

1. Brief introduction _/3

My champion feature will be making the camera move around in a fancy way that respond to the player's cursor position in addition to the player's position. By default, the camera will follow the player wherever they go, keeping them at the center of the screen. With the functionality of my champion feature, the camera will also shift around in the direction of the cursor. The camera will also shift farther away from the center of the screen and zoom farther out the farther away the cursor is from the player.

2. Use case diagram with scenario _14

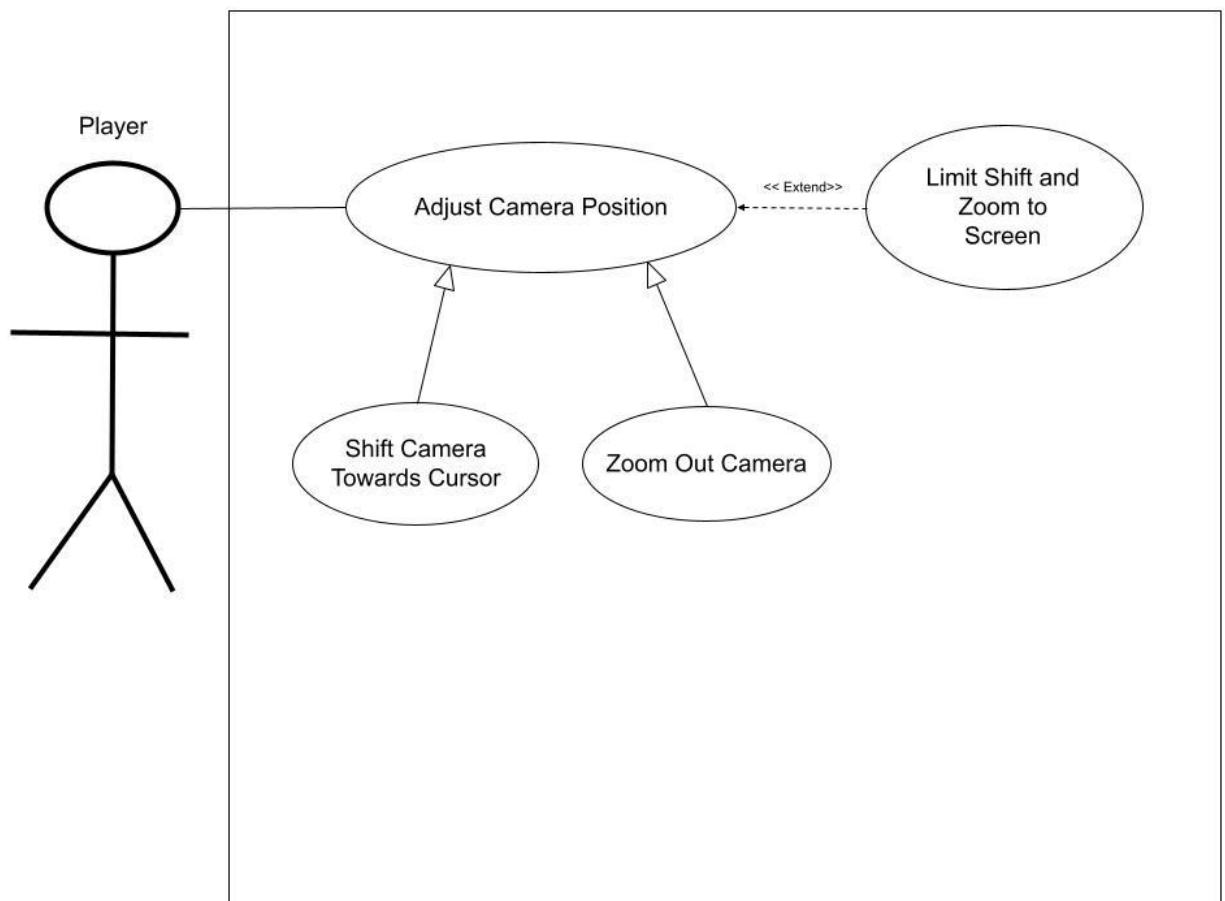
[Use the lecture notes in class.

Ensure you have at least one exception case, and that the <<extend>> matches up with the Exceptions in your scenario, and the Exception step matches your Basic Sequence step.

Also include an <<include>> that is a suitable candidate for dynamic binding]

Example:

Use Case Diagrams



Scenarios

Name: Adjust Camera Position

Summary: Camera gets shifts itself around the player and calculates how much it should zoom in or out based on the location of the cursor relative to the player.

Basic sequence:

Step 1: Get player's position.

Step 2: Get cursor's position

Step 3: Get the 2D vector difference between the cursor's position and the player's position ($\text{Vector2 cursor} - \text{Vector2 player}$) and store it as some Vector2 v.

