

Pace University
New York, New York

CS 389
May 5, 2004

Viktor Geller Naseer Haniff Nikita Lukish

Requirements Document v3.0

Pace Schedule Builder ("PSB")

Table of Contents

1. Business Requirements	3
1.1.1 Background, Opportunity, and Customer Needs.....	3
1.1.2 Objectives and Success Criteria.....	3
1.1.3 Business Risks	3
1.2. Vision of the Solution	3
1.2.1 Vision Statement	3
1.2.2 Major Features	4
1.2.3 Assumptions and Dependencies	4
1.3. Scope and Limitations.....	4
1.3.1 Scope of Initial Releases	4
1.3.2 Limitations and Exclusions	4
1.4. Business Context	4
1.4.1 Stakeholder profiles	4
1.4.2 Project Priorities	5
1.5. Business Rules	5
2. Use Cases.....	7
UC-1	8
UC-2	9
UC-3	10
UC-4	11
UC-5	12
UC-6	13
UC-7	14
UC-8	15
UC-9	16
3. Software Requirements Specification	17
3.1. Introduction	17
3.1.1 Purpose	17
3.1.2 Project Scope and Project Features	17
3.2. Overall Description.	18
3.2.1 Product Perspective	18
3.2.2 User Classes and Characteristics	18
3.2.3 Operating environment.....	19
3.2.4 Design and Implementation Constraints	19
3.2.5 User Documentation	19
3.3. Other Nonfunctional Requirements	19
3.3.1 Performance Requirements	19
3.3.2 Security Requirements	19
3.3.2 Software Quality Attributes.....	19
Appendix A – Data Dictionary	20
Appendix B – User Interface Prototype	22
Appendix C – Sample Pace University Class Schedule	23

1. Business Requirements

1.1.1 Background, Opportunity, and Customer Needs

Every semester, thousands of Pace students go through the class selection process for registration. It takes an average of 2 hours to create a schedule without any conflicts, 30 minutes to draw in on paper or type it in Excel, and another 30 minutes to evaluate the schedule in terms of the break size between classes, total credits, and total time left for such activities as work. A significant amount of time is wasted on these manual computation tasks. Unfortunately, the whole process must often be started over from scratch if one of the classes gets moved or cancelled.

Many students have complained about the inadequacy of the current Pace registration system – a simple form that allows you to input CRN numbers. This is hardly an improvement from registering by phone or in person. What is needed is a system capable of automatically generating schedules based on a selection of classes, personal profile information and preferences. In addition the system should display the schedules in a graphical, printable format, similar to that offered in the Pace University Class Schedule booklet (see Appendix C). The system should cope with any class changes and cancellations and offer immediate alternatives when such occur.

1.1.2 Objectives and Success Criteria

- BO-1: Reduce the average time required to create a schedule by 75%
Scale: Time spent creating a schedule by a Pace University student
Meter: Examination of system usage logs
Past: 180 minutes
Plan: 45 minutes
- BO-2: Have 70% of students initially introduced to the system continue using it for future registration activities.

1.1.3 Business Risks

- RI-1: Excessive use of the system may cause performance issues and/or shutdown of the Pace University class schedule system. (Probability: 0.05, Impact=9)
- RI-2: Too few students will use the system; the amount of time being spent on system development will be greater than the total amount of time savings as a result of the system. (Probability: 0.3, Impact: 1)
- RI-3: There will be no more support for the system after the current development team graduates from Pace. (Probability: 0.8, Impact: 5)

1.2. Vision of the Solution

1.2.1 Vision Statement

For students who are preparing for next semester's registration, the Pace Schedule Builder is a web-based application that will automatically generate schedules, store relevant profile information, and display schedules in a graphical, printable format, similar to that offered in the Pace University Class Schedule booklet (see Appendix C).

1.2.2 Major Features

- FE-1: Allow user to specify actual classes to take, rather than only CRN numbers
- FE-2: Automatically create schedules based on class selections and profile information.
- FE-3: Save, modify, and delete generated schedules.
- FE-4: Save user information for later use.
- FE-5: Display schedules in a graphical, printable, and easy to understand way.

1.2.3 Assumptions and Dependencies

- AS-1: Required hardware and software will be provided free of charge by Pace University to host the application.
- DE-1: If the Pace ClassSchedule system is decommissioned, a replacement must be found that has access to the internal Pace database.

