PLACEMENT COORDINATION SYSTEM

Team No D2

TEAM MEMBERS

Sl.No	Name	USN#
1.	Ninaad R Rao	01FB16ECS232
2.	Midhush Manohar T.K.	01FB16ECS208
3.	Nishant Ravi Shankar	01FB16ECS236
4.	Meghana Ganesh	01FB16ECS205
5.	M Sumukha	01FB16ECS185
6.	Nesara B R	01FB16ECS226
7.	Mohammed Faizan	01FB16ECS211
8.	M Pradeep Kumar	01FB16ECS183
9.	Manish Chandrashekar	01FB16ECS191
10.	Neha B Garg	01FB16ECS224
11.	Omkaram Dhanush	01FB16ECS240

Placement Coordination System

Placements are a crucial part of college, yet is often always chaotic and disorganized. In order to provide a seamless and convenient placement experience, we have developed a Placement Coordination System.

The goal of the project is to make the handling of major functionalities regarding placement processes easier for both the students and the placement coordinator.

The features of the system are divided into two broad categories, each having their own components:

1. User (Student) - The system allows registration for placements,displaying college policies for students, easy one click registration for a particular company which includes eligibility check,displays placement statistics statistics based on tier, company, CTC etc and has a prediction model to predict the CTC of the next company. Company related information is made more accessible and students can use a job search filter to do so and students can even view upcoming events. The entire placement process is made more organised by allowing the user to keep track of the companies visiting along with the dates, provides the necessary venues as well as seating allotments for the students. It also tracks the schedules of the companies visiting. An ‘Alumni Connect’ blog service feature enables users to view previous interview/code experiences of various companies. Additionally, the system includes a ‘Skill Refinement’ feature that allows the students to hone their skills on several interview-themed subjects.
2. Admin (Coordinator) - The coordinator can keep track of students placed (and if the student has full time + internship or only full time or only internship). In addition, the system also encompasses an ‘Automail’ feature that automates the process of sending mails to the

different companies, as well as updating their responses into the system. The admin will also be allowed to control the pre-placement and venue details via the 'Schedule' feature.

The user logins to the site using their USN/ID and password and new users will be redirected to a separate sign up page. Once logged in, the user can search and find the details of a particular company. The user can view a list of upcoming companies through the Company Tab, company statistics and trends through the Statistics tab, seating allotment and information about the registered companies through a Profile tab and any notifications/updates through a notifications tab.

There are 4 components and 2 interfaces in the Placement Coordination System. The components include :

Browser- It is an application used to access and view the website.
Accounts Application - Contains the login page and password reset page to authenticate users and handle any exceptions by allowing the password to be reset. There are two views for this. One for the admin(placement coordinator) and the other for students. Based on the credentials, the app will redirect to the respective view(studentapp or coordinator app).
Student Application - The students application contains the student dashboard which displays student information and related functionality such as registering, viewing stats etc.
Coordinator Application- The coordination web application is for the placement coordinator where he/she can track the students who are part of the placement cycle.
Database-Contains all information regarding students and users, companies,labsand seating, scheduling etc. All the other components will query this particular component to display and update data.

The design is based on modularity and information hiding and follows MVC approach.

FEASIBILITY STUDY

Contents

1. Problem Statement

Placements at many colleges happen in a very haphazard way. Taking our own college as an example, students do not know where the online test is going to be conducted neither do they have reliable information regarding the schedule of the visit to the campus. This project will focus on managing this entire process with an application that will have various kinds of functionalities. The main aim will be to make the entire process smooth, efficient and reduce any kind of confusion.

2. Executive Summary

Placements are perennial activities that play a crucial role in the life of a student. While the current system relies on a mail based mechanism, there have been various cases of the information not being made available to the entire student body. Certain situations of malpractices have also been lost under the radar. Better execution of this procedure can play a pivotal role in easing the mindset of a student as he/she enters the examination hall. With the goal of combating these difficulties, this project has been proposed. It aims to build a central and transparent platform that students can rely on completely for all things related to placements. Powered by the cloud, this would be a one stop portal that contains information regarding the details of the respective company, a tentative list of the inbound hirers, as well as a uniform seating allotment system assigned beforehand that would ensure minimal overall chaos.

This document covers the current systems and processes in existence for dealing with some of these problems. It then moves on to mention its shortcomings, and proposes a system that would tackle these problems. A high level block diagram is provided to delineate the workings of the proposed system, followed by a depiction of the basic user interface. An overview of the implementation details are covered in the technology considerations section, along with the customer benefits and potential marketing strategies for the product. The various issues that a developer may come across are also listed, along with a few plausible assumptions and constraints. An alternative solution is described, which provides a perspective on the advantages of the current chosen path. A plan of the project is also formalised towards the end, which aids in defining the end goals in concordance with a tangible timeline.

3. Current Systems and Processes

3.1 Current Operations

Main operations in the placement coordination system are:

1. **Sharing company details**: Every student receives company details through email. Eligible students have to fill the google form or register on PESU App.
2. **Check every student meets eligibility criteria:** If the company has any eligibility criteria, they send the shortlist for the eligible candidates for the first round. The location of the pre-placement talk is shared.
3. **Pre Placement talk:** Company talks about the job description, work culture, benefits, etc.

4. **Written Test:** The placement head announces the location of the written test. Eligible students take the written test under a proctored environment.
5. **Shortlisting students:** Based on cut off marks, students are shortlisted and they are called for the interview process.
6. **Interview:** We need to make sure that all shortlisted students attend the interview. If someone fails to attend the same, further action has to be taken.

3.2 Physical Environment

1. **Computer labs** -Once the company is done with the initial shortlisting of students based on the CGPA cutoff, computer labs are needed to conduct online exam which is a part of the placement process.
2. **Email service** - Currently, to communicate and share company information with the students, email service is being used. Company registration forms are being sent through emails where student fills his details and notes down the date.
3. **Rooms** - After round-1 is over, to conduct face to face interview for the eligible students, rooms are allotted based on the number of students shortlisted for the interview process.

3.3 User Organization

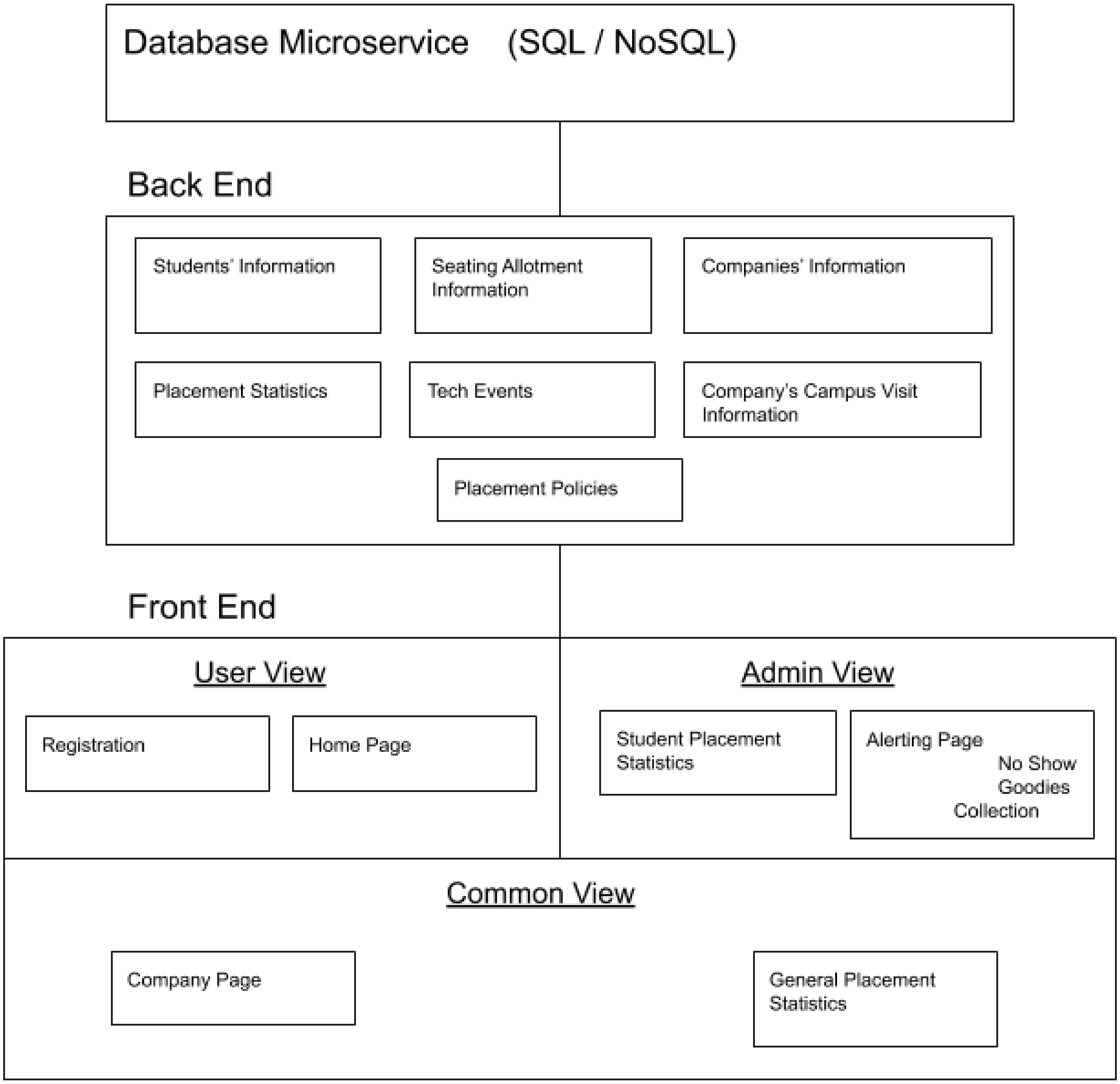
Since the system is used in various colleges and universities, users of the system are 3rd and 4th year students aiming to get a job/internship. This system is being used by the placement office to fetch information and status of students registered for placement and any other relevant details.

