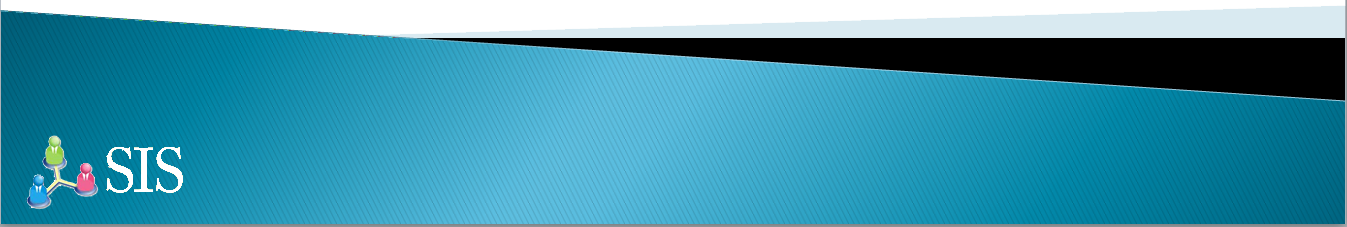
|  |
| --- |
| Student Information System (SIS) |

**Anupama Karumudi**

**Veekija Thirumalai**

**A Special Project**

**Submitted in Partial Fulfillment of the**

**Requirements for the Degree of**

**Master of Science**

**In**

**Computer Information Technology**

**Department of Computer Science**

**Central Connecticut State University**

**New Britain, CT**

**MAY 2013**

**Project Advisor: Dr. Chad Williams**

**Abstract**

Student information system serves as the technology infrastructure for schools. SIS provides quick and easy access to student performance data, which is critical in making timely strategic decisions. It helps identify potential problems and take the appropriate prescriptive or preventative actions to ensure each student is performing to the best of his or her abilities. Combined with parental involvement and increased student accountability, this is a recipe for success in the world of individualized learning. SIS would be accessible from both web and mobile devices. It enables today's educators to make timely decisions that impact student performance while creating a collaborative environment for parents, teachers and students to work together. This proposal document elaborates the capabilities of SIS application, its modules, and software/hardware requirements.

**Student Information System (SIS)**

Contents

[1 Revision History 6](#_Toc354959445)

[2 Purpose 9](#_Toc354959446)

[2.1 Background: 9](#_Toc354959447)

[2.2 Introduction 10](#_Toc354959448)

[2.3 Overview 12](#_Toc354959449)

[2.4 Glossary 14](#_Toc354959450)

[3 Requirements Specification 16](#_Toc354959451)

[3.1 Functional Requirements 16](#_Toc354959452)

[3.1.1 Access SIS home page 16](#_Toc354959453)

[3.1.2 Login 16](#_Toc354959454)

[3.1.3 Logout 17](#_Toc354959455)

[3.1.4 Update Profile 17](#_Toc354959456)

[3.1.5 View attendance 18](#_Toc354959457)

[3.1.6 Update attendance 18](#_Toc354959458)

[3.1.7 View progress 19](#_Toc354959459)

[3.1.8 Update progress 19](#_Toc354959460)

[3.1.9 View grades 20](#_Toc354959461)

[3.1.10 Update grades 20](#_Toc354959462)

[3.1.11 View message center 21](#_Toc354959463)

[3.1.12 Communicate with the other users through the message center 21](#_Toc354959464)

[3.1.13 Set objectives to IEP student 22](#_Toc354959465)

[3.1.14 View IEP student progress 23](#_Toc354959466)

[3.1.15 Update IEP student progress 23](#_Toc354959467)

[3.1.16 Student/Parent: View attendance 24](#_Toc354959468)

[3.1.17 Student/Parent: View progress 24](#_Toc354959469)

[3.1.18 Student/Parent: View grades 25](#_Toc354959470)

[3.1.19 Prospective students 25](#_Toc354959471)

[3.1.20 Update prospective students page 26](#_Toc354959472)

[3.1.21 Submit online application 27](#_Toc354959473)

[3.1.22 Track application status 27](#_Toc354959474)

[3.1.23 View all new applications 28](#_Toc354959475)

[3.1.24 Update application 29](#_Toc354959476)

[3.1.25 Update admission status 29](#_Toc354959477)

[3.1.26 Create class and schedules 30](#_Toc354959478)

[3.1.27 View teachers 31](#_Toc354959479)

[3.1.28 Add teacher 32](#_Toc354959480)

[3.1.29 Update teacher 32](#_Toc354959481)

[3.1.30 Enroll new students 33](#_Toc354959482)

[3.1.31 View student records 34](#_Toc354959483)

[3.1.32 Updated student records 34](#_Toc354959484)

[3.1.33 Handle state reports 35](#_Toc354959485)

[3.1.34 View ad-hoc reports 36](#_Toc354959486)

[3.1.35 Mobile Login 37](#_Toc354959487)

[3.1.36 Mobile Logout 37](#_Toc354959488)

[3.1.37 Mobile menu options screen 38](#_Toc354959489)

[3.1.38 Mobile view message center 38](#_Toc354959490)

[3.1.39 Mobile view schedules 39](#_Toc354959491)

[3.1.40 Mobile view score card 39](#_Toc354959492)

[4 System Architecture Design 42](#_Toc354959493)

[4.1 Overview 42](#_Toc354959494)

[4.2 System Architecture Design – Non-functional characteristics 48](#_Toc354959495)

[5 Database Schema Design 50](#_Toc354959496)

[5.1 Entity Relationship Diagrams 50](#_Toc354959497)

[5.2 Database Table Descriptions 51](#_Toc354959498)

[6 System Documentation 54](#_Toc354959499)

[6.1 Rationale for selected technology: 54](#_Toc354959500)

[6.2 System setup and Installation Instruction: 55](#_Toc354959501)

[6.2.1 Required Software 55](#_Toc354959502)

[6.2.2 Database Setup Instructions 55](#_Toc354959503)

[6.2.3 SIS Web application Setup Instructions 56](#_Toc354959504)

[6.3 System Features: 57](#_Toc354959505)

[6.3.1 Common modules 59](#_Toc354959506)

[6.3.2 Public user modules 61](#_Toc354959507)

[6.3.3 Administrator Module 66](#_Toc354959508)

[6.3.3.1 Administrator Home page 66](#_Toc354959509)

[6.3.3.2 SIS System Management 66](#_Toc354959510)

[6.3.3.2.1 SIS Content Maintenance 67](#_Toc354959511)

[6.3.3.2.2 School year Maintenance 68](#_Toc354959512)

[6.3.3.2.3 Grade Level Maintenance 69](#_Toc354959513)

[6.3.3.2.4 Subject Maintenance 70](#_Toc354959514)

[6.3.3.2.5 Period Maintenance 71](#_Toc354959515)

[6.3.3.2.6 Teacher/Subject Schedule Maintenance 72](#_Toc354959516)

[6.3.3.3 Admission Management 74](#_Toc354959517)

[6.3.3.4 Teacher Management 77](#_Toc354959518)

[6.3.3.5 Student Management 77](#_Toc354959519)

[6.3.3.5.1 Student Maintenance 78](#_Toc354959520)

[6.3.3.5.2 Student Grade Level Enrollment 79](#_Toc354959521)

[6.3.3.5.3 Student Subject Enrollment 80](#_Toc354959522)

[6.3.3.5.4 Process student school year results 82](#_Toc354959523)

[6.3.3.6 Reports Management 83](#_Toc354959524)

[6.3.3.6.1 Application/Admission Status report 83](#_Toc354959525)

[6.3.3.6.2 Student Grade level report 84](#_Toc354959526)

[6.3.3.6.3 Teacher schedule report 85](#_Toc354959527)

[6.3.3.6.4 State report 86](#_Toc354959528)

[6.3.4 Teacher Module 89](#_Toc354959529)

[6.3.4.1 Teacher Schedules (Home Page) 89](#_Toc354959530)

[6.3.4.2 Take Attendance 90](#_Toc354959531)

[6.3.4.3 View Attendance 91](#_Toc354959532)

[6.3.4.4 Update Grades 92](#_Toc354959533)

[6.3.4.5 View Grades 93](#_Toc354959534)

[6.3.4.6 Individualized Education Plan (IEP) 94](#_Toc354959535)

[6.3.4.7 Individualized Education Plan (IEP) - Goals 96](#_Toc354959536)

[6.3.4.8 Add/Edit IEP Goal 97](#_Toc354959537)

[6.3.4.9 IEP Goal Progress 99](#_Toc354959538)

[6.3.4.10 Add/Edit IEP Goal Progress 100](#_Toc354959539)

[6.3.4.11 Message Center 101](#_Toc354959540)

[6.3.4.12 Message Details 102](#_Toc354959541)

[6.3.4.13 Reply Message 104](#_Toc354959542)

[6.3.4.14 Compose New Message 105](#_Toc354959543)

[6.3.5 Student /Parent Module 108](#_Toc354959544)

[6.3.5.1 Student- Schedules (Home Page) 108](#_Toc354959545)

[6.3.5.2 Student - View Attendance 109](#_Toc354959546)

[6.3.5.3 Student - View Grades 110](#_Toc354959547)

[6.3.5.4 Student - Message Center 111](#_Toc354959548)

[6.3.5.5 Student - Message Details 112](#_Toc354959549)

[6.3.5.6 Student - Reply Message 113](#_Toc354959550)

[6.3.5.7 Student - Compose New Message 114](#_Toc354959551)

# Revision History

|  |  |  |
| --- | --- | --- |
| **Revision** | **Description of Revision** | **Date** |
| 1.0 | Initial Draft | 01/15/2013 |
| 1.1 | Added functional requirements with use cases | 01/22/2013 |
| 1.2 | Added system architecture design | 01/29/2013 |
| 1.3 | Updated the use case diagram.  Added following new use cases:   * Mobile view message center * Mobile view schedules * Mobile view score card * Mobile log in * Mobile menu option screen * Mobile log out   Updated the following use cases:   * Set objectives to IEP student * View IEP student progress * Update IEP student progress * Prospective students * Update prospective student’s page * Create Class and schedules * Enroll new students * Handle State report   Updated system architecture design diagram for mobile web.  Updated sequence diagram for bean load process.  Updated Presentation tier for Mobile Web Architecture.  Updated Business Object tier for bean load process.  Updated Security for SSL needs.  Added Database schema design | 02/05/2013 |
| 1.4 | Made following changes to the database schema:   * Updated table and column names to follow consistent naming conventions and used the camel case for both entities and attributes. * Renamed Student Enrollment to StudentGradeLevel. * Removed GradeLevelId column form StudentGrade table. * Renamed StudentGrade to StudentScoreCard. * Removed GradeLevelSubjects table, this table is no longer needed. * Removed GradeLevelId from SchedulesTeacherAssignment table. * Renamed SchedulesTeacherAssignment to SubjectSchedule. * Added new StudentSubjectSchedule table. * Removed GradeLevelId from AttendanceTracking table. * Added GoalTitle column to the IepGoals table. * Added GoalDescription column to the IepGoals table. * Added StartDate column to the IepGoals table. * Added EndDate column to the IepGoals table.   Renamed User table to Users.  Priority for mobile Use cases are changed to Conditional | 02/11/2012 |
| 1.5 | Allocate assignment (in teacher module) and cafeteria provision (in student module) is removed in the proposal section. | 02/18/2013 |
| 1.6 | System Documentation updated | 04/22/2013 |
| 1.7 | Completed adding system documentation | 04/29/2013 |

**Student Information System  
Purpose**

# Purpose

## Background:

In today’s school system, the classroom landscape is changing and the increased communication between parents, students and teachers is impacting student achievement in new ways. Everyone teachers, parents and students looking for more advanced and collaborative tools to make timely decisions. At the same time, the public schools must comply with federal, state and local regulations and reporting. There are numerous state and federal reports such as Public School Information System (PSIS) or SEDAC (special education report) that must be completed accurately and in a timely manner. Because most of public schools are non-profit organizations, the income comes from the state and local taxes. Therefore, each department receives a budget based on the active number of students, staff, etc. in the district. Failure to comply with accurate information will result in a citation from the state and possible loss of budget for the department.

One of the major problems in the Public Schools is the amount of non-related databases being used at every school. Even though all elementary school students are being provided with the “same” education, each school collects and grades the information differently. Each school designed and developed multiple non-related databases, usually in Excel, to satisfy their needs, but this becomes a problem when reporting information to the state, or when the database administrator tries to upload the grades.

To resolve the above discussed constraints, it was decided to design and develop Student Information System (SIS) that enables today's educators to make timely decisions that impact student performance while creating a collaborative environment for parents, teachers and students to work together. The following sets of features are implemented in Administrator, teacher and student modules. In administrator modules, the streamlined admission workflow process is implemented. This process will eliminate the manual admission process followed in many school systems. The extensive reporting capabilities are implemented to generate various reports in a real time that replaces the usage of paper based reports and reduces lot of manual effort as well. In teacher modules, the real time attendance tracking, score updates and IEP tracking features are implemented. To establish effective communication between teachers and students/parents the message center feature is implemented.

## Introduction

The main intent of SIS application is to provide a solution to automate the schools day to day functions such as: student admission, student enrollment, and student grade management. At the same time providing a system to accurately store and manage all student information which would allow generating accurate reports required by the state such as PSIS. SIS creates a collaborative environment for parents, teachers and students to work together to improve the student performance. The primary beneficiaries of this application are teachers, students, parents, and administrative staff members. The features/functions of SIS would divide into multiple modules to target different users of the system, which includes a student admission workflow process, special education management (IEP), teachers’ module, parent/student module, report module, user and role management. SIS enables each user login with secured credentials registered with the application and allows access to the functionality designated to only that user role.

Admin workflow process provides an automated streamlined approach to manage student admission process. This workflow process will be triggered by the system either by submitting on-line application forms through SIS web application by parents or by entering student information based on the submitted hard copy of the application. SIS application controls the workflow by moving into various steps after an appropriate review and approvals. In each step of the workflow, system assigns a unique status to track the life of the workflow uniquely. The various statuses of the workflow include: Application Submitted, Application Review In-progress, In-person Interview Requested, In-person Interview Completed and Admission Granted. Along with admissions and enrollment, admin users can manage staff and courses.

Individualized Education Plan (IEP) allows schools to store information regarding the students with special education needs. The access to IEP is restricted to strictly authorized users only because of the sensitivity of the data. Essentially, it is used by the special education teachers, school psychologists, IEP coordinators, to enter individualized education plans for students with disabilities. This module would allow users to store files, recordings and videos. For instance, the school psychologist worked with the special education student on some goals or objectives. Using the iPad or any other device, the school psychologist recorded the session. The school psychologist uploads the video to the web server so that the file is accessible to all the “team” working with the student. The video can be used by the team to discuss new objectives or observations based on the student’s behavior.

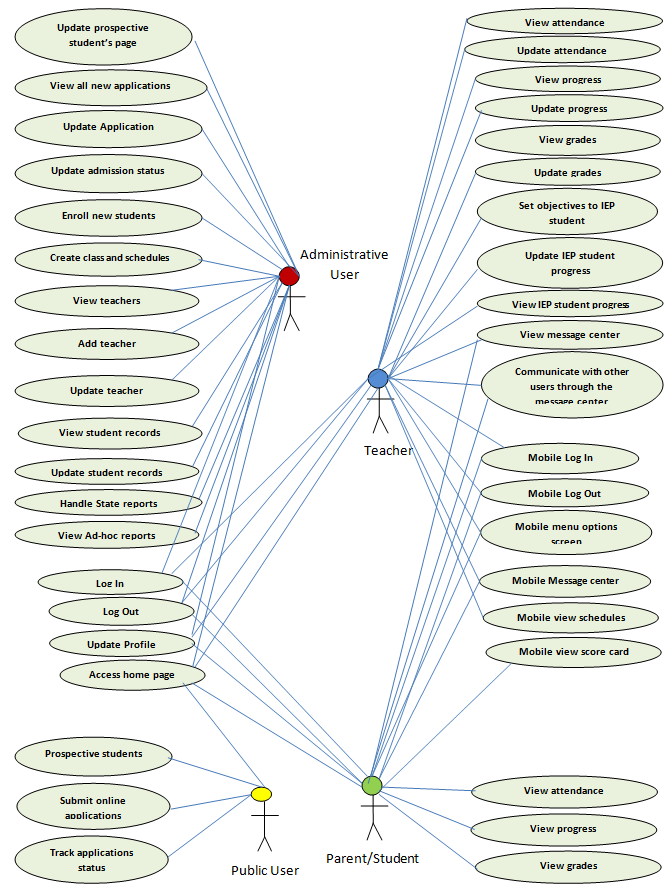
Teacher module would be mainly used by teachers to perform their day to day activities effectively. This module also provides an ability to manage the attendance and grades for all students. In addition to this, it allows teachers to collaborate and communicate effectively with students and parents by sending messages or real time feedback on student’s performance.

Student/parent module is mainly intended for students/parents. Using this module, students/parents would be able to view the grades, attendance status, and feedback/comments from the teachers and can effectively collaborate and communicate with teachers by sending messages or inquiries. User/Role management process would allow administrative users to create various roles and users to secure the application’s functionality from un-authorized users.

Report modules would provide administrative users and educational departments to generate various reports by selecting various criteria. Users would be able to accurately generate student information required by the state such as PSIS (Public School Information Systems). Users would be able to generate the reports either in web page or in excel sheet.

## Overview

There are four categories of users of the system: Administrative user, the parent/student user, teacher user and the public user. These users and the action each of them can perform on the system are listed in the use case diagram below.



## Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| SIS | Student Information System |
| IEP | Individualized Education Plan |
| Data Entry | Form filled out and submitted by user. |
| Web Site | A place on the world wide web |
| Screen | A specific UI on the SIS web site |
| TBD | To be decided |
| TBN | To be named |
| QA | Quality assurance |
| SRS | Software Requirements Specification |

**Student Information System  
Requirements Specification**

# Requirements Specification

## Functional Requirements

### Access SIS home page

|  |  |
| --- | --- |
| **Use case name** | **Access SIS home page** |
| **Actors** | Administrative user, Teachers, Parent/Student, Public users |
| **Priority** | Essential |
| **Description** | Users are going to access the SIS from a hosted web url. The home page gives an overview of SIS application and provides an option to login for registered users. Other public users can browse and access the content like inquiries, submit new inquiries, submit new applications and track application status. |
| **Data** | URL |
| **Stimulus** | None |
| **Response** | When users enter SIS url in their browser, it displays the home screen. |
| **Comments** | All users |

### Login

|  |  |
| --- | --- |
| **Use case name** | **Log In** |
| **Actors** | Administrative user, Teachers, Parent/Student |
| **Priority** | Essential |
| **Description** | After user submitting the user name and password, the system will performs the validation and display the respective interface based on the user type. An error message will be displayed if the user name is not found or the password doesn’t match with the one in the system. |
| **Data** | User name, Password |
| **Stimulus** | Log In command |
| **Response** | If log in is successful, system will display welcome scree according to the user type. Otherwise error message will be displayed |
| **Comments** | 3 types of users – Administrative user, Teachers, Parent/Student |

### Logout

|  |  |
| --- | --- |
| **Use case name** | **Log Out** |
| **Actors** | Administrative user, Teachers, Parent/Student |
| **Priority** | Essential |
| **Description** | After user click on Log out button or if the user is inactive for a certain length of time, the system will not allow them access without a re-log in. |
| **Data** | Button click, or Session time out |
| **Stimulus** | Log Out command |
| **Response** | Log Out message Or a redirect to Log In screen. |
| **Comments** | 3 types of users – Administrative user, Teachers, Parent/Student |

### Update Profile

|  |  |
| --- | --- |
| **Use case name** | **Update Profile** |
| **Actors** | Administrative user, Teachers, Parent/Student |
| **Priority** | Essential |
| **Description** | User will click on the update profile link and it allows users to modify,   * Contact details * Communication address * Change password |
| **Data** | Contact details, communication address and new password |
| **Stimulus** | Update Profile command |
| **Response** | It displays a confirmation message if the update is successful otherwise display an error message. |
| **Comments** | 3 types of users – Administrative user, Teachers, Parent/Student |

### View attendance

|  |  |
| --- | --- |
| **Use case name** | **View attendance** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher will click on the view attendance link and it will ask user to choose a class from list of all assigned classes to the teacher. Once teacher selects a class, it will display list of student attendance details by day/week/month. |
| **Data** | Class ID, filter type day/week/month |
| **Stimulus** | View attendance command |
| **Response** | It displays tabular list of attendance details for all students. |
| **Comments** | Teachers |

### Update attendance

|  |  |
| --- | --- |
| **Use case name** | **Update attendance** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher will click on the view attendance link and it will ask user to choose a class from list of all assigned classes to the teacher. Once teacher selects a class, it will display list of student attendance details by day/week/month in edit mode. Teacher clicks on update attendance button to submit the attendance details. |
| **Data** | Class ID, filter type day/week/month |
| **Stimulus** | Update attendance command |
| **Response** | It displays a confirmation message if the update is successful otherwise display an error message. |
| **Comments** | Teachers |

### View progress

|  |  |
| --- | --- |
| **Use case name** | **View progress** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher will click on the view progress link and it will ask user to choose a class from list of all assigned classes to the teacher. Once teacher selects a class, it will display list of all students and their progress. |
| **Data** | Class ID |
| **Stimulus** | View progress command |
| **Response** | It displays tabular list of progress for all students. |
| **Comments** | Teachers |

### Update progress

|  |  |
| --- | --- |
| **Use case name** | **Update progress** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher will click on the view progress link and it will ask user to choose a class from list of all assigned classes to the teacher. Once teacher selects a class, it will display list of all students and their progress. Teacher will select a student from the list to edit their progress. |
| **Data** | Class ID, Student ID, Progress Details |
| **Stimulus** | Update progress command |
| **Response** | It displays a confirmation message if the update is successful otherwise display an error message. |
| **Comments** | Teachers |

### View grades

|  |  |
| --- | --- |
| **Use case name** | **View grades** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher will click on the view grades link and it will ask user to choose a class from list of all assigned classes to the teacher. Once teacher selects a class, it will display list of all students and their grades. |
| **Data** | Class ID |
| **Stimulus** | View grades command |
| **Response** | It displays tabular list of grades for all students. |
| **Comments** | Teachers |

### Update grades

|  |  |
| --- | --- |
| **Use case name** | **Update grades** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher will click on the view grades link and it will ask user to choose a class from list of all assigned classes to the teacher. Once teacher selects a class, it will display list of all students and their grades. Teacher will select a student from the list to edit their grades. |
| **Data** | Class ID, Student ID, Grade details |
| **Stimulus** | Update grade command |
| **Response** | It displays a confirmation message if the update is successful otherwise display an error message. |
| **Comments** | Teachers |

### View message center

|  |  |
| --- | --- |
| **Use case name** | **View message center** |
| **Actors** | Teachers, Parent/Student |
| **Priority** | Essential |
| **Description** | User will click on the message center link to see the list of all incoming messages in their inbox from other users. User selects a message link to see the details. |
| **Data** | Message ID |
| **Stimulus** | Message center command |
| **Response** | It displays tabular list of all incoming messages from user’s inbox. |
| **Comments** | Teachers, Parent/Student |

### Communicate with the other users through the message center

|  |  |
| --- | --- |
| **Use case name** | **Communicate with other users through the message center** |
| **Actors** | Teachers, Parent/Student |
| **Priority** | Essential |
| **Description** | User will click on the message center link to see the list of all incoming messages in their inbox from other users. User selects a message link to see the details and selects a reply button to send response. User can also click on the compose button to create a new message and send it to single or multiple users. |
| **Data** | Subject, message, recipients |
| **Stimulus** | Send message command |
| **Response** | It displays a confirmation message if the message successfully sent otherwise display an error message. |
| **Comments** | Teachers, Parent/Student |

### Set objectives to IEP student

|  |  |
| --- | --- |
| **Use case name** | **Set objectives to IEP student** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher clicks on the IEP link from the menu.  Teacher selects a class from the list of all assigned classes to her/him. The application displays tabular list of all IEP students from the selected class with links to “goals & objectives”, “view progress” and “update progress”.  Teacher selects “goals & objectives” link next to the student name.  It displays a one big text area for entering (multiline text) goals and objectives for the IEP student.  Teacher updates the goals, objectives and status.  Teacher clicks on submit button to save the changes. |
| **Data** | Class ID, Student ID, goals & objectives text from the text area and status. |
| **Stimulus** | Submit command |
| **Response** | It displays a confirmation message if it saves the changes successfully otherwise display an error message. |
| **Comments** | Teachers |

### View IEP student progress

|  |  |
| --- | --- |
| **Use case name** | **View IEP student progress** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher clicks on the IEP link from the menu.  Teacher selects a class from the list of all assigned classes to her/him. The application displays tabular list of all IEP students from the selected class with links to “goals & objectives”, “view progress” and “update progress”.  Teacher selects “view progress” link next to the student name.  It displays the IEP student progress along with links to any shared files upload for review. |
| **Data** | Class ID, Student ID |
| **Stimulus** |  |
| **Response** | It displays the IEP student’s detailed progress information. |
| **Comments** | Teachers |

### Update IEP student progress

|  |  |
| --- | --- |
| **Use case name** | **Update IEP student progress** |
| **Actors** | Teachers |
| **Priority** | Essential |
| **Description** | Teacher clicks on the IEP link from the menu.  Teacher selects a class from the list of all assigned classes to her/him. The application displays tabular list of all IEP students from the selected class with links to “goals & objectives”, “view progress” and “update progress”.  Teacher selects “update progress” link next to the student name.  Teacher updates the progress and optionally uploads any files to share with other users and clicks on submit button to save the details. |
| **Data** | Class ID, Student ID, Progress Details, Optional files to upload. |
| **Stimulus** | Submit command |
| **Response** | It displays a confirmation message if it updates the details successfully otherwise display an error message. |
| **Comments** | Teachers |

### Student/Parent: View attendance

|  |  |
| --- | --- |
| **Use case name** | **View Attendance** |
| **Actors** | Parent/Student |
| **Priority** | Essential |
| **Description** | * Parent/Student will click on the view attendance link. * System will display the attendance details for the student. |
| **Data** | Attendance information |
| **Stimulus** | * View Attendance Link * Back command |
| **Response** | System displays the attendance details associated to the logged-in student. |
| **Comments** | Parent/Student will perform this function. |

### Student/Parent: View progress

|  |  |
| --- | --- |
| **Use case name** | **View Progress** |
| **Actors** | Parent/Student |
| **Priority** | Essential |
| **Description** | * Parent/Student will click on the View Progress link. * System will display the progress details for the student who is part of identified education plan program. |
| **Data** | IEP progress information |
| **Stimulus** | * View IEP Progress Link * Back command |
| **Response** | System displays the IEP progress details associated to the student. |
| **Comments** | Parent/Student will perform this function. |

