# https://pbs.twimg.com/profile_images/525654909123710977/nwJ7ttFX.png

Bilkent University

Department of Computer Engineering

CS-319 Project

Project short-name: TrackIn

Analysis Report

Muammer TAN, Ali Semi YENİMOL, I. Kayacan KAYA, Eniselda TUSKU

Table of Contents

[1 Introduction 3](#_Toc495168155)

[2 Current System 3](#_Toc495168156)

[3 Proposed System 4](#_Toc495168157)

[1.1 Overview 4](#_Toc495168158)

[1.2 Functional Requirements 4](#_Toc495168159)

[1.3 Non-functional Requirements 7](#_Toc495168160)

[1.4 Pseudo Requirements 8](#_Toc495168161)

[4 System Models 8](#_Toc495168162)

[1.1.1 Scenarios 8](#_Toc495168163)

[1.1.2 Use-Case Model 9](#_Toc495168164)

[1.1.3 Object and Class Model 17](#_Toc495168165)

[1.1.4 User Interface 18](#_Toc495168166)

[5 References 27](#_Toc495168167)

# Introduction

Finding an internship according to a student’s profile is a very difficult process. However even more challenging is what happens after the student has been accepted for an internship in a specific company. The whole communication process between the student and the company, assignment and evaluation of the interns work together with several bureaucratic issues, is very inefficient and time consuming. Some big companies have a very large number of interns and it becomes particularly hard to track them and fulfill all of their needs. Both the students and the companies lose time during this process because currently it is possible to do some operations using HR software but there is not a specific software to follow the process and operations between the company and school of interns. TrackIn is a general web-based app that severs precisely to the purpose of facilitating the communication process between the interns and the company. This application is very helpful to the companies as it tracks the interns work and performance throughout the whole internship, but also to the students as it facilitates their bureaucratic necessities and keeps them updated with their work.

# Current System

In the current system, tracking the assignments of interns and communicating with all of them for even the most basic things is very difficult for big companies since there is no electronic platform specialized for this purpose, and each company uses its own system to perform this task. The current system is not efficient and very time consuming for all of the users involved.

**Supervisors:**

* Spend a large amount of time assigning tasks to interns or constantly checking their work.

**Companies:**

* Store data and documentation of interns in different platforms which is very hard to deal with.
* Face challenges in providing the interns only with limited information and access so the privacy of the company and of the current projects remains intact.

**Students**:

* Face difficulties to access the supervisor each time he completes and assignment and requires a new one.
* Obtaining certain documentation required for his university or his future applications such as acceptance letter or internship

# Proposed System

## Overview

TrackIn is a web based application that provides companies, interns and supervisors with an easier way to manage the internship procedure. Before internship starts, company matches the intern with an available supervisor who will follow the intern’s entire process in the company. Interns will be able to get tasks from supervisors, point out their completed assignments online and retrieve all the necessary documents which will be provided to him by the company. Supervisors can assign tasks to interns and when they are notified that the given task is done, they can go in order to check and later approve the given task on the system. Also, interns and supervisors will be able to make a scheduler, note their plans, jobs and meetings on calendar. Companies will have the ability of storing files and documents of intern in the intern's profile page. Company can announce any event to the interns such as meeting, seminar, dinner, party etc. on the events page. After a certain time completing his tasks, the intern is assigned a checkpoint, which will be to complete weekly parts of internship report. After the intern has completed the report in the end of his internship he will upload it to the system, form where the company sends it to the department secretary of the intern, so the student will be excluded from the exhausting delivering report stage.

This program would be very useful to companies as several of them stated when we interviewed them in order to see their response to using such a program. They said that it would help them to keep track of their interns work but also to facilitate all the services that the company and the interns could require from one another. In addition the fact that each company creates its own requirements for the intern was also very appalling to the companies interviewed.

## Functional Requirements

When using the program, the user will have the possibility to sign up as a student, supervisor or a company and will have its profile accordingly. As a student the profile will firstly include all the basic information and it will be independent of the internship selection of the student. This means that even after completing an internship he will still have an active profile that he can use in order to be enrolled in different companies.

When the student has already been accepted for an internship in a specific company he will select: “Join a company” and will insert there his personalized username and password provided by the company. After this he has to complete additional information that is required by the company (different companies may have different requests) while the information that already existed in his personal profile will be automatically integrated. In this way a student’s profile inside a specific company would be obtained. This profile will include:

* **Acceptance letter**: Immediately after being enrolled in the system the student will have access to an acceptance letter provided to him by the company which he may use for academic purposes.
* **List of tasks.** During the internship the student will be assigned different tasks and he will be able to see them detailed in his profile.
* **Scheduler -** The student will have a scheduler where he can see his deadlines together with other events in the company.
* **General Events: -**  The students will be able to see in details the events that occur in the company in which they may have interest to be involved in.

According to the company policies for the internships the students may not be assigned an individual account but integrated in a team profile if the internship is organized in that way. Even in that case the profile will be organized in the same way with the only difference that the student will also be able to see the information of his team members.

If the user signs up as a supervisor he will have to insert all his personal information together with his department and role in the company. His profile will include:

* **The list of all the interns that he has under supervision:** He will be able to see and access the profile of all his interns together with additional information like the starting and ending date of the student’s internship and the status of the intern such as continuous.
* **Creating and assigning individuals to a team.**  If the company organizes the internships in teams than it is and attribute of the supervisor to create those teams and to assign specific individuals to them.
* **Add and approve tasks.** Every intern or team of interns will have their tasks throughout the internship. These tasks are assigned by the supervisor together with their deadlines. When the student completes his task he notifies his superior who after confirming that the task is completed marks it as approved in the system.
* **Scheduler:** The supervisor will also have his scheduler where he can see his deadlines and other different events that happen in the company.
* **List of tasks:** The supervisor will also have his list of tasks that are assigned by the company. For example he will put check points as a task to the students so they can report the work done throughout the internship. When the students have a final report he has to send it to the company so that they can send the reports of the students who need this service to their department secretary. In this way the process of writing and delivering a report would be more efficient and more transparent.

If the program is accessed with a company profile all the necessary information related to the company must be completed. This profile will include the following elements:

* **The total number of interns:**  The company will have access to the total number of interns together with their specific departments and their supervisors.
* **The total number of supervisors:** The information on the supervisors will be also provided.
* **Defining the student profile:** The company will decide on the additional information that the student must insert when he is enrolled in the internship program. The information may be different for each company.
* **Matching interns with supervisors**: Taking in consideration the profile of the student the company will assign his department together with the specific supervisor that will be observing the intern’s performance.
* **Check supervisor’s tasks :** The company will also be responsible to check the supervisors tasks and make super he is fulfilling all his assigned duties.
* **Provide communication with the university:** In the case of students who have internships as part of their curriculum the company may send to the university the intern’s report which was delegated to them by the supervisor. In addition to this they may respond to some questions that the university may have related to the performance of the intern.
* **Provide documentation:** The company will also provide documents that show the period when the student completed his internships so that the student can use it in his future academic life.
* **Events –** The company may also create different events that are visible form the students and the supervisors.

