My Reflection on Our Group Project for COSC101 Class

Introduction

The COSC101 class project was like a big adventure for me and my project partner Sabin Dhital. We were excited because we had the chance to create our own video game. This was a big task, but it was also a chance for us to use what we had learned in class, lectures, and tutorials in a real-world setting. We decided to use the Scrum Development methodology as our software development life cycle to help us keep track of our work. This was a new method for both of us. We also used the Model-View-Controller (MVC) design, where I oversaw the "controller" part, and my partner oversaw the "view" part, which included the sounds, images, and animations in the game. For this, we first created a simple GUI and then divided our tasks for both of us.

Working in a Team

One big challenge we faced was that we were not in the same place. We had to work as a team but from different locations. This made it hard for us to keep up with each other's work sometimes. However, we had regular meetings online using different social medias where we could update each other on our progress. We also used tools that allowed us to share our code like google drive and so on and make changes together.

Working in a team also meant that we had to learn to work out our differences. There were times when we had different opinions on how the game should look or work. But we learned to listen to each other and talk about our ideas openly. We found that by doing this, we could find a solution that we both agreed on. The main advantage for both of us was we were from same home country Nepal and hence it was easy for us to communicate our design and concepts with each other. Upon a lot of discussions, we made a conclusion of dedicating the game towards the pride of our nation and hence tried our best to implement the concept, design and themes related to the pride of our nation. We wanted to implement the more themes and design related to our nation but due to time frame we were succeeded to use only few of them.

Despite these challenges, we were happy with what we accomplished. Our game had six detailed cities, a timer to add excitement, random missiles falling from the sky, and a special circular battery that launched counter-missiles. My partner did a great job finding the suitable sounds, images, and animations that made the game more enjoyable.

Implementation

Creating the game was not an easy task. For example, adding a timer feature to our game was very hard. It took a lot of testing and adjusting to get it working just right. Another big challenge was creating a way for the game to know when a missile had hit a counter-missile or a city. To solve this, we had to use some math to calculate distances and a method called "bounding box" to check for collisions. There were times when we had to deal with many kinds of debugging while implementing our work. We did a lot of hard research and took help from internet blogs and codes to find the solution and implement it in our game. One of the parts among many, was to use the GifAnimation for our game. Although it was not as we thought, we were compelled to use them due to lack of time as our deadline was approaching nearer. We used the help of internet and sometimes the popular AI chatbot chatGPT too to deal with our design, concepts and codes to complete our game.

Looking back, we think there are ways we could improve our approach. We could have spent more time planning before we started coding. This might have helped us avoid some of the problems we faced later. In the future, we want to spend more time thinking about the problem and breaking it down into smaller parts before we start coding.

We also think there are tools that could make our work easier in the future. For instance, there are project management tools that could help us keep track of our progress more easily. Also, learning more about good coding practices and software design could help us code more efficiently.

There are also parts of our game that we would like to improve. For example, the "bounding box" method we used for checking if a missile has hit a city could be more accurate. And the feature that makes missiles and counter-missiles bounce back when they hit the edge of the screen could be smoother.

Conclusion

Overall, working on this project was a great learning experience for us. Even though it was tough, it also made us work together and solve problems, and that's something we are proud of. As we keep learning more about computer science, we'll remember this project as a big step in our journey. We're excited to keep learning and improving our skills, and to use what we've learned to create even better things in the future.