### Student/Parent: View grades

|  |  |
| --- | --- |
| **Use case name** | **View Grades** |
| **Actors** | Parent/Student |
| **Priority** | Essential |
| **Description** | * Parent/Student will click on the View Grades link. * System will display the grades for the student. |
| **Data** | Student Information, subject and grade details. |
| **Stimulus** | * View Grades Link * Back command |
| **Response** | System displays the grade details associated to the student. |
| **Comments** | Parent/Student |

### Prospective students

|  |  |
| --- | --- |
| **Use case name** | **Prospective students** |
| **Actors** | Public User |
| **Priority** | Essential |
| **Description** | * User will click on the prospective student’s link on the application start up page. * This link will provide information for prospective students. The contents of this page will be loaded from a database configuration table, which will be maintained by the admin user. |
| **Data** | Meta data information needed for prospective students. |
| **Stimulus** | Dynamic page content will be loaded from a database table. |
| **Response** | Information for prospective students will be displayed. |
| **Comments** | Public User |

### Update prospective students page

|  |  |
| --- | --- |
| **Use case name** | **Update prospective students page** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * The admin user clicks on update prospective student’s page. * System maintains the page contents in a database configuration table, possibly in a rich text format. * Application will load the page contents from database table and displays in a rich text editor for editing. * Once user modifies the page content, he/she will click on the save button to write to the database table. |
| **Data** | Meta data information needed for prospective students. |
| **Stimulus** | Save Button |
| **Response** | Application displays a confirmation message if it updates the information for prospective student’s successfully otherwise display an error message. |
| **Comments** | Administrative User |

### Submit online application

|  |  |
| --- | --- |
| **Use case name** | **Submit Online Application** |
| **Actors** | Public User |
| **Priority** | Essential |
| **Description** | * User will seek for the admission by submitting the online application form. * User will enter all the required fields that are needed for the online application. The details required to fill in online application is:   + Student personal details   + Student demo-graphic details.   + Student’s contact details * User clicks submit button. * System saves the entered details and automatically generates a tracking number. * System displays tracking number in the screen. Users will use this tacking number to track the status of their application. |
| **Data** | Student personal details, demo-graphic information, contact details and tracking number. |
| **Stimulus** | * Submit command * Cancel command |
| **Response** | System stores online application information and displays system generated tracking number. |
| **Comments** | Public User |

### Track application status

|  |  |
| --- | --- |
| **Use case name** | **Track Application Status** |
| **Actors** | Public User |
| **Priority** | Essential |
| **Description** | * Public user will track the application status using tracking number. * User clicks the track application status link. * User enters tracking number and click submits button. * System displays the application status on the screen. |
| **Data** | Tracking number and the status of the tracking number. |
| **Stimulus** | * Submit command * Back command |
| **Response** | System displays the status of tracking number. |
| **Comments** | Public User |

### View all new applications

|  |  |
| --- | --- |
| **Use case name** | **View All new Applications** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View All Applications link. * The system displays all the submitted application. * User will select a specific application to view the details of the specific application. * System displays the details of the selected application. |
| **Data** | List of submitted applications and the details of the selected application. |
| **Stimulus** | * Link to the selected application. * Back command |
| **Response** | Displays the details of the selected application. |
| **Comments** | Administrative User |

### Update application

|  |  |
| --- | --- |
| **Use case name** | **Update Application** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View All Applications link. * The system displays all the submitted application. * User will select a specific application to update the details of the specific application. * System displays the details of the selected application in an editable mode. * User updates the application details and clicks Submit button. * The modified application details will be stored in the database. |
| **Data** | Application details |
| **Stimulus** | * Link to the selected application. * Submit command * Cancel command |
| **Response** | Displays the message stating that application details are saved. |
| **Comments** | Administrative User |

### Update admission status

|  |  |
| --- | --- |
| **Use case name** | **Update Admission Status** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View All Applications link. * The system displays all the submitted application. * User will select a specific application to update the status of the specific application. * System displays the following details:   + Admission workflow Status history and comments   + Admission workflow status dropdown and editable text area to enter the comments.   + The possible admission workflow statuses are:     - Application Received     - Application Review In-progress     - In-person Interview Requested     - Admission Granted     - Admission Denied * User selects an appropriate admission status, enters comment and clicks the Save button. * The admission workflow status will be updated and comments will be saved. |
| **Data** | Admission workflow status history, comments, admission workflow status. |
| **Stimulus** | * Link to the selected application. * Save command * Cancel command |
| **Response** | Displays the message stating that admission workflow details are saved. |
| **Comments** | Administrative User |

### Create class and schedules

|  |  |
| --- | --- |
| **Use case name** | **Create class and schedules** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks Create class and schedules link. * Admission user selects a grade level and click search button. * System will display the list of subjects associated to the grade level. * Then Admin user will assign the teacher to each grade level and subject based on their available slots. |
| **Data** | Grade level, subject and teacher assignments will be done. |
| **Stimulus** | * Search command * Assign teachers to subject command |
| **Response** | Displays the message stating that Teachers are assigned. |
| **Comments** | Administrative User |

### View teachers

|  |  |
| --- | --- |
| **Use case name** | **View teachers** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View teacher’s link. * System displays the search criteria to select the teacher. Search criteria includes the following teacher attributes:   + Teacher First Name   + Teacher Last Name   + Teacher Data of birth   + Teacher ID * User clicks on search button after entering the above information * System displays the teacher data. |
| **Data** | Teacher details. |
| **Stimulus** | * Search command * Link to select a specific teacher record. |
| **Response** | Displays the teacher information. |
| **Comments** | Administrative User |

### Add teacher

|  |  |
| --- | --- |
| **Use case name** | **Add teacher** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks Add teacher link. * User will enter all required fields that are needed for the teacher profile. The details needed for the teacher are:   + Teacher personal details   + Teacher demo-graphic details.   + Teacher contact details * User clicks submit button. * System saves the entered details into the system. |
| **Data** | Teacher personal details, demo-graphic information, contact details. |
| **Stimulus** | * Submit command * Cancel command |

### Update teacher

|  |  |
| --- | --- |
| **Use case name** | **Update teacher** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View Teachers link. * System displays the search criteria to select the teacher. Search criteria includes the following teacher attributes:   + Teacher First Name   + Teacher Last Name   + Teacher Data of birth   + Teacher ID * User clicks on search button after entering the above information * System displays the teacher details. * User will click update button to update the teacher information. * User updates teacher details and clicks save button. * Teacher details will be saved. |
| **Data** | Teacher details. |
| **Stimulus** | * Search command * Update Link * Save command in update screen |

### Enroll new students

|  |  |
| --- | --- |
| **Use case name** | **Enroll Students** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks Enroll Students link. * Admission user selects a grade level and click search button. * System displays the list of regular and IEP students admitted for the selected grade level. * Admin user will assign the students to each grade level. |
| **Data** | Grade level and student assignments will be done. |
| **Stimulus** | * Search command * Assign students to grade level command |
| **Response** | Displays the message stating that Students are assigned to the grade level. |
| **Comments** | Administrative User |

### View student records

|  |  |
| --- | --- |
| **Use case name** | **View Student Records** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View Student Records link. * System displays the search criteria to select the student. Search criteria includes the following student attributes:   + Student First Name   + Student Last Name   + Student Date of birth   + Student ID   + Student Grade level * User clicks on search button after entering the above information * The search will return one or more records and it displays the student record details in a tabular format along with an “edit” link next to each student record. |
| **Data** | Student record details. |
| **Stimulus** | * Search command * Link to select a specific student record. |
| **Response** | Displays the student information. |
| **Comments** | Administrative User |

### Updated student records

|  |  |
| --- | --- |
| **Use case name** | **Update Student Records** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View Student Records link. * System displays the search criteria to select the student. Search criteria includes the following student attributes:   + Student First Name   + Student Last Name   + Student Date of birth   + Student ID   + Student Grade level * User clicks on search button after entering the above information * The search will return one or more records and it displays the student record details in a tabular format along with an “edit” link next to each student record. * User clicks on the edit link next to student name. * User updates health and other user demographics and clicks save button. * Student details will be saved. |
| **Data** | Student record details. |
| **Stimulus** | * Search command * Save command in update screen |
| **Response** | Updated the student health information and demographic information message will be displayed in the screen. |
| **Comments** | Administrative User |

### Handle state reports

|  |  |
| --- | --- |
| **Use case name** | **Handle State Reports** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks Handle State Report link. * System displays a year drop down and generate report button. * User selects a specific year and clicks on Generate report button. * System generates CAPT (Connecticut Academic Performance Test) report for the selected year. The format of CAPT is attached below. |
| **Data** | Student and their academic information. |
| **Stimulus** | * Generate reports command |
| **Response** | The reports will be generated. |
| **Comments** | Administrative User |

### View ad-hoc reports

|  |  |
| --- | --- |
| **Use case name** | **View Ad-hoc reports** |
| **Actors** | Administrative User |
| **Priority** | Essential |
| **Description** | * Admin user clicks View Ad-hoc link. * System displays the search criteria to select the student. Search criteria includes the following student attributes:   + Student First Name   + Student Last Name   + Student Data of birth   + Student ID   + Student Grade level * System shows “Generate Report” button. Such reports are:   + Applications Status Report   + Students Grade report   + Teachers Schedule report |
| **Data** | Student, Application and Teacher information. |
| **Stimulus** | * View reports command |
| **Response** | The selected reports will be generated. |
| **Comments** | Administrative User |

### Mobile Login

|  |  |
| --- | --- |
| **Use case name** | **Mobile Log In** |
| **Actors** | Teachers, Parent/Student |
| **Priority** | Conditional |
| **Description** | After user submitting the user name and password, the system will performs the validation and displays respective mobile menu options screen based on the user type. An error message will be displayed if the user name is not found or the password doesn’t match with the one in the system. |
| **Data** | User name, Password |
| **Stimulus** | Log In command |
| **Response** | If log in is successful, system will display respective mobile menu options based on the user type. Otherwise error message will be displayed |
| **Comments** | Teachers, Parent/Student |

### Mobile Logout

|  |  |
| --- | --- |
| **Use case name** | **Mobile Log Out** |
| **Actors** | Teachers, Parent/Student |
| **Priority** | Conditional |
| **Description** | Users will be redirected to login screen once he/she clicks on Log out button or if the user is inactive for a certain length of time. |
| **Data** | Button click, or Session time out |
| **Stimulus** | Log Out command |
| **Response** | Redirects to mobile Log In screen. |
| **Comments** | Teachers, Parent/Student |

### Mobile menu options screen

|  |  |
| --- | --- |
| **Use case name** | **Mobile menu options screen** |
| **Actors** | Teachers, Parent/Student |
| **Priority** | Conditional |
| **Description** | Once users successfully logs-in, they will be redirected to the mobile menu options screen. The menu options on this screen will vary based on the user type. |
| **Data** | None |
| **Stimulus** | Teachers: Message center, Schedules, Log Out  Parent/Student: Message center, Score card, Log Out |
| **Response** | Displays mobile menu options. |
| **Comments** | Teachers, Parent/Student |

### Mobile view message center

|  |  |
| --- | --- |
| **Use case name** | **Mobile view message center** |
| **Actors** | Teachers, Parent/Student |
| **Priority** | Conditional |
| **Description** | User will click on the message center link from the mobile view menu options to see the list of all incoming messages in their inbox from other users. User selects a message link to see the details. |
| **Data** | Message ID |
| **Stimulus** | Message center command, Back command |
| **Response** | It displays tabular list of all incoming messages from user’s inbox. |
| **Comments** | Teachers, Parent/Student |

### Mobile view schedules

|  |  |
| --- | --- |
| **Use case name** | **Mobile view schedules** |
| **Actors** | Teachers |
| **Priority** | Conditional |
| **Description** | Teacher will click on the schedules link from the mobile view menu options to see the list of her/his schedule. All schedules will be displayed in the screen. |
| **Data** | Teacher ID |
| **Stimulus** | Schedules command, Back command |
| **Response** | It displays tabular list of teacher’s schedule. |
| **Comments** | Teachers |

### Mobile view score card

|  |  |
| --- | --- |
| **Use case name** | **Mobile view score card** |
| **Actors** | Parent/Student |
| **Priority** | Conditional |
| **Description** | User will click on the score card link from the mobile view menu options. Application will display the current available grades for the student. |
| **Data** | Student ID |
| **Stimulus** | Score card command, Back command |
| **Response** | Application displays the current available grades for the student. |
| **Comments** | Parent/Student |

**Student Information System  
System Architecture Design**

# System Architecture Design

## Overview

Student Information System (SIS) is a web based application which will be developed in Java/J2EE technologies and will be hosted in Glassfish application server. As SIS is web based application, it can be accessed through the pre-configured URL (uniform resource locator) from any computers and smartphones over the internet. This web application will follow the well-known industry standard MVC (Model View Controller) architecture. In addition to that, the application will be designed based on multi-tiered/multi-layered architecture. The detailed architecture diagram and its sequence diagram are depicted below:





* **Client Tier:**
  + Client tier represents the users of SIS and the different devices that are used by users to connect to it.
    - Users are: Admin Staff, Teacher, Student, Parent and Public User.
    - Potential Devices used (not limited to): Laptop, Desktop Computers, Smart phones, Tablets etc…
  + Users will access SIS by entering its URL in their browsers. Once URL is entered, browser will send the request to the web server through internet.
  + The components in this tier will be built based on the following technologies:
    - HTTP protocol will be used to send the request and receive the response from the server.
    - Industry standard web browsers will be used to access the application.
    - JavaScript will be used to handle the client side validation.
* **Presentation Tier:**
  + Presentation tier contains the View architectural components of MVC pattern.
  + The components in this layer are responsible for user interface screen display and its associated logic.
  + Once user completes their data entry and when they click Submit or Hyperlinks, the respective servlet or the methods attached to its managed bean will be invoked.
  + Apache web server will run separately and handles all web related requests such as session time out, http request and response handling etc. Apace web server will route the calls to Glass fish application server.
  + As this application receives the request from both standard web browsers and mobile web browsers, for better usability and readability experiences a user agent detector component will be developed. User agent detector will identify the user-agent of the requestor’s device (using HTTP request’s *user-agent* header attribute) and it redirects the requests to the appropriate web pages. For instance if the request is received from standard web browsers then it will redirect the request to standard web pages otherwise it will redirect the request to mobile web pages.
    - **Standard Web View Components/Pages:**
      * Standard web view components will be built based on the following technologies based on its suitable functional and technical needs:
        + HTML will be used to display the static pages.
        + JavaScript will be used to perform basic client level validation to avoid unnecessary server hits.
        + Style sheets will be used to define the consistent screen layouts, fonts, colors headers and menus across all screens of the application.
        + Java server pages (JSP) will be used to display and handle the dynamic content of the pages.
        + In JSP pages, custom tag libraries and JSTL (Java server tag libraries) will be used to handle the reusable display widgets and its data contents.
        + Java Server Faces (JSF) will be used to display and handle the dynamic content based to achieve event driven similar experience.
        + AJAX will be used to perform server side validation and pre-population of certain widgets based on the selection of the value in its previous widgets without refreshing entire web page.
    - **Mobile Web View Components/pages:**
      * Mobile web view components will be built based on the following technologies:
        + HTML will be used to display the static pages.
        + JavaScript will be used to perform basic client level validation to avoid unnecessary server hits.
        + Java server pages (JSP) will be used to display and handle the dynamic content of the pages.
        + jQuery framework and its associated style sheets will be used to design mobile web pages.
        + Firefox and its user-agent plug-in will be used to perform mobile web pages testing to identify different user agents.
        + iBBDemo2 tool will be used to test the mobile web pages from the desktop to simulate how they would function on iPhone devices.
* **Business Objects Tier**
  + Business object tier contains the Model and Controller architectural components of MVC pattern.
  + Controller components are responsible handling the request from JSP and JSF pages. Once it receives the request from the pages, based on the parameters submitted these components will control the flow of application by invoking the respective model components.
  + Controller components will be built based on the following technologies:
    - Servlet will be used to control the get and post method request from the pages.
    - Managed Bean will be used to control the request from JSF pages.
  + The creation of java beans reside in business objects tier. Controller components will interact with database while performing database operations (create, read, update and delete) through java beans using Java Persistence API. During this process, Entity Manager Factory, Persistence Unit and User Transaction classes will be used to perform database operations.
  + Model components are responsible handling the request from Servlets and Managed beans. Model components represent the real time physical entities such as School, Student, Teacher and logic business entities such as Enrollment, etc…In addition to that model components also contain any specific rules that are associated to its business process workflow.
  + Model components will be built based on the following technologies:
    - Managed Entity Bean – This java object represents the characteristics of real life physical objects with its appropriate getter and setter method to access those characteristics.
    - The respective Annotations will be used to map the managed bean to the physical database table.
* **Integration Tier**
  + Integration tier contains the components and framework that are used to connect the information tier in SIS application.
  + Integration tier components will be built based on the following technologies:
    - Java Persistence API framework will be used to map and populate the Student data model into java object models.
    - Entity Manager and User transactions objects will be used to handle user transactions.
* **Information Tier**
  + Information tier contains the database components that are used in SIS application.
  + Student database will be used to maintain the business data and audit data needed to meet SIS application’s business requirements.
  + In addition to that the meta data and configuration data needed for SIS application will be maintained in the database tables.
  + The student database will contain various tables such as:
  + Derby database will be used to build the Student database.
* **Distribution Tier**
  + Distribution tier contains the set of reusable components that are used in SIS application across various tiers. The list of reusable components include:
    - File Handler components
    - Message board handler components
    - Validation Utility components

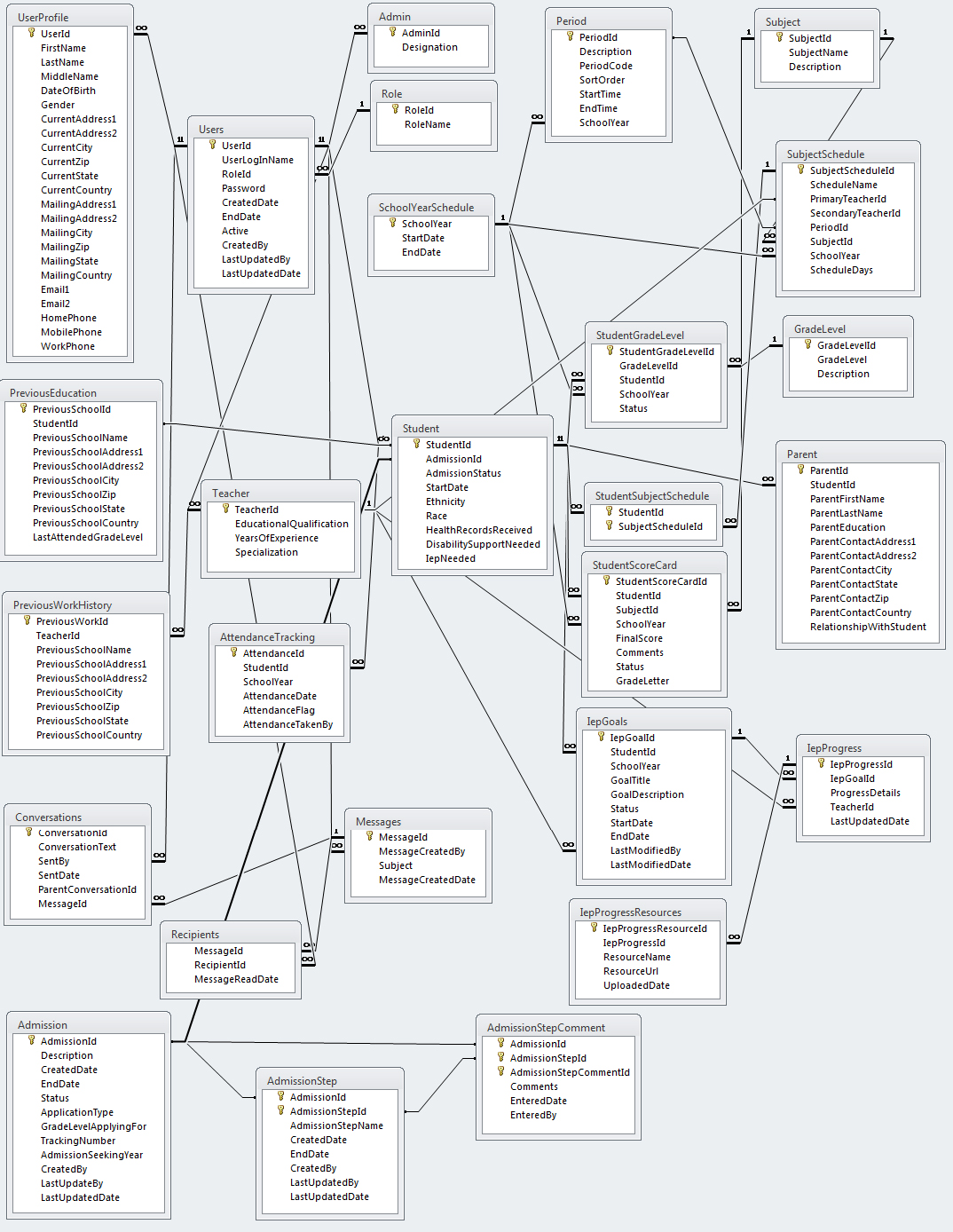
## System Architecture Design – Non-functional characteristics

* **Scalability:**
  + Due to the layered architecture, components in each layer are loosely coupled. With this architecture, the application can be scaled (both vertically and horizontally) with minimal configuration without modifying the application’s code.
* **Performance:**
  + All the screens will be rendered within 2-3 seconds.
* **Manageability:**
  + Application will be designed and developed based on the industry standard J2EE design patterns and coding guidelines.
  + Configuration and Meta data information of SIS application will be stored in database tables. Due to this, application will be configured and maintained easily through these configuration tables without any code changes.
* **Usability:**
  + Consistent style sheets will be used for better usability experiences.
  + Based on screen layouts and functions, the respective buttons and links will be displayed.
* **Security:**
  + SIS application handles and process student information. This is one of the sensitive information. So, SSL certificates will be used to secure SIS application.
  + In addition to SSL, Users of this application will be provided with an appropriate user id and password to access the application.

**Student Information System  
Database Schema Design**

# Database Schema Design

## Entity Relationship Diagrams



## Database Table Descriptions

|  |  |
| --- | --- |
| **Table Name** | **Table Description** |
| Admin | Admin is a type of user to SIS application. This table extends the user table. This table is used to maintain admin specific information. |
| Admission | This table is used to maintain and track the admission workflow. |
| AdmissionStep | This table is used to maintain and track the admission workflow steps. |
| AdmissionStepComment | This table is used to maintain and track the admission workflow step comments entered by the user. |
| AttendanceTracking | This table is used to track the attendance of the student in a given grade level for a particular day. |
| Conversation | This is table used for keeping track of all conversations between the uses by message id. |
| GradeLevel | This master table is used to maintain the grade level offered. |
| StudentSubjectSchedule | This table maintains the student subject enrollments. |
| IEPGoals | This table is used for storing goals and objectives of all IEP students. |
| IEPProgress | This is table is for keeping track of IEP student progress. |
| IEPProgressResources | Table for storing the shared resources of IEP students. |
| Messages | This is table contains the list of all composed message titles. |
| Parent | This table is used to maintain the parent details of the student. |
| Period | This master table is used to maintain periods and its timing of the school for a given school year. |
| PreviousEducation | This table is used to maintain the prior educational history of the student. |
| PreviousWorkHistory | This table is used to maintain the prior work history of the teacher. |
| Recipients | This table is used for keeping track of all recipients in the message center conversations. This table also keeps track of read and unread messages. |
| Role | This table is used to maintain the role of the users of SIS application. The different roles of this application are: Student, Teacher and Admin |
| SubjectSchedule | This table is used to maintain the schedules for each subject by teacher and period. |
| SchoolYearSchedule | This master table is used to maintain the school year and its corresponding start and end date for the school. |
| Student | Student is a type of user to SIS application. This table extends the user table. This table is used to maintain student specific information such as Race, Ethnicity, and Health records etc... |
| StudentGradeLevel | This table is used to maintain the list of students enrolled for a given grade level. |
| StudentScoreCard | This table is used to maintain the scores of the subject for each student. |
| Subject | This table is used to maintain the list of subjects taught in the school. |
| Teacher | Teacher is a type of user to SIS application. This table extends the user table. This table is used to maintain teacher specific information such as educational qualification. |
| Users | This is a generic user table to track the users of the application. |
| UserProfile | This table is used to maintain user personal and demographic information. |

**Student Information System  
System Documentation**

# System Documentation

## Rationale for selected technology:

During the initial research, it was learned that most of the school systems are built based on client server architecture. Due to this, students, teachers and administrative staffs don’t have an option to access the school system without logging into the school network/building. Because of this, they cannot be able to access any of their school records whenever they need. Students/parents, teachers and administrative staffs are looking for a system that can be accessed by 7\*24 from anywhere in the world and through any devices such as Laptop, Desktop, PDA, iPad and Smartphones. At the end of the research, it was evident that there was a need for a web system that should be available at any time. So, it was decided to build a distributed web based system that automates school’s academic process. The design for mobile web version of this application was also considered by tweaking the changes on the view layer without changing model and control layer components.

As we decided to develop a SIS as a Web based application, we considered two technologies suites and those are J2EE/Derby and ASP/.Net/SQL server technologies. Even both technologies provide the similar capabilities; we decided to proceed with J2EE/Derby technologies due to the following reasons: (1) as part of our web programming coursework, we learned J2EE and its related concepts. We would like to utilize those technological skills in the real time application. (2) Familiarity of the technology to complete the project within a given timeframe (3) Built-in Mobile enhancement capabilities.

The application is designed and developed based on MVC (Model view and controller) architecture. In general, the following design approach is followed throughout the application. Each and every screen is designed and developed based on XHTML facelets with JSF. The rationale for selecting JSF over JSP is to leverage JSF’s built-in binding features to the java model beans along with its corresponding validation framework. In the screens, data grids are bound to the associated java model beans to render the screen, form controls are bound to the associated java object model to populate user entered values into the java objects and the events are mapped to the corresponding managed bean controller’s operation. In the controllers, Java Persistence API is used to persist the values that are in java objects into the database. Derby database is used to store and retrieve SIS application data. In addition to that, Rich faces framework is used within JSF facelet for calendar controls and rich text editor throughout based on the need. JSF resources are used (CSS) to maintaining consistency throughout the site. Web filters is used in Java Servlet’s Web Filter with URL patterns to protect secured pages. In the report modules, to export the report content into excel the JSP pages are used with “application/vnd.ms-excel” MIME type.

## System setup and Installation Instruction:

Using the set of instructions provided below, any technical person will be able to install and set this application along with source code and database.

### Required Software

* NetBeans IDE 7.2.1 with JDK 1.6 or higher version
* Derby database
* Glass Fish server or any Industry standard equivalent application servers.