Step 4: Divide v by i (some positive number TBD).

Step 5: Calculate the distance between the cursor's position and the player's position and store it as a number d.

Step 6: Divide d by j (some positive number TBD).

Step 6: Make the camera's new position the player's current position + v.

Step 7: Make the camera's zoom level be equal to k (some positive number TBD) + d.

Step 8: Repeat every frame.

Exceptions:

Step 1: The cursor goes beyond the width or the height of the game's window.

Step 2: The camera assumes that the cursor is on the edge of the window rather than outside of it before doing any calculations to prevent the camera from shifting farther than it should be able to.

Post conditions: Camera is shifted towards the cursor and zoomed out based on the distance between the player and the cursor.

Priority: 3

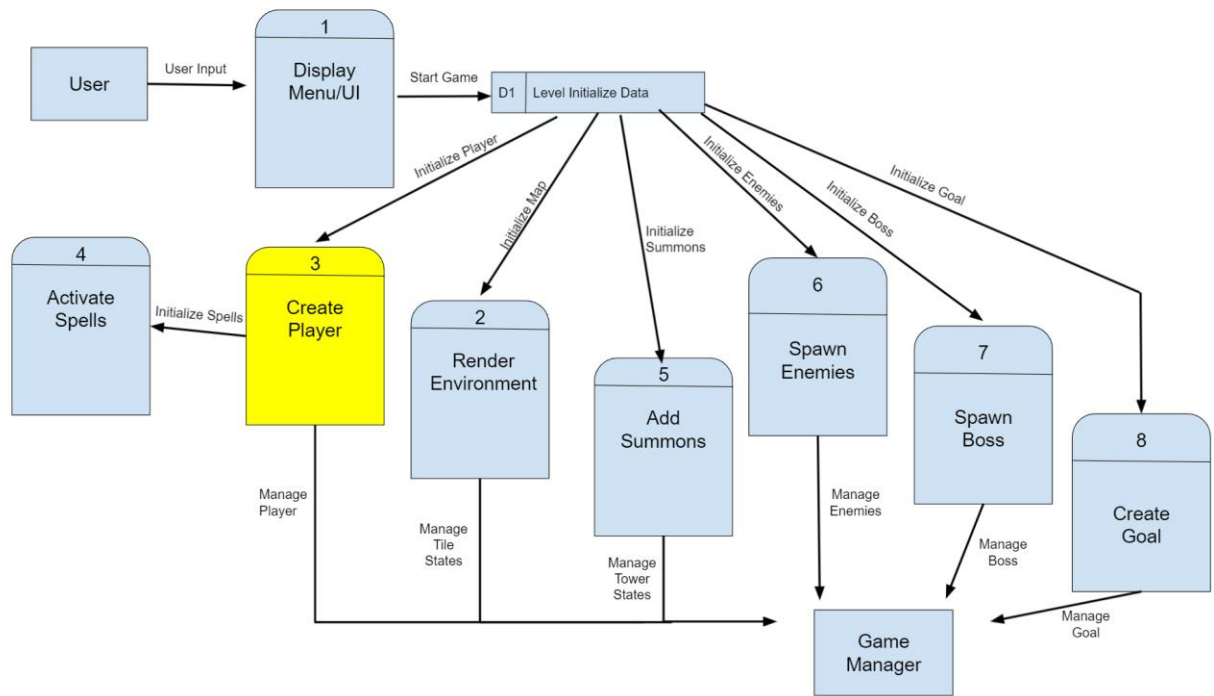
ID: ACP1

*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

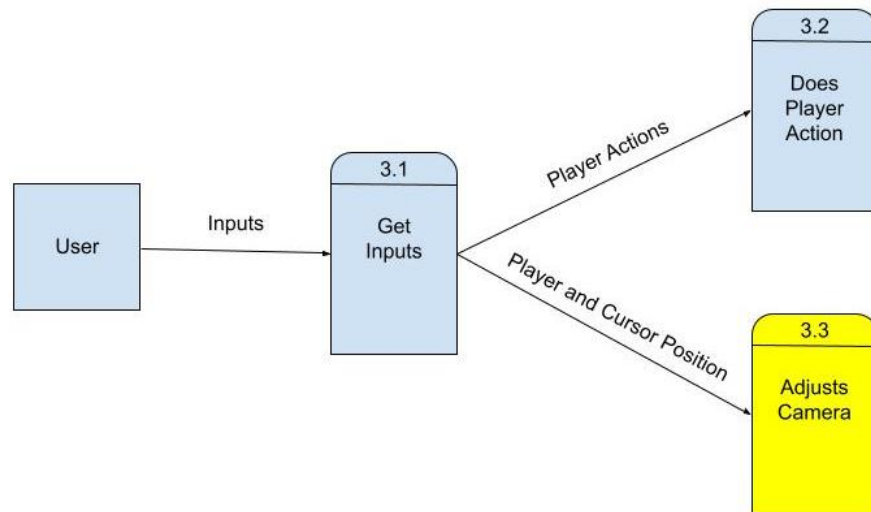
3. Data Flow diagram(s) from Level 0 to process description for your feature ____14