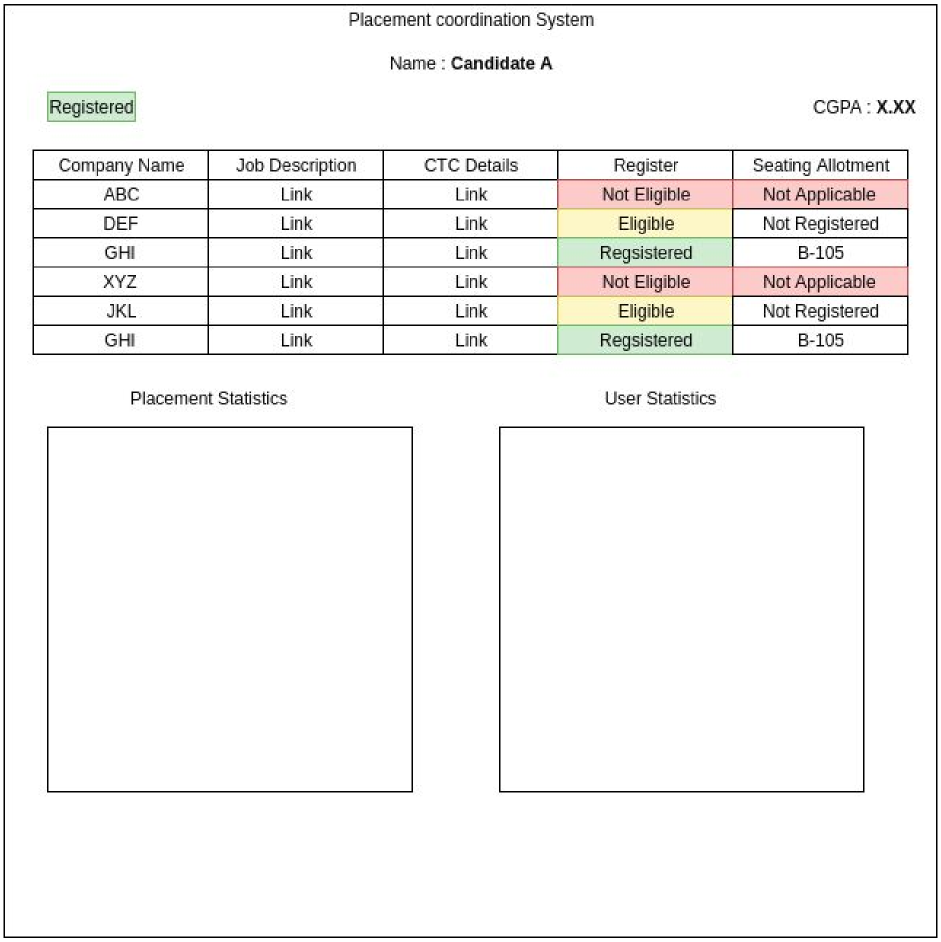
4. System Objectives

4.1 Description of Products and Services

There are a lot of companies that visit the campus every year. Keeping track of each and every company becomes difficult. The current system sends a mail to students every time and students have to register in a google form repeatedly. We are trying to reduce this labour by creating a web application that would include all the details provided by the students and will have all the information related to the company in one place. Many students also give fake data to qualify the first round. This system will reduce such cases. It will also include the seating allotment information for eligible students to appear for Round-1. This system will automatically disallow students who already have an offer (only fte or only internship or fte+internship) or are not part of the placement cycle. Doing so will ensure that the placement process is fast, easy and efficient. By providing notifications for any last minute changes, students will be aware of the situation and there will be minimal confusion and chaos.

4.2 High Level Block diagram showing the solution


Database Microservice (SQL / NoSQL)
Back End
Students' Information
Seating Allotment Information
Companies' Information
Placement Statistics
Tech Events
Company's Campus Visit Information
Placement Policies
Front End
User View
Registration
Home Page
Admin View
Student Placement Statistics
Alerting Page
No Show
Goodies
Collection
Common View
Company Page
General Placement Statistics



Placement coordination System

Name : **Candidate A**

Registered

CGPA : **X.XX**

Company Name	Job Description	CTC Details	Register	Seating Allotment
ABC	Link	Link	Not Eligible	Not Applicable
DEF	Link	Link	Eligible	Not Registered
GHI	Link	Link	Regsistered	B-105
XYZ	Link	Link	Not Eligible	Not Applicable
JKL	Link	Link	Eligible	Not Registered
GHI	Link	Link	Regsistered	B-105

4.3 Targeted Customers and Benefits

This entire software will benefit both students and the placement office. Students can keep track of the companies that they have registered and get relevant information regarding the whole process with just one click and placement office can upload all the details of the company at one particular place rather than sending mails to all the branch groups.

This software helps students to keep track of the companies that are to visit the campus. It is difficult to keep track of the online tests and the interviews of several companies that are to visit over a week. This software alarms them with a reminder about the upcoming tests and interviews which they have registered.This software helps the placement office to post the schedule and the seating allotment and send it to the registered students.

4.4 Technology Considerations

The placement department currently does not have an on-site IT team to implement the digital solution, and has previously tied up with the PES Academy to deploy it on the PESU App. But it is highly unreliable, since there are issues with real-time updates and inconsistent usage of the application, since the department asks students to register through the PESU App only for some of the companies, and relies on emails for the others. The placement department can either outsource this to a company or tie up with the CS department to implement this. Since students of the CS department have the required technical skills, the latter option allows the CS students to develop and test the application. The hindrance here might be the students' hectic schedule, which may result in a delayed deployment. Collaborating with a company can lead to faster deployment, with reliability since there will be dedicated engineers working on application, but is expensive.

The department's IT infrastructure currently consists of desktops in every cabin, with access to high speed internet, which is enough to use the application in real time.

The application will follow a microservices based architecture. The user interface application will be implemented using either Angular 6 or ReactJS. The backend will be split into a set of services, and each of these services will be a small Django application. Each of these services will be deployed in a container and will communicate with each other via REST APIs. This ensures modularity and ease of development. To address scalability concerns, we may implement a Kubernetes based solution in the future, to automate scaling of individual services. NoSQL and Relational databases will be used for storage, which will eventually depend on the type of data that needs to be stored.

The department should consider using AWS/GCP to host the application since it ensures reliability, speed and cost-effectiveness. On the other hand, keeping an in-house server requires a lot of investment upfront, along with reliable power supply.

The placement department will not have to be concerned about the adoption of the application, since the technology literacy of the students as well as the placement department is already very high.

5. Product/Service Marketplace

The primary idea of our placement coordination system is to keep track of inbound companies, registration of students and also seating arrangement of students. As placements at many colleges happen in a very haphazard way, our software can be used by the PESU Placement cell. This software can be used by other colleges and software organizations too.

Upwork, LinkedIn, Glassdoor and many more job portals already have very well maintained and established online presence for off campus placements. As there's no such portals available for on campus placements, this software is going to provide a convenient way for students, placement cell and also companies.

Talking about placements in our college, registrations and all the updates are done through either PESU app or Email. Our platform will include features like one-click registration, calender system for notifications, keep track of offered students, seating allotment etc.

Why should you use this software?

- It will *create awareness among students regarding available career options and help them in identifying their career objectives*.
- It will ensure *Time efficiency*.
- Seating allotment information for every placement activity.
- Reducing malpractices by keeping track of the students who are offered internships, tier-1 company, etc.
- One click registration for students which means students don't have to fill their information for each and every companies which *reduces data generation and unnecessary updates*.

6. Marketing Strategy

The first step of the marketing strategy is to showcase the current drawbacks with the fragmented placement process to the authorities. After highlighting the drawbacks, a list of advantages with respect to our software can be listed out. For example, one click registration of students. After this, continuous demo of sub products or partial deliverables could be shown to attract the placement office to use the software. After this, in a beta version, we could run a demo with the full software and show them all the specifications.

In the later stages we are trying to spread the application across others colleges and college students. As most of the college students are active on social media we can use paid advertisements and internet marketing to spread our idea. But not all of them are attracted towards social media ,so we can visit targeted college campus and conduct seminars, workshops, or presentation that creates a buzz through word-of-mouth. This is a great way to involve students and show them what we have to offer. To attract the placement offices we can post the demo of our sub products on youtube.

7. Organization and Staffing

Currently there will be 11 students working on this project in various domains like front-end, back-end, data analytics, and cloud computing. Once the software is up and running, a minimum of 5 people would need to maintain the software for long run. There can be multiple updates that need to be done based on the dynamic nature of how placement happens every year. To make it even simpler, a shift to an android application might be desirable to make it more convenient and additional staffing for android developers need to be created. Since the company is thinking of expanding every year, additional staffing of at least 50 might be needed

8. Schedule

Placement Coordination system project is expected to take two -three months from project approval to launch. The following is a high level schedule of some significant milestones for this initiative:

High Level Overview (Subject to Change)

Date	**Task**
Aug 20th, 2019	Initiate Project
September 2nd, 2019	Project kickoff meeting and Database creation and population begins.
September 8th, 2019	Database populated. Assigning work to the subteam.
September 15th, 2019	Basic UI completed
September 29,2019	Work on backend
October 20th, 2019	Completion of the backend and Starting Integration
October 25th, 2019	Completion of Integration, Testing begins
November 18th, 2019	Completion of the project

9. Financial Projections

The financial projections for the placement coordination system is highlighted below.

Assumptions are:

1. Year-1: Only one college uses this software
2. Year-2 : colleges in Bengaluru use the software
3. Year-3 : Karnataka colleges use the software
4. Year-4 : All Indian colleges use the software

Measure	**Year-1**	**Year-2**	**Year 3**	**Year 4**	**4 year total**
Application sales projection	Rs 50 lakhs	Rs 4.73 crores	Rs 30 crores	Rs 100 crores	Rs 135.23 crores
Cloud computing costs	Rs 1 lakh	Rs 20 lakhs	Rs 1 crore	Rs 10 crores	Rs 11.21 crores
Additional staffing costs	Rs 20 lakhs	Rs 1 crore	Rs 10 crores	Rs 30 crores	Rs 41.20 crores
Training for sales and marketing staff	Rs 10 lakhs	Rs 5 lakhs	Rs 3 lakhs	Rs 2 lakhs	Rs 0.20 crores
Design and Build of Online platform	Rs 5 lakhs	Rs 15 lakhs	Rs 1 crore	Rs 10 crore	Rs 11.20 crores
Total additional costs	Rs 36 lakhs	Rs 1.4 crores	Rs 12.03 crores	Rs 50.02 crores	Rs 63.81 crores
Cash inflow	Rs 14 lakhs	Rs 3.33 crores	Rs 17.97 crores	Rs 49.98 crores	Rs 71.42 crores

10. Issues

After a detailed prognosis, the following points have been mentioned as commonly encountered issues:

1. Susceptibility of the Database to Race Conditions - in scenarios where the database is accessed simultaneously by a large number of users, it is possible to encounter race conditions.
2. Immunity to Security Issues:
    a. SQL Injections - a type of web application security vulnerability in which an attacker attempts to use application code to access or corrupt database content.
    b. Cross Site Scripting (XSS) - which targets an application's users by injecting code, usually a client-side script such as JavaScript, into a web application's output. The

concept of XSS is to manipulate client-side scripts of a web application to execute in the manner desired by the attacker, allowing them to execute scripts in the victim's browser which can hijack user sessions, deface websites or redirect the user to malicious sites.

   c. Broken Authentication & Session Management - possibility to wreak havoc on the seating allotment.
   d. Insecure Direct Object References - when a web application exposes a reference to an internal implementation object (files, database records, directories and database keys) to gain access to a user’s personal data.

