



Fruit Smashers: Project Proposal Draft

Matej Hamas, Daniel Keyes, Nemanja Bartolovic, Delio Vicini

March 14, 2016

Contents

1	Game Description	2
1.1	Summary	2
1.2	Detailed Description	2
2	Technical Achievement - Vehicle AI	3
2.1	Overview	3
2.2	Implementation Plan	4
3	Big Idea Bullseye	5
4	Development Schedule	5
4.1	Layered functionalities	5
4.2	Task Timeline Overview	7
5	Assessment	7
6	References	8

1 Game Description

1.1 Summary

Our game idea is inspired by Hollywood car chases. In many of those car chases, one or more fruit stands are getting destroyed as the car chase goes on.

The story around our game concept is the following: In the beautiful and usually peaceful city of XYZ, there exist two competing groups of fruit traders, which over the years developed a very strong rivalry between each other. One sunny day, a car chase took place in this city, during which fruit stands of both factions were destroyed! Of course, each party blamed the other one for their loss. Since they could not settle their dispute, they started to drive into fruit stands of their opponents on purpose and started an epic feud between the two clans!

In our game, the goal is to use your car to destroy as many fruit stands as possible within a certain time limit. The game is primarily a multiplayer experience, where you compete with your friends. The gameplay is very dynamic, as you have to race to the next fruit stand to destroy through the streets of the current map. A typical round of the game will feature both racing and "car combat" aspects, e.g. ramming of other players.

1.2 Detailed Description

A typical round of the game will look as follows: the players in the current game are divided into two competing teams. The multiplayer gameplay is realized using split screen. Each player controls his car from 3rd person, as shown in Figure 1 on the following page. The graphics are 3D and the gameplay potentially also takes place in 3D and is not necessarily limited to the 2D plane. This means there could be height differences or ramps in the map. As the game starts, a fruit stand is randomly spawned at an accessible spot on the map. The discrete set of spawn location will be created by hand. One team which is randomly chosen has to defend the fruit stand for a certain time, while the other team tries to crash into it and destroy it. The location of the fruit stand is visually indicated on the heads-up-display of each player. Each player also sees a minimap depicting the street layout and the player position (and possibly the position of other players).

The defending team can ram the attacking teams cars or try to block access to the fruit stand. Both teams may also employ gadgets or boosts which can be picked up on the map, similar to Mario Kart. These powerups can be collected on predefined locations on the map. A player may hold up to 3 powerups and employ them at any time he desires. Powerups include but are not limited to speed boost, health boost, invincibility and score boost (e.g. double the score a team gets while it is activated). Also defensive powerups such as barricades which can be dropped behind the players car or bananas to make the surface slippery may be employed. Further ideas for gadgets may arise during development and playtesting, as those gadgets potentially play some role in balancing the game to be fair and fun for all players. Note that cars can possibly get destroyed if they take too much damage from ramming or crashing. In this case, the player is respawned at a random location on the map. If a car is flipped upside down or stuck, it can be reset to a near feasible driving position (possibly specified by hand in advance).

Once the current fruit stand is destroyed or the time limit is up, the team which successfully achieved its assigned objective earns a point and the next fruit stand is spawned in another random location. Furthermore, the attacker and defender roles switch, which means the teams have to quickly adjust to their new objective. After a certain point limit is reached by one of the teams, this team wins and the round is over. At this point, either a new map is loaded or the round restarts on the same map.

A map depicts a small part of the city. Due to the short development period, the number of maps will be fairly limited (we plan to have at least two different maps) and all maps will probably have the same visual style.

