6/04/2013

**Sprint 2 - Version 2.0**

**IUNotifier Application**

**Project Note**

Contents

[**I.** **Release Note** 3](#_Toc353606529)

[1. Getting-Course-Information use case 3](#_Toc353606530)

[2. Creating-course-announcement use case 4](#_Toc353606531)

[**II.** **Development Note** 5](#_Toc353606532)

[1. Version Control 5](#_Toc353606533)

[**1.1.** **Log messages** 5](#_Toc353606534)

[**1.2.** **Difference between two files** 5](#_Toc353606535)

[2. Sample Java source with refactoring 6](#_Toc353606536)

[**2.1.** **Replace magic number with symbolic constant** 6](#_Toc353606537)

[**2.2.** **Decompose conditional** 6](#_Toc353606538)

[**2.3.** **Extract method** 7](#_Toc353606539)

[**2.4.** **Split temporary variable** 8](#_Toc353606540)

[**III.** **Test Note** 8](#_Toc353606541)

[1. Test Case 8](#_Toc353606542)

[**1.1.** **Test cases for Login use case** 8](#_Toc353606543)

[**1.2.** **Test case for Logout use case** 10](#_Toc353606544)

[**1.3.** **Test cases for Reading-news use case** 10](#_Toc353606545)

[**1.4.** **Test cases for Getting-events-information use case** 12](#_Toc353606546)

[**1.5.** **Test cases for Getting-course-information use case** 13](#_Toc353606547)

[2. JUnit Test Java Source 14](#_Toc353606548)

# **Release Note**

## Getting-Course-Information use case

|  |  |
| --- | --- |
| Name | UC-6: Get-course-information |
| Summary | Allow user to get information about all the courses of IU |
| Rationale | The International University has many departments and each of them has many courses correspondingly. Many people who care about the courses that the IU offers such as student’s parents, IU’s guesses and especially IU’s students. Currently, the IU offers 2 ways to get the course information: online and offline. The online way is to use the Edusoft website and the offline way is to use the IU handbook for students. The “Get-course-information” function of the IUNotifer app consist of these 2 ways of getting the course information. That means the user can use the app with internet connection to access the latest course information and when there is no internet connection, the user can still use the app however the information is not the latest one. |
| Users | All users |
| Preconditions | * The user installed the IUNotifier application. |
| Basic course of events | 1. The user indicates that the application is to provide information about a specific course of the IU. 2. The application responses by requesting the user to tap the “Courses” icon in the main menu of the application. 3. The user taps the “Courses” icon and indicates that he/she will see a list of departments of the IU. 4. The application responses by displaying the department list and requests the user to select a department. 5. The user selects a department and indicates to see all the courses of the selected department. 6. The application responses by display the course list of the selected department and request the user to select a course. 7. The user selects a course and indicate that he/she will see the information of the selected course. 8. The application responses by display the information of the selected course |
| Alternative paths | 1. The user may decide to return to previous screen at any time during Steps 2, 3, 4, 5, 6, 7, or 8. 2. In step 4, the user could be able to get the currently studying course list by tapping the “My Courses” button. The application then request the user to login using his/her account and then display the list of studying courses. |
| Postconditions | The user could be able to get the information of a specific course. |

