Software Requirements Specification

For

Degree Plan Automation

Version 1.0

Prepared by

Nanditha Bodanapu

Sharanya Gottimukkula

Sri Harshini Vallabaneni

Aravind Thottempudi

SHAN

10/03/2018

Table of Contents

1. Introduction

This section introduces the web application that we would like to build, the purpose serves.

1.1 Purpose

This document specifies the requirements of Degree Plan Automation system. The degree plan submission system is a system used by Department of Computer Science and engineering at the University of North Texas to track the list of courses taken by students and the list of courses they would take in near future. This will help the department to guide the students such that they complete the mandatory courses and met the required grade expectations for graduation. The goal of this project is to provide an online system (Automation) for the existing manual process of degree plan submission.

1.2 Document Conventions

The document uses Times New Roman with font size 12. The headings are highlighted in bold and has single line spacing. The sub headings have a spacing of 6pt. Used tab space to segregate Functional requirements of a use class. Glossary (Appendix A) has all the abbreviations of the document.

1.3 Intended Audience and Reading Suggestions

The intended audience for our project Degree Plan automation is our professor Dr. Hyunsook Do (CSCE -5430 Fall '2018), group members of Shan (Nanditha Bodanapu, Sharanya Gottimukkula, SriHarshini Vallabaneni and Aravind Thottempudi) for development purpose and the remaining students of CSCE-5430 Fall '2018 who would be performing peer review of document.

From reading point of view this document includes five sections followed by two appendixes.
The second section provides information about the product perspective, users and list out few assumptions that we have made. The third section describes about the possible scenarios and crud operations among the users that we would like to implement for our project. The fourth section deals with the different hardware software and user interface. The fifth section deals with non-functional requirements like performance and security. The sixth section deals with database requirements.

1.4 Project Scope

The web application which is being developed is called "Degree Plan Automation". It is produced for the CSCE Department of UNT. This application is designed to facilitate the students and professors of Computer Science Department to process the degree plan submission process online. The user can access the application through a web browser.

The DPA System will allow students and faculty to register to the system. Once the student is registered and approved by the admin he/she can send requests to registered professors. Similarly, once the professor is approved by admin the professor can use the web application to respond to advisor requests or degree plan approval requests sent by the student. We also have an automated email system which would send the student a copy of intermediate degree plan request that he/she submitted to professor. We also support a help tab which would list out a html page containing common FAQ. The DPA system would have the following constraints. The Administrative specialist, Chair and Associate Chair cannot register directly through the web application. They should be added

by Admin. Though the application is online, it is not global meaning the users cannot access this website from different country.

1.5 References

[1]https://computerscience.engineering.unt.edu/sites/default/files/Computer_Science_MS_Degree_Plan.6.15.2017_1.pdf
[2] https://computerscience.engineering.unt.edu/people/faculty
[3] https://www.oracle.com/technetwork/articles/java/json-1973242.html
[4] https://en.wikipedia.org/wiki/Java_Persistence_API
[5] https://www.youtube.com/watch?v=xp7pvUUWWzs
[6] https://belitsoft.com/php-development-services/software-requirements-specification-document-example-international-standard
[7] Contacted Melanie for the approximate number of students every year in the CSCE department.
[8] SRS Template by Karl E. Wiegers (Resource provided in blackboard)
[9] IEEE recommended practice for software requirement specifications - IEEE std 830-1993
[10] https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html

2. Overall Description

This section lists out the high-level architecture of the system, details of what the web application can do and cannot do. It also lists out the assumptions and scenarios which were intentionally left out.

2.1 Product Perspective

DPA is a new, self-contained web application which provides functionalities for students, professor and the Administrative specialist to keep track of submitted applications. The application should be available for the students, professors, Administrative Specialist and Admin of the CSCE department provided they have computer resource with internet connection. The application should be compatible with chrome. Apart from this there are no hardware or software requirements for building the web application. This project uses Spring MVC [10] layered architecture.

