Student Scheduling System User Manual

Student Scheduling System

Team 06

Douglass Kinnes Alexey Tregubov Mihir Daptardar Ihsan Tolga Simone Lojeck

- : Project Manager, Quality Focal Point, Implementer
- : System Architect, UML Modeler, Implementer
- : Operational Concept Engineer, Quality Focal Point, Implementer
 - : Life Cycle Planner, Feasibility Analyst, Implementer
 - : IV&V, Quality Focal Point, Shaper

Version History

Date	Author	Version	Changes made	Rationale
03/29/2013	Ihsan Tolga	1.0	 Draft version of UM is formed. Initial information and guidelines added. 	 To satisfy the requirements of the CCD/TRR and to provide instructions to the users how to use the scheduling system software.
04/05/2013	Ihsan Tolga	1.1	 Sections updated. Guidelines for operation added. 	 Updates made with respect to upcoming CCD.
04/13/2013	Ihsan Tolga	1.5	 Screenshots updated. Minor revisions made. Admin side is updated with respect to the CCD session. Error acknowledgments added. 	 CCD concern evaluated and changes made. New screenshots due to new UI.
04/20/2013	Ihsan Tolga	1.7	 Student Side updated. Admin side screenshot updated. 	 Updates/revisions made for TS with full system capability.

Table of Contents

VE	VERSION HISTORYII									
ТА	TABLE OF CONTENTSIII									
ТА	BLE OI	TABLESIV								
ТА	BLE OI	FIGURESV								
1.	Intro	oduction1								
	1.1	System Overview1								
	1.2	System Requirements								
2.	2. Installation Procedures									
	2.1	Initialization procedures								
	2.2	Re-installation								
	2.3	De-installation								
3.	3. Operational Procedures									
4.	4. Troubleshooting									
	4.1	Frequently Asked questions								
	4.2	Error Codes and Messages								
	4.3	Note								

User Manual (UM)

Table of Tables

No table of figures entries found.

Table of Figures

Figure 1: Starting PLAY Console
Figure 2: Courses Page
Figure 3: Add Course Page
Figure 4: Add Course Group Page
Figure 5: Course Group Page
Figure 6: Requirements Page7
Figure 7: Degree Program Page
Figure 8: Documentation Page
Figure 9: Degree Selection Page9
Figure 10: Degree Selection Page - Courses
Figure 11: Study Plan Page
Figure 12: No Study Plan Page
Figure 13: Course Add Page
Figure 14: Course Edit Page
Figure 15: Missing Login Screen
Figure 16: Degree Program Add Page

1. Introduction

1.1 System Overview

The purpose of Student Scheduling System is to provide a study plan creation facilitator for both undergrad students and faculty of Steven's Institute of Technology.

With current manual method, forming a single study plan for a single student may take hours of collaborative work of advisors and students and regarding each student has different course history or plans, they need to form unique study plan for the majority of the students.

By using the Student Scheduling System; this process is automated; in other words, after entering the data representing his/her course history/graduation plans, a student can get possible study plan generated by the system. And for the administrative side; they can add/remove courses to the system, modify courses' features/information, form new course groups and create new degree programs. Meanwhile they are also able to delete/modify these course groups, degree programs or even requirements to complete these programs.

1.2 System Requirements

- Minimum hardware/software requirements for running the Student Scheduling System are given below under the related headers.
- Requirements might show minor differences among the different systems or hardware/OS equipment. Therefore; the given requirements below represent a recommended configuration as well to operate the Student Scheduling System properly.

1.2.1 Hardware Requirements

Server Side Hardware Requirements:

- Processor : Intel® Xeon E3xxx Processor or equivalent.
- Memory : 3GB
- Storage : 5GB for database tier
- Data Bandwidth : (Relative to the simultaneous numbers of users.) ~1 Mbit/s
- Domain Name and Dedicated IP address
- I/O Devices for Maintenance Issues

User Side Hardware Requirements:

- Processor : Intel® Centrino/Ix/Celeron/PentiumIV; AMD AthIon/Phenom/FX processors or equivalent.
- Memory : 1GB free space
- Internet Connection
- I/O Devices to operate Web Browsers properly.

1.2.2 Software Requirements

Server Side Software Requirements:

- UNIX Server or Windows Server
- Java Runtime Environment 6 or higher.
- MySQL
- PLAY framework for Java
- Java Editor (Recommended for maintenance issues)
- XAMPP (Recommended for maintenance issues)
- Java IDE for maintenance.

User Side Software Requirements:

- Windows, Linux, MacOSx or equivalent (mobile) operating system that can operate supported web browsers.
- Google Chrome, Mozilla Firefox web browser. (Strongly recommended.)
- Java Runtime Environment 6 or higher.