## Non-functional Requirements

* These requirements are going to be based on the response time in which it is shaped by the expectations of the student, supervisor and company users.
* The response time of the updates of the tasks given to the student by the supervisor must be less than 1 second so that the tasks given by the supervisor will be immediately seen by the student’s interface.
* In addition to the tasks, the response time of the student’s responses to the tasks must also be less than 1 second in order the supervisor to immediately analyze whether the task is accomplished or not. These two optimization issues are about how fast we are retrieving from and writing to the database.
* In addition to the relationship of the student and supervisor users, the response time of the company to which it matches the student and the supervisor, gives tasks to the supervisor and creates general events for both students and the supervisors must be less than 1 second for each feature. By these limits, the aim is to deliver a real-time database which does the requested jobs in small amounts of time in order to give the users best experience.
* Apart from the coding part, there is another responsibility for the supervisor who should go and check some accomplished tasks -there are several tasks that the supervisor needs to check face to face- of the students. It is supervisor’s job to make this procedure faster therefore every planned requirements are going on well with the expectations of any kind of users.
* The system must support the number of students and the supervisors which is specified by the company.
* Any number of users must be able to use the system unless they are not logged in.
* The input types of the users must be determined by the system itself in order to avoid the users making errors.
* The size of the database must be big enough to hold the data needed for every student, supervisor and the company.
* If an input error occurred, system should inform the current user about that situation and maintain itself to its previous state in which no error has just occurred.
* In a situation of system crash, no data must be lost.

## Pseudo Requirements

The implementation language is Java. As it is going to be a web application, HTML, CSS, and JavaScript programming languages are going to be used.

# System Models

### Scenarios

* Bob manages to be accepted as an intern in Pear Inc. Then, Pear sends him an e-mail containing a unique key which will have him access to TrackIn system of the company.
* Bob registers TrackIn system, and creates his own profile. After his profile has been created, he has been asked to enter the unique key he got from the company in order to enroll Pear Inc’s internship tracking system and he will provide the additional information required by the company.
* Bob is officially registered as an intern in Pear Inc. Now Bob has no supervisor as he has just been enrolled to the system, so the user of the system who has been logged in as Pear Inc. will assign him an available supervisor, Alice.
* Alice who signed in as a supervior of Pear Inc. will supervise the whole process of Bob’s internship. Supervision of Bob starts with assignment of different cumulative tasks together with the deadlines and information on each. After the assignment, Bob starts working with his first task and after completing it, he notifies Alice.
* Alice receives the notification about the finished task of Bob, she either personally or through the system -changes according to the task’s nature- checks Bob’s work done and marks the status of the task as completed, in progress or in need for review according to her feedback.
* If the task is completed, then Bob carries on his next task, and this process repeats until all the tasks are finished. Between a certain number of tasks that is determined by Alice, Bob needs to deliver a report on what he has been working so far to Alice.
* After all the tasks and written reports are completed, Alice checks the final report delivered by Bob. If she approves the report, it is sent to Pear Inc. user who will deliver it to the school by mailing or e-mailing.
* If Bob wants, he can get his acceptance letter available in his profile since the start of the internship and an evidence paper released at the end of his internship by the company which indicates that he has completed his internship in the specified time period in Pear Inc.
* In addition, Bob, Alice and Pear Inc’s user will have schedule which will contain the deadlines for their respective tasks and for general events happening in the company.
* The general events are going to be managed by the Pear Inc’s user who can add, remove and promote an event happening in the company.

### Use Case Diagram1.jpgUse-Case Model

|  |  |
| --- | --- |
| Use case name | Request Information |
| Participating actors | Initiated by Company  Communicates with Student |
| Flow of events | 1-Company makes an Information Request to the TrackIn with specifying the student and type of the request.  2-TrackIn receives the request and create a task on students profile as a form.  3-Student receives the request and provide the information.  4-Company gets the provided information. |
| Entry condition | Company is logged into TrackIn |
| Exit condition | Student has provided the information |
| Quality Requirements | - |
| Use case name | Request a Photo, Image |
| Participating actors | Inherited from Request Information use case |
| Flow of events | 1-Company specifies the size and description of the image or photo request  2-TrackIn receives the request and create a task on student’s profile in compliance with photo uploading.  3-Student receives the request and provide the photo or image.  4-Company gets the provided photo or image. |
| Entry condition | Inherited from Request Information use case |
| Exit condition | Inherited from Request Information use case |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Request a File |
| Participating actors | Inherited from Request Information use case |
| Flow of events | 1-Company specifies description of the file  2-TrackIn receives the request and create a task on student’s profile in compliance with file uploading.  3-Student receives the request and provide the file.  4-Company gets the provided file. |
| Entry condition | Inherited from Request Information use case |
| Exit condition | Inherited from Request Information use case |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Request a Document |
| Participating actors | Inherited from Request Information use case |
| Flow of events | 1-Company specifies description of the document  2-TrackIn receives the request and create a task on student’s profile in compliance with document uploading.  3-Student receives the request and provide the document.  4-Company gets the provided document. |
| Entry condition | Inherited from Request Information use case |
| Exit condition | Inherited from Request Information use case |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Assign Task |
| Participating actors | Initiated by Supervisor  Communicates with Student |
| Flow of events | 1-The supervisor makes an Assign Task to the TrackIn specifying the tasks that the student should complete throughout his internship.  2-TrackIn receives the request and displays the list of tasks in the Student Profile.  3-Student receives the list of tasks he has to complete throughout the internship. |
| Entry condition | Supervisor is logged into TrackIn |
| Exit condition | Student has received his assigned tasks. |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Approve a completed task |
| Participating actors | Inherited from Assign Task use case |
| Flow of events | 1-After completing one of his tasks the Student notifies the Supervisor that a task is completed.  2- The supervisor receives the notification and with means outside the system (personally for example) checks the completeness of the task.  3- According to task completeness he changes the state of the task as: COMPLETED  4-The student sees the change and can continue to his next task. |
| Entry condition | Inherited from Request Information use case |
| Exit condition | Inherited from Request Information use case |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Update the Schedule with dates and deadlines |
| Participating actors | Initiated by Company, Supervisor  Communicates Student |
| Flow of events | 1-Company may insert all the necessary events in the Scheduler and they will be visible to interns and supervisors.  2-Supervisors can make changes to their Schedule adding their deadlines and also events that will be visible to the interns.  3-Students will see the deadlines and the events that he has access to in his scheduler. |
| Entry condition | Company and Supervisor are logged into TrackIn |
| Exit condition | The necessary information is visible in all the Schedulers. |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Login |
| Participating actors | Initiated by Intern, Supervisor or Company |
| Flow of events | Any of the users enter their username and passwords and press the login button.  TrackIn recieves the information given by user and checks if there is any user with given username and password. If not, error is given user by TrackIn.  If there is a user with given username and password, TrackIn redirects the user to homepage. |
| Entry condition | Company, Supervisor and Student are logged into TrackIn |
| Exit condition | User has entered valid username and password. |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Match Supervisors and Interns |
| Participating actors | Initiated by Company  Communicated by Intern and Supervisor |
| Flow of events | When a Intern included into the system, Intern has to match with a supervisor.  Company can see the free Interns and available Supervisors at homepage.  Company selects a Supervisor and a Intern and clicks match button. |
| Entry condition | Company is logged into TrackIn  At least 1 available Supervisor and 1 free Intern should be in TrackIn |
| Quality Requirements |  |

