

**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

CZ2002 Assignment:



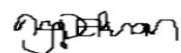
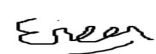

**My STudent Automated Registration System
(My STARS)**

Declaration of Original Work for CE/CZ2002 Assignment

We hereby declare that the attached group assignment has been researched, undertaken, completed and submitted as a collective effort by the group members listed below.

We have honored the principles of academic integrity and have upheld Student Code of Academic Conduct in the completion of this work.

We understand that if plagiarism is found in the assignment, then lower marks or no marks will be awarded for the assessed work. In addition, disciplinary actions may be taken.

Name	Course	Lab Group	Signature / Date
Chua Ming Hui	CZ2002	DSAI2	
Liu Wing Lam	CZ2002	DSAI2	
Ng Pek Han	CZ2002	DSAI2	
Ng Xiang Qing, Erica	CZ2002	DSAI2	
Yee An Qi	CZ2002	DSAI2	

Design Considerations

Login Page

LoginPage class uses several dependency relations such as PasswordReader and LoginInfoManager as seen in the class diagram below. This also promotes loose coupling.

Editing Student Access Period

This function is designed for the admin to edit the access period of current students and new students. There is a *boolean* edit parameter passed into editStudentAccessPeriod() method in the StudentAccessPeriodEditor class. This *boolean* parameter customises this edit operation -- (1) for a current student, program edits access period via prompting for student's username and; (2) for a new student, program edits access period via using username parameter passed into the editStudentAccessPeriod() method and adds student into the system. This design displays reusability as *boolean* edit can be incorporated with minimal disruption to the operation body code to reuse the main purpose of the edit operation.

```
public void editStudentAccessPeriod(boolean edit, String _username){
    Scanner sc = new Scanner (System.in);
    LoginInfoFileEditor accessPeriod = new LoginInfoFileEditor();
    String username = null;
    int accessPeriodLineNo;

    if (edit == true && _username == null) {
        System.out.println("You are editing the Student Access Period.");
    }

    do {
        if (edit == true && _username == null) {
            System.out.print("Enter the username of the student: ");
            username = sc.next();
        }

        else if (edit == false && _username != null){
            username = _username;
        }

        accessPeriodLineNo = accessPeriod.lineNo(username);
    } while (accessPeriodLineNo == -1);
}
```

Adding New Student

Applying Single Responsibility, we tried to create classes that are catered to one responsibility / functionality such as the PasswordModifier class which is used to change the student's password. We adapted encapsulation to store the details of the new student in the NewStudent class. To ensure information hiding and prevent the administrator from knowing the password of the new student, we randomly generate a password of length 8 consisting of all digits, lowercases and uppercase for the new student, and email them their personalised login information. Beyond that, we have an added function for students to change their password.

File Format

We applied abstraction to categorise the information used in storing data in text files while catering to limitations (e.g.: password can contain any character), we could not use a delimiter to store usernames and passwords. Thus, arranging the login information in separate lines as shown below is a better alternative.

```
m200033
fOWWQ
10/12/2020|09:00|12/12/2020|13:00
```

Notification Mode

We took into account various ways to notify the student such as email, whatsapp etc. using polymorphism and abstraction. With reference to the class diagram, we have an *abstract* NotificationMode class which includes an *abstract* sendNotification method. This class generalises EmailNotification. As shown in the screenshot below, we can create WhatsappNotification and SMSNotification classes by inheriting NotificationMode and implementing sendNotification in both subclasses.

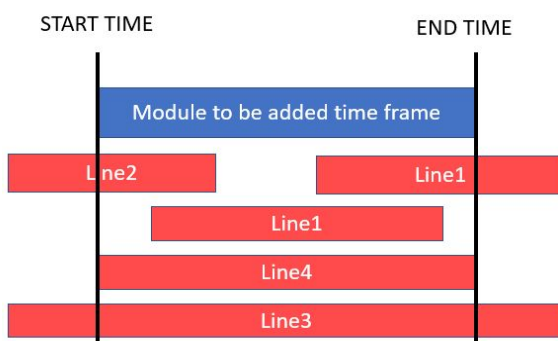
```
public class StudentNotification {
    public StudentNotification() {}

    public void notifyStudent(NotificationMode notiMode) {
        notiMode.sendNotification();
    }
}
```

Error Checking Class

Noticing that several inputs of similar types (dates, times) are required from the user to store into the text files, we used abstraction to simplify our codes by constructing different methods in the error checking class for different types of inputs.