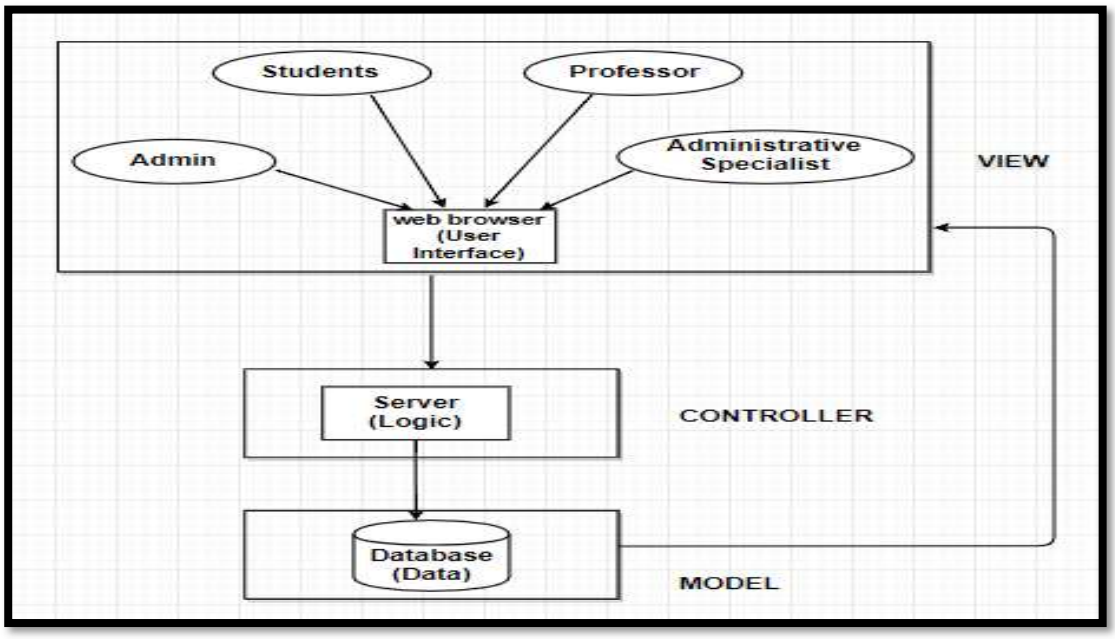


Fig 1.1 - Architecture

2.2 ProjectFunction

The following are the list of functions supported by DPA.

Request for Advisor:
The student can send requests to professor to be his or her advisor for courses.

Accept Advisor Request:
The professor can choose to accept or deny the request sent by the student.
The response would be sent to student.
Degree plan approval request:
The student can fill the degree plan and then send the form to the advisor for approval.

Degree plan approval request acceptance/rejection:
The professor can accept or reject the approval sent by the student.
If approved, the copy of degree plan would be sent to Administrative specialist who would further send it to Associate Chair and Chair.