## Creating-course-announcement use case

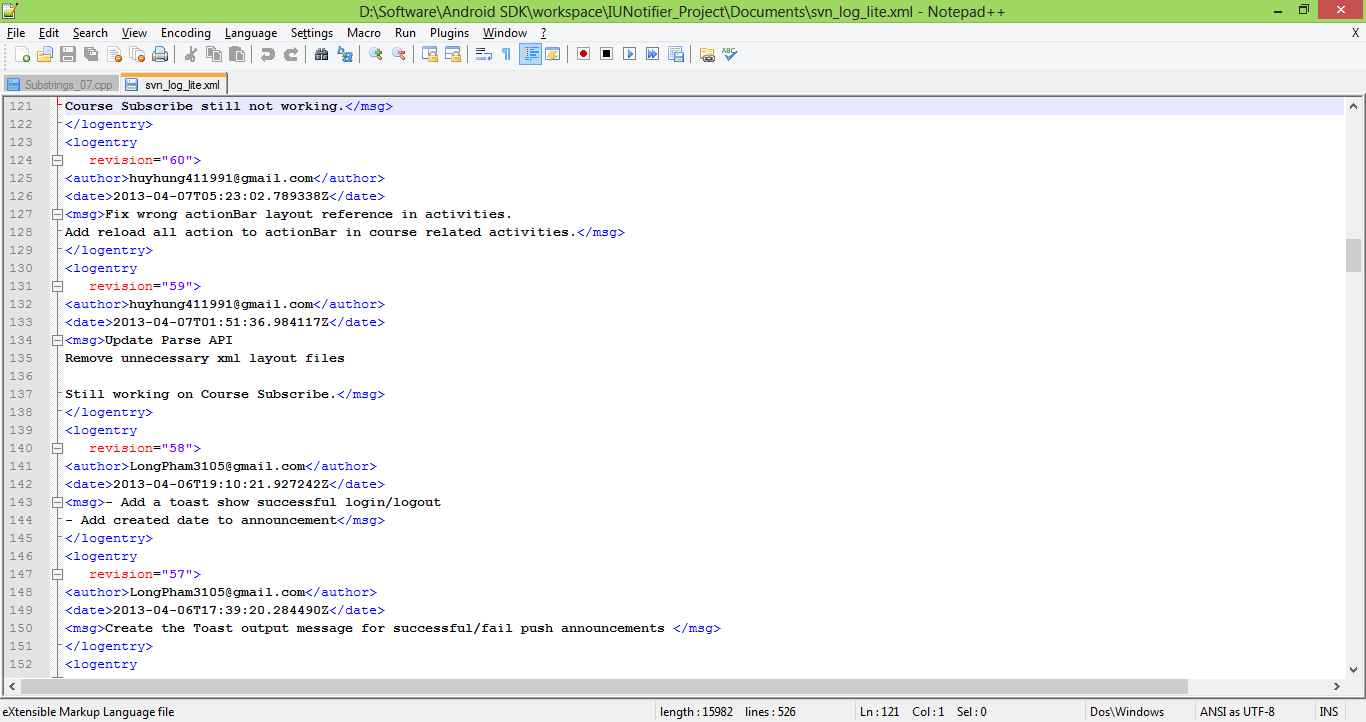
|  |  |
| --- | --- |
| Name | UC-7: Create-course-announcement |
| Summary | Allow the department staff or professor to create a quick announcement about a specific course to students. |
| Rationale | Occasionally, there is a sudden change in the schedule or location of a course. In that case, the department staff or professor, who is responsible to announce this change to students, often stick an announcement paper on the notice board of the department. This method of announcement is very inefficiency because the student cannot know about the change immediately and this may make them feel uncomfortable. By using the “Create a course announcement” function of the IUNotifer application, the department staffs or professors can make an announcement that will be immediately received by the students. |
| Users | * Department staff * Professor |
| Preconditions | * The device must have internet connection. * The user installed the IUNotifier application. |
| Basic course of events | 1. The user indicates that the application is to provide a create-course-announcement function. 2. The application requests the user to choose a specific course to make an announcement by navigating to the course information screen as guided in the “Get-course-information” use case. 3. The user navigates to the course information screen. 4. The application requests the user to login using his/her account. 5. The user logged in by using his/her account. 6. The application responses by displaying the “Create Announcement” icon and request the user to tap the icon. 7. The user taps the “Create Announcement” icon. 8. The application responses by requesting the user to provide the content of the announcement and tap the “Announce” button. 9. The user provides the content of the announcement and taps the “Announce” button. 10. The application responses by displaying the successfully message. |
| Alternative paths | 1. In step 2, the user can login in at any screen during the navigation to the course information screen and then he/she can skip steps 4, 5 and 6. 2. The user may decide to return to previous screen at any time during Steps 4, 5, 6, 7, or 8. |
| Postconditions | The user created an announcement about a specific course. |

