



## Fruit Smashers: Physical Prototyping

Matej Hamas, Daniel Keyes, Nemanja Bartolovic, Delio Vicini

March 20, 2016

# Contents

1	Physical Prototype Overview	2
2	Design and Build Phase	2
3	Game mechanics	4
3.1	Car Driving . . . . .	4
3.2	Setup and Basic . . . . .	4
3.3	Player's Turn . . . . .	5
3.4	Health points . . . . .	5
3.5	Scoring . . . . .	5
3.6	Power-ups . . . . .	6
3.7	End of Round . . . . .	6
4	Lessons Learned	6
4.1	Scoring . . . . .	6
4.2	Level Design & Spawning . . . . .	6
4.3	Power-Ups . . . . .	7
5	References	7

## 1 Physical Prototype Overview

We spent approximately half day designing and building the physical prototype. This phase was followed by another half day of playing it, validating the game play and refining where necessary. In total, we spent approximately one full day on this task, excluding write ups.

## 2 Design and Build Phase

We started by discussing ideas on how to turn the game concept into a physical game. After validating some core mechanic concepts (such as car movement) on A4 paper, we decided to make a model of the city from cardboard. We drew a square grid onto it both for being able to position buildings and unambiguously move cars around.

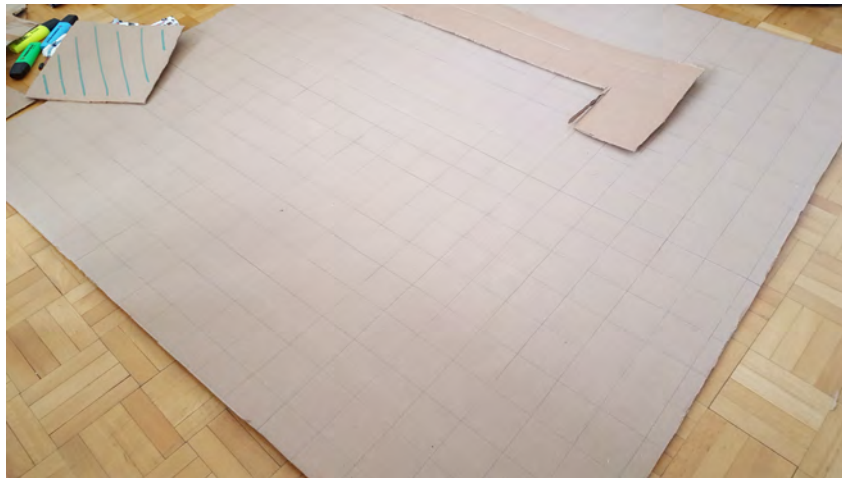


Figure 1: The ground of the city model.



Figure 2: Engaged in cardboard cutting.

Afterwards, we cut out 2D building outlines of various shapes from the cardboard and coloured their roofs. They can be easily placed onto the main plan, making the city map very flexible, as shown in Figure 3.

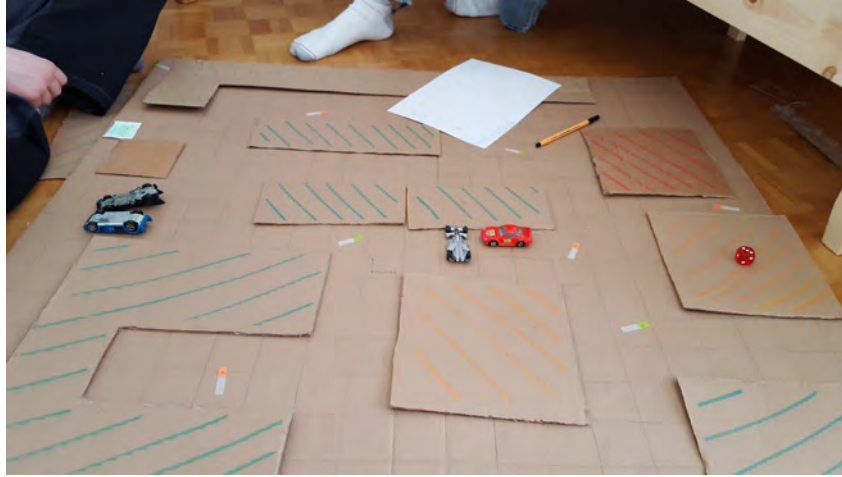


Figure 3: Map of the city with building positioned on the grid.

