



Fruit Smashers: Project Proposal Draft

Matej Hamas, Daniel Keyes, Nemanja Bartolovic, Delio Vicini

March 7, 2016

Contents

1	Game Description	2
1.1	Summary	2
1.2	Detailed Description	2
2	Technical Achievements	3
3	Big Idea Bullseye	5
4	Development Schedule	5
4.1	Layered functionalities	5
4.2	Task Timeline Overview	7
5	Assessment	7

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, evading civilian traffic and possible other obstacles. 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.

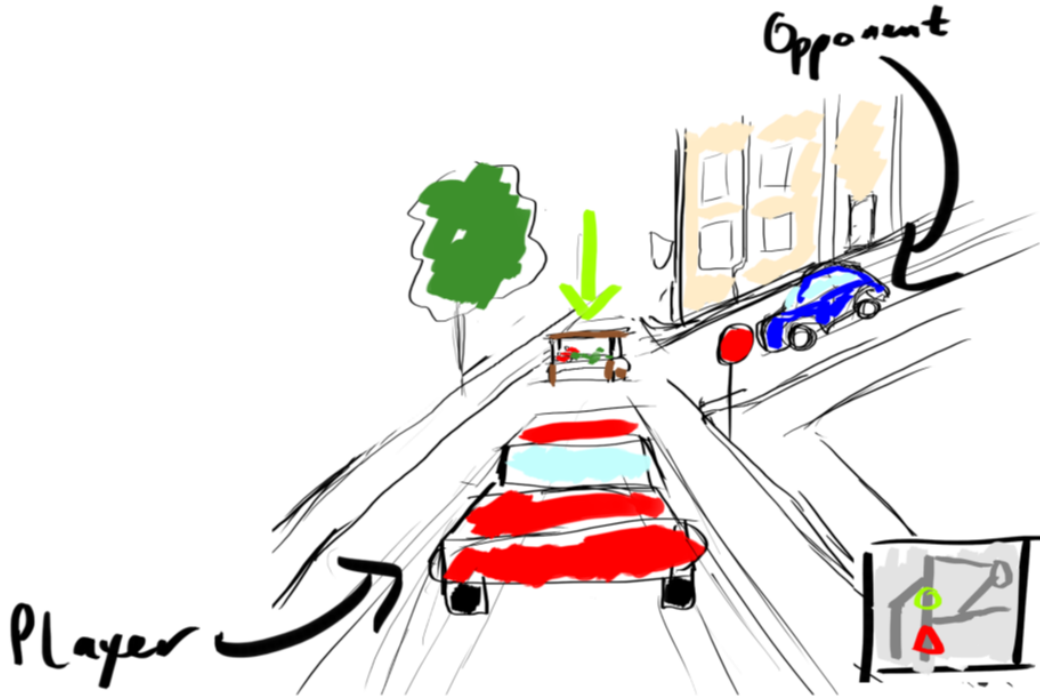


Figure 1: Sketch of the players view in from behind his car (in red).

2 Technical Achievements

Initial Idea (Discarded)

Originally, we were thinking about concentrating on the game (mainly car) physics and taking it into perfection. However, implementing accurate 3D physics would be a project on its own due to its immense intricacies. After reconsidering this, we have decided to go purely for a 3D physics library¹.

Proposed Idea

To make the game as entertaining as possible, we have decided to introduce **traffic** into the gameplay. This is a common element in car chasing games and makes the game more realistic as we are not used to empty city streets in the real life.