# **Development Note**

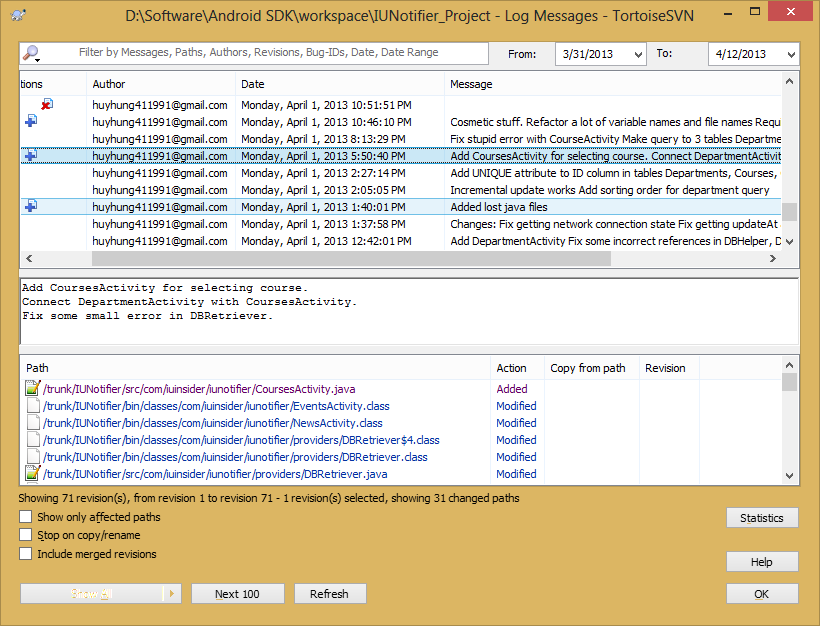
## Version Control

### **Log messages**

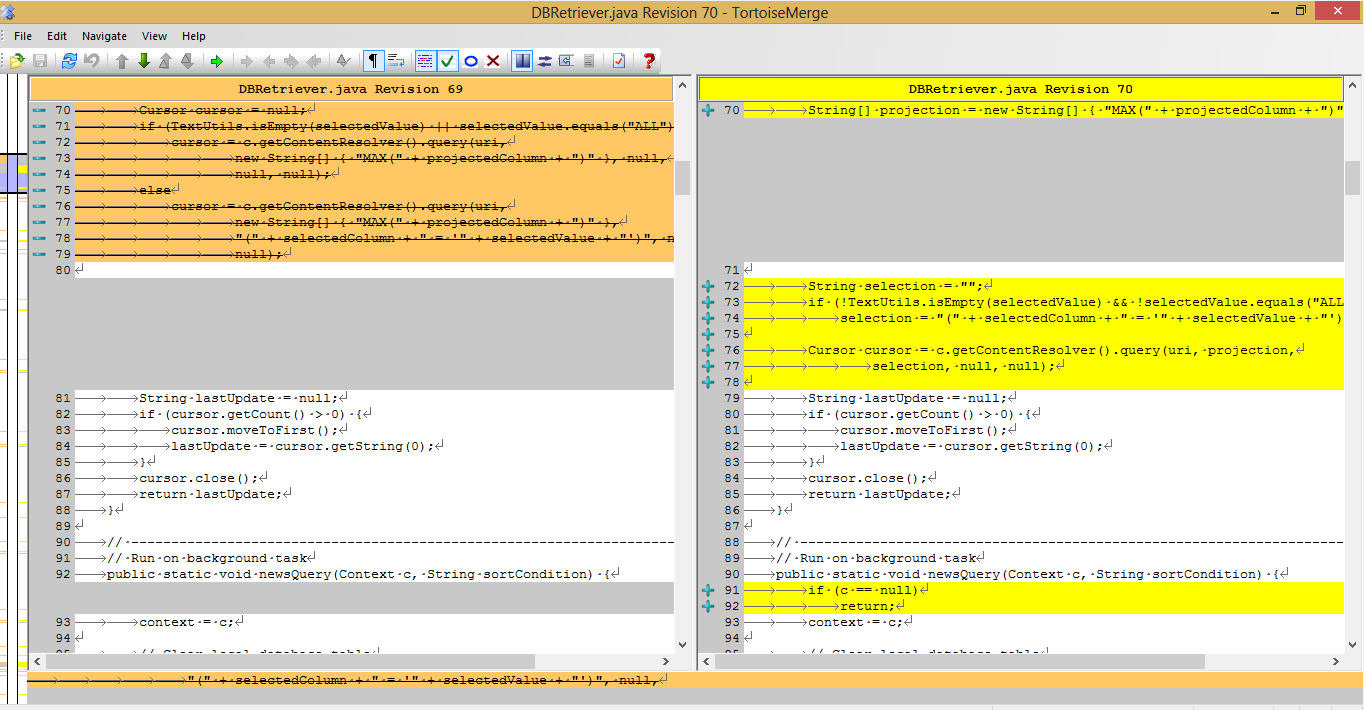
Using “svn log --xml --verbose > log.xml”



Using TortoiseSVN:



### **Difference between two files**



## Sample Java source with refactoring

### **Replace magic number with symbolic constant**

You have a literal number with a particular meaning. Create a constant, name it after the meaning, and replace the number with it:

* Before:



