

Milestone Report 1

Game Idea Pitch and Development Plan for a Twitch-based Puzzle-Action-Game

1 Game Description

1.1 Game Idea

Prepare cooking ingredients to serve up the desired orders. But that is easier said than done, because the Twitch-Chat might have something else in mind!

In this Twitch-based puzzle-action cooking game the twitch-streamer is trying to cook according to a recipe. He or she needs certain ingredients in pots and pans that are at the end of a field full of conveyor belts which transport the food components. The gameplay consists of controlling the conveyor belt tiles on the board so that the ingredients find their way to the desired destination. The player tries to complete as many dishes as possible in a limited amount of time to reach high scores.

The twist of this game is, however, that the twitch-chat has an influence by sending chat-messages to vote in polls. By voting, a twitch user can manipulate in which of the given positions the ingredients will spawn and also which ones will appear on the board. In addition to that, they can also vote on the tiles that the streamer gets access to.

1.2 Theme-based decisions

We took the given theme "Chaos and Order" and split it into its two aspects: One focus is on "order" as in "ordering food". Our game idea revolves around fulfilling the orders of customers by cooking food for them. But order can also mean "to bring structure into something" and this meaning was arguably even more important for our game idea decision process. The main premise is to order the conveyor belts so that the arising structures benefit the player and he or she can complete the given orders exactly as they were requested.

The other part of the project theme is responsible for our technical idea, i.e. integrating the twitch-chat into our game. The twitch-chat is the single most chaotic concept known to mankind so there is nothing more fitting than using this chaos to ensure unstructured and unsettled behaviour of our game. Using the natural creativity of humans we can ensure that a streamer will not find himself in a boring loop of ever-repeating gameplay. If the player finds a strategy that works well to deliver food as planned, the chat users can also adapt and find weak spots in the streamers strategy almost like an artificial intelligence that is improving and changing with its opponent, except that it is not artificial.

Besides that, other concepts were also designed with the idea in mind of bringing in more action and excitement: Making the game real-time instead of turn-based and adding a timer as the finishing condition ensures that the streamer is stressed at all times in

the game, without giving him time to breath which should lead to the feeling of chaotic gameplay rather than giving the player the time to make optimal decisions.

Finally, we also decided to punish the input of wrong ingredients into the cooking pot by failing the entire recipe. The idea behind this is that the streamer has no other option but to repeatedly destroy his own pathways which were created for previous food deliveries. With this cycle of trying to create, destroy and recreate paths through the board there is the requirement to bring order as well as chaos to the conveyor belt structure.

1.3 Gameplay

In this section, we want to give a more concise description of the actual gameplay. However, many aspects are subject to change as we will constantly evaluate whether the game feels chaotic enough while both the streamer as well as the Twitch-chat users still have enough impact with their own decisions.

A session starts with a randomly selected recipe and an ingredient list that are displayed at the bottom of the screen. Next to that is a pot, which symbolizes the goal where the streamer will try to lead the ingredients. Above the pot is a tile-based board [1] where each tile is already filled with a randomly shaped conveyor belt. These conveyor belts move ingredients that are moved on the according field. At the top of the board are spawning tiles where the different ingredients will spawn. On the right side of the board are the conveyor belt tiles that the player currently has available for placement. These can be dragged and dropped onto the board to replace existing tiles. Further on the right side of the field are two live polls [2] that are influenced by chat messages from Twitch-chat users. The first poll shows a combination of an ingredient and a number which points towards one of the spawn points as well as a timer that shows the remaining time until this particular poll closes. Once one closes the shown ingredient will spawn at the indicated position. The second poll shows different conveyor belt tiles and works similar to the first one. After a certain time, the shown conveyor belt tile will be placed into the UI on the right side of the board where the streamer has the currently available conveyor belt tiles.

Once the game has started more and more ingredients will spawn at the top of the board. However, not all of them will be necessary for the order that the player is trying to complete. Therefore, the streamer has to carefully adjust the conveyor belts by replacing them so that only the correct ingredients will end up in the pots. The others will either fall of the board if the conveyor belts are placed accordingly or remain on the board in "dead spots" where opposing conveyor belts keep them until one of them is changed. Eventually, the time will run out and the streamer will receive a score based on his completed recipes and the game can start again from zero.