Data Flow Diagrams

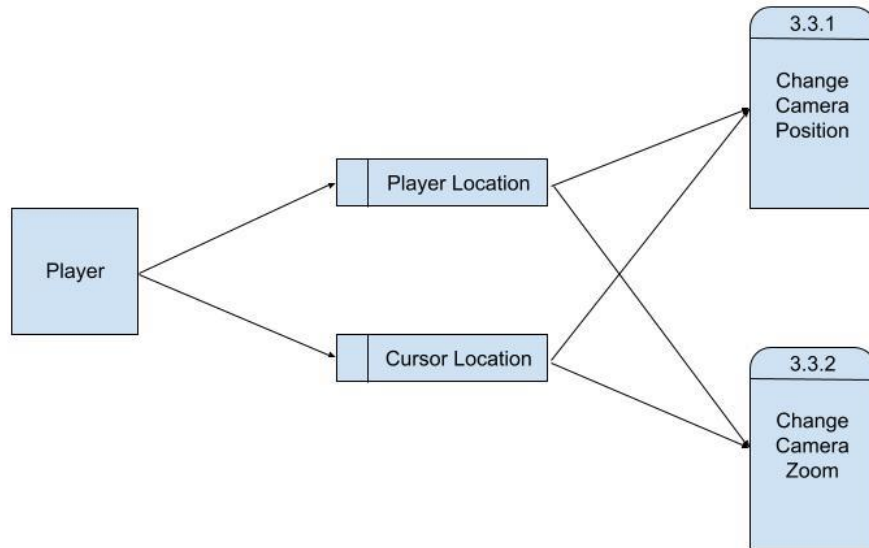
Diagram 0



3: Create Player



3.3: Adjust Camera



Process Descriptions

Change Camera Position:

Vector2 player

Vector2 cursor

IF cursor is outside the window to the left

 assume the cursor x position to be the left side of the window

ENDIF

IF cursor is outside the window to the right

 assume the cursor x position to be the right side of the window

ENDIF

IF cursor is below the window

 assume the cursor y position to be the bottom of the window

ENDIF

IF cursor is above the window

 assume the cursor y position to be the top of the window

ENDIF

Vector2 v = cursor – player

v = v / i

Camera Position = player + v

Change Camera Zoom:

Vector2 player

Vector2 cursor

num d = distance between player and cursor

d = d / j

Camera Zoom = k + d

4. Acceptance Tests _____9

Basic Value Verification

This test will be carried out by loading up the game in the unity preview and viewing the x and y coordinates of the player, the cursor, and the camera, as well as the zoom value of the camera.

Assume i and j (vector dividing values) are equal to 10 and that k (minimum camera zoom value) is equal to 5.

Player Position In World (x, y)	Cursor Position Relative to Player (x, y)	Expected Camera Position (x, y)	Expected Camera Zoom Value
(0, 0)	(-1, -1)	(-0.1, -0.1)	5.41421356
(0, 0)	(1, -1)	(0.1, -0.1)	5.41421356
(0, 0)	(1, 1)	(0.1, 0.1)	5.41421356
(0, 0)	(-1, 1)	(-0.1, 0.1)	5.41421356
(0, 0)	(-5, -5)	(-0.5, -0.5)	12.07106781
(0, 0)	(5, -5)	(0.5, -0.5)	12.07106781
(0, 0)	(5, 5)	(0.5, 0.5)	12.07106781
(0, 0)	(-5, 5)	(-0.5, 0.5)	12.07106781
(-5, 5)	(-1, -1)	(-5.1, 4.9)	5.41421356
(-5, 5)	(1, -1)	(-4.9, 4.9)	5.41421356
(-5, 5)	(1, 1)	(-4.9, 5.1)	5.41421356
(-5, 5)	(-1, 1)	(-5.1, 5.1)	5.41421356
(-5, 5)	(-5, -5)	(-5.5, 4.5)	12.07106781
(-5, 5)	(5, -5)	(-4.5, 4.5)	12.07106781
(-5, 5)	(5, 5)	(-4.5, 5.5)	12.07106781
(-5, 5)	(-5, 5)	(-5.5, 5.5)	12.07106781

(5, -5)	(-1, -1)	(4.9, -5.1)	5.41421356
(5, -5)	(1, -1)	(5.1, -5.1)	5.41421356
(5, -5)	(1, 1)	(5.1, -4.9)	5.41421356
(5, -5)	(-1, 1)	(4.9, -4.9)	5.41421356
(5, -5)	(-5, -5)	(4.5, -5.5)	12.07106781
(5, -5)	(5, -5)	(5.5, -5.5)	12.07106781
(5, -5)	(5, 5)	(5.5, -4.5)	12.07106781
(5, -5)	(-5, 5)	(4.5, -4.5)	12.07106781

Out of Bounds Testing

This test will ensure that if the player moves the cursor outside of the game window, then the camera will not continue to shift and zoom out with the cursor beyond the game window. This will be to ensure that players won't be able to abuse the camera's mechanics to see more on the screen or see things farther away by moving the cursor outside the game window.