* After:



### **Decompose conditional**

You have a complicated conditional (if-then-else) statement. Extract methods from the condition, then part, and else parts:

* Before:



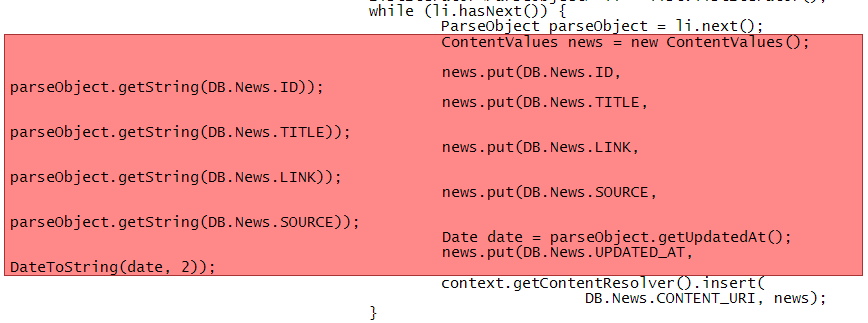
* After:



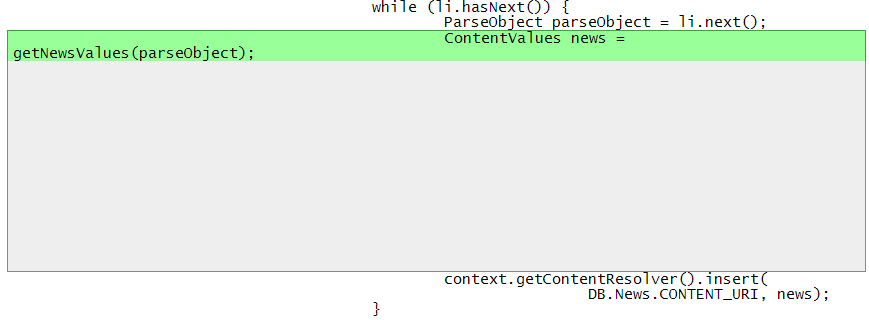
### **Extract method**

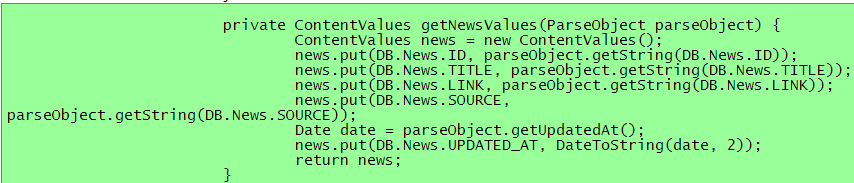
You have a code fragment that can be grouped together. Turn the fragment into a method whose name explains the purpose of the method:

* Before:



* After:

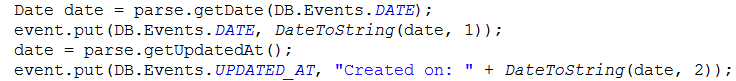




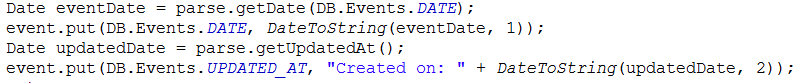
### **Split temporary variable**

You have a temporary variable assigned to more than once, but is not a loop variable nor a collecting temporary variable. Make a separate temporary variable for each assignment:

* Before:



* After:



# **Test Note**

## Test Case

### **Test cases for Login use case**

|  |  |
| --- | --- |
| **Name** | **TC-01: Verify that the user can login with a correct username and a correct password** |
| Requirement | FR-01: Login |
| Preconditions | The username *itiu09008* and password *12345* must be available in the database. |
| Steps | 1. Tap on the “Login” icon. 2. Tap on the “Username” field. 3. Enter *itiu09008.* 4. Tap on the “Password” field. 5. Enter *12345.* 6. Tap the “Sign in” button. |
| Expected Results | 1. The login form dismisses. 2. The “Login successfully” message displays. 3. The “Login successfully” message dismisses. 4. The color of the login icon changes to blue. |

|  |  |
| --- | --- |
| **Name** | **TC-02: Verify that the user cannot login with a wrong username and a correct password** |
| Requirement | FR-01: Login |
| Preconditions | * The username *abcd* is not in the database. * The password 12345 must be available in the database. |
| Steps | 1. Tap on the “Login” icon. 2. Tap on the “Username” field. 3. Enter *abcd.* 4. Tap on the “Password” field. 5. Enter *12345.* 6. Tap the “Sign in” button. |
| Expected Results | * 1. The login form dismisses.   2. The login form displays with the *abcd* value in the username field and there is five dots appeared in the password field.   3. There is an error message “*This username is incorrect*” appeared in the username field. |

