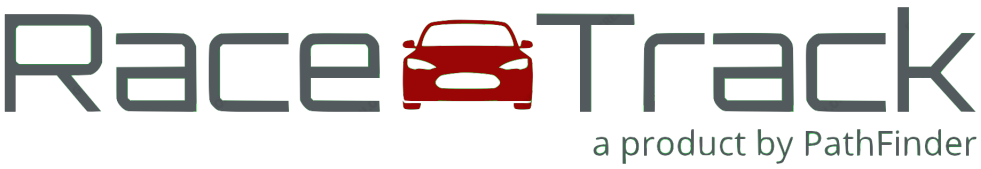
ZHAW Zürcher Hochschule für Angewandte Wissenschaften



Autoren

Marco Forster
forstma1@students.zhaw.ch

Manuel Berweger
berweman@students.zhaw.ch

Marvin Tseng
tsengmar@students.zhaw.ch

Dan Hochstrasser
hochsdan@students.zhaw.ch

05. März 2020

Project Outline

Contents

1 Baseline

In today's world, almost anything is being digitalised. The startup **PathFinder** wants to bring childhood memories and nostalgic game experiences to the digital age by digitalising the popular paper and pencil game Racetrack. Not only can PathFinder make a good profit from this game, but it can also provide a digital alternative to the original paper and pencil game and by that reduce the general paper waste.

2 Idea

The underlying idea of **RaceTrack** is a round-based racing game where the player has to strategically choose his/her next move out of multiple reachable positions. These positions are calculated based on the previous turn of the player. By relating the diffenerent turns, the game gets interessting and challenging. The player will be needed to demonstrate all his strategical skills and carefullness to beat his opponent. The game can train different players foresight thinking and basic understanding of vector mathematics.

