

---

**Computer Science 2212**  
**Assignment #3**  
**Due: March 14, 2014**

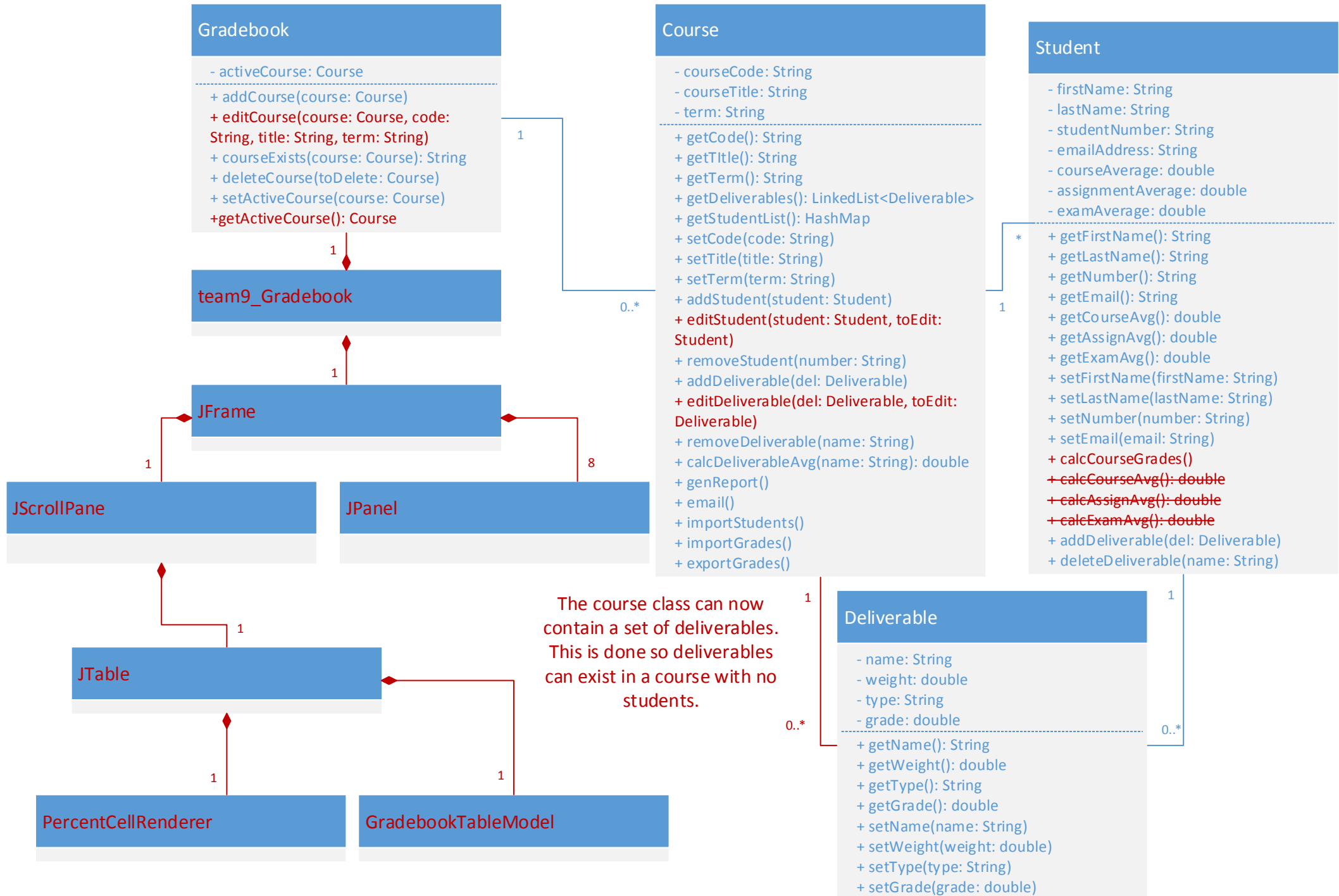
---

**Team 9**

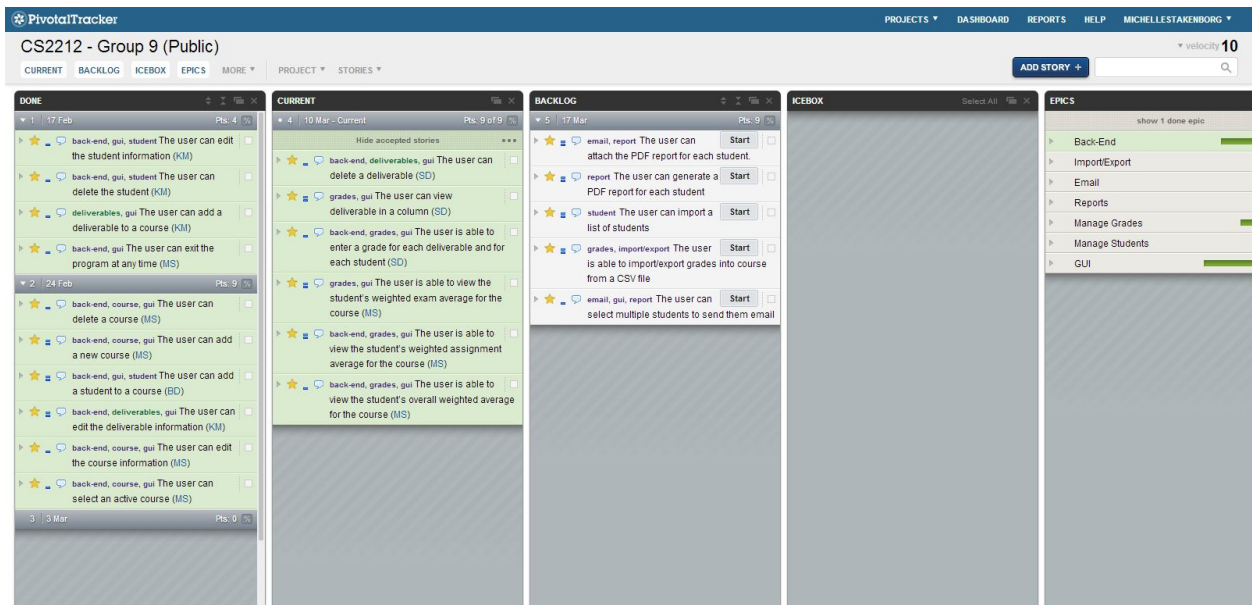
Sohrab Dolatabadi  
Bruno Felipe da Silva Correa  
Tyler Laporte  
Killian McGuinness  
Michelle Stakenborg

## 2 - REPORT

### 2.1 Revised UML Class Diagram



## 2.2 Revised User Stories



Link To Pivotal Tracker: <https://www.pivotaltracker.com/s/projects/1002276>

Completed User Stories (user stories completed since last assignment are bolded):

- The user can add a new course.
- The user can add a student to a course.
- The user can add a deliverable to a course.
- The user can exit the program at any time.
- The user can edit the course information.
- The user can select an active course.
- **The user can edit the student information.**
- **The user can delete the student.**
- **The user can delete a course.**
- **The user can edit the deliverable information.**
- **The user can delete a deliverable.**
- **The user can view deliverable in a column.**
- **The user is able to enter a grade for each deliverable and for each student.**
- **The user is able to view the student's weighted exam average for the course.**
- **The user is able to view the student's weighted assignment average for the course.**
- **The user is able to view the student's overall weighted average for the course.**

Instructions To test (included user stories completed for Assignment 2 because the GUI has been changed):