|  |  |
| --- | --- |
| Use case name | Create Events |
| Participating actors | Initiated by Company, communicated by Intern |
| Flow of events | Company adds any event that may be interested by interns  Added events will be shown on Intern -> Events page. |
| Entry condition | Company is logged into TrackIn |
| Exit condition | Cancel Button, OR  Company successfully adds event. |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Request Information |
| Participating actors | Initiated by Company  Communicates with Student |
| Flow of events | 1-Company makes an Information Request to the TrackIn with specifying the student and type of the request.  2-TrackIn receives the request and create a task on students profile as a form.  3-Student receives the request and provide the information.  4-Company gets the provided information. |
| Entry condition | Company is logged into TrackIn |
| Exit condition | Student has provided the information |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Request a Photo, Image |
| Participating actors | Inherited from Request Information use case |
| Flow of events | 1-Company specifies the size and description of the image or photo request  2-TrackIn receives the request and create a task on student’s profile in compliance with photo uploading.  3-Student receives the request and provide the photo or image.  4-Company gets the provided photo or image. |
| Entry condition | Inherited from Request Information use case |
| Exit condition | Inherited from Request Information use case |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Request a File |
| Participating actors | Inherited from Request Information use case |
| Flow of events | 1-Company specifies description of the file  2-TrackIn receives the request and create a task on student’s profile in compliance with file uploading.  3-Student receives the request and provide the file.  4-Company gets the provided file. |
| Entry condition | Inherited from Request Information use case |
| Exit condition | Inherited from Request Information use case |
| Quality Requirements | - |

|  |  |  |
| --- | --- | --- |
| Use case name | Request a Document | |
| Participating actors | Inherited from Request Information use case | |
| Flow of events | 1-Company specifies description of the document  2-TrackIn receives the request and create a task on student’s profile in compliance with document uploading.  3-Student receives the request and provide the document.  4-Company gets the provided document. | |
| Entry condition | Inherited from Request Information use case | |
| Exit condition | Inherited from Request Information use case | |
| Quality Requirements | - | |
| Use case name | | Enroll to a Company for Intern | |
| Participating actors | | Initiated by Intern, Communicated by Company | |
| Flow of events | | 1- Company communicates to the student via e-mail by sending a unique key.  3- TrackIn receives the unique code and checks its correctness.  2- Student enrolls to a company by means of loggig in by the unique key. | |
| Entry condition | | Student used unique key to enroll to a company | |
| Exit condition | | Student is enrolled to the company. | |
| Quality Requirements | | - | |

|  |  |
| --- | --- |
| Use case name | Provide Documents |
| Participating actors | Initiated by company, Communicated by Intern |
| Flow of events | 1-Company specifies description of the document  2-Company asks for the documents to the intern  3-Intern receives the request and provide the document.  4-Company gets the provided document. |
| Entry condition | Student is asked to provide the document |
| Exit condition | Document is provided |
| Quality Requirements | - |

|  |  |
| --- | --- |
| Use case name | Mark a task as Completed |
| Participating actors | Initiated by Student, Communicated by Supervisor |
| Flow of events | 1-Student finishes the task given by the supervisor.  2-Student is asking for the checking of his/her task's completeness.  3-Supervisor checks the task and marks it as completed. |
| Entry condition | Task is completed by student and notification is sent to the supervisor |
| Exit condition | Supervisor marked the task as completed |
| Quality Requirements | - |

### Object Diagram1.jpgObject Model

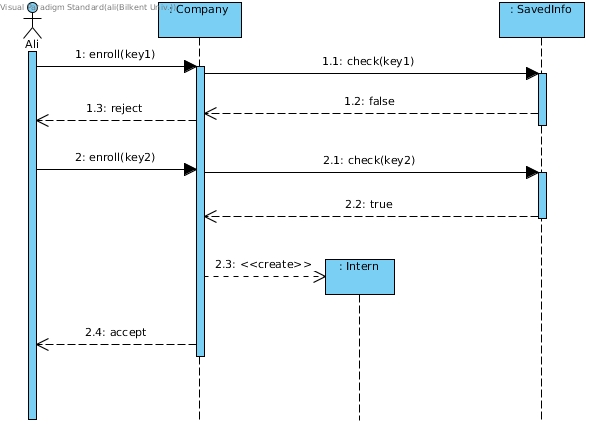
### Dynamic Model

Sequence Diagrams

**1- Enroll to a Company Sample Use Case Scenario:**

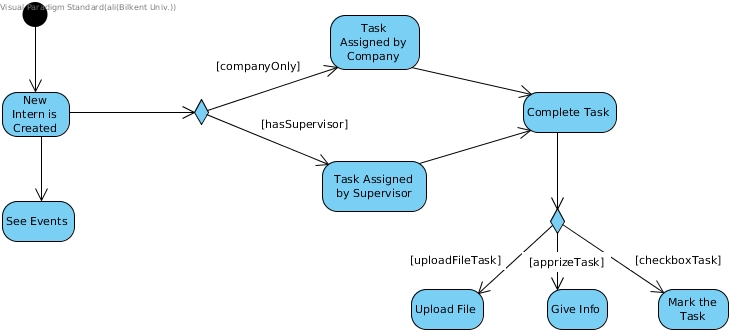
Ali who is a third year computer engineering student is accepted to the Jintel Company. Jintel company uses TrackIn program to track their interns and their tasks. Thus, they said Ali that he should sign up to TrackIn and they send him a enrolling key. After Ali registers to the TrackIn System, he clicks the “Join your company!” button and TrackIn asks for the enrollment key. Ali enters the key but he enters a wrong key(key1) and system shows a warning that he can’t enroll a company with the given key. Ali realizes that he entered the wrong key and he carefully enters the correct key again by clicking the “Join your company!” button. Then he is accepted to the company and see his profile page in the company.

