Piecing it Together:

Development of a Tower Defense Game

Rules

- **Tower Defense:**

You build static towers along a path to prevent a constant flow of monsters from reach the end of the path.

Research

- Tower Defense
- Research of other similar games:
  - Fieldrunners
  - Bloons Tower Defense
  - Kingdom Rush
  - Plants vs Zombies
  - Castle Creeps

Balloon TD


Plant vz Zombie

Kingdom Rush


OTTTD


Lich Defense


Beware Earth Planet


Three Kingdoms TD: Fate of Wei


Elf Defense


Castle Doombad

Stop The Knights


Cursed Treasure 2


Kingdom Rush


Lord Of Trap


Orc Must Die

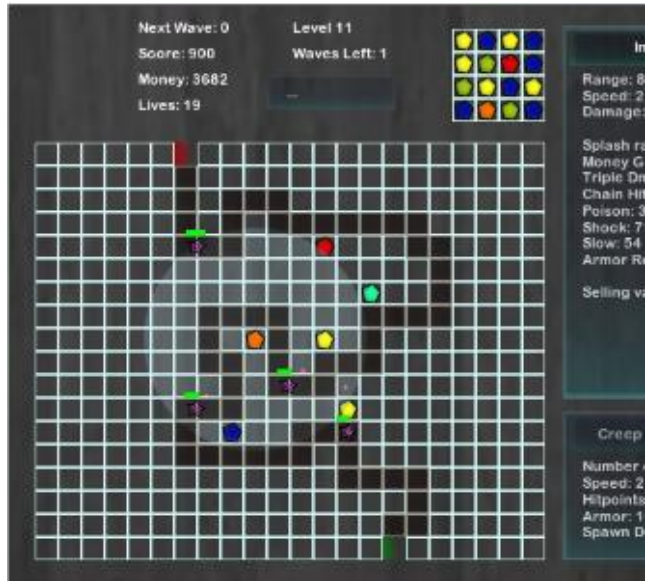
Madness TD

Empire Defense

Observations:

- **Single-path entry for monsters (Geo Defense) easier to maintain than multiple entry paths (Bloons).**
- **Limited build spots (Kingdom Rush) versus towers-as-walls (Desktop Defense) allows more even testing of the level and fewer variables.**
- **There was a matching of towers' power/weakness to a monster/weakness – and vice-versa (All).**
- **Each tower/monster had a strength and weakness (All).**
- **Smaller waves of monsters (PvZ) easier to test versus hordes (Kingdom Rush).**