### Database Setup Instructions

* Create blank database in derby, with the following names:
  + Database name = **SISDB**
  + User=**sisadmin**
  + Password=**sisadmin**
* Once the database is created, install the schema from the file “install-database.sql”. Please note that if there are any issues occurred during the installation, please execute “uninstall-database” and retry. The install and uninstall database script files are embedded below.



* Install database sql scripts will create the database schema (DDL), referential constraints and set of rows in the database tables. It also inserts 3 rows in user table to login to the SIS application. Those users are:
  + Admin User (user name = **admin** & password = **admin**)
  + Teacher User (user name = **teacher** & password = **teacher**)
  + Student User (user name = **student** & password = **student**)

### SIS Web application Setup Instructions

* Extract SIS source files from the sis2013.zip file provided.
* Open NetBeans IDE and open SIS web project (extracted in the previous step\_ in Netbeans IDE using Open Project option.
* The data source name used in the application is “**jdbc/SISDB**” and name of the persistent used is “**SISPU**”. Please create the data source and persistent units with the same name.
* Once project is opened successfully, build the project using clean and build option in NetBeans.
* Then locate /Webpages/index.xhtml, right click and choose Run File option. This will load Home page of SIS application.
* Use any one of the following user name and password to view admin, teacher and student modules.
  + Admin User (user name = **admin** & password = **admin**)
  + Teacher User (user name = **teacher** & password = **teacher**)
  + Student User (user name = **student** & password = **student**)

## System Features:

Once Student Information System (SIS) application and Student database is configured successfully SIS application can be accessed through any standard web browsers using the following URL if SIS is deployed in desktop’s application server instance. <http://localhost:8080/SIS/faces/index.xhtml>. This URL displays the home page of SIS application. By default, application’s database script creates an administrator user with user name as “admin” and password as “admin”. With this admin user id and password, administrator can be able to login to SIS application. SIS application contains the following sub functionalities based on the role of the user.

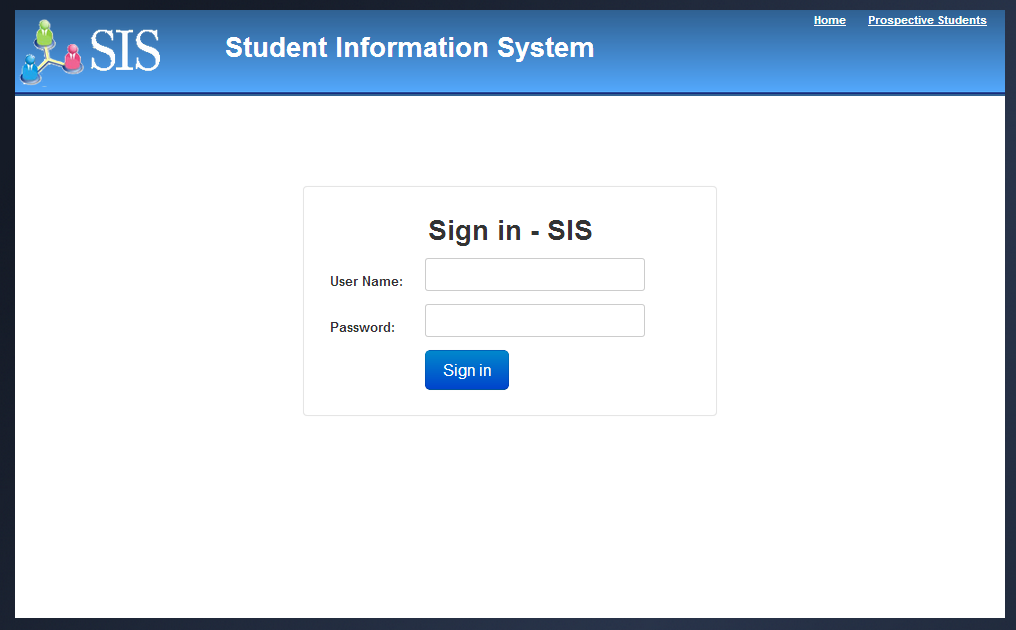
**Common & Public User Modules**

### Common modules

Common functionalities of SIS application are accessed by all authorized users of SIS application. They can be able to access these functions through the links provided in the top right side of each screens. Common modules include the following functions:

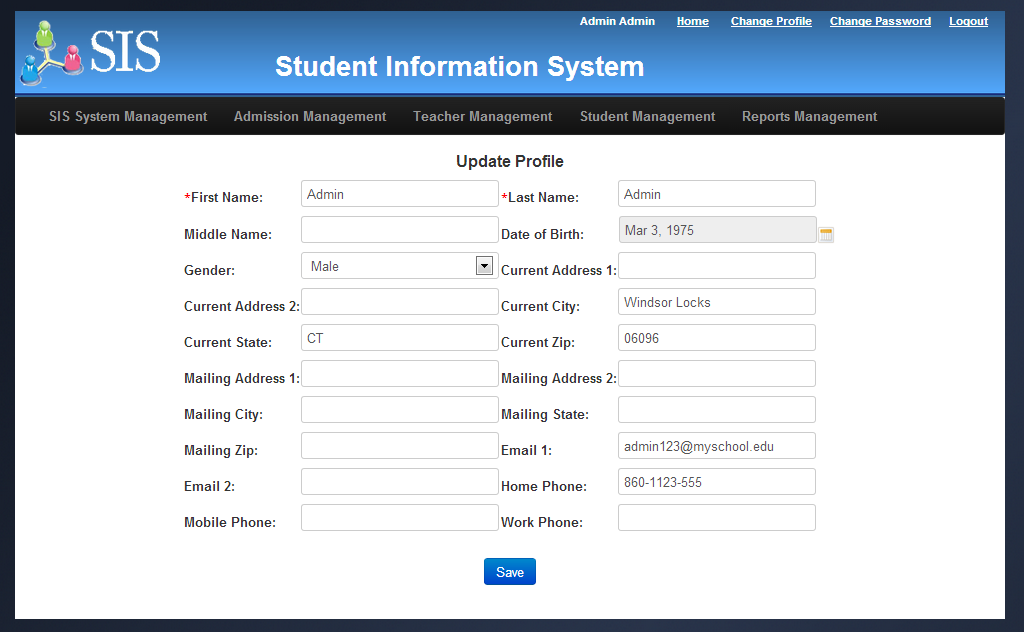
* **Login:**

This screen provides the ability for Teachers/Students to login to SIS application based on the valid user login name and password provided by Administrator. System automatically creates unique user login name and password for each teacher whenever a new teacher is added. For students, it will be created once their admission is granted. Administrator communicates user login name and password to the respective teachers and students after user login name and password is created by the system. System will display an error message if users tried to login with invalid user name and password.



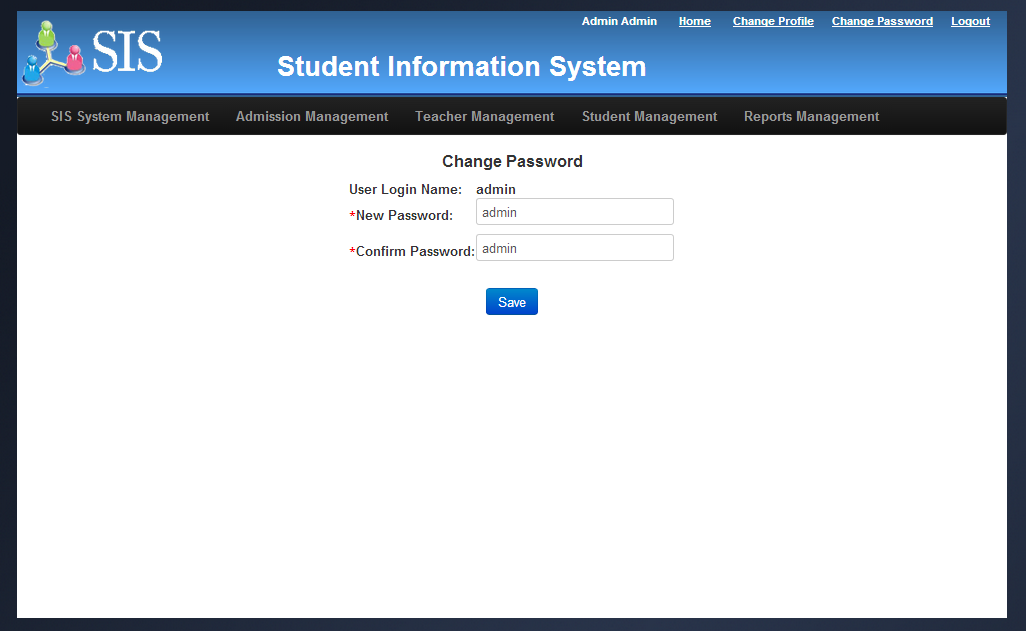
* **Change Profile:**

Once user clicks Change Profile link, change profile screen will be displayed in editable mode to update profile information including contact address, email and phone. If user clicks save button then the updated profile information will be stored.



* **Change Password:**

Once user clicks Change Password link, change password screen will be displayed in editable mode to update password. System will display new password and confirm password fields. If user clicks save button then the old password will be replaced with new confirmed password and “Password has been changed successfully” message will be displayed in the screen.



* **Logout:**

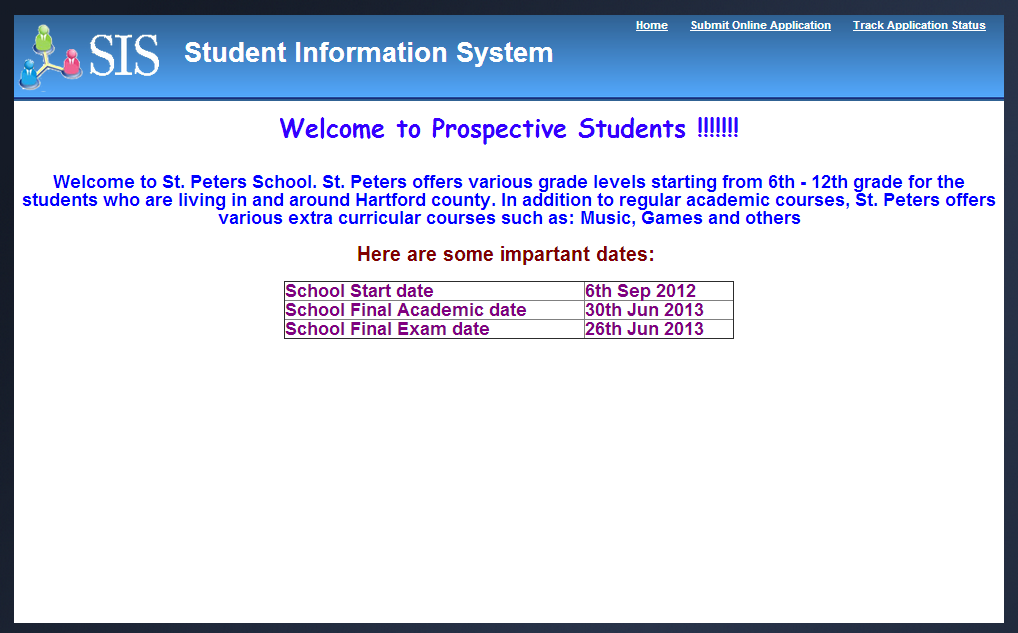
Once user clicks Logout link, users will be logged out from the system and they will be redirected to the login page.

### Public user modules

Public user’s modules/pages will be accessed by any public users. These screens will be accessed by users without logging into the system.

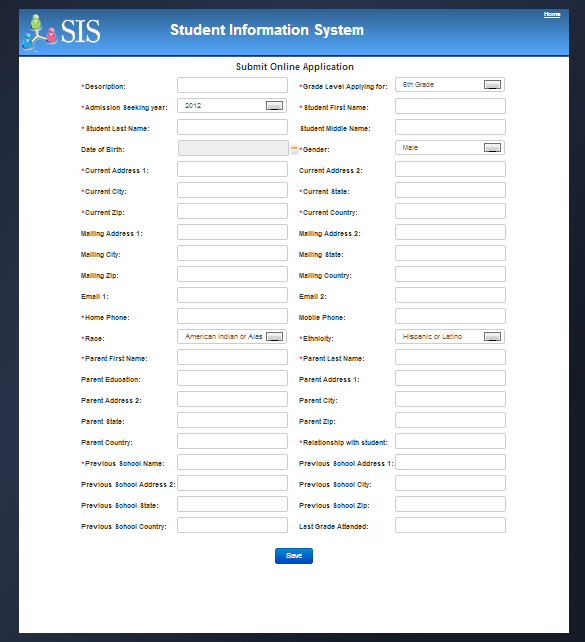
* **Prospective Students:**

Once user clicks Prospective Students link in the home page, the prospective student page will be displayed. The prospective student page will display the details about school information, school features and important academic year dates. The content of this screen will be managed by administrator. So, they have a flexibility to update the contents of this page periodically based on the important events.

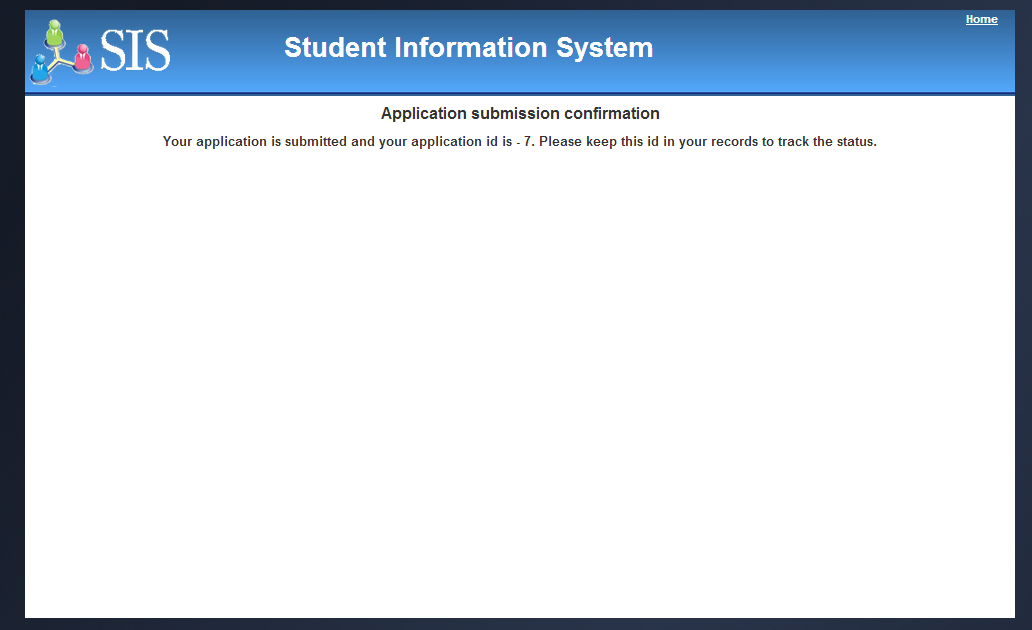


* **Submit Online Application:**

If students want to apply for the school for a specific grade level and academic year then they can be able to apply online through Submit Online Application link provided in the prospective page. Once users click Submit Online Application link, submit online application screen will be displayed. This screen gathers the student personal, demographic information, academic year and grade level for which students are seeking admission etc…Once the application is submitted, system initiates Admission Workflow and puts application/admission in in-progress status and it will be listed in administrator’s queue for review and next steps. Once the applications are submitted successfully system will provide a tracking number to the users. Using this tracking number, students can be able to track the status of their application at any point in time during the admission process using Track Application Status link provided in the Prospective Students page. Also, the applications that are received through this page will be treated as “online” application type for operational and tracking purposes.

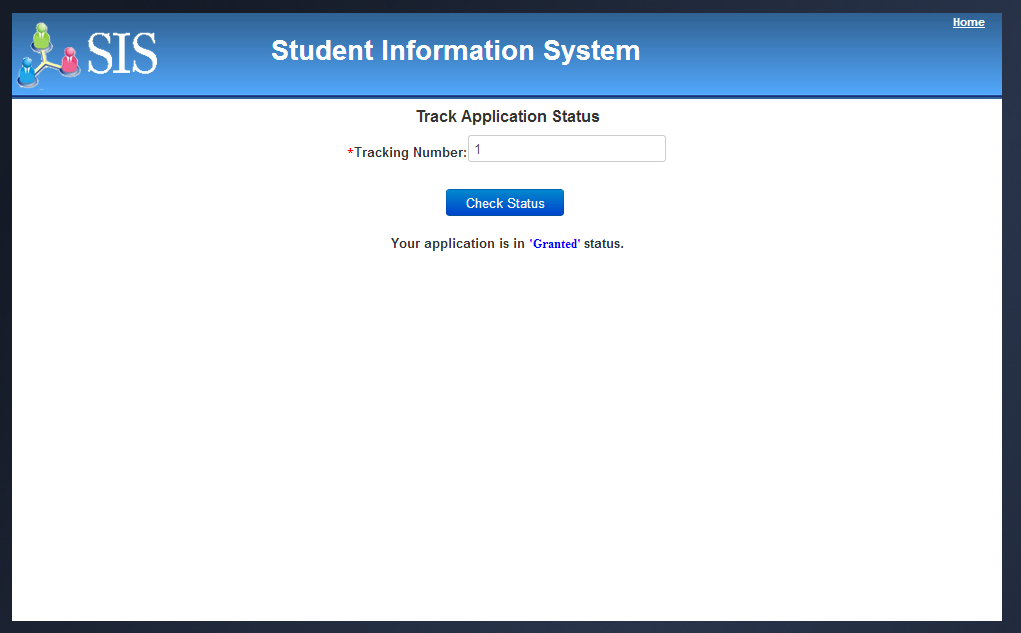


**Application Confirmation page:**



* **Track Application Status:**

Once users click Track Application Status link, system will display track application status page. Once user enters tracking number and clicks Check Status button, the screen will display the status of the application based on the tracking numbers. The possible statuses expected are: In-progress, Granted and Rejected.

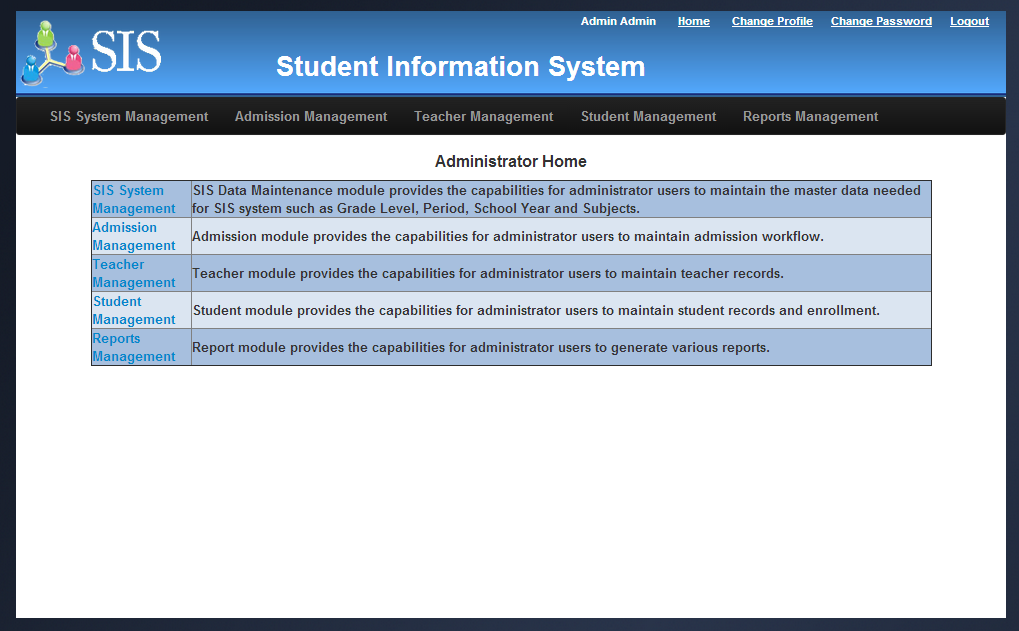


**Administrator Module**

### Administrator Module

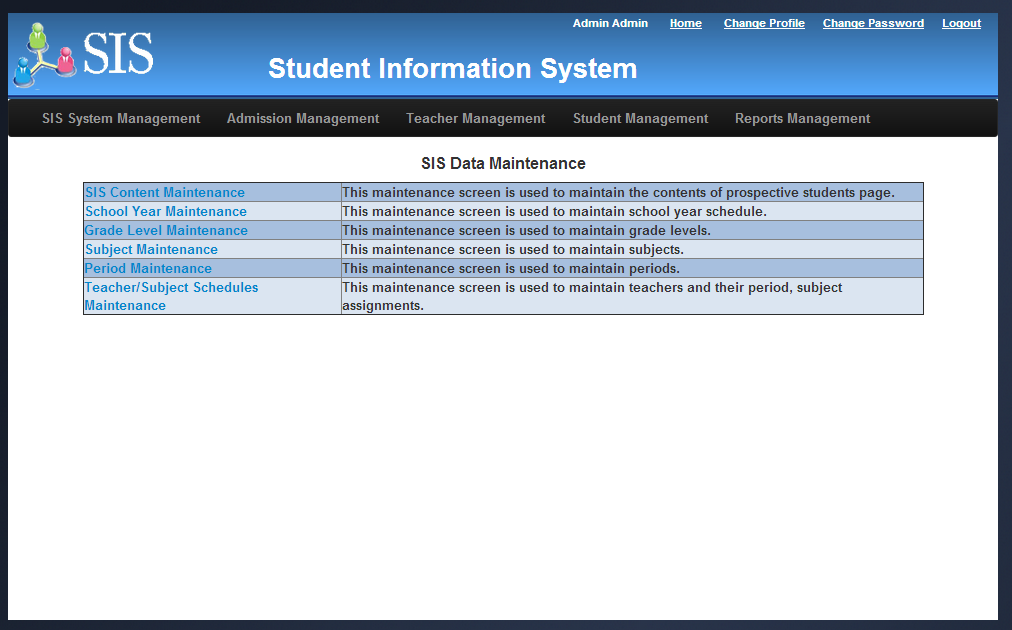
#### Administrator Home page

Admin users will be redirected to Administrator home page as soon as they login to the application using their user name and password. Administrator home page will provide the various options for administrators to perform various administrative related functions that support school’s academic process on a daily, monthly and/or yearly basis. In addition to that, Administrator will manage teacher records, student records student admission, student enrollment and various reports through the set of links provided in Administrator home page.



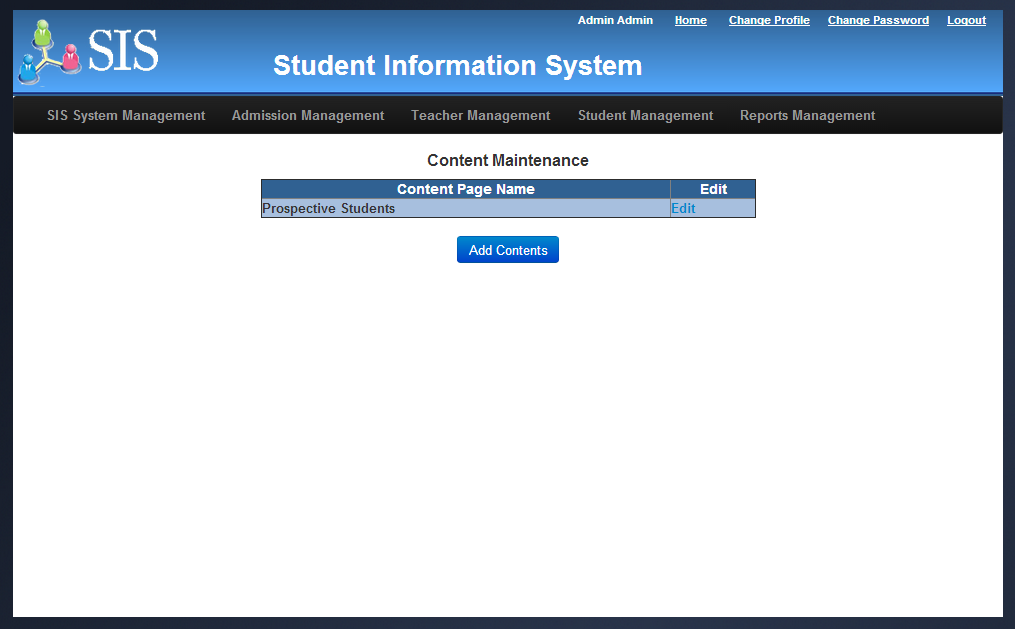
#### SIS System Management

Once user clicks, SIS System Management link in Administrator Home page, SIS System Management home page will be displayed. This page will contain the links to various academic data setup/maintenance pages that includes School year, Grade Level, Subject, Period and Teacher/Subject schedules.

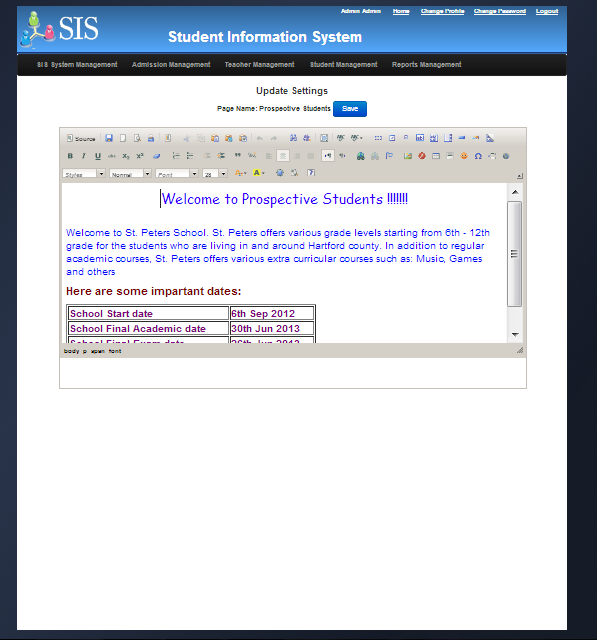


##### SIS Content Maintenance

Once user clicks SIS Content Maintenance link in SIS System/Data Management home page, SIS content maintenance screen will be displayed. This screen will display the data grid that contains the list of content pages that are configured with content text and an Edit option for each page. Currently SIS application maintains the content text only for Prospective Students. If there are no content text is configured for any pages then the data grid will not display any rows.