Sequence Diagram for this sample use case is shown below:



Activity Diagrams

Activity Diagram for Intern Object:



### User Interface

NAVIGATIONAL PATHS

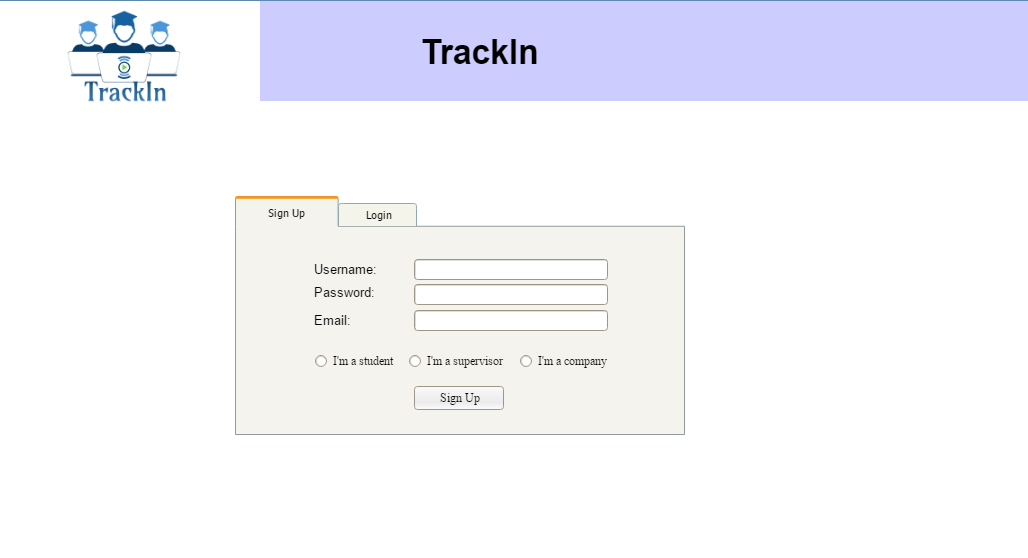


Image - Sign Up Page

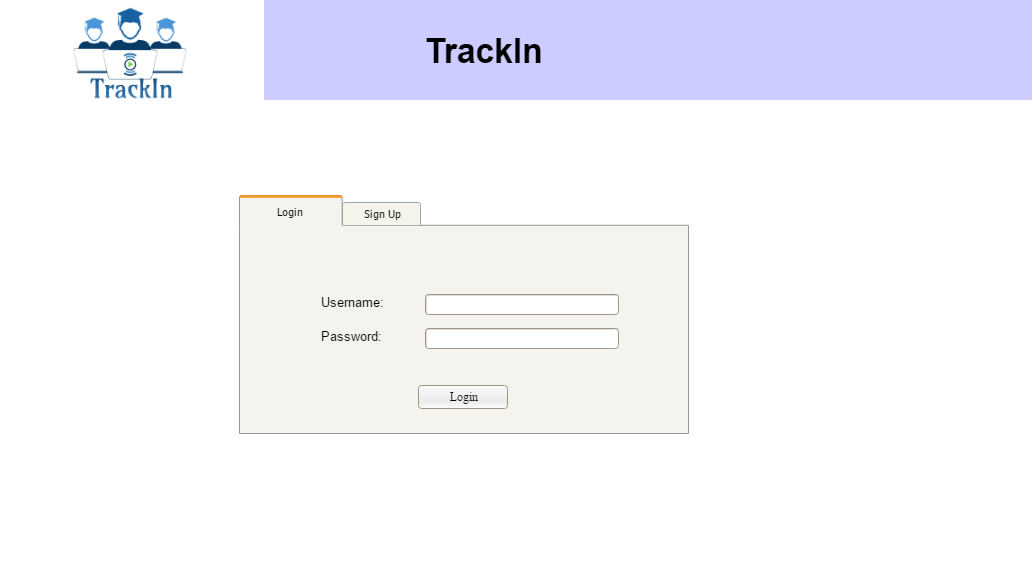


Image - Login Page

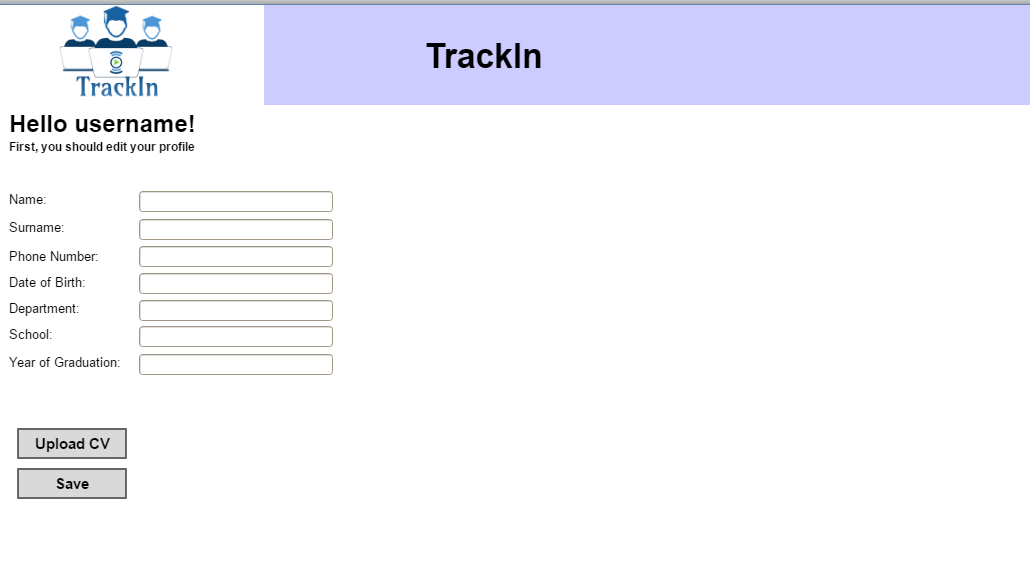