1.3. Scope and Limitations

1.3.1 Scope of Initial Releases

Feature	Release 1
FE-1	Accept class codes. CRN numbers if time permits
FE-2	Fully implemented
FE-3	Implemented if time permits
FE-4	Implemented if time permits
FE-5	Fully implemented

1.3.2 Limitations and Exclusions

- LI-1: The system will only work for undergraduates at the NYC campus.

1.4. Business Context

1.4.1 Stakeholder profiles

Stakeholder	Major Value	Attitudes	Major Interests	Constraints
Pace Students	Less time spent on registration activities	Very unhappy with the current system, takes too long to register	More time to party	Minimal training in the use of the system

1.4.2 Project Priorities

Dimension	Driver	Constraint	Degree of Freedom
Schedule			Release 1.0 by the end of the 2004 Spring semester
Features		All required features are fully operational	
Quality		95% of user acceptance tests must pass	
Staff	Projected teams size is a project manager, 2 developers, and a QA tester. Project manager and QA tester will help if development is necessary		
Cost			Budget for miscellaneous expenses and supplies should not exceed \$20.

1.5. Business Rules

ID	Rule Definition	Type of Rule	Static or Dynamic
BR-1	Two classes cannot conflict with each other	Fact	Static
BR-2	No classes are allowed during restricted times specified in the user profile	Constraint	Static
BR-3	An entered CRN will receive a higher priority than a course	Constraint	Static
BR-4	No more than 8 schedules may be generated at one time	Constraint	Dynamic
BR-5	A early-bird class is a class starting from 6:00 AM – 8:00 AM	Constraint	Dynamic
BR-6	A morning class is a class starting from 8:00 AM – 12:00 PM	Constraint	Dynamic
BR-7	An afternoon class is a class starting from 12:00 PM – 4:00 PM	Constraint	Dynamic

BR-8	An evening class is a class starting after 4:00 PM	Constraint	Dynamic
BR-9	A small gap between classes is 30 minutes – 90 minutes	Constraint	Dynamic
BR-10	A large gap between classes is 90 minutes – 180 minutes	Constraint	Dynamic
BR-11	A class which is offered less often than another class shall receive a higher priority	Constraint	Static
BR-12	Passwords must be encrypted using a one-way hash algorithm	Constraint	Static

2. Use Cases

Primary Actor	Use Cases
Pace Student	<ol style="list-style-type: none">1. Create new schedule2. Save schedule3. Delete schedule4. Modify schedule5. Create new account6. Modify profile
Administrator	<ol style="list-style-type: none">7. Delete user8. Modify user profile9. Create new user account

Use Case ID:

UC-1

Use Case Name:

Create new schedule

Created By:

Naseer Haniff

Last Updated By:

Nikita Lukish

Date Created:

March 6, 2004

Date Last Updated:

March 31, 2004

Actors:

Pace Student

Description:

The user will be generating their possible schedules after input of data.

Preconditions:

1. User is logged into the system

Postconditions:

None

Normal Course:

1. User will login

2. User will enter constraints in the "User Profile" page.

3. User will notify the system that he is done inputting constraints by clicking on "generate schedules" button

4. System will produce the output of various schedules.

Alternative courses:

None

Exceptions:

None

Includes:

None

Priority:

High

Frequency of Use:

High

Business Rules:

None

Special Requirements:

None

Use Case ID:

UC-2

Use Case Name:

Delete schedule

Created By:

Naseer Haniff

Last Updated By: Nikita Lukish

Date Created:

March 6, 2004

Date Last Updated: March 31, 2004

Actors:

Pace Student

Description:

User will delete selected schedules they do not like and narrow down their choices

Preconditions:

1. User is logged into the system
2. User has already saved at least one schedule

Postconditions:

1. Schedule information is deleted from the database

Normal Course:

1. Student clicks on delete schedule button
2. System deletes the schedule information from the database

Alternative courses:

None

Exceptions:

None

Includes:

None

Priority:

Medium

Frequency of Use:

2-3 times per day, 5-10 around registration time

Business Rules:

None

Special Requirements:

None

Use Case ID:

UC-3

Use Case Name: Save schedule

Created By: Naseer Haniff

Last Updated By: Nikita Lukish

Date Created: March 6, 2004

Date Last Updated: March 31, 2004

Actors: Pace Student

Description: After user is satisfied with all results, then he/ she will then save their desired schedules.