Checking for Clash in Timetable



```
public boolean checkTimeClash(int i, int j, int[][] _time) {
    if (time[i][0] < _time[j][0] && _time[j][0] < time[i][1]) // cl Line 1
        return true;
    else if (time[i][0] < _time[j][1] && _time[j][1] < time[i][1]) Line 2
        return true;
    else if (time[i][0] > _time[j][0] && time[i][1] < _time[j][1]) Line 3
        return true;
    else if (time[i][0]==_time[j][0] && time[i][1]==_time[j][1]) / Line 4
        return true;
    else
        return false;
}
```

■ : Modules currently registered for student

Comparison performed after checking that the day coincides

Student: Adding and Dropping Courses

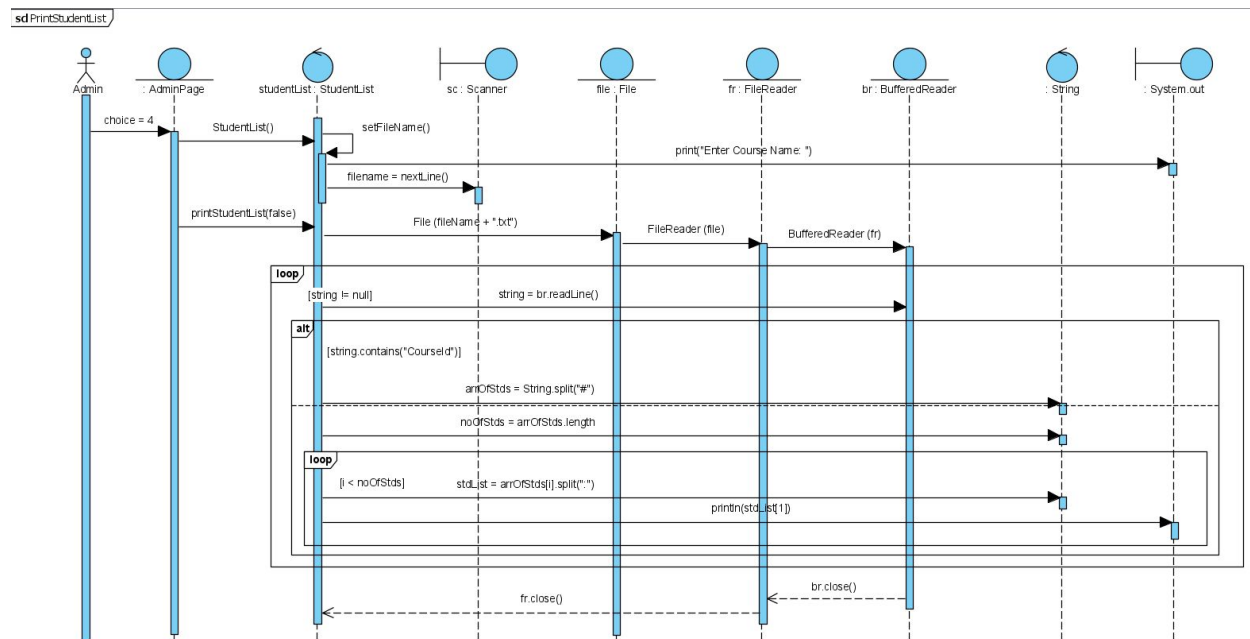


As seen from the diagrams above, both classes are the great-grandchildren of the `Course` class. To fulfill Single Responsibility Principle (SRP), inheritance is used to pass general attributes and methods from parent to child while more specific attributes and methods are created in the appropriate children classes. This design is extendable in a way that the `StudentNewCourse` and `StudentDropCourse` classes are used to derive the functionality of swapping indexes. The `StudentNewCourse` and `StudentDropCourse` classes can also be reused in different ways. For example, if the school decided to implement a bidding system, the classes can be reused with additional conditions added.