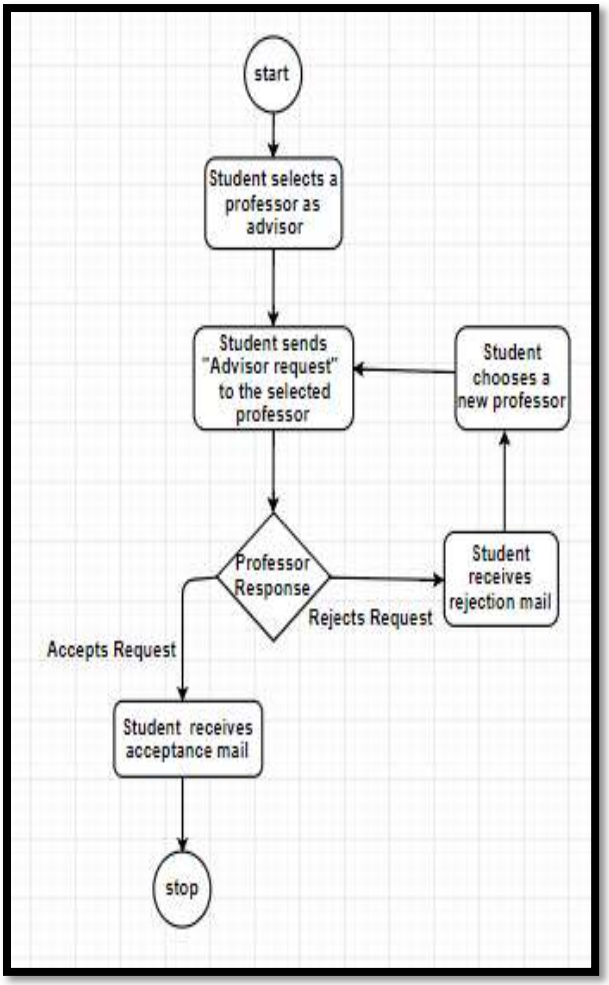


Fig 2.1

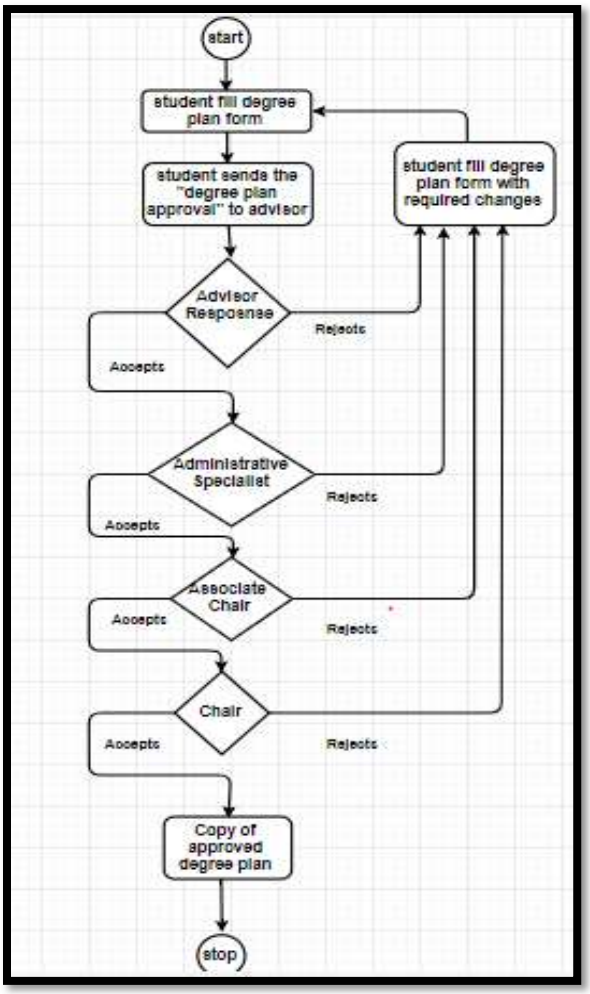


Fig 2.2

2.3 User Classes and Characteristics

Four classes of users interact with the system: Students, professors, administrative specialist and administrator and each of the user class has different requirements.

Students can manage their accounts, send requests to the professors and can also fill and submit the degree plan.

Professors can manage their accounts, receive requests, accept/reject the requests and can sign (approve) and submit the degree plans of students to administrative specialist or reject and suggest changes to the students.

Administrative specialist can manage both students as well as professors and is responsible for getting the second (associate chair) and third (chair) level signatures on the degree plans submitted to him/her. Administrative specialist can also update the degree plan information as per the department requirements.

Administrator/Admin manages all the three classes of users and is responsible for approving the student/professor registrations.
Out of the four classes administrator has privileges to control rest of the classes and is favored class when compared to other classes. Students and professor classes have same level of privileges. Administrative specialist is a bit high in terms of importance, when compared to student and professor classes.

2.4 Operating Environment

DPA system runs on all operating systems with a web browser Google chrome and Internet Explorer 11. The application operates on devices like computers and mobile phones that have an operating system with a web browser and internet connection.

2.5 Design and Implementation Constraints

The team has the knowledge of only SQL database, though there are many other databases like DB2 express, oracle which can relatively store more amounts of data, implementation is limited to MYSQL. Most of the fastest and efficient tools available for developing and deploying a maven project are paid and this limitation forced us to choose eclipse as the project development tool.

2.6 User Documentation

For this online web form, we will be providing the “Online HELP” Link at the top of the page to help the users to overcome the difficulties to use this form. As part of this online help, we will be “FAQ’s” that is frequently asked questions and the steps explaining how to use the online form. We also will be providing a link for submitting the incidents to report any issues while submitting or accessing the form.

2.7 Assumptions and Dependencies

We have assumed the following for DPA

- It is assumed that the admin exists for department and has access to the web application.

- Once the student is rejected by the professor for advisor request, the student will not send advisor request to the same professor again.
- Once the Chair approves the degree plan it is sent to tousle graduate school for updating the database and then the approval is sent to student. However, in our system a copy is sent to student once it's approved by the chair.
- Even if the Chair is the student's advisor, the Chair cannot sign as "the advisor" and "Chair" in a single turn. It must follow the sequential process as shown in fig 2.2. The same is applicable for Associate Chair.

3. System Features [6]

Use cases and functional hierarchy will organize this section. Four different classes of users can interact with the system and has different functionalities.

3.1 Use Case - Administrator

FR 3.1.1: Administrator Login
Sequence number: FREQ-1
Administrator needs to log in to access the functionalities on the home page.

Scenario: Login success: If the administrator types in valid credentials in the input login fields, Displays Administrator's home page.
Scenario: Login Failure: Reloads the login page and notifies the error to the administrator. This case arises if the user types in invalid inputs or wrong credentials in the input login fields.
If the Administrator forgets his password, he has an option to reset the password by clicking the 'forgot password' link.

FR 3.1.2: Manage Users
Sequence number: FREQ-2
Description: After the successful login, administrator has an option 'Manage Users', which is a dropdown field, it has four options.
Software Capability required: FREQ-1

Option 1: Manage Administrative specialist
If the administrator chooses this option, he should be able to add, delete or edit the information of administrative specialist.

Scenario: Add administrative specialist
Adds the new administrative specialist, when the administrator fills in the details and clicks add.
Scenario: Delete administrative specialist
When the administrator clicks delete, Deletes the existing administrative specialist.
Scenario: Edit administrative specialist
Update the details of existing administrative specialist.

Option 2: Manage CSCE Department Chair
If the administrator chooses this option, he should be able to add, delete or edit the information of the CSCE Department Chair.
The admin can perform CRUD operations on Department Chair.
Option 3: Manage CSCE Department Associate Chair
Admin can add, delete or edit the information of the CSCE Department Associate Chair.
The administrator can perform the CRUD the Department Associate Chair.

