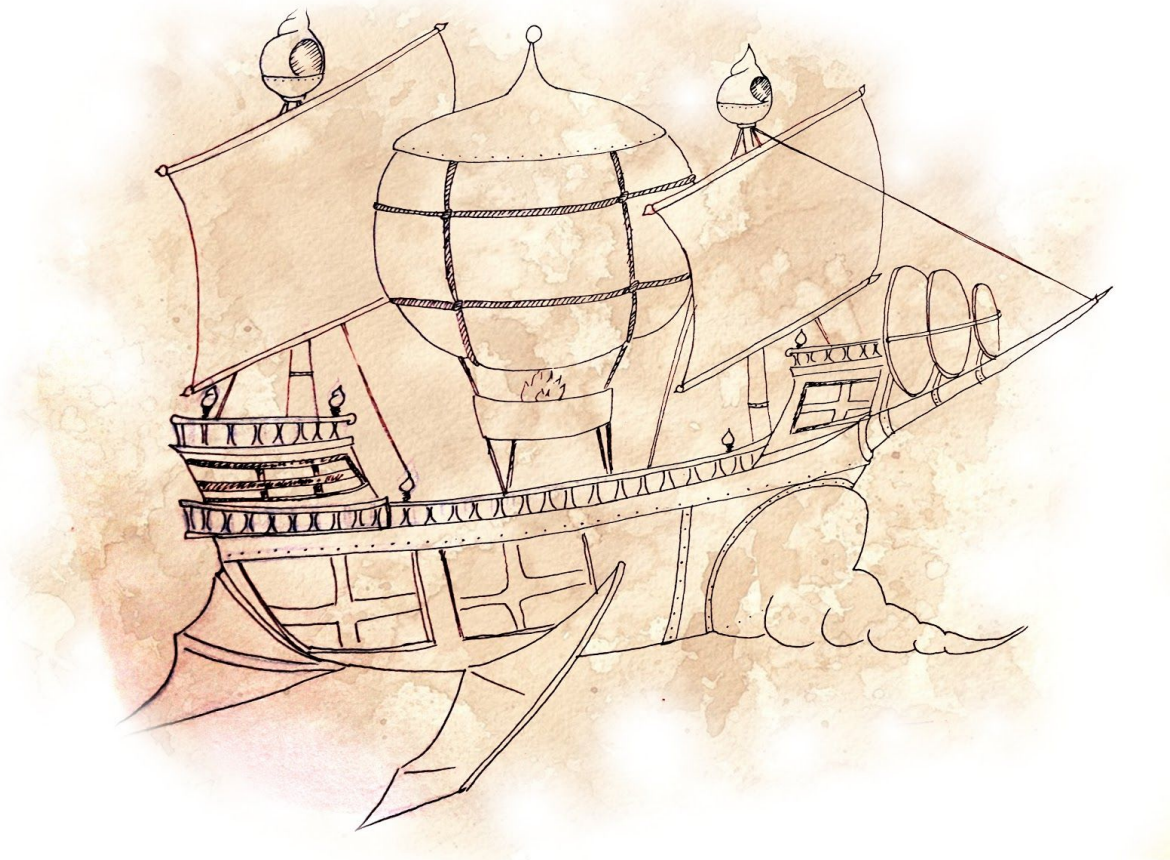# Conclusion

for

Master of Tempest

A game made by The Twisted Trash Pandas

Nikita Fetisov
Evgenija Pavlova
Moritz Schöpf
Maximilian Werhahn

Computer Games Laboratory 2018/2019

Table of Contents

1. Summary

1.a Final Game

Master of Tempest is an asymmetric, cooperative game in which two players have to fulfil their tasks and communicate in order to achieve their goal - reaching the core of "The Tempest". However, there are lots of dangers along their way. They have to navigate a flying ship through the tornado by avoiding various dangers such as rocks, parts of huge stone walls and communication- or manoeuvrability-prohibitive areas. Only then, they can avert the imminent apocalypse of this world and truly call themselves masters of The Tempest.
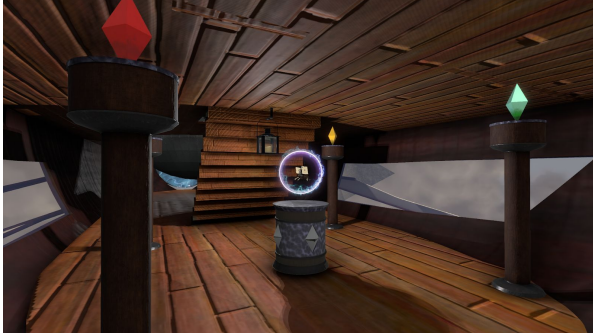
1.b Adjustments since Alpha Report

Player Controls

For both players, we added a small dot in the middle of the screen, functioning as a crosshair. That way they can teleport or target interactable objects more accurately.

