

Software Requirements Specification

Training and Placement Cell Portal



**Indian Institute of Information Technology
Kalyani**

Prepared by:

Abhishek Kumar - 10114008

Shikha Verma - 10114011

Anupam Saha -10114012

Shubham Kumar Singh Rajput -10114047

Content

1. Introduction

- 1.1 Purpose
- 1.2 Scope
- 1.3 References
- 1.4 Overview

2. The Overall Description

- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 User Characteristics

3. Specific Requirements

- 3.1 External Interfaces
- 3.2 Functions
- 3.3 Performance Requirements
- 3.4 Logical Database Requirements
- 3.5 Design Constraints
- 3.6 Software System Attributes
 - 3.6.1 Reliability
 - 3.6.2 Availability
 - 3.6.3 Security
 - 3.6.4 Maintainability
 - 3.6.5 Portability
- 3.7 Organising the Specific Requirements
 - 3.7.1 System Mode
 - 3.7.2 User Class
 - 3.7.3 Objects
 - 3.7.4 Stimulus and Response
- 3.8 Additional Comments

4. Change Management Process

5. Supporting Information

1. Introduction

1.1 Purpose

This software requirements specification provides a description of all the functions and constraints of the placement management system, developed for our college's placement cell.

The placement management system is for the students and companies which maintains the database for the students where all the students' records are entered including their academic details. It will also manage the data of the company which would comprise of the profile of the company, eligibility criteria and the facilities or the package it provides etc.

The system would provide the facility of viewing the academic information of the student and the profile of the company; it would also search for eligible students and company and deal with the insertion and deletion of records.

1.2 Scope

The system would store all the academic as well as personal details of the students who wish to be placed and the companies who offer jobs to the students. The details of the companies as well as the students may be updated or modified or deleted to keep the information up to date.

Also notifications would be sent to the students about the companies i.e. details like the company profile; eligibility criteria for the job profile etc. also the information regarding the placement activities or procedure or a particular company i.e. the selection rounds or procedure.

1.3 References

- <https://en.wikipedia.org/wiki/RubyGems>
- <https://github.com/thoughtbot/paperclip>
- <https://github.com/codahale/bcrypt-ruby>
- <https://rubygems.org/gems/pg/versions/0.18.4>
- <https://github.com/puma/puma>

1.4 Overview

This project is based upon Ruby on Rails framework and uses PostgreSQL for database management. It facilitates the user and company to connect with each other online and thus reduces the hassle of paperwork.

2. The Overall Description

This project is to facilitate students in college, company to register and communicate with Placement Office. The users can easily access the data and it can be retrieved easily in no time.

2.1 Product Perspective

In various colleges, training and placement officers have to manage the students' profiles and the documents of students for their training and placement manually.

Also Placement Officers have to collect the information of various companies who want to recruit students and notify students time to time about the placements.

Placement officer also have to arrange profiles of students according to various streams and notify them according to company requirements. if any modifications or updates are required in the profile of the students or the company, it has to be searched and done manually.

Hence the placement management system would maintain a huge database for the complete details of the students as well as the companies in the placement process which would help to save time and effort.

2.2 Product Functions

The placement management system is to be developed as an attempt to take a record of companies and students by restricting a large database that would be used for each.

The system would provide the facility of viewing the academic information of the students and also the profile of the company. The eligible students can see the details of the company, placement procedure and other details.

2.3 User Characteristics

There are two users: Student and Company.

Company is any organisation which is interested in hiring the students for the vacancies in that company

Student are from the pre-final or final year batch, who are looking to go to industry and make use of the skills that they have learnt in past few years.

3. Specific Requirements

This web application can be deployed on Linux or Windows machine with Puma Server and PostgreSQL as Database.

- Minimum RAM 512MB
- 2GB Storage Space
- Intel Dual Core Processor
- Internet Connectivity with Ports configured

This application can be accessed by user through a machine having any web browser with html javascript support. The client devices must preferably have browsers like IE9 or above, Mozilla firefox (version 3.5 or above) or Opera 10 or chrome (version 29 or above) or safari installed in their OS. Specified versions are preferred to get HTML 5 output.

The portal can be accessed through a mobile or PDA with internet access and a web browser supporting html javascript output.

3.1 External Interfaces

We need internet connection that links to intranet as well as internet. So that students can access the portal on intranet itself.

3.2 Functions

User can register itself by providing his or her academic details and few personal details. After registering he or she can view all the jobs posted by the company and apply for the jobs that interest him or her.

Company's HR can register on the portal and post jobs along with its their eligibility criteria and exam process.

Both user and company can view the past records, which are updated at the end of the placement season.

3.3 Performance Requirements

The system must be interactive and the delays involved must be less .So in every action-response of the system, there are no immediate delays. In case of opening windows forms, of popping error messages and saving the settings or sessions there is delay much below 2 seconds, In case of opening databases, sorting questions and evaluation there are no delays and the operation is performed in less than 2 seconds for opening ,sorting, computing, posting > 95% of the files.

Also when connecting to the server the delay is based editing on the distance of the 2 systems and the configuration between them so there is high probability that there will be or not a successful connection in less than 20 seconds for sake of good communication.