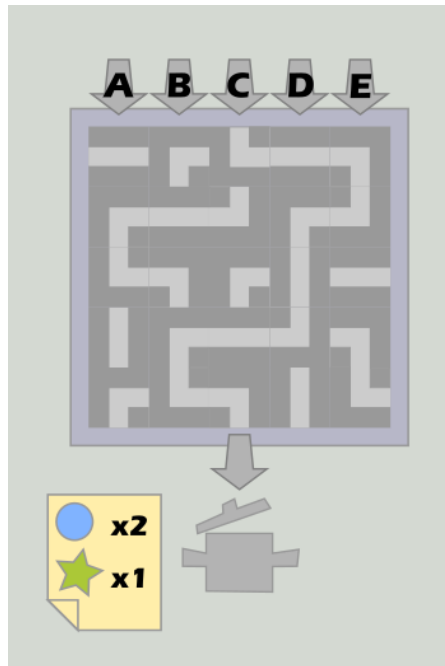


Figure 1: The board. The spawning points are above, marked with letters from A to E, the cooking pot - goal - is below the board. The recipe shows which ingredients have to be lead to the pot.

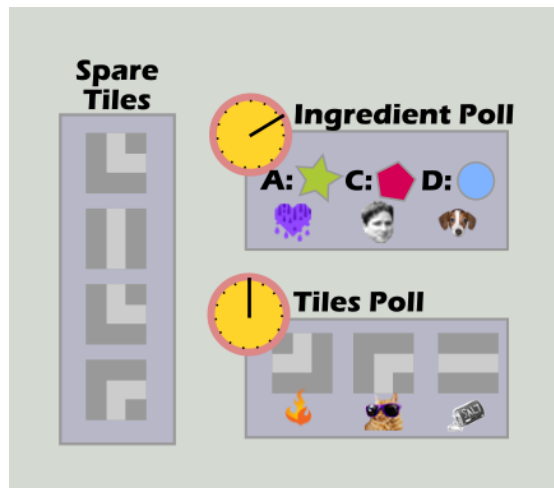


Figure 2: The polls. Above one for the ingredients, below one for the board tiles. The clocks are countdown timers that show how much time is left till the corresponding poll ends.

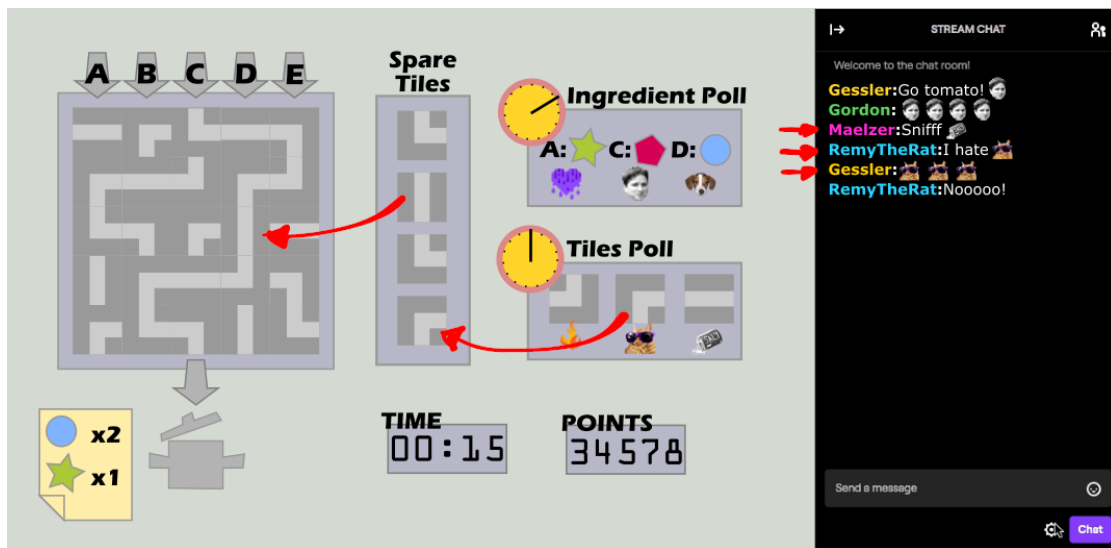


Figure 3: Whole game screen including Twitch chat (on the right). Stream viewers can influence the game and vote on tiles that will pop up in the Spare Tiles pool. These spare tiles can be then used by the player to build a conveyor belt that carry suitable ingredients to the pot.