3. Scalability of the system according to the traffic of users - a kubernetes orchestration tool should be in place for overcoming this issue.
4. Learning Curve - knowledge of React and Django required to develop the full stack of the application.
5. Reduction of downtime of the system - ensured via High Availability.
6. Ability of the application to serve data dynamically.

11. Assumptions and Constraints

The assumptions made are:

1) The application will be up and running throughout the year since the placement happens throughout the year.
2) The application will have a team monitoring it, and will change the business logic in scenarios where the placement rules might change sometime in the future. The team will also need to address any scalability issues, since the number of students are increasing year by year.
3) The application should be the go-to solution for the placement department. This means that the placement department should adopt this application to suit their needs 100%. If not, they may run into the current fragmented process all over again.
4) The placement department should allocate a fixed budget every year to update the application to the latest standards, along with migration to future device platforms.

The constraints are:

1) The proposed budget to maintain the application may be out of scope for the placement department.
2) The adoption of this application will require approval from multiple levels of authority.

12. Alternatives

12.1 Alternative 1

A possible alternative for placement coordination system would be to manually handle the placements . Students can be informed about companies visiting campus via email, which could be automatically synced to their calendar and registration can happen as and when companies arrive through google forms. Student information can be verified by getting details from the COE. A general seating is provided by allocating labs and rooms. Any last minute changes or any urgent notices will have to be broadcast to all students via email.Tracking of students that have offers will also have to be handled manually or maintained in a database.

This alternative offers less cost and resources, but is also time consuming, labour intensive and inefficient since tracking students happen manually. Students need to keep filling the form repeatedly with the same set of information.

13. Findings and Recommendations

Based on the information presented in this feasibility study, it is recommended that the university adopts the proposed Placement Coordination System. The findings of this feasibility study show that this initiative will be highly beneficial to the organization and has a high probability of success. Key findings are as follows:

13.1 Project objectives

- Summary of issues concerning:
    - Development and Implementation: The learning curve for the backend and frontend, as well as the scalability issue, are crucial for the implementation of this project.
    - Assumptions, constraints and limitations: Monetary constraints, though minimum, still play a pivotal role in the success of this project.
    - Project scope: By the end of this project, we aim to establish a transparent system that ensures a smooth operation throughout the placement season.

- Results of Research:

Technology:

- The existing authentication system of the PES application would be continued as it has already been well established.
- The database would be leveraged using Amazon's AWS.
- The backend of the application would be developed using Django - a Python web framework - and the frontend would be built using Angular. These have been chosen to ensure extensibility of the application.
- Once in place this technology is simple to operate and maintain for a relatively low cost.

Marketing:

- This initiative will allow the Placement Cell to reach out to a larger number of companies via a more methodical procedure.
- It will also ease the process of taking tests/interviews by leveraging the more transparent system.
- Malpractices would also be lowered to a great extent.

Organizational:

- There would not be any change to the existing organizational structure.
- No new facilities are required since the entire application is cloud based.
- Capital Investments would be minor, limited to the credits required for the online infrastructure.

REQUIREMENTS SPECIFICATION

Table of Contents

1. Introduction

The goal of this SRS document is to list out all the requirements for building a web based application for the task of placement coordination. This document will be an outline for the collaborators to meet the specifications and guidelines for the intended audience.

1.1 Purpose and Intended Audience

The purpose of this product is to make the entire process of placements unhindered, organised and efficient. This product aims to automate the process of placements with features like one-click registration, seating information for a particular test, placement related events using a calendar and an administrator side which provides features like sending out details, scheduling events and sending out notifications for any last minute changes. And also features like skill refinement, automail, alumni connect, company search engine and placement statistics.

The intended audience are all students, the placement cell of a particular college and the various associated companies.

1.2 Scope

Placement Coordination System is a web-based application that will streamline the entire process of placements from the commencement of a particular company's arrival to its selection of desired applicants.

The proposed system would let students register for the placement cycle and create their profile by filling their academic and other professional details which would be used to process and check the number of eligible students according to the job description.

- Major features are:
    - one-step registration
    - keeping track of companies and students
    - maintaining a scheduling system for managing the placement drive

We aim to deploy this application in PES University and later expand it to other colleges.

1.3 Definitions, Acronyms, and Abbreviations

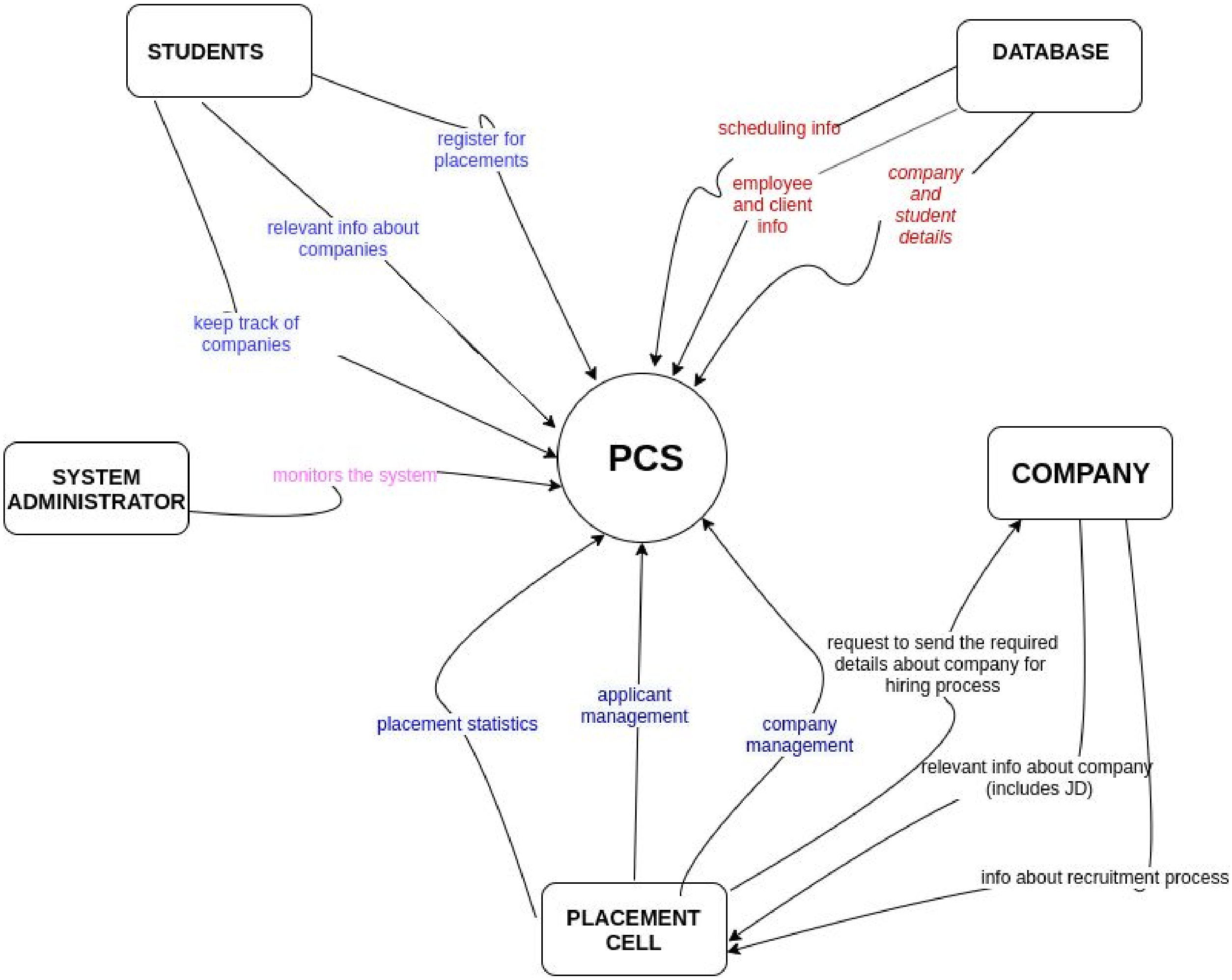
PCS - Placement Coordination system.
GUI - Graphical User interface.
Client-Server model - a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients.
Portal - a website or web page providing access or links to other sites.
AWS - Amazon Web Services
System refers to the placement coordination system.
CTC - Cost to company details often used to indicate the package offered by the company.
USN - User Serial Number often given to students by the college authorities to access the college facility.

Push Notification - an automated message sent by an application to a user when the application is not open.

