Code Nation - Python Game Project

Knight’s End

Group 2

Adam Butler

Mufazzil Chowdhury

Jamie Reid

Game Plot

In *Knight's End*, you play as a former Knight, who has woken up from unconsciousness in the the home of Ealdred. Ealdred came across your body and rescued you back to health. He explains that a witch, Morgath, has placed a curse over the town of Clayvault. You reveal that your brother (also a Knight), is missing.

Through this game, you come across various scenarios where you must fight using a battle system, and push through to Morgath's lair, in the hopes of saving your brother and lifting the curse on the town of Clayvault.

Characters

Protagonist: You play as a former knight, whose brother is missing. The player inputs their chosen name at the start of the game and is referred to this throughout.

Ealdred (NPC): An elderly apothecary, who found you unconscious in the woods and brought you back to his home to nurse you back to health. Ealdred is an elderly apothecary with a gentle demeanor and a seemingly wealthy knowledge of herbs and potions. He has a scruffy, greying beard and kind, blue eyes that twinkle with wisdom.

Matilda (NPC): A woman from Clayvault, who runs the local orphanage. She is a weary and cautious figure, with lines of worry etched deeply into her face. Her once-bright green eyes are now dulled by years of hardship and mistrust.

Maurin (NPC): The local butcher. a hulking figure with a grim demeanour and a perpetual scowl turns. His once-burly frame is now gaunt and his skin pallid, giving him a ghostly appearance.

Wild Dog (Enemy): This dog patrols the Cemetery, terrorising the local townsfolk. The dog is a pitiful sight, with matted fur and a gaunt frame. Its eyes are wild and bloodshot. Despite its weakened state, the dog is fiercely aggressive, driven by hunger and desperation. Its once noble features are now twisted and scarred, a shadow of the loyal companion it once was. It is said that it's owner passed and is buried there.

Bandit (Enemy): a grizzled veteran with a patch over one eye and a cruel smile. He wields a rusty sword that has seen better days.

Morgath (Enemy): Morgath is a powerful sorceress, shrouded in dark robes that seem to blend into the shadows. Her eyes burn with an intense, malevolent light, and her long, black hair flows like a river of ink. She wields a staff carved with ancient runes and crackling with dark energy. Morgath's voice is both mesmerizing and terrifying, capable of ensnaring the minds of those who listen. She is cunning, ruthless, and will stop at nothing to achieve her dark ambitions.

Endings

In this game there are 4 possible endings:

1) **Death opportunity:** the dog can kill you.
2) **Death opportunity:** the bandit can kill you.
3) **Death opportunity:** Morgath can kill you.
4) **Win:** Upon successfully defeating all enemies, you complete the game. In this ending, the curse on the town of Clayvault is lifted, however it is bittersweet, because the player's brother has been killed by Morgath.

How did we come to decide on this game idea?

As a group, we had a brainstorming session, where we noted down some ideas on Trello. Adam mentioned that he grew up playing some text-based or point & click adventure games, so spoke about how they are often well-suited to a fantasy style, for example Pirates, Medieval, Dungeons & Dragons, etc. The group then felt that a medieval setting would be the most interesting one to develop, as could feature fantasy characters such as Witches, and that is where the idea to have a curse that needed lifting came from.

At first, we were going to make this game one that focused on collecting and using items in your inventory (e.g. creating a potion by combining items discovered in the world) , very much in the style of the original point and click adventure games.

Mufazzil suggested having boss battles, and whilst this felt a bit advanced for a text-based game, Jamie reassured the group that this could be achieved using python code. We decided to go down this route, as doing a collect-a-thon game may have felt a little boring. This battle system would utilise code that would be more of a stretch challenge.

In order to improve the depth of the game, a combat system was added, which lead to a more dark fantasy aesthetic. This also brings the older style of text-based games to a more modern RPG gameplay (e.g. turn-based games like the Final Fantasy series).

Key Features

Battle System

To make this game feel a little more exciting, we developed a battle system. Each character has HP (Health Points) and Damage stats, and during fights the player takes it in turn to decide what command they'd like to do: Attack, Block & Dodge. Using a random number generator, we were able to assign values that would allow the player to predict the enemy's actions, and thus choose their command wisely?