3.4 Logical Database Requirements

The databases consists of six tables, namely companies, students, jobs, apply, exams and past_records.

- companies – It manages the records of the companies. Each company can edit its own details and also post jobs.
- students – It manages the records of all the students. Each student can modify his or her record at any point of tme.
- jobs – Whenever a company posts a job, the job gets updated in this table along with the company's id.
- apply – Whenever a student applies for a job posted by the company, this table get updated with the student id, job id.
- exams – It holds the details of several exams that are to be help for any job. It contains details like exam date, time, exam_type (online/offline).
- past_records – Whenever a placement season ends the database admin updates this table by entering the student_id, package, company_id, year, etc.

3.5 Design Constraints

- Each company can register only once.
- Only students who are appearing for the placement can register. Students of other year or other colleges cannot register.
- Company cannot delete a job, if any student has applied for the same.

3.6 Software System Attributes

3.6.1 Reliability

The site is built on Ruby on Rails and PostgreSQL which are widely popular, both of which are highly stable.

3.6.2 Availability

The website will be hosted on the college's sever, so, it can be access from anywhere, provided that one has internet facility and college's sever if working.

3.6.3 Security

The project has controller level and view level security.

- Controller Level Security – One cannot navigate to different location by changing the address. One student cannot view the details of other students.
- View Level Security – All parts of page are not view-able by everyone. On job page, company can only post job and the students can only apply. Even though both the functions of posting and applying are coded on the same page.

3.6.4 Maintainability

The site can be managed by system admin as he has access to entire database. Also, at the end of the session, he has to update the past records with student_id, company_id, package, year, etc.

3.6.5 Portability

The website can be hosted on any cloud server and thus is accessible anywhere, provided the server has Ruby on Rails and PostgreSQL installed and the client has HTML5 supported browser.

3.7 Organising the Specific Requirements

3.7.1 System Mode

The system works on the performance mode in which company can post the job and student can apply for the same.

3.7.2 User Class

The major User classes in the system would be:

- Student
New student needs to sign up or register giving complete details. They can update profile information. they can register for a particular company.
- Company
The company has to notify the admin or the placement officer. The company initially has to sign up. The company may shortlist the students who applied. They may use their details (academic as well as personal).

3.7.3 Objects

Attributes of Student

- Image
- Name
- Date of Birth
- Branch
- CGPA
- Subjects
- Languages
- Contact number
- Email id
- Address
- Registration number

Functions of Student

- Register
- Edit profile
- View available jobs
- View companies
- Apply for the job
- View exam details for the jobs applied

Attributes of Company

- Name
- HR Name
- Contact number
- Address
- Email id
- Website

Functions of Company

- Register
- Edit profile
- Post job
- Update job
- Post exam details
- Post results

3.7.4 Stimulus and Response

- Whenever a company posts a job, the job gets updated in this table along with the company's id.
- Whenever a student applies for a job posted by the company, this table get updated with the student id, job id.

3.8 Additional Comments

The project uses several validations:

- Client Side Validation
 - HTML 5 support validation like email id, contact number, website URL, etc
 - JavaScript validation is helpful when JavaScript is enabled in browser.
- Database Side Validation
 - PostgreSQL checks for the constraints and foreign key relations.

4. Change Management Process

The website has contact details of TP Cell Committee which can be contacted for any new feature request or any existing bug in the portal.

5. Supporting Information

Placement Cell, IIIT-KalyaniHOMEWhy IIIT-KalyaniCompany SectionStudent SectionContact



About Placement Cell

The Placement Cell at IIIT-Kalyani is a concept in its own form. It works professionally abiding the hierarchy of the placement officer followed by the placement secretary, the faculty members and finally the student representatives from each stream. The hierarchy aims at touching all the aspects required for the smooth functioning of placement body of the reputed Institution IIIT-Kalyani.

Contact Us

Indian Institute of Information Technology Kalyani,
Webel IT Park, Kalyani (741235),
West Bengal (India)

Abhishek Kumar	(M) +91 9999999999
Anupam Saha	(M) +91 9999999999
Shikha Verma	(M) +91 9999999999
Shubham Kumar	(O) +91 9999999999

External Links

- * Placement Brochure
- * Facebook

Home Page

Company Registration

Company Name	<input type="text" value="Enter Company Name"/>
HR Name	<input type="text" value="Enter HR Name"/>
Contact Number	<input type="text" value="Enter Contact Number"/>
Company Address	<input type="text" value="Enter Address"/>
Website URL	<input type="text" value="Enter Website URL"/>
Email	<input type="text" value="Enter Email"/>
Password	<input type="text" value="Enter password"/>
<input type="button" value="Sign up"/>	

Company's Registration Page

New Student

Image	<input type="button" value="Browse..."/> No file selected.
Student Name	<input type="text" value="Enter Student Name"/>
RegistrationNo	<input type="text" value="Enter RegistrationNo"/>
Date of Birth	<input type="text" value="2017"/> <input type="text" value="May"/> <input type="text" value="9"/>
Stream	<input type="text" value="Enter Branch"/>
CGPA	<input type="text" value="Enter CGPA"/>
Subjects	<input type="text" value="Enter Subjects"/>
Language	<input type="text" value="Enter Languages"/>

Student's Registration Page