|  |  |
| --- | --- |
|  | **Game design and development.** |
|  | |  | | --- | | **Sprint document RST**  Written by: Liam van Toorenburg  Class: CMV2C  Written for: Game Design (GL5)  Published on: 28-05-2017 | |  | |

Inhoud

[Analysis (sprint 1) 2](#_Toc487130059)

[Sprint 2 added features 4](#_Toc487130060)

[Sprint 3 Final changes 6](#_Toc487130061)

[Logbook Sprint 1 7](#_Toc487130062)

[Logbook Sprint 2 8](#_Toc487130063)

[Literature list 9](#_Toc487130064)

# Analysis (sprint 1)

In order to find out how to improve the 2D prototype I’ve decided to analyse the binding of Isaac and rogue legacy, which are two games I’ve enjoyed playing and played a lot of. I started out by analysing these games and the 2D prototype through the lens of Flow (Schell, J. 2014). By doing this I quickly realised each of these games had pretty clear cut goals:

In Binding of Isaac, the goal is to move through a set of dungeons, becoming stronger and progressing through levels that move up in difficulty along with your increase of power, beating bosses, periodically encountering new enemies and unlocking new items along the way. The very same description can also be used for rogue legacy, with the biggest change being the fact rogue legacy has a progression system that continues even after death, while binding of Isaac basically resets after each death except for unlocking new characters to play.

So, what does the 2D prototype have to offer? Well the goal of the game is to stay alive for as long as possible by gathering food and increasing difficulty without introducing new challenges. The goalpost is different from most roguelikes as the goal itself is to stay alive and move onwards, instead of it being the means to the goal, which is usually beating some big end boss or reaching the conclusion of a story. To put it in a more poetic light, the homeless man you play in the 2D prototype simply lacks a reason to live and therefor the player lacks incentive to keep playing except for perhaps beating a high score.

Perhaps even more important than a great goal however, is the lack of content and challenge. In the current prototype, it is actually possible to calculate the optimal numbers of steps to reach the next level with as much food as possible. Therefore, skilled players require on luck in random generation to get a better high score. Roguelikes also thrive on replay ability and thus the game will first and foremost need to have a lot of variety in order for a player to even care about the goal of the game, with is currently confined into one small boxy level with some food items, walls and one type of enemy that all get randomly placed. Not very exciting. If there is no challenge or fun involved in getting a high score, what is that score even worth? This, according to Bartle’s Taxonomy of player types is the mindset of Achievers (Schell, J. 2014). For this type of player there needs to be more challenge in the game at the very least, since otherwise a state of flow can never be reached. I therefor think it’s important to add a bigger variety of smarter enemies into the game, which keeps it fresh and adds to the learning curve. The roguelikes mentioned before do not just focus on pleasing achievers however, they also cater to a type of player called explorers. Explorers get their kick out of discovery. For them the fun is not so much in the challenge of the game, but much more in finding all there is to be found. Treasure, progression, secret areas you name it. In the case of the roguelike genre, achievers and explorers can easily live in harmony for the most part, A variety in level layouts for example can delight explorers with all the new content, while also making the game harsher and more unpredictable to achievers.

By playing rogue legacy especially I reached the conclusion that the two most important aspects of a roguelike for me are the combat system and the variety in levels it offers. If the game would sport a smaller variety of enemies It would quickly become boring as patterns are easily readable and after a while of playing some enemies on their own just become a grind to fight. The game also features a way for you to keep the level layout the same in your next run through the dungeon. I also found this to make the game more predictable and good for grinding but not for actually having fun.

The fact you cannot fight and have to run away from your enemies in the 2D prototype can be a fun challenge and way of playing and changing that will not have a massive impact on the game, as being able to kill your enemies actually makes the game easier, and doesn’t require much of a learning curve unless the enemies themselves get an overhaul. Adding new items right off the bat also won’t make the game much better, as the level is too small and there isn’t much of a challenge in the game that requires the use of powerups or items.

Therefor I think the best way to start is by changing the levels and variety of enemies. Even separately these features both drastically change the game. Random generation of level shape and size based on the level you are on for example already adds a whole new layer to the game since you are not able to see all enemies and food items in the game at one glance, increasing the difficulty of getting to the exit on time. Enemy variety in turn could make a difference in even smaller levels by for example adding ranged weapons that could increase the need for walls you can hide behind and make it harder to outrun enemies.

