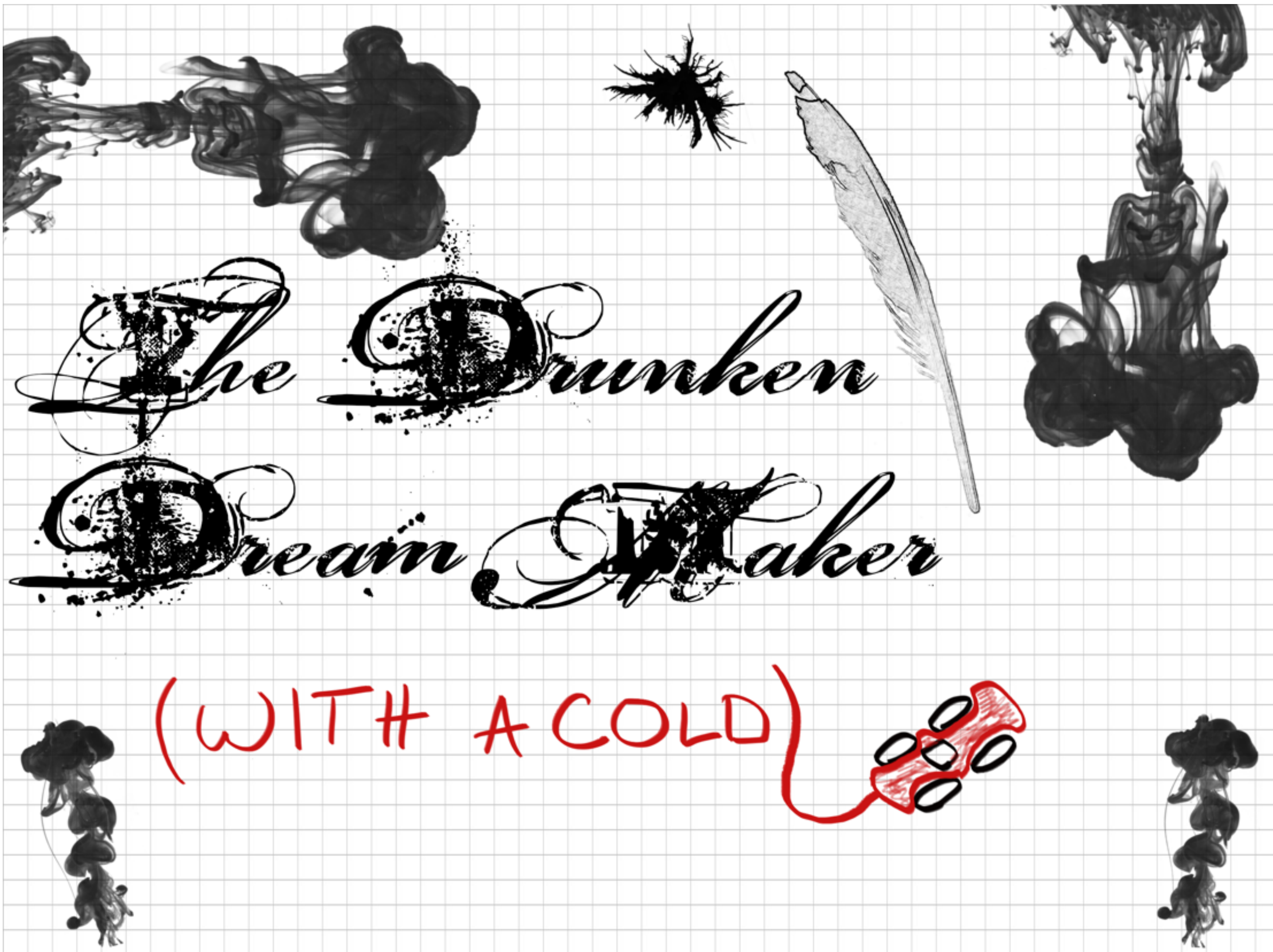
Triolozzi proudly presents



Riccardo Roveri, Vittorio Megaro & Emanuele Rudel

ETH Game Programming Laboratory
Spring Semester 2013

TABLE OF CONTENTS

Introduction

Our physical prototype is a turn based board game playable by 2-4 players, where one of them acts also as the "computer". Our goal was to model the following fundamental elements which will represent the main game mechanics of our final game:

- Players have to provide sleep quality to their associated child
- Players drive on a track
- Players can draw obstacles
- Being too far away from the player leading the race means losing sleep quality
- A match is composed of a series of mini-races along the track
- From time to time a REM phase occurs where players have to draw accurately

Figure 1. Final physical prototype

Equipment

Figure 2. Equipment

•One main board (*Track board*) with hexagonal tiles covered with transparent plastic

•One additional board (*REM phase board*) for each player, with hexagonal tiles and covered with transparent plastic as well

•2-4 erasable markers of different colors, depending on the number of players

•2 dice

•*Sleep quality cards* (see Figure 3)

•*Rem cards* (3 *REM phase cards* and 2 *no REM phase cards*) (see Figure 3)

•*Nightmares sticky notes*

•*Camera sticky note*

Figure 3. REM cards and Sleep quality cards (hint: Zürich trams usually have lots of these...)

Rules

The following rules have been decided after many modifications and attempts, and represent the best solution we found in order to make our game as funny and playable as possible.

Sleep quality and winning conditions

Players start the game with an initial sleep quality of some default units, described in the table below. A sleep quality unit is represented by a *Sleep quality card*.

Number of players	Initial Sleep quality cards per each player
2	3
3	6
4	9

A player that has no *Sleep quality cards* left is eliminated from the game.

In particular, the game ends when at least one of the following conditions is met:

- 5 races have been completed;
- all but one player have no *Sleep quality cards* left;
- a player crosses the finish line in the *Track board*.

In all these cases, the winner of the game is the player who collected more *Sleep quality cards*.

Mini-races

The main part of the game happens on the *Track board*, where players challenge each other in a series of mini-races.

The *Track board* represents a fixed track defined by some obstacles. An obstacle is a set of tiles colored in black against which all the players can collide. The borders of the track act as obstacles as well.

The position and the movements of each player are marked during the whole game by drawing on the board with erasable markers of different colors.

At any point during the game, players must be inside a hexagonal tile and in one of the states listed below:

- still;

- driving;
- accelerating. In this state a player has a direction *d*.

In a turn, the player first decides his or her next state. It is not allowed to go from the state "still" to the state "accelerating" in a single turn (without passing from the "driving" state). Then, the player rolls one die if in the still or driving states, two otherwise. The total sum of the dice determines the number of movements, where one movement is defined by shifting the player's position from one hexagon to an adjacent one. The player can move in any of the six directions except the one from which he or she came.

A player who was in the accelerating state in the previous turn must perform half of the movements (rounded up to the next integer) in the direction *d* given from the previous turn. The remaining movements can be performed in different directions.

If a player collides against an obstacle or another player, he loses the remaining movements and goes in the still state.

The players must keep track of their position and movements on the board in the following way:

- When a player visits a hexagonal tile while moving, must color the cell by drawing a line inside it with his or her color (Figure 4).
- When a player stops in a cell, he must draw his pawn inside the cell, depending on the current state. Figure 5 shows how the pawn must be drawn in each state.

Figure 4. A player's trail.

Every time a player lands on a tile colored by a different color than his or her own (there could be multiple colors), the player can jump in one of the five allowed directions if and only if the jumped tile and the landing tile are also colored with a different color than the player's. A jump is equivalent to two movements in the same direction but only accounts for one, giving thus a bonus move to the player. The jumped tile is of course colored with the player's color. This rule was introduced to model the speed up gained from following somebody else's trail that we want to have in our final game.