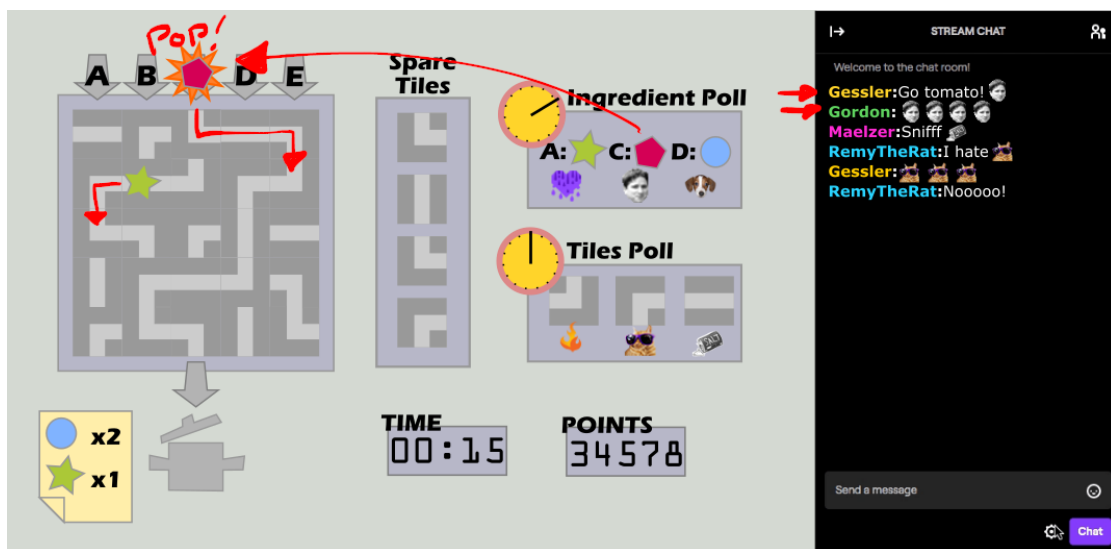


Figure 4: Whole game screen including Twitch chat (on the right). Stream viewers can influence the game and vote on ingredients that will spawn at defined spawnpoints. These ingredients will then ride along the built conveyor belts.



Figure 5: Core Idea: Completing food orders by ordering conveyer belts.
Technical Innovation: Integrating Twitch chat chaos via poll system.

2 Development Schedule

2.1 Layered Overview

Following the project overview document, the project's development process should be defined as a 5-layered step-by-step plan. In the following, all these 5 layers are defined according to their currently foreseeable functional requirements, starting with the minimum viable product.

2.1.1 Functional Minimum

We define the functional minimum as the technical core functionality all later development steps build upon. These are currently seen as, firstly, a rudimentary interconnection with the required Twitch APIs allowing basic inputs during a stream session from chat to be processed during the game's runtime, secondly, a basic grid-based representation of the puzzle-action-game with placeable tiles, and lastly, if the functional minimum already needs to be defined as a game, a very simple win-condition which can be added at this point.

2.1.2 Low Target

The low target is to be defined as a state where the game is fully playable with its core features, as stated in the previous sections, already implemented. These include the completely finished poll system executed via the Twitch-internal chat, the inclusion of several different basic conveyer-belt-tiles like, e.g., straight tracks, turns, blocking tiles, crossings, etc., and 3 to 5 different ingredients used in a single cooking method like 'boiling soup'. A scoring system is also required at this point.

