*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

Feature Document

User Story ID #726 Implement Conditions to Escape Room (Puzzle 2)

**Name:** Daniel Perez

**Team Member(s):** Armando Carrasquillo, Daniel Perez, Santiago Bolivar

**Project:** VR-Gaming to Broad Participation in CS

**Product Owner(s)**: Francisco Ortega

**Mentor(s)**: Francisco Ortega

**Instructor**: Masoud Sadjadi

**User Story Name: Implement Conditions to Escape Room (Puzzle 2)**

* Description: As a developer, I want the user to place the bad apple they found on one plate and only one good apple on the other plate, so that the user can escape the room if those two conditions are met.

Acceptance Criteria

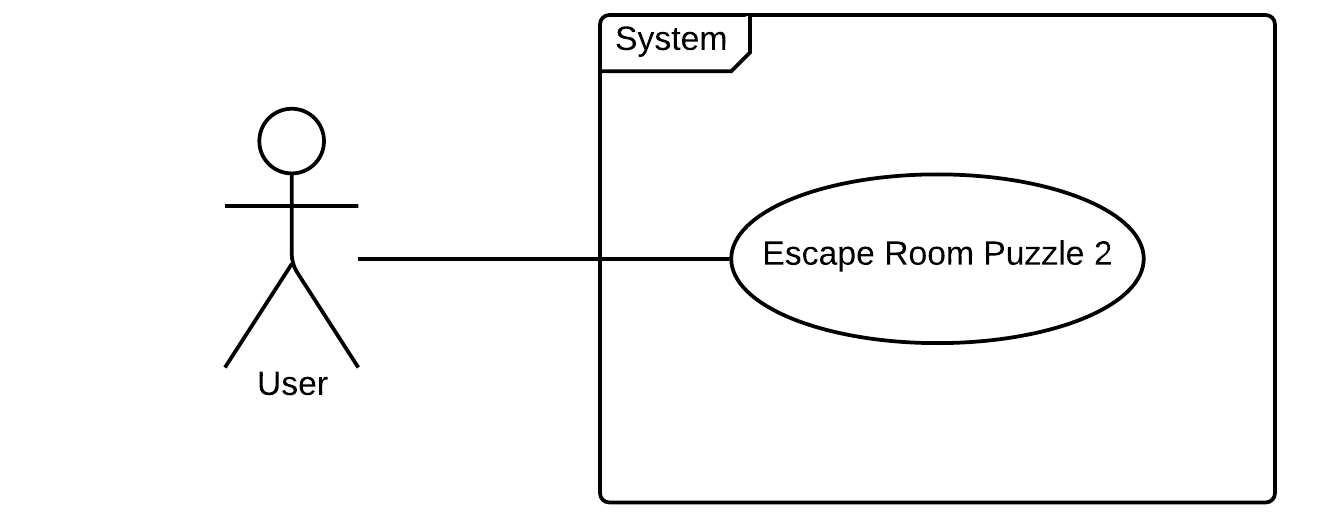
* Verify that the function recognizes the type of apple placed on each plate.
* Verify that the puzzle is solved if and only if the bad apple is on one plate and only one good apple is on the other plate.
* Verify that the system is notified when the puzzle is solved.
* Verify that the door opens when the puzzle is solved.
* Verify that the door stays open as long as the puzzle remains solved.

