

TIA3 - TERMINAL APP

Katrina Tomaszczyk

Due: 25 Sep 2022



WELCOME SPACE ADVENTURER

This presentation is my journey of building a Terminal Application using the Programming Language Python.

Requirements

- ◆ A walk-through of my Terminal application, its features and how it used
- ◆ A walk-through of the logic of my Terminal application and code
- ◆ A review of your development/build process including challenges, ethical issues, favourite parts, etc

CHOOSE YOUR OWN ADVENTURE

My Terminal Application

THE AGILE JOURNEY

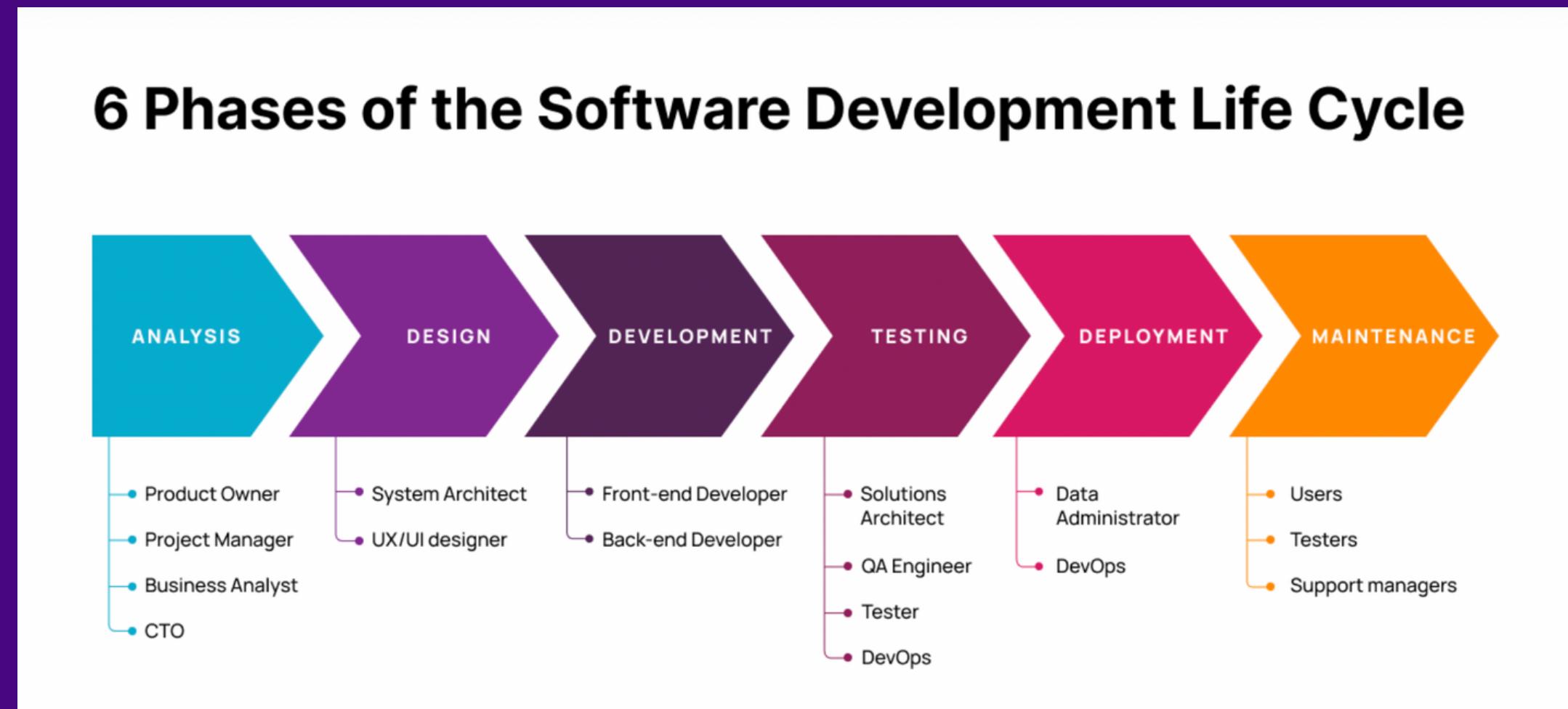


Image Source: BROCODERS, 2022

The foundation of building the Terminal App starts with the Agile Methodology. This is how I started my agile journey.

1. Analysis
2. Design
3. Development
4. Testing
5. Deployment
6. Maintenance

ANALYSIS

What do I want to build that will demonstrate my abilities as a developer and use a range of tools to achieve that.

Terminal Application

Brief

You are to design, implement and test a terminal application and throughout the process demonstrate that you are able to use a range of developer tools.