Once an attack is made, the health is adjusted. Once one of the characters health points reach 0, the fight is either lost or won.

Goals as a Team

To assign different areas of coding and creative aspects to each member and therefore work effectively in one code-sheet – this is to prevent the overwriting of code that an individual may be working on.

To speak openly about progress and any parts where an individual needed help. And to flag promptly where there are areas that need fixing.

Integrating story into the code.

To meet the above goals, we agreed regular stand-ups, where we would come back together. We stayed online in our desk calls, so that we could un-mute at any point and seek guidance from the group.

How We Planned the Game

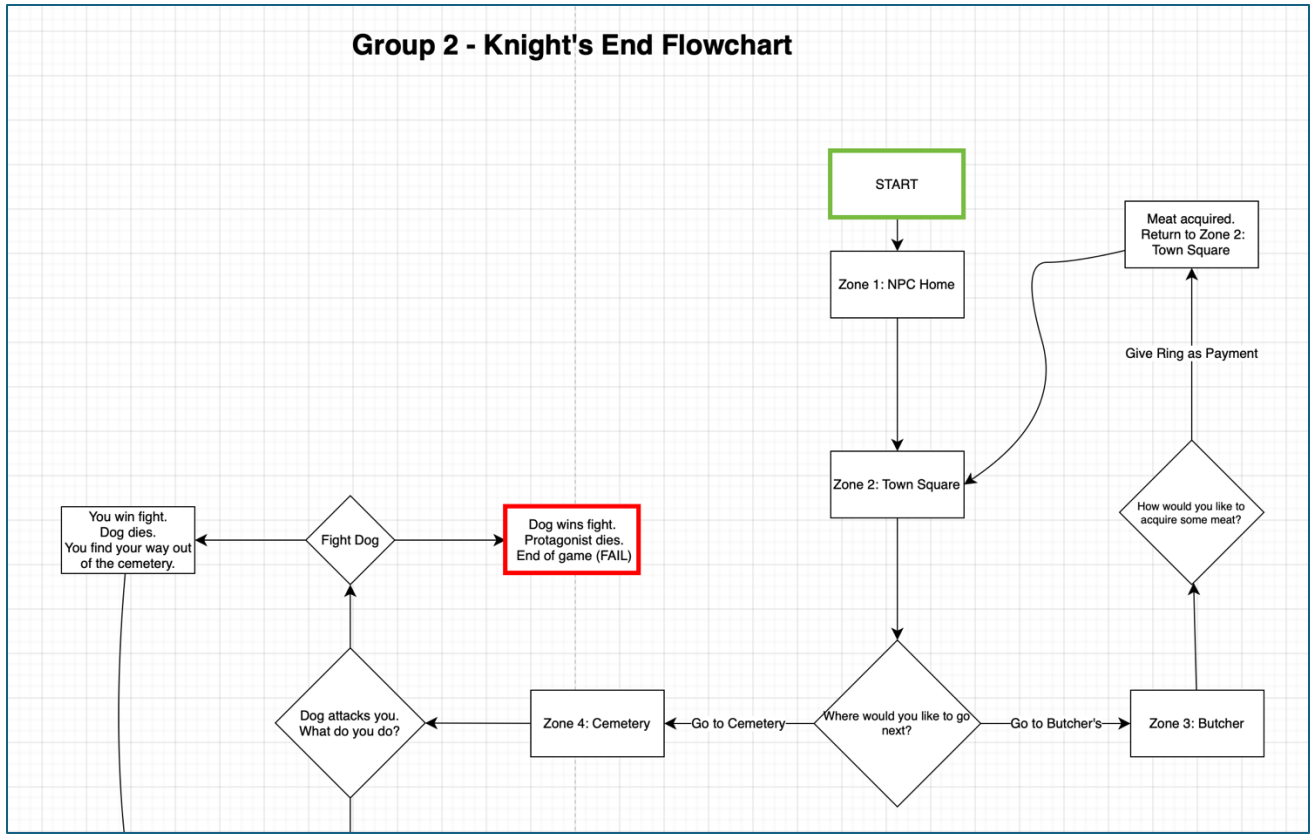
At the start of our process, we created a shared board on Trello, where we noted down any roles and tasks that we would likely need to undertake during this project.