For the initial prototype, I’m mostly planning on adding random generation of levels however and perhaps also changing/balancing the hunger system.

# Sprint 2 added features

* The first and most obvious change when starting the game is that it now has a new main menu. For now this does nothing except lead the player into the game. The menu still needs some polishing to look better and was mainly added due to the fact a high score menu is planned for the final sprint which will be accessible from this menu.

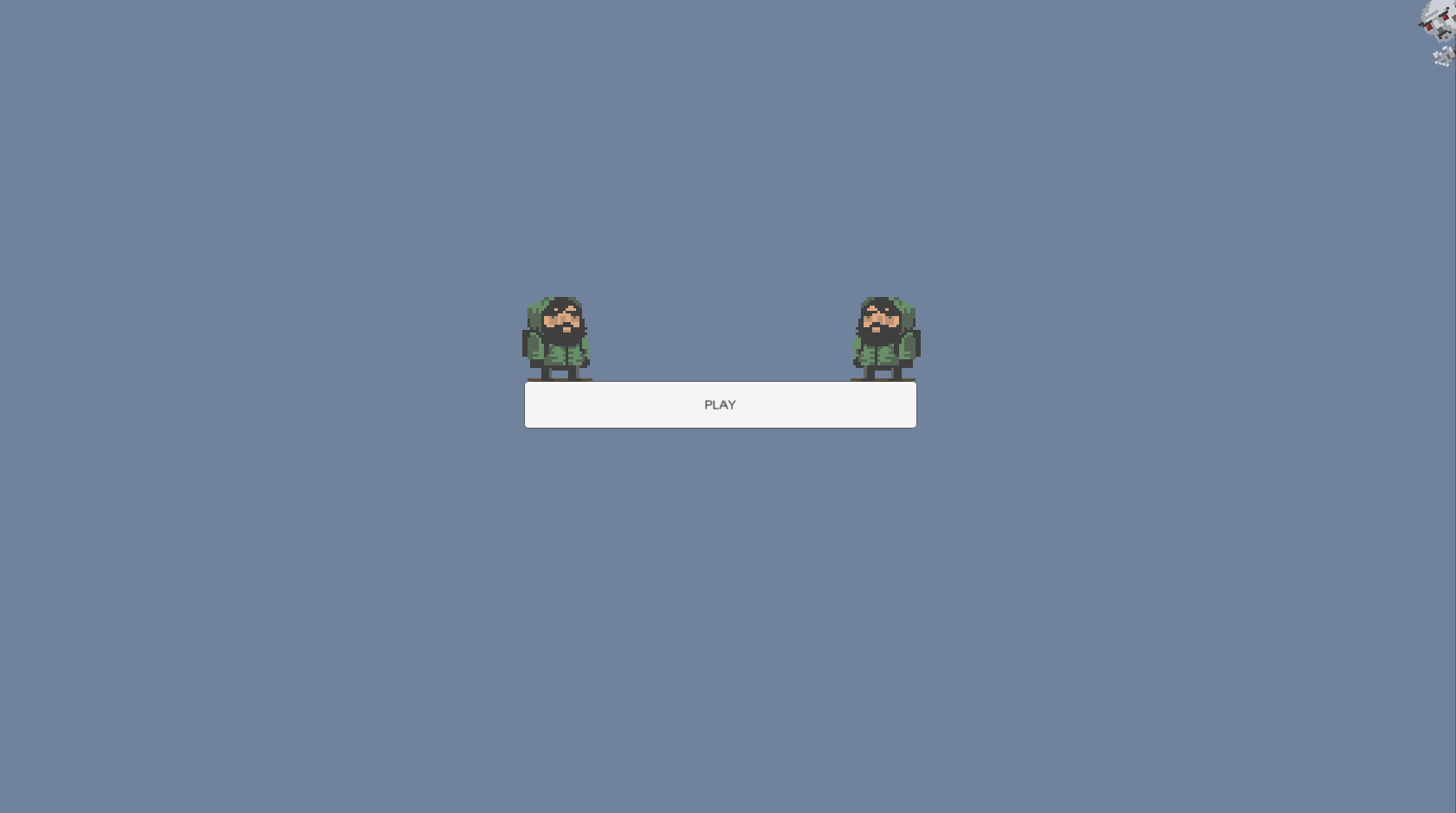


Illustration 1: Main menu

* Level generation has been improved upon. In the original game you only had one room and the only progression the game had was that it occasionally added another enemy or two. Right now however, the game starts with two rooms instead of one and each level either the enemy count or the room count has a chance of increasing, making the game longer and more challenging. Breakable walls are now also the only walls present in the game except for the outer walls. This was done because it adds a strategic way to avoid enemies even though it costs a lot of food to dig through them. The exit also always spawns in the last created room, whereas the player spawns in the first, meaning the player will have to go through most of the level to reach its end.