Handling Courses

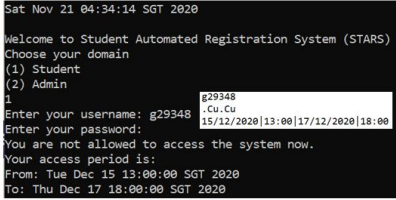
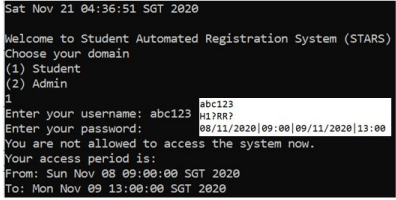
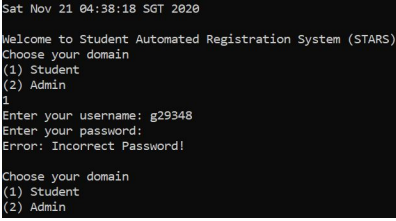
`ArrayList` is used to store all the information of each course, where information of each index number and information of each type of classes (e.g.: lecture, tutorial, lab) are stored in an `ArrayList`. The aim of using `ArrayList` is to store information dynamically as it is resizable. `ArrayList` also allows for random access of the information, allowing easier retrieval of information.

Detailed UML Sequence Diagram (“Print Student List By Course”)

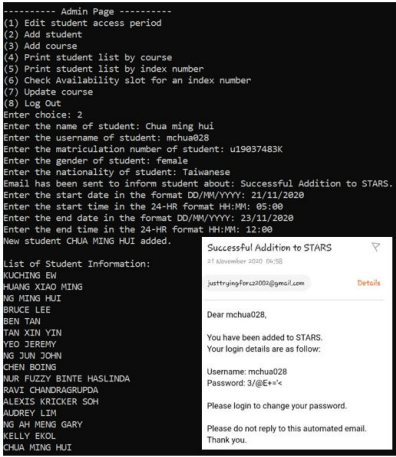


Test Cases for MySTARS Application

1. Student Login

	Test Case	Expected Outcome	Actual Outcome
1	Login before allowed period (dates)	Appropriate error message display	
2	Login after allowed period (dates)	Appropriate error message display	
3	Wrong password	Appropriate error message display. Prompted to re-enter login credentials.	

2. Add a student

	Test Case	Expected Outcome	Actual Outcome
1	Add a new student	A list of all students will be shown upon successful addition of student. An email will be sent to the student to inform them of their username and random password generated.	

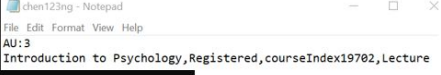
2	Add an existing student	Appropriate error message display	<pre> ----- Admin Page ----- (1) Edit student access period (2) Add student (3) Add course (4) Print student list by course (5) Print student list by index number (6) Check Availability slot for an index number (7) Update course (8) Log Out Enter choice: 2 Enter the name of student: chen boing Enter the username of student: chen123ng Enter the matriculation number of student: u3298132H Enter the gender of student: male Enter the nationality of student: Singaporean This username already exists! </pre>
3	Invalid data entries	Appropriate error message display	<pre> ----- Admin Page ----- (1) Edit student access period (2) Add student (3) Add course (4) Print student list by course (5) Print student list by index number (6) Check Availability slot for an index number (7) Update course (8) Log Out Enter choice: 2 Enter the name of student: hello123 Error: Name must not contain integer. Enter the name of student: Erica Ng Enter the username of student: abc,.de Error: Invalid Username. Enter the name of student: Erica Ng Enter the username of student: m190083 Enter the matriculation number of student: U1909203!/ Error: Invalid Matriculation Number. Enter the name of student: Erica Ng Enter the username of student: m190083 Enter the matriculation number of student: U1909203L Enter the gender of student: hello Error: Gender must be male or female. Enter the name of student: Erica Ng Enter the username of student: m190083 Enter the matriculation number of student: U1909203L Enter the gender of student: female Enter the nationality of student: sg50-/ Error: Nationality must not contain integer. </pre>

