

Developing a computer game that utilises 3-D graphics to be as enjoyable and engaging as possible so as to increase the average user's play time

In brainstorming for the concept of the game I was to develop, I considered my options for the genre and was immediately able to rule out quite a number of them. Due to time constraints I knew that designing characters and a number environments would be impractical, as this is usually done by a large group of people over a period of years^{1,2}. Furthermore, I wanted my game to have replay value, so I also rejected the idea of a game that revolved around storytelling.

Following the advice of Andrew Rollings and Dave Morris in 'Game Architecture and Design: A New Edition', I decided to let my creativity take over from here and assess the feasibility of each idea afterwards³. When I came to decide on a concept I had to reject my initial vision of a racing game due to the need for sophisticated artificial intelligence or online multi-playing which would require a dedicated server.

I decided on the space shooter proposal because it was the one I felt I would be happier playing; Rouse states this to be one of the most influential aspects in deciding the success of a game⁴. I also reasoned that its space-themed science fiction would be relatable to both modern gamers (having played games such as 'Sins of a Solar Empire'), and 'retro gamers' (being reminiscent of some of the early electronic games such as 'Elite'⁵). Coincidentally, the game can also be likened to infinite running games such as Temple Run which have certainly been popular and engaging despite their simplistic gameplay⁶. As a result, these games have been some inspiration, but I have been careful to avoid 'cloning' a game⁴ or forcing in elements of gameplay for their own sake⁷.

In designing my game I made some sketches and mindmaps of how the game was to look and operate. From Rollings' and Morris' list of goals in games I could see that my vision already had the goal of removing obstacles and I had also decided to incorporate a scoring system, which then fulfilled the criteria of 'collecting something'⁸. I prioritised the features of the game⁹ so that I would be able to develop the core gameplay first and use this as a prototype which would help me to assess how enjoyable the game is and potentially adjust the direction of the game accordingly¹⁰.

During the design phase, I was learning some of the basic skills I would need during development of the game, mainly controlling 3-D objects in the Blender Game Engine using logic bricks and the Python programming language. My prior knowledge of programming made the learning curve less steep, but I found the Blender Game Engine's Python API confusing, especially considering the lack of up-to-date documentation. Due to Blender being open source software, anyone can contribute to its development¹¹ and therefore the documentation¹² is open to editing by the public, making this source of information less reliable and also incomplete¹³. Books and articles written about the game engine may be obsolete if they are a few years old due to

development of the software. Learning was therefore a case of experimentation with information from various sources; luckily experimentation allows me determine which information is correct so that the quality of the product is not affected.

After the completion of the initial designs I modelled a simple spaceship based on the advice of Price¹⁴ and implemented basic controls for the spaceship. Development was then iterative prototyping which focused primarily on gameplay – this is ‘doing it properly’ according to Rollings and Morris¹⁵. After adding asteroids to the game I improved the graphics so that I could observe how much of the computing power was available for logic and non-rendering calculations – I used my own machines for testing and developed with the average modern computer in mind. Optimisation of my code and logic as well as changes to settings in the game engine improved the smoothness of the game by allowing more frames to be rendered per unit time.

Implementing health and shooting completed all of the primary features I had planned, meaning I was then able to conduct some playtesting with the main purpose of discovering annoyances in the game, and ensuring that players could understand and use the game without any trouble. This confirmed that my game had the potential to be very engaging while also highlighting the issue of visibility which prompted me to create a star-filled background and change the lighting configuration at the cost of realism.

I then sought to add as many of the secondary features as I could¹⁶ such as scoring, health packs and acceleration to increase difficulty as the game progresses. I also had to cut some features including pausing, saving high scores and an introduction or instruction menu simply because they did not affect how the game was played and would therefore not be as valuable a use of time as other features. Further ideas such as pickup weapons and enemy spaceships that were born during the development phase had to be rejected in order to avoid ‘feature creep’¹⁶.

I modified my game to output the time the player survived after each run so that I could record the data during playtesting of this latest version of my game. I also designed a set of questions to ask each player after they had played my game which I would use in conjunction with their times to improve the game.