Image - Intern First Sign Up Page

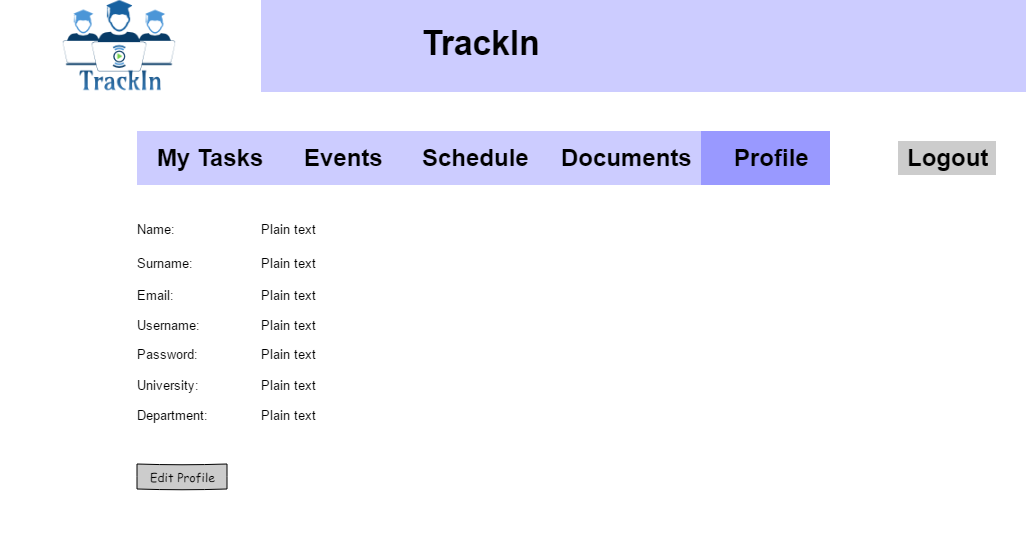


Image - Intern Profile Page

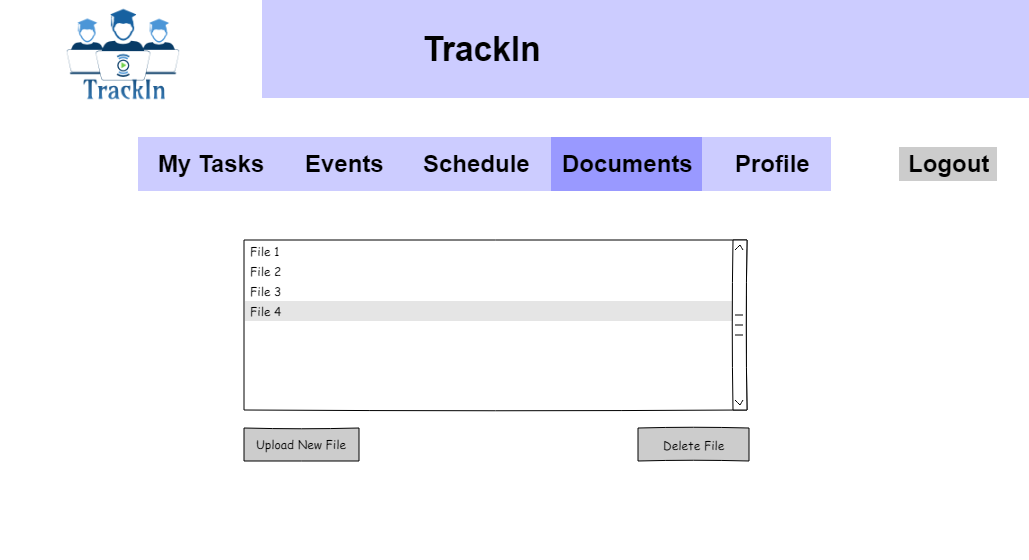


Image - Intern Documents Page

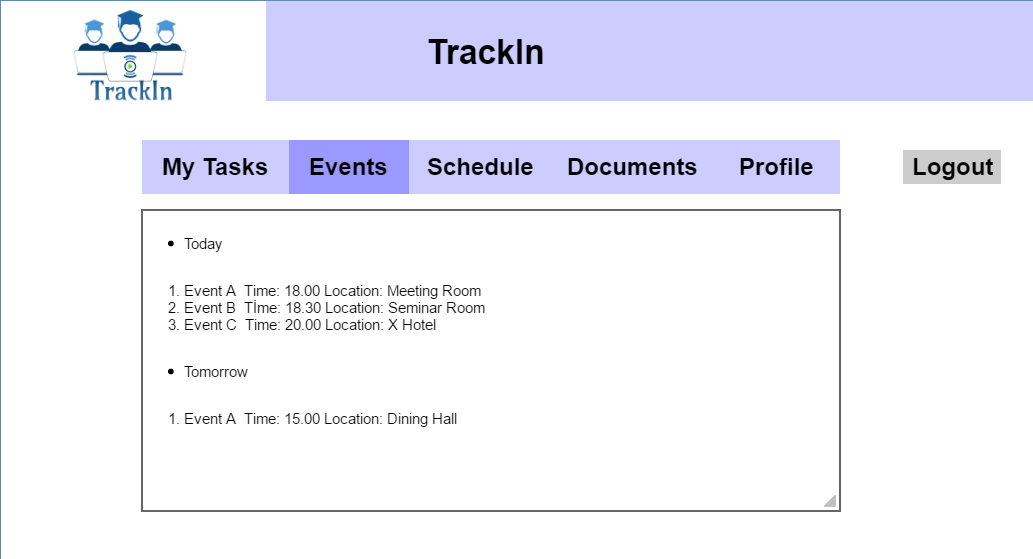


Image - Intern Events Page

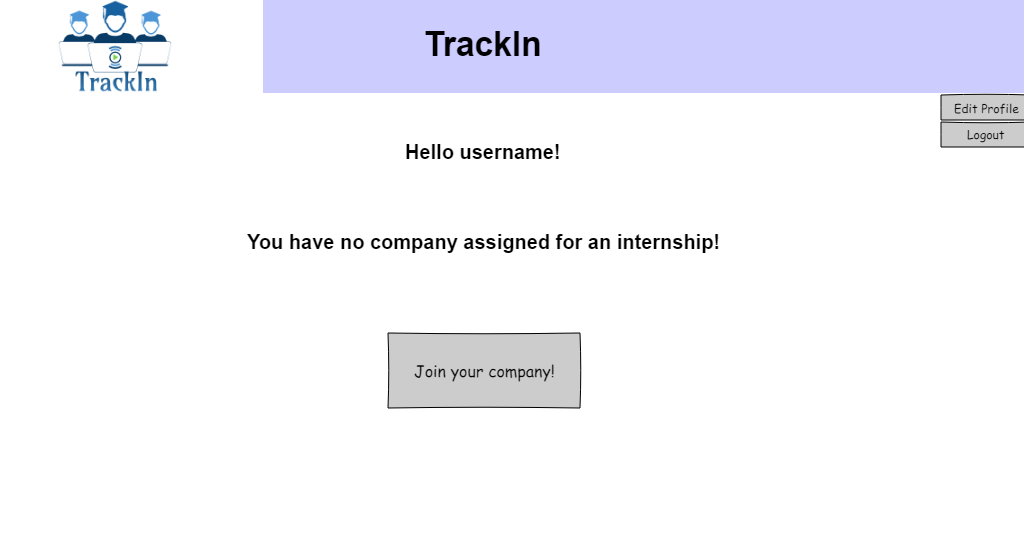


Image - Intern No Company Page

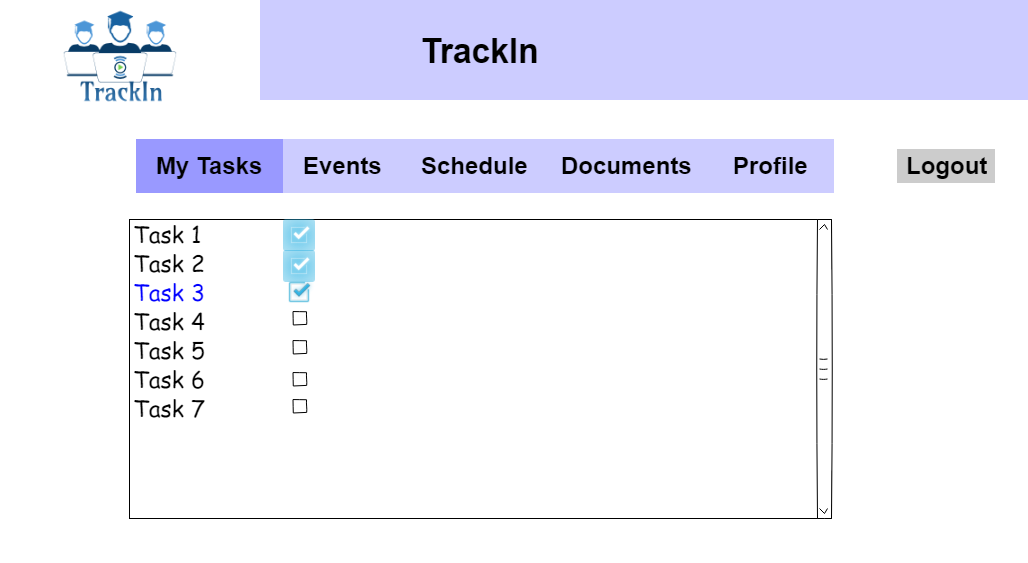


