**Familiarity Review Template**

**Name:** Bryce Blauser

**Date:** 6/21/2019

**Week:** 9

**Coding Topic:**

Sequence Diagrams

State Diagrams

System Tests

Use Case Diagrams

Use Case Documents

**Description of Understanding:**

I grouped all the diagrams and documents together since they are all kind of related with one another. I based the documents and diagrams off the same example code I submitted for my Application Controller program. Each topic will be on a separate page of this document with a separate description of understanding.

**Teaching Video:**None

**Starting at:**

**Also Integrated with:**

|  |  |  |  |
| --- | --- | --- | --- |
| **File** | **Git Link** | **What should I be looking for?** | **Sandbox or Your code?** |
| Sequence Diagram\_Example.jpg | [../Diagrams\_Documents/Sequence%20Diagram\_Example.jpg](https://github.com/Blauser-Bryce/CIT-360/blob/master/Diagrams_Documents/Sequence%20Diagram_Example.jpg) |  | Mine |
| State Diagram\_Example.jpg | [../Diagrams\_Documents/State%20Diagram\_Example.jpg](https://github.com/Blauser-Bryce/CIT-360/blob/master/Diagrams_Documents/State%20Diagram_Example.jpg) |  | Mine |
| System Tests\_Example.pdf | [../Diagrams\_Documents/System%20Tests\_Example.pdf](https://github.com/Blauser-Bryce/CIT-360/blob/master/Diagrams_Documents/System%20Tests_Example.pdf) |  | Mine |
| Use Case Diagram\_Example.jpg | [../Diagrams\_Documents/Use%20Case%20Diagram\_Example.jpg](https://github.com/Blauser-Bryce/CIT-360/blob/master/Diagrams_Documents/Use%20Case%20Diagram_Example.jpg) |  | Mine |
| Use Case\_Example.xlsx | [../Diagrams\_Documents/Use%20Case\_Example.xlsx](https://github.com/Blauser-Bryce/CIT-360/blob/master/Diagrams_Documents/Use%20Case_Example.xlsx) |  | Mine |

**Coding Topic:** Sequence Diagrams

**Description of Understanding:** I created a sequence diagrams showing the number guessing portion of my program. It shows how the user goes to the main menu, where it then prompts the user for an entry. If they enter an invalid selection, it loops them back to the main menu. If they enter “N”, then the request is handled and sent to the guess view. The user can then enter a guess and it will check if they guessed the correct number and return the result back to the guess view. If the user decides they are done, they can enter X to return to the main menu and quit the program.

A screenshot of a social media post

Description automatically generated

**Coding Topic:** State Diagrams

**Description of Understanding:** This diagram shows the state of the program as they user navigates through the menu and makes selections. It is similar to the sequence diagram, shows the different states the program can be in and how it flows between those states.