Preconditions:

1. User is logged into the system
2. User has already created a schedule

Postconditions: Schedule information is saved in a database

Normal Course:

1. User fills out all data fields.
2. User sees possible choices.
3. User then decides which schedules is most preferred and then saves.

Alternative courses: None

Exceptions: None

Includes: None

Priority: Medium

Frequency of Use: High

Business Rules: None

Special Requirements: None

Use Case ID:

UC-4

Use Case Name: Modify schedule

Created By: Naseer Haniff

Last Updated By: Nikita Lukish

Date Created: March 6, 2004

Date Last Updated: March 31, 2004

Actors: Pace Student

Description: After selecting preferred schedules from the possible layouts, the user will then have the option to modify the schedule if necessary.

Preconditions:

1. User is logged into the system
2. User has already saved at least one schedule

Postconditions:

1. Schedule information is changed in the database

Normal Course:

1. User will view all generated choices.
2. User will then select the desired choices.
3. User will then decide to add or remove a selected class or classes from the selected choices.
4. User will restart the process with their selected schedule to regenerate a whole new one with the new class added in.

Alternative courses: None

Exceptions: None

Includes: None

Priority: Medium

Frequency of Use: High

Business Rules: None

Special Requirements: None

Use Case ID:

UC-5

Use Case Name:

Create new account

Created By:

Viktor Geller

Last Updated By:

Date Created:

March 6, 2004

Date Last Updated: March 6, 2004

Actors:

Pace Student, Administrator

Description:

A user would create a new account by specifying their first name, last name, email address, username and a password.

Preconditions:

None

Postconditions:

1. Account information is stored in PSB database
2. A profile with default values is created for the user.

Normal Course:

1. User goes to the PSB website.
2. User clicks on the new account button.
3. User enters last name, first name, email address, username, and a password.
4. User confirms the information is correct.
5. System stores information in the database
6. System creates a default profile for the user.

Alternative courses:

None

Exceptions:

Username already exists in the system

1. System displays an error message suggesting that another username should be chosen (at step 4)
2. Go back to start of use case

Includes:

None

Priority:

Medium

Frequency of Use:

2-3 times per day, 5-10 around registration time

Business Rules:

BR-12

Special Requirements:

None

Use Case ID:

UC-6

Use Case Name:

Modify Profile

Created By:

Nikita Lukish

Last Updated by:

Nikita Lukish

Date Created:

March 06, 2004

Date Last Updated:

March 06, 2004

Actors:

Student

Description:

A student accesses the Pace Schedule Builder system, and modifies his/her profile, which consists of his/her personal settings - both required (password) and optional (email address, etc.) as well as a number of constraints, which are needed to generate schedules.

Preconditions:

1. A student is registered with PSB system.
2. A student is logged onto PSB.

Postconditions:

1. A student's record in the database is updated to reflect changes to his/her profile.

Normal Course:

2. Constrains selected by the student are checked for consistency
- 6.0 Edit a single setting

1. Student asks to view his/her profile.
2. System displays a list of student's personal settings.
3. Student selects a setting (s)he wants to change.
4. Student changes the setting.
5. Student indicates that the change is complete.
6. System makes sure that new settings are valid and consistent with each other.
7. System confirms whether the student wants to save new settings.
8. System updates the database to include new settings.
9. System notifies the student that the change is complete.

Alternative courses:

- 6.1 Change multiple settings (branch after step 4)

Exceptions:

1. Return to step 3.
- 6.0.E.1 New setting is invalid or inconsistent with student's other selections (at step 6)

1. System alerts the student that the new setting conflicts with other selections
2. Return to step 2.

- 6.0.E.2 Student chooses not to save new settings (at step 7)

Includes:

None

Priority:

High

Frequency of Use:

Hard to predict

Business Rules:

None

Special Requirements:

None

Use Case ID:

UC-7

Use Case Name:

Delete User

Last Updated By:

Created By:

Eric Chan

Date Last Updated: March 6, 2004

Date Created:

March 6, 2004

Actors:

User

Description:

Allows a user with administrative rights to have the ability to delete user accounts stored in the system.

Preconditions:

1. User is logged into PSB

Postconditions:

1. User has administrative rights

Normal Course:

Deletion of user

1. User request for deletion of targeted user account.
2. System verifies account being targeted.
3. System ask user to confirm deletion of targeted account.
4. User confirms deletion.
5. System deletes targeted account
6. System informs the user of completion.