2.1.3 Desirable Target

After achieving the low target, the desirable target includes features the game should have after all aspects our project must have are already put into place. This milestone primarily consists of a main menu, including graphics, sound, gameplay and twitch-based options, a highscore listing the players' scores, 2 to 3 cooking methods, 5 to 10 different ingredients and 1 to 3 conveyer-belt-tiles which render the gameplay more exciting (These are not fully discussed now, but could be, e.g., tiles which cause ingredients to be thrown away or which produce new ones from two or more inputs.). At this milestone, the inclusion of music and basic sounds are worthwhile too.

2.1.4 High Target

The primary high target should include an online leaderboard which lists the best players. Considering the gameplay, 3 to 5 cooking methods, 10 to 15 different ingredients and 5 to 10 bonus tiles producing and altering ingredients. At this point the addition of various graphical effects is also worthwhile to give the player a more immersive experience. More music and sounds (also for UI elements) are to be added and general improvements

need to be taken into consideration based on play experiences. However, these cannot be exactly pin-pointed at this point in time.

2.1.5 Extras

Extras could be, e.g. a two-player-mode, either co-op or versus, more twitch-chat mechanics like causing an earthquake when too many players spam a poll and eventually more visual effects if reasonable.

2.2 Development Timeline

The project's timeline follows a Waterfall-like approach. Development can be split into two major time intervals: The phase before the Alpha release during which the majority of the game's base functionality is implemented and the phase after the Alpha release during which we would like to improve our game based on playtesting results and our own experiences we hopefully will be able to collect during the course of the first phase. The entire timeline can also be seen on our wiki page of this year's Computer Games Laboratory.