The mandatory requirements/constraints for your application are:

- accept user input in the form of a file or text input
- produce printed output or interact with the file system

High Level Requirements

Accept User Input

Product Printed output

Documentation

Presentation

Code

Brainstorm/ Analysis

What have I learnt so far?

Control Flow

if/elif/ else

Boolean values

Boolean Operators And/ Or / Not

Bonus Match Case

Functions

Scope

Return

Parameters

Variables

Data Types

Advanced Data Types

Lists

Tuples

Sets

Dictionary

Loops

While Loops

For Loops

Modules and Packages

Virtual Environments

Importing

Examples from the ED Exercises

Calculate Rocket Fuel

Calculate Circus Tickets

Embark on a mission (space)

weather/ temperature

Calculate cost of cinema

Subtract two numbers

Combine Strings

Factor Check

print Hello World

printing apples

defeat monsters in dungeon

Update warehouse database

sum to 100

sum to x

sum for 100

sum for x

Break Statement

Enumerate Function

number below 100

sum until exit

sum of evens challenge

search the list

Continue Statement

Nested Loops

Birthday Day of the Week

How Old Are You In Days?

10 Second Challenge

ANALYSIS

- ◆ Identify the High Level Requirements
- ◆ Review and Document what I have learnt so far. Apply that knowledge in my Terminal App.
- ◆ Brainstorm an idea that covers what I have learnt so far
- ◆ Create an Opportunity Assessment to confirm if it the right idea. To meet the needs for this assessment.

ANALYSIS

Opportunity Assessment: Even though this isn't a business strategy, I found it still helpful to focus on what and why I want to achieve.