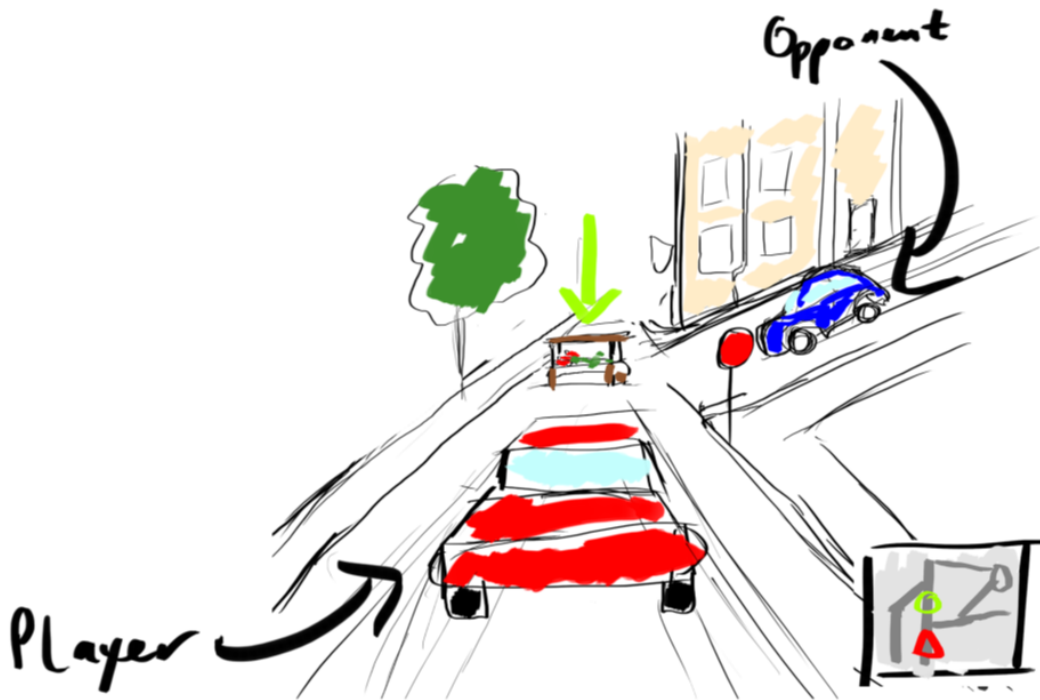


Figure 1: Sketch of the players view in from behind his car (in red).

2 Technical Achievement - Vehicle AI

2.1 Overview

Originally, we intended to include an element of passive traffic into the gameplay. After the feedback and advice from mentors, we have decided to discard this idea. Instead, we will implement basic **vehicle AI**¹.

We think that the vehicle AI will benefit game in following aspects:

- It will allow for more entertaining **single player** mode (other than just driving through empty streets of the town alone trying to crash fruit stands).
- By including multiple AI vehicles, split screen multiplayer mode (which is the most likely multiplayer scenario) will no longer be limited to 4 players which seems to be the technical limit for the single screen multiplayer.
- Trying to trick the AI logic will introduce a new aspect to the game which can for some players make the game more fun to play.

We are aware of the fact that the implementation of AI vehicles is quite challenging. That is one of the reasons why we decided to list this element as a *technical achievement*. None of us has previous experience in this area and hence we think it is important **not** to think huge.

¹artificial intelligence

However, AI elements are so common in racing games that we strongly believe that we can implement at least a simple model of AI vehicles.

2.2 Implementation Plan

Note that this plan has been written with minimal knowledge of how vehicle AI is usually solved in off-the-shelf games. Hence, it is subject to future changes.

1. Creating a graph representation of the map which can be utilized by the AI. Nodes represent crossroads and edges represent roads connecting them.
2. Implementing a car control mechanism for short term local decisions such as *"now go straight for 100 metres"* or *"turn right on this crossroad"*. This includes an algorithm for keeping the car on the road, e.g. by introducing invisible boundaries that cannot be crossed by the vehicle.
3. Writing a planning algorithm for navigating car from point A to point B on the map. Up to and including this point, the algorithm still does not handle any collisions. However, it should be already usable in the game, in its simplest form. It can both race towards an opponent fruit stand to crash it or towards its fruit stand to defend it by blocking the road. The behaviour in case of collision is not handled.
4. The next important step forward is the collision handling. This includes implementing a heuristics for returning on the road in case of crashes and being able to continue on a given path. We estimate that this part will be quite challenging and will offer lots of potential solutions ranging from naive to advanced techniques.