Option 4: Manage Students/Professors
Administrator can delete and update the students/professors.

FR 3.1.3: Verify and approve students/professors
Sequence number: FREQ-3
Description: After login, administrator has an option to approve or reject the newly registered students or professors. Adds the new student or professor to the system, if the administrator clicks approve, otherwise deletes the student or professor data permanently from the system.
Software Capability required: FREQ-1

3.2 Use Case - Student

FR 3.2.1: Student registration
Sequence number: FREQ-4
Description: The new student of CSCE department should first register through this registration form available on the home page of the web application. This form demands username, email, admission year, unique identifier and password.

FR 3.2.2: Student Login
Sequence number: FREQ-5
Description: The student login page asks the user name and password for the login.

Scenario: Login success: If the student gives the valid credentials in the input login fields, redirects the student to the student home page.

Scenario: Login failure: Reloads the login page and notifies the error to the Student. This case arises if the student types in invalid inputs or wrong credentials in the input login fields.

If the Student forgets his password, he has an option to reset the password by clicking the 'forgot password' link.

Software Capability required: FREQ-1, FREQ-3 and FREQ-4

FR 3.2.3: Account management
Sequence number: FREQ-6
Description: The Student can manage his account that is the student can change his password by clicking 'change password' option, update his information like graduation year and major by clicking 'edit profile' at any time after creating the account.
Software Capability required: FREQ - 5, FREQ-1, FREQ-3 and FREQ-4

FR 3.2.4: Manage requests
Sequence number: FREQ-7
Description: Student has a capability to send the request to professor and withdraw

Scenario: Withdraw the request

The student can withdraw the sent request by clicking 'withdraw' button right next to the professor profile, at any time after sending the request to the professor and the acceptance by the professor.

Scenario: resend the request

The rejected request is resent when the student clicks resend.

Software Capability required: FREQ - 5, FREQ-1, FREQ-3 and FREQ-4

FR 3.2.5: Fill the Degree Plan form
Sequence number: FREQ-8
Description: Student fills in the degree plan form by clicking 'degree plan' option and then selecting the form appropriate to his/her major.

The form has drop downs and populates the subjects based on the selected degree, major and the specialization.
Software Capability required: FREQ - 5, FREQ-1, FREQ-3 and FREQ-4

FR 3.2.6: Manage the form
Sequence number: FREQ-9
Description: The student will need to do many further modifications and submissions for the form initially filled and saved

Scenario: Request the professor signature
The student submits his degree plan to his major professor by clicking the 'send to professor' button at the bottom of the form.
Scenario: Edit and resubmit the form
The student can edit the degree plan as per the suggestions of the professor by clicking 'Edit' button on the top of the form and resubmit by clicking 'send to professor' button at the bottom of the form.
Scenario: Submit to the administrative specialist
After the approval from the major professor, student submits to administrative specialist by clicking 'send to administrative specialist' button at the bottom of the form.

Software Capability required: FREQ-6, FREQ - 5, FREQ-1, FREQ-3 and FREQ-4

3.3 Use Case - Professor

FR 3.3.1: Professor Registration
Sequence number: FREQ-10
Description: The new Professor of CSCE department should first register through this registration form available on the home page of the web application. This form demands username, email, admission year, unique identifier and password.

FR 3.3.2: Professor Login
Sequence number: FREQ-11
Description: The professor login page asks the user name and password for the login.

Scenario: Login success: If the professor gives the valid credentials in the input login fields, redirects the Professor to the Professor home page.
Scenario: Login failure: Reloads the login page and notifies the error to the Professor. This case arises if the professor types in invalid inputs or wrong credentials in the input login fields.
If the professor forgets his password, he has an option to reset the password by clicking the 'forgot password' link.

Software Capability required: FREQ-1, FREQ-3 and FREQ-11

FR 3.3.3: Account management
Sequence number: FREQ-12
Description: The professor can manage his account that is the professor can change password by clicking 'change password' option, update his information by clicking 'edit profile' at any time after creating the account.
Software Capability required: FREQ - 10, FREQ-1, FREQ-3 and FREQ-11

FR 3.3.4: Manage requests
Sequence number: FREQ-13
Description: Professor can accept/decline student request. By clicking 'Accept' right beside the student profile the professor can accept the request and by clicking 'Reject' professor can reject the request.
Software Capability required: FREQ - 10, FREQ-1, FREQ-3 and FREQ-11

FR 3.3.5: Approve/Decline degree plan
Sequence number: FREQ-14
Description: The professor approves degree plan by clicking approved at the bottom of received degree plan file. He can reject the plan by clicking 'Reject' button and add the comments and changes that the student needs to make.
Software Capability required: FREQ – 10 and FREQ-11

3.4 Use Case - Administrative Specialist

FR 3.4.1: Administrative Specialist login
Sequence number: FREQ-15

> **Scenario:** Login success: If the administrator Specialist types in valid credentials in the input login fields, Displays Administrator's home page.
> **Scenario:** Login Failure: Reloads the login page and notifies the error to the administrator Specialist. This case arises if the user types in invalid inputs or wrong credentials in the input login fields. Administrator Specialist has 'forgot password' link.

