

**De La Salle University- Manila**

**Gokongwei College of Engineering**

LBYCPA1

Programming Logic and Design Laboratory

Project Proposal

“Code Crawler: An Educational Python Based Game”

Gill Whenzel Quijano

Jireh G. Robellon

Matthew Ceazar P. Talicol

**Project Description**

The Code Crawler project is a Python-based instructional game that seeks to give users a fun method to learn Python programming. Players will be led through a narrative story that has numerous instances of wrong code, and their companion "Dawn" will offer advice on how to fix the mistakes.

* Problem to be solved:

Finding a fun way to learn a programming language can be difficult for beginners, who may find it intimidating. This project intends to address this issue by creating a fun and instructive game that will aid players in learning Python programming.

* Technical Objectives:

1. Create a game engine for Python that can control player movement and communicate with other game elements.
2. Create and put into play the game's plot and levels, which will offer players varying difficulties.
3. Build "Dawn," an interactive companion that will offer advice and guidance throughout the game.
4. Make a database of typical Python programming faults, and then design in-game challenges that let players practice resolving them.
5. Make the user interface simple to use and navigate so that players may concentrate on the instructional aspects of the game.

* Implementation of the Project:

These stages will be used to carry out the Code Crawler project:

1. The project's needs and scope, including the game's gameplay, plot, and technical goals, should be gathered.
2. Create wireframes and the user interface for the game to plan its layout.
3. Create the game engine, and put the player interaction and movement into practice.
4. Construct the plot and game levels, including the companion character "Dawn" and interactions with improper code.
5. Provide a database of common Python programming faults and in-game scenarios so that players may practice repairing them.
6. Create the user interface for the game, which should include menus, instructions, and game controls.
7. Test the game and make any necessary technical and gameplay adjustments.
8. Publically release the game and get user feedback for the next updates.

These stages will enable the Code Crawler project to provide an entertaining and instructive Python-based game that aids users in learning programming.

**IPO**

Input: JuPyter Notebook, The Laboratory Manual, Google Documents

Process: The backbone of the program will be organized first, making a template encounter for both Dawn and an Enemy Code to then reuse and randomize throughout the game, next will be the implementation of the information to teach the user, as well as the incorrect code of the enemy. Finally, the supplementary dialogue for Dawn will be implemented afterwards, as well as attempting to make the game fully randomized

Output: A Program that can assist new users to learn the basics of python using a game taught format.

**Methodology**

The process for developing Code Crawler is as follows:

1. Identify the Learning Objectives:

* The first step in creating Code Crawler is to identify the learning objectives. The game should teach players basic Python concepts such as syntax, data types, loops, conditional statements, and functions. Additionally, the game should focus on common coding mistakes, such as incorrect indentation, syntax errors, and logic errors.

1. Develop a Storyline:

* The game's storyline should be designed to provide a compelling narrative that keeps the player engaged. The storyline should be an ongoing linear story that players can follow as they move through the game. The story should also be designed to introduce players to coding concepts and common coding mistakes in a fun and engaging way.

1. Create the Game Levels:

* The game levels should be designed to provide players with a range of coding challenges. Each level should focus on a specific coding concept, such as data types or loops. The levels should gradually increase in difficulty, allowing players to build on their knowledge as they progress through the game. Each level should also introduce players to common coding mistakes and provide tips on how to avoid them.

1. Develop the User Interface:

* The user interface should be designed to be user-friendly and easy to navigate. The interface should include a dashboard that displays the player's progress and allows them to access different levels of the game. The interface should also provide players with access to tips and resources that can help them solve coding challenges.

1. Incorporate Dawn:

* Dawn should be incorporated into the game as the player's guiding companion. She should provide tips and guidance to the player as they move through the game. She should also be designed to provide a fun and engaging personality that players will enjoy interacting with.

1. Testing and Feedback:

* Testing is an essential part of creating a Code Crawler. The game should be tested extensively to ensure that it is free of bugs and glitches. Feedback from users should also be collected to identify any areas where the game could be improved. This feedback can be used to make adjustments and improvements to the game over time.

To develop Code Crawler, several Python concepts will be used. These concepts will be introduced to players gradually through the game levels, building on their knowledge and understanding as they progress. Some of the key concepts that will be used include:

1. Syntax:

* Python syntax is the set of rules that define how code should be written in the language. Players will learn about Python's syntax rules, including proper indentation, the use of colons and parentheses, and the syntax for defining variables and functions.

1. Data Types:

* Python has several data types, including integers, floats, strings, booleans, and lists. Players will learn about these data types and how to use them in their code.

1. Conditional Statements:

* Conditional statements are used to make decisions in code based on certain conditions. Players will learn how to use if-else statements to create branching logic in their code.

1. Loops:

* Loops are used to repeat code blocks multiple times. Players will learn about for and while loops and how to use them to automate repetitive tasks.

1. Functions:

* Functions are reusable code blocks that can be called from other parts of the code. Players will learn how to define and call functions and how to pass parameters to functions.

1. Debugging:

* Debugging is the process of finding and fixing errors in code. Players will learn how to use print statements and other debugging techniques to identify and fix common coding mistakes.

By introducing these concepts gradually and in the context of the game's challenges, players will learn how to use Python effectively and avoid common coding mistakes. Code Crawler aims to provide a fun and engaging way to learn Python, making it accessible to people of all ages and backgrounds.

Code Crawler is an educational Python-based game that provides users with a fun and interactive way to learn coding concepts and avoid common coding mistakes. The game is built around an ongoing linear story and a guiding companion, Dawn, who helps players navigate coding challenges and provides useful tips along the way. The methodology outlined in this essay provides a framework for creating Code Crawler, and the game's success will depend on its ability to engage users and provide them with valuable knowledge and skills in Python programming.

**Schedule of Activities**

****

Provide a timetable or Gantt chart of your deliverables. Indicate who does what and when the deliverables will be accomplished.

**References**

Correia, A. (2017, September 12). How to Create a Python Text-Based Game. Instructables.

<https://www.instructables.com/How-to-Create-a-Python-Text-Based-Game/>

Jaiswal, D. (2021, August 13). Text-Based Adventure Game in Python. TheCoderPedia.

<https://www.thecoderpedia.com/blog/text-based-adventure-game-in-python/>

Melvin, J. (2020, May 5). 5 open source Python games to learn coding. Opensource.com.

<https://opensource.com/article/20/5/python-games>

Raj, S. (2021, July 7). Text Based Adventure Game in Python – An Ultimate Guide for

Beginners. AskPython. <https://www.askpython.com/python/text-based-adventure-game>

Singh, A. (2021, February 22). How to Create a Text Adventure Game in Python. MakeUseOf.

https://www.makeuseof.com/python-text-adventure-game-create/