**Proposal for**

**Empire Tycoonville**

**Section 1: Summary**

The idea is to create a Tycoon style game (think Lemonade Tycoon, Rollercoaster Tycoon) but with a twist: a continually fluctuating market based on other actions. A smart player can time the market and create an advantage for themselves beyond just the normal built-in game play.

**Beginning**

A player will start out being able to create a lemonade stand, which will generate a specific $ per minute in revenue. As the player accumulates more revenue, he can purchase upgrades which increase the $ per minute rate of the lemonade stand.

**Next Level**

The player purchase another lemonade stand that can keep adding to the revenue per minute dollar amount. As the player accumulates more and more money, they can unlock bigger businesses that give them better revenue generators

**The Market**

Each business will accumulate value over time, based on the upgrades it has, and the player can sell the business for a profit. The actual amount of what the player can sell for will be market based which fluctuate over time, based on other user’s behavior. So each time you sell a lemonade stand, the next ones become worth lower because the market drops.

If no one sells any lemonade stands, the market demand goes higher, and the players can cash out at higher amounts. It would be a balance of market timing by the player to get good value for the businesses.

The market would be displayed by a graph on the dashboard, showing where the market currently is with a historical chart, so the player can see if it’s at the top or bottom of the market.

**Section 2: Wireframes included by hand**

**Section 3: Feature List:**

* P0: User can purchase businesses, upgrade them and sell them
* P0: Market algorithm, single player. Value of businesses will be market based, scaling up or down.
* P0: View each business
* P1: Dynamic web design.
* P2. Mobile friendly, responsive
* P2. Multiplayer
* P2: Multiplayer Market algorithm, multiplayer. Value of businesses will scale up or down depending on how often the other player is selling.
* P3: Winning condition.
* P2: Images and animations
* P3: Refine higher tiers for better QOL for the player, and unlock bigger upgrades.
* P3: Player can generate more valuable resources at higher tiers (gold, gems, etc)
* P3: View other player’s dashboards

**Section 4: Technical Considerations:**

Will be built on Python with Django and HTML, CSS, JQuery

**Section 5: Work Distribution:**

Front end page layouts: Carmine and Landon build out the pages

Database and file structure: Doina

JQuery and AJAX: Carmine

Market graph and algorithm: Landon

Images and graphics: Doina

Multiplayer: No idea yet