Updates on Assessment 1 Deliverables

Requirements

New Requirements document: https://lloydbanner.github.io/SEPR-Team-7/Req2.pdf

The Requirements have been updated as to provide clearer wording that all requirements are rather than design choices taken by the software team before implementation of the software. A minor amount of changes were made in the system functionality column as to provide the details on what the system provides functionality wise rather than how we are going to implement said functionality. There is no change to the actual number of requirements to the original requirements document as we believe the requirements we have provide enough detail from the specification and client for the software and they are not unnecessary design pollution from such as choices we have made with no functional improvement.

In our explanation of how the requirements were acquired from the specification and client more justification has been provided as to why we used those methods to acquire the requirements and how we feel they were a good choice, including how it would provide a better experience during the design, implementation, and testing stages of our software development cycles.

When we move onto talk about research we have also provided more justification to choices made in the process, such as why we used the IEEE recommended practice of writing software requirements due to it being an already known, flexible, and easy to use choice. Additional Justification of research into other games of a somewhat similar style/genre was provided as we believe by using similarities between popular games, that makes them popular or why they are enjoyable to play as an end-user, would provide a much better end-user experience to the software. Justification has also been provided in the document as to why we not only researched popular but personal favourite games, this stems from the fact that a game that provides such a strong memory of being a favourite in that genre must have a reason, that being may be certain art styles, movement style, or the general fact that you have different choices to play(i.e. Classes).

Methods and Planning

New planning document: https://lloydbanner.github.io/SEPR-Team-7/Plan2.pdf

Changes

Over the winter break we changed our Scrum format. We had been having weekly meetings to discuss what needed to be done for the following week. When we got to the Winter break we realised we needed to work differently, as a result we made a plan for the winter before we left and agreed to a weekly group call to check on our progress and make any changes if needed. This was more appropriate for this time as it would be unrealistic for the team to meet over the break and it still allowed us to get a good amount of work done.

Within the first few weeks of the project we switched From using Jmonkey to LibGDX. It was taking a long time for us to understand how to use Jmonkey and as it was a 3D engine we had to work out how to convert the 3D parts of the engine in to 2D which required extra work for every step in the process. Another axis meant we had to convert everything to be two dimensional and meant extra work with the camera. This resulted in our change to LibGDX and has made making a 2D game much easier. There is much more community support for LibGDX and it more closely fits the type of tool that we need for the game we are making. Although LibGDX is a library rather than a full game engine, this has not proved to be an issue and has even helped in some ways allowing us to use Eclipse, an IDE we have all used before, more.

In our original plan we didn’t consider tools used for art or maps. Since we have been working on the project we realised these tools were very important to convey our requirements and allow the player to understand what is on the screen. We have used Piskel [1] an online pixel art tool to create our 2D pixel art for the game, this has allowed us to make recognisable locations and items related to the university. We also used Tiled [2] to create our tilemaps for the game. This allowed the us to assign strings to each tile making collision checking easier, and allowed us to have a foreground tile layer giving more freedom with the way the game looked.

Finally, we have updated our plan for the rest of the project. While doing this we noticed we had allocated no time for the planning of the assessed presentation on the gantt chart for assessment 4 and have modified this. In addition we have given a more detailed plan for assessment 3 considering dependencies. We believe not fully considering dependencies in the last assessment made it harder to manage and harder to decide which parts of the project to carry out next, as a result we have more clearly considered these in the plan for the next assessment. We hope this will make the next assessment more organised.

Risk Assessment

New planning document: https://lloydbanner.github.io/SEPR-Team-7/Risk2.pdf

After having some issues at the beginning of the assessment with using Jmonkey setting us back and causing us to reconsider the implementation of our game we realised it was important to identify a method for finding and adding new risks to the risk assessment. As a result we have decided to consider risks when we start new processes that we haven't done before or when we use new tools. This will prevent us from choosing incorrect tools for our tasks as we will first consider the best tool for the job. This resulted in us adding some new risks to the risk assessment that were based on our experiences after creation of the first version of the risk assessment. It would have been hard to identify these before starting the project so we believe this is an important change.

Furthermore, we have added a risk ownership section to our risk assessment. Before we did this we found that people didn't keep track of risks as well as they should have. Giving responsibility of a risk to someone means that they know what they need keep track of, as a result they don't have to consider everything making the risks easier to manage. In addition, this prevents a less significant risk that people may not consider all the time from becoming forgotten. This prevents small risks from becoming a large problem as they are always considered.

In addition, we added colour coding to the Risk assessment on the likelihood column. This is based on how likely the risk is allowing us to more easily visualize which risks to prioritize. This gives us a clear focus when regarding risks and allowed us to more easily identify how to assign risk ownership. As a result we gave a good spread of likely and unlikely risks to each member of the project. This means that everyone can consider the same amount of risks and that each person has risks to prioritize avoiding. This gives us a clear method to avoid greater risks.

The rest of our risk assessment has remained unchanged as we believe it worked effectively and justified risks well. Furthermore, we believe our method for finding risks and our justification for finding the was strong enough to be useful throughout the project. Our risk recording table seems to be effective and has allowed us to easily expand on the information within it. Adding new tasks and risk ownership was easy with this method and so we believe it makes sense to continue to use it. Also, we have found no further research to suggest a better method than the IEEE method we have been following.

References:

[1] Piskel, Sprite Creator, Available: https://www.piskelapp.com
[2] Tiled, Map editor, Available: https://www.mapeditor.org/