1.2.3 Other Requirements

- It is highly recommended for the student users to be familiar about the courses and their course histories, especially in the cases they are transfer students.
- For administrative side; it is highly recommended to be familiar about the relations between course groups and their effects on graduation satisfaction.
- It is highly recommended for maintainers to feel comfortable with Java and Web developing and using MySQL like relational database software.

2. Installation Procedures

Follow the steps below to complete installation of Student Scheduling System successfully:

- Copy the "student_scheduling_system" directory into the server hard-drive.
- Install Java PLAY! Framework to the system if it is not already installed.
- Configure the system's path for operating easiness.
- Start database application. "db_schema", "db_init_values"
- Open Java PLAY! Framework console from command window. Run the "student_scheduling_system" application under the related directory.
- You can reach the working version on "localhost". (Preferably on port 9000)
- Enter the domain address in address bar of web browser.

2.1 Initialization procedures

- Copy the "student_scheduling_system" directory into the server hard-drive.
- Install Java PLAY! Framework to the system if it is not already installed.
- Configure the system's path for operating easiness.
- Start database application. "db_schema", "db_init_values"
- Open Java PLAY! Framework console from command window. Run the "student_scheduling_system" application under the related directory.

- You can reach the working version on *"localhost"* by calling it from your browsers address space. (Preferably on port:9000)
- Enter the domain address in the address bar of your web browser.