We think that completing aforementioned steps would suffice to create a playable AI. However, there is a lot of space for improvement we are aware of. This includes things such as

- intelligent chasing of the opponent with intention to crash him.
- cooperation between multiple AI's in order
 - to avoid crashing each other or a human team-mate.
 - to avoid concentrating at a particular spot (i.e. not blocking just a single street).

We do not expect to have enough time to implement all of the advanced AI features. We will treat them as potential high target or extra options.

3 Big Idea Bullseye



4 Development Schedule

This section presents the overall development process, described by a layered structure where each layer represents a collection of functional aspects that make up the game, as well as a loose time schedule that describes separation of the development process into smaller tasks that can be completed individually, with low coupling between the tasks themselves.

4.1 Layered functionalities

1. Functional Minimum

- A single 3D level/map of a city (with placeholders)
- Fruit stands randomly spawning throughout the reachable area, 1 existing at a time
- A single user-controlled car hitting the stands as they appear
- Basic game goal of hitting the stands under a given time limit
- Basic camera control
- Simple game visuals

2. Low Target

- Basic game physics and collision detection
- Concept of car "health"
- Basic power-ups (health boost, speed boost, invincibility and score boost)
- Multiple players through split screen(2 or 4)
- A different game goal where one team defends the stands and one team crashes them, in multiple time-limited rounds

- Basic GUI
- Improved game visuals

3. Desirable Target

- Mini-map
- More advanced power-ups (bananas, barricades)
- Sound effects
- Improved fruit stand visuals / physics
- Simple car AI (pathfinding)

4. High Target

- Music/Soundtrack
- Advanced car physics
- Advanced car AI (attacking and defending)
- More maps/levels
- Additional visual improvements (normal and shadow maps)

5. Extras

- Online multiplayer with an additional 8-player game mode
- Even more advanced power-ups
- Car weapons
- Procedural level design
- Different types of cars with differing qualities
- Progressive visual deterioration on cars

