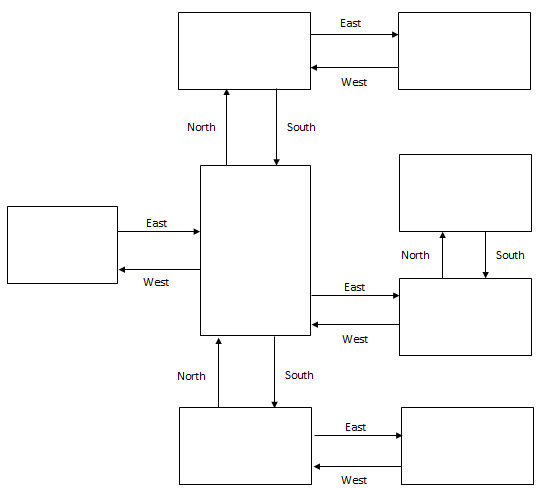
# IT 140 Design Document Template

Jeffrey VanMeter

## Storyboard (Description and Map)

There is a giant frog that has appeared on the patio and swallowed your dog whole. You refuse to let the frog get away with it. To get your dog back, the frog has to throw up and go away. You will need to create a special elixir to give to the frog. To do this you will need six items, a bucket from the garage to put your ingredients into, a spoon from the kitchen to mix the ingredients, a Poinsettia flower from the garden to make it’s stomach upset, olive oil from the pantry to make it’s tongue slippery, a jalapeño from the greenhouse to make him open his mouth, and a rabbit’s foot from the trophy room for good luck.



**Greenhouse**

Item: Jalapeño

**Garden**

Item: Poinsettia

**Pantry**

Item: Olive Oil

**Patio**

Frog!

**Foyer**

**Kitchen**

Item: Spoon

**Trophy Room**

Item: Rabbit foot

**Garage**

Item: Bucket

Pseudocode or Flowchart for Code to “Move Between Rooms”

End

Output “You got your dog back!”

Output “You got eaten by the frog!”

No

Yes

No

Yes

Is inventory full?

Output items in inventory

Yes

No

Output current room

Output items in inventory

Output current room

Update current room

Is current room equal to final room?

Output “Enter a valid command”

Input Direction

Is direction in current room

Start

## Pseudocode or Flowchart for Code to “Get an Item”

Start

Yes

Yes

No

Would you like to take the item?

No

Is there an item in current room?

Output “Your intuition told you to pick the item up”

Update inventory

Output “There are no items here”

End

* What **input** do you need from the player? How will you prompt the player for that input? How will you validate the input?

I only need to know which direction they wish to travel. The game will tell them which directions are valid by using a list containing the information needed.

* What should the program do if the player enters a valid direction? What **output** should result?

The room will change based on the direction they chose to travel. The output will be the updated room that may or may not contain an item based upon if they have been there already or not.

* What should the program do if the player enters an invalid direction? What **output** should result?

The player will stay in the same room until a valid input is entered. Outputs will be similar to enter North or South to change rooms.

* How will you **control the program flow** with decision branching and loops?

This also answers all of the bullet points from #5.

The player will be able to explore until they enter the final boss room. Depending on if they have fully explored to get all the items or not will determine if the player will win or lose when entering the final boss room. The only way to lose this game will be to not explore all the rooms. The player will be forced to pick up items upon entering new rooms.