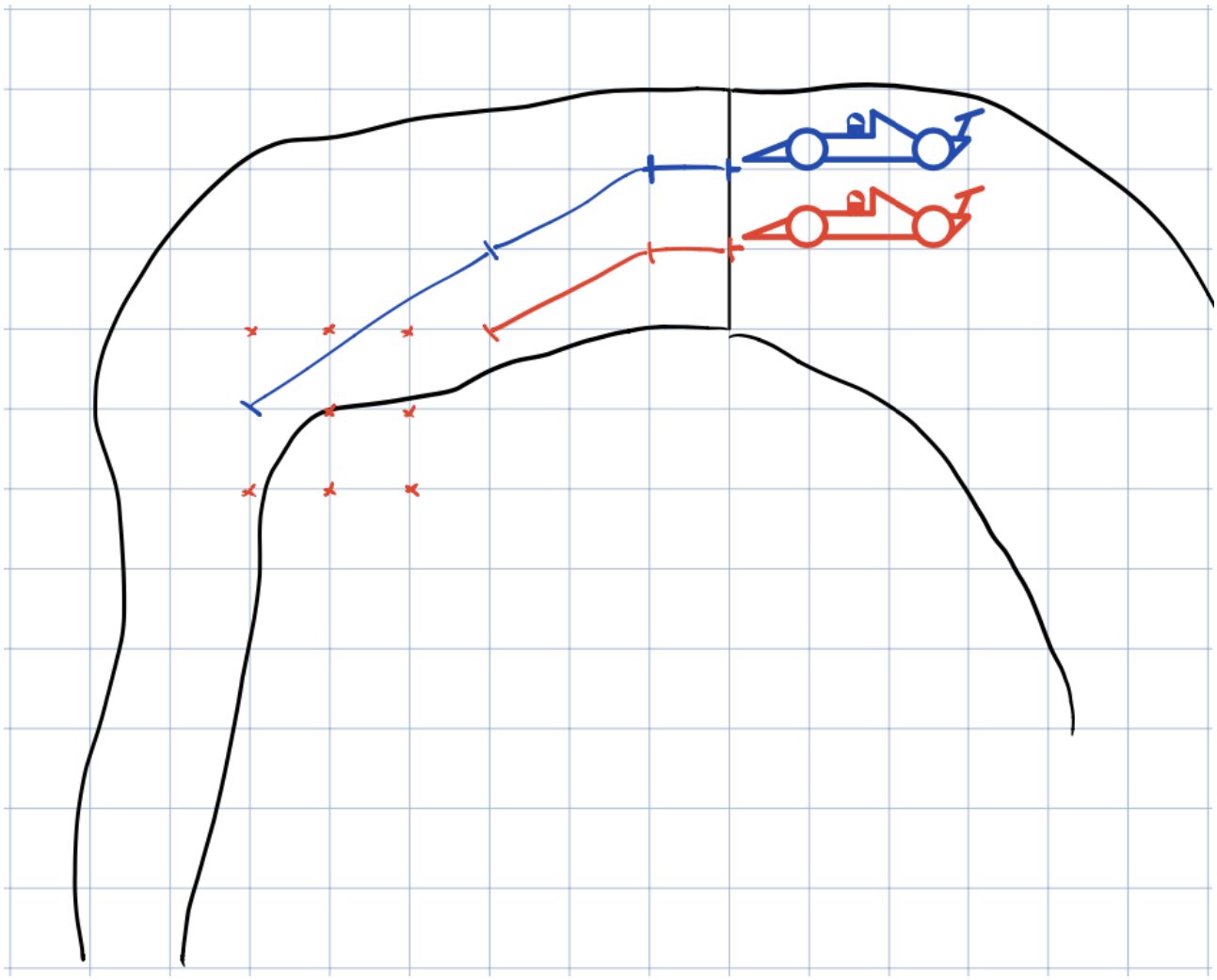


Figure 1: A turn in the Game RaceTrack (Mock-up)

3 Client Benefits

RaceTrack brings multiple benefits to its users/players:

- The player can play the game either with friends or alone.
- No paper is needed and wasted. Instead, larger computer screens ensure better visibility than on a sheet of paper.
- A game can be restarted with the same settings (e.g. same number of players, same track, same cars, etc.)
- The players can compare their scores with previous players
- There's also an educational aspect where players can learn about the concept of vectors and vector addition:

- The player can train his reasoning abilities by playing the game and see dependencies with other vectors.
- It can even help to improve decision-making skills.

4 Competition Analysis

Research has shown that the game Racetrack, also known as Vector Race, has never been realized as a computer desktop game [1] [2]. Currently, two Android applications with the same concept as RaceTrack are distributed on the Google Play Store [3]. However, these are not suitable for multiplayer or educational purposes, as the screen size is limited by the smartphone.

Thanks to the implementation with the programming language Java, RaceTrack is platform-independent. This gives the user the advantage of running RaceTrack game on Windows, macOS or Linux. In addition, a larger audience is reached and therewith a larger potential of paying customers.

RaceTrack distinguishes itself by adding the option to add special items to the track, e.g. boosts or obstacles. This feature can be activated during the creation of the game round.

5 Main Use Case

//TODO add additional rules if any are missing or added

The main use case is the player, or a group of players playing a round of RaceTrack:

1. On his computer, the player starts the application, which he already downloaded and installed.
2. After the main menu has finished loading, the player starts a new game by selecting the corresponding entry in the menu.
3. The player configures his game session with his options, e.g. which track, his car, how many players etc.
4. After the game has started, the player is able to make his first turn.
5. The Player chooses his move for his current turn, based on the given possible moves the player can take. The possibilities are being shown on the track.
    - The player's possible moves are calculated at the beginning of his turn, based on his current velocity and last moves.
6. The player's car moves to the selected position on the track

As a turn-based game, every other player needs to do his turn first, before a player gets the ability to do his next turn. Each player repeats steps 6-7 until i'ts the first player's turn again.

9. The game continues, turn after turn, until the first player has reached the finishing line.
10. While playing the game, following rules are being enforced:
    - You can not drive backwards.
    - You can not drive to a position already occupied by another player.
    - You can not drive past the track limits, doing so will result in a crash in which the crashed player get's disqualified.
11. After the first player has reached the finish line, the game can be continued until every other player reaches the finish line or it can be finished at that moment.
12. After the session has finished, the player's results are displayed in comparison with the track's highscore.
13. The player can choose to return to the main menu, restart the session with the same settings or to end the game (application) completely.

6 Additional Requirements

//TODO edit additional requirements may be needed

- It should be possible to save the current game session, without losing any game progress.
- It should be possible to load a saved game.

- It should be possible to pause the current game session.
- It should be possible to restart the current game session.
- It should be possible to play the game alone or in a group of up to four people.
- When crashed, instead of disqualifying a player, the player's velocity resets and starts his next turn from the same position. This should be an option.
- After the first player reaches the finishing line, the game should give the option to end the session now or let it play out, until every player has reached the finishing line.
- After the game has ended, it should be possible to start another session with the same options.
- There should be at least two different tracks to choose from at release.
- There should be the option to activate special items to the track, which places special items like boosts or obstacles on the track.
- Following requirements needed to be implemented after the initial release:
  - The ability to buy additional tracks and cars (Microtransactions) from the official store
  - The store needs to be fully secured.