If a player goes back on a tile that he or she has already colored, that tile becomes an obstacle filled with the player's color. Moreover, when a player intersects his or her own trail (drawing thus a closed shape), the tiles on the border and inside the polygon become an obstacle filled

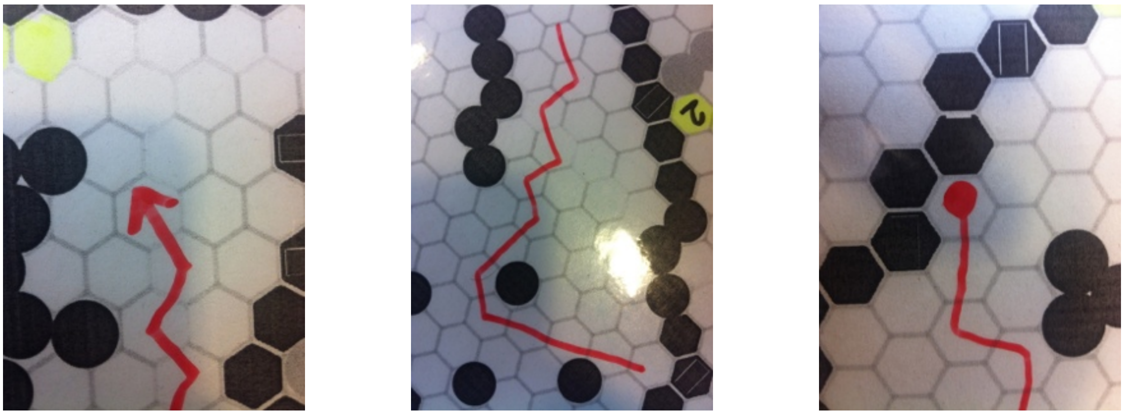


Figure 5. Accelerating state, driving state and still state.

with the player's color. It is important to notice that the shape becomes an obstacle if and only if no other player lays on one of the shape's tiles. An obstacle created by a player may collide with other players but it does not interact with the player who created it. In order to make the obstacle recognizable the player has to fully color all the tiles belonging to it with his color, as shown in Figure 6.

Figure 6. An obstacle created by the red player.

Every time a player leaves a shape created by himself (for example, right after having drawn it), he gains one additional movement in the current turn. More formally, a player leaves a shape when moving from a tile belonging to an obstacle with the player's color to a tile that does not belong to it. This allowed us to model the speed up gained from closing a shape.

The track initially contains trails made up of one or more grey tiles. Players can exploit those trails in order to create obstacles with their own color: they can indeed draw a shape which is composed of both grey and colored tiles. The only difference is that once a player leaves a shape created with the help of grey trails, he does not gain the additional movement specified above. In the same way as explained above, the shape will become an obstacle of the player's color if and only if no other player is inside the polygon.

The track is marked by lines that act as checkpoints at regular intervals. They are needed in order to establish who is the player leading the race and to determine the starting points for the mini-races.

At the beginning of the game, players draw their pawns (in a "still" state) on the first checkpoint, which represents the start line of the track, and take turns in the order determined by chance (e.g. rolling a die) before the game.

A checkpoint becomes the *reference checkpoint* whenever a player crosses the preceding one.

At the end of one turn, the player with the shortest path from his or her position to the *reference checkpoint* (avoiding obstacles and walls) is the one leading the race.

Players with a beeline distance of 7 or more tiles from the first player (including the tiles on which the players are) are eliminated from the mini-race and lose a number of *Sleep quality cards* defined in the following table.

Player eliminated ...	Lost Sleep quality cards
first	1
second	2
third	3

After each round, the player "computer" is in charge of calculating the distances and checking if any players have been eliminated from the current mini-race.

A mini-race ends when all players except the leader are eliminated, and a new mini-race begins right afterwards at the *reference checkpoint*. The pawns are drawn on the *reference checkpoint* in an order determined by chance.

REM-phase

At the end of each mini-race, the player acting as the computer picks a card from the *REM deck*. The deck contains 5 cards of two types: *no REM phase* and *REM phase*. If the former was picked, the game continues with a new mini-race, otherwise all players will play a REM phase before starting the mini-race.