|  |  |
| --- | --- |
| **Name** | **TC-03: Verify that the user cannot login with a correct username and a wrong password** |
| Requirement | FR-01: Login |
| Preconditions | * The username *itiu09008* is available in the database. * The password aaaa is not in the database. |
| Steps | 1. Tap on the “Login” icon. 2. Tap on the “Username” field. 3. Enter *itiu09008.* 4. Tap on the “Password” field. 5. Enter *aaaa.* 6. Tap the “Sign in” button. |
| Expected Results | * 1. The login form dismisses.   2. The login form displays with the *itiu09008* value in the username field and there is four dots appeared in the password field.   3. There is an error message “*This password is incorrect*” appeared in the password field. |

|  |  |
| --- | --- |
| **Name** | **TC-04: Verify that the user cannot login with a wrong username and a wrong password** |
| Requirement | FR-01: Login |
| Preconditions | * The username *abcd* is not in the database. * The password aaaa is not in the database. |
| Steps | 1. Tap on the “Login” icon. 2. Tap on the “Username” field. 3. Enter *abcd.* 4. Tap on the “Password” field. 5. Enter *aaaa.* 6. Tap the “Sign in” button. |
| Expected Results | 1. The login form dismisses. 2. The login form displays with the *abcd* value in the username field and there is four dots appeared in the password field. 3. There is an error message “*This password is incorrect*” appeared in the password field. |

### **Test case for Logout use case**

|  |  |
| --- | --- |
| **Name** | **TC-05: Verify that the user can logout off their current logging account** |
| Requirement | FR-02: Logout |
| Preconditions | The user has logged in with the username *itiu09008* and password *12345.* |
| Steps | * 1. Tap on the “Logout” icon.   2. Tap on the “Yes” button. |
| Expected Results | * 1. The logout-verification windows dismisses.   2. The “Logout successfully” message displays.   3. The “Logout successfully” message dismisses.   4. The color of the login icon changes to white. |

### **Test cases for Reading-news use case**

|  |  |
| --- | --- |
| **Name** | **TC-06: Verify that the user can get a list of news from the HCMIU news source** |
| Requirement | FR-03: Get-news-from-source |
| Preconditions | * The main menu page is loaded. * There are two news from the HCMIU news source in the database. |
| Steps | Tap on the “News” icon in the main menu. |
| Expected Results | 1. The main menu page dismisses. 2. The “News” page displays. 3. The “News From” drop-down menu displays. 4. The default value of the “News From” drop-down menu is *HCMIU*. 5. The news list displays and it contains two news. |

|  |  |
| --- | --- |
| **Name** | **TC-07: Verify that the user can get a list of news from the Edusoft news source** |
| Requirement | FR-03: Get-news-from-source |
| Preconditions | * The news list of the HCMIU news source is loaded in the News page. * There are three news from the Edusoft news source in the database. |
| Steps | 1. Tap on the “News From” drop-down menu. 2. Tap on the *Edusoft* value. |
| Expected Results | 1. The news list of the HCMIU news source dismisses. 2. The news list of the Edusoft news source displays and it contains three news. |

|  |  |
| --- | --- |
| **Name** | **TC-08: Verify that the user can get a list of news from the IU Community news source** |
| Requirement | FR-03: Get-news-from-source |
| Preconditions | * The news list of the HCMIU news source is loaded in the News page. * There are four news from the IU Community news source in the database. |
| Steps | 1. Tap on the “News From” drop-down menu. 2. Tap on the *IU Community* value. |
| Expected Results | 1. The news list of the HCMIU news source dismisses. 2. The news list of the IU Community news source displays and it contains four news. |

|  |  |
| --- | --- |
| **Name** | **TC-09: Verify that the user can read the content of a specific news** |
| Requirement | FR-04: Get-news-content |
| Preconditions | The news list of the HCMIU news source is loaded. |
| Steps | Tap on the “Hanh Trinh IU In Singapore” news. |
| Expected Results | 1. The “News” page dismisses. 2. The content of the “Hanh Trinh IU Singapore” news displays. |