2. General Description

2.1 Product Perspective

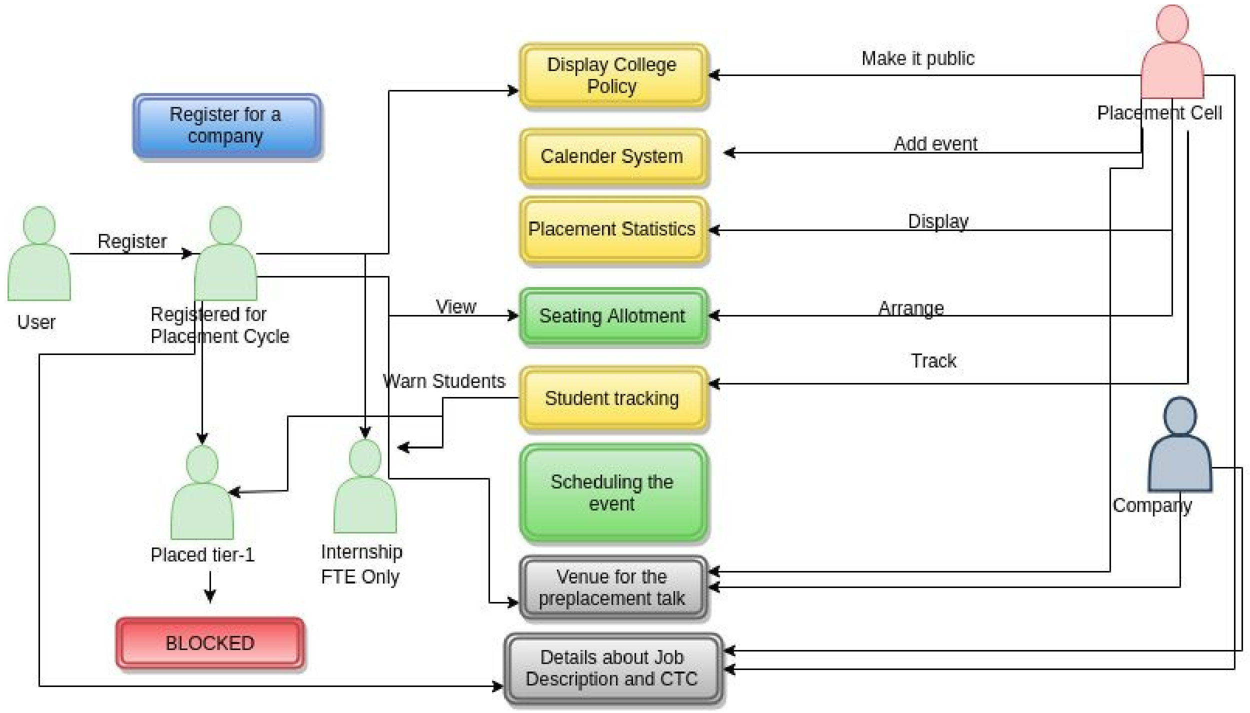
PCS is an application that replaces the current infrastructure which relies on a mail based mechanism. It is a web based system implementing client-server model. This portal provides platform that students can rely on completely for all things related to placements.



2.2 Product Features

FE-1 : Registration for Placement Cycle.
FE-2 : Displaying College policy for students to be aware of.
FE-3 : Registration for a particular company.
FE-4 : Keeping Track of companies.

FE-5 : Maintaining a calendar system for students.
FE-6 : Seating allotment for shortlisted students.
FE-7 : Venue for the Pre-Placement Talk/Interviews.
FE-8 : Displaying Placement Statistics.
FE-9 : Company Information and Job Search Filter.
FE-10 : Keeping track of students who are already placed/have an internship.
FE-11 : Keeping track of the right branches to be informed.
FE-12 : Making sure there is no overlap in any of the events of various companies.
FE-13 : Company Search Engine
FE-14 : Alumni Connect
FE-15 : Skill Refinement
FE-16 : Automail



2.3 User Characteristics

System Administrator	System administrator is a body that monitors the health of the website.
Student	Student is a person who is taking part in the placement season, with the goal of finding a job or an internship. The student will use the system to keep track of the companies visiting the campus based on his/her desired job profile, and also registering for those desired companies.
Placement Officer	Placement Officer is a person who is in charge of organizing the placement season. This person will use the system to communicate with the companies, schedule the written tests/interviews with

	respect to time and location, alert students with the latest updates with respect to the placement season, and also view student information. The updates can be either regarding future companies visiting the campus (with CTC and the job profiles), or could be to send out lists of students who are selected for further rounds in a recruitment process. The placement officer may have to be trained to use the system efficiently and productively.

2.4 Operating Environment

OE-1:	System is not dependent on geographical areas.
OE-2:	System shall operate in newest versions of all web browsers.
OE-3:	There should be no constraint on users being able to access the system at a given time.
OE-4:	Data is generated by online registrations and stored in the mongoDB database of AWS.
OE-5:	Continuous service is preferred, but as long as there is no data loss, minor service interruptions can be tolerated.
OE-6:	Personal data will be stored in the mongo database, so the AWS database must be secure (The security methodology is yet to be explored).
OE-7:	React will be needed to create the user interface for the system.
OE-8:	The users will be notified via push notifications.
OE-9	Django will be used for the backend of the system.

2.5 Constraints

CO-1:	The user should have the latest browser version.
CO-2:	The AWS services for free-tier level is limited.

2.6 Assumptions and Dependencies

AS-1	No more than 3000 active users can access our system at a time. This will be worked upon after the initial version of the project.
AS-2	User's resume(pdf file) cannot be more than 10Mb in lieu of preventing excess storage usage.
AS-3	Users using this web application belong to the specific college and they have permission to access this application.

D-1	Initially, the data needs to be verified with the concerned authority to check if there is any mismatch with the data given.

3. Specific Requirements

3.1 Functional Requirements

FR1	Student Information	The system shall allow a user (limited to Placement officer) to view student information. A student's information can cover the status of the student in the placements season and academic details.
FR2	Seating Allotment Information	The system shall allow any user to view the seating allotment information for a company's written test. The allotment information includes the location and the time of the written test. The functionality shall also extend to dynamically updating the allotment information to account for last minute changes and alerting the students with the updates. The update functionality is restricted to the Placement officer.
FR3	Companies' Information	The system shall allow any user to view companies' information. The user shall be allowed to search for companies with input as specific job profiles, the CTC, and the date of visiting the campus. Editing and updating the companies' information shall be limited to the Placement officer.
FR4	Placement Statistics	The system shall allow any user to view placement statistics. The statistics can be represented graphically and numerically. These statistics are not limited to that current year's placement season, but also include previous years' placements. Editing or updating the statistics shall be limited to the Placement officer.
FR5	Placement Policies	The system shall allow any user to view the placement policies. Editing the placement policies shall be limited to the Placement officer.
FR6	Tech Events	The system shall allow any user to view the upcoming tech events in the campus. Updating, editing and adding these events will be limited to the Placement Officer. This is an additional feature that will be implemented only after completion of the placement specific functionalities.
FR7	Company's Campus Visit Information	The system shall allow any user to view a company's campus visit information. The information includes the date of visiting, the CTC and the job profiles being offered. However, this information will be sent in the form of real-time alerts to students via push notifications.
FR8	One Click Registration	The system shall allow students to register for a particular event through one click with the interaction with the database for past data

FR9	Scheduling event	The system shall allow the admin to schedule an event so that the students can have a smooth process for the same.
FR10	Company Search Engine	This component takes in a user query and returns the closest possible matches to the given user query. It’s intelligent information portal uses embeddings to accurately retrieve the information from the repository of companies stored in the database.
FR11	Alumni Connect	It is a blog service where alumni can add their experience about online round and interview round process. There’s a search feature to narrow down results.
FR12	Skill Refinement	It is a testing platform where students can hone their skills on different subjects. The testing is done on common interview questions which have been scraped from different sources. The feature also displays a performance chart to show his/her strengths and weaknesses.
FR13	Automail	This component is an automated mailing engine which sends out the appropriate form to the respected company on the click of a button. The form contains various details that the company can fill out, and its CSV responses are stored on Google Drive. This response is then uploaded onto the system, parsed into the correct JSON format and stored in the database for further use.

3.2 Non-Functional Requirements

NFR-1	Availability	The system will aim to achieve 99.9% availability using cloud services.
NFR-2	Maintainability	The architecture will be as simplified as possible to increase the maintainability of the system.
NFR-3	Performance	The product shall be web-based and has to be run from a web server. The product shall take initial load time depending on internet connection strength which also depends on the media from which the product is run. The performance shall depend upon hardware components of the client/customer. The system must support at least 1000 concurrent users at a time.

4 External Interface Requirements

4.1 User Interfaces

UI-1: Web application shall begin with a login page using Gmail account/USN and password. For new users, they will be redirected to a separate Signup Page for more information.

UI-2: Once logged in, web application will have a search functionality to aid in finding the details for a particular company.

UI-3: Web application will contain three separate navigation paths:

UI-4.1: A Company tab which displays a list of upcoming companies in decreasing order of date published.

UI-4.2: A Statistics tab which shows the general as well as company specific trends.

UI-4.3: A Profile tab which contains the information pertaining to the registered companies as well as seating allotment for the test.

UI-4.4: A Notifications tab highlighting the general updates for the students.

UI-4: For the administrator, there will exist a separate tab for uploading/updating new information of a company, as well as pre-placement and examination venue details.

4.2 Hardware and Software in terms of how they would interact or how they would be executed

4.2.1 Software Interfaces

SI-1: All students interact with the PCS to get all the relevant information like seating information, calendar of events, etc.

SI-2: Whenever a user chooses to register for a particular company, the system will communicate with the database to get all the relevant information for that user .

SI-3: The PCS shall communicate with the placement coordinator to verify the details of a particular company, to make sure if a particular detail needs to be made accessible to the students.

SI-4: The PCS will alert the placement officer if a particular candidate is already placed or if he has an internship or not.

SI-5: The system shall automatically filter students based on the eligibility criteria set by both the placement officer as well as the company.

SI-6: The system shall communicate with a database to display relevant details.

SI-7: The system shall communicate with the company for the final list of students and will help in coordinating any other additional procedure that needs to be completed for the offered students.(For example, collecting goodies, collecting the offer letter, filling any form and so on.)

4.2.2 Hardware Interfaces

Not applicable.

4.2.3 Communication Interfaces

CI-1: The system shall send a notification to the user as a reminder 24 hours prior to the events, including:

- Registration Deadline
- Pre-Placement Activities
- Placement Examinations

CI-2: In case of any changes, be it time or seating information, the user will be notified via a similar push notification.

5 Appendix A: Glossary

No glossary terms available at this time.