Alternative courses:

None

Exceptions:

Detection of Administrator (branches after step 2)

1. System informs user account is an admin.
2. Return to step 1

Includes:

Authenticate user's identity

Priority:

Low

Frequency of Use:

Rare

Business Rules:

None

Special Requirements:

None

Use Case ID:	UC-8		
Use Case Name:	Modify User Profile		
Created By:	Nikita Lukish	Last Updated by:	Nikita Lukish
Date Created:	March 06, 2004	Date Last Updated:	March 06, 2004
Actors:	Administrator		
Description:	An administrator accesses the Pace Schedule Builder system, and modifies his/her own or any other user's profile. Profiles consist of a user's personal settings - both required (password) and optional (email address, etc.) as well as a number of constraints, which are needed to generate schedules.		
Preconditions:	1. An administrator is registered with PSB system. 2. An administrator is logged onto PSB.		
Postconditions:	1. A user's (student's or administrator's) record in the database is updated to reflect changes to his/her profile. 2. Settings entered by the administrator are checked for consistency		
Normal Course:	8.0 Edit a single setting 1. Administrator asks to go to "view profile" mode. 2. System presents a list of all users of the system 3. Administrator selects a user for whom the settings are to be changed 4. System displays a list of selected user's personal settings. 5. Administrator selects a setting he wants to change. 6. Administrator changes the setting. 7. Administrator indicates that the change is complete. 8. System makes sure that new settings are valid and consistent with each other. 9. System confirms whether the administrator wants to save new settings. 10. System updates the database to include new settings. 11. System notifies the administrator that the change is complete.		
Alternative courses:	8.1 Change multiple settings (branch after step 6)		
Exceptions:	1. Return to step 5. 8.0.E.1 New setting is invalid or inconsistent with user's other selections (at step 8) 1. System alerts the administrator that the new setting conflicts with user's other settings elections 2. Return to step 4. 8.0.E.2 Administrator chooses not to save new settings (at step 9)		
Includes:	1. Return to step 4.		
Priority:	None		
Frequency of Use:	High		
Business Rules:	Hard to predict		
Special Requirements:	None		

Use Case ID:

UC-9

Use Case Name:

Add User

Last Updated By:

Created By:

Eric Chan

Date Last Updated: March 6, 2004

Date Created:

March 6, 2004

Actors:

User

Description:

Allows a user with administrative rights to have the ability to add new user accounts to the system.

Preconditions:

1. User is logged into PSB

Postconditions:

1. User has administrative rights

Normal Course:

Adding of new User

1. Users request to input a new account.
2. System verifies account.
3. System ask user to confirm creation of new account.
4. User confirms addition.
5. System creates new account.
6. System informs the user of completion.

Alternative courses:

None

Exceptions:

Duplicate account names (branches after step 2)

1. System informs user the name is already in use.
2. Return to step 1.

Includes:

Authenticate User's Identity

Priority:

High

Frequency of Use:

Once per new user on average

Business Rules:

BR-12

Special Requirements:

None

3. Software Requirements Specification

3.1. Introduction

3.1.1 Purpose

This Software Requirements Specification describes the software functional and nonfunctional for release of the Pace Schedule Builder (PSB). This document is intended for the purpose of creating and specifying the functionality requirements of the system.

3.1.2 Project Scope and Project Features

The Pace Schedule Builder allows users to create and manage class schedules with ease. Users will also be able to save and retrieve previous schedules generated.

3.2. Overall Description.

3.2.1 Product Perspective

The Pace Schedule Builder system is a new system that replaces the current manual process for creating schedules. The context diagram in Figure D-1 illustrates the external entities and system interfaces for release 1.0.