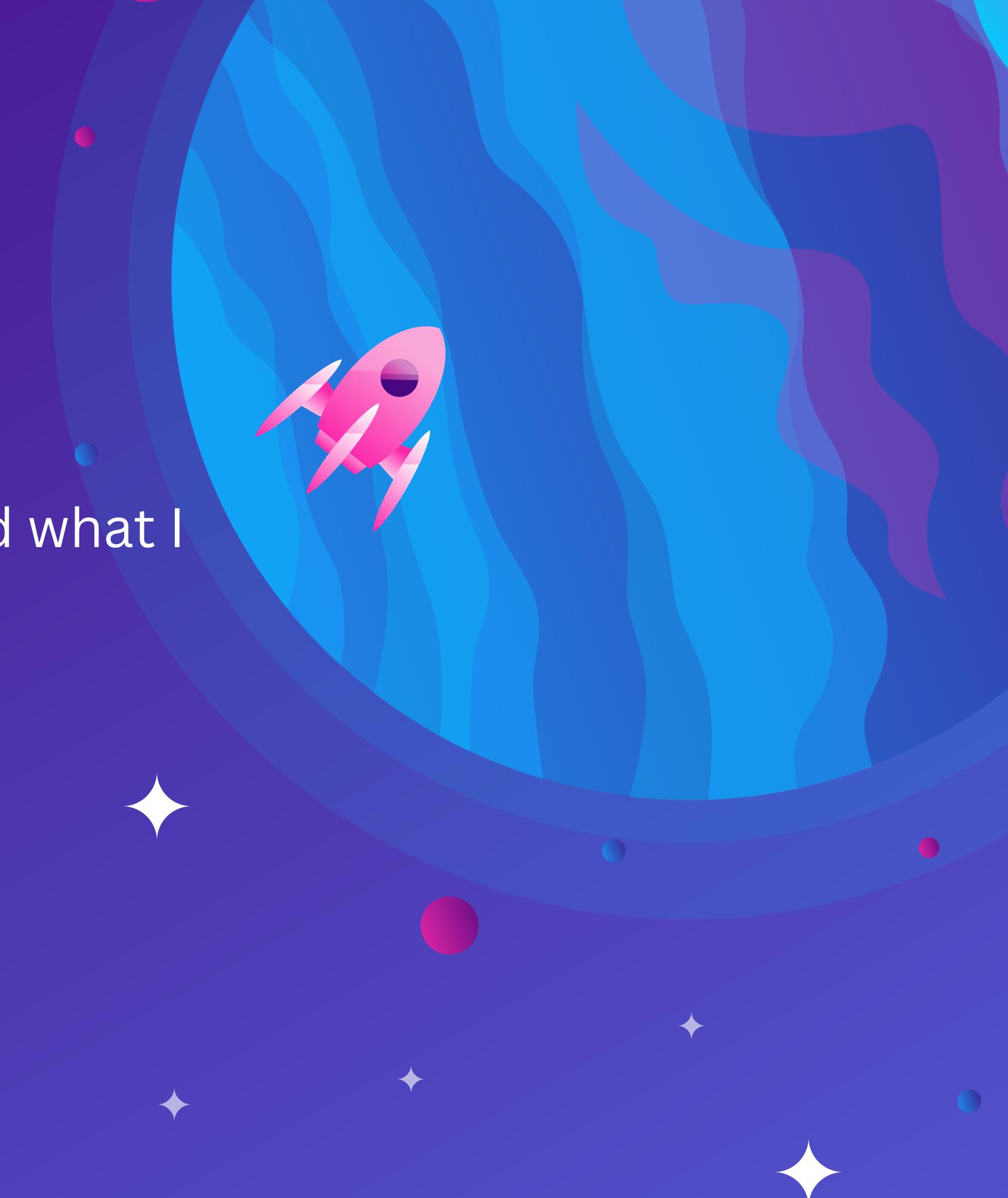


DESIGN

Next is to design the Terminal Application.

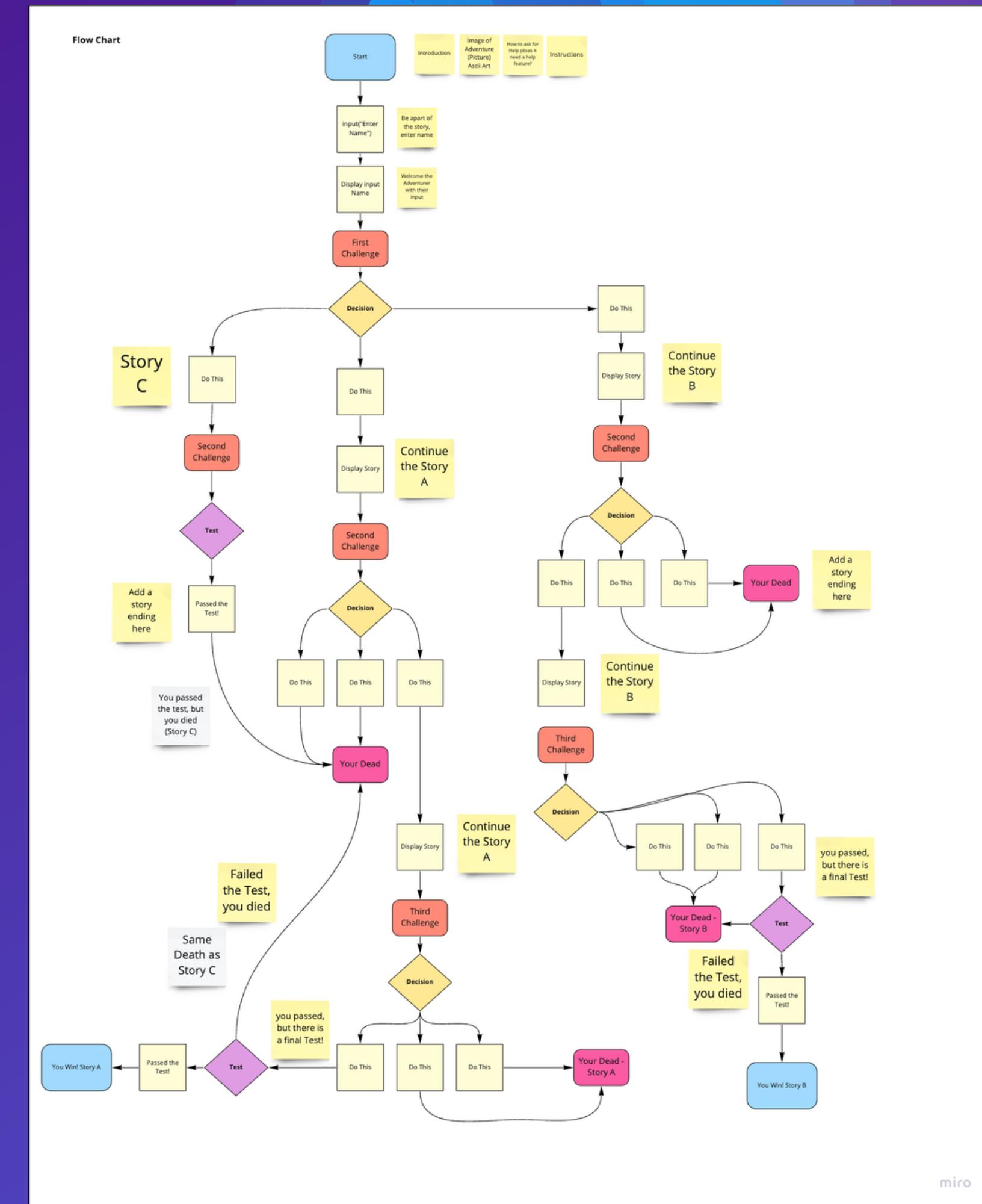