For the wizard, we adjusted the way the player is able to cast spells. Instead of having to run around the altar to insert the element charge into one of the power sources he can now press the keys 1 to 4 to target the corresponding source - starting from the northern one with the key 1 and then going clockwise. Additionally, the charging time was reduced drastically.

The apprentice's gameplay did not change too much. The only adjustment we made is to lower the repairing speed. In the alpha release the player could repair all ship parts in one repairing area, whereas now, the player only repairs the most damaged part at first.

Ship Movement & Controls

In the alpha release version, it was quite hard to tell in what way the ship was moving or will move after casting a certain spell. Additionally, it was moving with and along the tornado rotation. Instead of having this constant movement around the centre of the tornado, we implemented a more linear type of movement. The ship is now moving forward at a slow speed and can be controlled more accurately by the spells, for example, it can be turned exactly 90 degrees around the local up-vector. Furthermore, we added trails to the ship's crow's nests and vertical sails. That way, when turning or moving vertically, it is clearly visible in which direction you go. Also, the ship slightly rotates around the local z- or x-axis depending on the spell cast.

Magical Crystal

Adding a magical crystal, which is always visible, as the target at the centre of the tornado helped immensely in terms of improving the feeling of movement in relative to the environment.

Environment

We added fog not only to the rocks but also to the ocean and skybox to improve the way the players perceive the distance and size of rocks and dangers. Additionally, the rocks were slowed down greatly so it is easier to avoid them.

Sound

For the overall feeling of our world, we added background music and sound effects for the wizard and apprentice.

2. Experience

Overall, we are happy with the final state of the project. The asymmetrical cooperative gameplay feature was brought into life, and we've received enough positive feedback during the demo day and the alpha stage to say that it's a promising idea that people like. Almost all of the initial ideas about the setting and gameplay mechanics have found their way into the final product, except for the few that we had to exclude due to the time constraints. Such were the ideas like having world objects damage the apprentice directly or having other than navigation control spells. Also, it would take much more time to balance the game properly and to adjust the learning curve, to make sure that the game is both fun after playing multiple hours, but is not too challenging for the new players.

The development schedule that was created in the first stage was used more as a guideline and not as an actual schedule. It helped us to assign milestones to the actual points in time so that we were

able to see how good (or bad) we're doing on the global scale. Instead of making a schedule at the very beginning of the project and then inevitably failing to stick to it, we chose more agile approach with weekly (more frequent at the times of very intense development) calls, during which we were deciding what features should be implemented in the upcoming days, and then distributing tasks based on our personal preferences and areas of expertise. Our process resembled scrum methodology in some sense, however, it seems like it's virtually impossible to follow a "proper" scrum in a student project, because the number of hours people could spend on the project varied from week to week. The biggest deviation from the schedule for us was bringing the environment and the players' actions together. Both the environment and the player controls logic took us more time to implement than we initially anticipated, with changing the wizard's gameplay completely after the interim presentation. This led to the actual gameplay runs and tests being very late.

The prototype stage helped us to put our finger on the core gameplay idea, to figure out what makes our game unique and interesting, its strength and weaknesses. It is hard to estimate how much it has helped us, and whether or not we could've reached the same conclusion without investing our time into making the "real world" prototype. However, it might very well be that we could've spent time more efficiently without building the prototype and focusing on eliciting the design decisions instead because most of the time during the prototyping stage was spent thinking about how to make it work with real-world objects and the results were not transferable to the final product.

The playtesting stage was extremely helpful, however, we would've received more useful feedback if we could've postponed it by a week. When we went into the alpha release stage we had certain problems in the game, that we were aware of, and that dragged in too much of the players' attention. But even with a major problem like "players don't feel like they're influencing the game flow" in the way, the critique from the players was extremely valuable and helped us to look at our game under a different angle. It is unfortunate that the time constraints of having to create a project during one semester don't leave enough time for multiple playtesting iterations, as it is extremely satisfactory to watch how the game grows and becomes better based on the potential players' input.

From the organisational point of view inside the team, everything went very smooth. We didn't have interpersonal conflicts from the very beginning of the project to the end. The personal interests of the team members didn't overlap too much, the tasks distribution always happened without arguments, and everyone was content with the work they had to do.

3. Course personal impressions

Q: Did it [the course] meet your expectations?
Evgenija: Yes, it was exactly what I imagined
Max: Yes, even though I did not necessarily have expectations.
Moritz: Yes. we did what I expected.
Nikita: Yes, it really did provide an experience of building a game from scratch

Q: Are you happy and proud of your game?
Evgenija: Yes, I believe we managed the task at hand as good as we could have in the limitations of time.
Max: Yeah, overall it does feel like a complete game to play for one or two rounds and it is also fun to "role-play" as the corresponding characters.