The next step was to create game items, such as the model of the fruit stand. It occupies exactly one square on the map.



Figure 4: Model of fruit stand to be positioned on the map.

Besides the fruit stand, other game items are

- toy cars
- dice
- power ups - size of fruit stand, can be positioned on the grid (details later)
  - velocity boost
  - concrete block
  - healing kit

Finally, we marked 12 locations on the streets using small stickers, 6 green and 6 orange (see Figure 5). Each round, a fruit stand is randomly spawned at one of the green location. The location is selected by rolling a dice. Six orange locations are used as possible spawn locations for power ups, of which there are exactly 3 on the board at any time.

Note that we also chose fixed spawn location for every team which is not shown in the image.

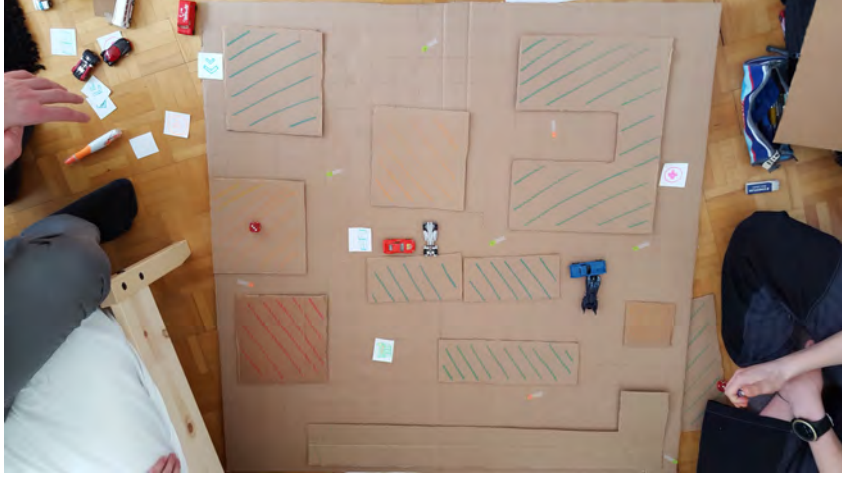


Figure 5: Complete model.

### 3 Game mechanics

#### 3.1 Car Driving

It's clearly not feasible to simultaneously simulate real-time driving of multiple cars on one board. Hence, we opted for discrete, turn-based approach.

At any time, a car has a 2D velocity  $\vec{v} = (x, y)$ . On player's turn, the player can change each component of the velocity by  $\pm 1$  or 0. E.g. change from  $(x, y)$  to  $(x, y + 1)$ ,  $(x - 1, x + 1)$  or  $(x - 1, y)$  is allowed, while a change to e.g.  $(x + 2, y)$  is not.

The aim of this mechanics is to sensibly simulate acceleration and steering, preventing players from being able to change driving direction in a sudden, unrealistic manner at high velocities. Concept of *drifting* is not represented in the paper prototype.

#### 3.2 Setup and Basic

We play-tested the game both 1-on-1 and 2-on-2. This section explains 1-on-1 scenario since 2-on-2 is just straightforward extension with fair turn taking.

For randomness, dice are used. We designed the game in a way that we only need to generate random numbers 1 – 6.

At the beginning, each player (one is attacker and one defender) positions his car at a per-team predetermined location at the edge of the board. The car has originally *3 health points*. One fruit stand is placed on the central green marked spot and 3 power ups are randomly placed on orange spots, face down.

For the rest of the game, players take turn as explained in the next section.

### 3.3 Player's Turn

Each turn of a player is logically divided into a sequence of steps.