<pre>[error] application - t Internal server error, for request [POST /courses/20] -> jaua.util.concurrent.TimeoutException: Futures timed out after [300000] millisec onds at akka.dispatch.DefaultPromise.ready(Future.scala:870> ~[akka-actor.jar :2.0.2] at akka.dispatch.DefaultPromise.result(Future.scala:874> ~[akka-actor.jar :2.0.2] at akka.dispatch.Mwait\$.result(Future.scala:74> ~[akka-actor.jar:2.0.2] at akka.dispatch.BefaultPromise.result(Future.scala:874> ~[akka-actor.jar:2.0.2] at akka.dispatch.BefaultPromise.result(Future.scala:974> ~[akka-actor.jar:2.0.2] at akka.dispatch.BefaultPromise.resultClasses.sendHandler\$1(Server.scala:59> [play_2.] [Infol play - Application started (Dev) [Student_scheduling_system]\$ [Infol play - Application from Cf.VUSers\pol>Besktop\USC Computer Science [Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_system [Infol beding project definition from Cf.VUSers\pol>Besktop\USC.20577b/20Software/ poleytextop/USC/20Computerx20Sciencex20Gclasses/20And/20Works/CSCI 20577b/20Software/ poleytextop/USC/20Computerx20Sciencex20Classes/20And/20Works/CSCI/20577b/20Software/ poleyt 2.0.4, ht</pre>		
<pre>! Internal server error, for request [POST /courses/201 -> java.util.concurrent.TimeoutException: Futures timed out after [300000] millisec onds</pre>	🔤 Komut İstemi - play	
<pre>java.util.concurrent.TimeoutException: Futures timed out after [300000] millisec onds at akka.dispatch.DefaultPromise.ready(Future.scala:870) ~[akka-actor.jar :2.0.2] at akka.dispatch.Await\$.result(Future.scala:874) ~[akka-actor.jar :2.0.2] at akka.dispatch.Await\$.result(Future.scala:74) ~[akka-actor.jar:2.0.2] at play.core.Reloadable0pplication.get(ApplicationProvider.scala:108) ~[play_2.9.1.jar:2.0.4] 9.1.jar:2.0.4</pre>	[error] application -	^
<pre>onds at akka.dispatch.DefaultPromise.ready(Future.scala:870) ~[akka-actor.jar :2.6.2] at akka.dispatch.DefaultPromise.result(Future.scala:874) ~[akka-actor.jar :2.6.2] at akka.dispatch.Await\$.result(Future.scala:74) ~[akka-actor.jar:2.0.2] at play.core.ReloadableApplication.get(ApplicationProvider.scala:108) ~[play_2.9.1.jar:2.0.4] at play.core.server.Server\$class.sendHandler\$i(Server.scala:59) [play_2. jar:2.0.4] at play.core.server.Server\$class.sendHandlerFor\$4.apply(Server.scala:8 play_2.0.4] at play.core.server.Server\$class.sendHandlerFor\$4.apply(Server.scala:8 [info] play_1.jar:2.0.4] scala:1.jar:2.0.4] Scala:2.0.4] Scal</pre>	! Internal server error, for request [POST /courses/20] ->	
at akka.dispatch.DefaultPromise.ready(Future.scala:870) ~[akka-actor.jar at akka.dispatch.DefaultPromise.result(Future.scala:874) ~[akka-actor.jar r:2.0.2] at akka.dispatch.Await\$.result(Future.scala:74) ~[akka-actor.jar:2.0.2] at play.core.ReloadableApplication.get(ApplicationProvider.scala:108) ~[play_2.9.1.jar:2.0.4] at play.core.server.Server\$\$anonfun\$getHandler\$1(Server.scala:59) [play_2. 9.1.jar:2.0.4] at play.core.server.Server\$\$anonfun\$getHandlerFor\$4.apply(Server.scala:8 9) [play_2.9.1.jar:2.0.4] [info] play - database [default] connected at jdbc:mysql://localhost/student_scheduling_system [info] play - Application started (Dev) [student_scheduling_system] \$ C:\Users\pc\Desktop\USC Computer Science Classes and Works\CSCI 577b Software En gineering B\skedch\student_scheduling_system\play [info] Loading project definition from C:\Users\pcDesktop\USC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_sys tem\project linfo] Set urrent project to student_scheduling_system (in build file:/C:/Users /pc/Desktop/USC20Computer%20Science%20Classes%20and%20Works/CSCI%20577b%20Software%20So		l millisec
at akka.dispatch.DefaultPromise.result(Future.scala:874) ~[akka-actor.ja r:2.0.2] at akka.dispatch.Await\$.result(Future.scala:74) ~[akka-actor.jar:2.0.2] at play.core.ReloadableApplication.get(ApplicationProvider.scala:108) ~[play_2.9.1.jar:2.0.4] at play.core.server.Server\$\$anonfun\$getHandler\$1(Server.scala:59) [play_2. 9.1.jar:2.0.4] by [play_2.9.1.jar:2.0.4] connected at jdbc:mysql://localhost/student_sch eduling_system infol play - database [default] connected at jdbc:mysql://localhost/student_sch eduling_system infol play - Application started (Dev) Istudent_scheduling_system] \$ C:\Users\pc\Desktop\USC Computer Science Classes and Works\CSCI 577b Software En gineering B\skedch\student_scheduling_system\play Calasses and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_sys tem\project Infol Set current project to student_scheduling_system (in build file:/C:/Users pc/Desktop/USC/20Computer%20Science%20Classes%20and%20Works/CSCI%20577b%20Software infol Set current project to student_scheduling_system/ play? 2.0.4, http://www.playframework.org > Type "help play" or "license" for more information. > Type "exit" or use Ctrl+D to leave this console.	at akka.dispatch.DefaultPromise.ready(Future.scala:870) ~[akka-	actor.jar
at akka.dispatch.Await\$.result <future.scala:?4> ~[akka-actor.jar:2.0.2] at play.core.ReloadableApplication.get(ApplicationProvider.scala:108> ~[play_2.9.1.jar:2.0.4] at play.core.server.Server\$class.sendHandler\$1(Server.scala:59) [play_2. 9.1.jar:2.0.4] at play.core.server.Server\$\$anonfun\$getHandlerFor\$4.apply(Server.scala:8 p) [play_2.9.1.jar:2.0.4] [info] play - database [default] connected at jdbc:mysql://localhost/student_sch eduling_system [info] play - Application started (Dev) [student_scheduling_system] \$ C:\Users\pc\Desktop\USC Computer Science Classes and Works\CSCI 577b Software En gineering B\skedch\student_scheduling_system>[info] Loading project definition from C:\Users\pc\Desktop\USC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_sys tem\project linfo] Set current project to student_scheduling_system (in build file:/C:/Users /pc.Desktop/USC/20Computerx20Sciencex20Classesx20and/20Works/CSCI/20577b/20Software ###220Brgineering/20B/skedch/student_scheduling_system/ [info] Set current project to student_scheduling_system (in build file:/C:/Users /pc/Desktop/USC/20Computerx20Sciencex20Classesx20and/20Works/CSCI/20577b/20Software/ ###20Brgineering/20B/skedch/student_scheduling_system/ [info] Set current project to student_scheduling_system/ /pier/Desktop/USC/20Computerx20Sciencex20Classes/20and/20Works/CSCI/20577b/20Software/ /pier/Desktop/USC/20Forputerx20B/skedch/student_scheduling_system/] [info] Set current project to student_scheduling_system/ /</future.scala:?4>	at akka.dispatch.DefaultPromise.result(Future.scala:874) ~[akka	-actor.ja