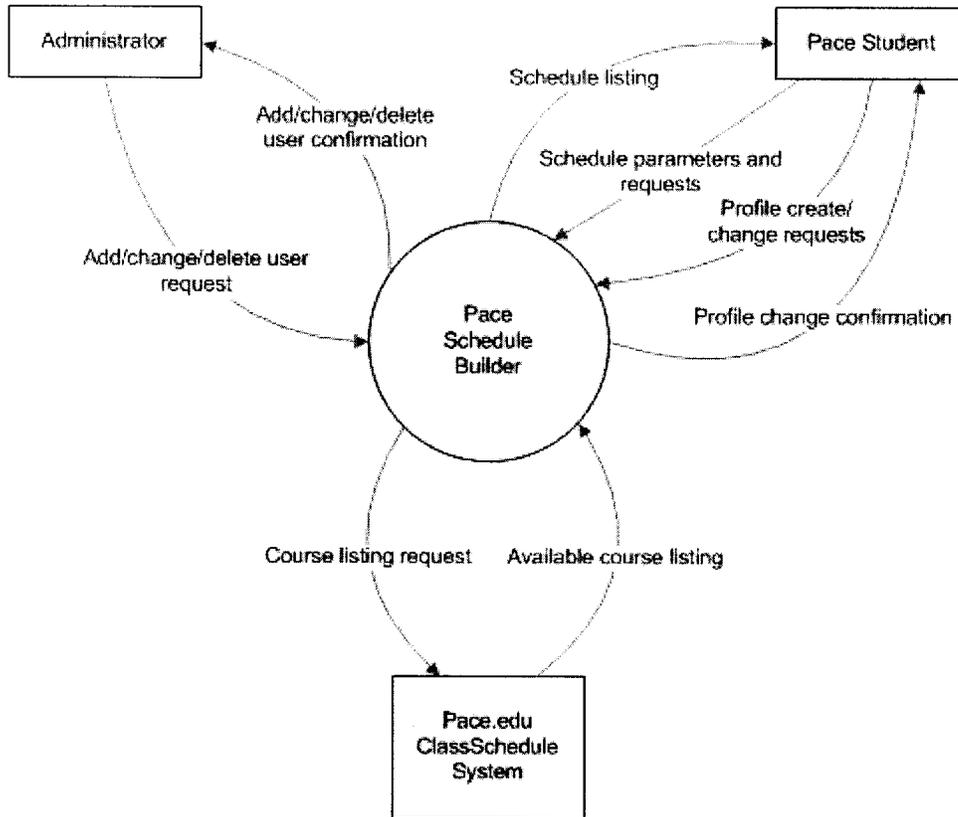


Figure D-1: Context diagram for release 1.0 of Pace Schedule Builder System

3.2.2 User Classes and Characteristics

User Class	Description
Administrator	An administrator is a user that can modify other user's settings. At beginning, there will only be administrator accounts for the members of the team "Untouchables". However, more administrators can be added in the future.

Student	The student is a Pace university student, who will register with the system, then select his personal preferences and launch the process for generating university schedules. The training for the students should be minimal.
---------	--

3.2.3 Operating environment

- OE-1: The Pace Schedule Builder will operate with the following web browser: Microsoft Internet Explorer version 6.0.
- OE-2: The Pace Schedule Builder will run on Linux and use the Tomcat 4.1 server
- OE-3: The system will use Oracle as the database

3.2.4 Design and Implementation Constraints

- CO-1: All HTML code must conform to the HTML 4.0 standard
- CO-2: All code will be written in Java (JavaScript and DHTML may be used in the GUI)
- CO-3: Must interface with the Pace ClassSchedule system

3.2.5 User Documentation

- UD-1: The system shall provide a basic online user tutorial.

3.3. Other Nonfunctional Requirements

3.3.1 Performance Requirements

- PE-1: The system shall accommodate 20 users during peak usage periods during registration week, with an estimated average session of 45 minutes.
- PE-2: A set of 6 schedules must be created in less than 30 seconds.
- PE-3: The login process shall take no longer than 10 seconds.

3.3.2 Security Requirements

- SE-1: All passwords shall be encrypted per BR-12.
- SE-2: Users must be logged into the system prior to operation.
- SE-3: Only an Administrator will be allowed to delete user accounts.
- SE-4: The system shall not permit students to view the schedules of other students.

3.3.2 Software Quality Attributes

- Availability-1: The PSB system must be available to all users 99% of the time.