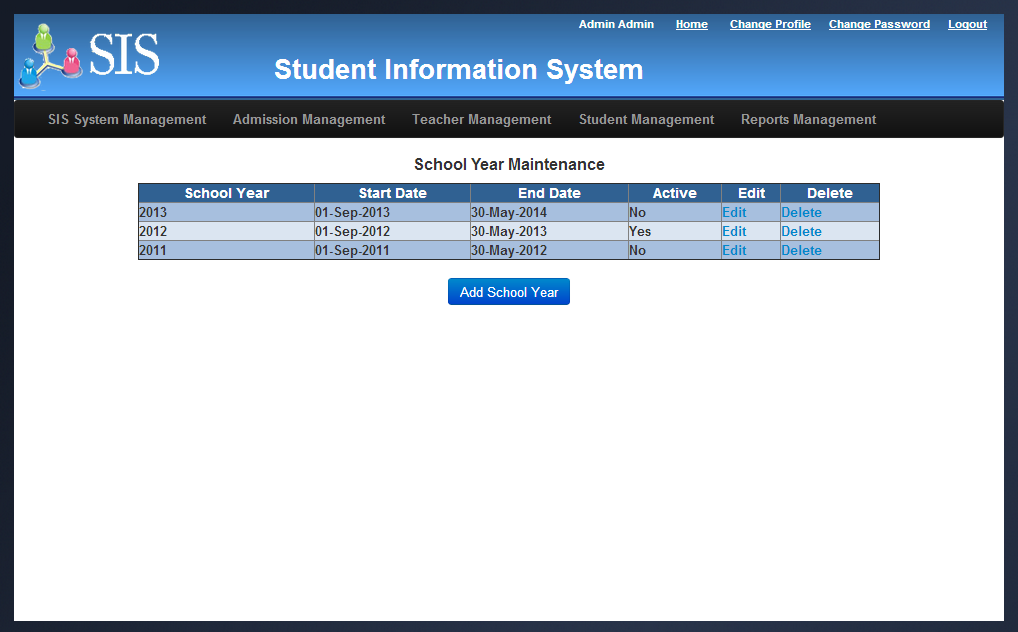


Using “Add Contents” Button, Administrator will add the contents using the rich text box provided in the Add Contents page. Using “Edit” link, Administrator will update the previously entered contents using the rich text box provided in the Update Contents page. The updated contents will immediately available in the prospective student page without any code changes and deployment.



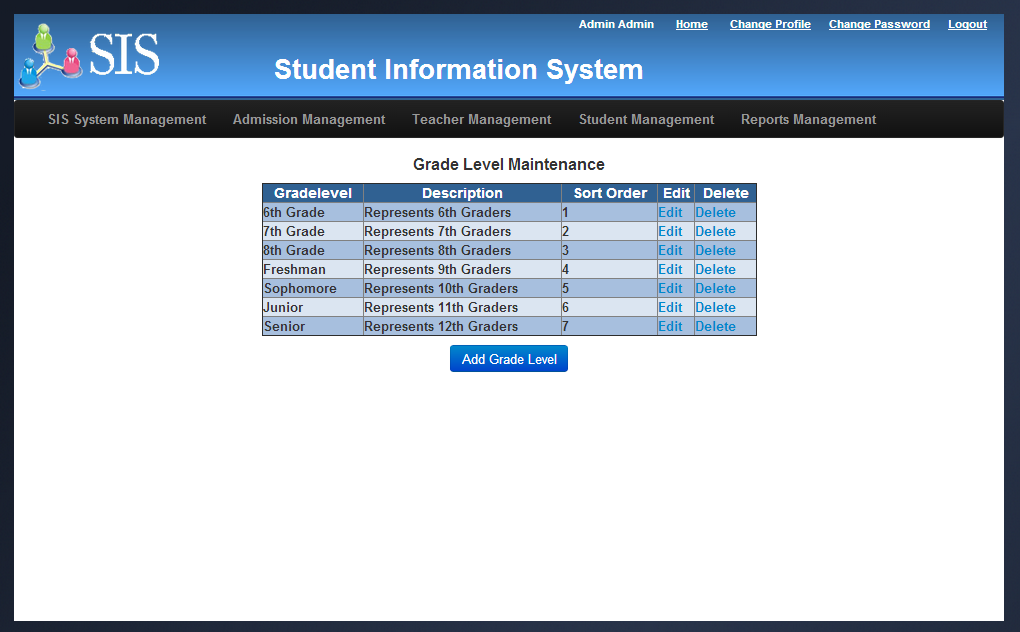
##### School year Maintenance

Once user clicks School Year Maintenance link in SIS System/Data Management home page, School Year Maintenance page will be displayed. This screen will display the data grid that contains the list of school years. In each row Edit and Delete option will be displayed. It will be used to update and delete school years. Add School Year button will be used to add the new school years. At any point in time, only one academic school year will be active. If Administrator tried to create a new active school year if an active school year already exists then system will display an error message.



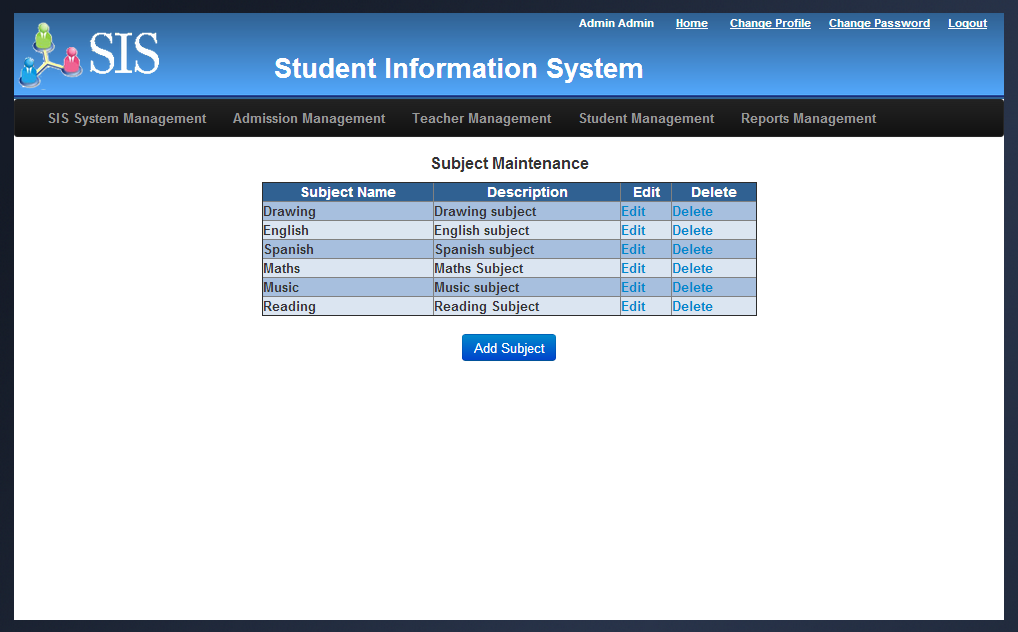
##### Grade Level Maintenance

Once user clicks Grade Level Maintenance link in SIS System/Data Management home page, Grade Level Maintenance page will be displayed. This screen will display the data grid that contains the list of grade levels. In each row Edit and Delete option will be displayed. It will be used to update and delete grade levels. Add Grade Level button will be used to add the new grade levels. Sort order attribute in the grade level will define the sequence of grade level (lower to higher) that helps system to move the students to next grade level during the enrollment process. Sort order will be unique at the grade level. If Administrator tried to create a new grade level with an existing sort order then the system will display an error message.



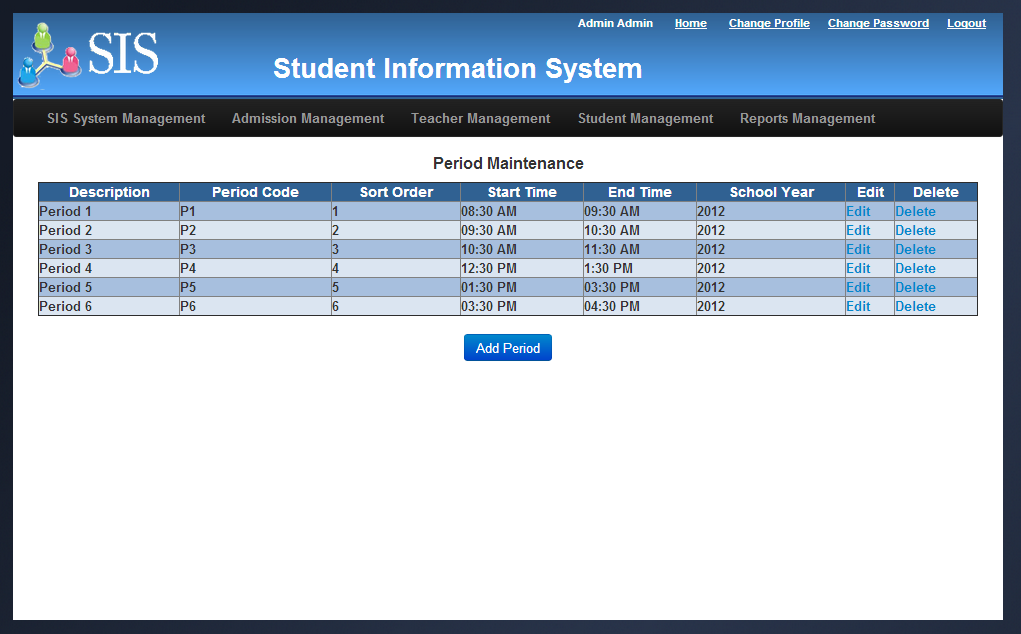
##### Subject Maintenance

Once user clicks Subject Maintenance link in SIS System/Data Management home page, Subject Maintenance page will be displayed. This screen will display the data grid that contains the list of subjects taught in the school. In each row Edit and Delete option will be displayed. It will be used to update and delete subject information. Add Subject button will be used to add the new subjects.



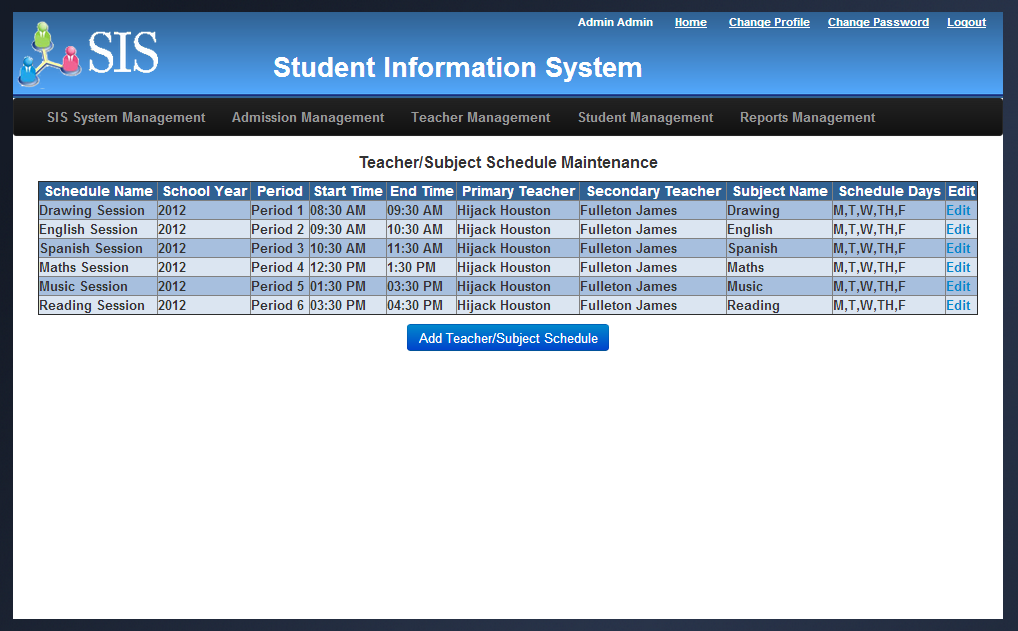
##### Period Maintenance

Once user clicks Period Maintenance link in SIS System/Data Management home page, Period Maintenance page will be displayed. This screen will display the data grid that contains the list of periods that are configured for the school year. In each row Edit and Delete option will be displayed. It will be used to update and delete periods respectively. Add Period button will be used to add the new periods. Sort order attribute in the period will define the sequence of period. Sort order will be unique for a period and school year. If Administrator tried to create a new period with an existing sort order then the system will display an error message.

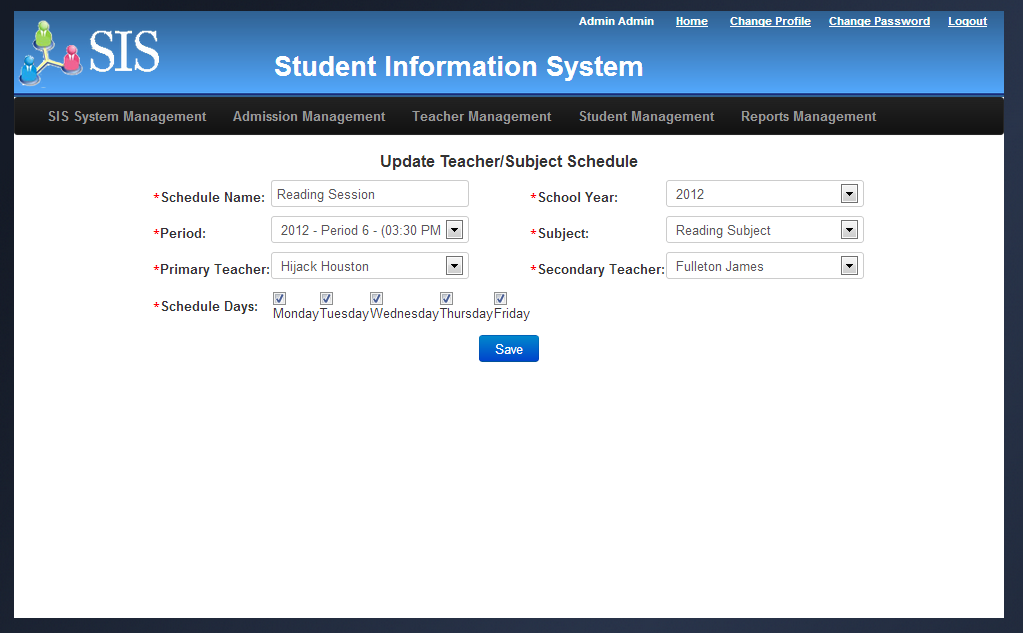


##### Teacher/Subject Schedule Maintenance

Once user clicks Teacher/Subject schedule Maintenance link in SIS System/Data Management home page, Teacher/Subject schedule maintenance page will be displayed. This screen will display the data grid that contains the list of teacher/subject schedules that are configured for the school year. In each row Edit option will be displayed. It will be used to update the teacher and subject schedules. Add Teacher/subject schedule button will be used to add the assign teachers to the specific subjects and periods. During the primary and secondary teacher assignment, if the user selects the same primary teacher and secondary teacher for the schedules then system will display an error message and prompts the administrator to select the different primary or secondary teacher. Also, if the selected primary or secondary teachers are already assigned to the same period then system will display an error message and prompts the administrator either to select the different primary/secondary teacher or different period.

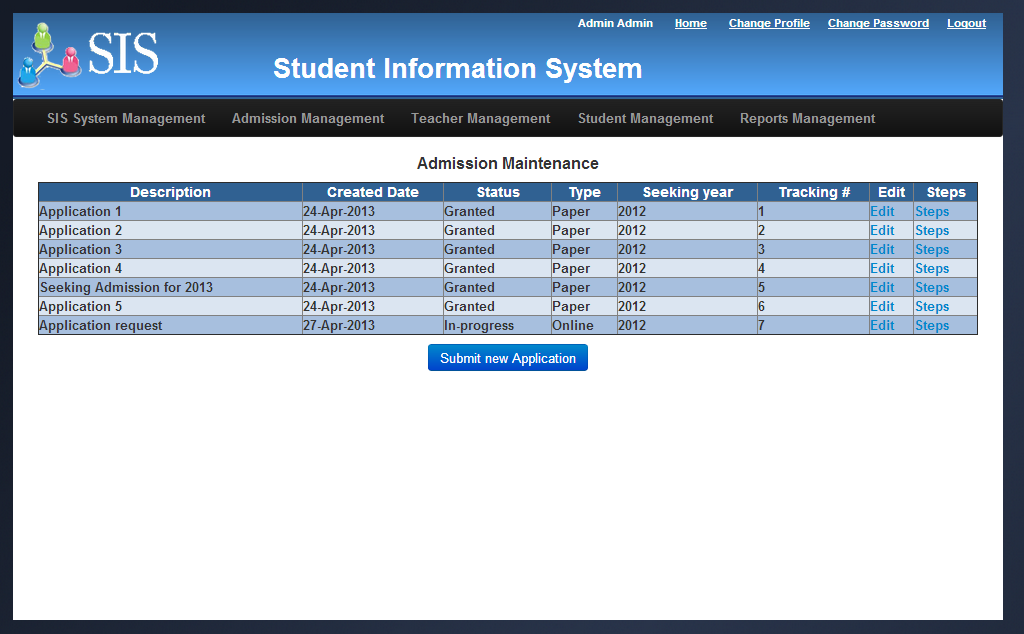


**Update Teacher/Subject Schedule Screen:**



#### Admission Management

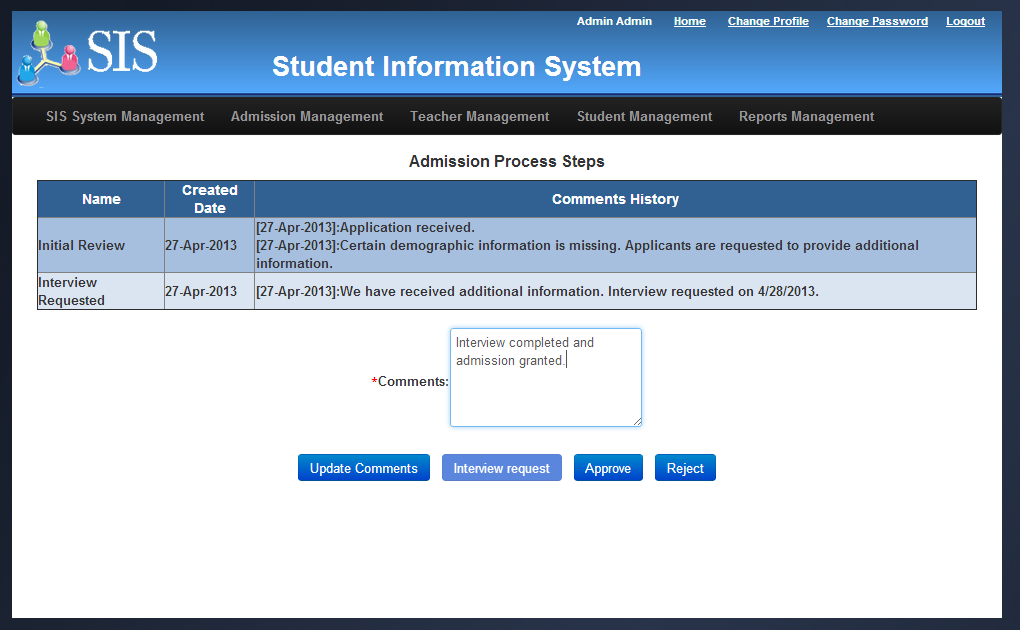
Once user clicks Admission Management link in Administrator Home page, Admission maintenance page will be displayed. This screen will display the data grid with the list of applications. The data grid contains the application description, application created date, status of the application in the admission workflow, application type and tracking number. Screen displays both application types such as “Online” (submitted by public users) and “Paper” (created by administrator based on the manually submitted paper application by the students or parents). Each row has Edit and Steps link that are used to update the application information and track/process the admission process respectively. Submit New Application button will be used to create a new application.

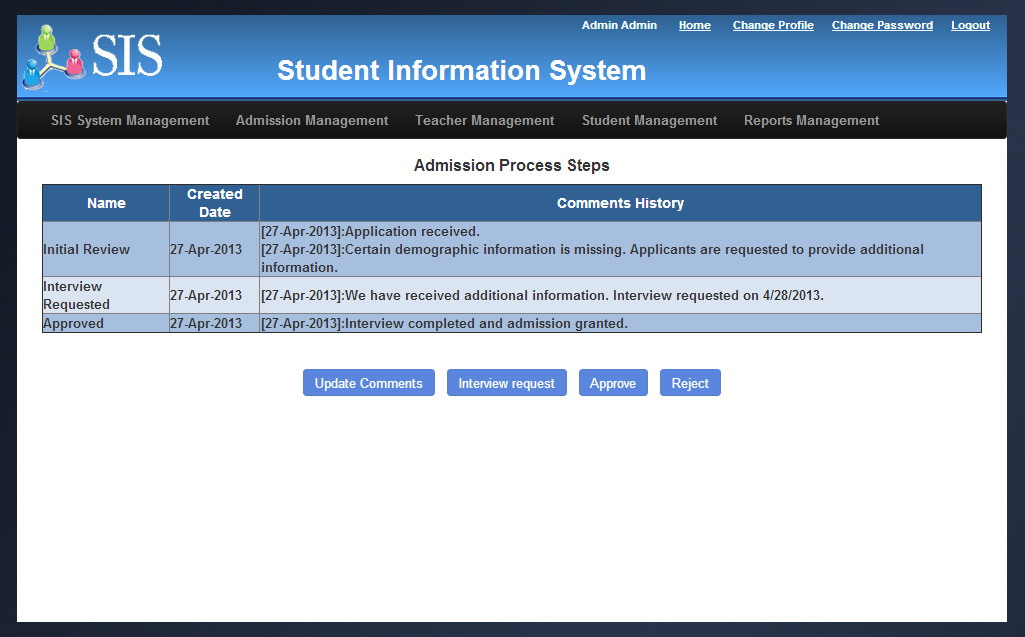


Once user clicks Steps link for any specific admission record in the data grid, system will display Admission Process Step screen. If the selected Application is in Granted or Rejected status then this screen will be displayed in read-only mode, otherwise this screen will be displayed in editable mode. In this screen, administrator will take various actions in admission workflow process. Administrator will either update any additional comments/notes or request and schedule Interview or approve the application or reject the application.

If administrator clicks on Approve button then admission workflow process will be completed and the status of the admission workflow will be moved to Granted status. Student associated to the Granted application will be eligible for grade level enrollment and they will be shown in Student Grade Level enrollment page. Also, system automatically creates unique user name and default password as “password” for the eligible students once their admission is granted which can be used by student to login to SIS application. The system generated user name will be displayed in the student maintenance screens.

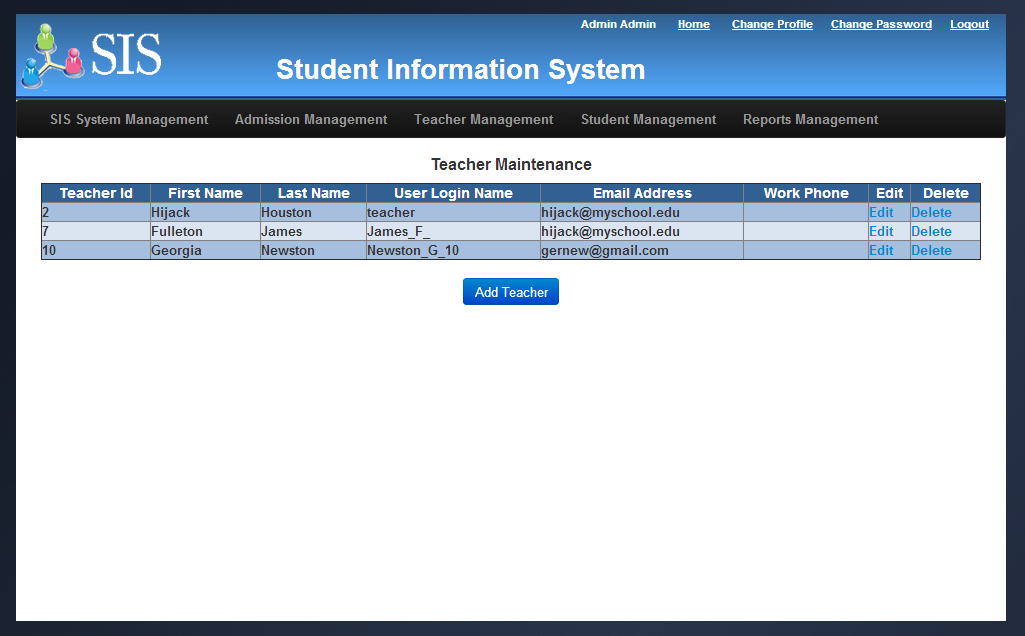
If administrator clicks on Reject button then admission workflow process will be completed and the status of the admission workflow will be moved to Rejected status.





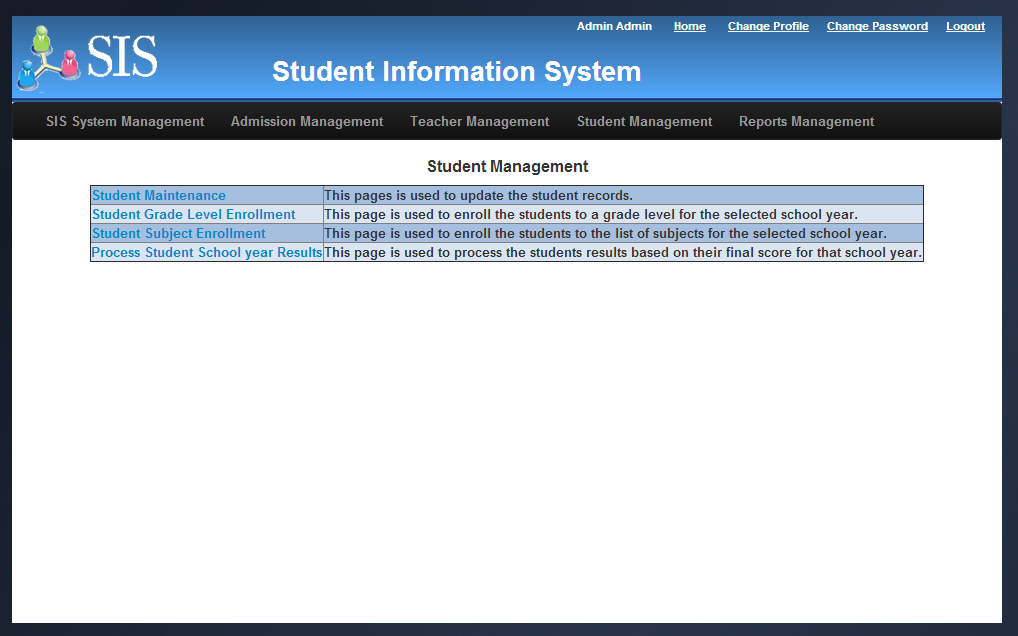
#### Teacher Management

Once user clicks Teacher Management link in Administrator home page, Teacher maintenance page will be displayed. This screen will display the data grid that contains the list of teachers those are working in the school. In each row Edit and Delete option will be displayed. It will be used to update the teacher information. Add Teacher button will be used to add new teachers.