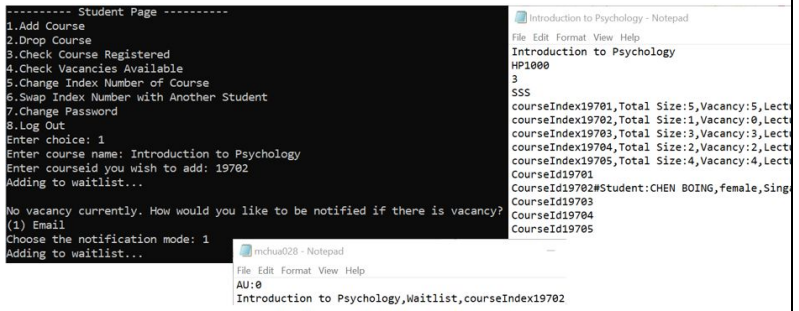
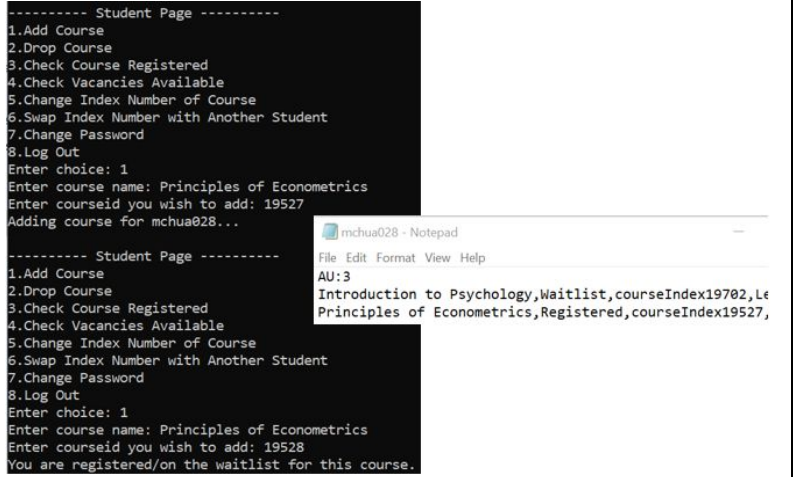
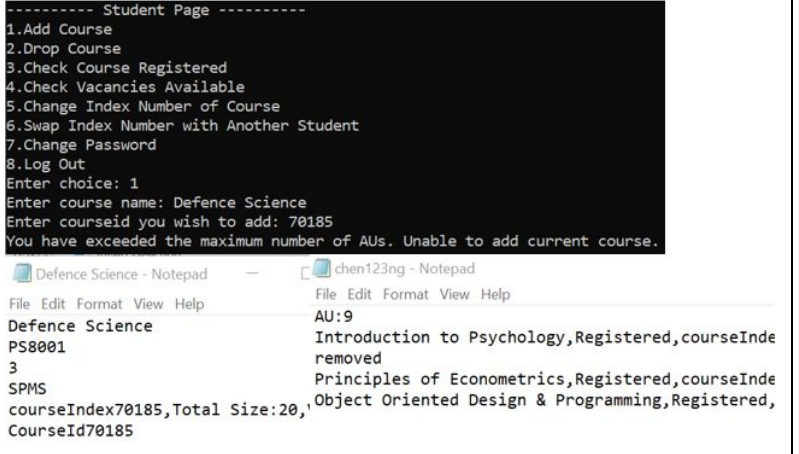
3. Add a course

	Test Case	Expected Outcome	Actual Outcome
1	Add a new course	A list of all courses will be shown upon successful addition of the course	<pre> Enter index number: 19705 Adding to course... Enter number of vacancy: 4 Enter lecture day: 1 Enter index lecture start timing: 1430 Enter index lecture end timing: 1630 Enter tutorial day: 4 Enter index tutorial start timing: 830 Enter index tutorial end timing: 930 Enter lab day: 0 Enter index lab start timing: 0 Enter index lab end timing: 0 Enter school of Course SSS Enter number of AUs for Course 3 Adding to course... List of all courses: Object Oriented Design & Programming Probability & Introduction to Statistics Defence Science Principles of Econometrics Introduction to Psychology </pre> <pre> Introduction to Psychology HP1000 3 SSS courseIndex19701,Total Size:5,Vacancy:5,Lecture Day:1,Lecture Start Time: courseIndex19702,Total Size:1,Vacancy:1,Lecture Day:1,Lecture Start Time: courseIndex19703,Total Size:3,Vacancy:3,Lecture Day:1,Lecture Start Time: courseIndex19704,Total Size:2,Vacancy:2,Lecture Day:1,Lecture Start Time: courseIndex19705,Total Size:4,Vacancy:4,Lecture Day:1,Lecture Start Time: CourseId19701 CourseId19702 CourseId19703 CourseId19704 CourseId19705 </pre>