Appendix A – Data Dictionary

class = *a 2 or 3-character alpha id followed by a numeric 3-digit section number (ie CS 389)*

course = CRN number
 + title
 + status
 + 1:m{meeting Time}
 + room
 + site
 + fee
 + remark
 + instructor
 + rank

course listing = 1:m{course}

CRN Number = *a unique 5-digit numeric number assigned by Pace University for each course offering*

end time = *time class ends in the format HH:MM AM/PM*

fee = *amount of money that is the fee for the course. Numeric*

instructor = *name of the instructor for the course. 50 characters alphanumeric*

meeting time = start time
 + end time

start time = *time class starts in the format HH:MM AM/PM*

student first name = *the first name of the student registering for the system. 50 characters alphanumeric*

student email = *the email of the student registering for the system. 50 characters alphanumeric*

student last name = *the last name of the student registering for the system. 50 characters alphanumeric*

student password = *the password chosen by the student registering for the system. 15 characters alphanumeric*

student username = *the user name chosen by the student registering for the system. 15 characters alphanumeric*

rank = *a numeric rank from 0 to 1000 assigned to each course by the system based on the user's profile and course selections. Higher rank means higher priority.*

status = [Cancelled | Closed | ""]

room = *room in which course is offered. 10 characters alphanumeric*

remark = *any remarks about the course. 100 characters alphanumeric*

schedule = 1:m{course}

schedule listing = 1:m{schedule}

site = *name of campus on which course is offered. 10 characters alphanumeric*

title = *the title of a course. 50 characters alphanumeric*

Appendix B – User Interface Prototype

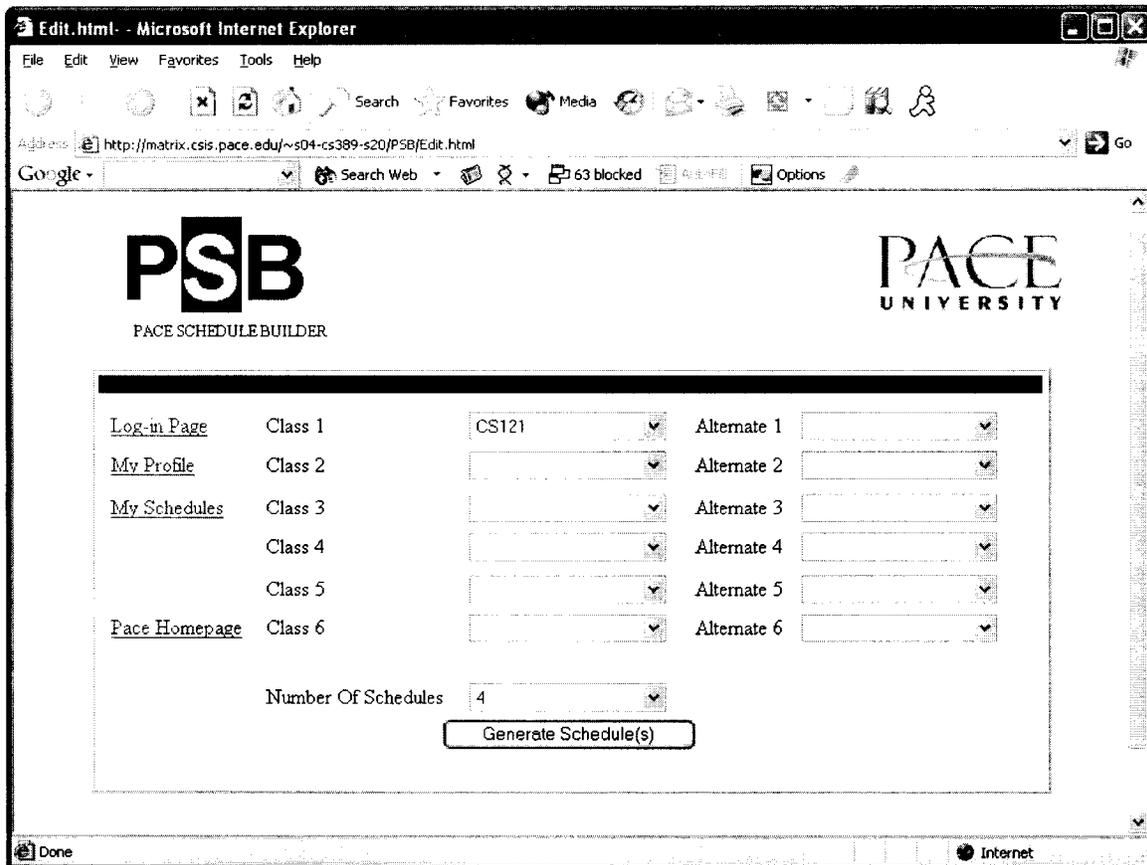


Figure D-2: Prototype for class information input screen