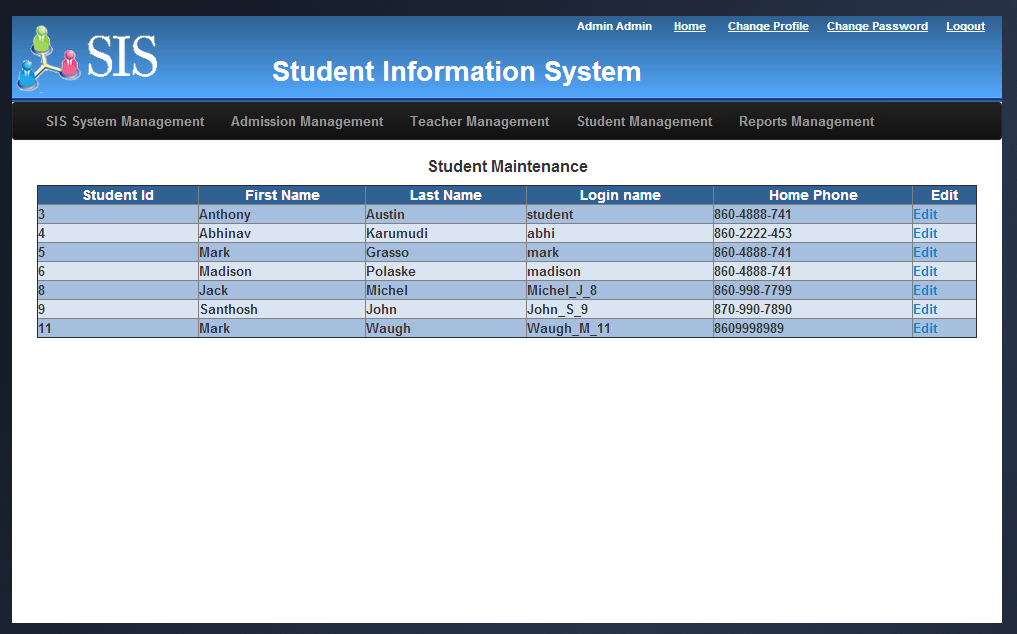
#### Student Management

Once user clicks Student Management link in Administrator Home page, Student Management home page will be displayed. Using Student Management option, administrator will perform student records related activities such as maintaining and updating student records, enrolling students to grade level, subjects and processing student’s final results at the end of the school year.



##### Student Maintenance

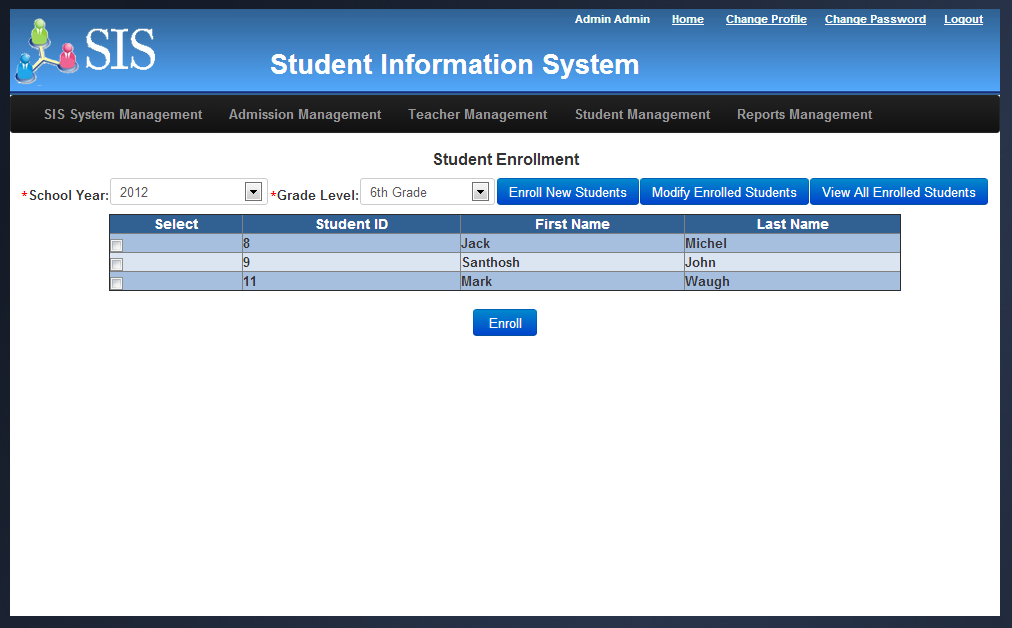
Once user clicks Student Maintenance link in Student Management home page, Student maintenance page will be displayed. This screen will display the data grid that contains the list of students those are studying in the school. In each row Edit option will be displayed. It will be used to update the student records.



##### Student Grade Level Enrollment

Once user clicks Student Grade Level Enrollment link in Student Management home page, Student grade level enrollment page will be displayed. This screen will be displayed with school year and grade level pick lists along with Enroll New Students, Modify Enrolled Students and View All Enrolled Students.

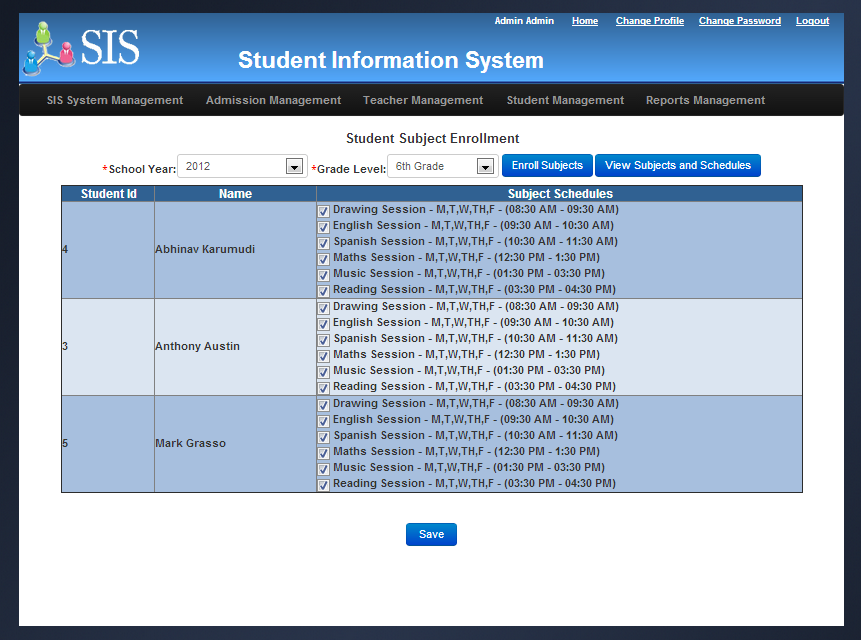
Once user clicks the Enroll New Students button, the system will display the list of new students those admissions are granted for the selected school year/grade level and the list of existing students those are passed previous grade level in the previous academic year. Administrator will select the students using the check box provided in each row and enroll them to the selected grade level by clicking Enroll button. Using Modify Enrolled Students button, administrator will modify and update the enrolled students. Using View All Enrolled Students button, administrator will view all the enrolled students for the selected school year and grade level.



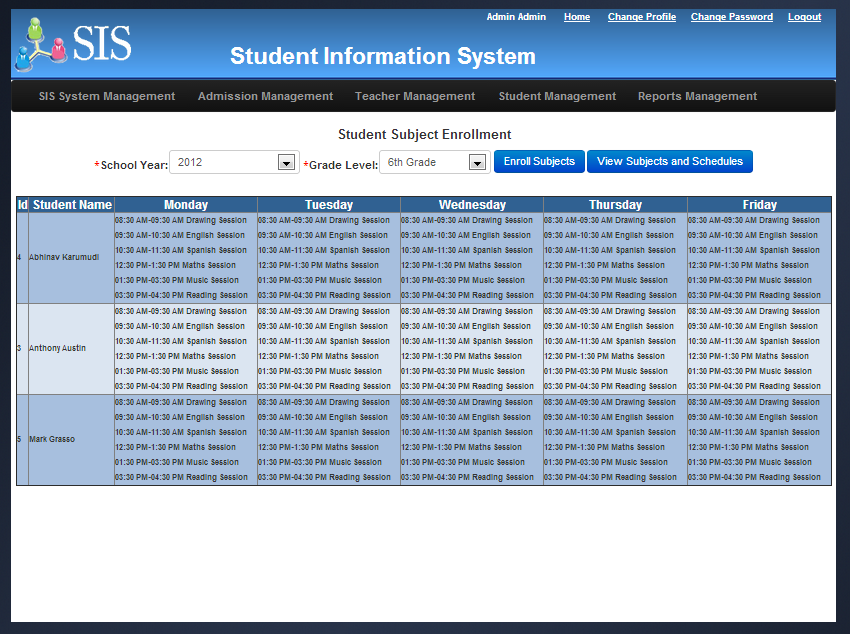
##### Student Subject Enrollment

Once user clicks Student Subject Enrollment link in Student Management home page, Student Subject enrollment page will be displayed. This screen will be displayed with school year and grade level pick lists along with Enroll Subjects and View Subjects and Schedules.

Once user clicks the Enroll Subjects button, the system will display the list of students enrolled for the selected grade level. Administrator will enroll the students to the specific subjects by clicking save button. Using View Subjects and Schedules button, administrator will view all the enrolled students and their corresponding subjects and schedules.



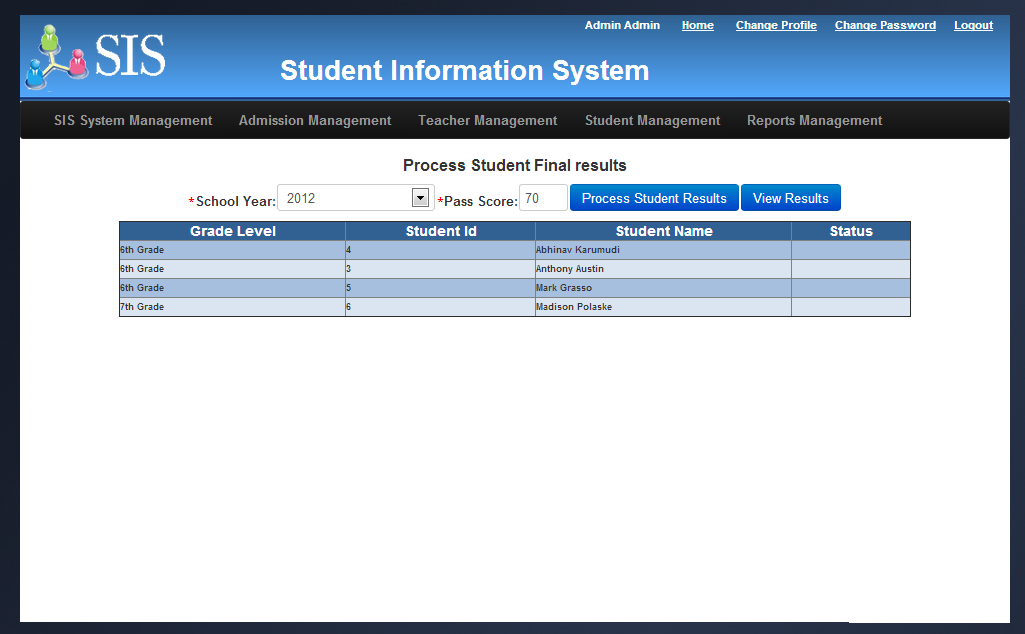
**View Subjects and Schedules Screen:**



##### Process student school year results

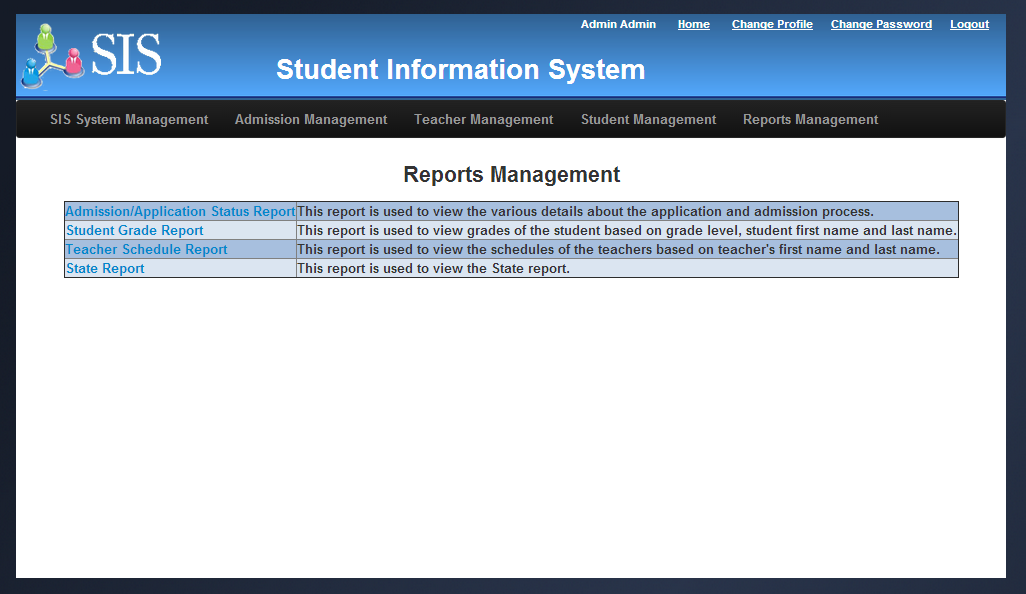
Once administrator clicks Process Student School year Results link in Student Management home page, Process Student School year Results page will be displayed. This screen will be displayed with school year and pass score input field along with Process Student Results and View Results buttons.

Once administrator clicks the Process Student Results button, system will identify all the students those final scores are greater than or equal to the pass score entered in the screen in their respective enrolled subjects. All such student’s status will be updated as “PASS” for that grade level and they will become eligible for next grade level in the subsequent academic year. The remaining student’s status will be updated as “FAIL” and they will not become eligible for next grade level. Administrator will perform this function towards the end of the school year once all the final scores are updated into the system by all the teachers for all the students. Using View Results, administrator will view the results of all the students for the selected school year.



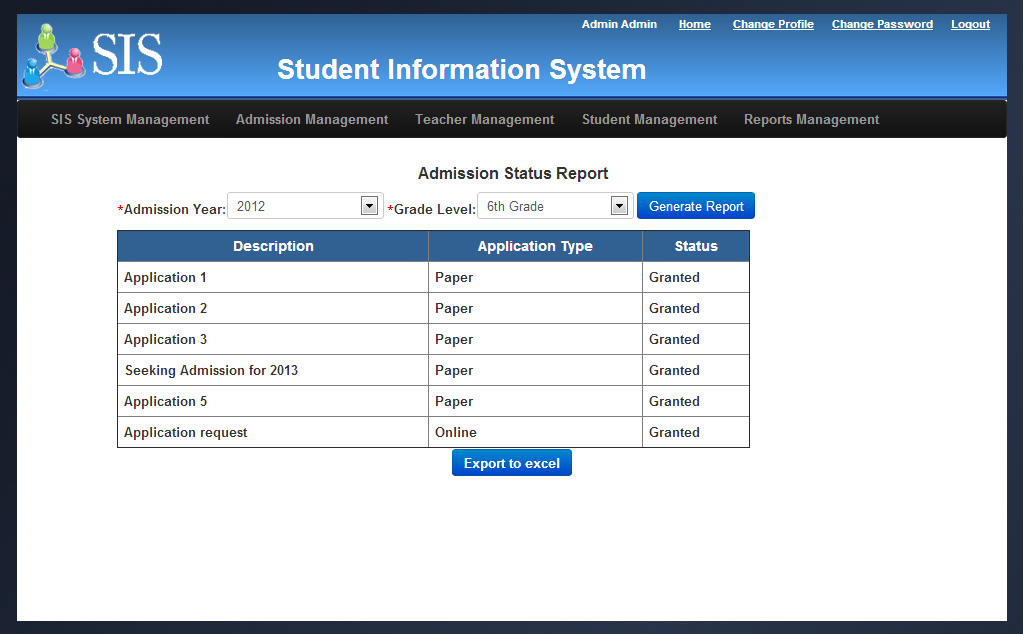
#### Reports Management

Once user clicks Report Management link in Administrator Home page, Reports Management home page will be displayed. This page contains the links to various reports including Application/Admission Status report, Student Grade report, Teacher schedule report and state report.



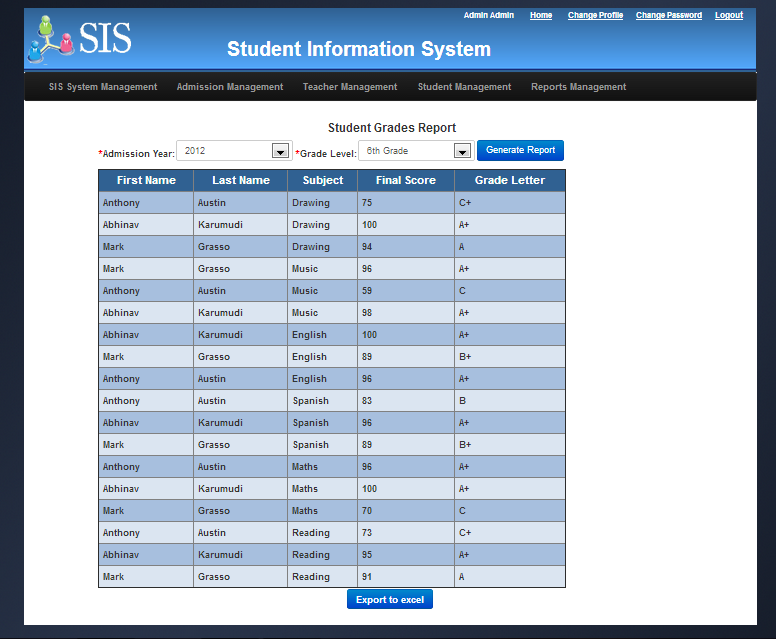
##### Application/Admission Status report

Once user clicks Application/Admission status report link in Reports Management Home page, the application/admission status report page will be displayed with Admission year, Grade level pick lists and Generate report button. Once user clicks Generate Report button, the list of applications and its corresponding statuses will be displayed based on the selected admission year and grade level. Once the report is generated, users can be able to export it to excel using Export to Excel button.



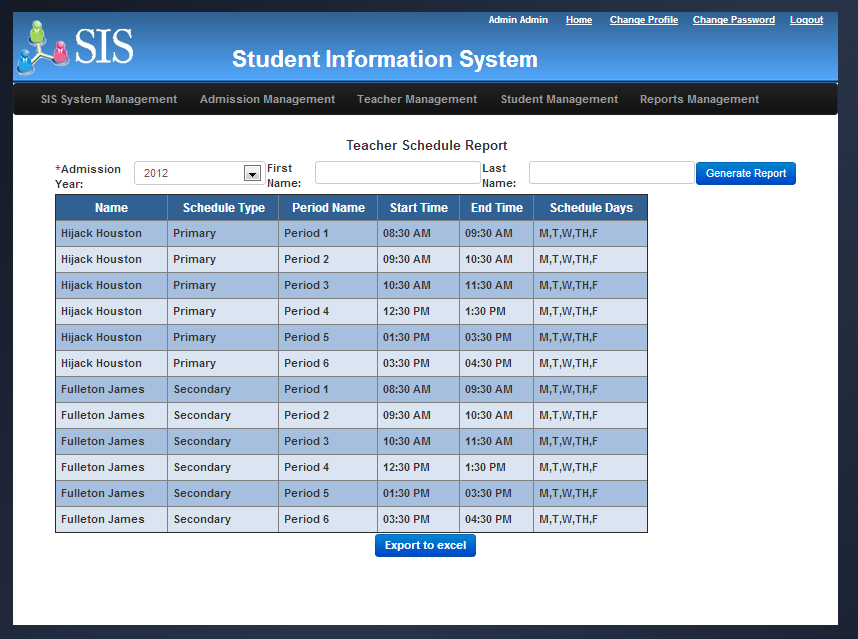
##### Student Grade level report

Once user clicks Student Grade level report link in Reports Management Home page, the Student Grade level report will be displayed with Admission year, Grade level pick lists and Generate report button. Once user clicks Generate Report button, the list of students and their scores and grade letters for each subject will be displayed based on the selected admission year and grade level. Once the report is generated, users can be able to export it to excel using Export to Excel button.



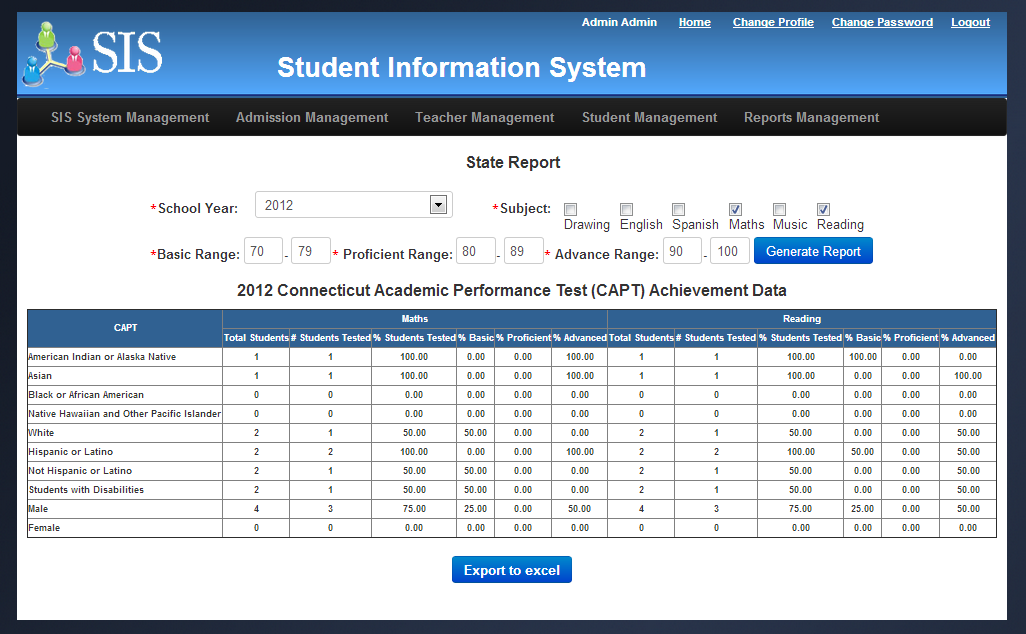
##### Teacher schedule report

Once user clicks Teacher Schedule report link in Reports Management Home page, the Teacher schedule report will be displayed with Admission year, First name, last name and Generate report button. Once user clicks Generate Report button, the list of teachers and their scheduled days and time will be displayed based on the selected admission year, first and last names. If there are no name is provide then system will display all the teacher schedules for the selected year. Once the report is generated, users can be able to export it to excel using Export to Excel button.



##### State report

Once user clicks State report link in Reports Management Home page, the State report will be displayed with Admission year, Subjects, Ranges (basic, advance and proficient) and Generate report button. This report is used by administrator to review the overall school performance based on various subjects and base on various student categories such as Race, Ethnicity, Gender and Students with disabilities. This report measures performs analysis based on different levels of scores. This report will be used to compare the school’s performance with other schools within the state and national level. Once the report is generated, users can be able to export it to excel using Export to Excel button.



**Teacher Module**

### Teacher Module

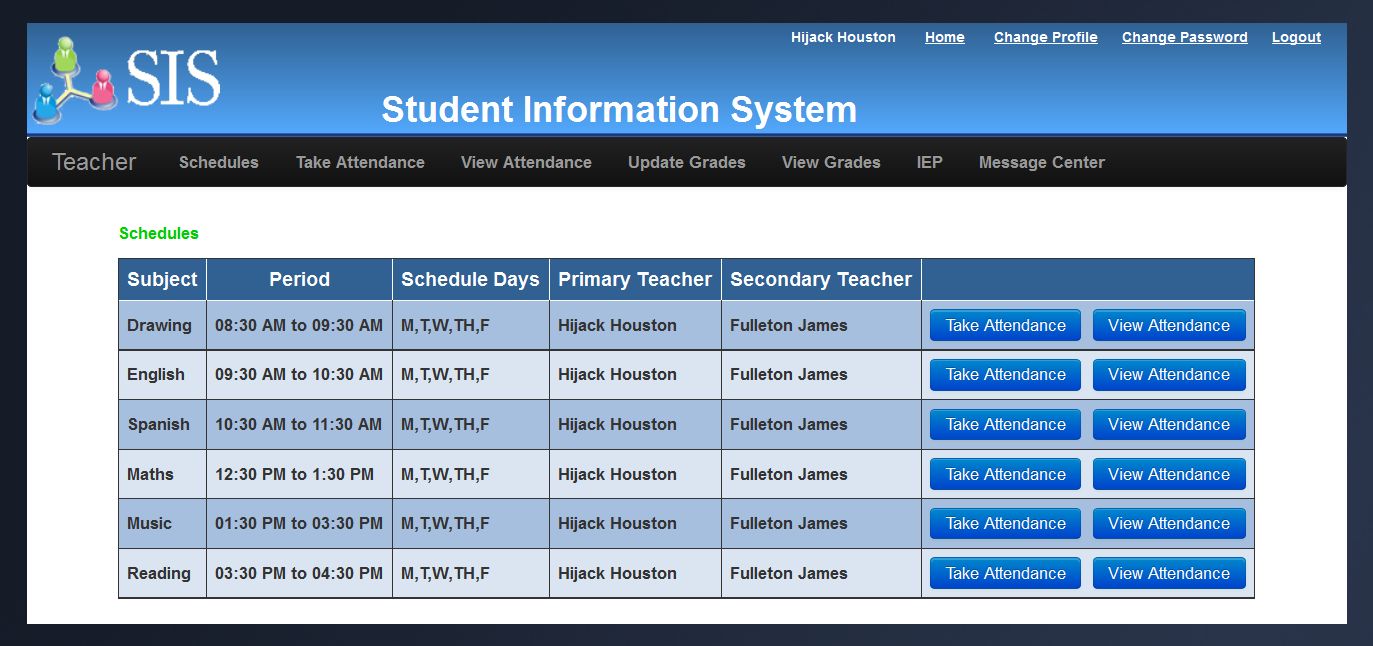
Teachers can see only the students in their class and has access to attendance, grade book, and IEP and message center. Teacher module comes with a powerful internal messaging system, no need for external e-mailing services for communication. All the screens in this module are user friendly and they were created with teachers in mind. Teachers with basic knowledge in computers can start using the system within minutes of their first login. The following sections explain the functionality of the teacher module in detail.

#### Teacher Schedules (Home Page)

Once login, the application displays teacher’s current schedules in the startup page, which is also a default home page for the teacher’s module. The screen has been carefully designed to allow teacher’s to access their most frequently used day to day information and tools from the home page itself.