2	Add an existing course	Appropriate error message display	<pre> ----- Admin Page ----- (1) Edit student access period (2) Add student (3) Add course (4) Print student list by course (5) Print student list by index number (6) Check Availability slot for an index number (7) Update course (8) Log Out Enter choice: 3 Enter Course Name: Defence Science Enter Course Code: PS8001 Enter the number of index number: 1 Course already exist! </pre> <p>Object Oriented Design & Programming Probability & Introduction to Statistics Defence Science Principles of Econometrics Introduction to Psychology</p>
3	Invalid data entries	Appropriate error message display	<pre> ----- Admin Page ----- (1) Edit student access period (2) Add student (3) Add course (4) Print student list by course (5) Print student list by index number (6) Check Availability slot for an index number (7) Update course (8) Log Out Enter choice: 3 Enter Course Name: Algorithm Enter Course Code: CZ2001 Enter the number of index number: a Please enter a positive integer Enter the number of index number: -1 Please enter a positive integer Enter the number of index number: 3 Enter index number: 10293 Adding to course... Enter number of vacancy: a Please enter a positive integer Enter number of vacancy: b Please enter a positive integer Enter number of vacancy: -7 Please enter a positive integer Enter number of vacancy: 4 Enter lecture day: 10 Please enter day from 1 to 7! Enter lecture day: rat Please enter day from 1 to 7! Enter lecture day: 4 Enter index lecture start timing: 1379 Please enter time in 24H format! Enter index lecture start timing: 1209 Enter index lecture end timing: 1349 </pre>

4. Register student for a course

	Test Case	Expected Outcome	Actual Outcome
1	Add a student to a course index with available vacancies	Successfully added, AU increases.	<pre> ----- Student Page ----- 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out Enter choice: 1 Enter course name: Introduction to Psychology Enter courseid you wish to add: 19702 Adding course for chen123ng... </pre> 

2	Add a student to a course index with 0 vacancies in tutorial or lab.	Appropriate error message display. Inform student they are placed on waitlist.	 <pre> ----- Student Page ----- 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out Enter choice: 1 Enter course name: Introduction to Psychology Enter courseid you wish to add: 19702 Adding to waitlist... No vacancy currently. How would you like to be notified if there is vacancy? (1) Email Choose the notification mode: 1 Adding to waitlist... mchua028 - Notepad File Edit Format View Help AU:0 Introduction to Psychology,Waitlist,courseIndex19702 </pre>
3	Register same course again	Appropriate error message display	 <pre> ----- Student Page ----- 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out Enter choice: 1 Enter course name: Principles of Econometrics Enter courseid you wish to add: 19527 Adding course for mchua028... ----- Student Page ----- 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out Enter choice: 1 Enter course name: Principles of Econometrics Enter courseid you wish to add: 19528 You are registered/on the waitlist for this course. mchua028 - Notepad File Edit Format View Help AU:3 Introduction to Psychology,Waitlist,courseIndex19702,Le Principles of Econometrics,Registered,courseIndex19527, </pre>
4	Register for new course which exceeds the maximum number of AU	Appropriate message display, unable to add course	 <pre> ----- Student Page ----- 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out Enter choice: 1 Enter course name: Defence Science Enter courseid you wish to add: 70185 You have exceeded the maximum number of AUs. Unable to add current course. Defence Science - Notepad File Edit Format View Help Defence Science PS8001 3 SPMS courseIndex70185,Total Size:20, CourseId70185 chen123ng - Notepad File Edit Format View Help AU:9 Introduction to Psychology,Registered,courseInde removed Principles of Econometrics,Registered,courseInde Object Oriented Design & Programming,Registered, </pre>