FR 3.4.2: Account management
Sequence number: FREQ-16
Description: The Administrator Specialist change his password by clicking 'change password' option, update his information by clicking 'edit profile' at any time after creating the account.
Software Capability required: FREQ - 15

FR 3.4.3: Manage CSCE Department degree plan information
Sequence number: FREQ-17
Description: Updates the courses in each of the four groups specific to the major.
Software Capability required: FREQ - 15

FR 3.4.4: Manage the degree plans
Sequence number: FREQ-18
Description: Has check, approve and reject options for each of student's degree plan received. The administrative specialist has option 'send to associate chair' and 'send to chair' options through which the administrative assistant can request their signatures on the plan.
Software Capability required: FREQ – 15

FR 3.4.5: Send approval notice to the student
Sequence number: FREQ-19
Description: After getting the signatures of associate chair and chair, administrative assistant sends approval notice to the student by clicking 'approved' button.
Software Capability required: FREQ – 15, FREQ-17

3.5 FR - Online Help

Sequence number: FREQ-20
Description: Online help link provides guidance to the users in terms of website usage.

3.6 FR - Notifications

Sequence number: FREQ-21
Description: All the users receive notifications of submissions and approvals or rejections.

Table 3.1 Plan for implementing project in three development phases

Functional Requirement	Development Phase	Start Date	End Date
FREQ-1	Phase 1	10/04/2018	10/05/2018
FREQ-4, FREQ-5	Phase 1	10/06/2018	10/07/2018
FREQ-10, FREQ-11, FREQ-15	Phase 1	10/08/2018	10/09/2018
FREQ-2	Phase 1	10/10/2018	10/11/2018
FREQ-7, FREQ-13	Phase 1	10/12/2018	10/16/2018
FREQ-8, FREQ-9, FREQ-14, FREQ-18	Phase 1	10/19/2018	10/24/2018
FREQ-6, FREQ-12, FREQ-16	Phase 2	10/25/2018	11/06/2018
FREQ-17	Phase 2	11/06/2018	11/08/2018
FREQ-19	Phase 2	11/09/2018	11/07/2018
FREQ-3	Phase 3	11/08/2018	11/14/2018
FREQ-20	Phase 3	11/15/2018	11/17/2018
FREQ-21	Phase 3	11/18/2018	11/20/2018

Gantt chart for development plan for 3 phases:

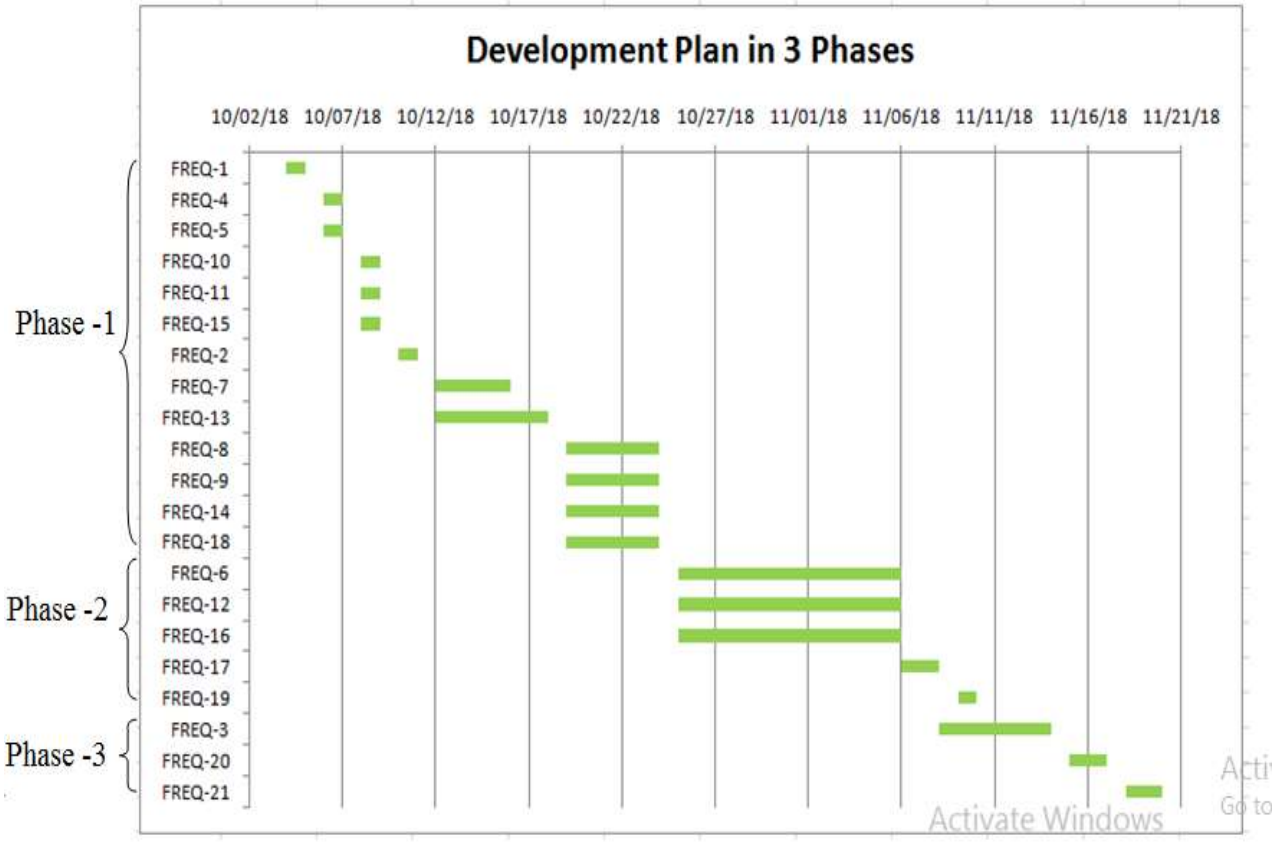


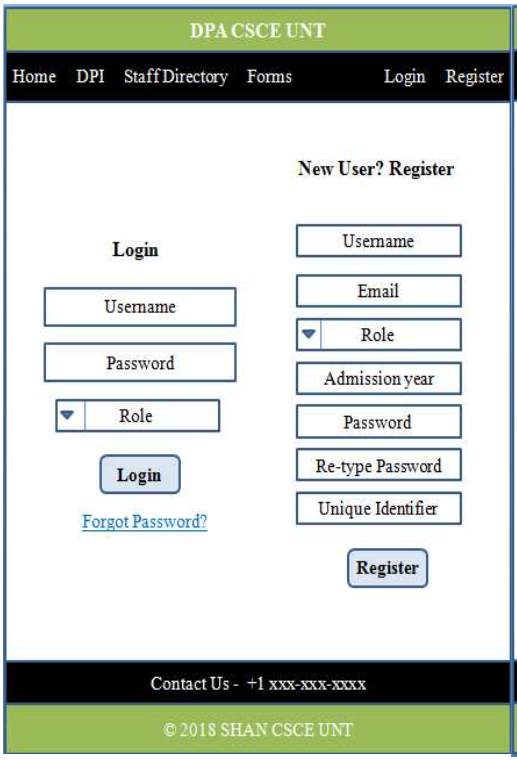
Fig 3.1

4. External Interface Requirements

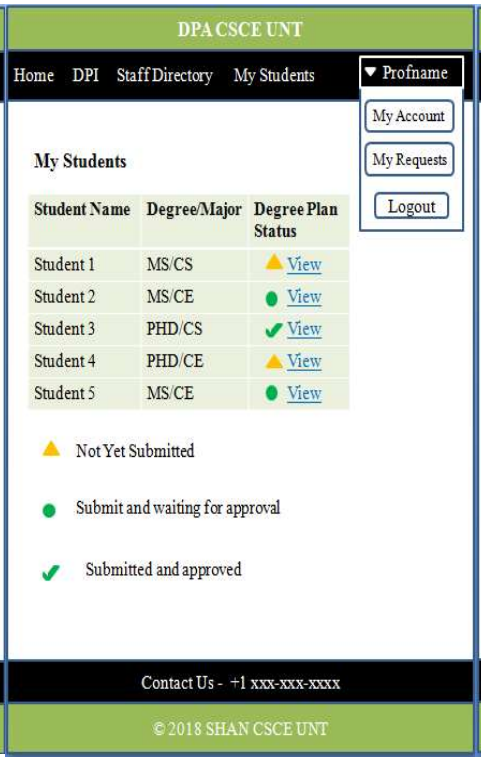
This section discusses the interfaces of the DPA system.

