

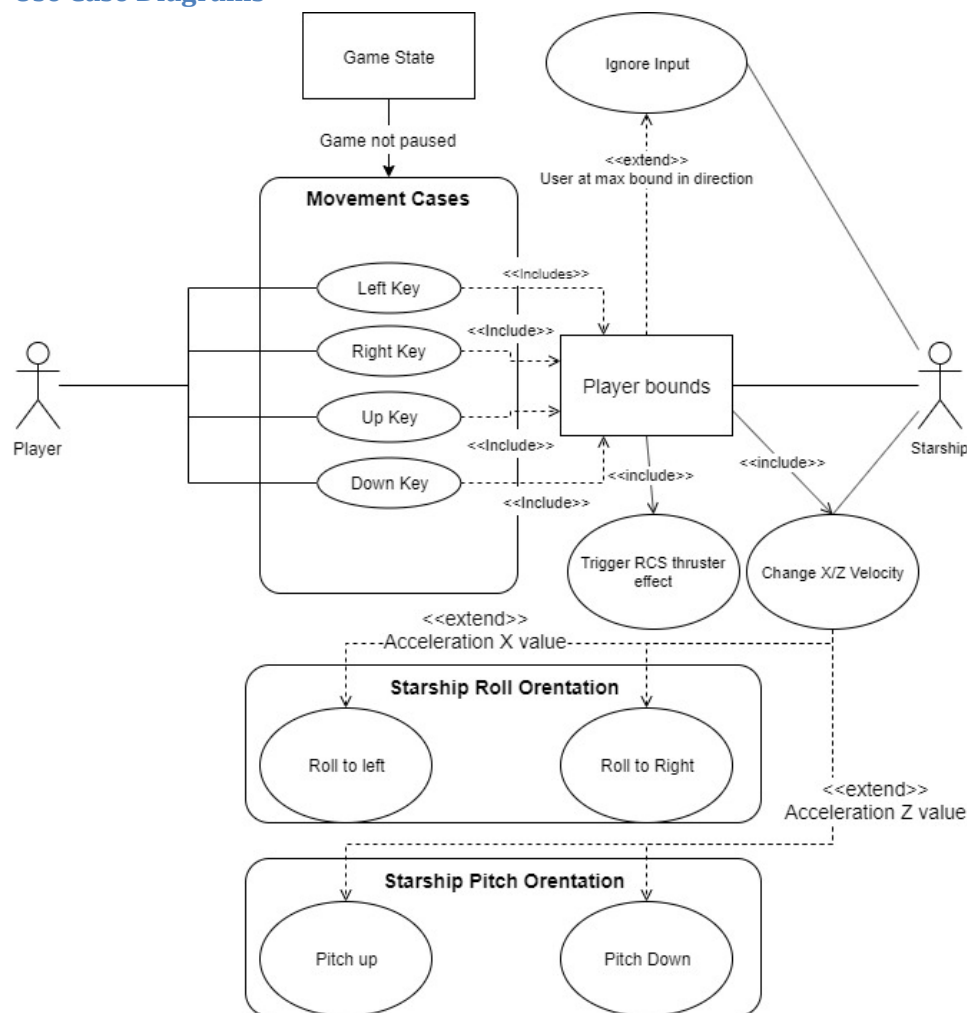
[Instructions: Remove everything that is not a heading below and fill in with your own diagrams, etc.]

## 1. Brief introduction \_\_/3

The following case diagram details the functions of user input on the space-craft during the game. Followed by the data flow diagram, and chart listing necessary time to write and implement said functionality.