**Key features:**

* Teacher can quickly peek through all his or her subject assignments.
* Teacher can quickly find the class scheduled days and timings from the same screen. We made this information available on the startup page, because this is one of the day to day information that the teachers required to know their upcoming class schedules.
* Teacher can access the take attendance screen with a single click from this page, which allows teacher to quickly take attendance for the current date and selected period. We made this functionality accessible from the home page, because attendance is one of the day to day activity and we want teachers to have the attendance tool handy as soon as they logs in.
* Teacher can also view a cumulative attendance report of all students for a selected subject. This is a useful report for teachers to determine the class participation of a student while updating the grade book.



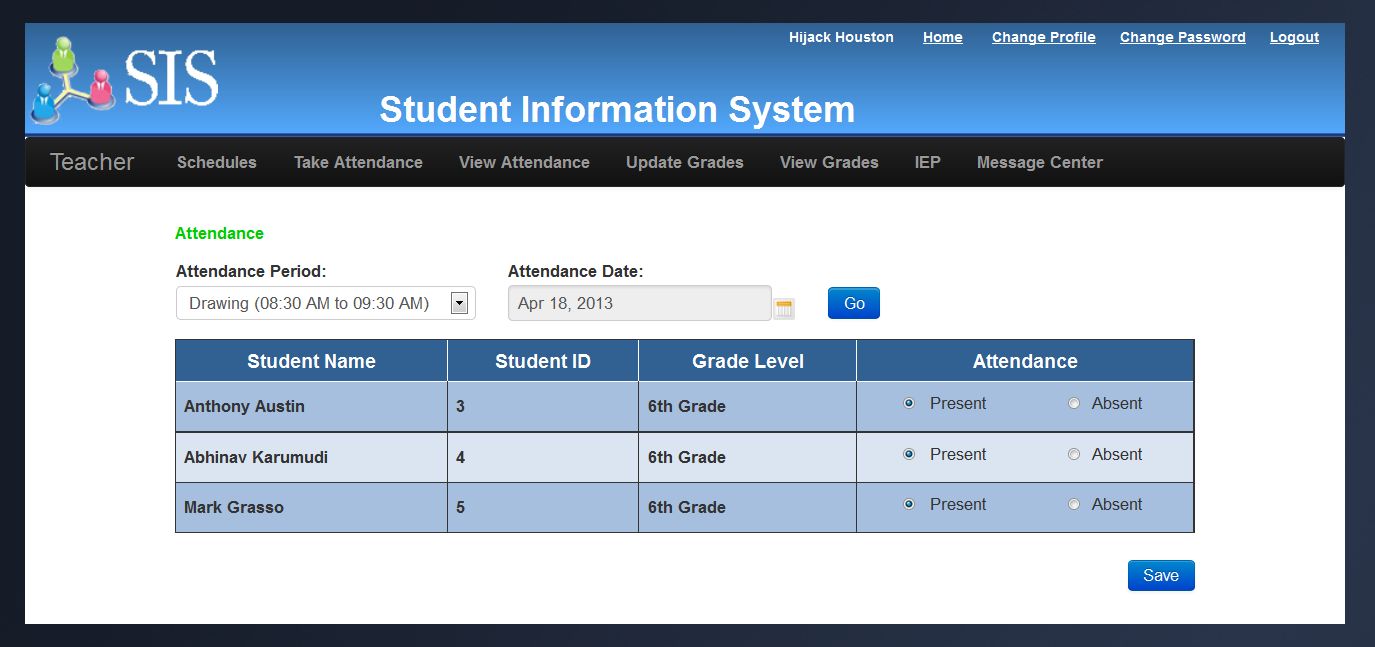
The managed java bean controller behind this screen is gets a list of all teachers’ schedules and binds to a data table in the jsf page. We made the controller session scoped, so that the same teacher schedules can be reused across all the pages during the session.

#### Take Attendance

SIS allows teachers to take attendance by period, by day and all accomplished with only a few clicks. Teachers can access this page either from their home page or from the application menu bar. Again the user interface is very user friendly and teachers can quickly update all students’ attendance by selecting either present or absent radio option. Since this is most frequently used tool by the teachers in their day to day activities, we took care in the design to avoid any complex selection/configuration.

**Key features:**

* Teacher can easily mark attendance of the respected students by the subject.
* Teacher can mark attendance for today or update attendance for any past dates.
* Allows teacher to mark attendance for all enrolled students irrespective of their grades.
* Allows teacher’s to switch to different periods (subjects) from the same screen.
* Allows switching between the periods without leaving the page.



The managed java bean controller behind this screen gets a list of all students enrolled in the selected subject. At the same time, it pulls the corresponding student attendance records for the selected date and subject if exists otherwise creates new attendance entries into the database.

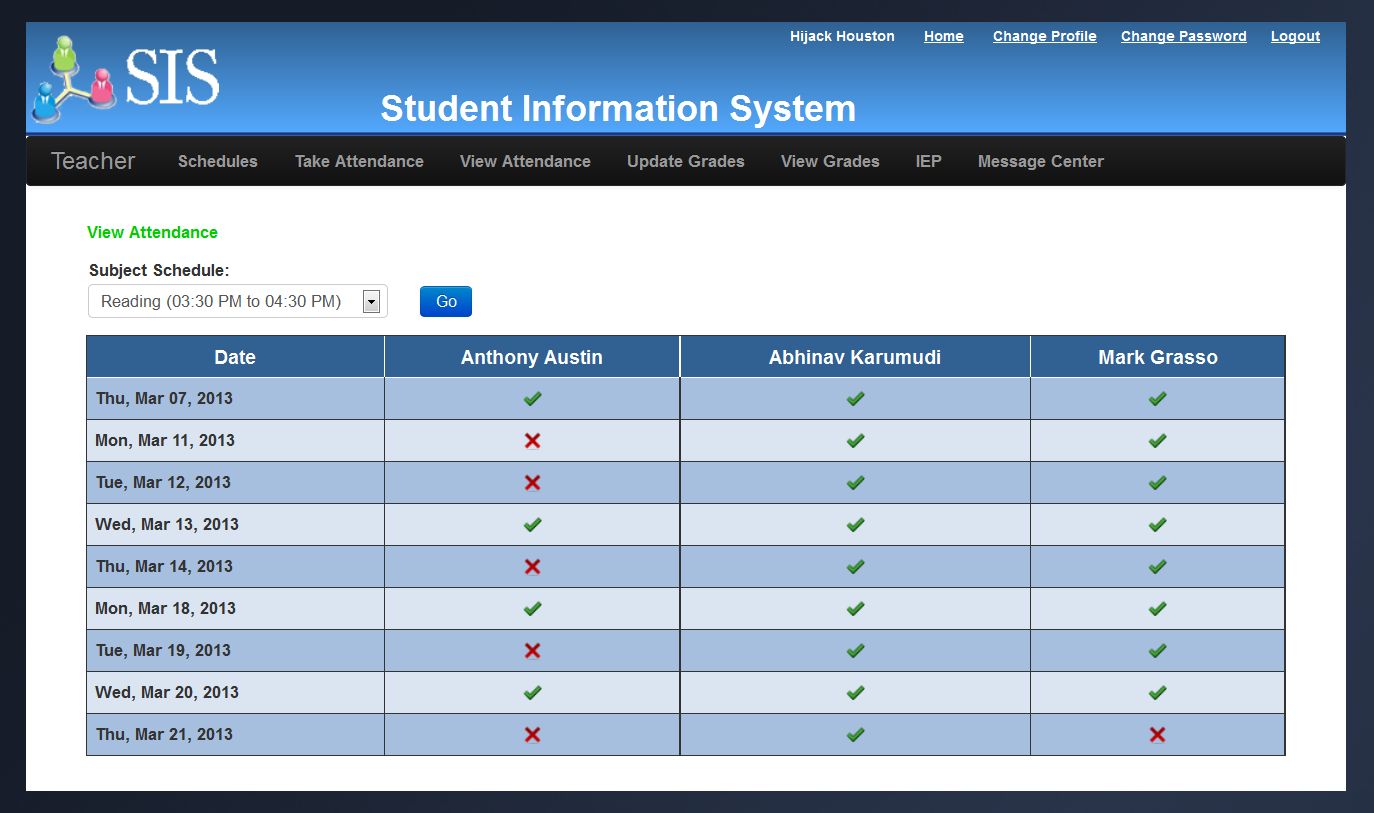
#### View Attendance

The view attendance screen allows teachers to track and view cumulative attendance of all students by subject. This tool allows teachers quickly glance and determine the class participation of a student.

This screen is accessible either from the teacher’s home page or from the application menu bar.

**Key features:**

* Teacher can easily track cumulative attendance of all students subject by subject.
* Displays the attendance in a visually rich appealing UI with red and green marks.
* Teacher can easily switch to different subject to view the attendance of all students in a single screen.
* Allows switching between the subjects without leaving the page.

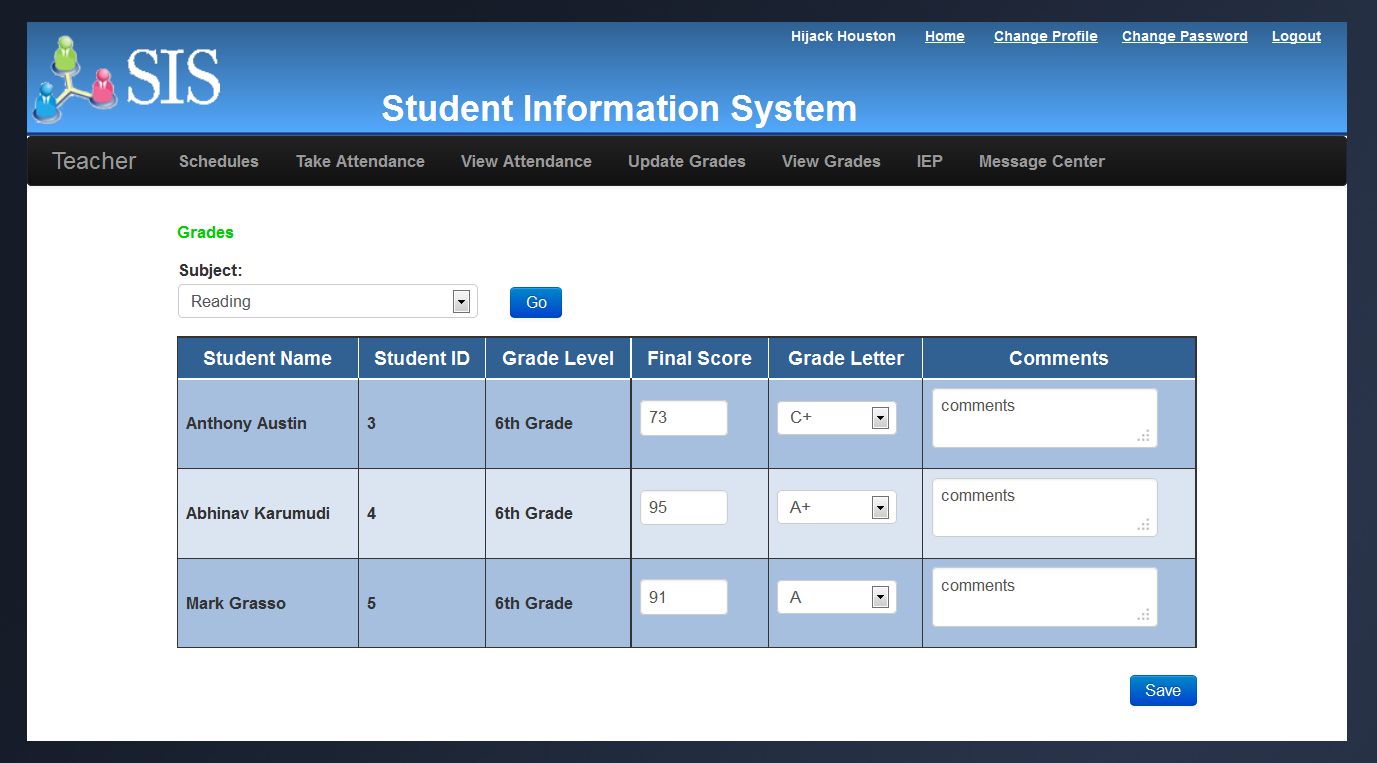
  
The managed java bean controller behind this screen gets a list of all students, their attendance records for a selected subject and combines the information into a single result set. It dynamically generates and populates a data table with varying number of columns based on the number of students enrolled in the selected class.

#### Update Grades

The grade book in SIS allows teachers to enter grades for each student by enrolled subject. We recognize the importance of simple tools for teachers and have created the most simple, user-friendly grade book. With a single click teachers can view and edit the grades of all students from single screen. This screen is accessible from the application menu bar.

**Key features:**

* One screen for both view and editing the grades.
* Teacher can easily updated the grades of all students subject by subject from a single screen.
* Allows updating grades and grade letters from the same screen.
* Teacher can provide additional comments along with grades for each student.
* Allows switching between the subjects without leaving the page.

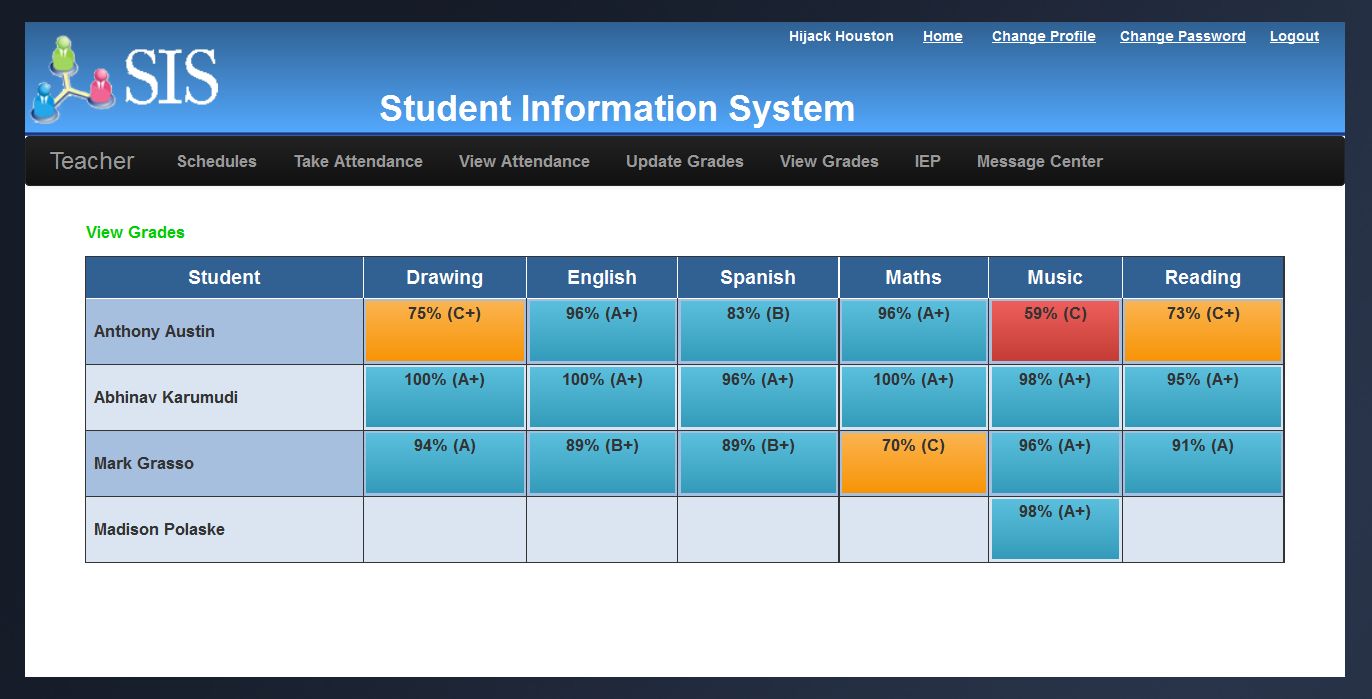
  
The managed java bean controller behind this screen gets a list of all students, their score cards for a selected subject and displays in an editable data table. The controller takes care of adding new score cards automatically if the one doesn’t exist. So it avoids one additional step for teachers to create new score cards first before editing them.

#### View Grades

The main difference between the previous screen and this one is teachers can see collective grade information of all students from all the subjects. This screen is accessible from the application menu bar.

**Key features:**

* One single screen with grade information of all students from each subject.
* This screen allows teachers to quickly look at all grades and the UI is presented with a nice grid layout with a different color codes for each grade range.
* The screen has been designed to displays both the grades and grade letters together.

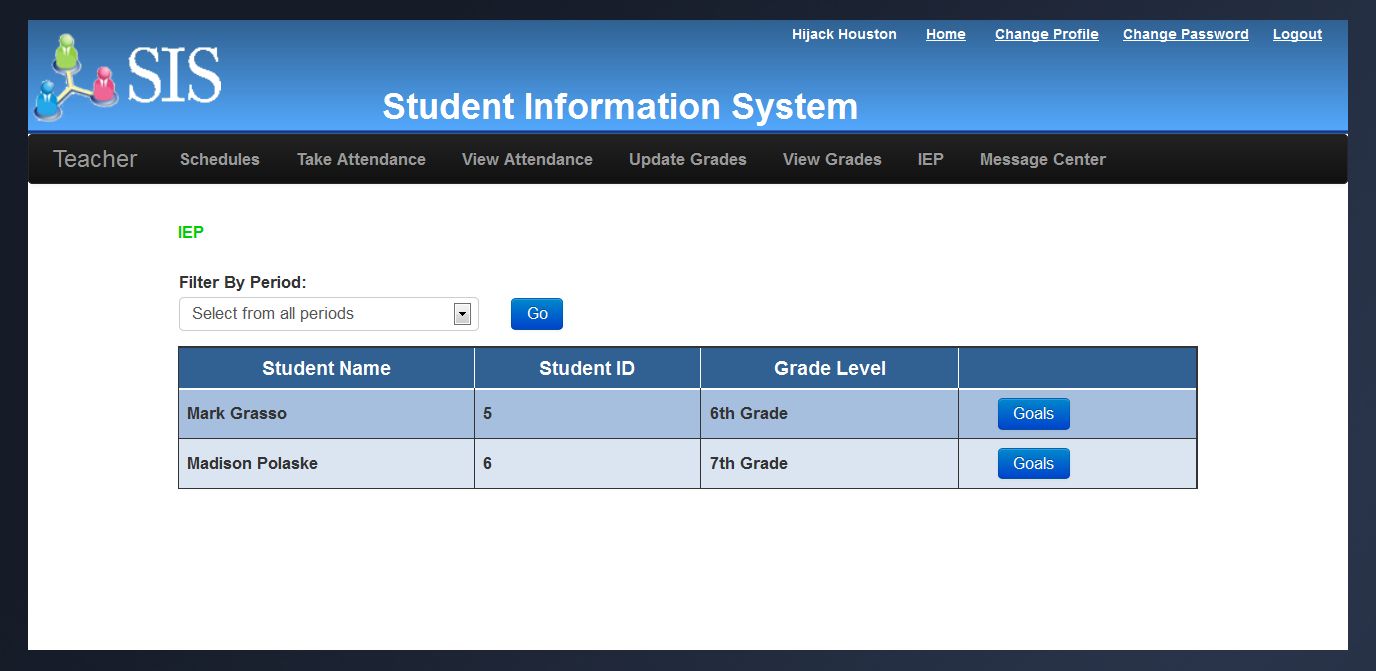
  
The managed java bean controller behind this screen gets a list of all students, their score cards for all subjects and displays in a data table. It dynamically generates and populates a data table with varying number of columns based on the number of subjects that the teacher has been assigned.

#### Individualized Education Plan (IEP)

Individualized Education Plan (IEP) allows schools to store information regarding the students with special education needs. The access to IEP is restricted to strictly authorized users only because of the sensitivity of the data. Essentially, it is used by the special education teachers, school psychologists, IEP coordinators, to enter individualized education plans for students with disabilities. This module would allow users to share files, recordings and videos. For instance, the school psychologist worked with the special education student on some goals or objectives. Using the iPad or any other device, the school psychologist recorded the session. The school psychologist can post a link to the video, so that the file is accessible to all the “team” working with the student. The video can be used by the team to discuss new objectives or observations based on the student’s behavior. IEP tool is accessible from the application menu bar.

**Key features:**

* The IEP main screen displays the list of all students those are eligible for special education.
* Teachers can access all IEP students’ information from different grades on a single screen.
* The screen also provides a filter to shorten the list by selecting IEP students from a selected period/subject.
* From the IEP main screen, teacher can click on the ‘Goals’ to access and manage the IEP goals associated with the student.

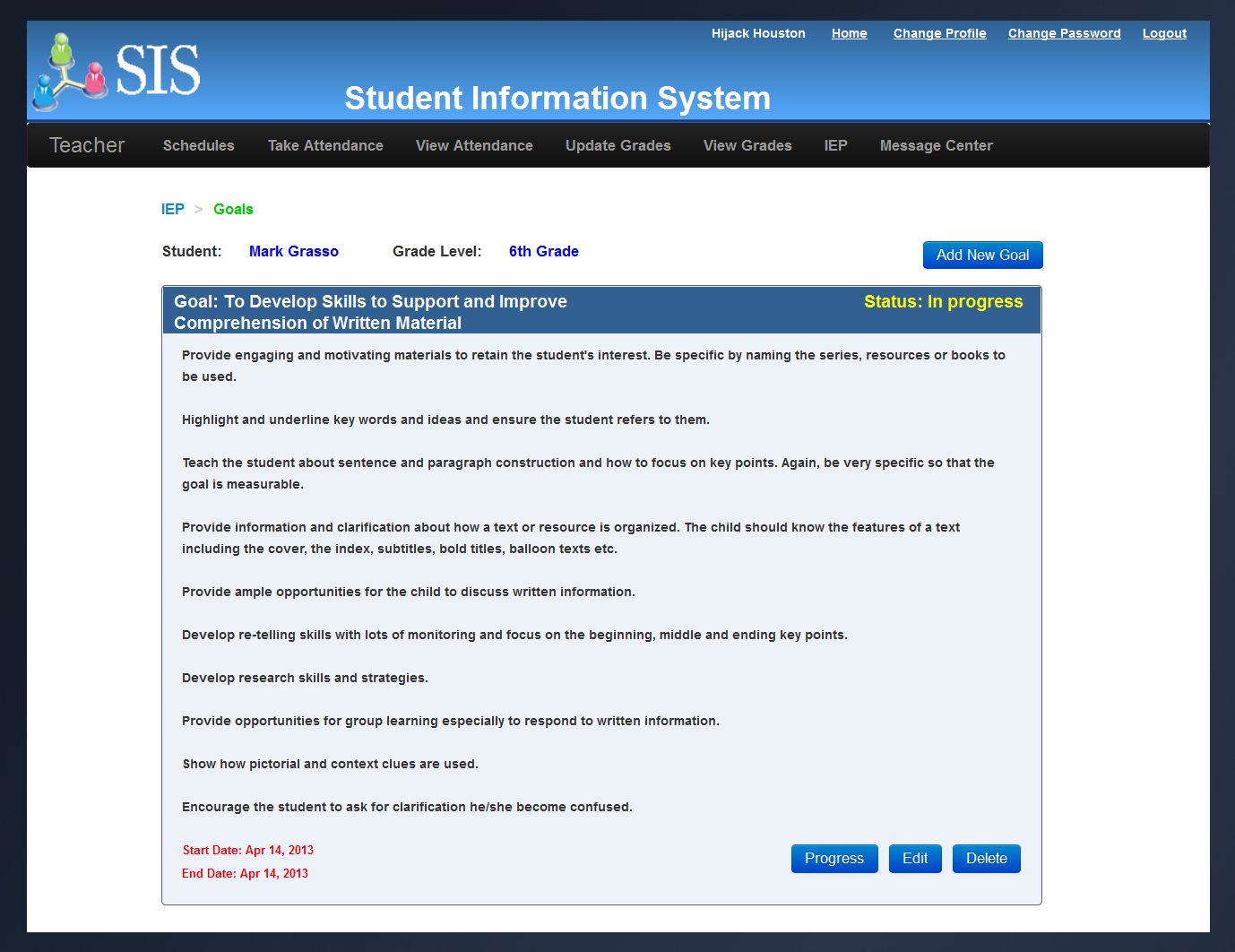
  
The managed java bean controller behind this screen gets a list of all students and their grades filtered by student’s IEP status and the selected period. It displays all IEP student records in a data table and provides a link to access each individual IEP student goals.

#### Individualized Education Plan (IEP) - Goals

Once teacher clicks on the ‘Goals’ , the application displays the list of all IEP goals associated with that student along with links to show goal progress, edit and delete links. This screen displays the IEP goal details, status, start and end dates. The application allows to setup more than one IEP goal for each student. Teacher can add new IEP goal by clicking on the “Add new goal” link on the top of this screen.

**Key features:**

* The screen displays the complete details of the student’s IEP goal and its current status.
* If there is more than one goal associated with the student then it displays another information block as shown in the below picture for each goal. It allows teachers to quickly look at the each goals details and status from single screen.
* Each goal details can be edited separately by clicking on the “Edit” button provided in each information block.
* The “Progress” link allows teachers to track the progress of each individual goal separately.