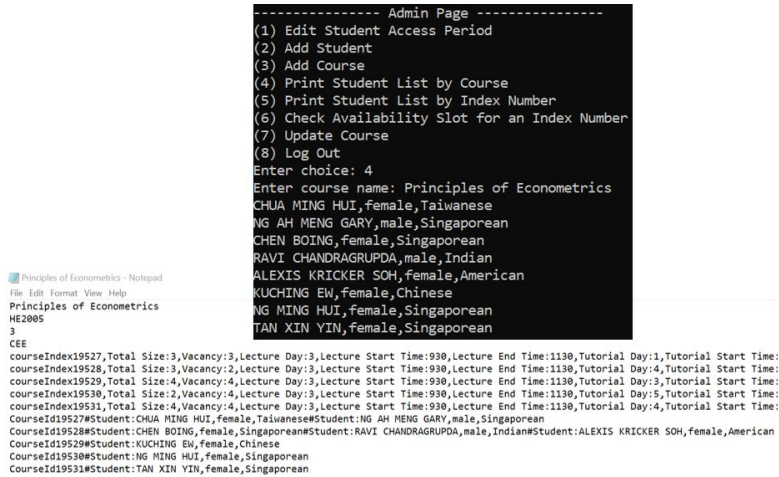
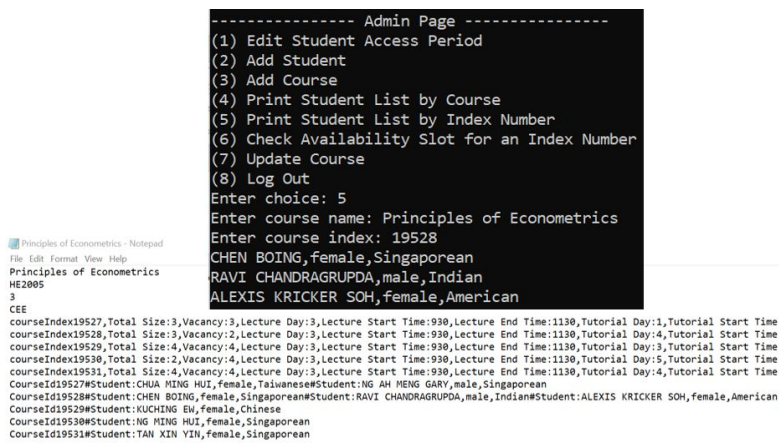
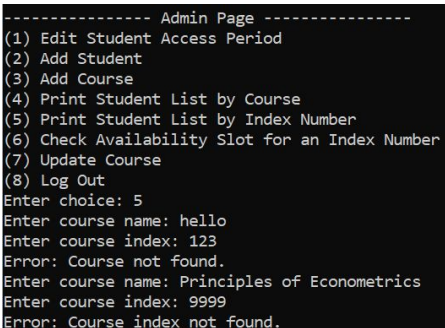
4	Invalid data entries	Appropriate error message display	<pre> ----- Student Page ----- 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out Enter choice: 1 Enter course name: hello Error: Invalid Course Name. </pre>
---	----------------------	-----------------------------------	--

5. Check vacancy in a class

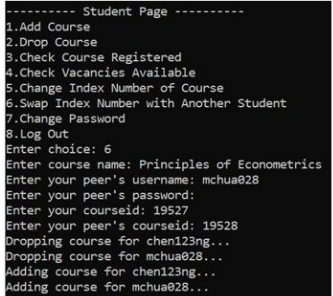
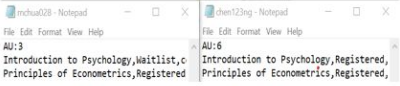
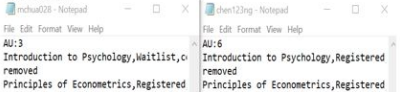
	Test Case	Expected Outcome	Actual Outcome
1	Check for vacancy in course index	Appropriate error message display	<pre> ----- Student Page ----- 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out Enter choice: 4 Enter course name: Principles of Econometrics Enter course index: 19528 Number of Vacancies in 19528: 3/3 ----- Admin Page ----- (1) Edit Student Access Period (2) Add Student (3) Add Course (4) Print Student List by Course (5) Print Student List by Index Number (6) Check Availability Slot for an Index Number (7) Update Course (8) Log Out Enter choice: 6 Enter course name: Principles of Econometrics Enter course index: 19528 Number of Vacancies in 19528: 3/3 </pre>