### **Test cases for Getting-events-information use case**

|  |  |
| --- | --- |
| **Name** | **TC-10: Verify that the user can get a list of upcoming events** |
| Requirement | FR-05: Get-event-list |
| Preconditions | * The main menu page is loaded. * The database contains an event named “BA Design Contest” which is held on 17/04/2013. * The current date and time on the phone are 12/04/2013 and 2:13 PM. |
| Steps | Tap on the “Events” icon in the main menu. |
| Expected Results | 1. The main menu page dismisses. 2. The ”Events” page displays. 3. The “Sort By” drop-down menu displays. 4. The default value of the “Sort By” drop-down menu is *Upcoming.* 5. The event list displays and it contains the “BA Design Contest” event. |

|  |  |
| --- | --- |
| **Name** | **TC-11: Verify that the user can get a list of all events** |
| Requirement | FR-05: Get-event-list |
| Preconditions | * The upcoming event list is loaded. * There are total nine events in the database. |
| Steps | 1. Tap on the “Sort By” drop-down menu. 2. Tap on the *All* value. |
| Expected Results | 1. The current event list dismisses. 2. A list of all events displays and it contains nine events. |

|  |  |
| --- | --- |
| **Name** | **TC-12: Verify that the user can get a list of today events** |
| Requirement | FR-05: Get-event-list |
| Preconditions | * The upcoming event list is loaded. * The current date and time on the phone are 12/04/2013 and 2:21 PM. |
| Steps | 1. Tap on the “Sort By” drop-down menu. 2. Tap on the “*Today*” value. |
| Expected Results | 1. The current event list dismisses. 2. The progress bar displays. |

|  |  |
| --- | --- |
| **Name** | **TC-13: Verify that the user can get the detailed information of an event** |
| Requirement | FR-06: Get-event-detail |
| Preconditions | * The upcoming event list is loaded. * The “BA Design Contest” event information must be available in the database. |
| Steps | Tap on the “BA Design Contest” event. |
| Expected Results | 1. The detailed information windows displays. 2. The “Description” field content is “Give your ideas to design the uniform and BA logo”. 3. The “Date” field content is “Wednesday, 17-04-2013”. 4. The “Time” field content is “08:00”. 5. The “Location” field does not have anything. |

### **Test cases for Getting-course-information use case**

|  |  |
| --- | --- |
| **Name** | **TC-14: Verify that the user can get a list of departments in the IU** |
| Requirement | FR-07: Get-department-list |
| Preconditions | * The information of ten departments of the IU must be available in the database. * The main menu page is loaded. |
| Steps | Tap on the “Courses” icon in the main menu. |
| Expected Results | 1. The main menu page dismisses. 2. The “Departments” page displays and contains name of the ten departments in the department list. |

|  |  |
| --- | --- |
| **Name** | **TC-15: Verify that the user can get a list of courses of a department** |
| Requirement | FR-08: Get-course-list |
| Preconditions | * The information of three courses “Software Project Management”, “Operating System” and “Computer Architecture” must be available in the database. * The department list page is loaded. |
| Steps | Tap on the “School of Computer Science and Engineering” department. |
| Expected Results | 1. The “Departments” page dismisses. 2. The “Courses” page displays and contains a list of three courses: “Computer Architecture”, “Operating System”, and “Software Project Management”. |

|  |  |
| --- | --- |
| **Name** | **TC-16: Verify that the user can get detailed information of a course** |
| Requirement | FR-09: Get-course-detail |
| Preconditions | * The “Courses” page is loaded. * The detailed information of “Software Project Management” course must be available in the database. * The user has not logged in. |
| Steps | Tap on the “Software Project Management” course. |
| Expected Results | 1. The “Courses” page dismisses. 2. The “Course Details” page displays. 3. The “Announcement” icon displays. 4. The “Course Details” table contains the following information:  * Course ID: IT056IU * Lecturer: Professor Hoang Phan * Credits: 4 * Theory: 6(1234) * Lab: 7(5678) * Prerequisite: None |

|  |  |
| --- | --- |
| **Name** | **TC-17: Verify that the user can get a list of studying courses** |
| Requirement | FR-10: Get-studying-courses |
| Preconditions | * The “Departments” page is loaded. * The user has logged in with username *itiu09008* and password *12345.* * The studying course of the user with username *itiu09008* in the database must be “Software Project Management”. |
| Steps | Tap on the “My Courses” button. |
| Expected Results | 1. The “Departments” page dismisses. 2. The “Courses” page displays and the course list contains a “Software Project Management” course. |

## JUnit Test Java Source