The REM phase in the board game was introduced to model the accurate drawing part that we plan to implement in our final game.

In the REM phase each player receives a *REM phase board* on which we printed a yellow arrow 75 tiles long and placed some post-it notes with scary shapes (*Nightmares sticky notes,* they represent nightmares). Players are placed at the beginning of the arrow and their goal is to get to the end of it following its curves and avoiding nightmares.

On each turn a player rolls the dice according to the movement system defined above. The post-it notes act as obstacle on all the tiles on which they lay or touch. The collision with a post-it note activates the nightmare and causes a loss of one *Sleep quality card*.

During this phase players cannot eliminate each other from the REM phase since they play on different boards. On the other hand, the camera field of view now moves at a certain speed rate and players must stay in it in order to remain in the game. The camera field of view is represented by the movable *Camera sticky note* on the board (see Figure 7).

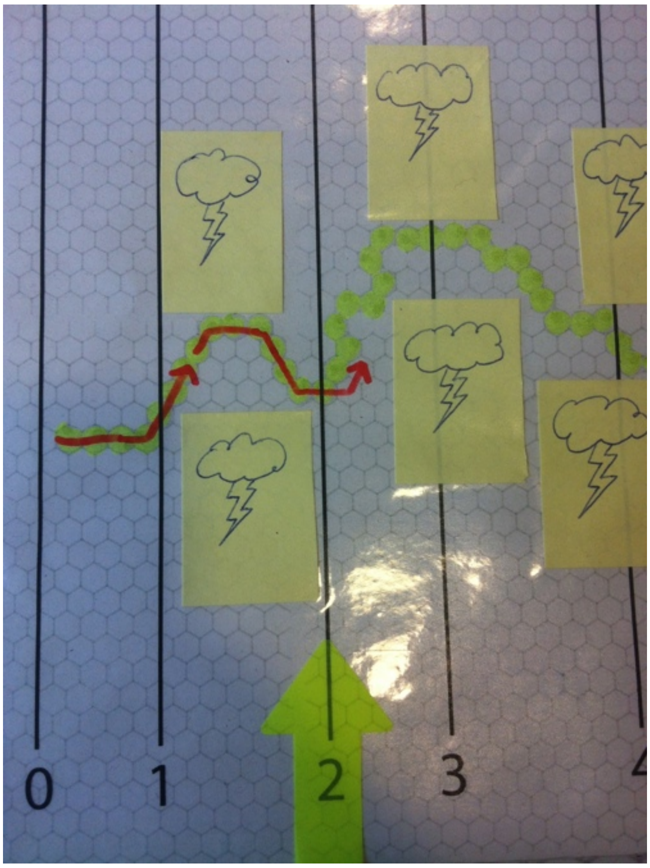


Figure 7. The sticky note camera on the bottom.

The yellow arrow is split in eight parts using nine vertical lines. The camera is always positioned on one of these lines starting from the first on the left and moving to the right once every two rounds. The computer player is in charge of moving the *Camera sticky notes* on all the boards. All players must be always at least at the same height of the current camera line, otherwise they will be eliminated from the current REM phase.

The REM phase ends when the camera reaches the last line and the players must count the sleep quality points obtained according to the table below.

Filled Arrow Tiles	Obtained Sleep Quality Cards
40-50	1
50-60	2

Filled Arrow Tiles	Obtained Sleep Quality Cards
60-70	3
70-75	4

After the REM phase, the order to draw the pawns and to roll the dice in the next mini-race is not determined by chance, but it depends on the order with which players reached the end of the arrow during the REM phase. In the eventuality that two players reached together the end of the arrow, the order is decided by rolling a die.

Physical Prototype iterations (i.e. how the game was created)

Fundamental Questions

In order to model the fundamental elements which represent the main game mechanics (listed in the *Introduction)* we asked ourselves the following questions:

- How can we model the race track?
- How can we design players movements so that the game resembles a racing game?
- How can the field of view of the camera be included in the physical prototype?
- How often should obstacles be drawn?
- How big is the influence of the trails left by the cars?
- How should we model the sleep quality to ensure the playability of the game?