ARCHITECTURE AND DESIGN

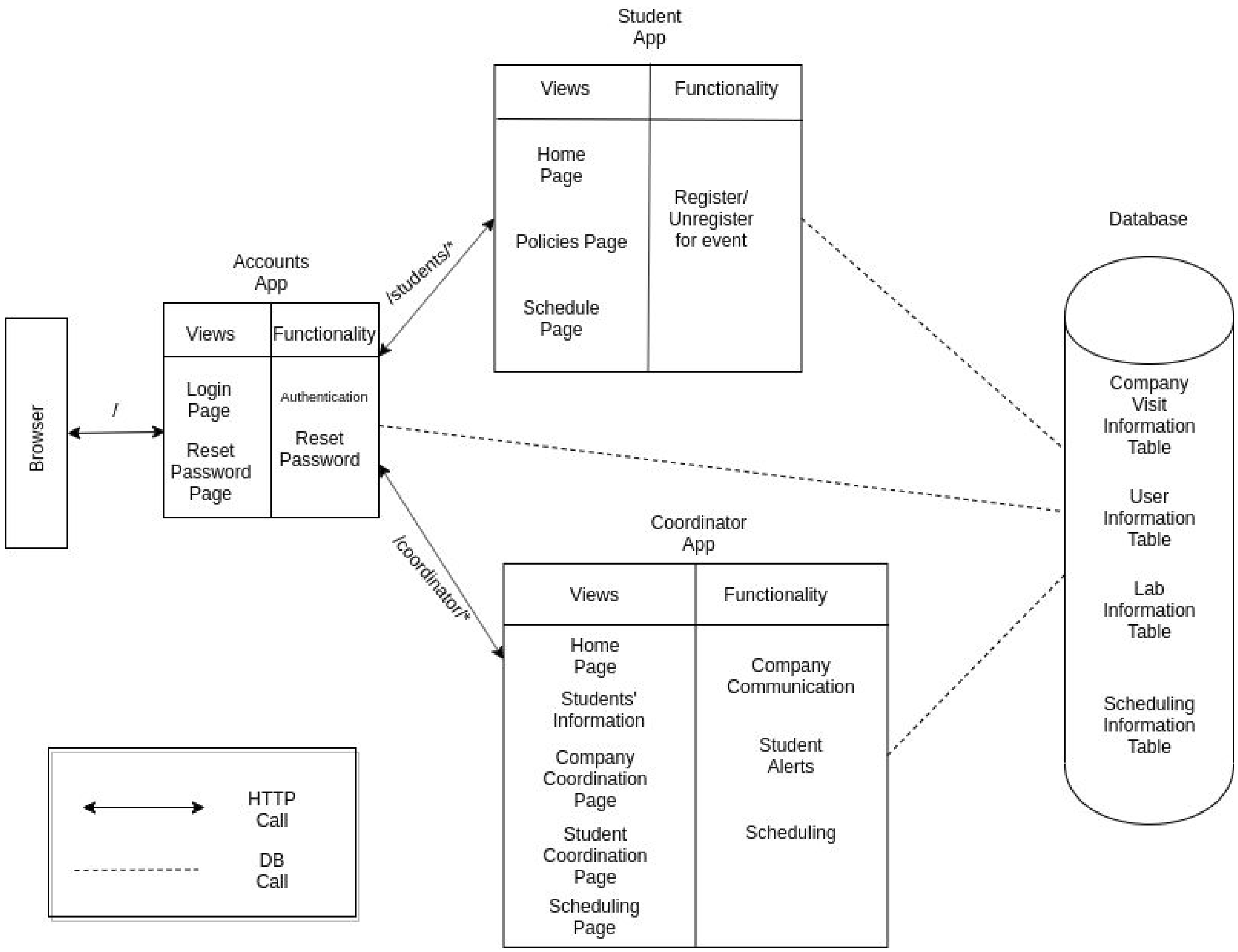
Contents

1.0 Architectural and component-level design

The main actors in the application are the students taking part in the placement process and the placement coordinator. The entire application is divided into 4 components, the Student Component, the Coordinator Component, the Accounts Component, and the Database Component. The division has been made based on the functionalities. The application, as seen by the placement coordinator, is completely different from that seen by the student, along with the different levels of access. Thus, the functionalities executed by the coordinator are different from that of the student, which motivates the division of the application into a Student Component and the Coordinator Component. Except for the Database Component, the components are accessible via rest calls, and the components also communicate with each other via rest calls.

2.0 System Structure

2.1 Architecture diagram



2.2 Description for Component

2.2.1 Browser- It is an application used to access and view the website.

2.2.2 Accounts Application - Contains the login page and password reset page to authenticate users and handle any exceptions by allowing the password to be reset. There are two views for this. One for the admin(placement coordinator) and the other for students. Based on the credentials, the app will redirect to the respective view(student app or coordinator app).

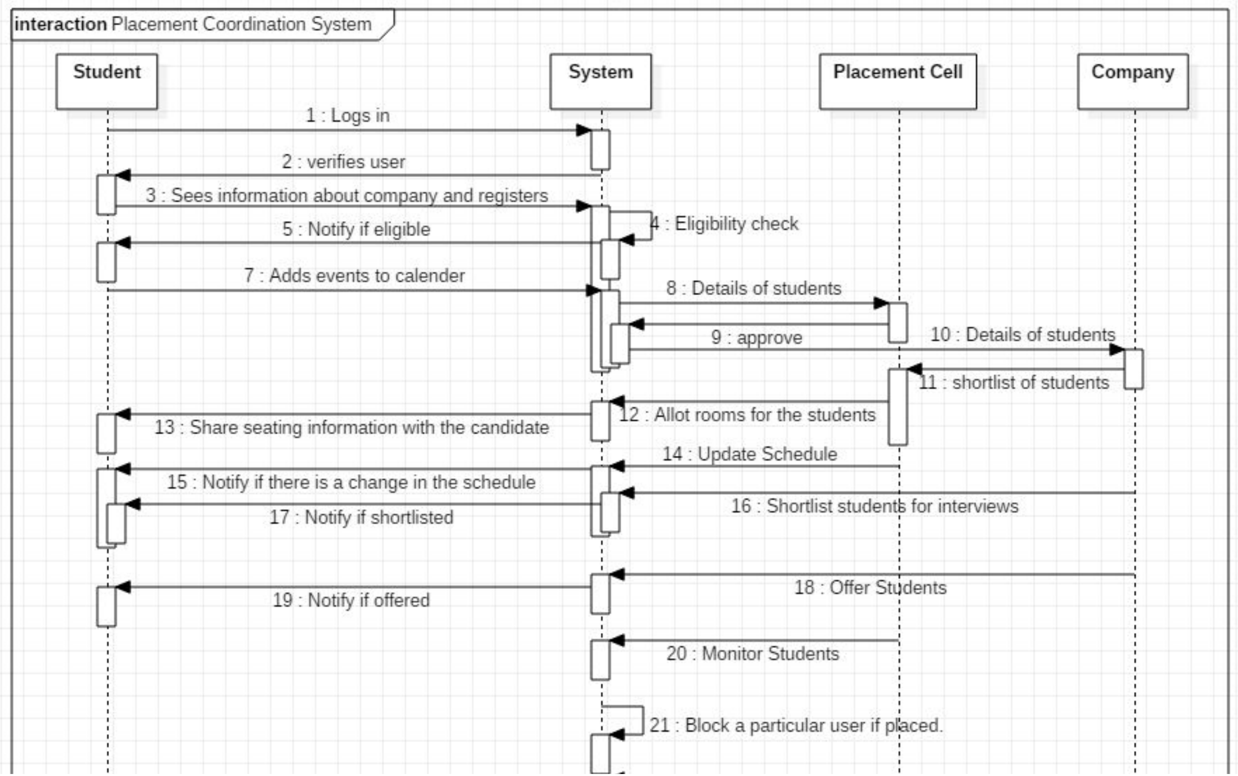
2.2.3 Student Application - The students application contains the student dashboard which displays student information, displays upcoming events, displays the schedule for a particular event. It also includes features like one click registration for students to register for a particular company. There is a feature for seeing placement statistics including the companies that have already come and any other general information that is relevant for a student. It will also have a calendar system to keep track of the events. There will be other features like student skill checker, alumni connect and company search engine where students can query appropriate terms.

2.2.4 Coordination Web Application- The coordination web application is for the placement coordinator where he/she can track the students who are part of the placement cycle. There is also a feature for the coordinator to schedule an event by querying the database for finding the desired timeslot. The coordinator can make a particular company details available for the students. The coordinator can share the student details with the company. There is a feature for automail where the placement office contacts the particular company to fill the details and as soon as the response is given, if it is uploaded to the software, data base is populated and corresponding details will be viewed in the students side.

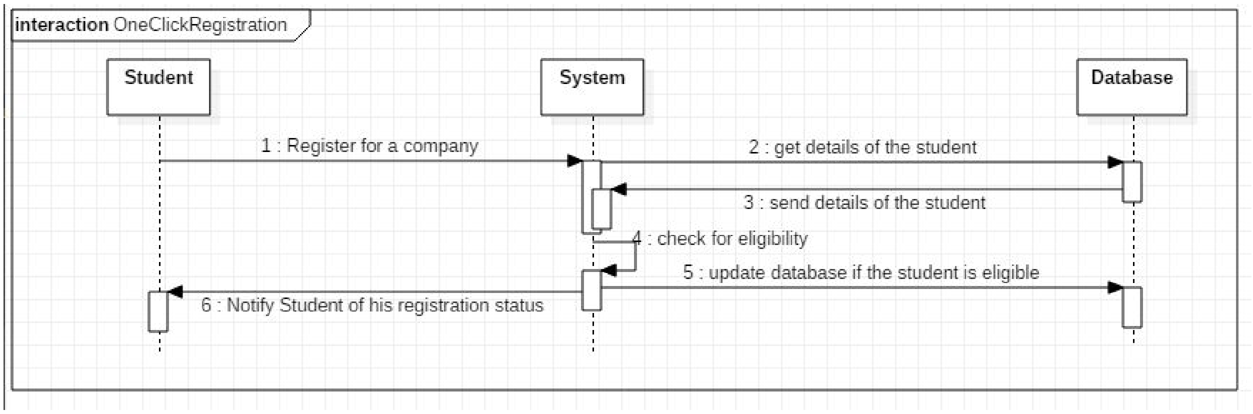
2.2.5 Database-Contains all information regarding students and users, companies,labs and seating, scheduling etc. All the other components will query this particular component to display and update data.