This test will be carried out by determining the expected camera x and y position values and the expected camera zoom value for when the cursor is on the edge of the screen (screen size dependent, will need to calculate expected values later). Once those values are calculated, the cursor will be moved outside the game window to the left, top left, above, top right, right, bottom right, below, and bottom left. In each of those positions, the x position, y position, and zoom values of the camera will be checked to ensure that they are within the expected bounding values that were calculated in advance.

5. Timeline ____/10

[Figure out the tasks required to complete your feature]

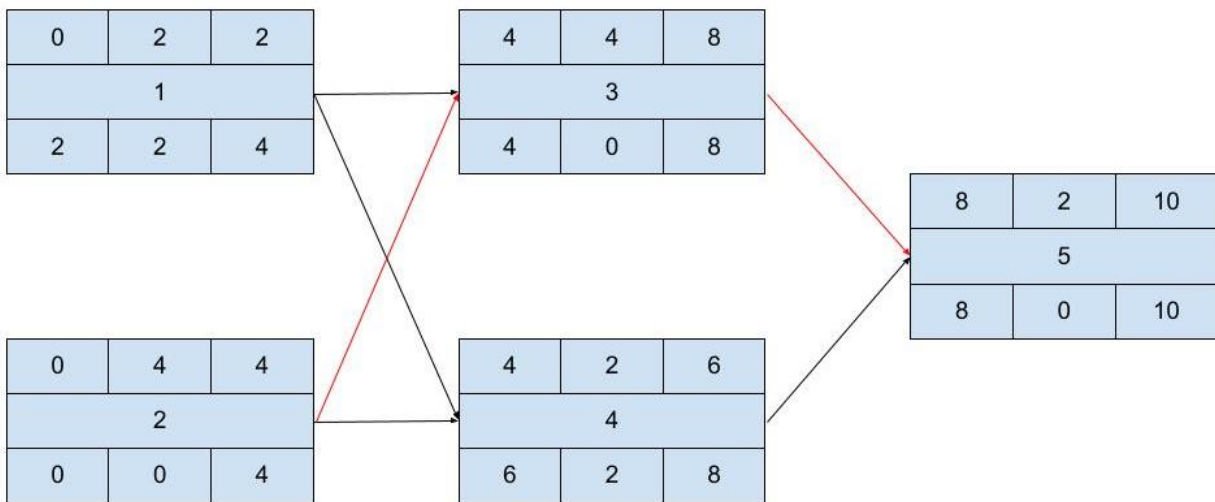
Example:

Work items

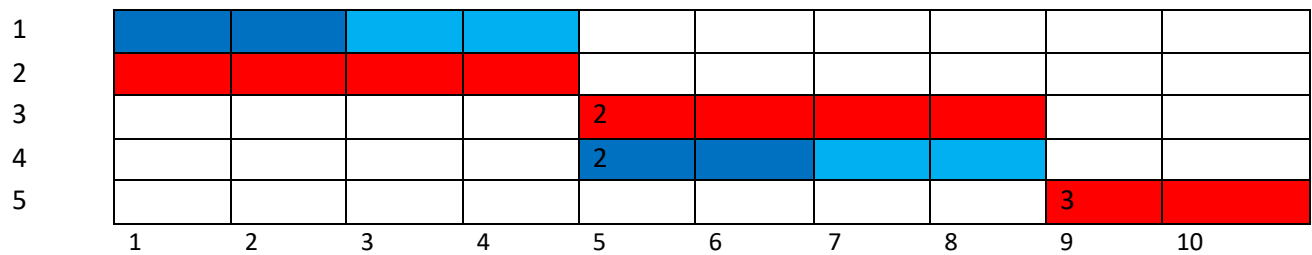
Task	Duration (Hours)	Predecessor Task(s)
1. Get camera to follow player	2	-
2. Get mouse position with	4	-

unity's new input system		
3. Get camera to shift based on mouse position	4	1, 2
4. Get camera to zoom based on mouse position	2	1, 2
5. Testing	2	3, 4

Pert diagram



Gantt timeline



Red: Critical Path

Blue: Alternate Path

Turquoise: Slack