4.1 User Interfaces: Sample Screens [5]

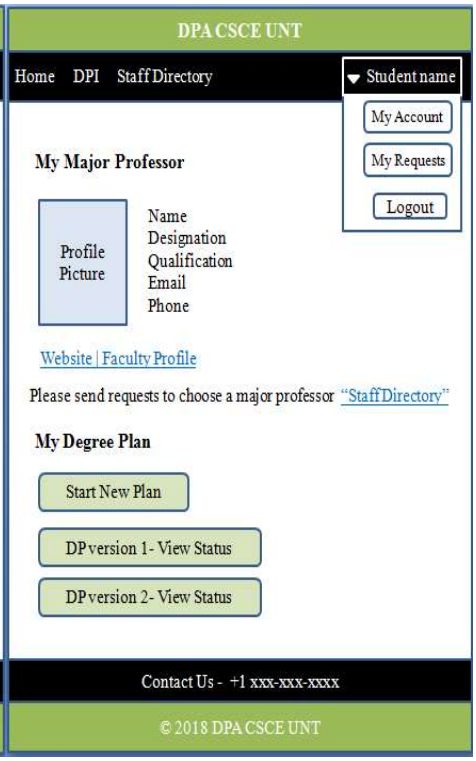
Screen 1



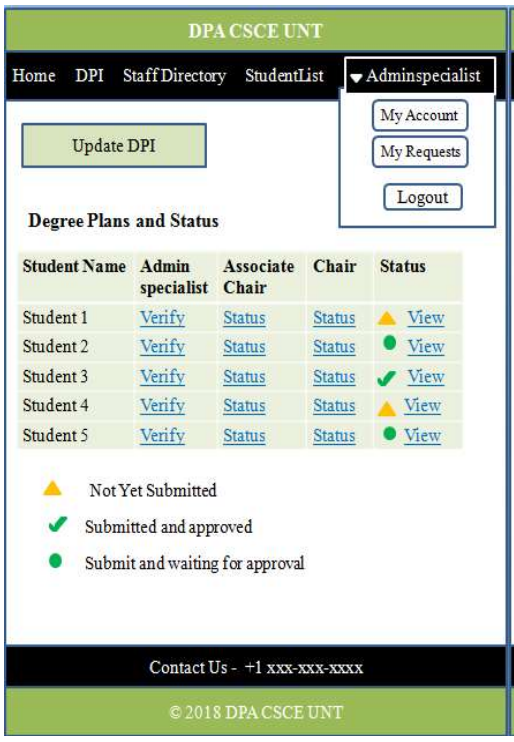
Screen 2



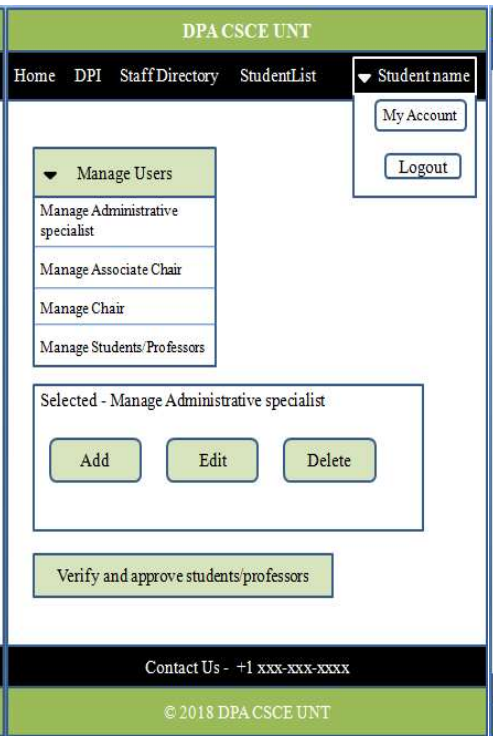
Screen 3



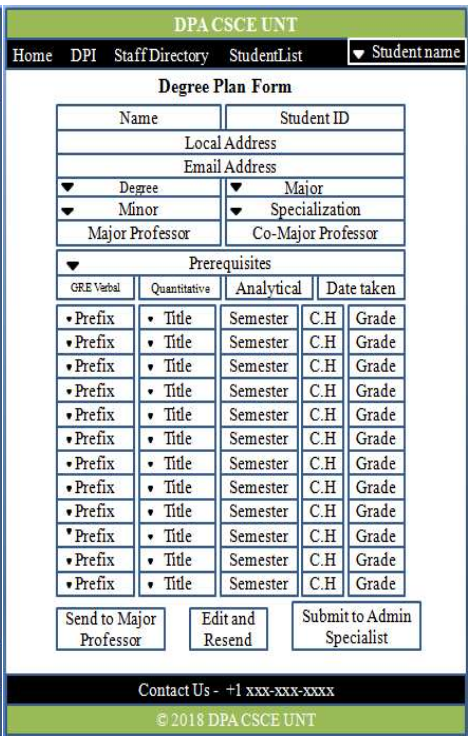
Screen 4



Screen 5



Screen 6 [1]



GUI standards or style guides: Created a set of guidelines for styles.
Guidelines on text format, input field and button appearance, colors and visual correctness.

Screen Layout Constraints: Devices of different screen sizes support the application screen layout. Bootstrap adjusts the page layout to the screen size making the application responsive.

Standard buttons/functions on every screen: Home and Help button appears on all the screens.

Error message display standards:
Error messages are short and precise and not give out unnecessary information.
Displays error messages in red color and have different style.