2.3 Interaction Diagram

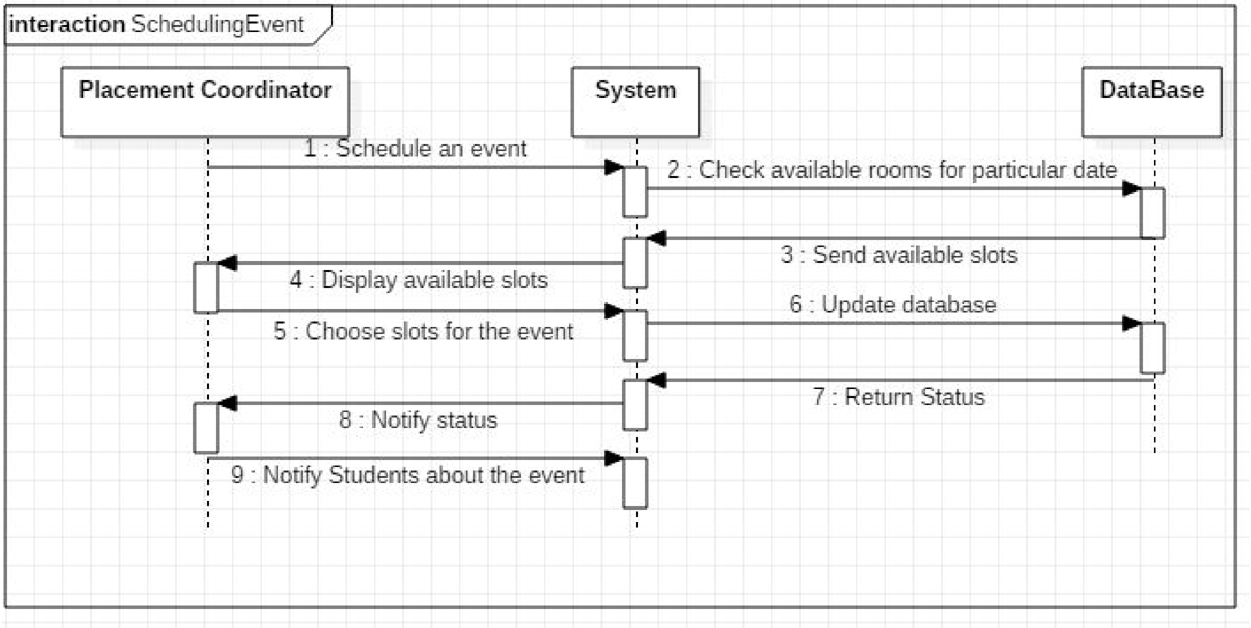
General Flow:


interaction Placement Coordination System
Student
System
Placement Cell
Company
1 : Logs in
2 : verifies user
3 : Sees information about company and registers
4 : Eligibility check
5 : Notify if eligible
7 : Adds events to calender
8 : Details of students
9 : approve
10 : Details of students
11 : shortlist of students
12 : Allot rooms for the students
13 : Share seating information with the candidate
14 : Update Schedule
15 : Notify if there is a change in the schedule
16 : Shortlist students for interviews
17 : Notify if shortlisted
18 : Offer Students
19 : Notify if offered
20 : Monitor Students
21 : Block a particular user if placed.

One Click Registration:


interaction OneClickRegistration
Student
System
Database
1 : Register for a company
2 : get details of the student
3 : send details of the student
4 : check for eligibility
5 : update database if the student is eligible
6 : Notify Student of his registration status

Scheduling event:


interaction SchedulingEvent
Placement Coordinator
System
DataBase
1 : Schedule an event
2 : Check available rooms for particular date
3 : Send available slots
4 : Display available slots
5 : Choose slots for the event
6 : Update database
7 : Return Status
8 : Notify status
9 : Notify Students about the event

2.4 Describe usage scenarios and how you would test that

Usage Scenario(STUDENT):

- **REGISTRATION FOR PLACEMENT CYCLE** : User performs a one-click registration for placements, and this action should be recorded in the database.

  A test could be run on the status check of the student on registering for placements, to see if the database action has been performed. This is followed by the uploading of a resume in a specified format. A test to check if the right format of the file has been uploaded will be done here.

- **ONE-CLICK REGISTRATION** : If user wishes to register, he/she can do so just by confirming the registration. Users details will be extracted from the database for further process. If user does not have an offer, registration will be successful or else registration will be unsuccessful.

  All the details will be obtained from the database and appropriate status codes will be checked for both success and failure.

- **NOTIFICATION** :
  - Placement officer will communicate with a company and alert students with the latest updates about the company.
  - Time, venue and seating allotment for online assessment will be sent to the user.
  - Shortlisted for the interview process will be posted.
  - Shortlisted user will be notified about the offer.

- **Alumni Connect** : It is a blog service where alumni can add their experience about online round and interview round process. There's a search feature to narrow down results.

- **Skill Refinement:** It is a testing platform where students can hone their skills on different subjects. The testing is done on common interview questions which have been scraped from different sources. The feature also displays a performance chart to show his/her strengths and weaknesses.

Usage Scenario(PLACEMENT OFFICER):

- **SCHEDULING** :

  schedule the written tests/interviews with respect to time and location. A test could be done to verify if the particular time slot is available in the respective computer labs.
- **AUTOMAIL :**

  This component is an automated mailing engine which sends out the appropriate form to the respected company on the click of a button. The form contains various details that the company can fill out, and its CSV responses are stored on Google Drive. This response is then uploaded onto the system, parsed into the correct JSON format and stored in the database for further use.

- **POST NOTIFICATIONS** :

- alert students with the latest updates about company.
- post about online assessment details.
- post about the results of online process and interview process.
- Using Postman, we will check the required status and response for a given api calls.

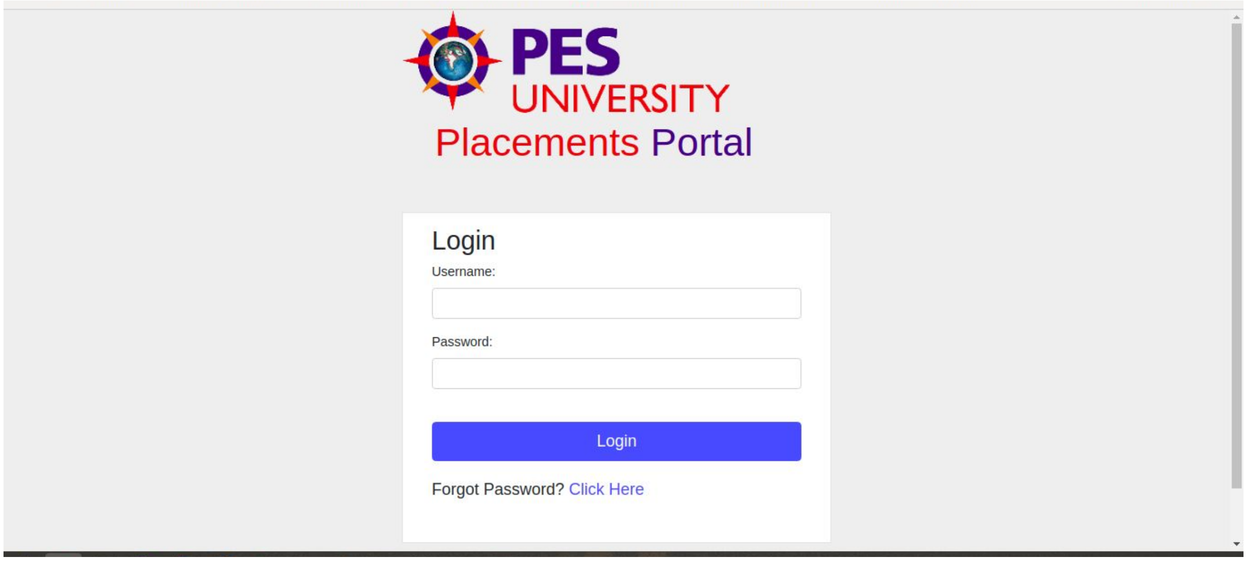
2.5 Architectural Styles and Patterns considered and for what reason

S.No	Architectural Styles/ Pattern	Intent of this pattern	Rationale for choosing or not choosing
1.	Repository Style	A repository architecture is a system that will allow several interfacing components to share the same data. Each component interfaces the same dataset that is utilized system wide. Data manipulation taking place in one component will reflect an identical representation of data in another component. Components can be interchanged and are independent of other system components.	The main intention for using repository style is because our application requires data to be generated by both students as well as placement cell. Students generate data by registering for companies and giving other details. The placement cell generates data by giving information about a particular event(company details and the schedule). These information are exchanged very frequently and thus communicate with the common database.
2	Model View Controller Architecture	The **Model-View-Controller** (MVC) is an **architectural** pattern that separates an application into three main logical components: the **model**, the **view**, and the **controller**. Each of these components are built to handle	Since we are using Django as our backend, our entire system is an MVC architecture.

		specific development aspects of an application.	
3.	Microservice Architecture	Microservices - also known as the microservice architecture - is an architectural style that structures an application as a collection of services that are highly maintainable and testable, loosely coupled and independently deployable.	Since the entire system will be used by students and placement coordinators of a college(right now one college), to ensure the system is highly scalable, available, a microservice architecture is used. The main aim is to make sure that a particular service does not choke the other services that are requested by other users(For example, if a particular seating information has been made available and many are accessing it, then it shouldn't avoid the placement coordinator to access or upload some company information.
4.	Pipes and filters	The Pipe and Filter is an architectural design pattern that allows for stream/asynchronous processing. In this pattern, there are many components, which are referred to as filters, and connectors between the filters that are called pipes. Each filter is responsible for applying a function to the given data; this is known as filtering. Filters can work asynchronously. The final output is given to the consumer, known as a sink.	Pipes and filters architecture is mainly used for large processes that can be broken down into multiple steps. If there needs to be more interaction between functions (the filters), this would not be a good architecture design since filters only communicate between each other for input/output. Since our system will have a lot of components that will be communicating with each other in terms of reading the database, displaying the content and updating the database, we will **not** be considering this architectural style.