Below are listed the steps that form the iteration process we went through in order to answer all the questions above.

Iteration Process

Iteration 0

Description:

We drew the track on a squared pattern sheet and we randomly placed some obstacles along the track.

We cut out a small window from another piece of paper to model the camera field of view. Its role was to follow the first player and eliminate the players that were not visible in the window.

Each turn consisted in rolling two dice and performing the amount of movements (square by square) indicated by their sum.

Analysis:
We realized that the obstacles were completely useless, since nobody collided with them, and the camera system was not practical to handle manually. Moreover, the game ended up being only a matter of pure luck, it could have been reduced to a simple coin-toss game.

Iteration 1

Description:
To make players collide with obstacles more often we introduced a reaction time phase before each player's turn. It consisted in forcing the player to move in his or her current direction by a number of squares given by a rolled die.

The camera system was drastically simplified and consisted in checking the distance between the first player and all the others. Every player farther than ten squares from the leader was eliminated.

Analisys:
We got good responses from the reaction time phase since some players were actually crashing against obstacles. The simplified camera system was also more satisfactory than the previous attempt.

Unfortunately, in general matches were too short. It was very likely that a player won a mini-race after just a couple of turns.

Refinements:
We reduced the number of dice from two to one, in order to avoid having short matches.

Iteration 2

Description:
To reduce even more the gap between the leader and the other players we introduced the concept of cars trails. Players following them could move faster.

Analysis:
With these modifications we got opposite results than the previous ones: players were almost never able to eliminate each other. On the other hand, the game was much more dynamic and interesting than the previous versions.

Refinements:
We changed the board from a squared pattern to a hexagonal pattern to allow moving in more directions and to give more freedom to the players.

We slightly reduced the boost gained from following a trail, giving advantage to the player only on straight trails.

Iteration 3

Description:
Two important changes were made in this step: the introduction of three states to better model the reaction time of the players and the possibility to draw obstacles (with or without the help of the grey stripes).

Analysis:
The three states improved a lot the whole playing experience. We are convinced that they are a good abstraction of what usually happens in racing games: when a player is accelerating he has more chances to collide with an obstacle than the ones who drive slowly and carefully.

The introduction of drawn obstacles represented an improvement to the game as well. The leader of the race was often able to force the elimination (or at least a collision) of the opponents following him.

Refinements:
Even though players could eliminate each other, this happened way too rarely and thus a match could last too long. To avoid this we added more obstacles, grey stripes and tight passages along the track.

Rem phase Iterations

At the beginning, players could not be eliminated by the camera (which was not even present), but rather were forced to drive only forward.

All the players ended up rolling always only one die (without accelerating) to be as accurate as possible, not caring at all about finishing the REM phase in a short time.

The introduction of the camera movement helped a lot in this sense, since the players finally started sacrificing some accuracy for speed, to avoid being eliminated by the camera.

Final Iterations

At this point we produced a robust physical prototype to play with and after playing many matches we tuned the sleeping quality parameters in the best way we could.

In general, after a lot of modifications we are now satisfied with the final result and we think that our game is finally funny to play. We realize that some of the rules may sound difficult to learn, but once assimilated they offer a wide range of possibilities to the players.

