**Project Plan**

**Advanced Software Engineering (CS 5551)**

**Project Group -3**

**Services for Students with Disabilities (SSD)**

**Suresh Yarra, Prudhvi Raj, Shravan Kumar, Anil Kanth**

# Introduction

Recent day’s government and educational institutions are vigorous and determined towards motivating and encouraging students with disabilities and it is becoming highly essential to find a viable solution that makes entire process efficient and easier, So that it will reach out everyone and make it successful. The main aim of this project is to develop an application for students with disabilities with criteria being providing services that will help students in Setting Goals, Organizing/Prioritizing tasks, Scheduling appointment for coaching and also route guidance to nearest parking spot/Accessible entry for students. In technical aspect, Architecture will be Service Oriented (SOA) advanced web technologies will be used and implementation is done by using XML, REST, and SOAP and supplemented by several different API’s over Internet Protocol Backbone.

This report presents our project plan which provides design, architecture and all essential and elaborate detail that gives overall picture and we are following agile development techniques in the implementation of this project.

## Project Back Ground and Related Work

This project was inspired by a real time need to develop an application that helps guiding and mentoring students with disabilities. Petri Alex Nicolle from Institute of Human Development UMKC with an idea to automate monitoring and mentoring students to guide them in the right direction gave us input to work for a cause, also a team of students in ADVANCED SOFTWARE ENGINEERING course from UMKC has done a project called I-plan key features include Scheduling and enrolment to courses and Providing nearby parking spot to location specified were the key features for that application. The project we are going to do this semester will be an upgrade to the already existing features and incorporating additional features like Goal Setting, Task Remainders and Appointment scheduling are going to be some of the key features of the project. This project also helps in stream lining and tracking individual progress of students who are registered with the application and also will help mentors to develop statistics which will help them to focus on areas that most students are interested in, so that there will be more scope to allocate resources which ultimately serves the entire purpose.

# Requirement Specification

This section covers several Technological, Functional and Non-Functional requirements which will be furthermore elaborated in detail below.

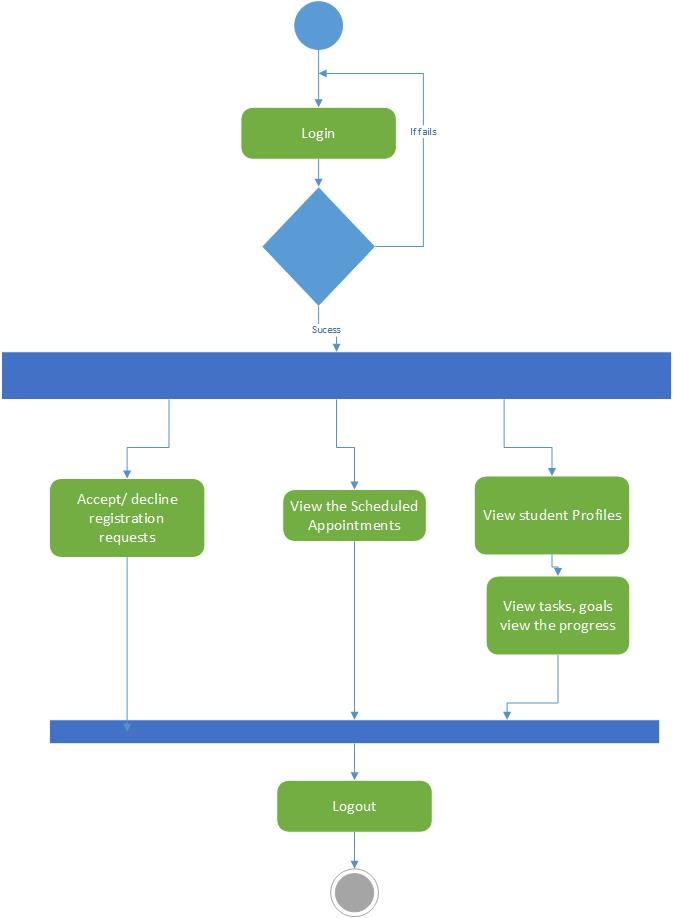
## Functional Requirements

* Students will have one time registration on initial visit, theoretically request for registration will be validated and he will be allowed to access application after he gets approval.
* Approved students will be able to access the application and features.
* Students will be able to Edit/Update their profile or contact information.
* Admin is one of the actors in this application and he/she has rights to view student profile and other related information.
* Students can add or view their tasks and goals, and those are maintained by score factor, if the student is week in a particular area then he/she was assisted by the admin by providing the tutor or by motivation etc. .
* Student will be able choose timings of appointment and request appointment from available appointments by face-face or over the phone.
* Students can search nearby available accessible parking spots and accessible entry points.
* Students can add tasks or remainders to follow up and set goals and rate it himself on scale by self-evaluation.