7 Ressources

The development of RaceTrack will be split between 4 programmers. Every member of the development team needs to have advanced knowledge in programming with Java. With the use cases, it is possible to split the tasks and resources between the developers. Also, knowledge in vector mathematics is needed to know possible directions for the car in the next turn.

Every developer needs a working computer to build and test the game. Git is used as a version control system and to track the project status. 400 to 500 man-hours are projected to be needed to develop the game and manage the project.

8 Risks

Since we are an emerging startup, we do not offer the same expertise as established game development companies. In addition, with a game like RaceTrack, the rules of the game must be well known to the player to ensure the fun factor.

9 Project Draft

The first release with basic functionality will be given a development period of twelve weeks. The development occurs iterative and incremental in accordance with the Unified Software Development Process. Following use cases and risks have been evaluated in a first round of evaluation:

9.1 Use Cases

//TODO edit use cases after new use cases have been defined

- Use Case UC1: View Highscore
- Use Case UC2: View Rules
- Use Case UC3: View Credits
- Use Case UC4: Set Game Options (after starting game)
- Use Case UC5: Start new Round of Game
- Use Case UC6: Reset Round of Game
- **Use Case UC7: Play one Round of Game**
- Use Case UC8: Exit Game
- Use Case UC9: Compare with Highscore (after one Round of Game)
- Use Case UC10: Load Game
- Use Case UC11: Save Game
- Use Case UC12: Start Game

9.2 Risks and Prevention

//TODO add additional technical risks if any come up

- Implementation of GUI unfamiliar

- Used programming language (Java) not optimised for game development
- A developer will assure that he gains the necessary knowledge of Java graphical user interface (GUI) development. He may or may not use a GUI Framework.

9.3 Rough Timeline

//TODO Edit timeline with real use cases with importance

400 - 500 man hours are estimated for the creation of the first version. The game will be developed in iterations of two weeks during a development period of twelve weeks after following timeline:

Phase	Iteration	Start/Duration[Week]	Goals
Inception	1	1 / 2	Project draft created, use cases defined, domain model drafted, development pipeline set up finished, architecture drafted
Milestone	**M1**	**End of week 2**	**Vision, goals and requirements defined**
Elaboration	2	3 / 2	First half of use cases specified in detail, other requirements identified and prioritised, UI prototype present, architecture stable
	3	5 / 2	Second half of use cases specified in detail, domain model finalised, architecture stable and as PoC verified
Milestone	**M2**	**End of week 6**	**Architecture verified**
Construction	4	7 / 2	Use cases for iteration realised and tested, UI prototype implemented
	5	9 / 2	Use cases for iteration realised and tested, UI updated
	6	11 / 2	Use cases for iteration realised and tested, UI updated
Milestone	**M3**	**End of week 12**	**First version realised and released with tests and documentation**

10 Profitability

With the required man-hours multiplied by the salary, the development costs of the application RaceTrack is estimated at 60,000 Swiss Francs. Additionally, there will be expenses of 1'000 Swiss Francs for licenses and infrastructure. Since our game can be downloaded for free, our income is generated by microtransactions. In future releases, players will be able to purchase add-ons such as new race tracks or additional cars with striking looks. The prices for these add-ons range from 2 to 15 Swiss Francs. With a predicted user base of 30'000 users and average revenue of 2 francs per installation, we expect the game to become profitable within a year.

11 Glossary

	Description
Microtransaction	Business model where users can purchase virtual goods with micropayments
UML	Unified Modeling Language - a general-purpose, developmental, modeling language
Git	Distributed version-control system for tracking changes in source code during software developmen
POC	Proof of concept - method or idea in order to demonstrate its feasibility
Add-on	In computing, a sofftware component that adds a specific feature to an existing computer program

12 References

Number	Reference
[1]	Microsoft. (2020). Microsoft Store - Search Results [Online]. URL: https://www.microsoft.com/en-us/search?q=vector+race [Accessed 28. Feb. 2020].
[2]	Apple. (2020). Games - Mac App Store Downloads on iTunes [Online]. URL: https://apps.apple.com/us/genre/mac-games/id12006?mt=12 [Accessed: 28. Feb. 2020].

Number	Reference
[3]	Google. (2020). Google Play - Search Results [Online]. URL: https://play.google.com/store/search?q=vector%20race [Accessed 28. Feb. 2020]