Image - Intern Tasks Page

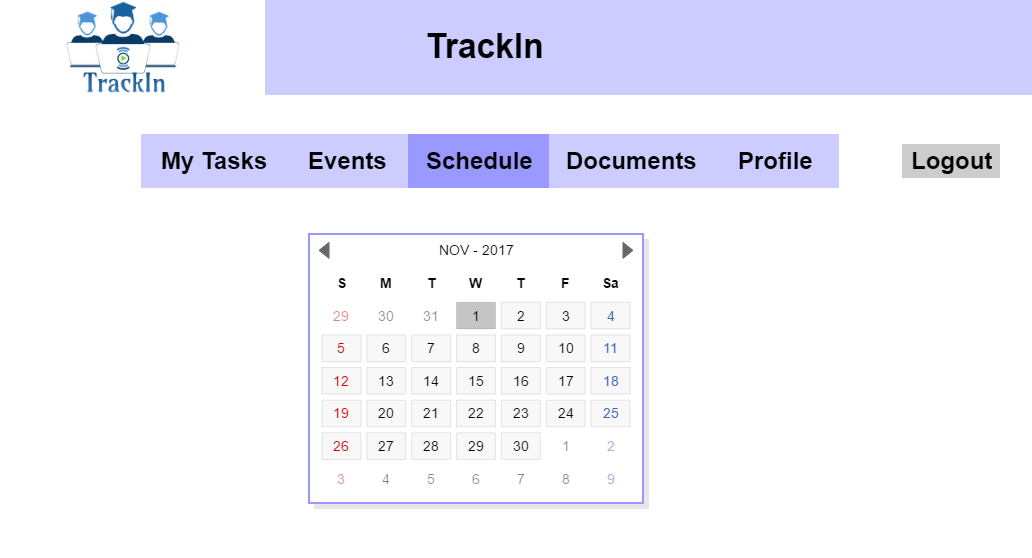


Image - Intern Schedule Page

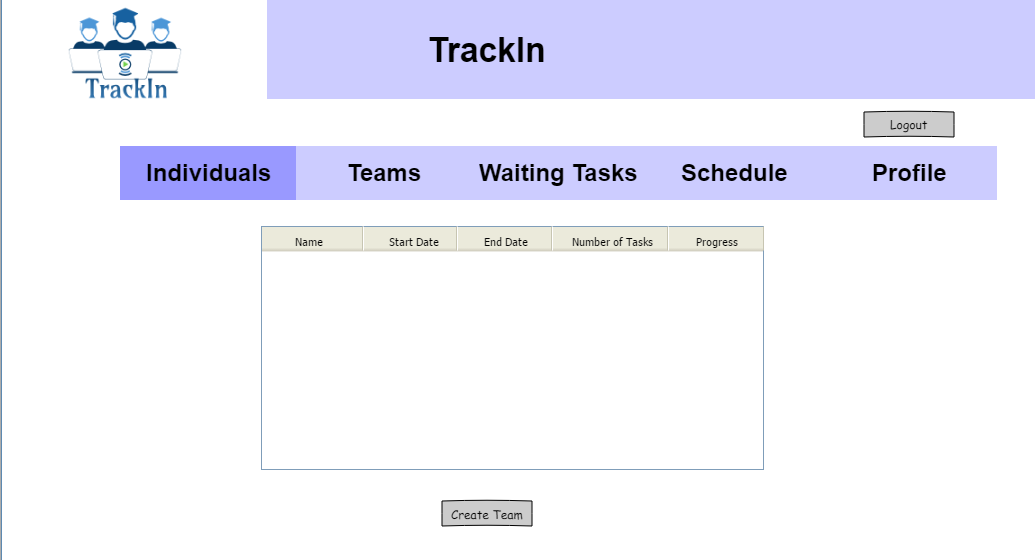


Image - Supervisor Individuals Page

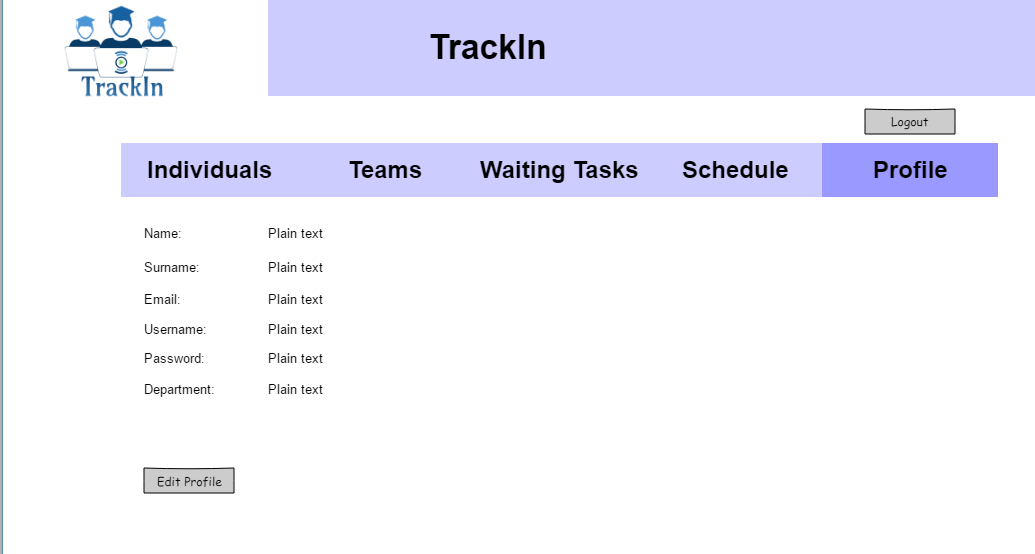


Image - Supervisor Profile Page

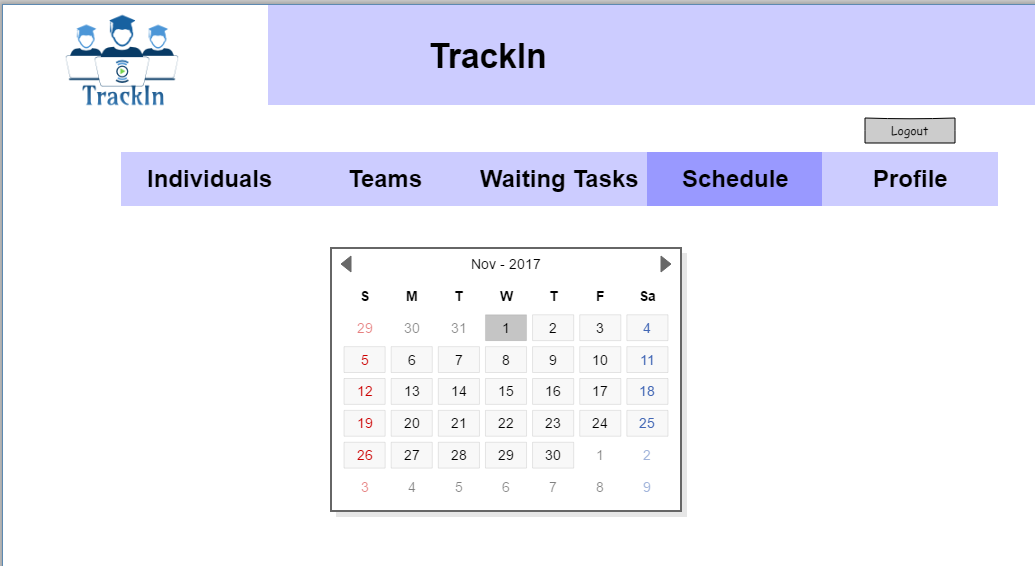


Image - Supervisor Schedule Page