at play.core.server.Server\$class.sendHandler\$1(Server.scala:59) [play_2. 9.1.jar:2.0.41 at play.core.server.Server\$\$anonfun\$getHandlerFor\$4.apply(Server.scala:8 9) [play_2.9:1.jar:2.0.41 Info] play - database [default] connected at jdbc:mysql://localhost/student_scheduling_system Finfo] play - Application started (Dev) Istudent_scheduling_system\$ SciUsers\pc>DesktopNUSC Computer Science Classes and Works\CSCI 577b Software En gineering B\skedch\student_scheduling_system>play Info] Loading project definition from C:\Users\pc>DesktopNUSC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_system temproject definition from C:\Users\pc>DesktopNUSC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_system temproject definition from C:\Users\pc>DesktopNUSC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_system temproject Users /pc>Desktop/USC/20Computer*20Science*20Classes*20and*20Works/CSCI*20577b*20Software are*20Engineering*20B/skedch/student_scheduling_system/> playf 2.0.4, http://www.playframework.org > Type "help play" or "license" for more information. > Type "exit" or use Ctrl+D to leave this console.	at akka.dispatch.Await\$.result(Future.scala:74) ~[akka-actor.ja at play.core.ReloadableApplication.get(ApplicationProvider.scal	
at play.core.server.Server\$\$anonfun\$getHandlerFor\$4.apply(Server.scala:8 9) [play_2.9;1.jar:2.0.4] Infol play - database [default] connected at jdbc:mysql://localhost/student_sch eduling_system Infol play - Application started (Dev) Istudent_scheduling_system] \$ C:\Users\pc.Desktop\USC Computer Science Classes and Works\CSCI 577b Software En gineering B\skedch\student_scheduling_system/play Infol Loading project definition from C:\Users\pc.Desktop\USC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_system \pc.Desktop\USC 200Computer X20Science220Classes.20andX20Works/CSCI x20577bX20Software \pc.Desktop\USC200Computer.220Science220Classes.20andX20Works/CSCI x20577bX20Software areX20Engineeringx20B/skedch/student_scheduling_system/> 	at play.core.server.Server\$class.sendHandler\$1(Server.scala:59)	[play_2.
<pre>[info] play - database [default] connected at jdbc:mysql://localhost/student_sch eduling_system [info] play - Application started (Dev) [student_scheduling_system] \$ [C:\Users\pc:Desktop\USC Computer Science Classes and Works\CSCI 577b Software En gineering B\skedch\student_scheduling_system>play [info] Loading project definition from C:\Users\pc:Desktop\USC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_sys tem\project [info] Set current project to student_scheduling_system (in build file:/C:/Users /pc/Desktop/USC200Computer*20Science*20Classes/20and*20Works/CSCI*20577b*20Softw are*20Engineering%20B/skedch/student_scheduling_system/> </pre>	at play.core.server.Server\$\$anonfun\$getHandlerFor\$4.apply(Serve	r.scala:8
<pre>Linfol play = Application started (Dev) Istudent_scheduling_system] \$ CstUsers'pc>DesktopNUSC Computer Science Classes and Works\CSCI 577b Software En gineering B\skedch\student_scheduling_system/play Linfol Loading project definition from C:VUsers\pc>Desktop\USC Computer Science Classes and Works\CSCI 577b Software Engineering B\skedch\student_scheduling_system/play Linfol Set current project to student_scheduling_system (in build file:/C:/Users /pc/Desktop/USC200Computer/20Science/20Classes/20Band/20Works/CSCI/20577b/20Softw are/20Engineering/20B/skedch/student_scheduling_system/></pre>	[info] play - database [default] connected at jdbc:mysql://localhost/st	udent_sch
C: VUsers \pc \Desktop \USC Computer Science Classes and Works \CSCI 577b Software En gineering B\skedch\student_scheduling_system>play Infol Loading project definition from C: \Users \pc \Desktop \USC Computer Science Classes and Works \CSCI 577b Software Engineering B\skedch\student_scheduling_sys tem\project Infol Set current project to student_scheduling_system {in build file:/C:/Users /pc/Desktop/USC/20Computer%20Science%20Classes%20and%20Works/CSCI%20577b%20Softw are%20Engineering%20B/skedch/student_scheduling_system/> //	[info] play - Application started (Dev)	
<pre>[info] Set current project to student_scheduling_system (in build file:/C:/Users /pc/Desktop/USC/20Computerx20Science:20Classes:20and/20Works/CSCI:20577bx20Softw arex20Engineeringx20B/skedch/student_scheduling_system/> </pre>	gineering Bvskedch\student_scheduling_system>play [info] Loading project definition from C:\Users\pc\Desktop\USC Computer Classes and Works\CSCI 577D Software Engineering B\skedch\student_sched	Science
i_i i play! 2.0.4, http://www.playframework.org > Type "help play" or "license" for more information. > Type "exit" or use Ctrl+D to leave this console.	[info] Set current project to student_scheduling_system (in build file:	/C:/Users bx20Softw
> Type "help play" or "license" for more information. > Type "exit" or use Ctrl+D to leave this console.		
> Type "exit" or use Ctrl+D to leave this console.	play! 2.0.4, http://www.playframework.org	
[student_scheduling_system] \$ run *	> Type "help play" or "license" for more information. > Type "exit" or use Ctrl+D to leave this console.	=
	[student_scheduling_system] \$ run	+