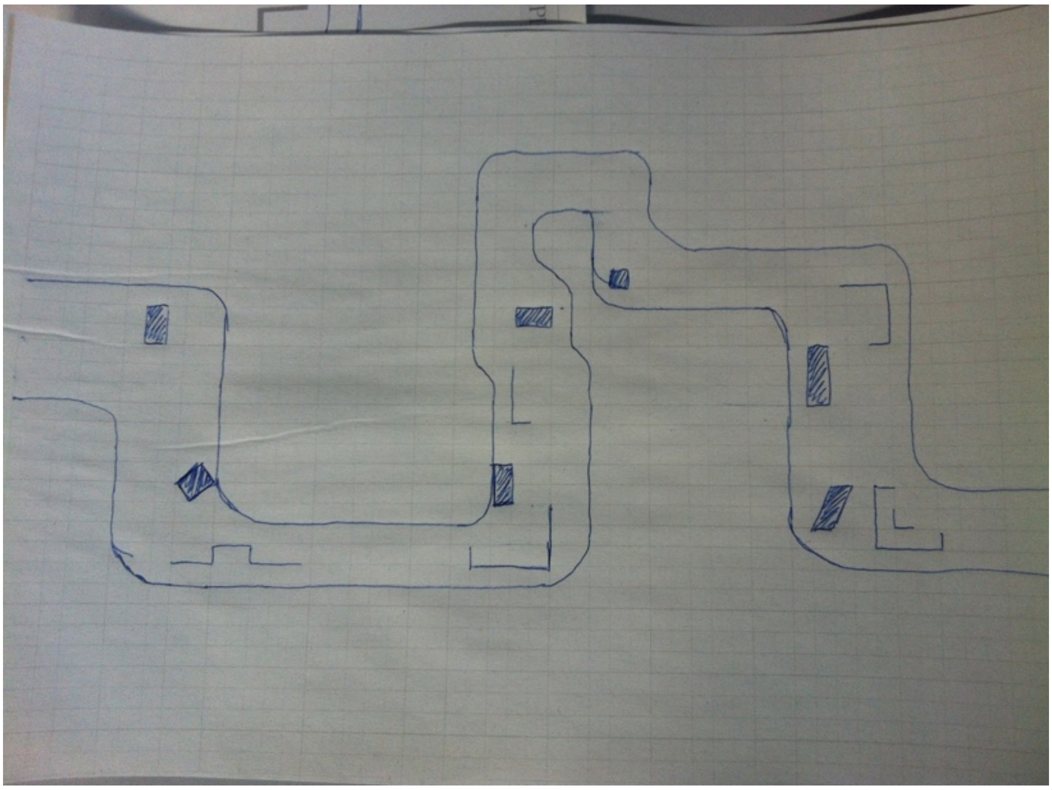
Conclusions

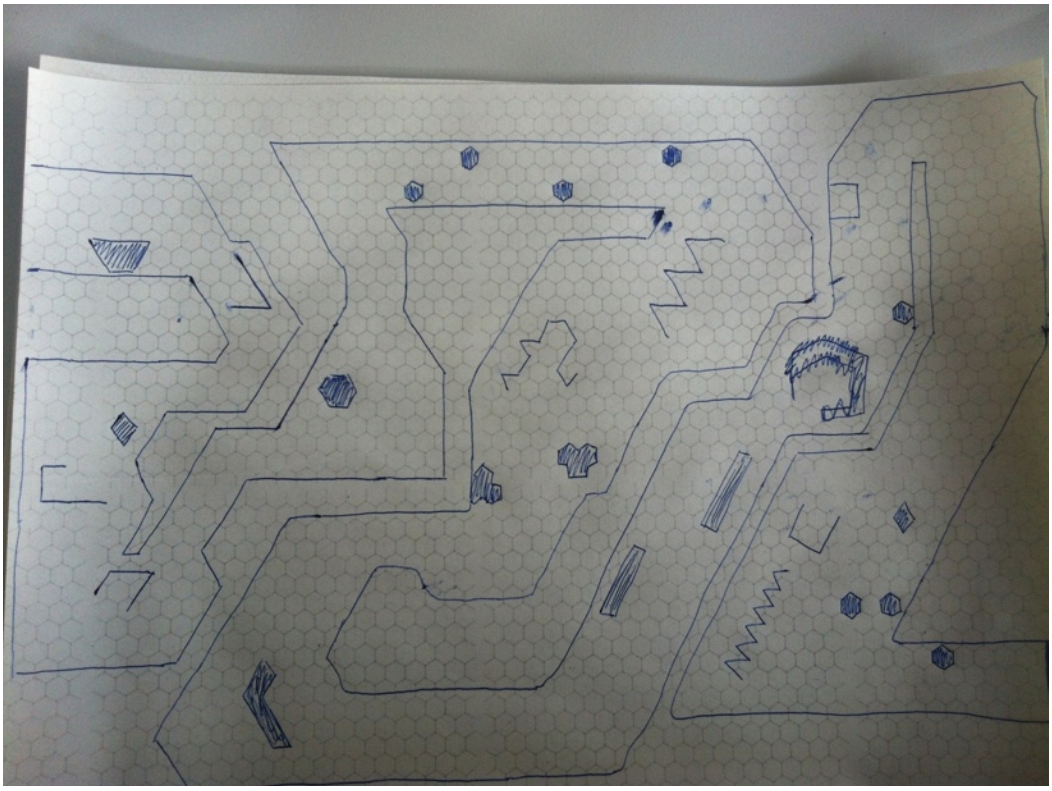
Strategic skills play a big role in this prototype. In our final game, though, we plan to give more importance to driving skills.