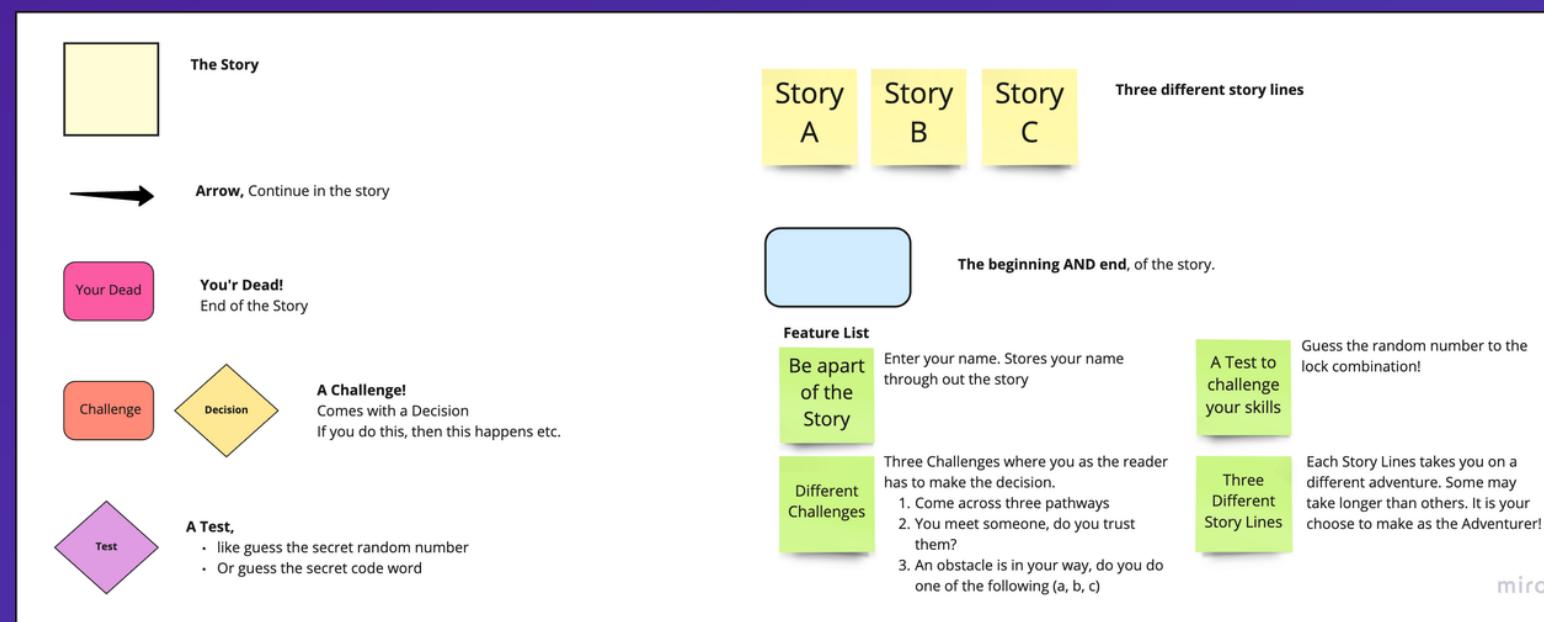
Now that I understand who my end user is and what I would like this application to achieve.