Figure 1: Starting PLAY Console

2.2 Re-installation

- Delete the path from the path address bar under environment settings.
- Execute *db_drop_schema* under *db* directory in source files directory for database reset.
- Delete the directory includes source files and .java classes from the hard-drive.
- Stop the PLAY! Framework (*Ctrl+d*), clean the workspace. Exit from PLAY! Console.
- Repeat the procedure for initial installation.

2.3 De-installation

Follow the steps below to successfully uninstall Student Scheduling System from the server.

- Delete the path from the path address bar under environment settings.
- Execute "db_drop_schema" under "db" directory in source files directory for database reset.
- Delete the directory includes source files and .java classes from the hard-drive.
- Stop the PLAY! Framework (Ctrl+d), clean the workspace. Exit from PLAY! Console.
- Delete the PLAY! Directory from the hard-drive.

3. Operational Procedures

For the Administrative Users:

- On login screen; type defined user name and password in related input boxes. Then click on the login button. The system will check your authorization for related side admin or student.
- You can return the homepage by clicking on **Home** button at any time.
- From the menu on the left of the screen; select the desired section. There are also links representing recent path in breadcrumbs form at each page's top section to return back.
 - For Administrative Users:
 - <u>Courses:</u> The courses and the information related to them in database are shown in the table as a list. To add a new course; click on the **Add Course** button on top of the list. To edit a course, click on the **Edit** button next to the course's title. To delete a course, click on the **Delete** button next to the course's title and confirm the operation on alert box by simply clicking on the "Yes".

	Docs fo	or TRR - Dropi	bex >	Course List × +	SOLULA NUMBER	an any second to			-0
longbeac	husc.ed	u :22825/cour	ses/					☆ ♥ C	P 🟦
tudent	Scl	hedul	ing Syst	em					
ourse									
Jourse		cetor							
ogout 🤇	curses								
	Cour	ses:							
	idd Ne	w Course							
urse Groups	Bdit	Delete	Course Number	Title	Preregs	Coreqs	Credits		
uirements	Edit	Delete		Data Structures	C\$115	CS135	3		
xee [Edit	Delete	CS135	Discrete Structures			3		
zrams	Edit	Delete	CS115	Introduction to Computer Science			4		
umentation	Edit	Delete	CS347	Software Development Process	((CS181 OR CS284) AND CS135)		3		
	Edit	Delete	CS105	Introduction to Scientific Computing			3		
[Edit	Delete	HPL111	Philosophy I			3		
	Edit	Delete	HPL112	Knowledge, Reality and Nature			3		
	Edit	Delete	HPL339	Ethics			3		
	Edit	Delete	HPL348	Aesthetics			3		
	Edit	Delete	CS146	Web Fundamentals			3		
[Edit	Delete	C8334	Automata & Computation	(CS135 AND CS115)		3		
	Edit	Delete	CS383	Computer Organization & Programming	C\$115	(CS284 OR CS181)	3		
	Edit	Delete	CS385	Algorithms	(CS284 OR CS181)		4		
	Edit	Delete	CS516	Compiler Design	((CS182 OR CS590) OR CS385)		3		
	Edit	Delete	C8521	TCP/IP Networks	(CS492 OR CS520)		3		
	Edit	Delete		Mobile Systems and Applications	((CS590 OR CS182) OR CS385)		3		
	Edit	Delete		Artifical Intelligence	((CS182 OR CS385) OR CS590)		3		
	Edit			Introduction to Developing Internet Applications	SOC605		3		
	Edit	Delete	SOC521	Software Requirements Analysis and Engineering	SOC606		3		
	Edit			Software Architecture and Component-Based Design			3		
	Edit	Delete	SOC567	Software Testing, Quality Assurance, and Maintenan			3		
	Edit	Delete	SOC605	Intoduction to Service Oriented Computing	C\$115		3		
	Edit	Delete		Software Engineering	((CS115 OR CS135) AND CS385)	SOC521	4		
n	Edit	Delete	CS105	Introduction to Scientific Computing			3		

Figure 2: Courses Page

Add New Course: You should enter the Title in a string form (the system will produce abbreviation from that title i.e. HUM, CS), Course Numbers / Credits as integers, "on campus" / "online" information by checking the related checkboxes. Finally, for the Prerequisites (Prereq) and Corequisites; you should enter those courses' names as a combination of Prefix and Course Number (i.e. CS105) by leaving a space between different courses (if there are any). Click on the Submit button to add the course.