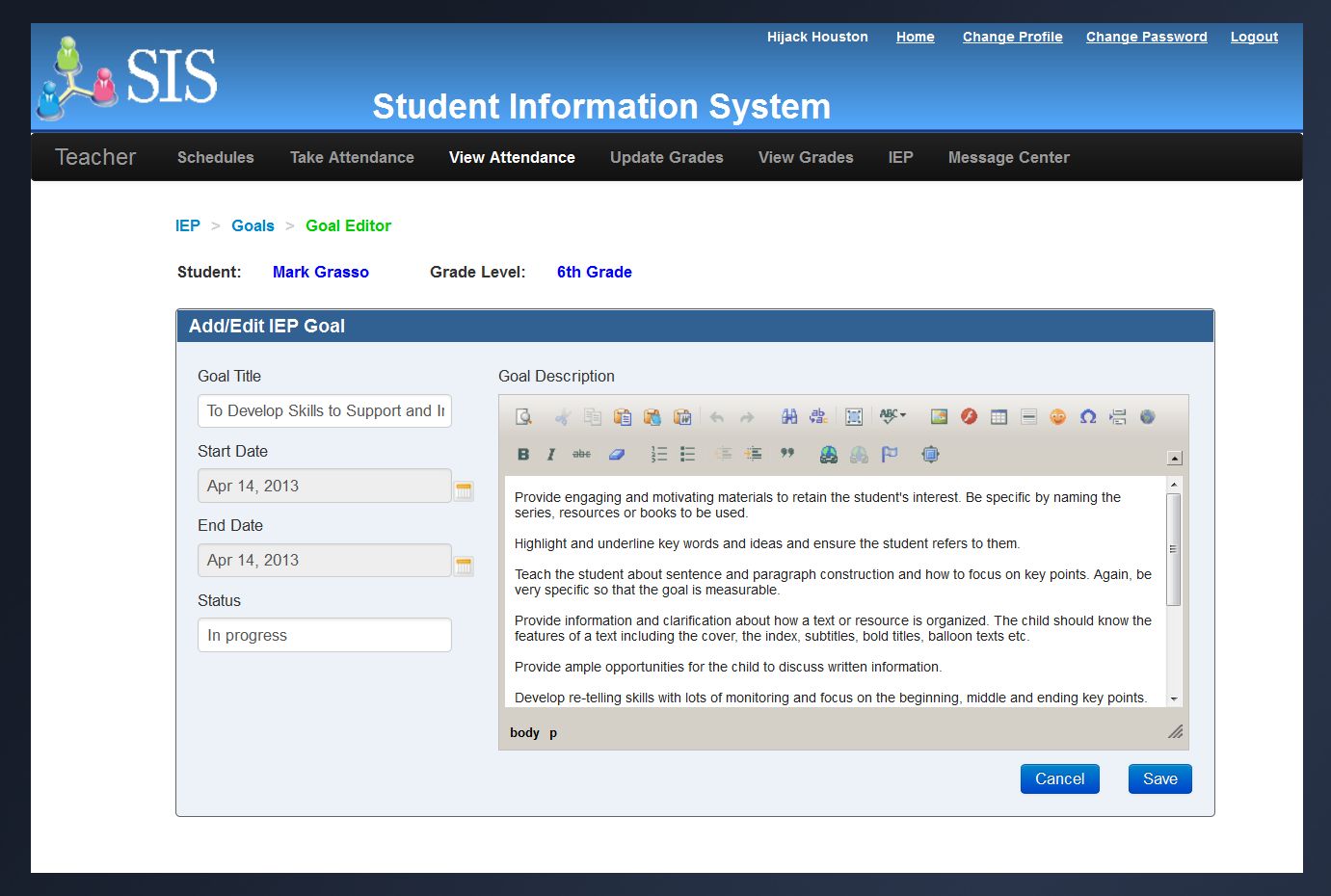
  
The managed java bean controller behind this screen gets a list of all IEP goals associated with student and displays the records using a repeat control in the jsf page. The controller also supports the functionality for saving and deleting the IEP goal records.

#### Add/Edit IEP Goal

Once teacher clicks on the “Add new goal” or “Edit” links from the previous screen, the applications displays a goal editor.

**Key features:**

* The screen has been designed to provide a compact and simple user friendly web form for adding/updating all goal details from a single centralized screen.
* The screen provides a rich text editor to add/edit the goal details. The rich text editor supports a full-fledged text formatting toolbar similar to Microsoft Word.
* The screen also provides a nice calendar control for choosing the start and end dates of the goal.
* Also teacher can update the goal status from the same screen.
* The screen provides a nice breadcrumb navigation to allow teachers to navigate back each step by clicking on the breadcrumb links.

  
The managed java bean controller behind this screen gets the selected IEP goal record and binds to the jsf form. The screen implements multiple components like rich text editor and calendar control from the rich faces framework. The controller supports the functionality to serialize form data to the database.

#### IEP Goal Progress

This screen shows the progress details of a selected IEP goal. Anyone working with this student goal can update the progress. SO the goal can have multiple progress records from either same teacher or from multiple teachers working on this student goal.

**Key features:**

* The screen displays multiple progress records in a separate individual information blocks for easily tracking the IEP goal progress.
* It displays progress details along with other resource links shared by the teachers.
* The screen also displays the updated date and time along with the name of the person that posted the progress details.
* This is centralized screen for managing all IEP goal progress records, at any time teachers can add, edit and delete the IEP progress records from this screen.

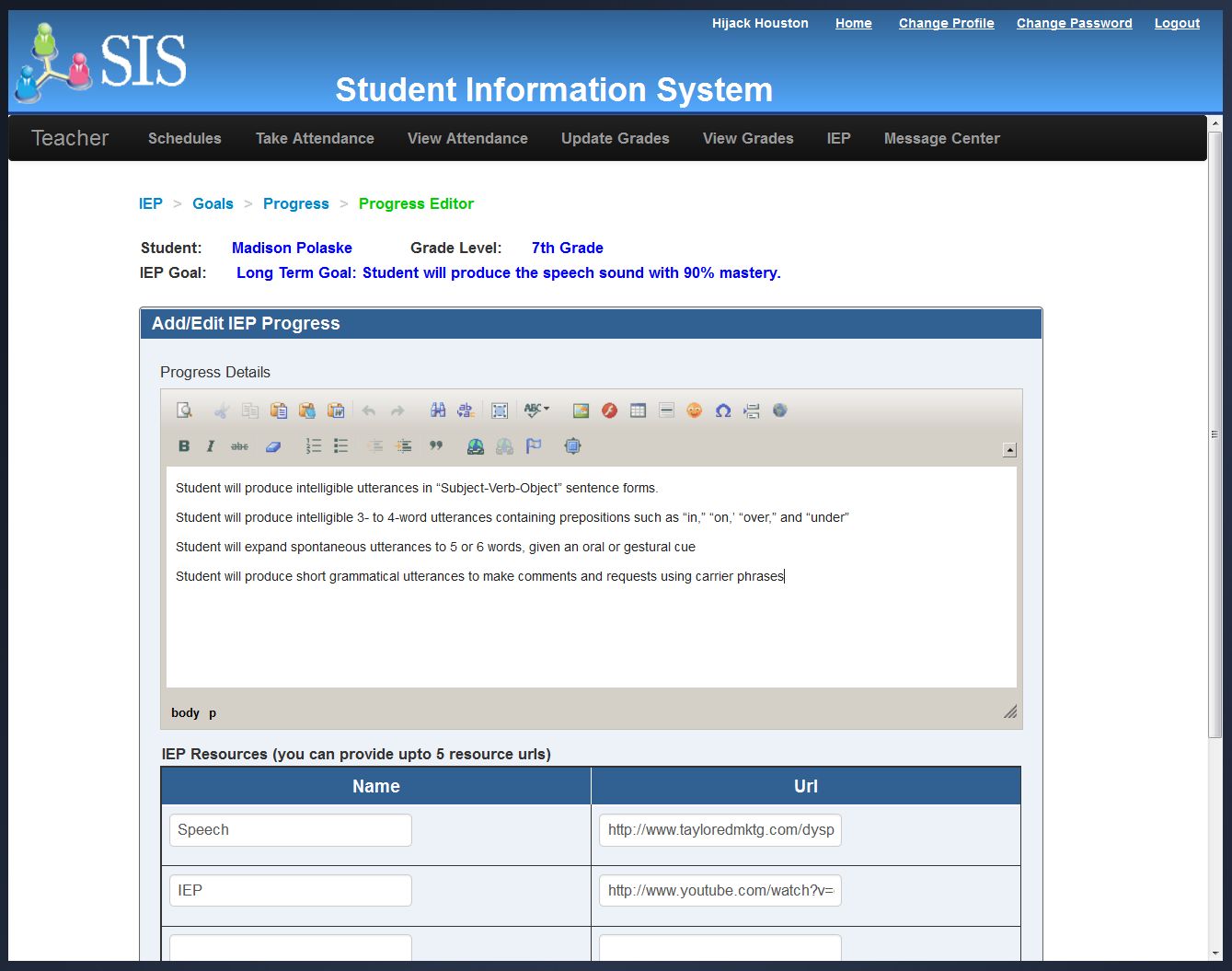
  
The managed java bean controller behind this screen gets a list of all progress records for the selected IEP goal and displays using jsf repeat control as a separate information blocks.

#### Add/Edit IEP Goal Progress

Once teacher clicks on the “Add progress” or “Edit” links from the previous screen, the applications displays an IEP goal progress editor.

**Key features:**

* The screen displays a rich text editor for adding or updating the IEP goal progress details.
* The rich text editor supports a full-fledged text formatting toolbar similar to Microsoft Word.
* Along with progress, teachers can share additional resource links like files, videos with other teachers working with this IEP student.
* The screen provides a nice breadcrumb navigation to allow teachers to navigate back each step by clicking on the breadcrumb links.

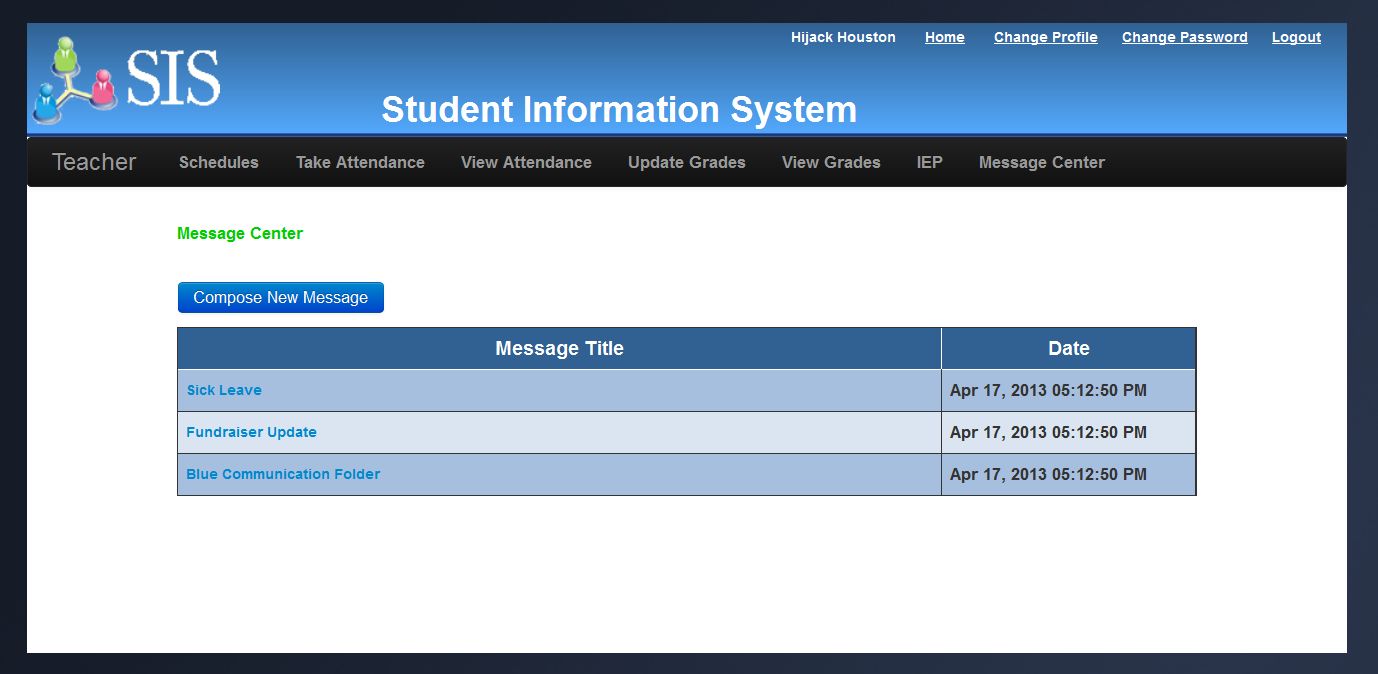
  
The managed java bean controller behind this screen gets the selected IEP goal progress record and binds to the jsf form. The screen implements a rich text editor control from the rich faces framework. The controller supports the functionality to serialize form data to the database.

#### Message Center

SIS allows teachers to collaborate and communicate effectively with students and parents by sending messages or real time feedback on student’s performance through the message center. For the purpose of collaborative environment we created an inbuilt messaging system. It’s a powerful internal messaging system and it doesn’t require an external e-mailing service for communication. This is also one of the day to day activities of the teachers, so we made this UI user friendly as well and looks familiar interface like an inbox from most of the email clients. Message center tool is accessible from the application menu bar.

**Key features:**

* It’s an easy and quick way to send messages to any user in the system.
* It doesn’t require an external e-mailing service for communication.
* The main screen of the message center looks like an inbox and shows all incoming messages received by the teacher.
* All incoming messages are organized by received date.
* Teacher can read the message details by clicking on the message title.
* Teacher can compose a new message by clicking on the “Compose New Message” link on the top.

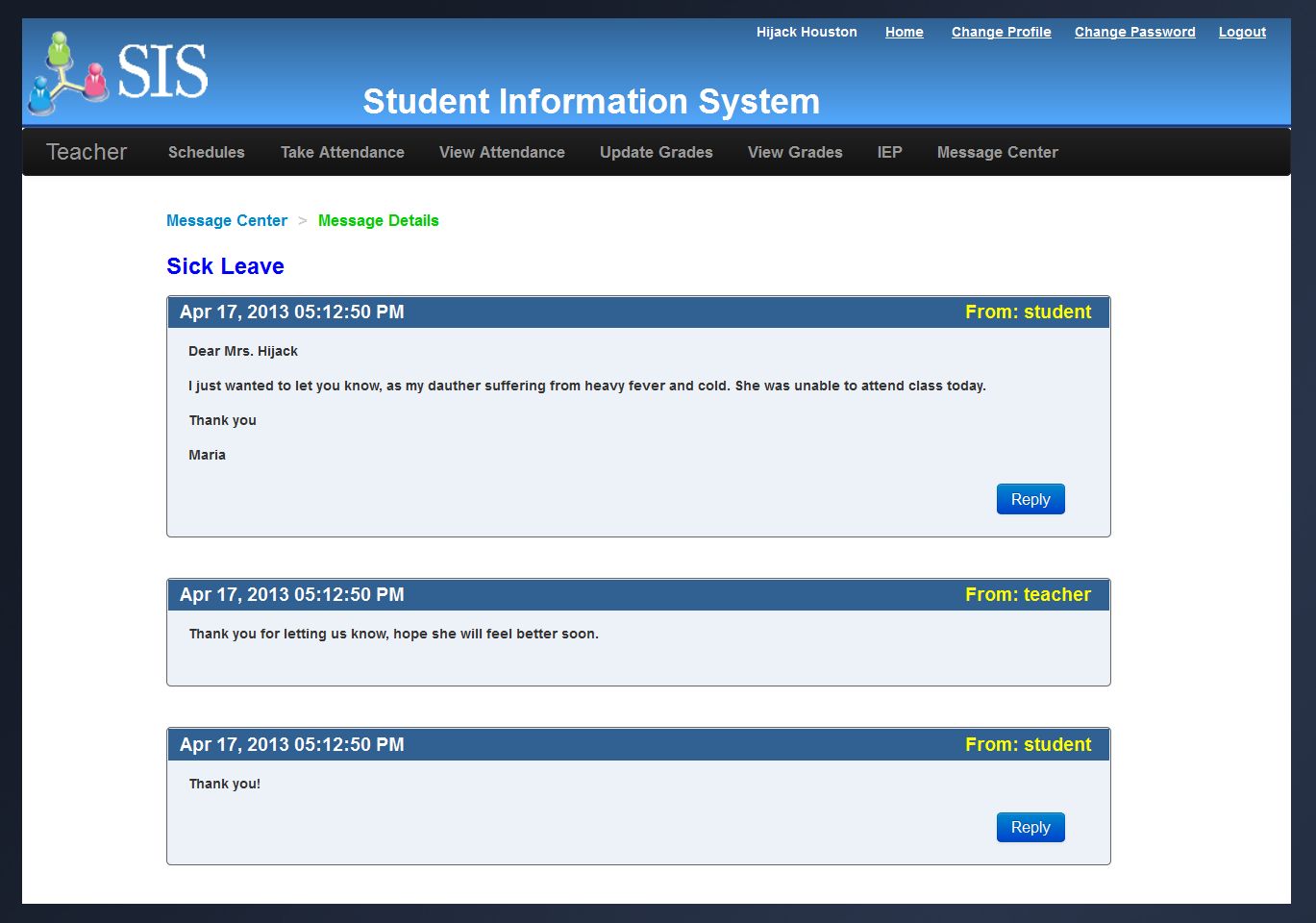
  
The managed java bean controller behind this screen gets a list of all message received by the teacher and displays in a data table like an inbox style. All records are sorted by the messages received date.

#### Message Details

Once teacher clicks on the message title, it redirects to the message details page.

**Key features:**

* This screen displays all chained related responses for the selected message in received date order.
* Each message is displayed on its own message block and the message block shows the received date and sender name.
* A replay button appears at the bottom of each message block if the message is received from other users.
* The screen provides a nice breadcrumb navigation to allow teachers to navigate back each step by clicking on the breadcrumb links.

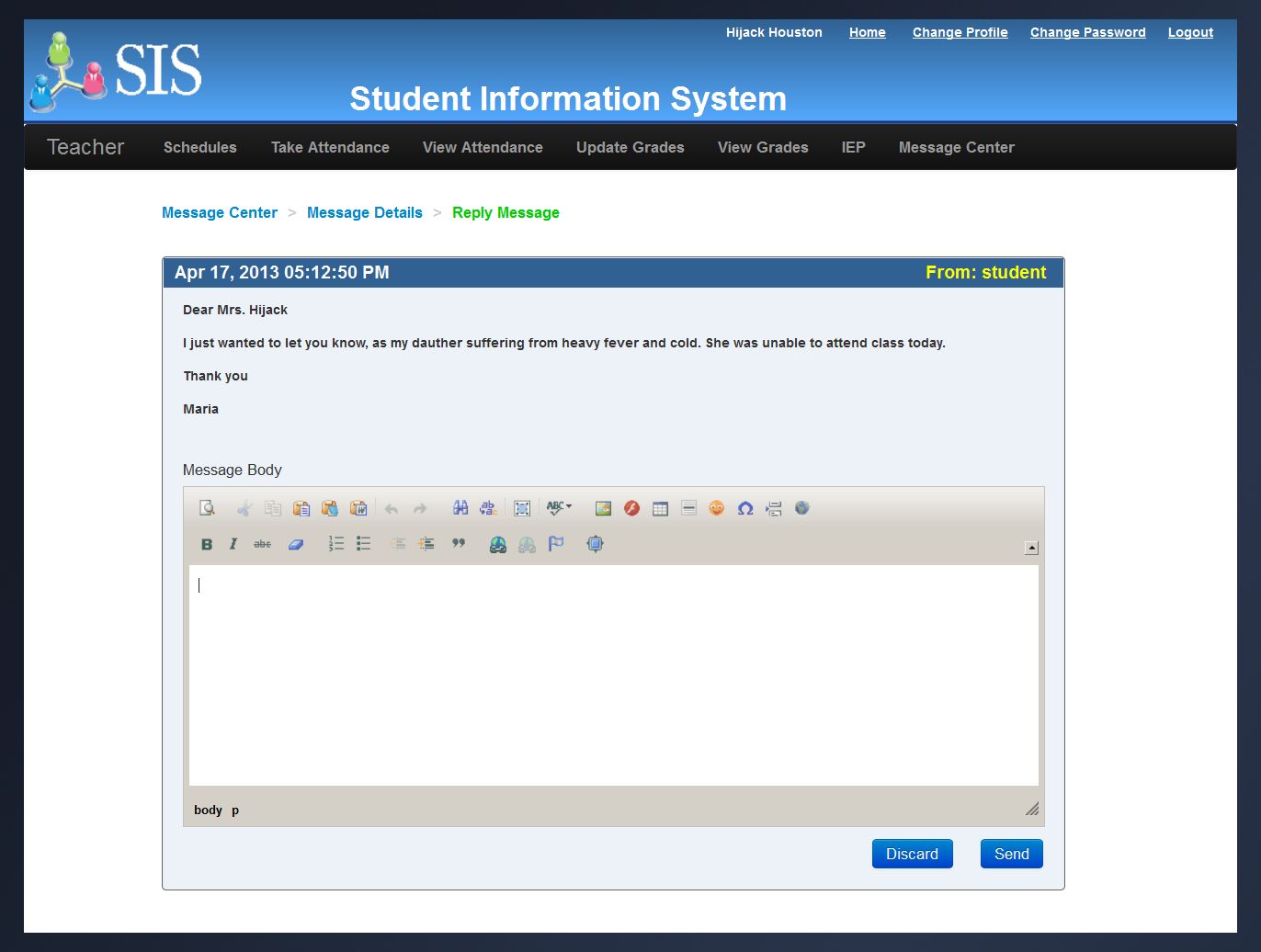
  
The managed java bean controller behind this screen gets a list of all chained messages associated to the selected message. The reply button redirects to the compose screen.

#### Reply Message

Once teacher clicks on the replay button on the previous screen, it redirects to the compose message screen.

**Key features:**

* This screen displays a selected replay from the other user as well as an editor for composing a new replay message.
* Teachers can use the rich text editor to format the text and also can embed the hyperlinks.
* The advantage of this screen is teacher doesn’t need to know the sender id. The message will be automatically sends to the sender of the selected message.
* The screen provides a nice breadcrumb navigation to allow teachers to navigate back each step by clicking on the breadcrumb links.

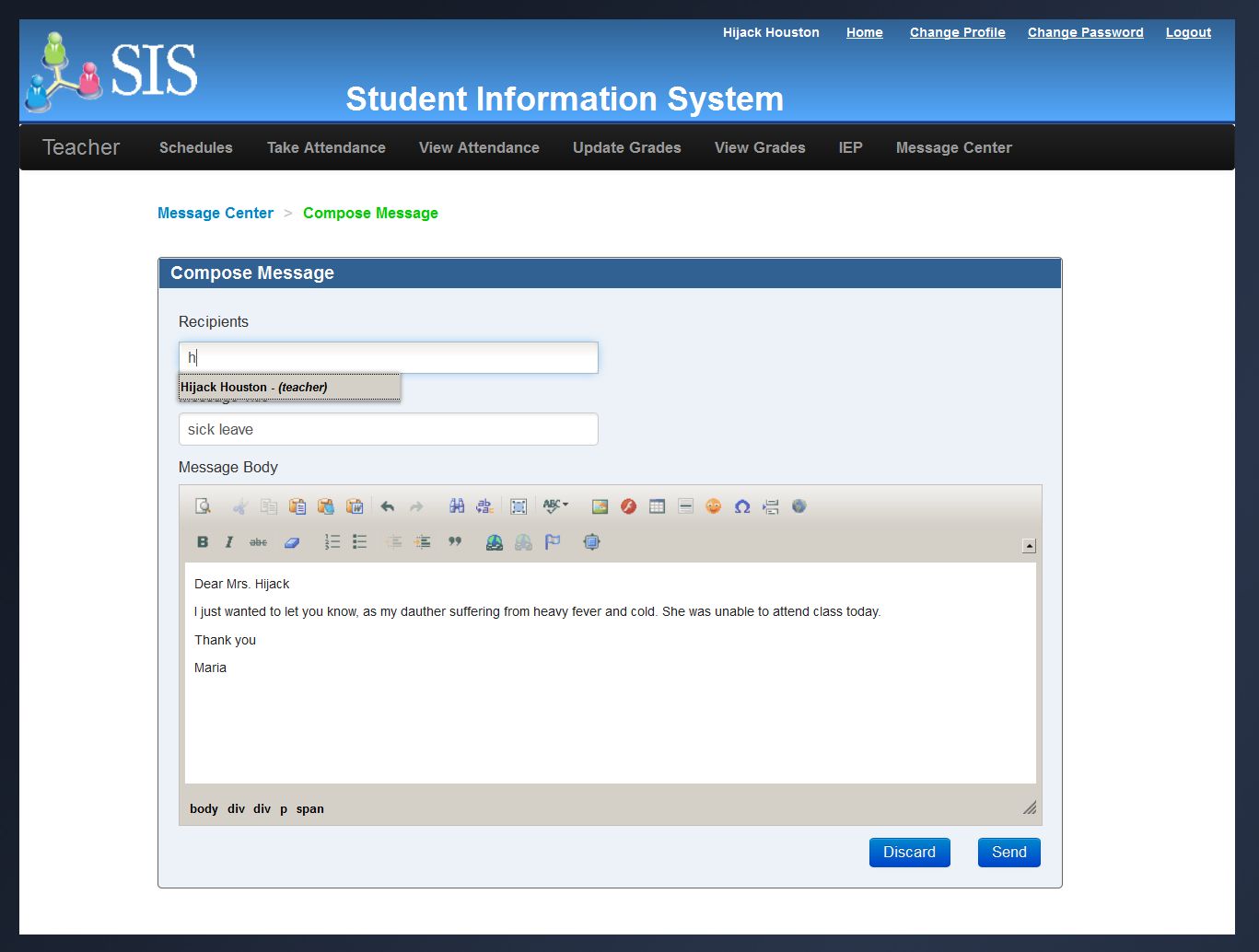
  
The managed java bean controller behind this screen gets the details of selected reply message and automatically sends the composed message to the sender.

#### Compose New Message

Teacher can click on the compose button to create a new message and send it to either single or multiple users.

**Key features:**

* This screen allows teacher to compose a new message in the rich text editor.
* The recipient field allows teacher to specify multiple recipients separated by either coma (,) or semi colon (;).
* The recipient field is an auto complete text field, so user can type in single letter and the application will find all matching users starts with that letter and displays as auto completion list. This feature is very useful, because teacher doesn’t need to remember the userid’s of the recipients.
* Once click on the send button, the message will be sent to all recipients specified in the recipients field.

  
The managed java bean controller behind this screen sends the message to all recipients specified in the recipient’s field. The controller also supports an AJAX request for serving matching users list to display in the auto completion control.

**Student Module**

### Student /Parent Module

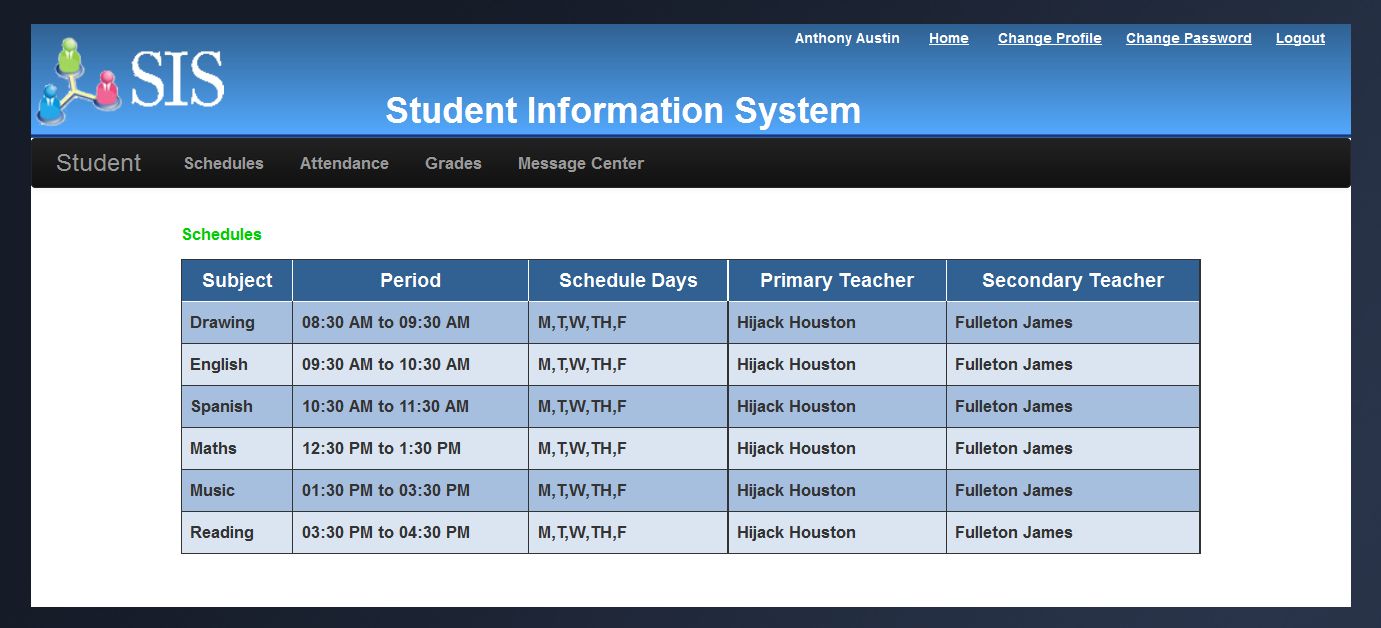
Using this module, students/parents would be able to view the schedules, grades, attendance status, and feedback/comments from the teachers and can effectively collaborate and communicate with teachers by sending messages or inquiries.