Illustration 2: Level 3 of a randomized run.

* The food system has also been slightly tweaked. As shown in the image above the UI for the food count is now in the bottom left for example. While not necessarily a big improvement it does illustrate the changes made. Another change to the food system that has a big more impact is that the food count now resets every level, meaning that each level while more difficult than the last does give a fresh start so to speak. This makes the game easier to balance, but removes an element of progression the game had. It is however a trade-off I feel will be worth it since having a balanced game is more important.

# Sprint 3 Final changes

* The first change I worked on for this sprint was balancing. While my first sprint added the possibility of digging through outer room walls, this was only meant as a temporary measure. Eventually I wanted to restore the indestructible walls to the outside of the rooms so you could not just dig straight to the exit and take shortcuts. This due to the fact the player already has a tool at his disposal to get through narrow hallways. Enemies will simply follow you through vertical hallways if you are in their vertical line, which is something I hadn’t realised until this sprint. This interesting change in my perspective of enemy UI now makes hallways more easily traversable if you are skilled at the game. Therefore, there is no need for digging through big walls and will now only provide a way of certain death for the player as he can get stuck in the wall with an enemy following him or his food can run out from digging, meaning the strategy has no real benefit unless the exit is right around the corner, which makes using the option feel like cheating.
* After changing the walls back to how they were supposed to be, I decided It was time to change the art of the game. I am however not a very skilled pixel artist and I soon gave up on my idea to create a cyberpunk setting with robot enemies and new pickups, instead deciding to simply recolour all of the assets. I didn’t do this without a purpose however. Since I programmed in a feature that has the sprite sheet change every 5 levels, providing a fresh new environment for players to walk in. So far, I only included one new sprite sheet as a proof of concept, which just cycles with the old one every 5 seconds, this since It is not one of my major changes and I didn’t want to spend too much time adding new things in the final sprint.

# Logbook Sprint 1

03/05/2017

8:30-15:30  
I spend the first class setting up the unity project and playing the game, writing down points of improvements and relating them to other games, specifically Rogue legacy and binding of Isaac, after which I went home to work on it some more.

In this session, I came up with a few interesting points:

First of all, the game lacks real progression. The monsters that get extra spawns every few levels are boring and either quite easy or impossible to avoid depending on the level. The game also seems to lack a variety in mechanics and doesn’t have much replay value due to lack of a progression system and with only one small level to explore that except for some obstacles is always the same.

While I could probably think up some grand overhaul that would make the game really enjoyable, there simply isn’t enough time to create any giant changes due to my lack of C# knowledge. Therefore, I’m planning to work on a new system for random level generation, starting with a tutorial I found related to this project on the unity website. Besides that, I’m also thinking of adding a few new items, though the exact nature I will decide on in the next working session.

11/05/2017

19:00- 22:00

I spend today working on writing the analysis for rogue legacy and binding of Isaac, relating why I think the games do not become repetitive as quickly as the 2D prototype with a heavy focus on the level generation and combat. This analysis can be found in the next chapter of this document. While I could have written a more detailed analysis of every single aspect of the game I don’t think it will matter much in the end, since due to limitations in programming skills and scope only a few changes can be implemented. Therefor I decided to focus on the big picture first and the basics that every rogue like needs. I didn’t focus too much on what made the games unique and why these in particular stand out from the crowd simply because I feel uniqueness and quality means little in this assignment especially compared to awareness on the genre and what is the bare minimum of making a playable rogue like.

14/05/2017

15:00 – 21:00

Today I worked on the unity tutorial and adding level generation. I ran into a bug where my player refused to move however, so I will need to ask for help at school tomorrow.