**Use Case**

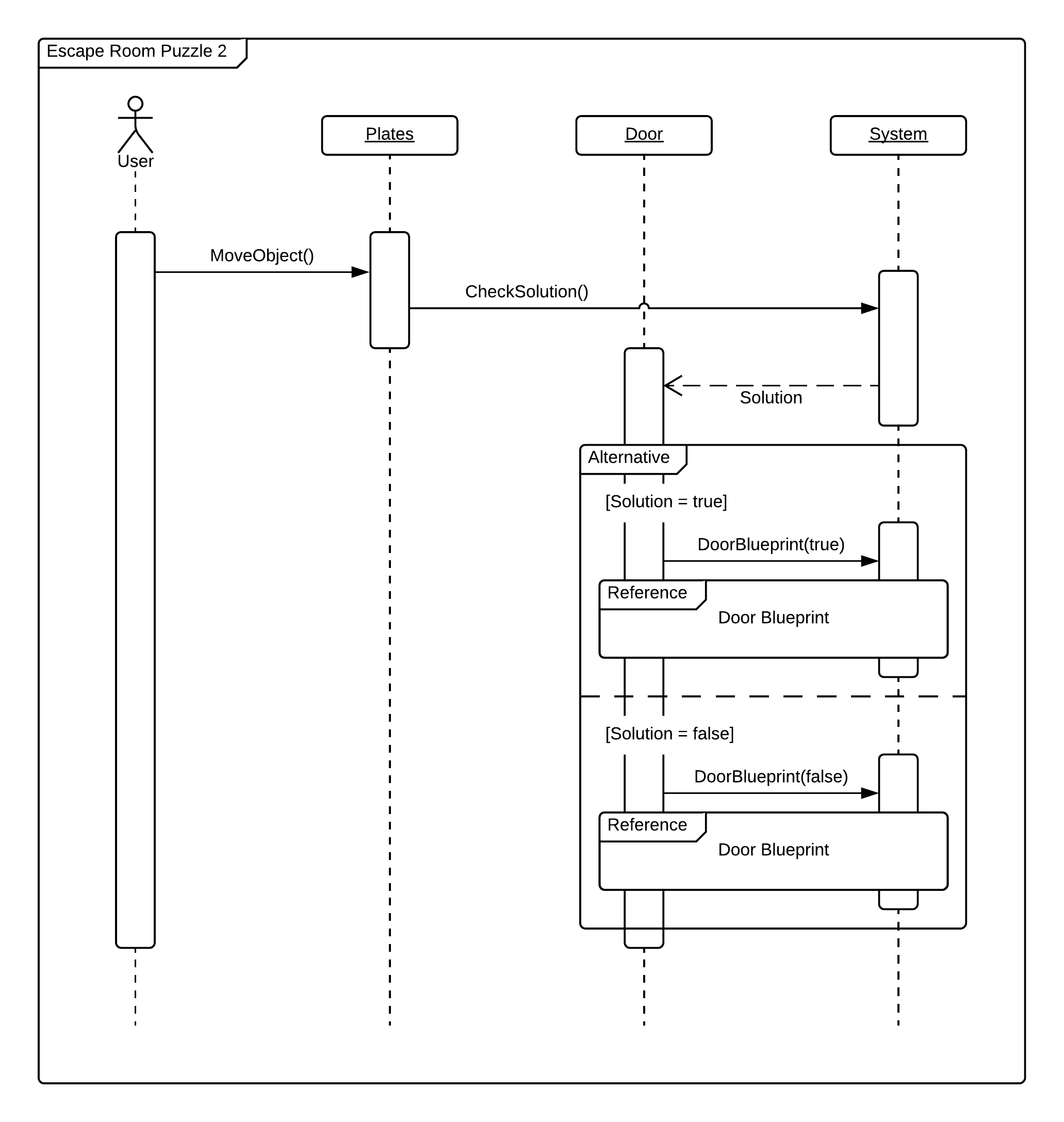
* Name: Escape Room Puzzle 2
* Actor: User
* Preconditions: The user is able to move apples around the room.
* Description <Flow of events>:

1. The user places one apple on each plate.
2. Check the users solution.
   1. If the bad apple is on one plate and only one good apple is on the other plate, then the puzzle is solved and the system is notified.
      1. The door opens, allowing the user to escape the room.
      2. If the user changes their solution, then the door closes.
   2. If the door remains closed, then the user must try a new solution.