The results of the playtesting showed that players would find it more enjoyable for single runs of the game to last 1.64 times longer on average and that damage should be more avoidable but in larger amounts. As a result I increased all damage and healing effects by 22% and reduced the acceleration of the asteroids as the game progresses by 50%. The mathematical reasoning behind this is that we can interpret the game as needing to be 1.64 times easier; $\frac{1}{1.64} \approx 0.61$ times harder. $(1 + 0.22) \times 0.5 = 0.61$ times harder. After this modification, the mean survival time of players was 1.78 times greater than before, indicating that my game is close to the most desirable play length. However, this is not such a fair comparison, as I had to use different players in each of

my playtests because people are likely to do better on their second play due to experience gained the first time around.

Finally, I produced a looping music track for the game with both a techno and digital lo-fi theme befitting of the futuristic nature of the space travel concept while also reminding players of classic retro space games which were known for their 8-bit soundtracks¹⁷. I also produced computer-generated sound effects of the same theme to give the user feedback on events occurring in the game, seeing as visual feedback could go unnoticed in the midst of frantic gameplay.

In conclusion, I feel that I have produced a relatively enjoyable and engaging game that is likely to see up to a few hours of play time per person due to its replay value and convenient length.

References

¹Pauline Jacquey in R. Weber. On Reflections: First interview with the Ubisoft studio's new MD. *Gamesindustry.biz*, 28 February 2013. [Online] Available at: <http://www.gamesindustry.biz/articles/2014-02-26-on-reflections-first-interview-with-the-ubisoft-studios-new-md>. [Accessed 5 March 2014].

²Masahiro Sakurai in A. Seedhouse. Sakurai shares his thoughts on game development length. *Nintendo Insider*, 18 February 2014. [Online] Available at: <http://www.nintendo-insider.com/2014/02/18/sakurai-shares-thoughts-game-development-length/>. [Accessed 5 March 2014].

³A. Rollings, D. Morris. First Concept. *Game Architecture and Design: A New Edition*. Indianapolis: New Riders Publishing, 2004, p.6.

⁴R. Rouse III. What Players Want. *Game Design: Theory & Practice*. 2nd ed. Plano, Texas: Wordware Publishing, Inc., 2005.

⁵Frontier. *Frontier Developments*. [Online] Available at: <http://frontier.co.uk/games/elite>. [Accessed 12 January 2015].

⁶N. Lovell. Four reasons why Temple Run is dominating the app store. *Gamesbrief*, 16 January 2012. [Online] Available at: <http://www.gamesbrief.com/2012/01/four-reasons-why-temple-run-is-dominating-the-app-store/>. [Accessed 10 August 2014].

⁷A. Rollings, D. Morris. Core Design. *Game Architecture and Design: A New Edition*. Indianapolis: New Riders Publishing, 2004, p.38.

⁸A. Rollings, D. Morris. Core Design. *Game Architecture and Design: A New Edition*. Indianapolis: New Riders Publishing, 2004, p.41.

⁹E. Bethke. Controlling Feature Creep. *Game Development and Production*. Plano, Texas: Wordware Publishing, Inc., 2003, p.288.

¹⁰R. Rouse III. Getting the Gameplay Working. *Game Design: Theory & Practice*. 2nd ed. Plano, Texas: Wordware Publishing, Inc., 2005.

¹¹Blender. *About - blender.org - Home of the Blender project - Free and Open 3D Creation Software*. [Online] Available at: <http://www.blender.org/about/>. [Accessed 29 January 2015].

¹²Blender. *Blender Manual Contents – Blender Reference Manual*. [Online] Available at: <http://www.blender.org/manual/>. [Accessed 29 January 2015].

¹³BlenderFoundation. (2014). *The new Blender Documentation project – Francesco Siddi and Campbell Barton*. [Online Video]. 25 October. Available from: <https://www.youtube.com/watch?v=OLQGGbiKfSs>. [Accessed 29 January 2015].

¹⁴Blender Guru. *Create a Spaceship – Part 1 of 2*. [Online] Available at: <http://www.blenderguru.com/tutorials/create-a-spaceship-part-1/>. [Accessed 18 May 2014].

¹⁵A. Rollings, D. Morris. Use of Technology. *Game Architecture and Design: A New Edition*. Indianapolis: New Riders Publishing, 2004, p.525.

¹⁶E. Bethke. Controlling Feature Creep. *Game Development and Production*. Plano, Texas: Wordware Publishing, Inc., 2003, p.290.

¹⁷K. Collins. *From Pac-Man to Pop Music*. Ashgate Publishing, 2008, p.2.