1. Flowchart
2. Design - PEP 8
3. Challenges (What are they and how many)
4. Identifying the Story line - The Theme
... (it's a Space Adventure)



DESIGN - FLOWCHART

Through the use of a flow chart, I am able to visually represent the data flow.



DESIGN

PEP 8 - Style
Guide for Python
Code



DESIGN - FEATURES

- ◆ Be apart of the story! Enter your name to see yourself right in the action!
- ◆ Different story lines. Which one will you choose? Or why not do of of them?
- ◆ Test your skills with some problem solving.



**WE HAVE THE IDEA!
WE KNOW THE
DETAILS...**

Whats Next?

Next few slides will go into the Journey of
Planning and Development Process



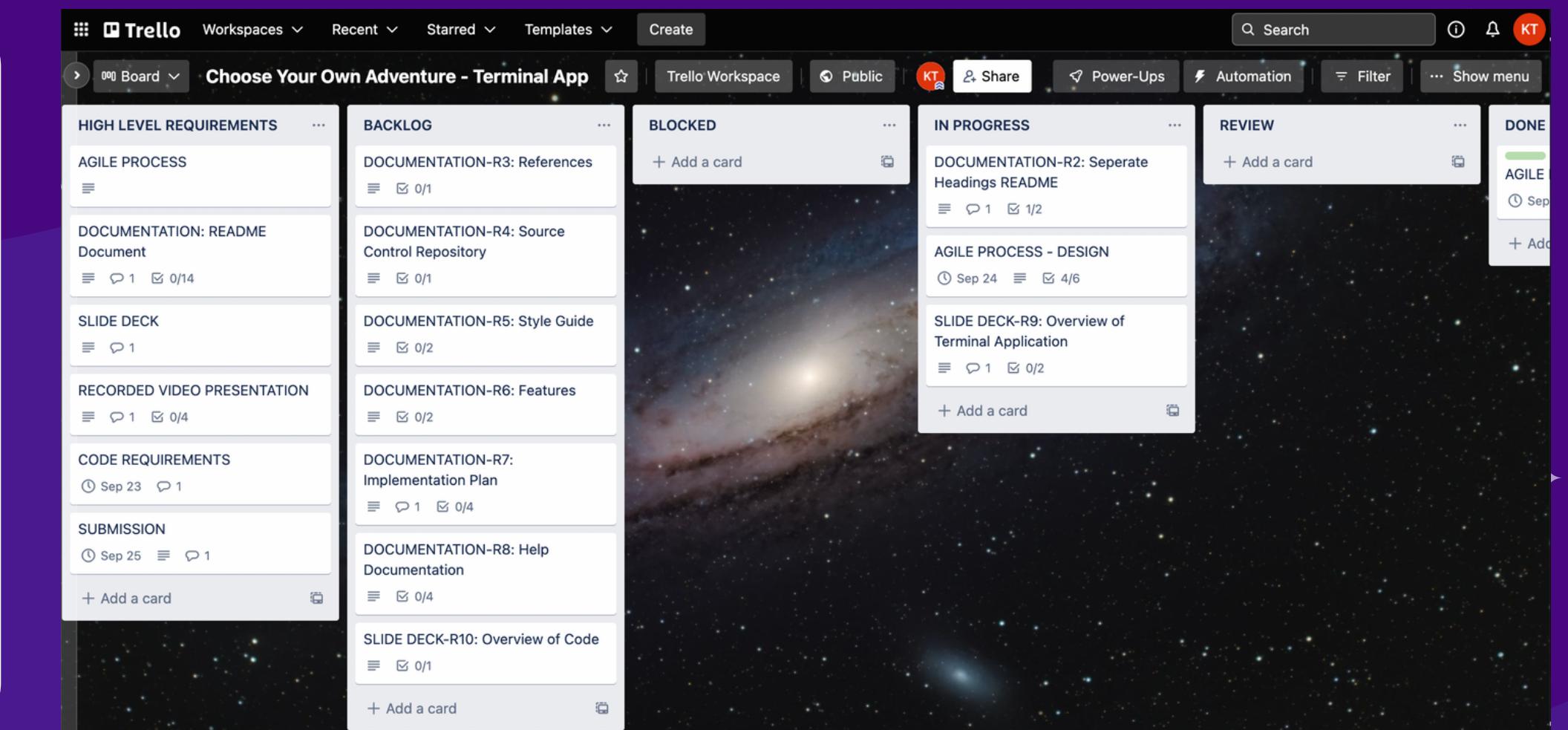
PLANNING - SET UP

SETTING UP A PROJECT

To succeed in development, a few housekeep items to keep myself on track.

- Set up a Trello Board (Backlog, Blocked, In Progress, Review, Done)

Initial set up has been captured but now that I know the project, I can expand on the development tickets



<https://trello.com/b/KSBbpWHq/choose-your-own-adventure-terminal-app>

PLANNING - SET UP

Tickets that required a definition of done.

- Title (easy to see from the Trello Board)
- Short description of the ticket
- Date Due (Keep myself accountable)
- Acceptance Criteria List (some have more/ others have less information)

The screenshot shows a Trello card for 'STORY C (Poisonous Plants) > door' in the 'End to End Testing (User Testing)' list. The card has a due date of 'today at 5:00 PM' marked as 'complete'. The description states: 'This card captures the requirements for STORY C. The room of poisonous plants.' The acceptance criteria list contains 13 items, all of which are checked:

- ✓ See Welcome Instructions
- ✓ Are you ready for your adventure (Yes) / (No)
- ✓ Yes—See Welcome Space Traveller!
- ✓ Enter in name, what shall I call you?
- ✓ First decision, choose (door) / (hide) / (run)
- ✓ Select (door) and see you are in a room of poisonous plants
- ✓ First Test—Select a bottle of antidote (Choose any number or letter)
- ✓ Enter in a number
- ✓ You grab a bottle with number on it and drink it, (You died!)
- ✓ This is the end of your Space Adventure.

