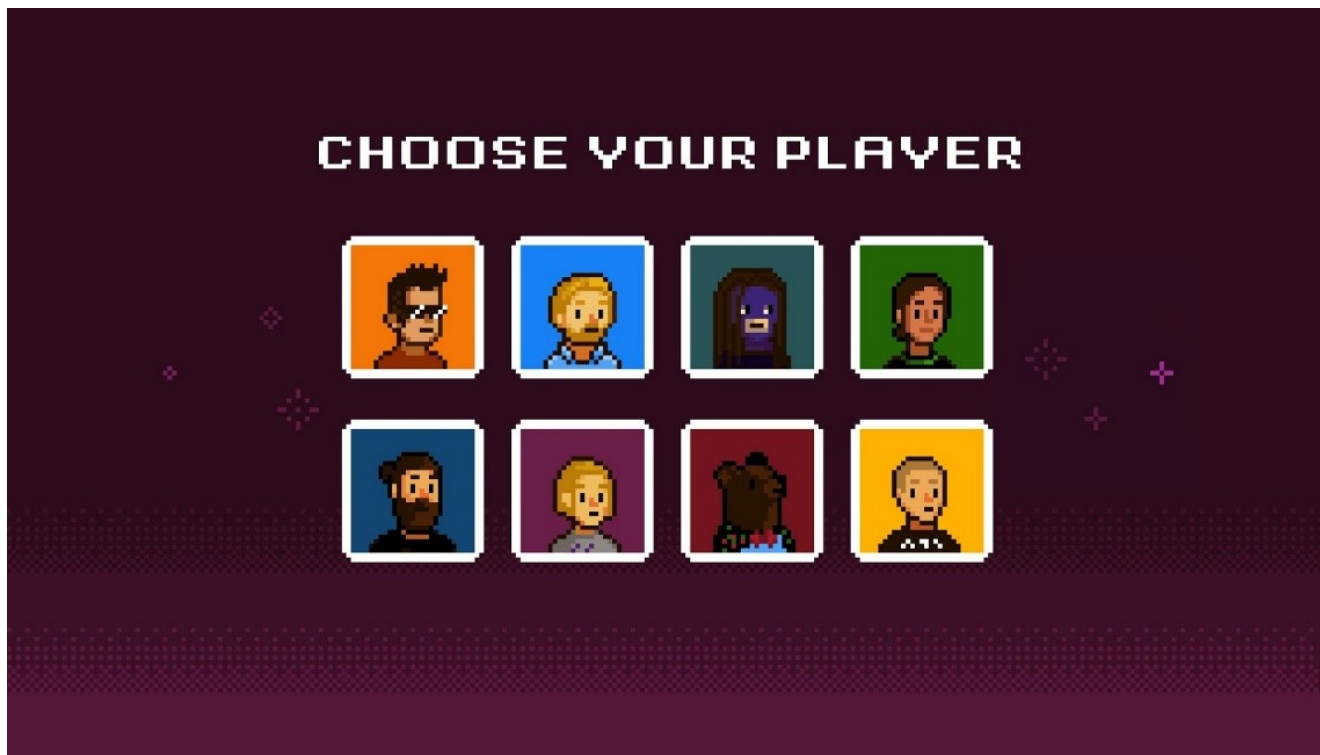


# Minimum Viable Product

*Our game's minimal viable product (i.e. a product that that will actually be beneficially usable by a customer) will consist of choosing a character, entering the player's name, an opening cutscene, a simple tutorial level, and a straightforward README file.*

## Choosing A Character

Before the game starts, the player will be prompted to pick one of the two characters allowed (i.e. either a woman or a man). This character will be the entity that the player controls with the keyboard keys and will use him or her to manipulate objects on the island for the rest of the game. Allowing game players to choose one of the two characters would satisfy our *Variability Within Gameplay* and *Personalization* user stories requirements since it allows a degree of personalization (i.e. the user can choose which character he or she wants) all while introducing an aspect of replayability within the game (e.g. once a user completes the game as a woman, he or she may want to replay the game again as a man instead). The following illustration provides an example for how the prompt will look, although for our game, users will only have two characters to choose from.



## Entering the Player's Name

Once the game player chooses one of the two characters, the game will prompt once again for user input, but this time for a name that will be assigned to the character. In other words, the game will ask the

user to input his or her name and assign it to the chosen character. It is important to note that the name inputted by the user cannot be too long since it will be displayed on a Java applet, so the input length will be constrained to 10 characters or less. Allowing game players to enter their name would (again) satisfy our *Personalization* user story requirement since displaying the user's name promotes a personalized gaming experience. The prompt to enter a name will be simple, somewhat like the following illustration.



### [Opening Cutscene](#)

Before users choose one of the two characters and enter their name, the game will display a small cut scene with an image similar to the following:



After the opening cutscene ends, the game will display a 2D, top-view isolated island (similar to the one below), thus indicating that the user survived the airplane crash. This opening cutscene will satisfy the *Storyline Consistency* user story requirement because it is feasible to comprehend that such a timeline of events could occur. The opening cutscene will also allow users to fully understand the origin of the storyline and will give them a reason to complete assigned tasks.



Furthermore, the game will also display a small dialogue that the game character is having with himself or herself. Such dialogue will be simple, similar to the following:

*“.... WHAT HAPPENED? THE PLANE MUST HAVE CRASHED! BUT IT LOOKS LIKE I'M THE ONLY SURVIVOR. I NEED TO GET OFF THIS ISLAND SOMEHOW!”*

### Simple Tutorial Level

After a user enters his or her name and chooses a game character, the game will prompt the user (using a set of instructions) to complete one simple task: move the character along the screen using the arrow keyboard keys to collect rocks. In essence, this tutorial level will be used to make sure that the user can control the character using the four designed arrow keyboard keys. Furthermore, because this is technically the game's “first level,” its complexity will be minimal and will focus more on the user's familiarity with controls. Allowing game players to complete a simple tutorial level would satisfy our *Game Complexity* and *Simple Controls* user stories requirements because starting the game with a simple task is essential (i.e. levels should start easy and get more difficult as they progress). Furthermore, by limiting the user to four keys for moving the character, the game will have an intuitive and simple control scheme. As the following animation demonstrates, the user will only need to move in all four directions to successfully complete the tutorial level (our game will not have the ‘slashing’ movements).



## Simple README File

Lastly, our game will have a simple and straightforward README file that will carefully outline what steps are needed to run and start the game. Such instructions could, for example, tell the user how to compile and run the source code and what outputs or inputs (if any) are expected. Including a simple set of instructions in a README file would satisfy our *Instructions and Accessibility* user story requirement because it would facilitate an easier and smoother gaming experience and would allow users to start playing the game without any additional help. A great example for how our README file will look like can be seen in the following illustration. It is critical that the instructions are simple to follow and provide useful comments on how to run the game.

```

README.txt
*****
* To Compile *
*****
make

*****
* To Run *
*****
java Pacman

*****
* Output *
*****
There is no output.  Just play the game.  Enjoy!

*****
* Turn into a Jar File *
*****

```

## Conclusion

Once our team completes these five tasks, we will officially have a minimum viable product (MVP) that has enough features to satisfy our initial set of user stories requirements. Furthermore, our MVP will allow us to gather constructive feedback for any future product development (e.g. moving the character would be easier using a different set of four keyboard keys, our storyline needs more detail, etc.).