1. Compile: mvn package
2. Run program:  
java -jar target/team9-gradebook-1.0-SNAPSHOT-jar-with-dependencies.jar
3. Test "The user can add a new course" user story:
  - Click Add Course → Enter Information → Click Add
4. Test "The user can select an active course" user story:
  - Choose a course in "Set Active Course" combo box
5. Test "The user can edit the course information" user story:
  - Click Edit Course → Change Information → Click Edit
6. Test "The user can add a student to a course" user story:
  - Click Add Student → Enter Information → Click Add
7. Test "The user can add a deliverable to a course" user story:
  - Click Add Deliverable → Enter Information → Click Add
8. Test "The user can exit the program at any time" user story:
  - Click the close button (top right)
9. **Test "The user can edit the student information" user story:**
  - **In the table, double click an existing student's name → Edit Information → Click Edit**
10. **Test "The user can delete the student" user story:**
  - **In the table, select one or more students name → Press the "Delete" key → Click Yes**
  - Or**
  - **When editing a student, click Remove student → Click Yes**
11. **Test "The user can delete a course" user story:**
  - **Click Delete Course → Click Yes**
12. **Test "The user can edit the deliverable information" user story:**
  - **In the table, double click a deliverable header → Edit Information → Click Edit**
13. **Test "The user can delete a deliverable" user story:**
  - **When editing a deliverable, click Delete Deliverable → Click Yes**
14. **Test "The user can view deliverable in a column" user story:**
  - **View table**
15. **Test "The user is able to enter a grade for each deliverable and for each student" user story:**
  - **Double click the cell containing the grade you wish the enter (or press the "Enter" key) → Enter new grade → Press the "Enter" key**
16. **Test "The user is able to view the student's weighted exam average for the course" user story:**
  - **View table (assuming there is at least one marked exam deliverable)**
17. **Test "The user is able to view the student's weighted assignment average for the course" user story:**

- View table (assuming there is at least one marked assignment deliverable)

18. Test “The user is able to view the student's overall weighted average for the course” user story:

- View table (assuming there is at least one marked deliverable)

## 2.3 Acceptance Tests

### Adding a New Course - Acceptance Tests

<b>ID</b>	1
<b>Testing</b>	Admin User adding a new Course - Invalid Course Information
<b>References</b>	Requirement 4.2.1
<b>Test Case</b>	1. Click on Add Course 2. Leave the Course Code field blank 3. Fill out remaining information with title and term
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Please enter every course attribute

<b>ID</b>	2
<b>Testing</b>	Admin User adding a new Course - Invalid Course Information
<b>References</b>	Requirement 4.2.1
<b>Test Case</b>	1. Fill in code with a valid and unique course code 2. Leave the course title field blank 3. Fill out course term
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Please enter every course attribute

<b>ID</b>	3
<b>Testing</b>	Admin User adding a new Course - Invalid Course Information
<b>References</b>	Requirement 4.2.1
<b>Test Case</b>	1. Fill in valid and unique course code and title 2. Leave the course term field blank
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Please enter every course attribute

<b>ID</b>	4
<b>Testing</b>	Admin User adding a new Course - Invalid Course Information
<b>References</b>	Requirement 4.2.1
<b>Test Case</b>	1. Enter a course code, title, term to a combination that has already been entered
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> A course already exists with this code, title, and term.

<b>ID</b>	5
<b>Testing</b>	Admin User adding a new Course - Duplicate Course Information
<b>References</b>	Requirement 4.2.1
<b>Test Case</b>	1. Enter an existing course code and term 2. Enter a unique course title
<b>Expected Output</b>	<b>Warning Popup</b> <b>Message:</b> Either the course code exists under a different course title or the course title entered exists with a different course code. Would you like to add the course anyway? (Yes or No Options).

<b>ID</b>	6
<b>Testing</b>	Admin User adding a new Course - Duplicate Course Information
<b>References</b>	Requirement 4.2.1
<b>Test Case</b>	1. Enter an existing course title and term 2. Enter a unique course code
<b>Expected Output</b>	<b>Warning Popup</b> <b>Message:</b> Either the course code exists under a different course title or the course title entered exists with a different course code. Would you like to add the course anyway? (Yes or No Options).