		waitlist.	
2	Drop a studentB from the same course index	StudentB successfully dropped and studentA successfully added. Notification will be sent to StudentA to inform them that they are added.	<div> <div> <p>----- Student Page -----</p> <ol style="list-style-type: none"> 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out <p>Enter choice: 2</p> <p>Enter course name: Introduction to Psychology</p> <p>Enter courseid you wish to drop: 19702</p> <p>Dropping course for chen123ng...</p> </div> <div> <p>Successfully added to course Introduction to Psychology</p> <p>justtryingforz2002@gmail.com</p> <p>Wed 11/25/2020 12:36 AM</p> <p>To: #CHUA MING HUI#</p> <p>Dear mchua028,</p> <p>You have been added to:</p> <p>Course: Introduction to Psychology</p> <p>Index Number: 19702</p> <p>Please do not reply to this automated email. Thank you.</p> <p>Please print. Thank you! Did you do this?</p> <p><input type="checkbox"/> Are the suggestions above helpful? Yes No</p> <p>Reply Forward</p> </div> </div> <div> <p>mchua028 - Notepad</p> <p>File Edit Format View Help</p> <p>AU:3</p> <p>Introduction to Psychology,Waitlist,courseIndex19702,Le</p> <p>Principles of Econometrics,Registered,courseIndex19527,</p> </div> <div> <p>mchua028 - Notepad</p> <p>File Edit Format View Help</p> <p>AU:6</p> <p>removed</p> <p>removed</p> <p>Principles of Econometrics,Registered,courseIndex19527,</p> <p>Introduction to Psychology,Registered,courseIndex19702,</p> </div>
3	Display StudentA timetable	Show their timetable and also the new course added	<div> <p>----- Student Page -----</p> <ol style="list-style-type: none"> 1.Add Course 2.Drop Course 3.Check Course Registered 4.Check Vacancies Available 5.Change Index Number of Course 6.Swap Index Number with Another Student 7.Change Password 8.Log Out <p>Enter choice: 3</p> <p>Principles of Econometrics,Registered,courseIndex19528,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:4,Tutorial Start Time:1230,Tutorial End Time:1430,Lab Day:0,Lab Start Time:0,Lab End Time:0,</p> <p>Object Oriented Design & Programming,Registered,courseIndex10200,Lecture Day:2,Lecture Start Time:930,Lecture End Time:1030,Tutorial Day:2,Tutorial Start Time:1030,Tutorial End Time:1130,Lab Day:6,Lab Start Time:830,Lab End Time:1030,</p> <p>Introduction to Psychology,Registered,courseIndex19702,Lecture Day:1,Lecture Start Time:1430,Lecture End Time:1630,Tutorial Day:5,Tutorial Start Time:1130,Tutorial End Time:1230,Lab Day:0,Lab Start Time:0,Lab End Time:0,</p> </div> <div> <p>chen123ng - Notepad</p> <p>File Edit Format View Help</p> <p>AU:9</p> <p>removed</p> <p>removed</p> <p>Principles of Econometrics,Registered,courseIndex19528,Lecture Da</p> <p>Object Oriented Design & Programming,Registered,courseIndex10200,</p> <p>removed</p> <p>Introduction to Psychology,Registered,courseIndex19702,Lecture Da</p> </div>