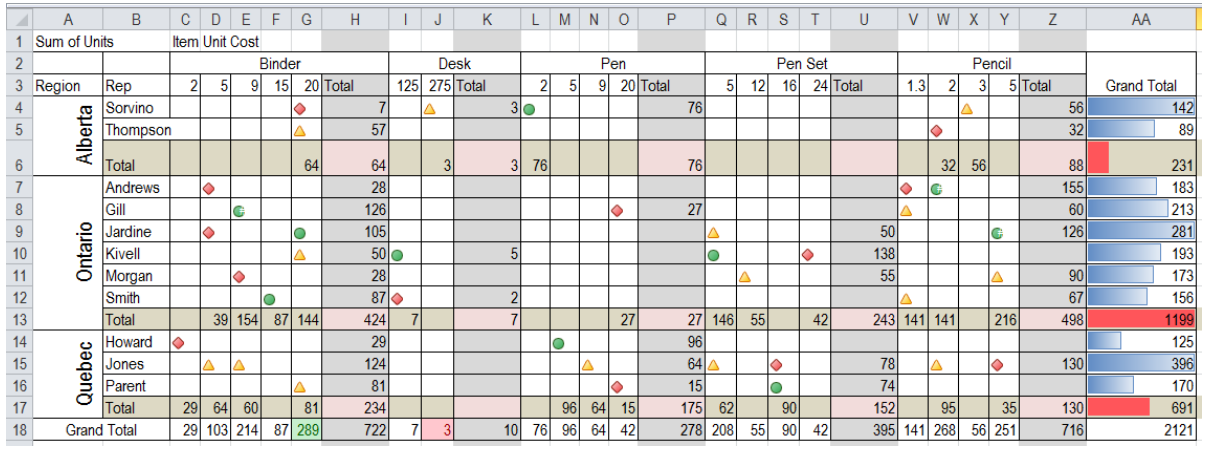
- **But the biggest surprise was…..**



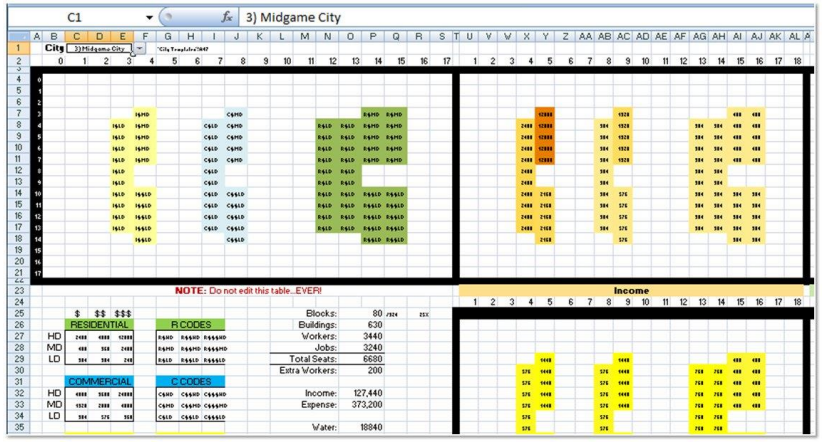
Tower Defense Games Are Spreadsheet-Driven

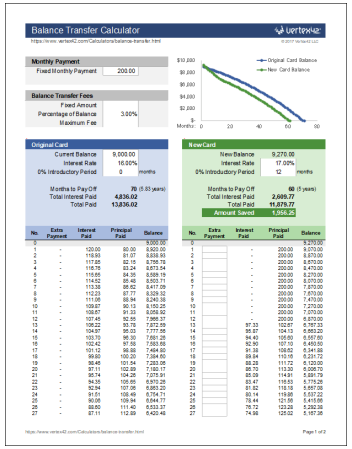
Creep	Level	HP	Special Features	Speed	Armor
Footman	1	48	none	fast	0
Knight	2	96	none	very fast	1
Rifleman	3	192	none	fast	2
Gryphon Rider	4	150	flying	fast	0
Paladin	4	300	divine shield	medium	4

- **A huge load of variables that had to work together so that the player felt like there was a well-planned arsenal against well-planned monsters.**
- **Interview with Alphonzo Alvara: “Level designers play their levels 50 times”.**
- **We had to pinpoint the biggest issues with making this game and prepare for it ahead of time.**