3.0 User interface design

A couple of UI elements have been added in the document. These include:



PES UNIVERSITY

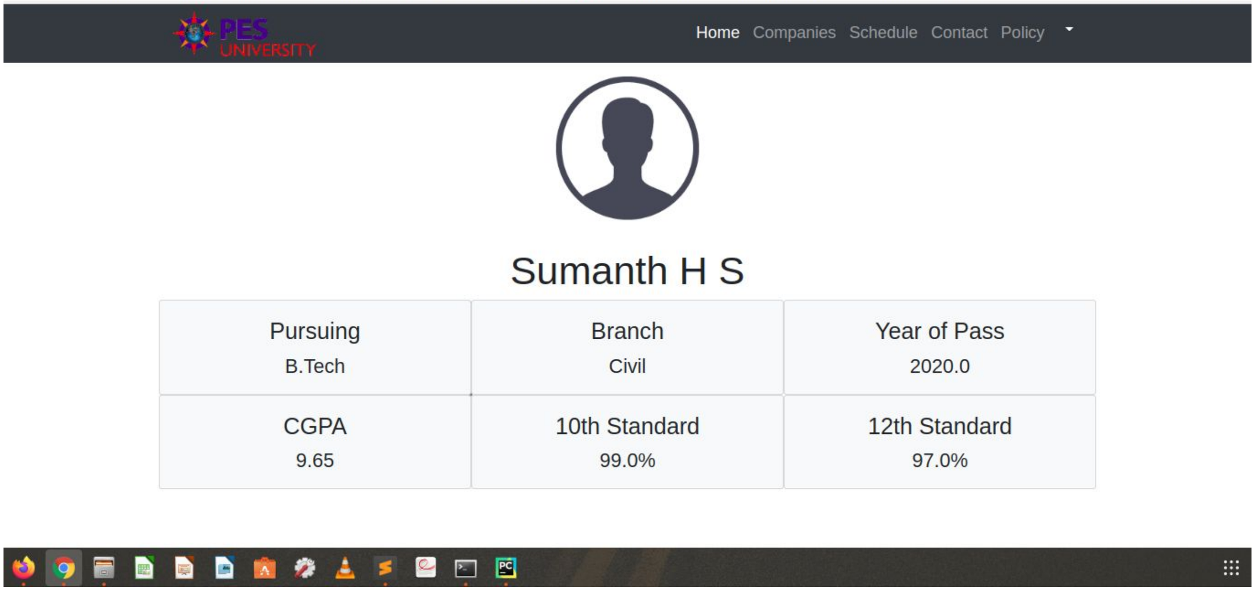
Placements Portal

Login

Username:

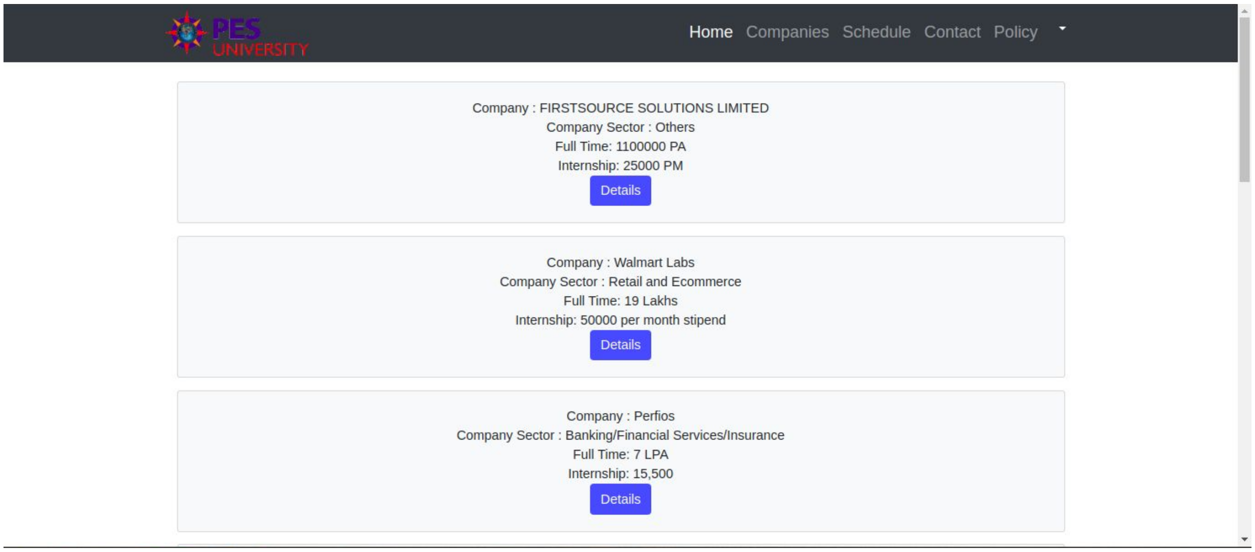
Password:

Forgot Password? Click Here



Sumanth H S

Pursuing B.Tech	Branch Civil	Year of Pass 2020.0
CGPA 9.65	10th Standard 99.0%	12th Standard 97.0%



Company : FIRSTSOURCE SOLUTIONS LIMITED
Company Sector : Others
Full Time: 1100000 PA
Internship: 25000 PM
Details

Company : Walmart Labs
Company Sector : Retail and Ecommerce
Full Time: 19 Lakhs
Internship: 50000 per month stipend
Details

Company : Perfios
Company Sector : Banking/Financial Services/Insurance
Full Time: 7 LPA
Internship: 15,500
Details

A Login Page - this takes in the USN and password from the user (student/coordinator) and renders the respective homepage.

A Home Page - this is a generic homepage whose layout will be common to both the student as well as the coordinator. The coordinator will have the added functionality to change various elements, including the scheduling details.

4.0 Detailed Design Approach

The project has 2 types of interfaces: Students and Coordinator.

There are backends for each interface, which interacts with the front end. This interaction is via a Model-View-Controller approach, where the front-end is the “View” (what the user sees), the back-end is the “Controller” (what orchestrates the work), and the Mongo database is the “Model” (what stores the information). Whenever a request comes in, it first goes to the Controller before it can be converted into instructions for the View or Model.

The design enabling techniques in the architecture and design of the project is centred around two primary concepts:

- Modularity: The enormity of the project dictates the imperative need for Modularity, which is the degree to which individual components may be separated and recombined. This will aid in flexibility as well as increase independence during the development stage.
- Information Hiding: In general, an abstraction of the interface from the business logic of the internal components is necessary to ensure consistency. The development of this project requires constant modifications to the backend logic, mainly for expansion to incorporate more students and labs, as well as bug fixes

and added functionalities. Ensuring appropriate information hiding is therefore a mandatory part of designing components for this project.

4.1 Design patterns considered and for what reason

S.No	Design Pattern	Intent of this pattern	Rationale for choosing or not choosing
1	Structural Design Pattern - Bridge	To separate an object's interface from its implementation	● Decouple an abstraction from its implementation so that the two can vary independently. ● Different hierarchies for the interface and implementation.
2	Private Class Data	To restrict accessor/mutator access	Students cannot access certain classes of the Coordinator
3	Creational Design Pattern - Object Pool	To avoid expensive acquisition and release of resources by recycling objects that are no longer in use	This design pattern is necessary as the Placement Coordination System's student base changes every year, which necessitates recycling of resources.
4	Creational Design Pattern - Prototype	To create a fully initialized instance to be copied or cloned	A Prototype of a student is created, which contains properties that are similar for each and every user.
5	Behavioral Design Pattern - State	To alter an object's behavior when its state changes	Once a student gets placed, his/her state needs to reflect that based on several conditions. This will determine his further actions.
6	Behavioral Design Pattern - Strategy	To encapsulate an algorithm inside a class	The scheduling algorithm is built inside the coordinator class for the allotment of lab seats.

TESTING

TABLE OF CONTENTS

1 INTRODUCTION

This test plan document documents specifies the strategy being followed for Testing of the project and sample of the test cases being written for the project

2 TESTING MODEL

During the development phase, we used the prevention model as our primary model for testing. The reason for using this was to prevent errors in the error by including all kinds of use-cases and corresponding error codes. Each of the module designed was planned accordingly and hence the prevention model was the most appropriate model for testing.

3 TESTING TYPES BEING USED

3.1 List all the Different Types of tests WHICH you plan to run

1. **Unit Testing:**
   a. Unit testing was done on individual components developed for any syntactic or semantic errors found in the code. Boundary conditions were tested and corresponding output scenarios were noticed.
   b. To make the code efficient, continuous optimization were done. Optimization were in terms of the data structures used for fast retrieval of information, and the algorithm used for faster computation.
2. **Integration Testing:**
   a. Since each of the scrum teams worked on different features, the system had to be integrated in an incremental process bottom up approach. This was helpful since each component could be integrated step by step.
   b. Dependency between components were minimal since all the apis written were independent of each other.
   c. Continuous testing was done in order to make sure that the system is working as expected.
3. **Grey Box Testing:**
   a. In order to take the advantages or both black box testing as well as white box testing, grey box testing was employed.
   b. For testing the semantics of the algorithms and data structures that were used, white box testing was used and in order to test if the corresponding functionality works as expected, black box testing was used.
   c. Hence this combination led us to use grey box testing.

3.2 List the Phases of the LifeCycle and the V&V done for each phase in the project

1. **Requirement Engineering phase:**

   a. During this phase, verification was done on whether any feature is being built in the right way.
   b. Code review and walk through was done in order to check if a particular requirements is meeting its criteria.
2. **Design Phase:**
   a. During the design, all the related components were kept in one place in order to achieve high cohesion and low coupling.
   b. Corresponding code review was done to see if the design meets the requirements or not
3. **Code phase:**
   a. The decisions taken during the requirements and the design phase was adhered during the code phase and individual functionality were written in a modular approach.
   b. Since we are using django as our backend, each and every functionality were written in a class structure with different http methods being the functions inside the class.
   c. Appropriate HTTP response codes were returned for different use cases and code review and code walkthrough was done for static analysis of the code.
4. **Testing phase:**
   a. Appropriate testing models mentioned above were followed in order to test the software.

3.3 Set out the Test adequacy criteria for your project

1. The criteria that were set for this project are as follows:
   a. Appropriate functionalities were found for the software. If the functionalities are working as expected, then testing phase can be stopped.
   b. The statement coverage adequacy criterion: This is satisfied by a particular test suite for a particular program if each executable statement in the program (i.e., excluding comments and declarations) is executed by at least one test case in the test suite.
   c. A fault-based adequacy criterion that seeds a certain set of faults would be satisfied if, for each of the seeded faults, there is a test case that passes for the original program but fails for the program with (only) that seeded fault.