4.2 Task Timeline Overview

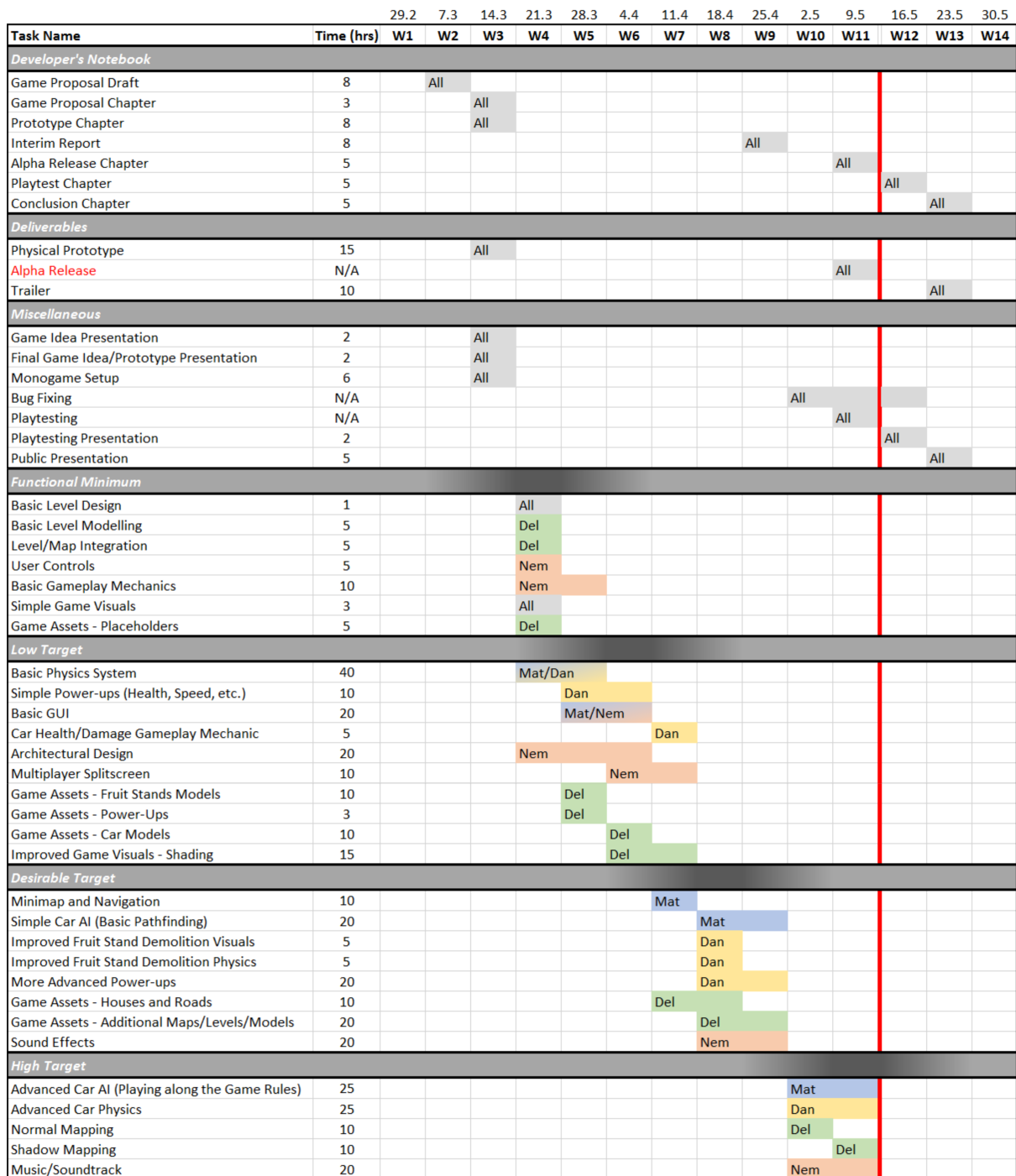


Figure 2: Proposed task schedule with allotted time frames for each

5 Assessment

The main appeal of this game is the wonton (yet structured) mayhem that comes with racing to destroy the fruit stands, which we highlight in the "Big Idea Bullseye".

In other racing games (like GTA), many players have much more fun destroying the world

and creating their own stunt scenarios than just racing, so we want this game to gather that excitement into a single package. Some games, like Rocket League, Twisted Metal, or even Mario Kart with battle mode, have directly used car-to-car combat to create fun challenges, and we hope the Hollywood-style realism of our game gives this combat a fresh spin. Furthermore, this sort of mischief and destruction is not just unique to racing games—shoot 'em up and hack'n'slash games reward the player causing big explosions, so the goal of the game is to appeal to those genres, too.

In addition, the simple objectives and alternating attacker-defender roles make the game frenetic and fast-paced. As a criteria for success, we want the frantic timing and rampant destruction to give the game lots of replay value, so that players crave to play again, round after round. Our "technical achievement" (vehicle AI) should also add to this experience, so that the game is still exciting when played alone or enhanced when playing with others.

Also, as a more general criteria for success, we want the control and look-and-feel of the game to be intuitive and natural. To this end, we would like the car physics and object collisions to reflect the real world, so that players feel like they are playing a realistic racing game. The time we spend on map design should also contribute to this look-and-feel, immersing the player in the city destruction experience.

6 References

Sources of the images used in the proposal:

- **logo on the first page:**
<http://weareforeal.com/content/01-projects/tropical-blast-2014/main-01.jpg>
- **bulls eye idea:**
http://img15.deviantart.net/ca71/i/2014/323/f/7/nissan_s13_4-colored_by_splicer-d86yk08.jpg