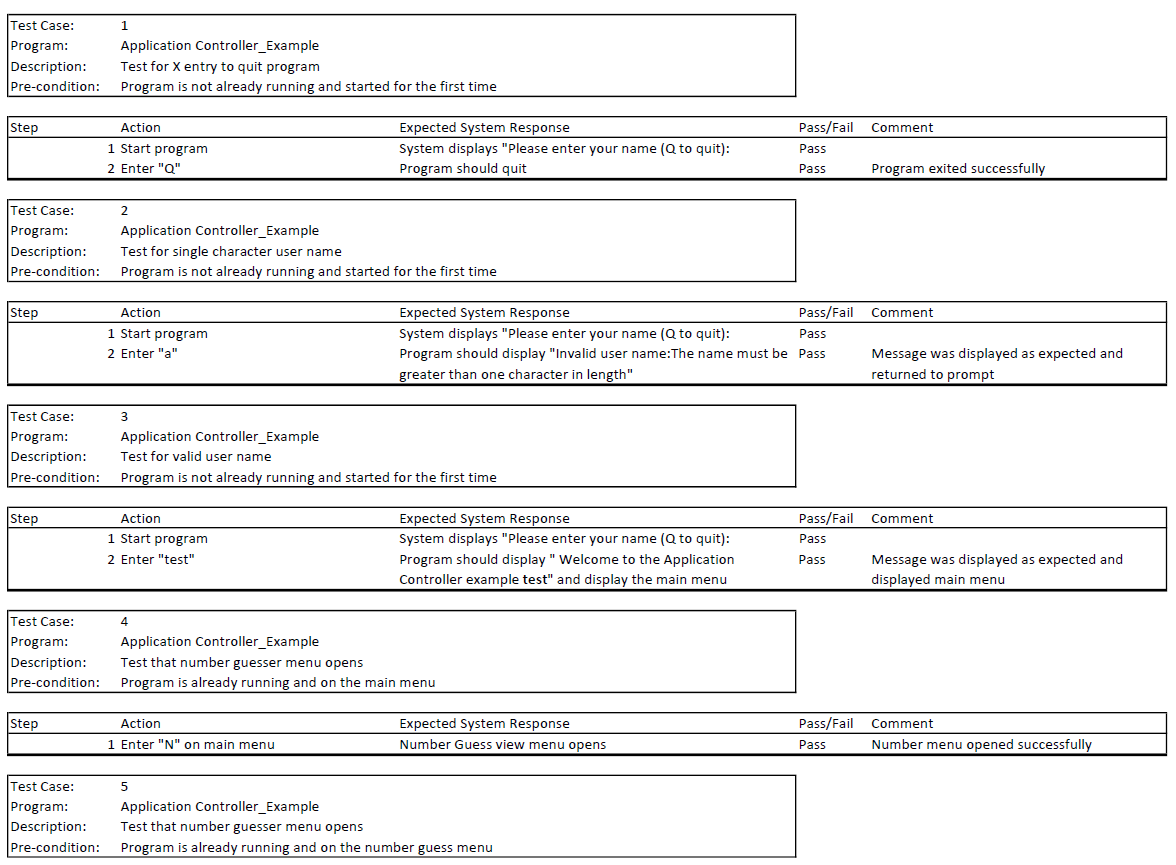
A picture containing screenshot

Description automatically generated

**Coding Topic:** System Tests

**Description of Understanding:** I created 8 user tests based on the different values a user can enter. I tested both valid entries and invalid entries to ensure the code handled and displayed error messages properly. I showed the steps to repeat the tests, what results should be expected, whether they passed or failed and comments for each.

Use GitHub link to view entire PDF [../Diagrams\_Documents/System%20Tests\_Example.pdf](https://github.com/Blauser-Bryce/CIT-360/blob/master/Diagrams_Documents/System%20Tests_Example.pdf)



**Coding Topic:** Use Case Diagrams

**Description of Understanding:** I created a basic use case diagram show the systems functions, the actor and how they interact with the functions in the program. The user can enter their name, guess a random number, get the results displayed and quit the program. The different functions show how they handle the user’s request and the relationship.

A close up of a map

Description automatically generated

**Coding Topic:** Use Case Document

**Description of Understanding:** I created a very simple use case document to demonstrate how one would be created for the Guess Number portion of the program. It shows preconditions, where the case starts, success criteria and what the results should be. It also shows other success routes for errors and shows the rules that the program must adhere to.

|  |  |
| --- | --- |
| **Use Case:** | GuessNumber |
| **Summary:** | Documents how the Number class functions |
| **Preconditions:** | The user was able to successfully enter the main menu screen by entering a valid user name |
| **Case Starts With:** | The user selects the option to play Guess Number by pressing "N" from the main menu |
| **Main Success:** | 1. The system generates a random number between 1 and 10 2. The user enters a valid number value into the guess prompt (between 1 and 10) |
| **Other Success Route:** | No other success routes. An error will be thrown if an invalid value is entered. |
| **Postconditions:** | 1. The system responds by notifying the user if they guessed the number correctly |
| **Rules:** | 1. The guessed value must be a numeric between 1 and 10 |
| **Notes:** | User can exit the prompts back to the main menu by entering "Q" |
| **Author:** | Bryce Blauser |
| **Date:** | 6/21/2019 |