We then had a brainstorming session, where we noted down ideas. We used the method of "No Idea is a Bad Idea". Jamie & Mufazzil hadn't really played many text based or point & click adventure games before (as this style is rather out-dated), but Adam grew up playing them, so he spent a little bit of time talking about the kinds of things that take place in them – usually the collect-a-thon style. He found a video on youtube that showed the typical point & click puzzle design style – this video can be found here. Mufazzil also read games articles for inspiration.

After this initial session, we started thinking about some plot lines and objectives that could potentially shape the narrative of the game. Very early on, we talked about a witch placing a curse on someone, and the over-arching game objective was to break the curse.

Once we had this objective, it made sense to start building the skeleton of the game and map out a draft of the game's decision flow, by creating a flowchart on app.diagrams.net. Whilst this was initially rough, it laid down the foundations of how we could effectively refine these decisions and plan out our time before deadline.

How We Planned the Game



An example of our flowchart used in the planning stages

Your Role in the Team

Adam

The role I undertook in this project was mainly the creative side of this project. I was given this role as I had experience with playing these kinds of games initially and also really enjoy being creative, whereas Jamie and Mufazzil had experience coding in their spare time, so wanted to try some code that was more ambitious.

I took pleasure in building the story and world of the game on top of the skeleton plot that the group came up with. With a text-based game, it's important to build a mental visual for the player. I then wrote the narrative/story code into the project.

I worked well in the team during stand-ups, taking initiative to be proactive, and also suggest ways of working as a group, so that we could continue to communicate effectively as a group and utilise each other's strengths.

I think I could improve on the advanced coding side of things, attempt to learn some of the more advanced techniques that Jamie wanted to implement. However we didn't really have enough time to do that on this project.

Your Role in the Team

Jamie

My role in the team was to design the more advanced parts of the game such as the combat system and assisting in testing and bug fixing. I was given this role because I am confident in my abilities and am familiar with python which allowed me to complete my tasks in a timely manner.

When planning, I helped explain the technical aspects and limitations to maximize the potential of the game. Throughout the project I was available to offer support if my peers ran into any problems.

One area that I feel that I could improve is when it came to the tone and theme of the game I took more of a backseat. I should have researched the topic in more detail to assist in this aspect.

Your Role in the Team

Mufazzil

My role in this project was building a basic foundation for the storyline creating interactions and moving the character to different locations while adding items to an array.

I was given this role because I do not have the skills or knowledge for the more advanced mechanisms of the game but I have a basic understanding and knowledge of it.

I was able to do that. I worked well as a team as I offered clear communication however I could work on being confident on the part I’m doing.

Challenges if Front-End was Incorporated into this Project?

If we were to incorporate front-end into this project, this would be a big challenge as would require far more complicated coding, that is very resource-intensive. Therefore we would require more time than just a couple of days.

Advantages

- More exciting and would immerse you into the game world. Break up the monotony of just reading text.

Disadvantages

- More time-consuming - have to spend time creating the assets themselves too.
- Have to learn more coding languages.

Trello

Positives

- This ensures that first you look at the project as a whole, and then break it down further into more granular detail. It helps to plan and divide your time across the critical path to deadline, and note these resource-intensive tasks before you mentally get silo'd into the smaller tasks.
- This allows you to assign work fairly, and plan based on an individual's strengths and weaknesses, so that the group can support each other where necessary.
- Ensuring that you've got every piece of the puzzle present and accounted for.

Negatives

- You have to assign time to planning, and keep it updated.
- If this project was really big, it would require a project manager to keep the scale of a project in check. Very easy for it to become outdated.

Microsoft Teams

Positives

- Strong communication within your team.
- Useful for sending over files and having a shared file repository.
- Integrates the rest of the Microsoft Office Suite for maximum usability.