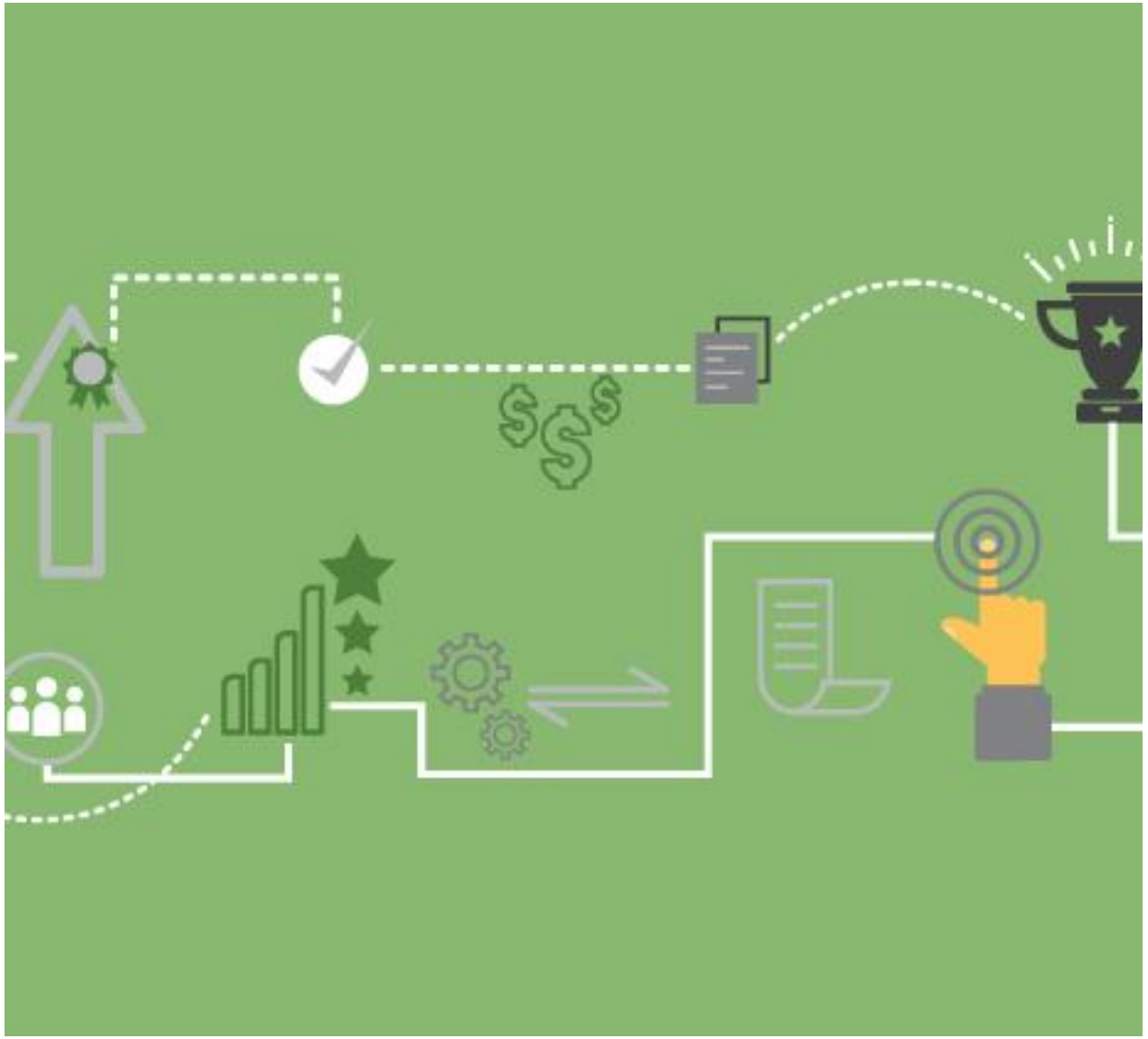
Sum of Units	Item Unit Cost																								
		Binder						Desk			Pen					Pen Set					Pencil					
Region	Rep	2	5	9	15	20	Total	125	275	Total	2	5	9	20	Total	5	12	16	24	Total	1.3	2	3	5	Total	Grand Total
Alberta	Sorvino						7			3					76										56	142
	Thompson						57																		32	89
	Total					64	64		3	3	76				76							32	56		88	231
Ontario	Andrews						28																		155	183
	Gill						126								27										60	213
	Jardine						105													50					126	281
	Kivell						50			5										138						193
	Morgan						28													55					90	173
	Smith						87			2															67	156
	Total		39	154	87	144	424	7		7				27	27	146	55		42	243	141	141		216	498	1199
Quebec	Howard						29								96											125
	Jones						124								64					78					130	396
	Parent						81								15					74						170
	Total	29	64	60		81	234					96	64	15	175	62		90		152		95		35	130	691
Grand Total		29	103	214	87	289	722	7	3	10	76	96	64	42	278	208	55	90	42	395	141	268	56	251	716	2121