## 2. Use case diagram with scenario \_\_14

### Use Case Diagrams



### Scenarios

**Name:** User Movement input

**Summary:** The user uses input keys to test bounds of starship then move if within said bounds.

**Actors:** Player, Starship

**Preconditions:** Game not paused

**Basic sequence:**

**Step 1:** Player presses key

**Step 2:** Game check present bounds of starship

**Step 3:** Game update velocity of starship

**Step 4:** Based on acceleration change orientation

**Exceptions:**

**Step 1:** If Player/Starship at max bounds do no change

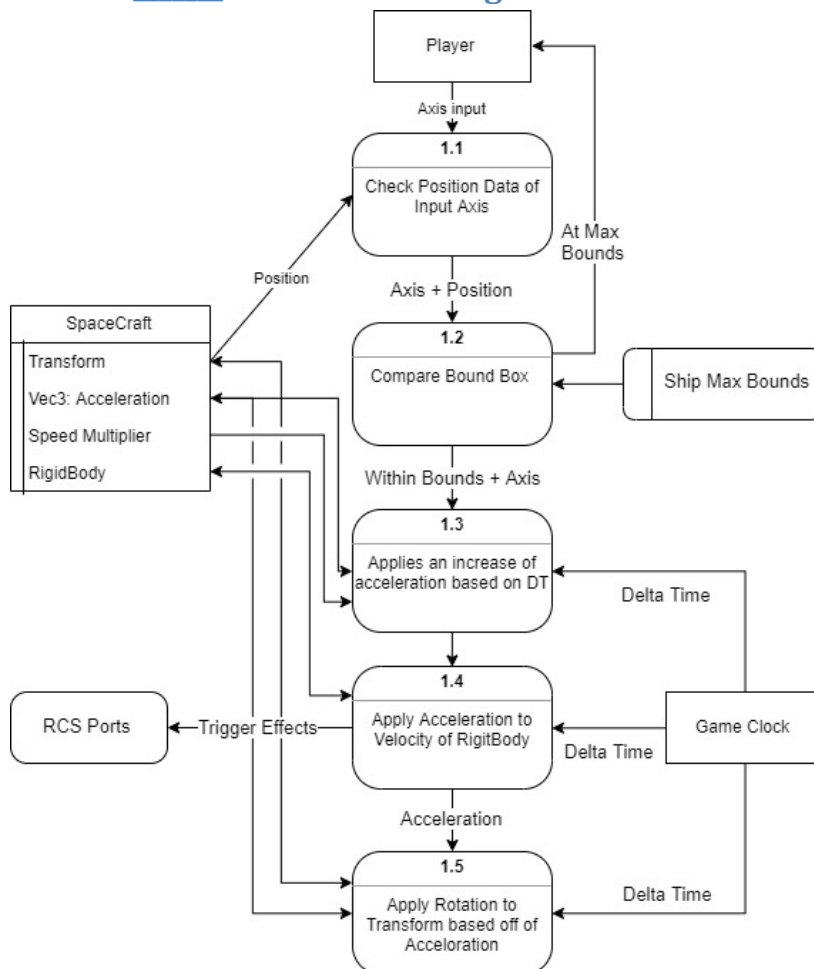
**Step 2:** Reduces pitch/roll to compensate

**Post conditions:** Craft moved

**Priority:** 1

**ID:** C01

### 3. Data Flow diagram(s) from Level 0 to process description for your feature \_\_\_\_14Data Flow Diagrams



### Process Descriptions

Movement:

- i. Occurs every frame tick to check player input.
- ii. Takes in spacecraft current position
- iii. Compares to game max bounds on axis of input
- iv. Modifies spacecraft member variables to change acceleration, velocity and position.
- v. Based on previous factors change rotation of craft to give visual representation to user.

#### 4. Acceptance Tests \_\_\_\_\_9

Create C# unit test that outputs all potential rotation outputs over a particular interval but gives focused input of any oddities of angle. Second to this and more useful would be general user test messing around with inputs. The best way to break something is to give it to the user.

#### 5. Timeline \_\_\_\_\_/10

[Figure out the tasks required to complete your feature]

##### Work items

Task	Duration (PWks)	Predecessor Task(s)
1. Develop test Object	1.5	-
2. Create test GUI for object	1.5	1
3. Program initial input and transition.	3	1
4. Program and check rotation/transition	5	1, 2, 3
5. Program connection to RCS	3	4
5. User Documentation	6	5
6. Testing	3	6
7. Installation	1	4, 6