On the right side of the card, there are several options for 'Add to card': Members, Labels, Checklist (which is checked), Dates, Attachment, Cover, and Custom Fields. Below these are sections for 'Power-Ups' (Add Power-Ups), 'Automation' (Add button), and 'Actions' (Move). A 'Start free trial' button is also visible.

<https://trello.com/b/KSBbpWHq/choose-your-own-adventure-terminal-app>

DEVELOPMENT

Following the FlowChart											
	Create Skeleton with comments	Create Inputs	Start Story	Add Traveller Name	First Decision (run, hide, door)	Story C (door) - Challenge	Story A (run)	Story A (dec_two)	Story B (dec_three)	Story B (hide)	Story B Challenge
Title	Are you ready to start	Create Welcome Message	Create input/variable	if, elif, else	Create challenge, guess any number function	Create a decision_two (dec_two)	Create Story - pick it up (Trible)	Continue Story, feed it?	Decision (Y/N) input	Enter success, you win!	Restart game function
Instructions	While True > Continue, Else: END GAME	Create instructions	Create Story to add Traveller Name	validate "anything else in input"	Build with error validation	Y/No input	Yes/ No decision - input	Yes/ No decision - input	Select no, add story, you die	3 attempts fail (you die)	You died function
Story A	clear terminal/window	Add ascii art			Create story to match challenge	If/ elif, else	If/ elif, else	If/ elif, else	If yes, see challenge, guess the password	Restart game function	finished story function
Story B	validate "no" flow	return here - start again main function			validate "anything else in input"	validate "anything else in input"	validate traveller name to story	validate "anything else in input"	validate "anything else in input"	validate "anything else in input"	
Story C	validate "anything else in input"				validate "anything else in input"						

STORY MAPPING

Break down the high level and tasks associated with the Development.

DEVELOPMENT PLAN

Identify the tasks required and create Trello Tickets with acceptance criteria. (can't capture everything at the time)

INTERATIVE TESTING/ REFACTORING

Continual Testing. Test Driven Development. Write code so that it can have a test or handle exceptions.



```
src > main.py > ...
1  # Choose Your Own Adventure Application
2
3  # Imported Modules
4  from functions import guess_number, password_guess, random_container
5  import random
6  from sys import argv
7  import clearing
8  from story_line import instructions
9  from story_line import main_story
10 from ending import you_died
11 from ending import restart_game
12 from ending import finished_story
13 from pyfiglet import Figlet
14 from ascii import ascii_title
15
16 def main():
17
18     # Instructions
19     instructions()
20
21     # Are you ready to play game - Control Flow (77 Characters)
22     start = input("Are you ready for your journey? (Yes) / (No):\n").lower().strip()
23
24     while True:
25         if start == "y" or start == "yes":
26             print("Lets Begin!")
27             clearing.clear()
28
29             ascii_title()
30             main_story()
31
32             # Ask for Traveller name - Be in the story
33             traveller_name = input("What shall I call you?\n").strip()
34             clearing.clear()
35             print(f"\nIt is hard to say, but I will call you {traveller_name}!")
36             print(f"\nThe alien creature points and says, '{traveller_name}, we must go.'")
37             print("Except you hear something coming...\n")
38
```

```
src > functions.py > ...
1  import random
2  from ending import restart_game, you_died
3
4  # Function to continually ask the Traveler for a number. Story Line C.
5  def guess_number():
6      number = None
7      while number is None:
8          guess = input("Quick!, which bottle do you choose? Guess ANY number or letter: \n")
9          try:
10              number = int(guess)
11          except ValueError:
12              print("\nThis one looks weird, try a different bottle with a number?\n")
13          print(f"\nYou grab the bottle with the label, {guess} on it and drink it!")
14          print("On no... you don't feel so good...")
15
16
17  # Function to pick a random number. Story line A - Random container to feed
18  def random_container():
19      while True:
20          try:
21              number = random.randrange(0,11)
22              random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
23              while random_guess != number:
24                  if random_guess < number:
25                      print("That one doesn't look right, pick a higher number")
26                      random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
27                  elif random_guess not in range(0, 11):
28                      print("Try a number between 1 and 10.")
29                      random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
30                  else:
31                      print("Doesn't look right, pick a lower number.")
32                      random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
33              print("\nYou open the container and throw the food at it. Look! Its eating!")
34              break
35          except ValueError:
36              print("That doesn't look right, try entering a number between 1 and 10.")
37
38
```

```
src > ending.py > ...
1  # Ending functions
2
3  # Function when the traveller dies in the story
4  def you_died():
5      print("\nYou Died!")
5      print("That is the end of your Space Adventure.")
5
6
7  # Function to finish the story
8  def finished_story():
9      print("\nTHE END")
9      print("That is the end of your Space Adventure.")
9
10
11  # Function to restart the game
12  def restart_game():
13      restart = input("\nWould you like to play again? Type: (Yes) / (No)\n").lower()
14      if restart == 'yes' or restart == 'y':
15          print("Lets Begin!")
16      elif restart == 'no' or restart == 'n':
17          print("Sorry to see you go.")
18          exit()
19      else:
20          print("Maybe next time.")
21          exit()
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
```