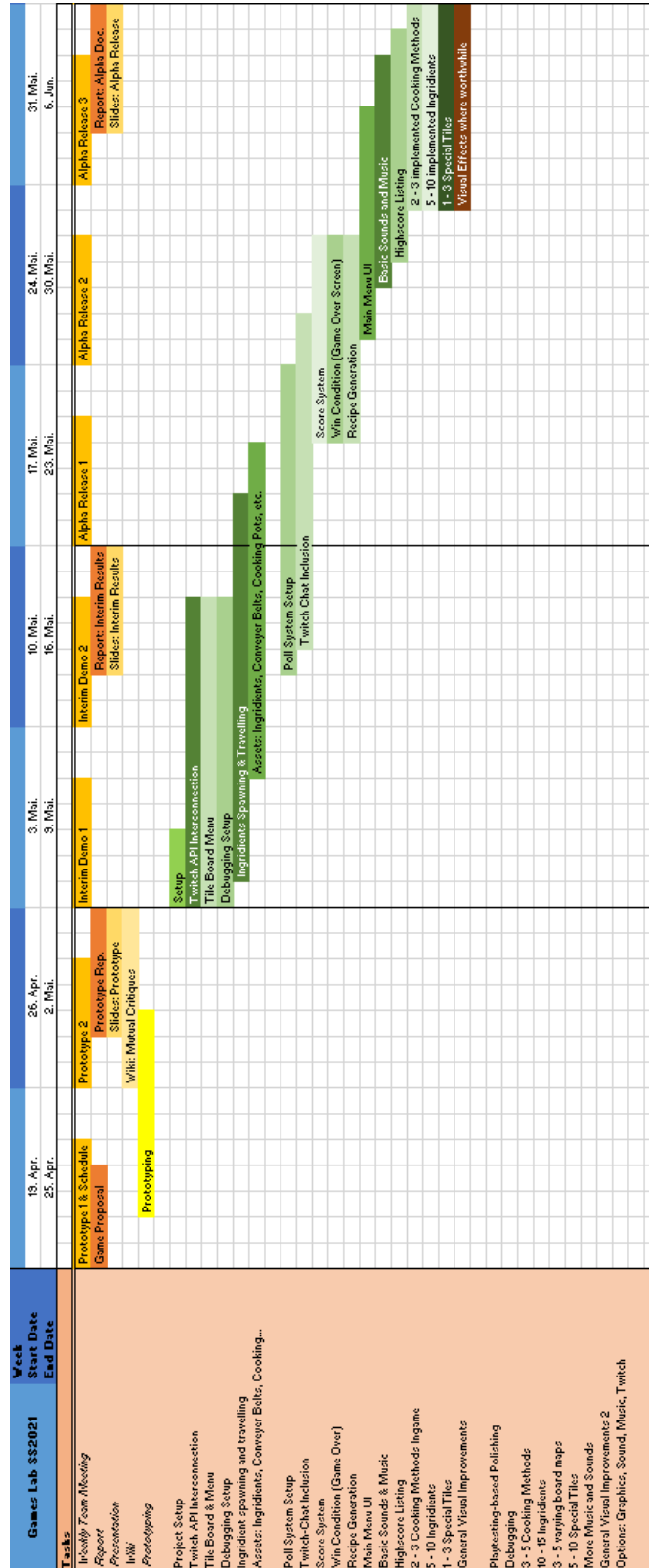


Figure 6: Waterfall Timeline until Alpha Release

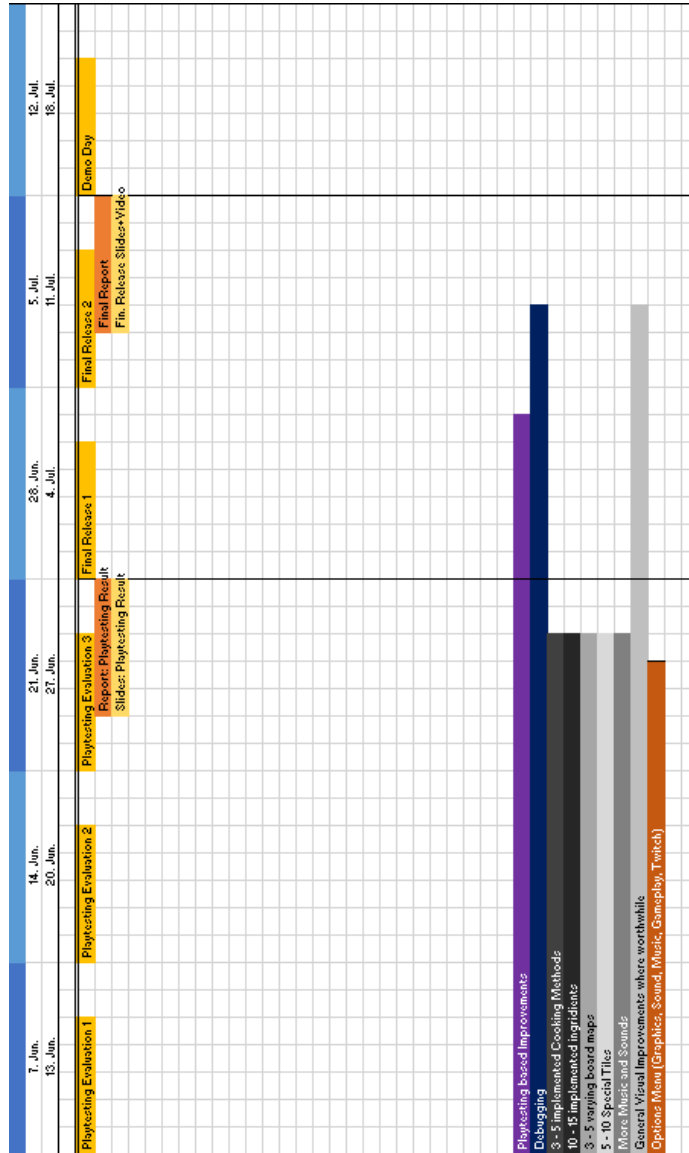


Figure 7: Waterfall Timeline after Alpha Release

Task Assignments												
Dorota Ingredient spawning and travelling Assets: Ingredients, Conveyor Belts, Recipe Generation 2 - 3 Cooking Methods: Ingame 5 - 10 Ingredients Playtesting-based Polishing Debugging 3 - 5 Cooking Methods 10 - 15 Ingredients More Music and Sounds	Ingredient spawning & Travelling Assets: Ingredients, Conveyor Belts, Cooking Pots, etc.	Recipe Generation										
	2 - 3 Implemented Cooking Methods 5 - 10 Implemented Ingredients											
	James Project Setup Twitch API Interconnection Twitch-Chat Inclusion Highscore Listing General Visual Improvements Playtesting-based Polishing Debugging 3 - 5 varying board maps 5 - 10 Special Tiles General Visual Improvements 2	Setup Twitch API Interconnection	Twitch Chat Inclusion									