¹such as BEPU or Bullet

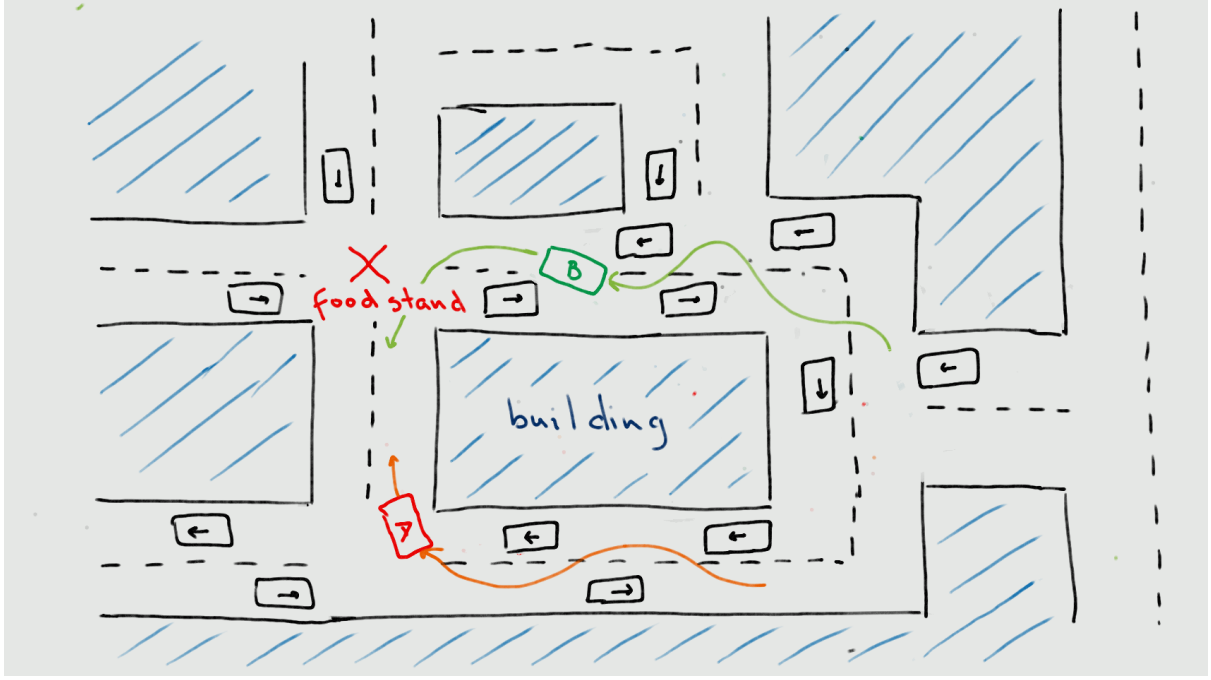


Figure 2: Top down view of the map showing 2 plays: red A trying to crash the food stand and green B trying to protect it. The passive traffic vehicles are shown in black, with arrow indicating their movement.

Traffic consist of cars that are moving through the streets of the city. As a prerequisite, we will have to model the city as a graph where crossroads are nodes and streets connecting them are edges. This can be done either manually during the map creation process or automatically, should we decide to go for a semi-automatic map creation in case of time abundance.

Having the graph of the city, we will need to generate starting and finishing location for every traffic unit (vehicle) and route it through the city. We expect to be facing following challenges:

- Making sure the traffic goes in the right lane.
- Ensuring that the traffic units don't collide.
- Handling the traffic behaviour on crossroads.
- Recovery after being crashed by a racing car and pushed off track
 - either the car stops moving and plays a role of passive obstacle for the rest of the round,
 - or the car is navigated back to its track to finish the route through the city.

The advantage of this idea is that it has clear structure in terms of the subproblems. Moreover, the used model of the traffic can vary dramatically in terms of the difficulty. Hence, it allows for gradual transition from the simplest to more difficult models, for example as shown below.

1. One way traffic on a single straight road.
2. Two way traffic on a single straight road.

3. The response to a crash between racing car and a traffic vehicle is passive and the traffic unit plays passive obstacles for the rest of the round.
4. Traffic on multiple roads and handling interaction on crossroads, such that vehicles don't crash.
5. Traffic vehicle tries to get back on its track after being crashed by a racing car.

Another advantage of the traffic gameplay elements is its easy extensibility. There is a lot of room for numerous advanced features such as traffic vehicles blocking or chasing racing cars, blocking streets to make the objective of crashing fruit stands more difficult etc. This could be achieved either by introducing appropriate heuristics or AI element. We realistically don't expect to have enough resources to implement these but it still stays a feasible option to consider in possible future development.

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 non-player-controlled traffic

4. High Target

- Music/Soundtrack
- Advanced car physics
- Advanced non-player-controlled traffic
- 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

4.2 Task Timeline Overview

Task Name	Time (hrs)	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
Developer's Notebook															
Game Proposal Draft	8		All												
Game Proposal Chapter	3			All											
Prototype Chapter	8			All											
Interim Report	8									All					
Alpha Release Chapter	5										All				
Playtest Chapter	5											All			
Conclusion Chapter	5												All		
Deliverables															
Physical Prototype	15			All											
Alpha Release	N/A											All			
Trailer	10													All	
Miscellaneous															
Game Idea Presentation	2			All											
Final Game Idea/Prototype Presentation	2			All											
Monogame Setup	6			All											
Bug Fixing	N/A										All				
Playtesting	N/A											All			
Playtesting Presentation	2												All		
Public Presentation	5													All	
Functional Minimum															
Basic Level Design	1				All										
Basic Level Modelling	5				Del										
Level/Map Integration	5				Del										
User Controls	5				Nem										
Basic Gameplay Mechanics	10				Nem										
Simple Game Visuals	3				All										
Game Assets - Placeholders	5				Del										
Low Target															
Basic Physics System	40				Mat/Dan										
Simple Power-ups (Health, Speed, etc.)	10				Dan										
Basic GUI	20				Mat/Nem										
Car Health/Damage Gameplay Mechanic	5							Dan							
Architectural Design	20				Nem										
Multiplayer Splitscreen	10					Nem									
Game Assets - Fruit Stands Models	10				Del										
Game Assets - Power-Ups	3				Del										
Game Assets - Car Models	10					Del									
Improved Game Visuals - Shading	15					Del									
Desirable Target															
Minimap and Navigation	20							Mat							
Simple Non-player-controlled Traffic	10									Mat					
Improved Fruit Stand Demolition Visuals	5							Dan							
Improved Fruit Stand Demolition Physics	5							Dan							
More Advanced Power-ups	20							Dan							
Game Assets - Houses and Roads	10						Del								
Game Assets - Additional Maps/Levels/Models	20							Del							
Sound Effects	20							Nem							
High Target															
Advanced Traffic	25										Mat				
Advanced Car Physics	25										Dan				
Normal Mapping	10										Del				
Shadow Mapping	10											Del			
Music/Soundtrack	20										Nem				

Figure 3: 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 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.

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. Our "technical achievement" (traffic simulation) as well as the time we spend on map design should also contribute to this look-and-feel, immersing the player in the city destruction experience.