15/05/2017

13:00 – 16:00

I tried to fix the bug, however our projects programmer was ill today and I couldn’t quite fix it even though I rebuilt the project a few times and asked around for help. However basic random generation is already in there.

# Logbook Sprint 2

design:

As mentioned before during the presentation this sprint for me is going to me mostly about changing the level generation as this will add to exploration and challenge. I plan to take the level generation scripts i handed in last time, fix the errors, balance the game out and then add the generation to a progression system where every next level either spawns more enemies, more rooms or less food, while also adding new sprites every few levels. This will hopefully make the game more challenging, change up the levels and because of the new sprite sheets add a bit of exploration to the game.  Other than that I want to make some changes to the enemies themselves. I want them at the very least to have a set distance to the player to be within to be able to move. This will remove some of the waiting time between turns as well as make it easier to move around without all enemies following you. It also adds more realism to the AI.  The reasons for focusing on these aspects has already been explained in my last sprint document and is based on analysis of different rogue likes as well as some theory from the book of lenses by Jesse Schell.

Day 1:

20/05/2017

start: 13:00

End: 17:00

effective work time: 3 hours.

I started out my sprint by attempting to fix the bugs I left behind. Specifically the food text no longer displaying and the player occasionally spawning in the wrong way. It took me quite a while to figure out it would be better to just create new code for the food text and remove the existing prefab. Once I did however I managed to get it to work pretty quickly with some help from unity documentation. All in all I probably spend a good three hours on this.

Day 2:

25/05/2017

start: 14:00

End: 20:00

Effective hours: 5

Today I worked on restoring the exit to the game and making sure players could get to a new level. This turned out to be a lot of work as the way the new board creator script was set up didn’t match the original code for changing levels. By the time I finally managed to get a new system for level progression up and running and fix some bugs with player spawning it was 8pm. The level mechanic is still basic however and needs some improvement to add actual progression.  I also noticed my food text resets every level, though I intend to keep this as a feature since It’s much easier to balance the game later if I can calculate an amount of food points per level, however this does effectively mean that previous levels have no effect at all on the current level, which is something rogue likes usually thrive on. I feel this tradeoff can be well justified however by the fact balancing will be easier and will save a lot of time.

Day 3:

27/05/2017

start:13:00

end: 17:00

Effective hours: 4

Today I continued working to actually add a new progression system to the levels. I started out by writing code that changed the spawn rate of enemies/rooms to scale with levels, thus increasing difficulty per level. At first I wanted to scale food as well, but I decided against that since this could lead to impossible levels if done incorrectly.  Afterwards I also wrote some code that made the game restart itself after a game over, as well as remove the food text after death. This was mostly for player convenience.

Day 4:

27/05/2017

start:13:00

end: 15:00

Effective hours: 2

I made a main menu today that starts the game. I also worked on the document and deciding on what to do next sprint.

Next sprint:

Next sprint I’m planning to focus mostly on balancing and art. Specifically adding new sprites every 5 levels and turning the player sprite around when he moves left or right.  Although I also still want to include changes to the monster AI I fear this will be a bit too hard as I couldn’t quite figure out where to start with it this sprint either. I suppose being able to kill them would be my best bet, however I’m hesitant since I think the game will just change in a way that just makes it feel like a grind. Besides these changes, I’d also like to add a high score system to the game to promote competitiveness and motivate players to keep playing. This shouldn’t be too much work.

# Logbook Sprint 3

04/06/2017

start: 13:00

End: 18:00

Effective hours: 5

Today I started out by looking through the document and evaluating my choices so far. I had learned during the sprint 2 presentations that the enemy AI behaved slightly differently from what I had noticed so far. This change, as listed in the sprint 3 changes section has led me to work on adding walls back into the game and re-balancing the game around that. The rebalancing didn’t quite feel right and so I decided to tweak a lot of the enemy spawn code. However, after implementing a new system it actually felt worse so I went back to the old code. This wasted me a few hours.

05/06/2017

start: 13:00

End: 18:00

Effective hours: 5

After ending on my failed balancing attempt the day before I decided on doing something else. I changed my sprite sheet to have a different palette, since I didn’t have the skills to create entire new sprites and then proceeded to focus on creating the system to swap both sprite sheets every 5 levels.

# Literature list

Schell, J. (2014). *The Art of Game Design: A book of lenses*. CRC Press.