PACKAGES

Importing the correct packages

MODULES

Separate functionality

DRY

Separate code, keep in line with DRY - Don't Repeat Yourself. Create Functions

Welcome Space Traveller!
I have brought you here as we are in grave danger.
I need your help!
You look around....
And realise you have been transported to the distant future.
The alien creature is looking at you and asks.
What shall I call you?

Welcome to your Space Adventure



Follow me, I found something I want to show you.

The alien creature shows you a small metallic box.
It has familiar etchings on it and a keypad.
What do you think the password could be?
You see the first letter as 'S' and the last letter as 'E'

Enter your guess to see if it works:

█

You push the door to reveal a room full of poisonous plants!
You accidentally brushed against one and start to feel sick.
You see bottles of gooey liquid, one must be antidote!

Quick!, which bottle do you choose? Guess ANY number or letter:
sdfgh

This one looks weird, try a different bottle with a number?

Quick!, which bottle do you choose? Guess ANY number or letter:
8

You grab the bottle with the label, 8 on it and drink it!
Oh no... you don't feel so good...
Last thing you hear is, Katrina, that wasn't antidote!

You Died!
That is the end of your Space Adventure.

Would you like to play again? Type: (Yes) / (No)

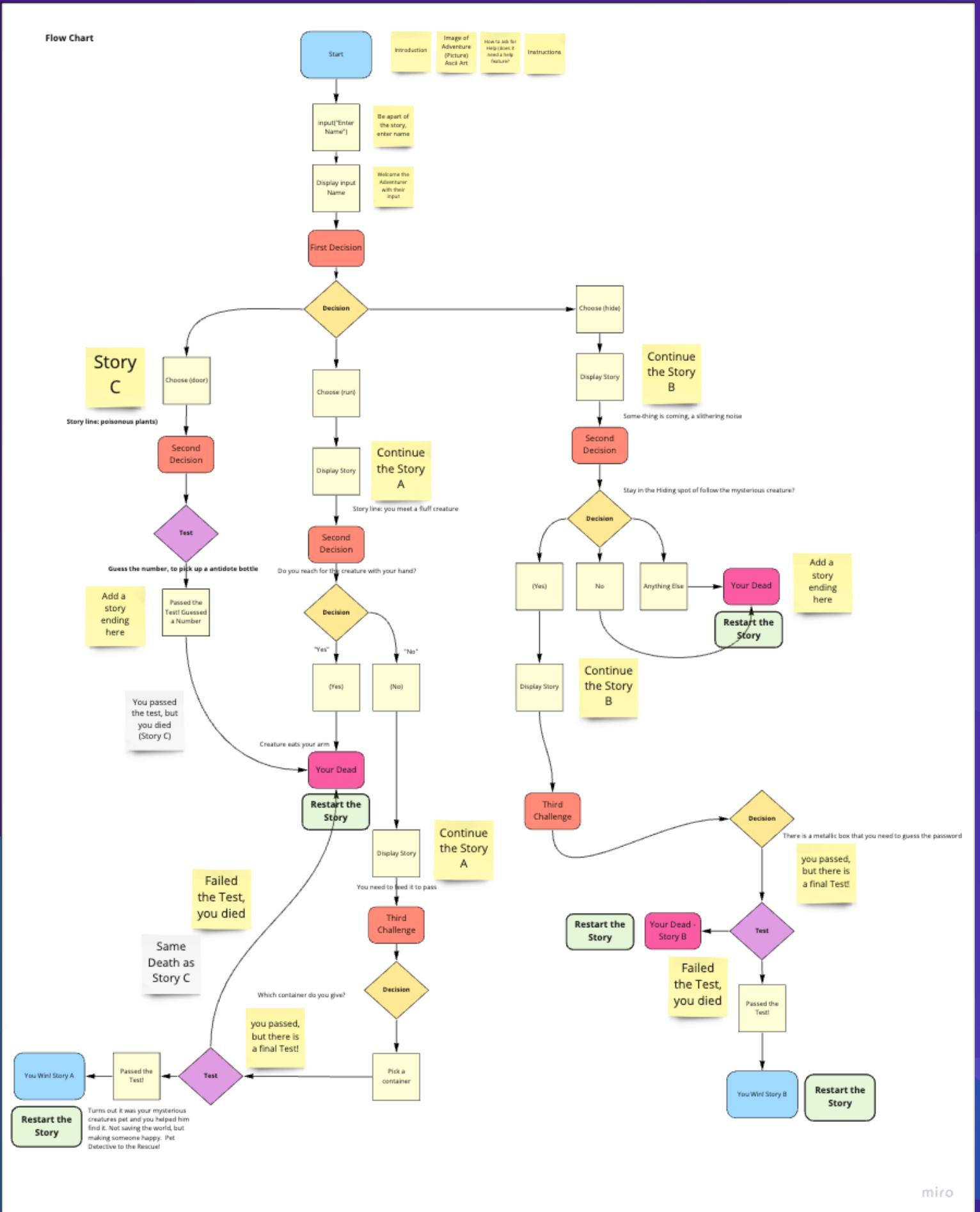
Experience Space! ✨

- Enjoy the story
- Enter in your name to be apart of the story
- If you die, start again
- Three different Adventures
- Try the challenges!



Flow Chart - Final

- Continually referenced the flow chart
 - Updated the flow chart to add the decisions and challenges
 - Add notes to help follow and build the testing plan. Did the flow chart meet the Acceptance Criteria



TESTING

	A	B	C	D	E	F	G
1	Test Case Title	Testing Instructions	Expected Results	Pass/Fail	Error Handling	PEP 8	Comment/Note
2	User sees instructions (print)	Instructions function works, instructions are displayed. Character lines max is 72 as per PEP 8 Guide.	As Testing Instructions	Y	N/A	Y	
3	Start Game, are you ready (input)	Start Game. Enter in (Yes) / (No), on a new line. Only "yes" or "y" will Start the Game	As Testing Instructions	Y	Use of module lower() and strip(). If statement handles "yes" and "y". Anything else goes to "Maybe next time"	Y	
4	(negative path)	Are you ready, enter "no", see "Thats too bad, maybe next time."	As Testing Instructions	Y	elif statement to capture response "no". Break.	Y	Not asked to restart the game
5	(negative path)	Enter in anything else. A number, missing text and see "Are you sure you entered the right phrase?". Prompt to restart the game.	As Testing Instructions	Y	else statement to capture anything else that occurs.	Y	
6	User sees Welcome message (print)	Welcome message from the client displays with ascii art (Space Adventure) on two lines to match the Terminal window and PEP 8 Style guide.	As Testing Instructions	Y	N/A	Y	ascii_title() function displays as per testing instructions