4 TEST CASES

4.1 TEST CASE 1

- Input validation for coordinator scheduling API
- Authenticated coordinator account
- Make a GET request with the company visit ID. The test cases are:

    - GET request with valid company visit ID
    - GET request with invalid company visit ID
- POSTMAN tool to be used as the testing environment
    - Result of test case 1: 200 (OK) Response Code
    - Result of test case 2: 400 (Bad Request) Response Code

4.2 TEST CASE 2

- Removing a booking made by the coordinator
- The coordinator must be authenticated and the company visit ID must be valid
- Click on the "Cancel Schedule" button and check the schedule for the company after that
- Test Environment/Data to be used
- Two possible scenarios:
    - The success scenario is when the schedule gets removed for that specific company on the database, and on checking the schedule, we should see nothing booked by the coordinator
    - The failure scenario is when the booking is not removed from the databases and continues to show on checking schedule.

4.3 TEST CASE 3

- Writing a blog in the alumni connect feature
- User must be logged in
- Once clicked on the add post feature, corresponding blog entries are filled
- Logged in user makes a POST request to add a blog
- Blog gets displayed in the alumni connect page.

4.4 TEST CASE 4

- Registering for Placements
- User Must be logged in
- User clicks on register button

- Test data to be used is corresponding user data from the database.
- Appropriate message to be displayed if he has registered, already registered, or not eligible to register.

4.5 TEST CASE 5

- Automated mailing engine
- Coordinator must be logged on
- Coordinator either sends mail to a particular company or uploads the CSV response into the system
- Test Data to be used is the filled out Google Form downloaded as a CSV file
- Two possible scenarios:
  - In case of sending mail, success is when the company receives the mail from the placement office.
  - In case of uploading the CSV, success is when the CSV parsed JSON file is inserted into the company database

4.6 TEST CASE 5

- Skill Refinement
- User must be logged on
- User take the test for a particular subject
- Test Data to be used is the scraped interview questions JSON file, which is parsed into the system
- After the completion of the test, a successful scenario is when the user's profile gets updated with the new scores.

5 TEST RESULTS

CASE #	CTIVE/functionality	Kind	STATUS		N
1	COORDINATOR SCHEDULING AND BOOKING SLOTS	UNIT TEST / INPUT VALIDATION / WHITE BOX TESTING	SUCCESSFUL	NISHANT RAVI SHANKAR	17/11/2019
2	COORDINATOR CANCEL BOOKING AND SLOTS	CTIONALITY TEST / WHITE BOX TESTING	SUCCESSFUL	NISHANT RAVI SHANKAR	17/11/2019
3	ALUMNI CONNECT FEATURE	FUNCTIONALITY TEST / WHITE BOX TESTING	SUCCESSFUL	NINAAD R RAO	17/11/2019
4	ONE CLICK REGISTRATION	SYSTEM WIDE TESTING / FUNCTIONALITY TEST / WHITE BOX TESTING	SUCCESSFUL	NINAAD R RAO	17/11/2019
5.	AUTOMAIL	SYSTEM WIDE TESTING / FUNCTIONALITY TEST / WHITE BOX TESTING	SUCCESSFUL	MIDHUSH MANOHAR T.K.	17/11/2019
6.	SKILL CHECKER	UNIT TEST / INPUT VALIDATION / WHITE BOX TESTING	SUCCESSFUL	MIDHUSH MANOHAR T.K.	17/11/2019

6.0 Requirement Traceability Matrix

Sl. No	Req Id	Brief Desc	Architecture Ref Section	Design Ref	Code File Ref	Unit Test Cases	Function/ System test cases
1		Student Information	FR1	2.2.3	students/ views.py	5.4	-
2		Seating allotment for	FR2	2.2.3,2.2.4	students/ views.py and	5.1	5.1

		shortlisted students			coordinator/views.py		
3		Placement Statistics	FR3	2.2.3	students/views.py	5.4	-
4		Placement Policies	FR4	2.2.3	templates/homepage.html	5.4	-
5		Maintaining a calendar system for students	FR5	2.2.3	students/views.py	5.1	5.1
6		Company information and job search filter	FR6	2.2.3	students/views.py	-	-
8		One click registration	FR8	2.2.3	students/views.py	5.4	5.4
9		Scheduling Event	FR9	2.2.3,2.2.4	coordinator/views.py	5,1,5.4	5.1,5.4
10		Company Search Engine	FR10	2.2..3	students/views.py	-	-
11		Alumni Connect	FR11	2.2.3	students/views.py	5.3	-
12		Skill Refinement	FR12	2.2.3	students/views.py	5.6	
13		Automail	FR13	2.2.4	coordinator/views.py	5.5	5.5

REVIEW COMMENTS

FEASIBILITY STUDY- TEAM D1(Marketplace to connect donors to charitable organizations)

1. PROBLEM STATEMENT

Well written.

2. EXECUTIVE SUMMARY

Have explained clearly what is the problem they are trying to solve and a brief summary of what the document contains.

Have not mentioned about the blogging feature in both the executive summary or the problem statement. Its mentioned later on as a feature. Could have been added here.

3. CURRENT SYSTEMS AND PROCESSES

3.1 CURRENT OPERATIONS

Current operations well written. Three main competitors who perform similar task in a different way.

3.2 PHYSICAL ENVIRONMENT

Have explained the current software being used pretty well.

Little bit more analysis on the hardware side could have been done to overcome challenges that the team is going to face when they are going to work on the project.

3.3 USER ORGANISATION

Well written!

4. SYSTEM OBJECTIVES

4.1 DESCRIPTION OF PRODUCT AND SERVICES

Mentioned about the different functionalities that are going to be considered.

As an add-on, can you create a feature that lets users ask one of these charity organizations to post a particular cause(Since users wont be able to post any events?)

4.2 HIGH LEVEL BLOCK DIAGRAM

If the user is not verified does it mean he/she can create more than one account?

4.4 TARGETED CUSTOMERS AND BENEFITS

Very well written about their targeted customers and benefits.

Could have included a little more about users. Might be large companies that want to donate could have been listed?

4.5 TECHNOLOGY CONSIDERATIONS

Have mentioned all kinds of technologies that needs to be used in a very systematic way starting from front end, database and backend.

Why SQL?

The last few points the technology that is going to be used isnt clear. For example for calender system or the badge and rating system.

5. PRODUCT AND SERVICE MARKETPLACE

Clearly listed out why this problem is important i.e continual support.

The various organizations mentioned in 3.1 could have been added here and as a whole it could have been mentioned in the current operations.

6. MARKETING STRATEGY

Well written. Mentioned about social media, word of mouth, advertisements and many more

How is this platform different?

Could have mentioned this in the Product/Service Marketplace section to indicate what's new in your website.

Target groups :

Again could have mentioned this in the targetted customers and benefits section

7. Organisation and Staffing

The manual verification of partner organisation could be tedious. You can think of automating this similar to how KYC is done by many platforms such as Paytm, Amazon etc.

What about staffing for maintaining the website,database managers?

8. SCHEDULE

Very high level overview and few of them are deliverables that are already expected. Could have looked at it as when the database population will be completed. What all features will be done in a particular week etc.

9. FINANCIAL PROJECTIONS

Written about a possible way to earn money and where they will be spending money.

Costs for Webserver hosting could have been written in a tabular way on an yearly basis.

10. ISSUES

Well written!

11. Assumptions and Constraints

Listing this in the financial projection would be better.

11.5 Legislative and policy mandates or issues

Requirement Specification- D6(Website for NPO)

1. Introduction

1.1 Purpose and Intended Audience

Well Written!

1.2 Scope

Well written in terms of what will be done as a part of this project.

Not listed out what will not be done and will not be considered in the project.

1.3 Definitions, Acronyms, and Abbreviations

None.

Defining any kind of acronyms like NPO could be added here

2. General Description

2.1 Product Perspective

Well done.

2.2 Product Features

Difference between FE-3 and FE-6?

2.4 Operating Environment

OE-1 and OE-3 are non functional requirements. Reliability not mentioned in section 3.2. What type of database are you going to use?

2.5 Constraints

No comments

2.6 Assumptions and Dependencies

Why 500?

3.1 Functional Requirements

FR4- Do you mean OCR?

FR7- Calender services can be edited only by the admin?

3.2 Non-Functional Requirements

Well written

4 External Interface Requirements

4.1 User Interfaces

The UI can be more detailed with respect to the different tabs/sections that offer a particular functionality.

4.2 Hardware and Software in terms of how they would interact or how they would be executed

4.2.1 Software Interfaces

What about the calender service? Will it not have any db communication?

4.2.2 Hardware Interfaces

4.2.3 Communication Interfaces

No comments

Architecture and Design- Team D5(Conference Management System)

1.0 Architectural and component-level design

IT IS CLEARLY WRITTEN. (Furthermore, application tier consists of the script that performs the periodic scraping of websites, to keep the whole application up-to-date. : It is a very good idea)

2.0 System Structure

2.1 Architecture diagram

2.2 Description for Component

2.2.5 and 2.2.8 almost similar!!

2.2.2 where is it in the architecture diagram? (if it is the CRON job, it should be made explicit)

2.3 Interaction Diagrams

2.3.1 and 2.3.2 is clear (the application of a cronjob for automation is a good idea)

2.3.3 trigger based on? And isn't the cron job a loop based on seconds,minutes, hours, days, weeks, or months?

2.4 Describe usage scenarios and how you would test that

UI update functionality could be added and also about search filter function which is based on location, date and subject of the conference

They have talked about conference organizer whereas no usage functionalities about conference organizer has been mentioned.

How to test is WELL WRITTEN in a compact manner

2.5 Architectural Styles and Patterns considered

Neatly written and given example for choosing the style and pattern.

Could have added what is not going to be used just to get an idea of what makes the chosen styles or patterns better.

3.0 User interface design

USER INTERFACE DESIGN IS FINE.

The pictures selected for the home page might hinder the visibility of the text

4.0 Detailed Design Approach

Well written!!

One suggestion, with thousands of conferences, Object Pool - a creational design - pattern could have been considered for recycling of resources (not sure how out-of-date conferences are being handled currently)