<b>ID</b>	7
<b>Testing</b>	Admin User adding a new Course - Success
<b>References</b>	Requirement 4.2.1
<b>Test Case</b>	1. Enter a valid course code, title, and term
<b>Expected Output</b>	New course will be available to set as active course

### Editing An Existing Student - Acceptance Tests

<b>ID</b>	8
<b>Testing</b>	Admin User editing an existing Student; Invalid student ID
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Select an existing course using “Set Active Course” dropdown box 2. Double-click an existing student’s name 3. Delete student number information 4. Click the edit button
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid Student ID

<b>ID</b>	9
<b>Testing</b>	Admin User editing an existing Student; Invalid student ID
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Enter 10 digits for student number
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid Student ID <b>Tooltip of Student</b> <b>Number text:</b> Valid Student Number: unique 9 digit number

<b>ID</b>	10
<b>Testing</b>	Admin User editing an existing Student; Invalid student ID
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Enter 8 digits for student number
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid Student ID <b>Tooltip of Student</b> <b>Number text:</b> Valid Student Number: unique 9 digit number

<b>ID</b>	11
<b>Testing</b>	Admin User editing an existing Student; Invalid student ID
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Enter 'abc' for student number
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid Student ID <b>Tooltip of Student</b> <b>Number text:</b> Valid Student Number: unique 9 digit number



<b>ID</b>	12
<b>Testing</b>	Admin User editing an existing Student; Invalid First Name
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Hit cancel 2. Double-click existing student's name 3. Delete first name information
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid first name.

<b>ID</b>	13
<b>Testing</b>	Admin User editing an existing Student; Invalid First Name
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Edit the first name to "123"
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid first name.

<b>ID</b>	14
<b>Testing</b>	Admin User editing an existing Student; Invalid Last Name
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Hit cancel 2. Double-click existing student's name 3. Delete last name information
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid last name.

<b>ID</b>	15
<b>Testing</b>	Admin User editing an existing Student; Invalid Last Name
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Edit the last name to "123"
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid last name.

<b>ID</b>	16
<b>Testing</b>	Admin User editing an existing Student; Duplicate student ID
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	<ol style="list-style-type: none"> <li>1. Hit cancel</li> <li>2. Add a student with the 9 digit ID, "000000000"; the name, "John Johnson"; and the email address, "<a href="mailto:jjohnson@uwo.ca">jjohnson@uwo.ca</a>"</li> <li>3. Edit an existing student (not the one just added) by double clicking its name</li> <li>4. Edit the student number to "000000000"</li> </ol>
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Duplicate Student Number

<b>ID</b>	17
<b>Testing</b>	Admin User editing an existing Student; Duplicate email
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	<ol style="list-style-type: none"> <li>1. Hit cancel</li> <li>2. Edit an existing student (not the one just added) by double clicking its name</li> <li>2. Edit the email address to "jjohnson@uwo.ca"</li> </ol>
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Duplicate email

<b>ID</b>	18
<b>Testing</b>	Admin User editing an existing Student; Invalid email: no "@" or "."
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Edit the email address to "jjohnson"
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid email address.

<b>ID</b>	19
<b>Testing</b>	Admin User editing an existing Student; Invalid email: no “.”
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Edit the email address to “jjohnson@uwo
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid email address.

<b>ID</b>	20
<b>Testing</b>	Admin User editing an existing Student; Invalid email: no “@”
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Edit the email address to “jjohnsonuwo.com
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid email address.

<b>ID</b>	21
<b>Testing</b>	Admin User editing an existing Student; Invalid email: nothing after “.”
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Edit the email address to “jjohnson@uwo.”
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid email address.

<b>ID</b>	22
<b>Testing</b>	Admin User editing an existing Student; Invalid email: nothing after “@”
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	1. Edit the email address to “jjohnson@.com”
<b>Expected Output</b>	<b>Error.</b> <b>Message:</b> Invalid email address.

<b>ID</b>	23
<b>Testing</b>	Admin User editing an existing Student; Success
<b>References</b>	Requirement 4.3.1 and 4.3.2
<b>Test Case</b>	<ol style="list-style-type: none"> <li>1. Hit cancel</li> <li>2. Edit an existing student by double clicking its name</li> <li>3. Edit the information with valid information:  Student Number: "250678999"  First Name: "ValidFirstName"  Last Name: "ValidLastName"  Email Address: "thisisavalid@email.address"</li> </ol>
<b>Expected Output</b>	The student's information will update.