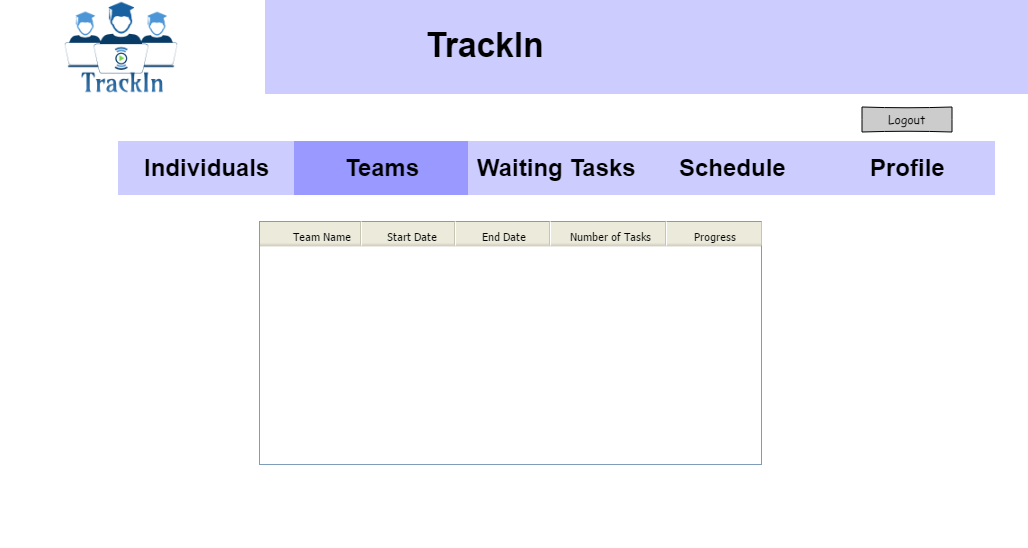


Image - Supervisor Teams Page

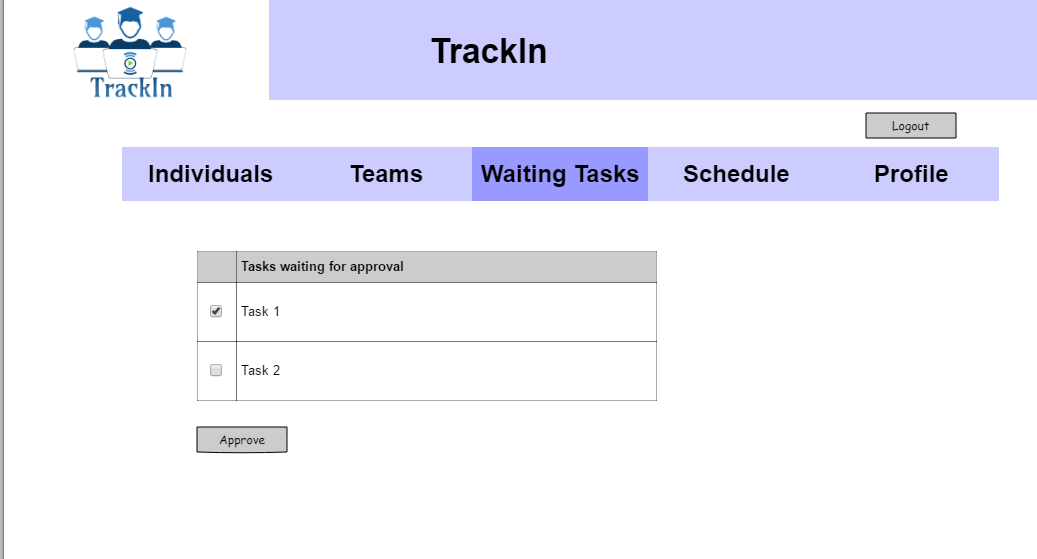


Image - Supervisor Waiting Tasks Page

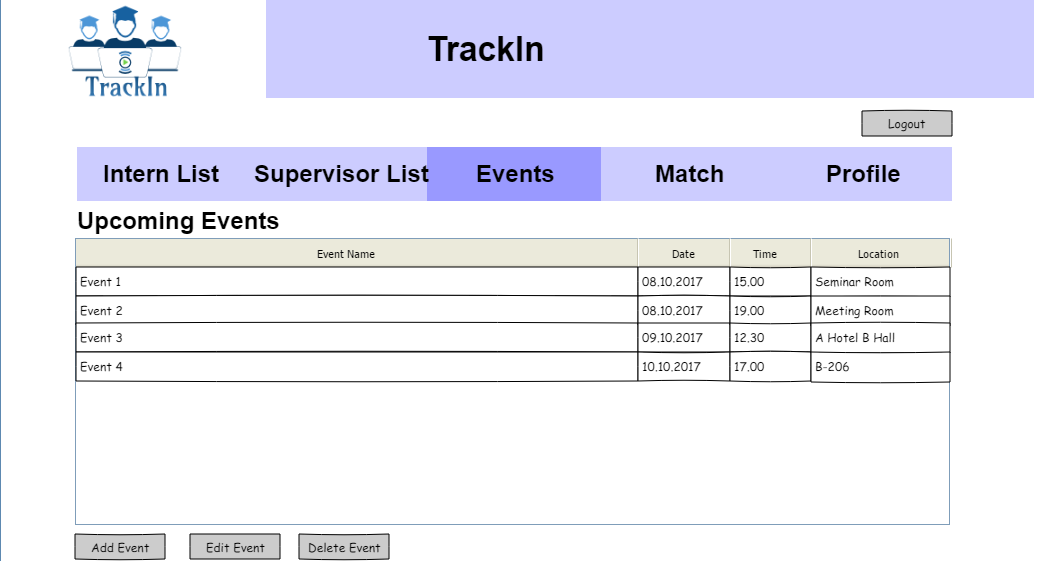


Image - Company Events Page

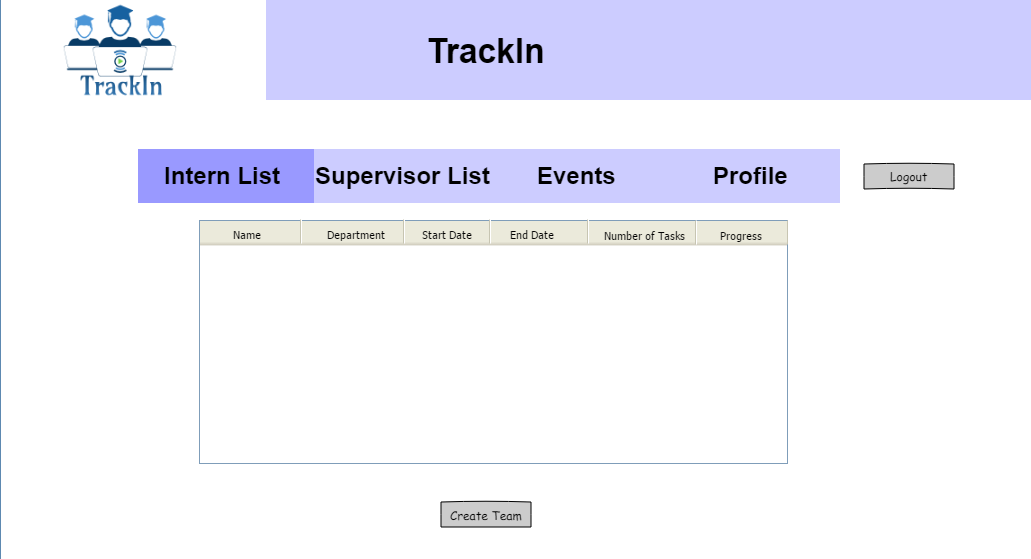


Image - Company Interns Page

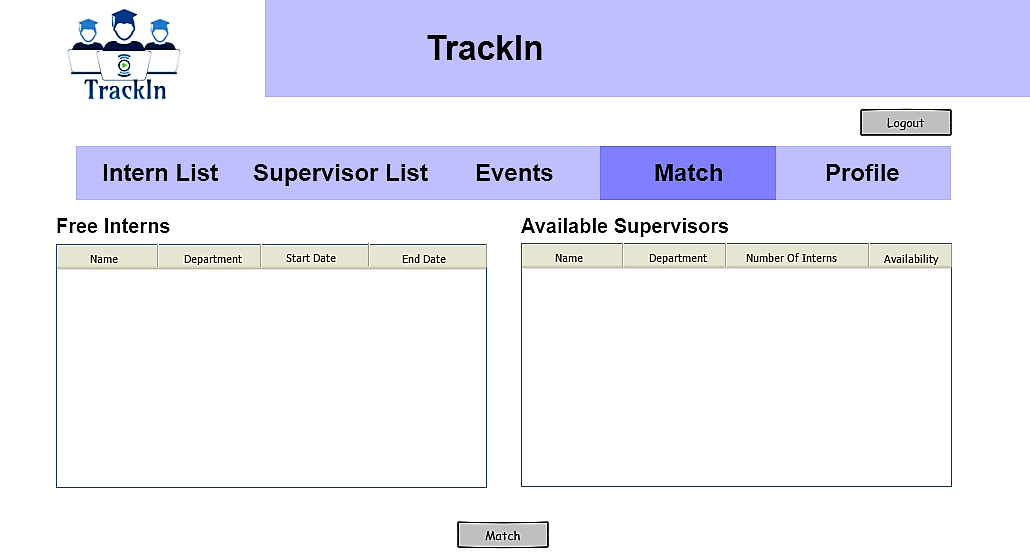