Moritz: Yes, the game turned out very fun and it doesn't have many issues considering the complexity of the game.

Nikita: Yes, considering the amount of time we had at our disposal.

Q: Do you feel there wasn't enough time or that the schedule was too compressed?

Evgenija: Winter semesters are always hard because of the lack of time. However, the time between the interim report milestone and the alpha release seemed too short

Max: I think there should've been more time for the playtesting phase, so it is actually possible to improve the game multiple times via the gained feedback.

Moritz: There should have been a bit more time before the interim demo.

Nikita: It does seem like the schedule is too compressed: the time dedicated to the active development is very short, and the playtesting stage way too short to properly reflect on the received feedback.

Q: What was the biggest technical difficulty during the project?

Evgenija: Networking and creating the right movement of the ship

Max: Networking and creating an immersive feeling of the environment. (+ Balancing)

Moritz: Networking and the visuals for the environment + balancing

Nikita: Networking and the environment visuals

Q: What was your impression of working with the theme?

Evgenija: It was hard to think outside the box. The theme itself is defined through too clear definitions.

Moritz: A theme is nice and makes sure that we focus on one thing. But it would also be nice to have even tighter requirements.

Nikita: Having a theme helped our new team to start a conversation about the game ideas. Without a common theme from the beginning, it would've been harder to decide on the game idea to start actual work.

Q: Do you think the theme enhanced your game, or would you have been happier with total freedom?

Evgenija: Yes, it did. The theme was challenging, but it concentrated our thoughts for the game in one direction. I prefer theme-based projects.

Max: I think it is better to orientate along a certain direction otherwise you might get lost if you have too many ideas. So yes, it enhanced our game.

Moritz: The game is completely based on the theme! Its harder to be creative without any requirements I think.

Nikita: The game is inspired by the theme, so it helped us a lot.

Q: What would you do differently in your next game project?

Evgenija: Invest more time into the game right at the beginning. It would have been more relaxing in the end.

Max: Working on gameplay prototypes first would've been helpful to see which way of casting spells or other controls would be best.

Moritz: Thinking about the game design a lot more.

Nikita: Spend more time on the game design discussions, until everyone agrees and there are no questions left.

Q: What was your greatest success during the project?
Evgenija: The unique gameplay (although it still needs balance); the "look and feel" of the game (the storm looked amazing!); The good working network
Max: The asymmetric, coop gameplay is really fun + visuals + networking.
Moritz: Steam Networking implemented completely by us down to the bytes!
Nikita: The network element is definitely the strongest point of our game: even with not the best connection quality the game plays fine.

Q: Are you happy with the final result of your project?
Evgenija: Yes, although it would have been nice to rethink and retest the game a few times, to polish game and gameplay (eg. gameplay of the wizard)
Max: Yes.
Moritz: Yes
Niktia: Not happy, but the result is good enough all things considered.

Q: Do you consider the project a success?
Evgenija: It was up and running at the demoday (with only few problems appearing), so yes!
Max: Yes, apart from a few more or less game breaking bugs it was fine. People seemed to enjoy it.
Moritz: Sure
Nikita: Yes.

Q: To what extent did you meet your project plan and milestones (not at all, partly, mostly, always)?
Evgenija: The hardest milestones were the alpha release and the interim report. I believe we had a slow start, but we managed at the end.
Max: Probably the desirable target/interim report was not fully achieved. Few things like background visual effects or basic sound were missing. Other than that it went quite smoothly.
Moritz: Most of the milestones were met!
Nikita: The interim report milestone wasn't quite reached, but we've managed to catch up over the Christmas break.

Q: What improvements would you suggest for the course organization?
Evgenija: No suggestions that come to my mind.
Max: Better planning for the separate stages(maybe more time in later stages, especially playtesting).
Nikita: It would be nice to have more time after the playtesting stage to be able to go through 2 iterations and see how the feedback changes. Also, I'm not convinced by the real-world prototype assignment: with the time issue that we already have, it feels like an impediment. Also, it doesn't make sense to ask students to prepare a schedule with tasks week by week by person for 3 months in advance. It would never work, neither in student projects nor in the industry. This approach might be useful for well-specified and typical

projects, but even for those projects, it's not always advisable. As for our situation, we had this assignment before we even went into the "Prototype" stage, and we didn't have the gameplay specified yet. Considering that we also wanted to try things we didn't use before, our time estimates could not have been precise at all. Also, application architecture and application design were not touched in the course. I think it would be better if for the development schedule it was enough to set up milestones with describing functional minimum, desirable, etc. requirements and drop the real-world prototype task, and instead focus more on the game design and application design. It would've been interesting to see how other teams designed their game from the software engineering perspective, not only game design decisions.