#### Student- Schedules (Home Page)

Once login, the application displays student’s current schedules in the startup page, which is also a default home page for the student’s module. The screen has been carefully designed to show the complete class schedule for the student with scheduled days and timings.

**Key features:**

* Student can quickly peek through all his or her enrolled subjects.
* Student can quickly find the class scheduled days and timings from the same screen. We made this information available on the startup page, because this is one of the day to day information that the students required to know their upcoming class schedules.



The managed java bean controller behind this screen is gets a list of all students’ schedules and binds to a data table in the jsf page. We made the controller session scoped, so that the same student schedules can be reused across all the pages during the session.

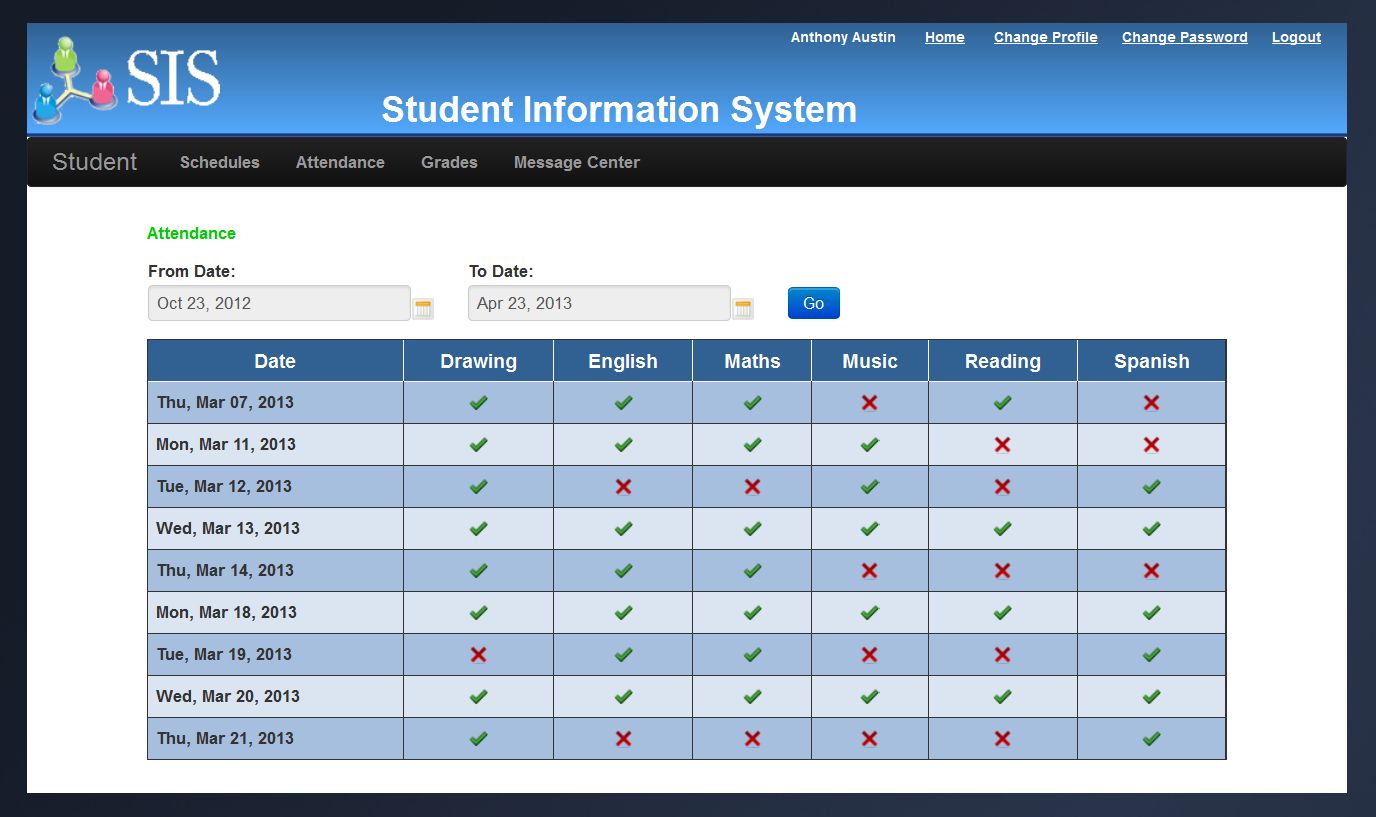
#### Student - View Attendance

The view attendance screen allows students to track and view cumulative attendance of all subjects by a selected date range. This tool allows students quickly glance and determine their class participation of each subject.

This screen is accessible from the application menu bar.

**Key features:**

* Student can easily track cumulative attendance of all subjects by a date range.
* Displays the attendance in a visually rich appealing UI with red and green marks.
* Students can easily switch to different data range to get their attendance report for week or a month, etc.

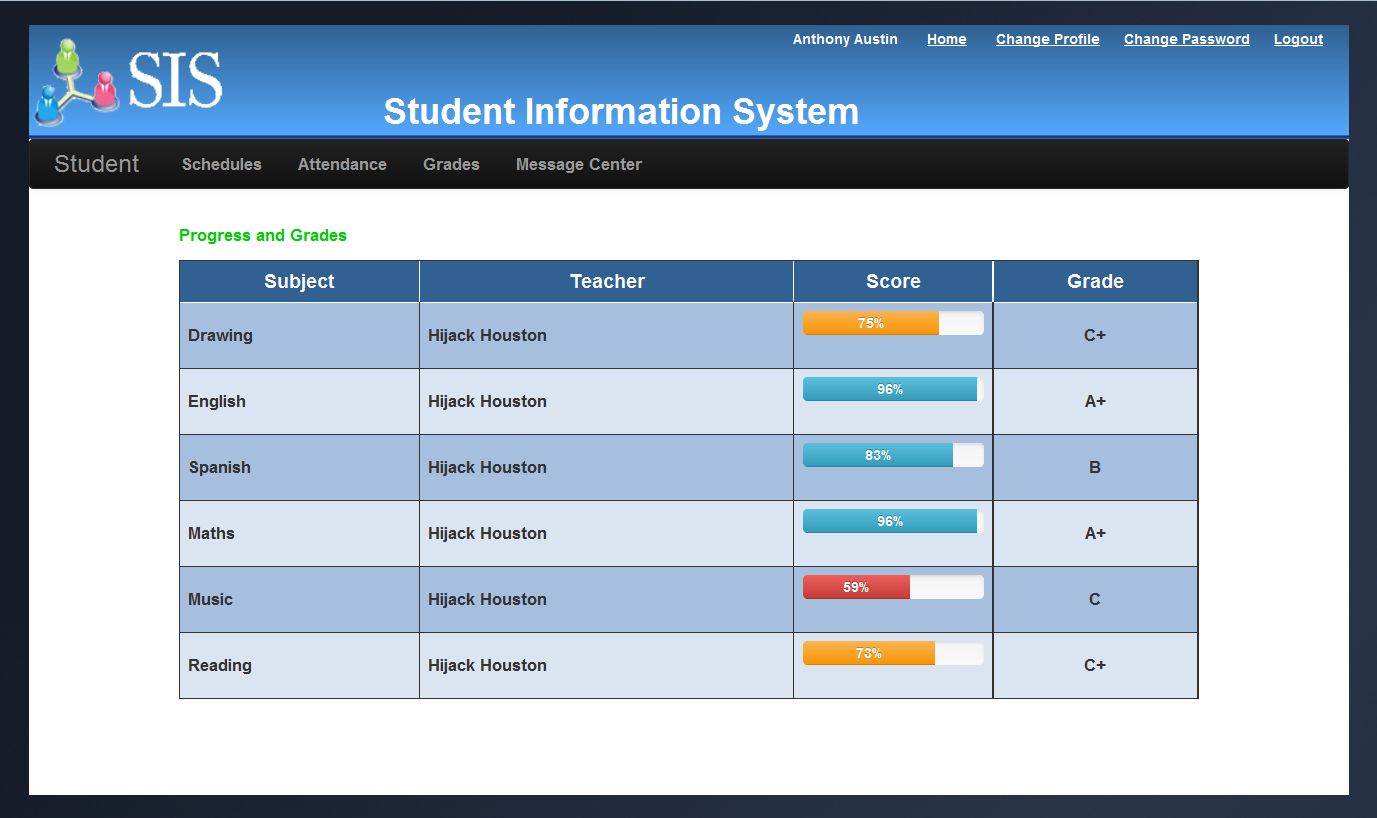
  
The managed java bean controller behind this screen gets a list of all students subject; attendance records for each subject for a given data range and combine the information into a single result set. It dynamically generates and populates a data table with varying number of columns based on the number of subjects enrolled by the student.

#### Student - View Grades

Student can see collective grade information of all the subjects from a single screen. This screen is accessible from the application menu bar.

**Key features:**

* One single screen with grade information of all subjects.
* This screen allows students to quickly look at all grades and the UI is presented with a nice grid layout with a different color codes for each grade range.
* The screen has been designed to displays both the grades and grade letters together.

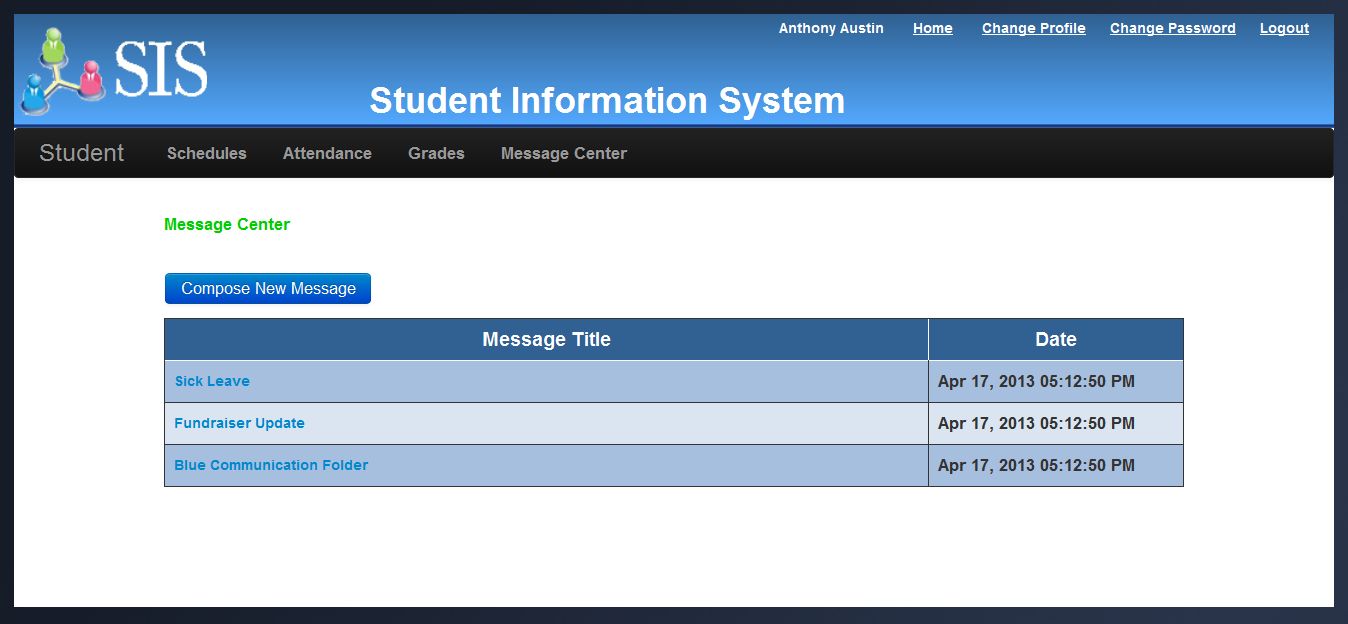
  
The managed java bean controller behind this screen gets a list of all subjects, their score cards for all subjects and displays them in a data table.

#### Student - Message Center

SIS allows students and parents to collaborate and communicate effectively with teachers by sending messages the message center. For the purpose of collaborative environment we created an inbuilt messaging system. It’s a powerful internal messaging system and it doesn’t require an external e-mailing service for communication. This is also one of the day to day activities of the students, so we made this UI user friendly as well and looks familiar interface like an inbox from most of the email clients. Message center tool is accessible from the application menu bar.

**Key features:**

* It’s an easy and quick way to send messages to any user in the system.
* It doesn’t require an external e-mailing service for communication.
* The main screen of the message center looks like an inbox and shows all incoming messages received by the student.
* All incoming messages are organized by received date.
* Student can read the message details by clicking on the message title.
* Student can compose a new message by clicking on the “Compose New Message” link on the top.

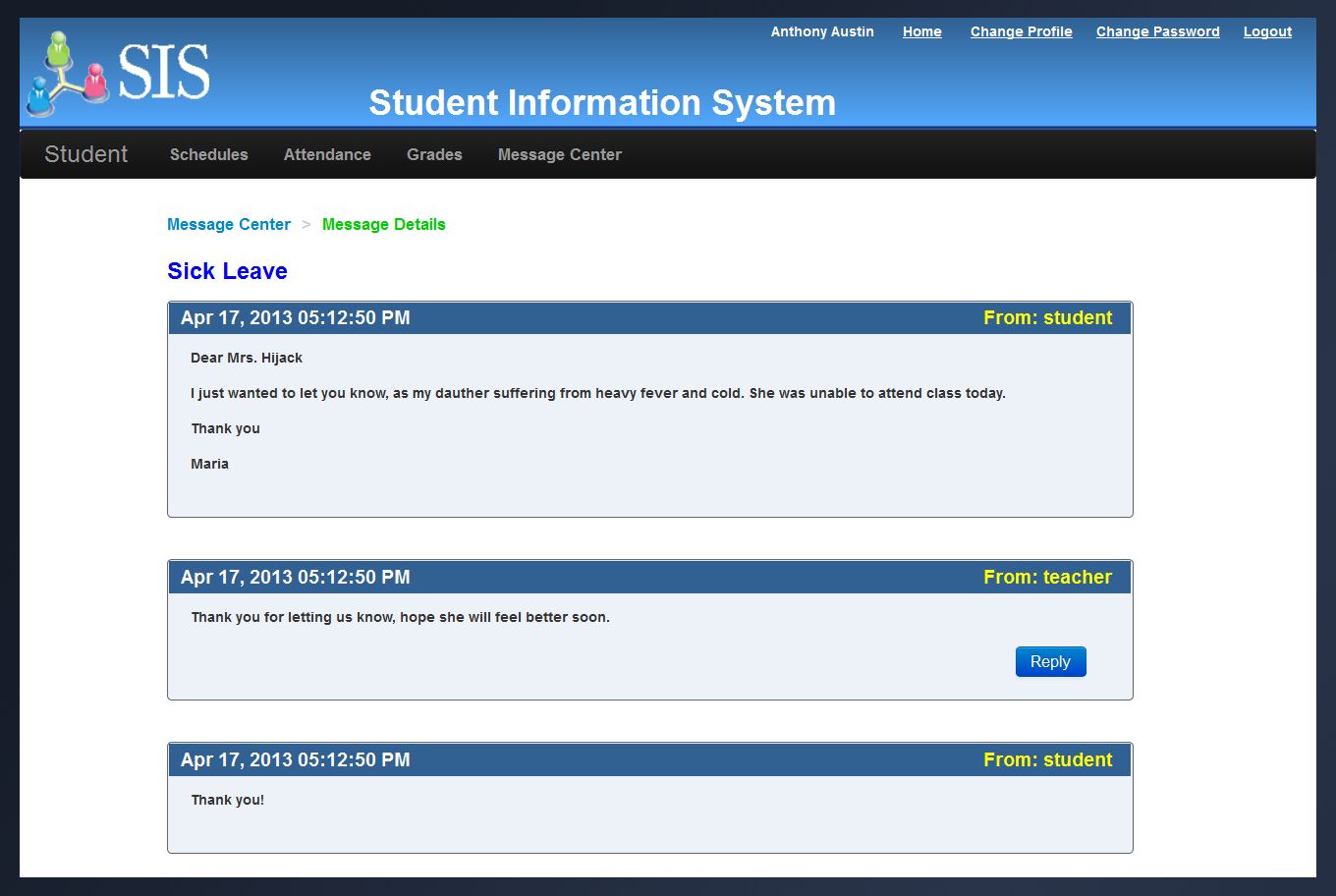
  
The managed java bean controller behind this screen gets a list of all message received by the student and displays in a data table like an inbox style. All records are sorted by the messages received date.

#### Student - Message Details

Once student clicks on the message title, it redirects to the message details page.

**Key features:**

* This screen displays all chained related responses for the selected message in received date order.
* Each message is displayed on its own message block and the message block shows the received date and sender name.
* A replay button appears at the bottom of each message block if the message is received from other users.
* The screen provides a nice breadcrumb navigation to allow teachers to navigate back each step by clicking on the breadcrumb links.

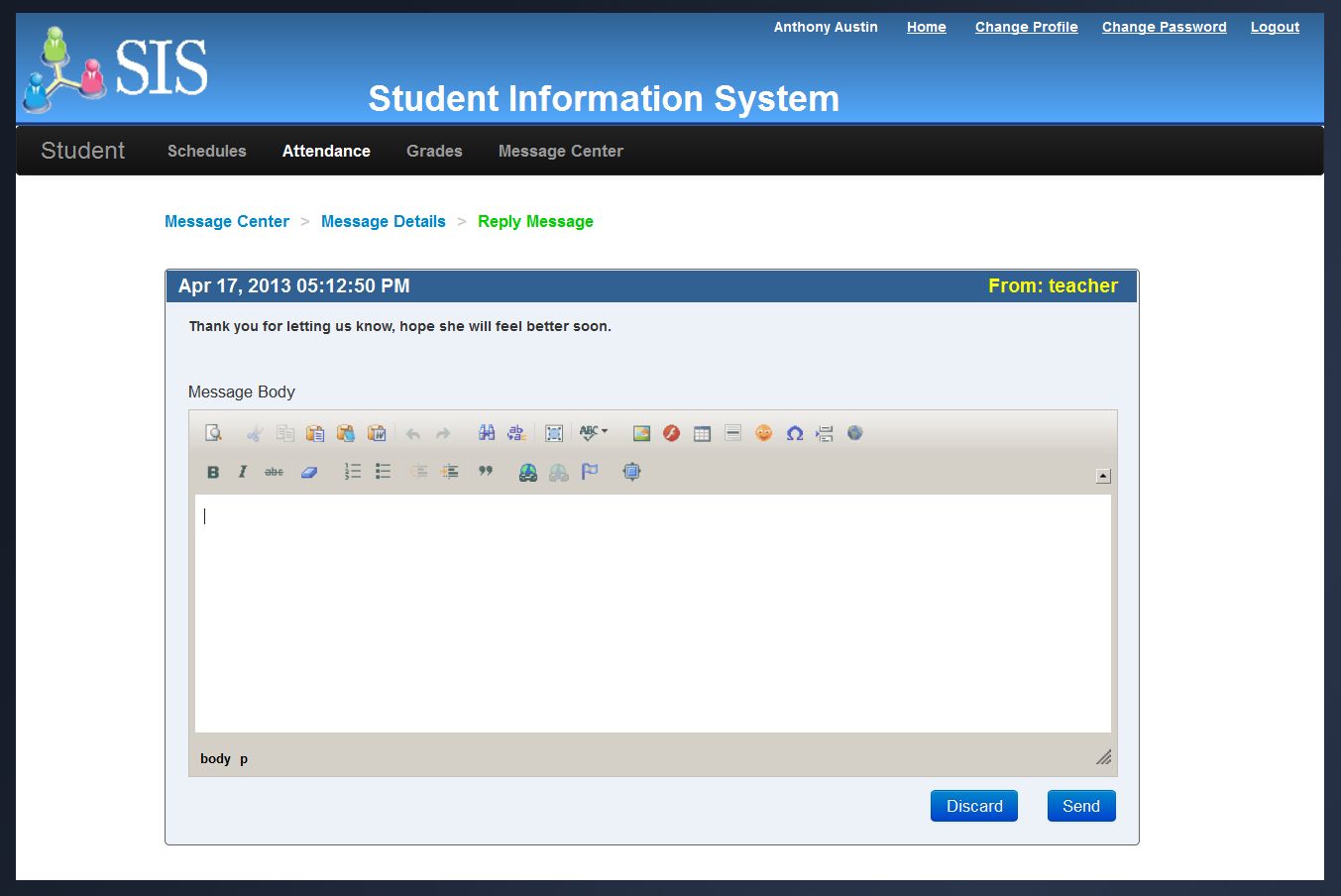
  
The managed java bean controller behind this screen gets a list of all chained messages associated to the selected message. The reply button redirects to the compose screen.

#### Student - Reply Message

Once student clicks on the replay button on the previous screen, it redirects to the compose message screen.

**Key features:**

* This screen displays a selected replay from the other user as well as an editor for composing a new replay message.
* Students can use the rich text editor to format the text and also can embed the hyperlinks.
* The advantage of this screen is student doesn’t need to know the sender id. The message will be automatically sends to the sender of the selected message.
* The screen provides a nice breadcrumb navigation to allow teachers to navigate back each step by clicking on the breadcrumb links.

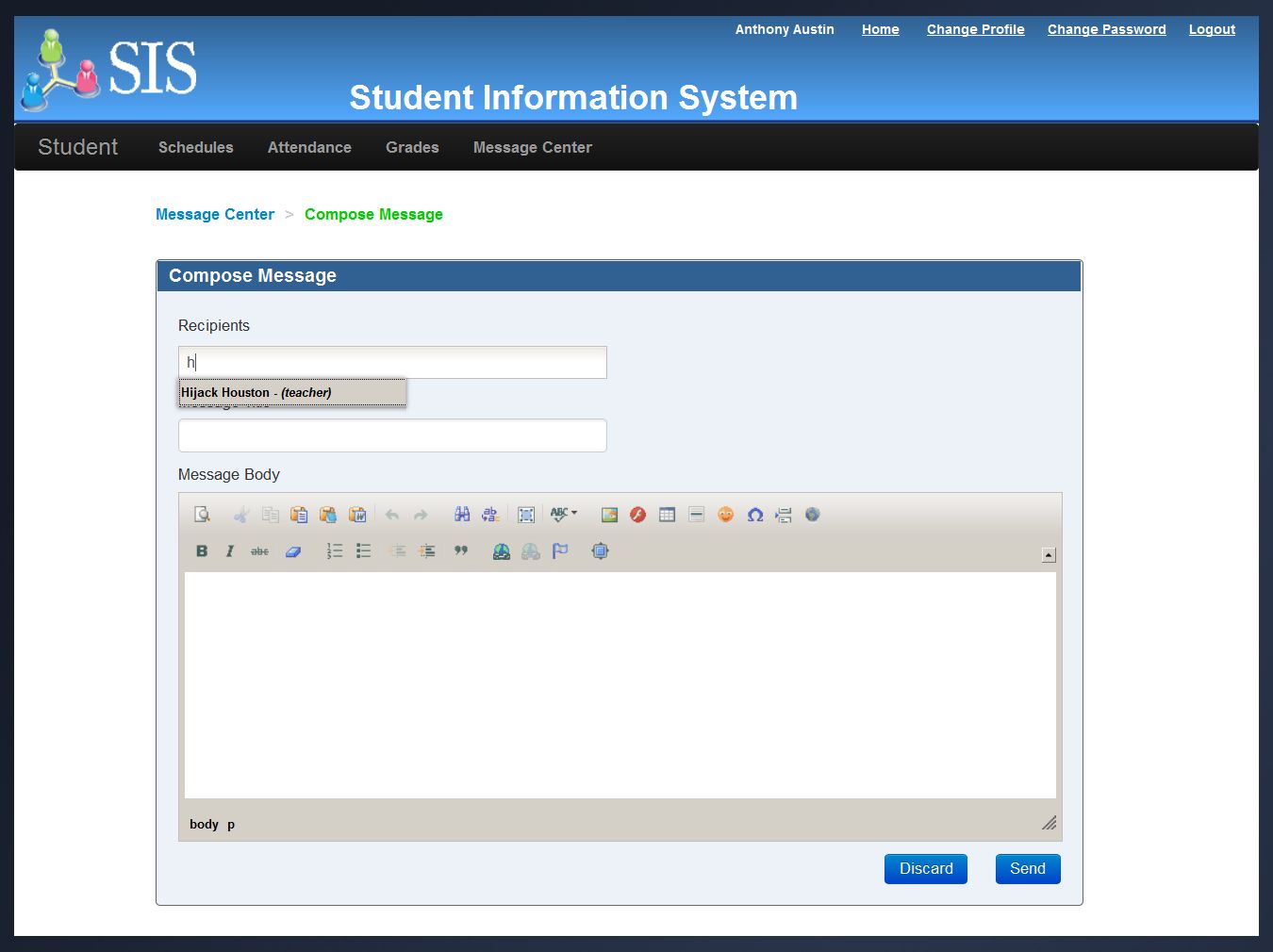
  
The managed java bean controller behind this screen gets the details of selected reply message and automatically sends the composed message to the sender.

#### Student - Compose New Message

Student can click on the compose button to create a new message and send it to either single or multiple users.

**Key features:**

* This screen allows student to compose a new message in the rich text editor.
* The recipient field allows student to specify multiple recipients separated by either coma (,) or semi colon (;).
* The recipient field is an auto complete text field, so user can type in single letter and the application will find all matching users starts with that letter and displays as auto completion list. This feature is very useful, because student doesn’t need to remember the userid’s of the recipients.
* Once click on the send button, the message will be sent to all recipients specified in the recipients field.

  
The managed java bean controller behind this screen sends the message to all recipients specified in the recipient’s field. The controller also supports an AJAX request for serving matching users list to display in the auto completion control.