Image - Company Match Page

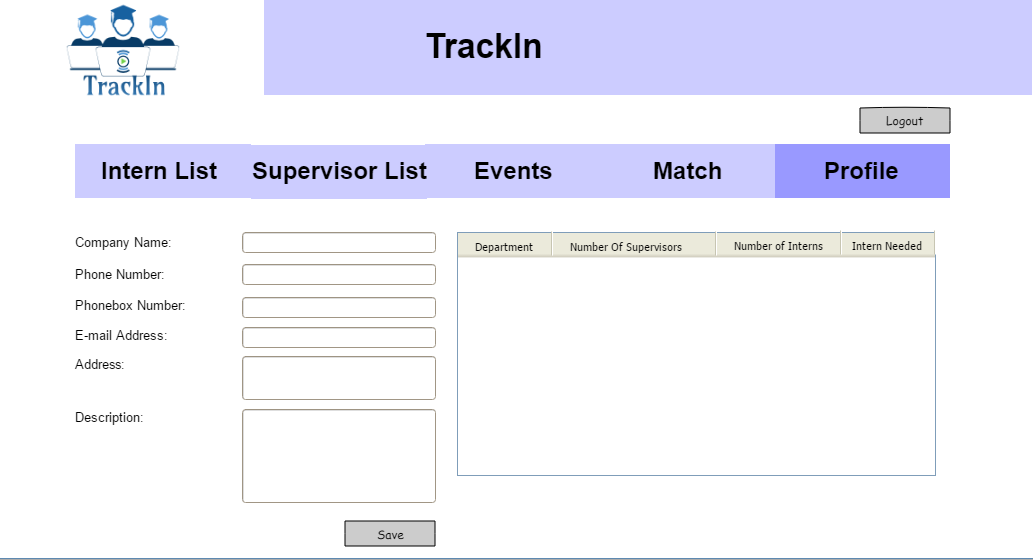


Image - Company Profile Page

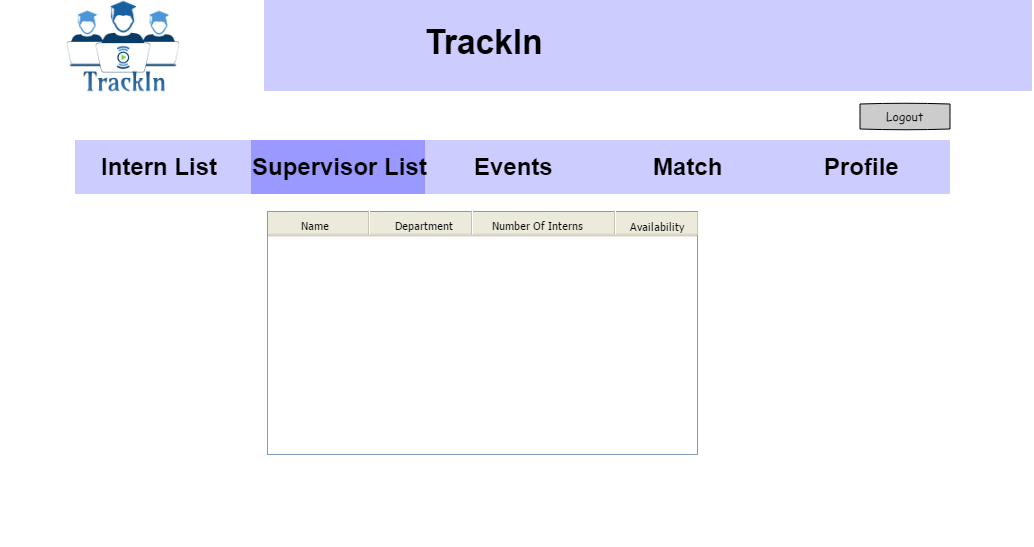


Image - Company Supervisor Page

# References

Object-Oriented Software Engineering, Using UML, Patterns, and Java, 3rd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2010, ISBN-10: 0136066836.