Figure 3: Add Course Page

- <u>Course Groups</u>: The course groups and their related information are shown on the table on this page. To edit/delete any course group, click on the **Delete** or **Edit** button next to the related course's name. To add/form a new course group, click on the **Add New Course Group** button on top of the list.
 - Add New Course Group: You should enter the name of the new course group in related text box and the abbreviation for this particular course group in the given text boxes. And regarding which way is desired to form the new course group, either List of Course or Combination of Groups should be selected. Click on the Submit button after the required information is entered.

Docs for TRR - Dropbox	< > Course Group Add × +	AUGUSTAN DESCRIPTION AND INCOMENTS		
longheach.use.edu/20025/coune-groups/adu	KouseGeopPage/		🟫 👻 🖱 🛛 🛃 – Geogle	P 🔒
dent Scheduling Sy	vstem			
irse Director				
inse Director				
Course Groups > Course Grou	p Add			
Name:	entername	Course Group Reference		
	Name of the new Course Group			
Groups D List of Courses		Abbreviation Title H11.02 Humanities: 305-200 level		
	Course Numbers in the Course Group separated by con C\$115, C\$225, C\$135	nas ie.: 1111.02 Humanities: 100-290 level 1131.02 Humanities: 300 level or higher		
cienta	CSIID, CS225, CSID	HUM03 Humanities		
 Combination of Groups 	Boolean equation made with Course Group Abbreviatio			
13	HUM01 AND HUM02	SOE02 Software Electives		
	Submit Query	CS-02 Computer Science - Mandatory		
enterlices F		S-P03 Science - Physics		
		S-Cl4 Science - Chemistry		
		S-C05 Science - Chem & Bio		
		S-C06 Science - Chem & Bio2		
		S-P04 Science - Phys & Bio		
		SDE02 Software Development Electives		
		C3L03 CS 300 Lavel		
		C4L03 C8 400 Level M M03 Mathematics - Mandatory		
		M-M04 Management - Mandatory		
		II-803 Humanitites - Special		
		H-203 Humanities - 200 Level		
		H-102 Humanities - 100 level		
		H-302 Humanities - 300 level		
		11-402 Humanities - 400 level		
		H-L02 Humanities - Literature		
		II-P02 Humanities - Philosophy		
		H-H02 Humanities - History		
		II-S04 Illumanities - Social Science		
		SEC01 Science-Math Electives courses		
		CAB91 CAL humanities Humanities Humanities		
		Humanities Humanities CSM01 CS mandtory		
		ECM01 Estra CS mandaroty		
		FRH01 Free humanities		

Figure 4: Add Course Group Page

 List of Courses: The combination of Prefix and Course number (i.e. CS105) should be entered in this box by leaving one space between them if this radio button is selected. Combination of Courses: The abbreviation of desired course groups should be entered in this text box by leaving one space between them to form a new course group including already existing course groups.



Figure 5: Course Group Page

- Requirements: Existing requirements and information related to them are shown in the table on this page. To edit/delete a requirement; click on the Edit or Delete button next to the name of each existing requirement. To add a new requirement, click on the Add New Requirement button on top of the list.
 - Add New Requirement: Enter the **Name** and the **Abbreviation** for each of the new requirement. Select the requirement type and select the required information for the selected requirement if needed (i.e. for "Simple"). It will be added to the list on the previous Requirements page. Click on the **Submit** button after the required information is entered. Requirement types are given below:
 - Simple (Select group)
 - Degree Specific Prerequisite
 - o Degree Specific Corequisite
 - Predefined Semester (Select semester)
 - Combination

Firefox *	🐯 Docs for 1	TRR - Dropito	н	× Þ Course List		
🕨 🕨 longbes	och usc.edu i	22825/require	ments/		☆ ♥ C 🛃 • Google	۹ (م
studen	t Sch	eduli	ng Sys	stem		
Course	Dire	ctor				
Logout	Requirem	ents	-			
lenu	Requi	rement	s:			
Courses	Add New	Requirem	mt			
	Edit 1	Delete		Requirement Humanities: 300 level and above		
equirements						
heree		Delete		Humanities: 100-200 level		
tograms		Delete		Technical Electives		
ocumentation		Delete		Software Electives		
		Delete		Required Computer Science		
		Delete		Required Math		
		Delete		Management - Mandatory		
		Delete		Science Math Electives		
		Delete		Humanitites - Special		
		Delete		CAL humanities		
		Delete		Free humanities		
		Delete		Science - Physics		
		Delete		Science - Chemistry		
		Delete		Chem & Bio		
		Delete		Chem Bio 2		
		Delete		Physics & Bio		
		Delete		Science sequences		
		Delete		complex demo requirement		
	Edit	Delete	DEM01	demo		