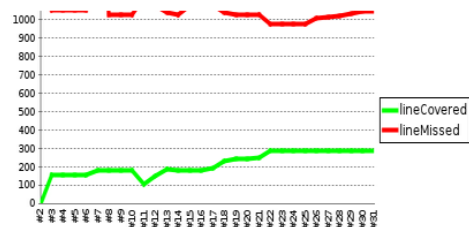
## 2.4 Jenkins CI

Jenkins Server: <http://ec2-54-81-118-198.compute-1.amazonaws.com/job/team9/>

TA User: ta

TA Password: jyandy7

## JaCoCo Coverage Report



### Overall Coverage Summary

name	instruction	branch	complexity	line	method	class
all classes	M: 6271 C: 1263 17%	M: 269 C: 171 39%	M: 331 C: 114 26%	M: 1040 C: 287 22%	M: 168 C: 52 24%	M: 47 C: 6 11%

### Coverage Breakdown by Package

name	instruction	branch	complexity	line	method	class
ca.uwo.csd.cs2212.team9	M: 6271 C: 1263 17%	M: 269 C: 171 39%	M: 331 C: 114 26%	M: 1040 C: 287 22%	M: 168 C: 52 24%	M: 47 C: 6 11%

Page generated: Mar 15, 2014 5:30:28 PM [REST API](#) [Jenkins ver. 1.553](#)

Link to JaCoCo Report:

<http://ec2-54-81-118-198.compute-1.amazonaws.com/job/team9/9/jacoco/>

		Task Mode ▾	Task Name ▾	Duration ▾	Start ▾	Finish ▾	Resource Names ▾	% Complet ▾
9								
10	✓	🚀	Team Assignment 2	14 days	Tue 14-02-11	Fri 14-02-28		100%
11	✓	📄	Revised UML Class Diagram	2 days	Sat 14-02-22	Mon 14-02-2	Tyler	100%
12	✓	📄	Revised User Stories	3 days	Mon 14-02-2	Wed 14-02-2	Bruno,Killian,Michel	100%
13	✓	📄	Class Descriptions	3 days	Tue 14-02-25	Thu 14-02-27	Michelle,Tyler,Sohra	100%
14	✓	📄	Project Plan Documentation	2 days	Wed 14-02-2	Thu 14-02-27	Sohrab,Killian	100%
15	✓	📄	GUI Design	3 days	Tue 14-02-25	Thu 14-02-27	Bruno	100%
16	✓	🚀	Milestone 2: Team Assignment 2	0 days	Fri 14-02-28	Fri 14-02-28		100%
17								
18	✓	🚀	Team Assignment 3	10 days	Mon 14-03-0	Fri 14-03-14		100%
19	✓	📄	Revised UML Class Diagram	2 days	Mon 14-03-1	Tue 14-03-11	Tyler	100%
20	✓	📄	Revised User Stories	3 days	Mon 14-03-1	Wed 14-03-1	Bruno,Killian,Michel	100%
21	✓	📄	Acceptance Tests	2 days	Mon 14-03-1	Tue 14-03-11	Bruno,Michelle	100%
22	✓	📄	Jenkins CI	3 days	Mon 14-03-1	Wed 14-03-1	Bruno	100%
23	✓	📄	Project Plan Documentation	2 days	Tue 14-03-11	Wed 14-03-1	Sohrab	100%
24	✓	📄	Prototype	5 days	Tue 14-03-04	Mon 14-03-1	Bruno,Killian,Michel	100%
25	✓	📄	Unit Tests	3 days	Tue 14-03-11	Thu 14-03-13	Killian	100%
26	✓	🚀	Milestone 3: Team Assignment 3	0 days	Fri 14-03-14	Fri 14-03-14		100%
27								
28		🚀?	Team Assignment 4					0%
29		🚀?	Revised User Stories					0%
30		🚀?	Project Video or Design Report					0%
31		🚀?	Project Plan					0%
32		🚀?	Unit Tests					0%
33		🚀?	GUI					0%
34		🚀?	Milestone 4: Team Assignment 4	0 days				0%
35								
36		🚀?	Final code submission					0%
37		🚀?	Code the User Interface					0%
38		🚀?	Code the Student class					0%
39		🚀?	Write unit tests for the Student class					0%
40		🚀?	Milestone 5: code submission	0 days				0%