1. **Change the velocity** of your car according to the rules in section 3.1. It is a duty of every player to keep track of the current velocity of his car (and the number of health points) by any means.
2. **Move the toy car** on the board. If the straight trajectory<sup>1</sup> on which you are moving intersects with any other object on the map, it's resolved as a crash (see later).
3. If you **hit (drive over) the fruit stand**, the round is over. Remove the stand from the board, add yourself points (see section 3.5) and randomly generate a new fruit stand at another location<sup>2</sup>.
4. **Pick up any power-up** you drive over and add it to the pile of power ups you have already collected. New power up is randomly generated on the map. See section 3.6 on power-up usage.
5. If you **crashed other player's car**, ask a third person (referee) to act as a 'physics simulator' and simulate collision as accurately as possible using human judgement, including information about velocities of both cars. In the absence of the referee, players can agree on a plausible outcome of the collision. Both cars lose one health point (see section 3.4). *Side note: We have originally tried to invent precise rules for simulating car-car collisions but we found it very hard and decided not to waste time on this.*
6. **Crashing into building** makes the car stop immediately (i.e. reducing it's velocity to (0,0)) and the car loses one health point (see section 3.4).

### 3.4 Health points

As written earlier, every car has 3 health points. For simplicity, it loses always one upon any collision with another car or with a building.

If the car loses all health points, it's destroyed and the player **skips one turn**. Then, the car is **respawned** at its original starting location with 3 health points and the player takes turn normally.

### 3.5 Scoring

When the attacker's car drives over the fruit stand, his team gets number of points corresponding to the largest velocity component (in absolute terms) of that car<sup>3</sup>.

If the defender crashes the fruit stand, the attacker's team gets points close to the maximum achievable velocity on this map (map specific). This is to prevent the defender to intentionally crash the fruit stand and / or reward the attacker if he manages to push the defender into the stand.

---

<sup>1</sup>connecting the centers of the square cells

<sup>2</sup>Fruit stand cannot be generated in the vicinity (circle with radius 1, diagonals included) of any player.

<sup>3</sup>For vel.  $\vec{v}$ , the number of points is  $|\vec{v}|_\infty$ . One can imagine the 'fun' we had discussing different L norms :).

### 3.6 Power-ups

Picked power-ups are secretly held by a player and can be used at any time. There is no limit to the number of power-ups a player can hold. Power-ups we experimented with are:

- **Velocity boost**

The boost is played before the ‘velocity adjustment’ phase of the turn (phase 1). Player can change both velocity components by any number from -3 to +3 instead of from -1 to +1.

- **Concrete block**

A rigid block of size 1 square unit is placed right behind the car of the player. The block acts as a non-movable obstacle, i.e. it behaves as an ordinary building and other players can crash into it.

- **Healing kit**

Add one health point to your car.

### 3.7 End of Round

The round ends either when the fruit stand is crashed or when every player has made 10 turns. For the new round, attacker and defender roles switch and new fruit stand is randomly positioned at one of the possible spawn locations.

## 4 Lessons Learned

### 4.1 Scoring

While playing around with the prototype, we’ve come to the conclusion that not every destruction of a fruit stand should give the same number of points. The main idea is that scored points should depend on the velocity of the car that crashes the fruit stand. Besides this, the formula could be spiced up by evaluating the time left for the current round as well. Additionally, if the defender crashes the fruit stand, the attacking team will get an appropriate amount of points.

### 4.2 Level Design & Spawning

Level design and spawn location placement should be such that it favors neither attackers nor defenders and thus provides an equal fighting ground. This part of the game will need to be additionally tested and carefully crafted. The initial findings from the prototype suggest that generation of a new fruit stand should not happen anywhere close to the players in a certain (small) radius. In addition, there need to be more spawn locations for fruits than there are players, to prevent abuse.

At the beginning of the game, the fruit stand should spawn at the centermost location relative to the map, while the player cars should spawn at the edges / corners of the map. Car spawn locations also need to be predetermined, and respawning should happen at the initial spawning location, defined per car, per level. Also, the respawn timers for the round will need to be determined in such a way that they do not make suicides / self-crashing a viable strategy choice in most scenarios, while also not keeping the player away from the game for too long.

### 4.3 Power-Ups

One of the things that will need more precise definition are also the power-ups. Proposed power-ups proved to be useful in the physical prototype scenarios and we are certain that their utility will carry over into the digital version of the game as well. We noticed that defending a fruit stand could be more difficult than attacking it. To make sure that the game is fair, we need to have power-ups that help the defenders. As such, we've designed the "barricade" power-up, which enables the defender to instantly place a block/wall of concrete behind his car. These blocks last temporarily until the round is finished. In order to prevent abuse, we'll introduce a rule which either further limits the barricade lifetime or limits the number of different barricades placed at once.