



# **CS 1632 Software Quality Assurance**

## **Deliverable 2**

Member 1 Name: Henry Uz

Member 2 Name: Mark Kraus

Team Name: the legion

## 1. Introduction

Write your introduction here. Please describe the division of work between group members. Also, please describe any difficulties that you faced while using JUnit.

Mark's work:

- RoomUnitTest.java
- CoffeeMakerQuestUnitTest.java
- CoffeeMakerQuest.java
- CoffeeMakerQuestImpl.java
- Code Coverage Report Template

Henry's work:

- PlayerUnitTest.java
- GameIntegrationTest.java
- Player.java
- PlayerImpl.java
- Room.java
- RoomImpl.java

We faced small difficulties with live share, but this was more of a connectivity issue than a practical one. We wanted to avoid merge issues when we completed work, so our solution was once Henry completed a file, we would connect, and he would send me his file locally so I could copy it into my program. Then, I made sure it worked within my program, made changes to any bugs, and pushed the changes to the repository. This worked fine, and we had no major conflicts with this since we live together, we could easily exchange code and figure out what was and wasn't working. The issues we faced specifically in JUnit had to do with mocking. Mark wrote the JUnit test cases all without mock objects for Player and Rooms, and once I submitted them to Gradescope, I immediately realized I needed to make mock objects for those, so I had to rework some of my tests. Additionally, Mark had to add more test cases to reach 90% code coverage.

## 2. Code Coverage

At below, paste a screenshot of code coverage stats for CoffeeMakerQuestImpl.java, PlayerImpl.java, RoomImpl.java, and Game.java **after** having completed the test code, in that order.

### CoffeeMakerQuestImpl

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
● processCommand(String)		93%		75%	7	18	3	34	0	1
● areDoorsPlacedCorrectly()		96%		80%	2	6	1	9	0	1
● setCurrentRoom(Room)		91%		50%	2	3	1	5	0	1
● areRoomsUnique()		100%		83%	1	4	0	11	0	1
● CoffeeMakerQuestImpl(Player, ArrayList)		100%		n/a	0	1	0	6	0	1
● isGameOver()		100%		100%	0	2	0	3	0	1
● getHelpString()		100%		n/a	0	1	0	3	0	1
● static {...}		100%		n/a	0	1	0	1	0	1
● getCurrentRoom()		100%		n/a	0	1	0	1	0	1
● getInstructionsString()		100%		n/a	0	1	0	1	0	1
Total	15 of 340	95%	12 of 51	76%	12	38	5	74	0	10

### PlayerImpl

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
● getInventoryString()		100%		100%	0	4	0	14	0	1
● PlayerImpl()		100%		n/a	0	1	0	4	0	1
● hasItem(Item)		100%		100%	0	2	0	3	0	1
● addItem(Item)		100%		n/a	0	1	0	2	0	1
● static {...}		100%		n/a	0	1	0	1	0	1
Total	0 of 84	100%	0 of 8	100%	0	9	0	24	0	5

### RoomImpl

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
● getDescription()		100%		100%	0	3	0	8	0	1
● RoomImpl(String, String, Item, String, String)		100%		n/a	0	1	0	7	0	1
● static {...}		100%		n/a	0	1	0	1	0	1
● getFurnishing()		100%		n/a	0	1	0	1	0	1
● getAdjective()		100%		n/a	0	1	0	1	0	1
● getItem()		100%		n/a	0	1	0	1	0	1
● getNorthDoor()		100%		n/a	0	1	0	1	0	1
● getSouthDoor()		100%		n/a	0	1	0	1	0	1
Total	0 of 93	100%	0 of 4	100%	0	10	0	21	0	8

### Game

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
● main(String[])		81%		66%	2	4	6	23	0	1
● Game()		0%		n/a	1	1	1	1	1	1
Total	16 of 74	78%	2 of 6	66%	3	5	7	24	1	2