# Sample Rubric for IP.1 (Max Score: 93)

This is the sample rubric for IP.1. I will accept any name the student comes up with as a name for this puzzle, as long as it is consistent. It could be “System”, “PlanarApp” or whatever. But if you use different words across different use cases, then you will lose a point for each subsequent use. For example, if you use ‘system’ in the first use case, but ‘PlanarApp’ in all other use cases, you will lose one point for each subsequent use case.

~~The same thing is true of whether student uses “player” or “person” or “user”~~ In class and in lectures I said “Identify concepts and use same term(s) in all use cases” and I also suggested “use the terms that the client uses” and I instructed grader to insist on “player” and only take off one point for the first one. Be consistent across all use cases, or lose one point for each inconsistent usage.

# StoryBoard (15 points)

Student must provide a graphic showing the essential layout of the application and provide sufficient documentation so you know how the player should interact with the application. There are several alternatives:

* Four buttons for each of the extending directions
* No buttons for extension, but instead the player hits the arrow keys or WASD
* Player clicks on a specific cell
* Reset button to reset puzzle

|  |  |
| --- | --- |
|  | Points:  (+3) Ability to choose three configurations  (+4) Ability to indicate direction in which to extend puzzle  (+2) Ability to reset puzzle  (+2) Where congratulatory message is to be shown  (+4) Where puzzle is to be drawn |

When extending a color, the player could either (a) choose an empty square and request to fill from neighbor (i.e., Pull); or (b) choose a colored square and choose to fill in a certain direction (i.e., Push). **EITHER IS VALID**.

Need to have some way to select a level (1, 2, or 3). Player chooses a direction (up, down, left, or right) to extend. Reset brings back to the beginning.

There are five use cases

* Choose Configuration
* Select Square
* Extend Color
* Reset PlanarPuzzle
* Solve PlanarPuzzle

# Grading Rubric: (13 points minimum but there are more points at times)

+1 Participating Actor: Initiated by player (+ 1)

+1 Entry Condition: Valid (+1)

+1 Exit Criteria: Valid (+1)

+1 Flow of Events (+1 for numbered)

1. Player requests Something Valid (+1 Player, +1 Request)
2. PlanarApp SomethingValid and refreshes Display (+1 App, +1 Valid, +1 Refresh)

Student gets +1 just for identifying the different parts (ParticipatingActor,EntryCondition,ExitCriteria,FlowOfEvents)

Note: Actor should always be PLAYER (and not USER or anything else) since in my original description that is the term I used.

Whether Entry Condition or Exit Condition (vv. For Criteria) it is ok.

Student gets **“+4”** points simply for making sure that each use case is described first by saying “USE CASE: VERB NOUN.”

## Use Case: Choose Configuration (13 points)

Participating Actor: Initiated by player

Entry Condition: None (+1)

Exit Criteria: Selected Configuration is chosen for player in initial state (+1)

Flow of Events:

1. Player chooses to play a selected configuration
2. PlanarApp selects configuration and updates display

## Use Case: Select Square (14 points)

Participating Actor: Initiated by player

Entry Condition: Puzzle has not been solved (+1)

Exit Criteria: Chosen square is selected (+1) but it cannot be an unused square (+1)

Flow of Events:

1. Player requests to select a square
2. PlanarApp selects square and updates display

## Use Case: Extend Color (18 points)

Participating Actor: Initiated by player

Entry Condition: An empty square (+1) is adjacent in a given direction (+1) to a square filled with a color (+1) that has the highest label number for that color (+1).

Exit Criteria: The formerly empty square is filled with neighboring color (+1) and its associated label is one greater than largest for that color (+2)

Flow of Events:

1. Player requests to extend color in selected direction
2. PlanarApp fills empty square and refreshes display

## Use Case: Reset PlanarPuzzle (13 points)

Participating Actor: Initiated by player

Entry Condition: None (+1)

Exit Criteria: Squares returned to initial state (+1)

Flow of Events:

1. Player chooses to reset planar puzzle
2. PlanarApp resets puzzle and updates display

## Use Case: Solve PlanarPuzzle (16 points)

Participating Actor: Initiated by player

Entry Condition: SAME AS EXTEND COLOR (+1)

Exit Criteria: All squares are filled (+1) other than any unused squares, and (+1) the highest numbered square for every color is adjacent to one base square in that color while (+1) the other base square is adjacent to the square labeled “1” in that color.

Flow of Events:

1. Player requests to fill the last empty square
2. PlanarApp fills square (+1) and shows congratulation message (+1) and refreshes display (+1)

# End notes

1. If student has exit criteria logic mistakenly placed within the flow of events then take of -1 point for each occurrence.
2. Many students offer incorrect entry/exit criteria for Solve PlanarPuzzle. The exit criteria must be “No empty squares remain”. The entry condition must be “one move away from winning.”
3. Graded solution will be pro-rated to be out of 100 points.