Figure 6: Requirements Page

- <u>Degree Programs:</u> The offered "degree programs" exist in the database are shown on this page. To edit/delete an existing degree program, click on the Edit or Delete button next to its related degree program. To add a new degree program, click on the Add New Degree Program link on top of the list.
 - Add New Degree Program: Enter the desired name of the new degree program in the related text box. After that, select the **Requirements** among the existing ones in database which are required to satisfy to graduate from this particular degree program. Click on the **Submit** button after the required information is entered.

Degree Program Lit x C C calhost9000/degree-programs/ tudent Scheduling System ourse Director set ma Pegree Programs pegree Programs Ad New Degree Program per Program Ad New Degree Program per Program per Program Ad New Degree Program per	<u>∎</u>
State Degree Programs asse Degree Programs: asse Group: Add New Degree Program parenerst: Edit Delet: bar of the State C.S. Fatering, 2011_Starting, With_CS115	
State Degree Programs asse Degree Programs: asse Group: Add New Degree Program parenerst: Edit Delet: bar of the State C.S. Fatering, 2011_Starting, With_CS115	
Continuit Degree Programs Instance Degree Programs: Instance Add New Degree Program Instance Edit Delete Instance Edit Delete Instance Edit Delete Instance Edit Delete	
Bargence Programms REEG. COMP	
Bargence Programms Bargence Programms: Bargence Progr	
add New Degree Program partnersts Edir Delete Name gents	
Edit Delete Name rec Edit Doleta CS_Entering_2011_Starting_With_CS115 grans Starting With_CS115	
Edit Delete Name rec Edit Doleta CS_Entering_2011_Starting_With_CS115 grans Starting With_CS115	
grans Edit Dolete CS_Entering_2011_Starting_With_CS115 grans	
umentation	

Figure 7: Degree Program Page

- <u>Documentation</u>: You can get the desired documentation/manual by clicking on the related links on this page.
 - <u>i.</u>e. User's Manual

🕒 localhost:9000/document: × 🧧 TODO List - Google Drive 🗴 📃							
← → C Dicalhost9000/documentation/							
Soon to be implemented - Static page for both Students and Course Directors							

Figure 8: Documentation Page

For the Student Users:

_

- On login screen; type given **User Name** and **Password** to login. Then click on the **Submit Query** button.
- You can return the homepage by clicking on **Home** button at any time.
- To login as a "student" will lead you to the "Degree Selection" screen. At the top, there are links representing recent path in breadcrumbs form to return back.

🕨 🕨 Student Deg	yree Selection X	×
$\leftarrow \ \Rightarrow \ C$	li localhost9000/login/	\$ ≡
Studen	nt Scheduling System	
Menu	Home > Degree Selection	11
Home	Degree Selection	
	Desired Degree CS_Entering_2011_Starting_With_CS115 • Year of Entry(De_QS_Entering_2011_Starting_With_CS115 (c. Lattered my degree in 2008) Desired Number of Semesters (c. L want to graduate in 8 more semesters) Submit	E

Figure 9: Degree Selection Page

- For Student Users:
 - Select **Desired Degree** from the list with the same title. (i.e. CS_Entering_2011_Starting_With_CS115)
 - Enter the **Year of Entry** as degree catalog year in the related input box. (as an integer form i.e. 2008)
 - Enter the **Desired Number of Semesters** in the given input box as an integer (i.e. 8). Note that; if there is no possible study plan for the given number of semester, the system will not be able to provide a plan naturally.
 - Click on the **Submit Query** button after entering the required information.



Figure 10: Degree Selection Page - Courses

- From screen shown above:
 - Select minimum and maximum number of courses desired to be taken in each semester.
 - And for each requirement (they are listed in order); select the courses taken before or wanted to be taken particularly, besides its location and term only as optional. It should be noted that too narrow specifications may lead to no study plan solution found by the system.
 - If no course is selected within these course groups shown or there is are missing course/courses, the system will assign course on its own.
 - Click on the **Submit Query** button.
- Study plan screen will be retrieved after the system builds a study plan.