7	Capture Traveller Name (input)	Enter in a name and see your name in the story line	As Testing Instructions	Y	Use of strip() method to remove any spaces.	Y	Any name can be entered in this field, no need to display error handling for number or capital letters
8		See print line with traveller name in it	As Testing Instructions	Y	Working as expected - no silent errors / No error handling	Y	
9	Decision one (input) / Story A/ B and C	See 'Decision one' (door) / (hide) or (run) and then input your decision	As Testing Instructions	Y	Leads to next story. Either Story A, Story B or Story C (Poisionous plants)	Y	

TESTING PLAN

```
src > 🐍 functions.py > ...
1  import random
2  from ending import restart_game, you_died
3
4  # Function to continually ask the Traveler for a number. Story Line C.
5  def guess_number():
6      number = None
7      while number is None:
8          guess = input("Quick!, which bottle do you choose? Guess ANY number or letter: \n")
9          try:
10              number = int(guess)
11          except ValueError:
12              print("\nThis one looks weird, try a different bottle with a number?\n")
13  print(f"\nYou grab the bottle with the label, {guess} on it and drink it!")
14  print("On no... you don't feel so good...")
15
16
17 # Function to pick a random number. Story line A – Random container to feed
18 def random_container():
19     while True:
20         try:
21             number = random.randrange(0,11)
22             random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
23             while random_guess != number:
24                 if random_guess < number:
25                     print("That one doesn't look right, pick a higher number")
26                     random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
27                 elif random_guess not in range(0, 11):
28                     print("Try a number between 1 and 10.")
29                     random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
30                 else:
31                     print("Doesn't look right, pick a lower number.")
32                     random_guess = int(input("Pick a container between 1 and 10. Type your number:\n"))
33             print("\nYou open the container and throw the food at it. Look! Its eating!")
34             break
35         except ValueError:
36             print("That doesn't look right, try entering a number between 1 and 10.")
37
38
```

- Iterative Testing
- Planning each feature and function to gracefully handle errors and validation
- if/elif/ else statements capturing all scenarios
- Manual Testing to match Acceptance Tests as per the testing plan spreadsheet.
- Testing Happy/Sad flows. Entering in rogue fields, incorrect data etc.

THANK YOU

I hope you enjoyed this Space Adventure!