8. Print student list by index number, course

	Test Case	Expected Outcome	Actual Outcome
1	Print list by course	Student list by course is displayed	 <pre> ----- Admin Page ----- (1) Edit Student Access Period (2) Add Student (3) Add Course (4) Print Student List by Course (5) Print Student List by Index Number (6) Check Availability Slot for an Index Number (7) Update Course (8) Log Out Enter choice: 4 Enter course name: Principles of Econometrics CHUA MING HUI,female,Taiwanese NG AH MENG GARY,male,Singaporean CHEN BOING,female,Singaporean RAVI CHANDRAGRUPDA,male,Indian ALEXIS KRICKER SOH,female,American KUCHING EW,female,Chinese NG MING HUI,female,Singaporean TAN XIN YIN,female,Singaporean Principles of Econometrics - Notepad File Edit Format View Help Principles of Econometrics HE2005 3 CEE courseIndex19527,Total Size:3,Vacancy:3,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:1,Tutorial Start Time: courseIndex19528,Total Size:3,Vacancy:2,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:4,Tutorial Start Time: courseIndex19529,Total Size:4,Vacancy:4,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:3,Tutorial Start Time: courseIndex19530,Total Size:2,Vacancy:4,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:5,Tutorial Start Time: courseIndex19531,Total Size:4,Vacancy:4,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:4,Tutorial Start Time: CourseId19527#Student:CHUA MING HUI,female,Taiwanese#Student:NG AH MENG GARY,male,Singaporean CourseId19528#Student:CHEN BOING,female,Singaporean#Student:RAVI CHANDRAGRUPDA,male,Indian#Student:ALEXIS KRICKER SOH,female,American CourseId19529#Student:KUCHING EW,female,Chinese CourseId19530#Student:NG MING HUI,female,Singaporean CourseId19531#Student:TAN XIN YIN,female,Singaporean </pre>
2	Print list by Index Number	Student list by index number is displayed	 <pre> ----- Admin Page ----- (1) Edit Student Access Period (2) Add Student (3) Add Course (4) Print Student List by Course (5) Print Student List by Index Number (6) Check Availability Slot for an Index Number (7) Update Course (8) Log Out Enter choice: 5 Enter course name: Principles of Econometrics Enter course index: 19528 CHEN BOING,female,Singaporean RAVI CHANDRAGRUPDA,male,Indian ALEXIS KRICKER SOH,female,American Principles of Econometrics - Notepad File Edit Format View Help Principles of Econometrics HE2005 3 CEE courseIndex19527,Total Size:3,Vacancy:3,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:1,Tutorial Start Time: courseIndex19528,Total Size:3,Vacancy:2,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:4,Tutorial Start Time: courseIndex19529,Total Size:4,Vacancy:4,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:3,Tutorial Start Time: courseIndex19530,Total Size:2,Vacancy:4,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:5,Tutorial Start Time: courseIndex19531,Total Size:4,Vacancy:4,Lecture Day:3,Lecture Start Time:930,Lecture End Time:1130,Tutorial Day:4,Tutorial Start Time: CourseId19527#Student:CHUA MING HUI,female,Taiwanese#Student:NG AH MENG GARY,male,Singaporean CourseId19528#Student:CHEN BOING,female,Singaporean#Student:RAVI CHANDRAGRUPDA,male,Indian#Student:ALEXIS KRICKER SOH,female,American CourseId19529#Student:KUCHING EW,female,Chinese CourseId19530#Student:NG MING HUI,female,Singaporean CourseId19531#Student:TAN XIN YIN,female,Singaporean </pre>
3	Invalid data entries	Appropriate error message display	 <pre> ----- Admin Page ----- (1) Edit Student Access Period (2) Add Student (3) Add Course (4) Print Student List by Course (5) Print Student List by Index Number (6) Check Availability Slot for an Index Number (7) Update Course (8) Log Out Enter choice: 5 Enter course name: hello Enter course index: 123 Error: Course not found. Enter course name: Principles of Econometrics Enter course index: 9999 Error: Course index not found. </pre>

9. Swap course index with another student

	Test Case	Expected Outcome	Actual Outcome
1	Swap index with another student from another index	Appropriate message display, swapped.	 <p>----- Student Page ----- 1. Add Course 2. Drop Course 3. Check Course Registered 4. Check Vacancies Available 5. Change Index Number of Course 6. Swap Index Number with Another Student 7. Change Password 8. Log Out Enter choice: 6 Enter course name: Principles of Econometrics Enter your peer's username: mchua028 Enter your peer's password: Enter your courseid: 19527 Enter your peer's courseid: 19528 Dropping course for chen123ng... Dropping course for mchua028... Adding course for chen123ng... Adding course for mchua028...</p> <p>Before</p>  <p>After</p> 

Youtube link to our video:

<https://youtu.be/BJYpQbWpIbk>