Negatives

- Serves its purpose for communication well but limited to that one use.
- If the system goes down, your communication and work cannot be accessed. Single point of failure.
- Requires an internet connection.

Working in a Team

Adam

I really enjoyed working in a team, as it allowed us to pool together our ideas collaboratively, both relating to the story and also the coding.

Having regular stand-ups were essential, as it allowed us to keep each other updated as to our progress and make suggestions if someone had questions or needed guidance.

The Kanban method was useful, especially to begin with, as it helped us to plan all the aspects we needed to consider when dividing tasks and starting the game.

Communication within the team was good, as we just flagged when we needed help and gave each other ideas.

I offered help with the team, by using my attention to detail. Flagging where some errors in loops were occurring, so that we could fix them. I also provided a input and help with suggesting story or plot elements that could help them show off their coding skills.

An instance where I needed help from the others was when I was writing the story and conversation text code into the game & I found that I wanted each paragraph to go one at a time, so it wasn't such a mountain of text at once. Jamie suggested putting a blank input [input()] in so that the player would have to press enter. This worked very well.

I coded the game by writing the story and character descriptions and conversations into the game. This was interesting as was having to fit in around the coding that Jamie and Mufazzil had written, and wanted to ensure I didn't accidentally mis-order anything. Thankfully that didn't happen.

Working in a Team

Jamie

I found working in a team useful for the development of the game. Not only were we able to complete more work but we were also able to make the most out of each other's skillsets.

To communicate effectively we held stand-ups which helped to get an idea of how we were progressing and talk about any issues we had or assistance we needed.

Such as when during one stand-up I helped Adam develop a way to break up text using input() and when I asked Mufazzil if he could complete some parts of the code, I said I would do. To create a more balanced workload and give me time to implement all the features I needed to.

We also used Kanban planning to separate tasks out however this was mostly organised during stand-ups with the Trello board being adjusted later.

I primarily created my code using classes and functions which could then be inserted into the main code that Mufazzil and Adam were working on. Then during testing and bug fixes I was given the code to look over and adjust.

Working in a Team

Mufazzil

I really enjoyed working in a team because it brings people together with one goal, and we can brainstorm ideas, knowing that "no idea is a bad idea." We held consistent stand ups to update everyone and see the progress and which way we're going. I offered help was by suggesting ideas for the game's story. I also asked for help with my code when I had trouble fixing loops for the non-linear option. We coded this project in Replit using Python.

Testing

When working on our individual areas, we would run the section in terminal and test that it was working as it should. We then changed things with the code to ensure that inputs functioned correctly.

When bug testing our game, initially we worked through playing the game as individuals first (choosing different paths, including the non-linear version), and noting any issues that we came across, and then fixed them.

After we each did this as individuals, we then did this as a group.

This felt effective as we were able to identify different issues that we were able to resolve as a group.

How Would the Experience Be Different if You Did This Project Alone?

This project, if created solo, would have been more challenging. The best part of this process is having peers to bounce ideas off of, and having someone who can help assist when you hit a mental-block.

Each person having strengths and weaknesses, means that a stronger project will be created from our combined strengths.

If done alone, whilst you would have full creative control over the themes, the game would’ve been a lot smaller in scale and therefore a lot less enjoyable to play.

Overview

What went well?

This project was smooth – we had a good variation of skill sets. Some with more experience in coding, some with more experience in creative, some who are actual fans of games in this style. Therefore when we collaborated, we could have comfort that we could support each other well.

What would you do differently if we did this again?

We could identify our skills earlier, to help better plan the game.

We could have a better method of file sharing and collaborating on one file, so that everyone could see the latest version whenever needed without needing to send files back and forth.

We could spend a little more time during planning, looking at other examples of text-based games and finding elements that you found really fun that we could have adapted and added to our game.

How did you find using Python to make a text-based game?

We found this process enjoyable – python is a very readable code, so therefore it felt fine reading other's code without need for explanation. Therefore it felt very good for collaboration reasons.