4.1 Hardware Interfaces

Any device with a web browser supports this application. The devices include Personal computers, tablets and mobile phones. HTML and JavaScript by using event handlers handles the user inputs and actions through keyboard. For example, when the user clicks submit button, the 'on click' attribute of HTML invokes JavaScript event handler to handle the request, which sends the submission request to the server. This is how hardware communicates with the software.

4.2 Software Interfaces

Eclipse Photon is the tool to handle the maven project (Entire application as a whole) along with the server (Apache tomcat - version 9). Google chrome - version 69.0.3497.100, Firefox 62.0 and internet explorer supports this application.
HTML5, CSS3 and bootstrap for User interface development. JavaScript 8 to add dynamic nature to the HTML pages.
Java Mail API: To send email notifications after approval.

XMLHttpRequest API: Transfers data from the browser user interface to the web server. XMLHttp request is required wherever there is no need to reload entire page.
Java libraries like util, io and lang are used to handle the data and files generated by requests from user.

Java JSON library [3]**:** Used json library of java to handle the form data received from the user request in JSON format. The java json libraries help convert the JSON data to java objects and handle them. This is requiring handling the form data received through the AJAX request and store it into java beans.

JPA (Java Persistence API) [4]**:** it is a java API. It establishes connection to the database and manages the data between the java objects and relational database (MYSQL workbench). It helps mapping the java objects to the Relational database elements. This is required to store data from java beans into database.

4.3 Communication Interfaces

This is a web application uses electronic forms written in HTML and JavaScript. A web browser supports these forms and application. Browser uses the HTTP protocol to send request to from the browser to the server and response server to the client.

The request sent using POST method of HTTP rather than GET because get displays everything in the URL including password, which is not safe. GET method is used to access data.
The application requires administrative specialist to send email notification to the student upon approval. The Java mail API makes use of SMTP, IMAP and POP3 protocols for mail transfer and delivery. MD5 hashing for sensitive data like passwords. Encrypting the confidential messages before transmitting.

5. Other Nonfunctional Requirements

The following section discusses few nonfunctional requirements applicable for DPA system

5.1 Performance Requirements

The requirements in this section discusses the measurement placed on the performance of DPA system

PR1:

Criteria: Access of names of courses while filling the degree plan form.
Description: While the student begins to fill the degree plan form, as he/she begins to type the course name, the course name along with course number should pop-up dynamically.
Expectation: The names of the courses must be fetched within 2 seconds.

PR2:

Criteria: The recipient must receive the forms sent by the sender within 24 hrs.
Description: Students and professor may have to exchange requests and response forms like advisor request form, degree plan submission form etc.
Expectation: The requests sent by students or the response sent by the professor should reach the intended recipient within 24 hrs. from the time it was sent by the sender.

5.2 Security Requirements

SR1: Only registered users approved by the admin should be able to access the system.
SR2: The passwords stored in database should be stored in encrypted form.
SR3: The signatures of professors, administrative analyst, associate chair and chair should be digital signatures.

5.3 Software Quality Attributes

Usability: For now, the system only supports in chrome and internet explorer.

Reusability: For now, our project would support for master's and PhD students.
The code forms for masters and PhD students are similar hence we would be using the similar component of the code to ensure maximum reusability.

Testability: The project would be based on java, so we would be introducing junits to all possible functionalities. This would cover almost 60% of the code. The next 40% would be for integration, once integrated we would perform manual testing of product.

6. Other Requirements

Database Requirements:
Around 48 [2] faculty members and 200 to 300 students will be accessing this system and the database need to accommodate the transactions of around 500 members [7] and also be capable of storing files of the students. A centralized relational database system like MYSQL workbench would be ideal to meet the above criteria. It maintains synchronization with Java objects and relationships between tables using primary and foreign keys.

Appendix A: Glossary

Term	Definition
Admin/administrator	The admin is responsible for managing high level authority
Administrative Specialist	Coordinates the degree form from student to chair and associative chair
Advisor	A professor who accepts the student request to be his/ her advisor.
API	Application Programming Interface
CRUD	Create, read, update and delete
CSCE	Department of computer science and engineering
Degree Plan Automation (DPA)	The web application which is the automated version of manual degree plan submission process
DPI	Degree Plan Information
FAQ	Frequently Asked Questions
FREQ	Functional Requirement
GUI	Graphic User Interface
HTML	Hypertext mark- up language used for designing front end
HTTP	Hypertext Transfer Protocol
IMAP	Internet Message Access Protocol
MD5	Message Digest- an encryption algorithm
MySQL	Structured query language which used to insert/update (manipulate) data to database
POP3	Post office protocol version 3
PR	Performance Requirement
Professor	The Professor who teach in university
SMTP	Simple Mail Transfer Protocol
SR	Security Requirement
Student	The student who is enrolled in university
UNT	University of North Texas

Appendix B: Issues List

The online help button on the website to help the users is a very good feature for the application but the decision to work it out is still pending and the implementation of digital signatures is to be discussed.