**Product Requirements**

| **Team** | <Your Team’s Account and Name in Black> |
| --- | --- |

***Note: this is a “living document”, meaning its content will change with the implementation of the project. Use it to capture key project requirements and make sure that your product features match the requirements exactly – if you wish to add any features, they must be added first to the requirements. The requirements document, and all changes to it, must be approved by the customer (instructor).  
Remove this text and the descriptive paragraphs in each section stating what to do before you add this to your repository or turn it in to your instructor.***

# **Brief problem statement**

***Replace this text and the instructions below with your statement in black.***  
(2-5 lines describing the problem being addressed. Note that even if you are simply restating what is already in the needs document, you must rephrase it in your words. This gives an opportunity for the customer to identify and provide feedback on differences in interpretation, if any).

# **System requirements**

***Replace this text and the instructions below with your statement in black.***(1 or more lines identifying the system requirements for your solution. If you require particular languages and libraries, list them as well).

# **Users profile**

***Replace this text and the instructions below with your statement in black.***(Identify who will be using the system, in what mode, and their profile in terms of familiarity with using computers and such software. If taking material from needs document, then rephrase in your words).

# **Feature requirements (user stories)**

***Read the instructions below and fill in the table. Delete all the blue text before adding this to your repository or turning it in to your instructor.***  
(This is a numbered list of user stories that are the features of the system to be implemented. Each user story is an operation that the user can perform on/with the system. For each user story, provide a fairly detailed description so you know what to build and so you can write a test case to demonstrate that your system provides that feature. For each user story, you will identify (during release planning) the release in which it will be implemented: R1, R2 or R3. Typically, your system will have 10-20 stories or features, but feel free to add more table rows if you decide to use finer-grain stories).

| **No.** | **User Story Name** | **Description** | **Release** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Use case diagram**

***Read the instructions below and fill in the table. Delete all the blue text before adding this to your repository or turning it in to your instructor.***

***Draw the UML use case context diagram for the system. Make sure the use cases shown in the diagram correspond to the user stories described in the previous section.***

**Use case description**

| **Use Case Number:** | ***UC-01*** |
| --- | --- |
|  |  |
| **Use Case Name:** | ***Play Snake Game*** |
| **Overview:** | ***The purpose of this use case is to allow the user to play the classic Snake game.*** |
| **Actor(s):** | ***Player*** |
| **Pre condition(s):** | ***The player must run the code.***  ***The player has launched the game.*** |
| **Scenario Flow:** | ***Main (success) Flow:***  ***The game menu is displayed.***  ***The player selects "Play Game".***  ***The game board is displayed with a snake and food item.***  ***The player uses the arrow keys to move the snake around the game board.***  ***If the snake hits its own body, the game ends.***  ***If the snake eats the food item, it grows in size and the player scores points.***  ***Steps 4-6 are repeated until the game ends.***  ***When the game ends, the player's score is displayed along with the option to play again or return to the menu.*** |
|  | ***Alternate Flows:***  ***The player selects "How to Play" from the menu:***  ***The instructions for the game are displayed.***  ***The player returns to the menu.*** |
| **Post Condition:** | ***The player's score is displayed.***  ***The player can choose to play again or return to the menu.*** |