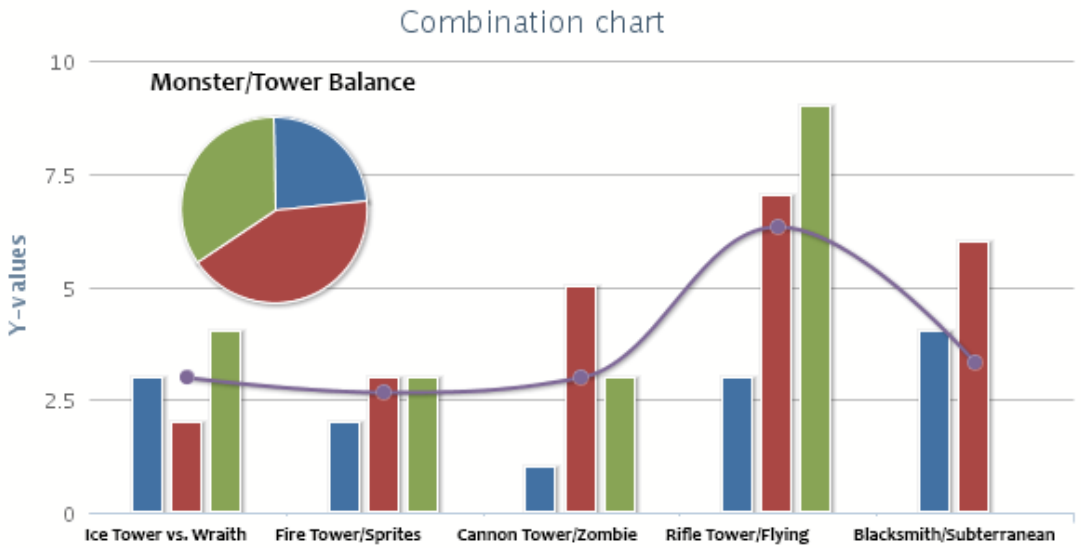
Concerns:

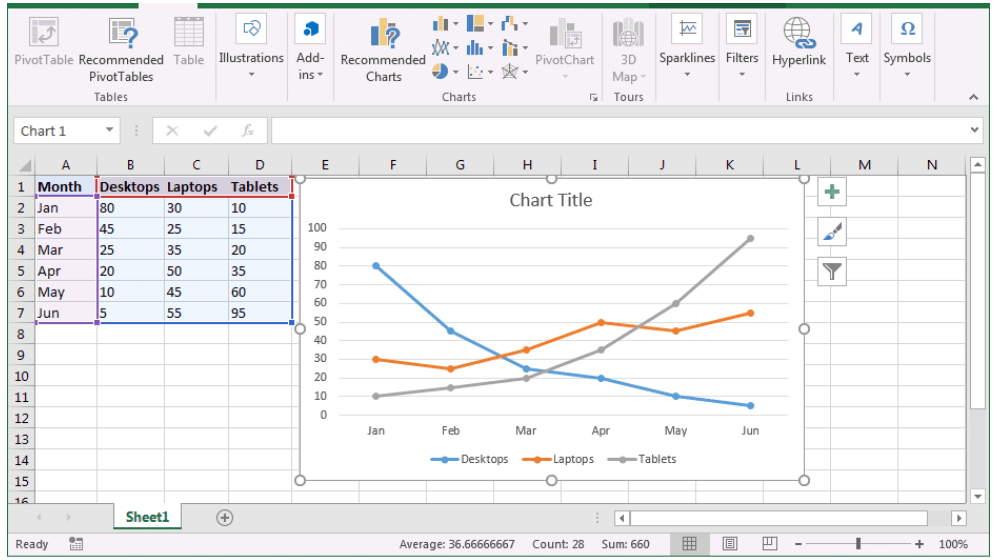
- **Balance:**
  - ***We needed to work out the data balance first***
- **Design:**
  - ***We needed a fast turnaround for designers to play their levels, make changes, and play levels again.***
  - ***Hardcoded values were not feasible.***
- **Economy:**
  - ***We needed an Economy (reward system)***



Balance Solution:

- **Read/Watched a few tutorials specific to Tower Defense Game Balance**
  - https://www.youtube.com/watch?v=FFR4EXnAEBA
  - https://www.gamasutra.com/view/feature/6400/understanding_balance_in_video_.php?print=1
- **Chose Excel for balancing:**
  - **Worked well, we were able to see balancing monsters and towers power nicely through the use of graphs**
  - **Ubiquitous software**
- **Gleaned a few default towers with default values from internet. (4 basics).**

