Version 1.0

syst 17796

Deliverable 3

## description

Complete the code to satisfy the rules of the game (requirements) and the use cases you defined (scope). Ensure the final version is checked in to your Git repository. You may choose to hardcode the values as and when required

After the code is finished, take the use cases from Deliverable 2 and analyze them to produce unit tests for your code. Make sure you have “good, bad and boundary” tests for each requirement/methods.

Create your tests using JUnit and code them so that they verify the good, bad and boundary cases of each method or unit. Marks will be awarded for proper design of tests as well as your source code so pay proper attention to the naming of the tests.

Finally, produce a Test Results Report that lists the requirement you are validating, the test that covers it and whether that test passed or failed on your final run (be honest, this is easily double-checked). An example of a test results report is below but you can modify it if you wish.

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement | Use Case | Test Method (ClassName.methodName) | Status (Date) |
| Player can draw a card from the deck | “Regular Play” | TestPlayGame.testDrawCard() | Pass (Apr 15, 2021) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Submission

Please submit **one PDF document per group as well as the access credentials (URL pointing to your git)**. This means that the document should be professionally organized and have a uniform style throughout. It should look as though it came from one team, not 4 separate students.

**You should also include a zip export of your project including the source code for your game and for your tests, labelled with your group name.**

### rubric

| Item | Criteria | Points | Weight |
| --- | --- | --- | --- |
| Source Code | Code is completed and conforms to the design specified by the class diagram. Code is functionally correct. Code follows standard coding conventions for comments, naming and indentation | 5 | 5 |
| Test Code + Scripts | JUnit tests are completed and pass. JUnit tests validate the one thing they are written to validate (each). JUnit tests conform to test standards taught in class. Manual test instructions are included for any requirements that are unable to be validated using JUnit | 5 | 5 |

## Final Document Layout

One PDF with the following sections denoted using page numbers, headers and a table of contents:

* Reference to the Git repository containing your source code
* Reference to the Git repository location (directory) of your JUnit tests
* Test Results Report

**AND** One zip archive containing the source code for your game and your JUnit tests, labelled with your group name.