				Highscore Listing								
												Visual Effects where worthwhile
	Georg Twitch API Interconnection Poll System Setup Basic Sounds & Music General Visual Improvements Playtesting-based Polishing Debugging 3 - 5 varying board maps 5 - 10 Special Tiles General Visual Improvements 2	Twitch API Interconnection	Poll System Setup									
												Basic Sounds and Music Visual Effects where worthwhile
	Rasoul Tile, Board & Menu Debugging Setup Assets: Ingredients, Conveyor Belts, Score System & Win Screen Main Menu UI 1 - 3 Special Tiles General Visual Improvements Playtesting-based Polishing Debugging 3 - 5 Cooking Methods General Visual Improvements 2 Options, Graphics, Sound, Music, T.	Tile Board & Menu Debugging Setup	Assets: Ingredients, Conveyor Belts, Cooking Pots, etc.	Score System & Win Screen	Main Menu UI							

Figure 8: Task Assignments until Alpha Release

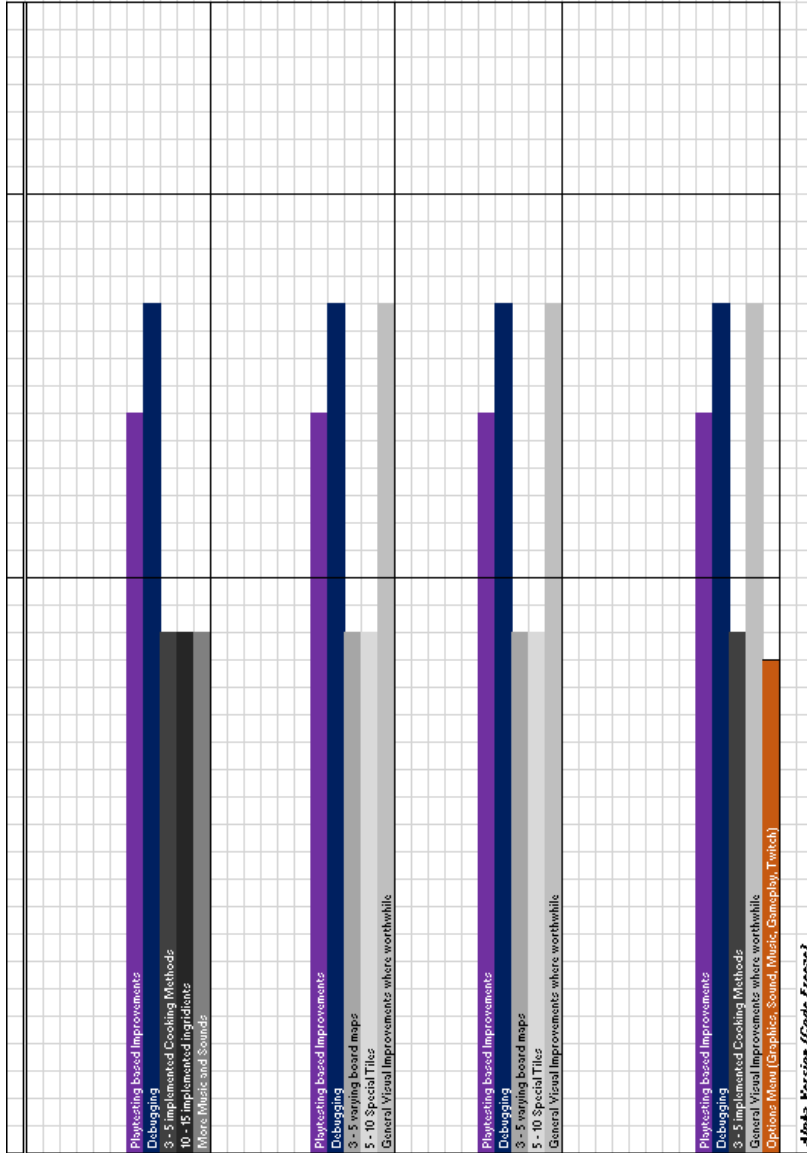


Figure 9: Task Assignments after Alpha Release