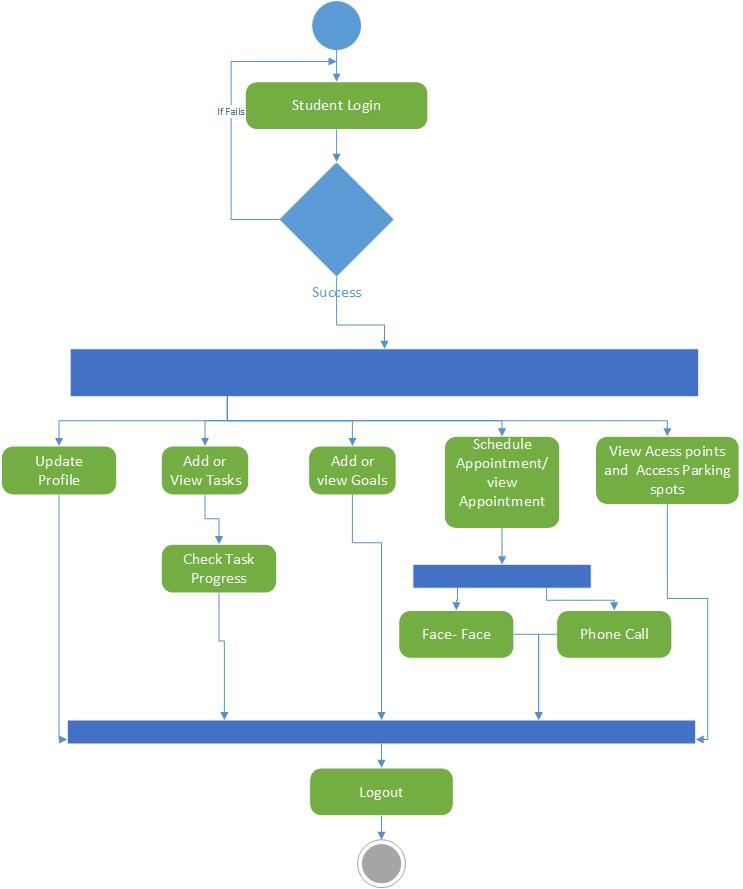
## Non-Functional Requirements

* There is no limit on number of students who registers with application.
* Validation restriction or suggestion should pop up if the input provided by user does not satisfy conditions.
* Students with valid email addresses in the format of [userid@institution.edu](mailto:userid@institution.edu) can only register with the application.
* Sessions are maintained throughout application and will have time out and user logs out automatically, if the user is idle for a long time.
* Application should be compatible with all the browsers that will support client side JavaScript.
* Navigation between pages should be convenient and response time for request should not exceed maximum of 4 seconds.
* Application is secure and session is maintained private.

## UML Activity Diagram-ADMIN



## UML Activity Diagram-Student



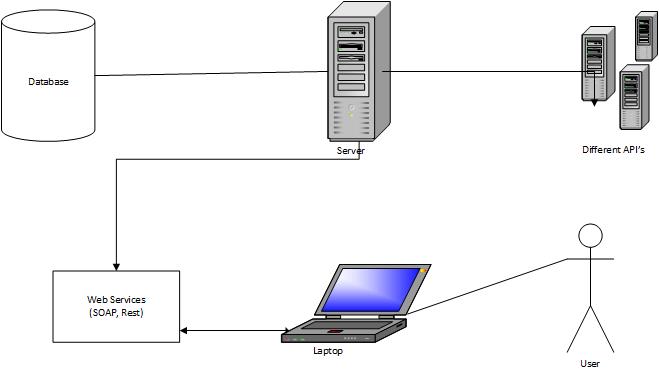
# Framework Specification

This Chapter elaborates Frame Work Specification in elaborate detail in following sections.

## Assumptions and Principles

* This application is open to students and faculty and they should email id that ends with .edu domain name at the time of registration.
* Since some features in this application are designed only considering premises of university, such as providing information about accessibility parking and access points, any other information that is out of scope may not be available.
* Considering the essential requirements and based on practical assumptions this application is mainly intended to leverage the overall system where most implementation is ledger based.

## System Architecture Diagram



# System Specification

This chapter mainly presents System features and specification in a higher level elaborately in upcoming sessions.

