IC470: Software Engineering
Lab1

Due: As per the course syllabus

Lab Focus: Functional Requirements and Acceptance Test Cases

Things you'll need: Craps.jar

Lab Partners: If there are other mids in this section from your capstone team, you may work together as a lab group. If not, pair up with another free mid from this section and work together on this lab.

Lab Partner Names _____

Functional requirements are the things that a Customer needs a software system to do. **Acceptance Test Plan test cases** are used to check/prove whether the functional requirements have been met by the operational software. We will consider 3 types of test cases:

- "**Normal**" test cases are used to demonstrate that the software meets the indicated function requirement. The *expected* uses of the system.
- "**Abnormal**" test cases demonstrate the result of the software entering an unusual (but possible) state, such as when a user provides invalid input. The *unexpected* (but possible) uses of the system.
- "**Not Useful**" cases describe a state that the properly running software cannot be placed in. Note that these are *unsuitable* as Acceptance Test Plan test cases since a properly functioning system cannot demonstrate passing such test cases.

1. **Passing Acceptance Test Plan test cases.**

a) Download, open and run Craps.jar (see the course web page). Roll the dice once and see if a number between 2..12 is produced. Complete the Testing Results column of the table below:

Functional Requirement	**Acceptance Test Plan** test case	**Testing results**
1a. Dice roll input. The Craps software system allows the user to simulate randomly rolling two standard six-sided die.	1.a User rolls dice, the sum of the two die is a number between 2..12. (Normal)	1. a System passes test case? Circle Yes or No and explain. Yes / No If no, explain what would need to change and how:

b) Using Craps.jar, roll the dice 25 times, and keep track of the number of occurrences of each resulting sum for the first 25 rolls of the two die. Complete the Number Observed and Percentage Observed columns below.

Frequency Distributions for 2 standard six-sided die (source: American Mathematical Society)

Total on dice	Pairs of dice	Probability	Number Observed	Percentage Observed
2	**1+1**	**1/36 = 3%**		
3	**1+2, 2+1**	**2/36 = 6%**		
4	**1+3, 2+2, 3+1**	**3/36 = 8%**		
5	**1+4, 2+3, 3+2, 4+1**	**4/36 = 11%**		
6	**1+5, 2+4, 3+3, 4+2, 5+1**	**5/36 = 14%**		
7	**1+6, 2+5, 3+4, 4+3, 5+2, 6+1**	**6/36 = 17%**		
8	**2+6, 3+5, 4+4, 5+3, 6+2**	**5/36 = 14%**		
9	**3+6, 4+5, 5+4, 6+3**	**4/36 = 11%**		
10	**4+6, 5+5, 6+4**	**3/36 = 8%**		
11	**5+6, 6+5**	**2/36 = 6%**		
12	**6+6**	**1/36 = 3%**		

c) Complete the Testing column of the following table based *just* on the values in your "Percentage Observed" column. If you say no, explain what would need to change (the Functional Requirement, the Acceptance Test Plan test case, etc) in order to better prepare for Acceptance Testing, and how.

Functional Requirement	**Acceptance Test Plan** test case	**Testing results**
1.b Dice rolls meet established frequency distributions. The sums produced from the dice rolls must agree with established frequency distributions.	1.b User rolls dice 25 times, and the resulting sums agree with established frequency distributions. (Normal)	1. b System passes test case? Circle Yes or No and explain. Yes / No If no, explain what would need to change and how:

2. Evaluating Acceptance Test Plan test case evaluations. Using Craps.jar, complete the following in the table below:

Functional Requirement	**Acceptance Test Plan test cases (scenarios showing that Functional Requirement is met)**	**Testing results**
2. Bankroll. Craps software system maintains a bankroll to which user wins are added and losses are deducted.	2.a Start Craps as needed until user wins on the first roll, check that bankroll *increases* by the amount bet. Is this test case (circle one): ○ Normal, ○ Abnormal, or ○ Not Useful	2.a System passes test case? Circle Yes or No. Yes / No If no, explain what would need to change and how:
	2.b Start Craps as needed until user wins on the first roll, check that bankroll *decreases* by the amount bet. Is this test case (circle one): ○ Normal, ○ Abnormal, or ○ Not Useful	2.b System passes test case? Circle Yes or No. Yes / No If no, explain what would need to change and how:

	2.c Add a new test case here that tests an aspect of this functional requirement that is not already being tested. Do not provide a Not Useful test case! Is this test case (circle one): ○ Normal, or ○ Abnormal	2.c System passes test case? Circle Yes or No. Yes / No If no, explain what would need to change and how:
3. Debt. User is not allowed to owe money	3.a Start Craps, play until user's bankroll becomes zero, and the user is prevented from playing further. Is this test case (circle one): ○ Normal, ○ Abnormal, or ○ Not Useful	3.a System passes test case? Circle Yes or No. Yes / No If no, explain what would need to change and how:
4. Bankroll File. System keeps track of the user's bankroll amount in a file when prompted via "Save Bankroll to File" button.	4.a Start Craps, but do not provide any input (do not click any buttons) for 30 seconds. System displays a prompt with the text "Are you still there?" Whenever the user selects "yes" within an additional 30 seconds of being prompted, the 30 second input countdown repeats, otherwise, the system exits. Is this test case (circle one): ○ Normal, ○ Abnormal, or ○ Not Useful	4.a System passes test case? Circle Yes or No. Yes / No If no, explain what would need to change and how:
	4.b Start Craps, play until user wins twice without losing, then click the "Save Bankroll to File" button. Game stores 60 in the bankroll.txt file. Is this test case (circle one): ○ Normal, ○ Abnormal, or ○ Not Useful	4.b System passes test case? Circle Yes or No. Yes / No If no, explain what would need to change and how:

3. Functional Requirements and corresponding Acceptance Test Plan test cases

Add to the below table at position 5. Your new table is to give one completely new functional requirement that the Craps software *does not currently do*, but is something that you think users would be willing to pay $$$ to have coded into the software. Also, give two acceptance test plan test cases (one Normal and one Abnormal) that can be used to *objectively verify* the new functional requirement that you added.

Functional Requirement	Acceptance Test Plan test cases (scenarios showing that Functional Requirement is met)	Your Evaluation
5. Add your completely new functional requirement here …	5.a Add your Normal test case for this new functional requirement here …	5.a Explain why this is a Normal test case.
	5.b Add your Abnormal test case for this new functional requirement here …	5.b Explain why this is an Abnormal test case.