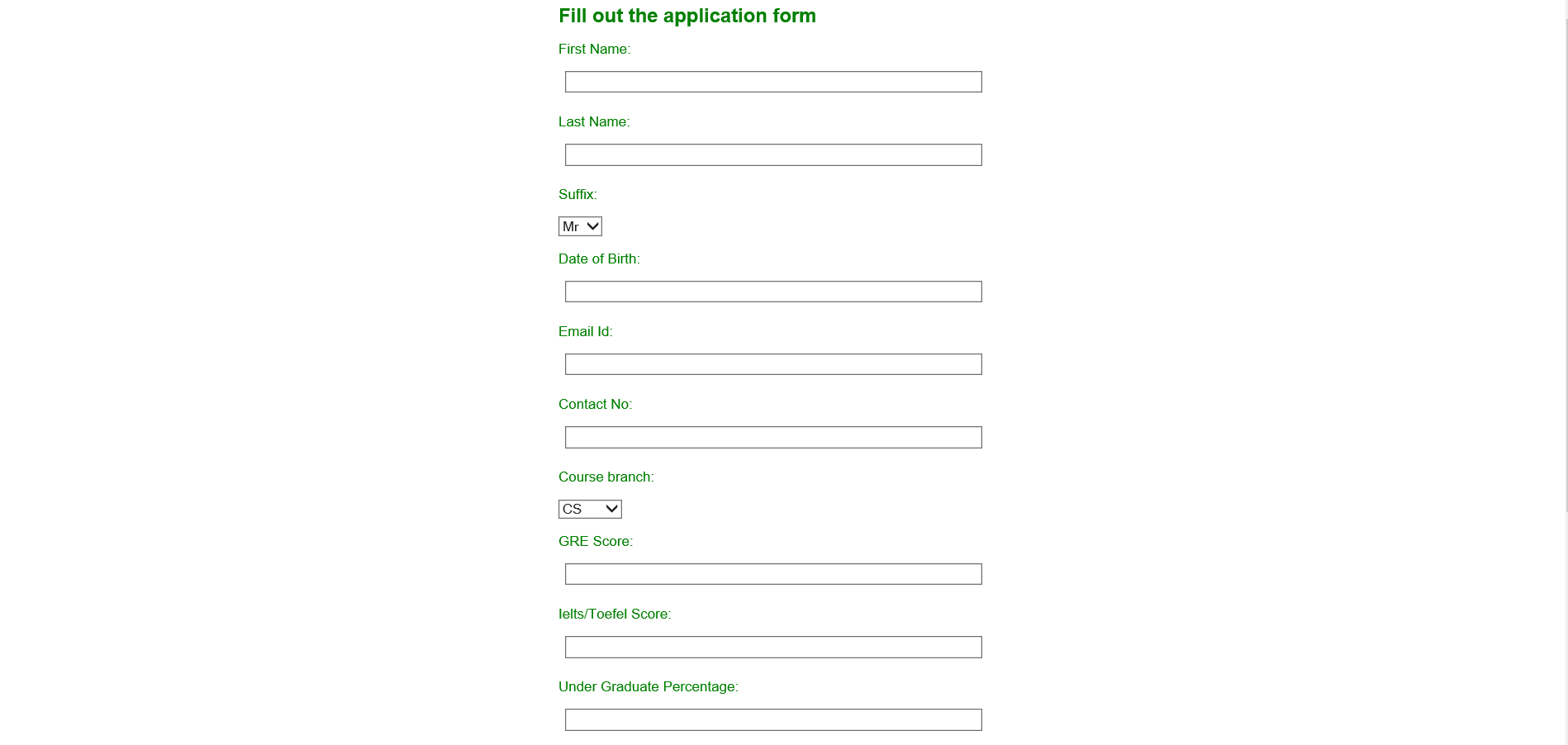
**Audience of the page:** user (student)

**Data fields:** firstname, lastname, suffix, dateofbirth, email id, contact no, course branch, gre score, ielts score, undergrad percentage, country, address line, city, state, zip

**Validations:** email should have the email format (with @ and .com)

**Buttons/Hyperlinks:** submit button to submit the form

**Special notes:** This is the form user submits for application to college.



**7. Students Page:**

**Name of the page:** students.jsp

**Purpose of the page:** Page for admin to display the students that have applied to the university, admin can accept or reject the application based on the student profile

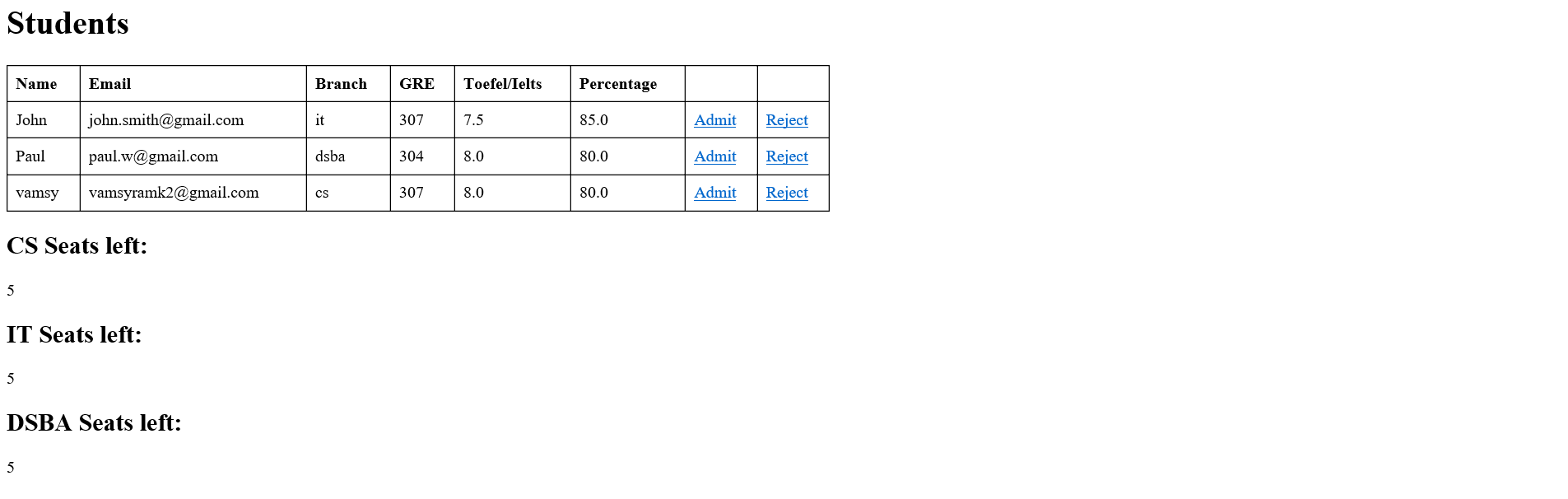
**Audience of the page:** Admin

**Data fields:** NA

**Validations:** NA

**Buttons/Hyperlinks:** accept, reject links

**Special notes:** This page also displays number of seats left in each department so that the admin can accept those department applications depending on the seats left in the respective department



**8. Students Login Page:**

**Name of the page:** studentpage.jsp

**Purpose of the page:** Page for student after he logs in where he can click on the application form button to fill an application or he can log out.

**Audience of the page:** Student

**Buttons/Hyperlinks:** application form



**CONCLUSION**

* University admission application is designed.
* It is tested within our constraints and the code has executed successfully.

**FUTURE SCOPE**

* Higher level Hashing and Salting user passwords.
* More details in the application form.
* Including more departments.
* Make a universal application like canvas.