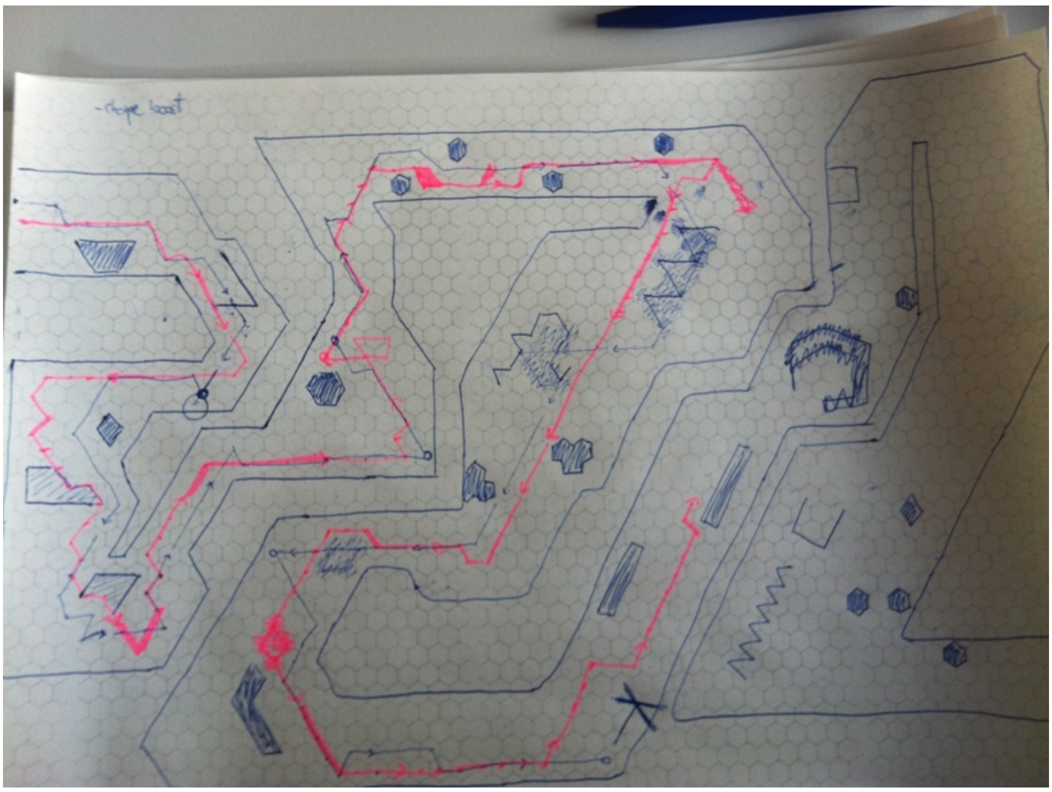
We remember a couple of quite epic and tense matches between the three of us, that brought out our competitive spirits.

We truly hope that we will be able to provide the same kind of experience to the players of our final game :)

Early game prototypes






-shape boat