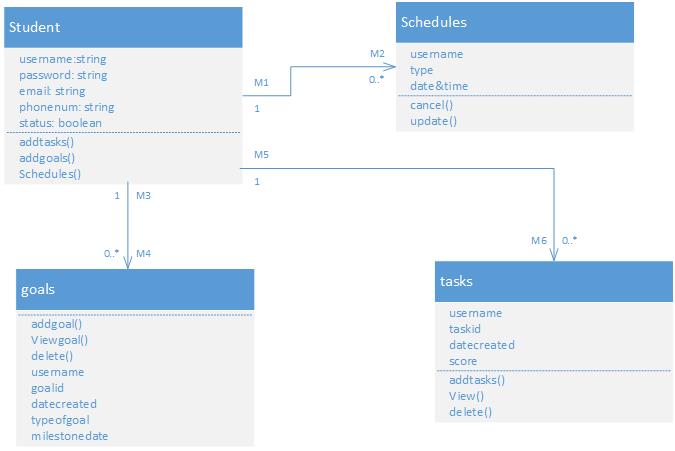
## Existing Service

The application which was developed initially in the earlier semesters supported Adding Courses, Adding Events and guiding to nearest parking spot.

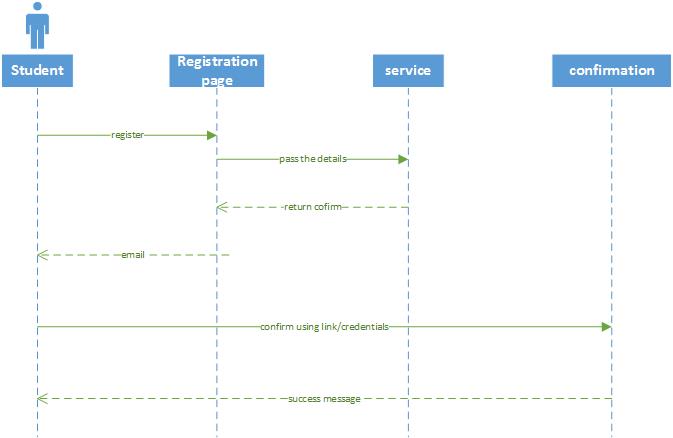
## New Service to Build

In the Application that we are going to develop we are going to enhance some of the features which already exist and it comes with add-ons which are discussed in earlier sections.

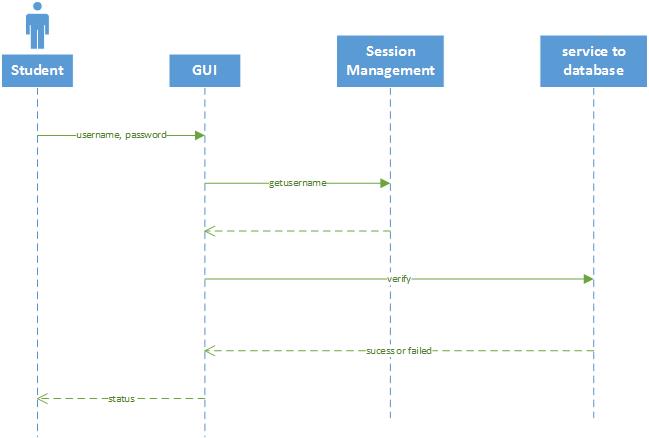
## Class Diagram



## Sequence Diagram-Registration



## Sequence Diagram-Login



# Design of Web Client

Rest Process is going to be adopted and implemented throughout our application development. We are also planning to use Twitter Bootstrap for the front end design and with a vigorous motive that application will be supported by all leading platforms which are available in market.

# Future Scope:

Incorporating Real time navigation support and voice over services to existing features over latest mobile platforms like Android, IOS and Windows.

# Risk Management

This chapter deals about Risk Management and Specifies Technological and Architectural Requirements. Whenever we start a web application development everyone will ask about the effective risk management program. The process for determining and managing the risk of websites and the data they store and retrieve is explain by four phases they are.

1. Asset Identification

2. Vulnerability Management

3. Reporting / Communication

4. Protection

## Technological and Architectural Requirements

Front End:

* HTML
* CSS
* Twitter BootStrap

Back End:

* UMKC Student Database
* SQL server 2014
* C#

Web Services:

* Microsoft Visual Studio 2010 ( C#)
* UMKC Server(KC-SCE-CS551.kc.umkc.edu)

UML /Architecture Diagrams**:**

* Microsoft Visio 2013

Our Scrumdo url is: https://www.scrumdo.com/projects/project/cs5551fall14/iteration/111950

# Bibliography

* <https://www.whitehatsec.com/assets/DS/DS_risk071009.pdf>
* <http://msdn.microsoft.com/en-us/library/office/fp161226(v=office.15).aspx>
* <https://docs.angularjs.org/tutorial/step_01>
* <http://getbootstrap.com/2.3.2/getting-started.html#file-structure>