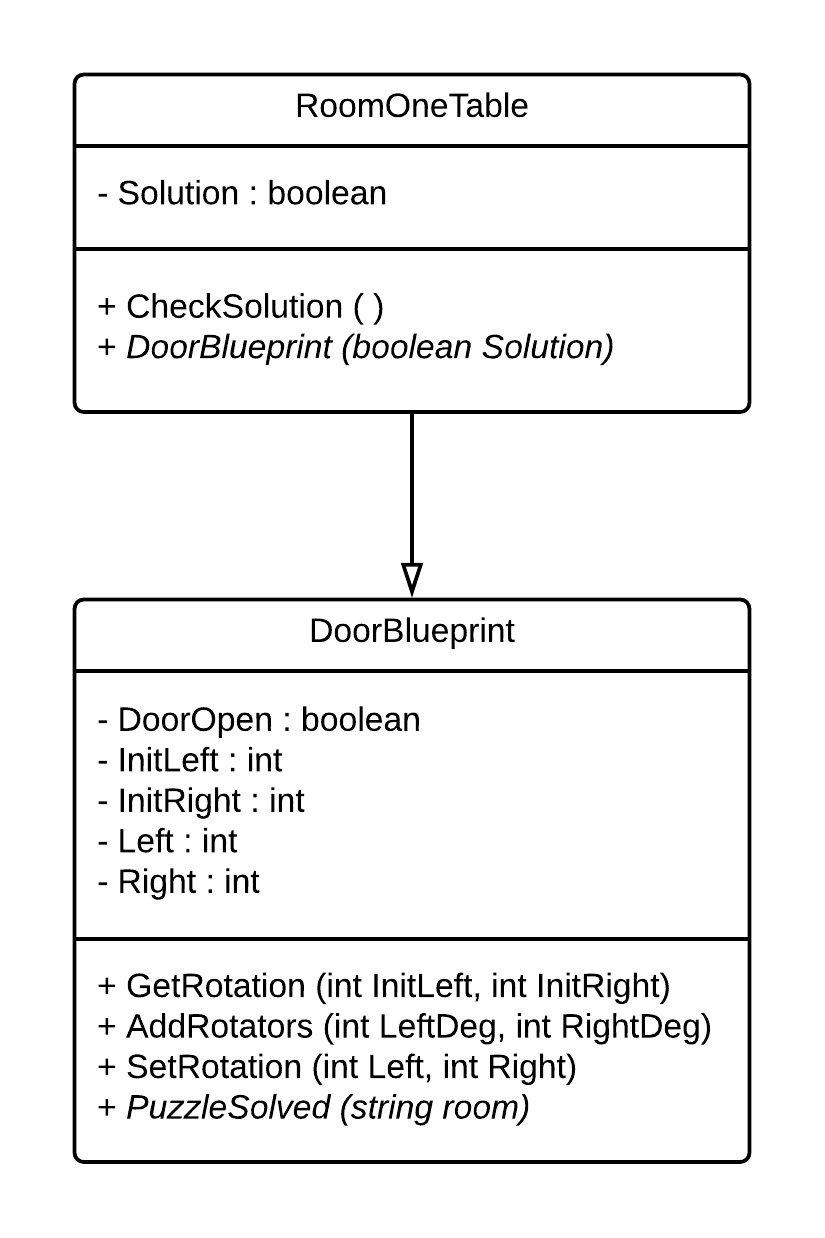
**Use Case Diagram**



**Sequence Diagram**



**Class Diagram**



**Unit Test**

* Test case ID: identify\_apple
* Description/Summary of Test: The plates identify the type of apple placed on it.
* Pre-condition: The user places an apple on the plates.
* Expected Results: Plates identify the type of apple.
* Actual Result: Plates identified the type of apple.
* Status (Fail/Pass): Pass
* Test case ID: check\_solution\_p2
* Description/Summary of Test: The puzzle is solved if and only if the bad apple is on one plate and only one good apple is on the other plate.
* Pre-condition: The user places an apple on each plate.
* Expected Results: A boolean data type.
* Actual Result: A boolean data type.
* Status (Fail/Pass): Pass
* Test case ID: notify\_system\_p2
* Description/Summary of Test: The system is notified when the puzzle is solved.
* Pre-condition: The user places the bad apple on one plate and one good apple on the other plate.
* Expected Results: System is notified.
* Actual Result: System is notified.
* Status (Fail/Pass): Pass
* Test case ID: door\_open\_p2
* Description/Summary of Test: The door opens when the puzzle is solved.
* Pre-condition: Puzzle is solved.
* Expected Results: Door opens.
* Actual Result: Door opened.
* Status (Fail/Pass): Pass
* Test case ID: door\_close\_p2
* Description/Summary of Test: The door closes when the puzzle is not solved.
* Pre-condition: Puzzle is not solved.
* Expected Results: Door closes.
* Actual Result: Door closed.
* Status (Fail/Pass): Pass

**Visual User Guide**



