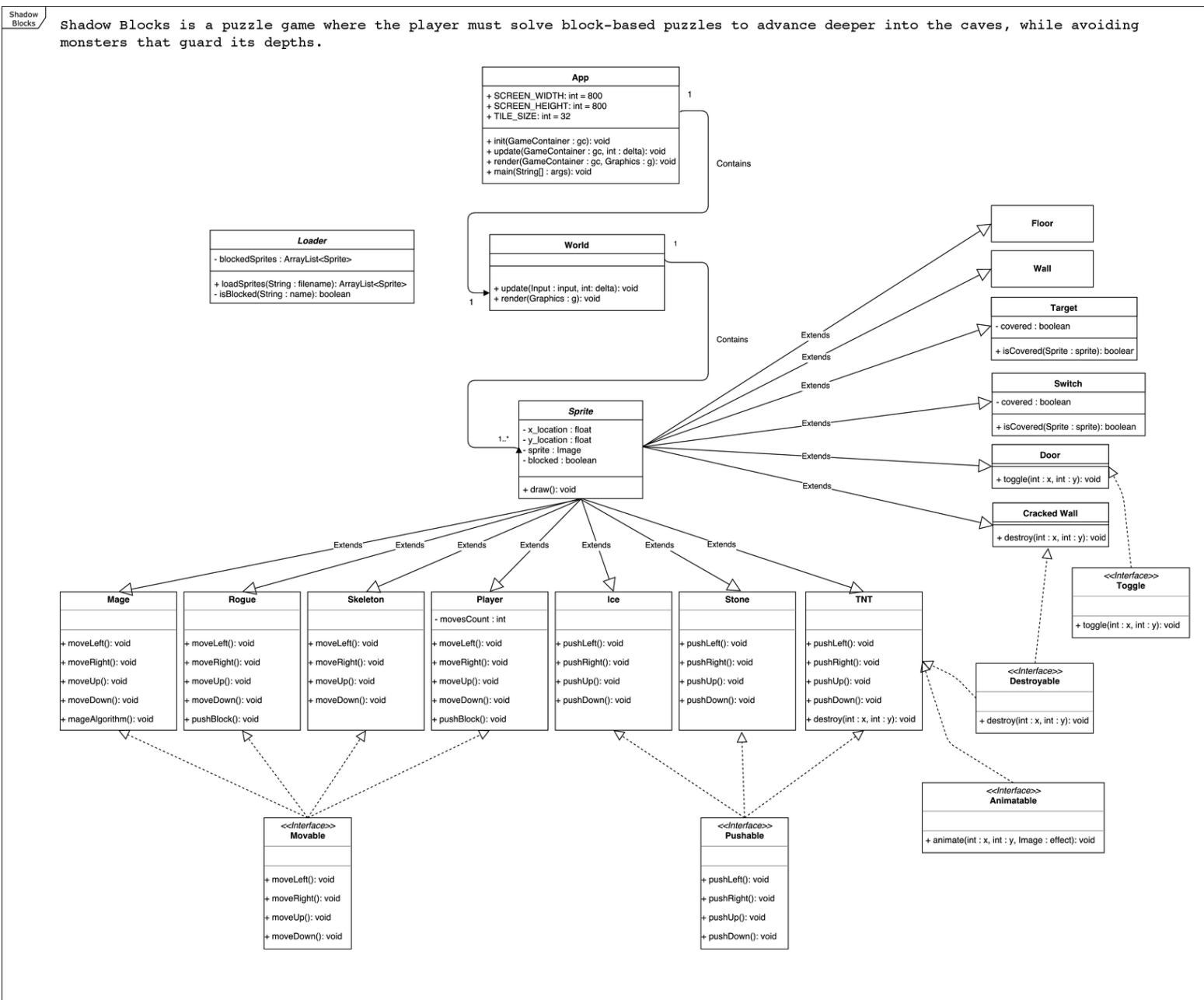


## REFLECTION – Danish Bassi (ID: 867811)

For this project, I was required to use the Slick2D library to create a 2D RPG game called “Shadow Blocks”. The “Shadow Blocks” is a puzzle game where the player must solve block-based puzzles to advance deeper into the caves, while avoiding monsters that guard its depths. The following image is the UML diagram I created which I used as a guide while developing the game. As I will discuss, a lot of the ways I thought about implementing the game features changed as the development went on.



Most of the interfaces I thought about implementing ended up not being used. In fact, the only interface from my UML diagram that was used was the Movable interface. This is because the Pushable interface was not needed at all since I found a different way to push blocks. I did this by checking the names of the blocks that the player object (or any other movable object) was touching

and used if-else statements to then push the blocks. The Destroyable interface was discarded because during the development, I could not figure out how to destroy a objects. So I worked around this by moving any “destroyed” objects off the screen away from the user’s view. The Toggle interface was not needed either for the same reason. Lastly, the Animatable interface was not required as only one object (the tnt) required it and the function(s) for that were put into the Tnt class.

There were a lot of issues with moving sprites that depended on the use of the ‘delta’ variable from the update method. It took me some trial and error to get things to work. There were also times when the code became unreadable since a lot of the larger methods were similar but not completely the same. It was difficult to scroll through the code sometimes. The issue which had me the most stumped however was near the beginning of when I started developing the program. I added the functionality to make the stones move, but whenever I would move a stone down or right, it would not render (until I moved it up or left). Turns out the 0.lvl text file provided in the first package was not written in the correct order. I spent over a week trying to understand what the problem was until I discovered that the 0.lvl file from the Assignment 2 starter pack was the correct version.

One key piece of knowledge that I learnt from the project was the importance of delegation. Although, I knew this before beginning the project, the large amount of features which needed to be implemented put things into perspective of how important this really is. I can only imagine how many lines of code the program would have been if everything was written in just one file!

There are a lot of things I would do differently if I did a similar project. One of them would be to start as early as possible! With a project of this size, there is a lot of thinking required beforehand which can become quite time consuming. Overall, developing the game was a great learning experience and has boosted my knowledge in object oriented programming.