🗲 🕨 Ion	igbeach.usc.edu 22825/student-degree-program-requirements/	☆ ▼ C 3 4 - Google 2 (14) ★ 1
Stude	ent Scheduling System	
Logout	Téone > Degree Selection > Smalty Plan Results Diere us ywar completed Study Plan.	
Menu	Courses in mill are courses you selected.	
Home	Fall 2012 184011 CH282 HPL339 HPL348 HPL340 Spring 2013 CH211 HPL345 HPL344 HPL442 Fall 2013 CH213 CH2147 HPL444 HPL445 Spring 2014 CH116 HPL451 HEIMA19 HEIMA15	
	Fall 2014 18HS311 HHS312 HHS313 HHS319 HPL111 Spring 2015 Fall 2015	
	Spring 2016 HLB116	

Figure 11: Study Plan Page

- Courses for constructed study plan are listed right section of their related semesters and year.
- Courses (prefix and number) in red color are the courses which the user particularly selected. Ones in white are the courses which the system selected in place of the user.
- If the system cannot construct a valid study plan, it will show "No Study Plan" page as a result. The reasons may be too few semesters, non-correlated courses (i.e. prerequisites) or too few maximum course numbers per semester. In other cases, desired classes might not be available on those selected semesters either.



Figure 12: No Study Plan Page

- In such cases (with too tight or too loose constraints), it may take up to one minute for system to construct a study plan or figure that there is no valid study plan with given constraints.
- To overcome these problems, relieve some of the constraints you gave to the system as inputs. That is sensible to start this relaxation process with the constraint with the least priority.

4. Troubleshooting

4.1 Frequently Asked questions

- Do I need to login either as a student or an administrator to interact with the system?
 - Yes, to build a study plan as a student, you need to login to the system with your personal credentials (input information); and to get access to the actions of administrative side like adding/deleting courses/requirements/degree programs, you need to login as an admin with individual credentials as well.
- For "Course Add" function; is there any preferred format to enter prerequisites and corequisites?
 - Yes; you need to enter these course related information with such format
 cprefixes><course number> with space/and/or between each course. For example:
 "CS105 and CS125"; "CS105 CS145"; CS105 or CS201" and so on.

Course Add	×						
← → C	localhost:900	00/courses/addC	oursePage/				☆ =
Studen	t Sched	luling Sy	vstem				
	Direct		stem				
Course	Directo	JI					
Logout							
Menu	Courses > Cour	rse Add					
~	Title:	Example Cours	e				
Courses		Full name of the	new course ie: Introduction to Alg	orithms			
Course Groups	Course Prefix:	PEP 💌					
		The prefix of the	e new course ie.: CS in CS115				E
Requirements	Course Numbe	r: 556					
Degree		The number of t	the new course ie.: 115 in CS115				
Programs	Credits:	4					
Deserventation		The number of t	the credits for the course.				
Documentation	Offered	When and when	te the course is offered every year.				
	OnCampus:	🗹 Fall	Winter	Spring	Summer I	Summer II	
	Online:	🗖 Fall	Winter	Spring	Summer I	Summer II	
	Prereqs:	PEP112 AND P	'EP111				
		The boolean eq	uation for the Prerequisites for the c	ourse ie.:CS115 AND(CS15	5 OR CS165) List of courses	under the form for reference.	
	Coreqs:	CH115					
		The boolean eq	uation for the Corequisites the same	e as Prerequisite.			
				Submit			
	Courses Refere	nce					
	Title		CourseNumber				
	Mechanics		PEP111				
	E&M		PEP112				
	Gen. Chem. I		CH115				

Figure 13: Course Add Page

- For "Edit Course" function; is there a preferred format to enter prerequisites and corequisites?
 - Yes, it requires the same format with the inputting of "Course Add" function as below:
 <prefixes><course number> with space/and/or between each course. For example:
 "CS105 and CS125"; "CS105 CS145"; CS105 or CS201" and so on.



Figure 14: Course Edit Page

4.2 Error Codes and Messages

• Login action: login = xxx, password: xxx: This error means that you entered wrong login information; the error might be either in username or password or both. The information you entered and the ones in database do not match.

▶ localhosts000/login/ × V I CDDD List - Geogle Drive ×	
← → C 🗋 localhost:9000/login/	☆ =
login action: login = wrong_user_name, password = wrong_password	

Figure 15: Missing Login Screen

 Blank Screen (soon to be filled with a valid error message screen) : This means that you did not enter sufficient information to commit the action you desire. For example; to add a new Degree Program, you must enter **Degree Program Name** and select at least one of the checkboxes which belong to existing requirements on database.



Figure 16: Degree Program Add Page

4.3 Note

- CCD : Core Capability Drive-through
- Courses to be added for initial tests are stated in the supporting document of "Demonstration Guideline".
- Administrators should get their authentication initials in person from system manager.