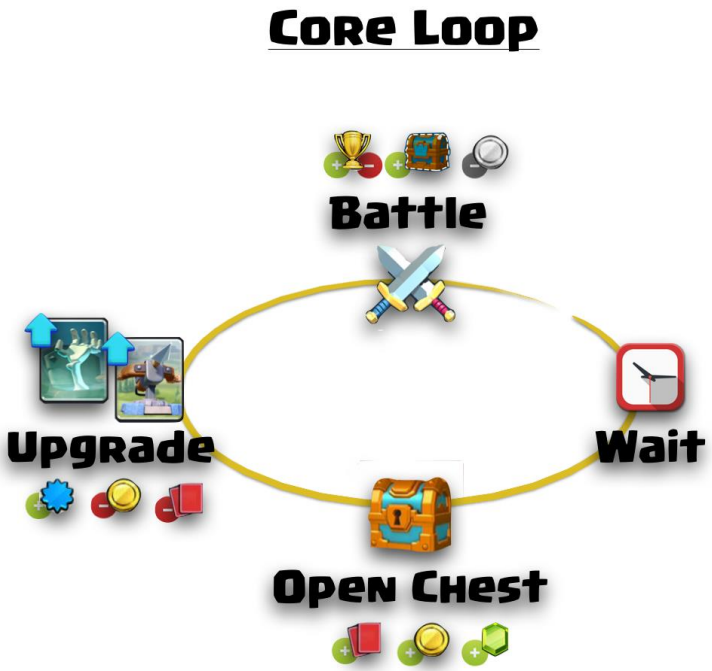


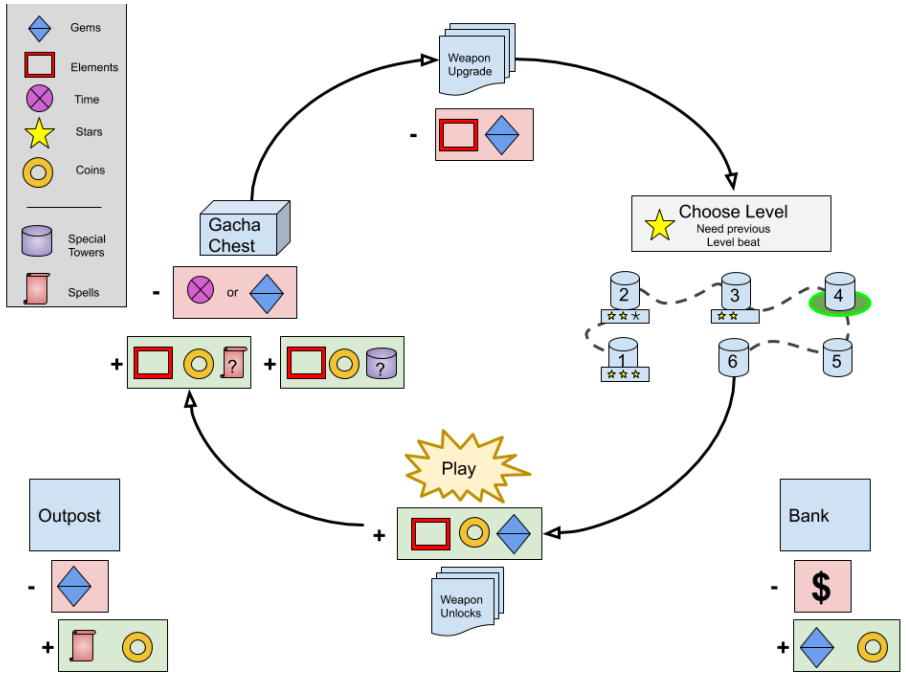


TechOnTheNet.com

Design Solution:

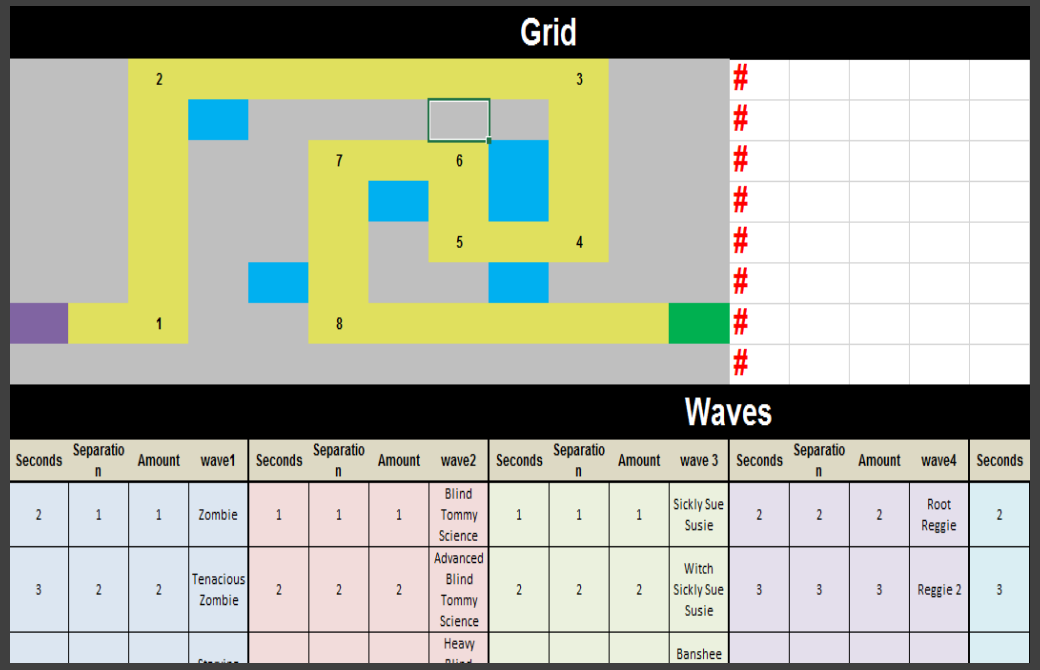
- **All monster and tower variables were in one excel sheet**
- **All level designs were in another excel sheet**
- **When values for either monster, tower, or level were tweaked, the excel sheets automatically exported the values as an XML file and Unity read it in next time it started.**
- **We made a basic Unity .exe file that ran a super-simple version of the game.**
- **http://www.scotteasley.com/mediaportfolio/Projects/EA/EA_page.html**



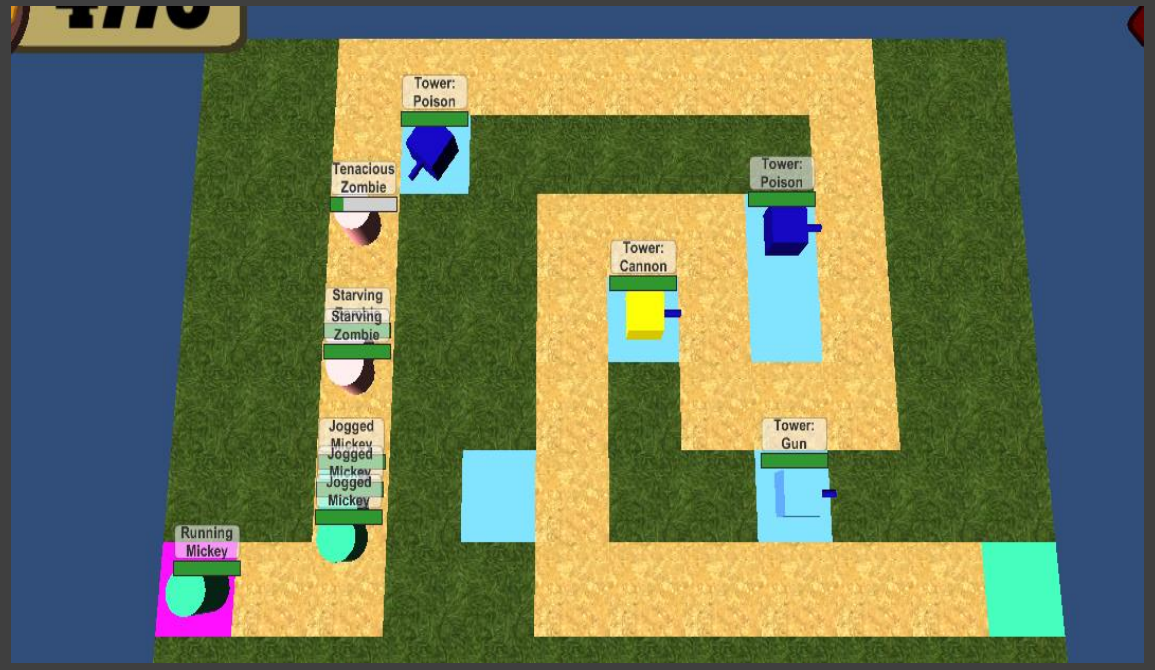


Economy Solution

- While one team worked on the game loop, our PM worked on the economy.
- Google Drawing with simple flowchart.
- We looked at economies for other games in this genre as a starting point.

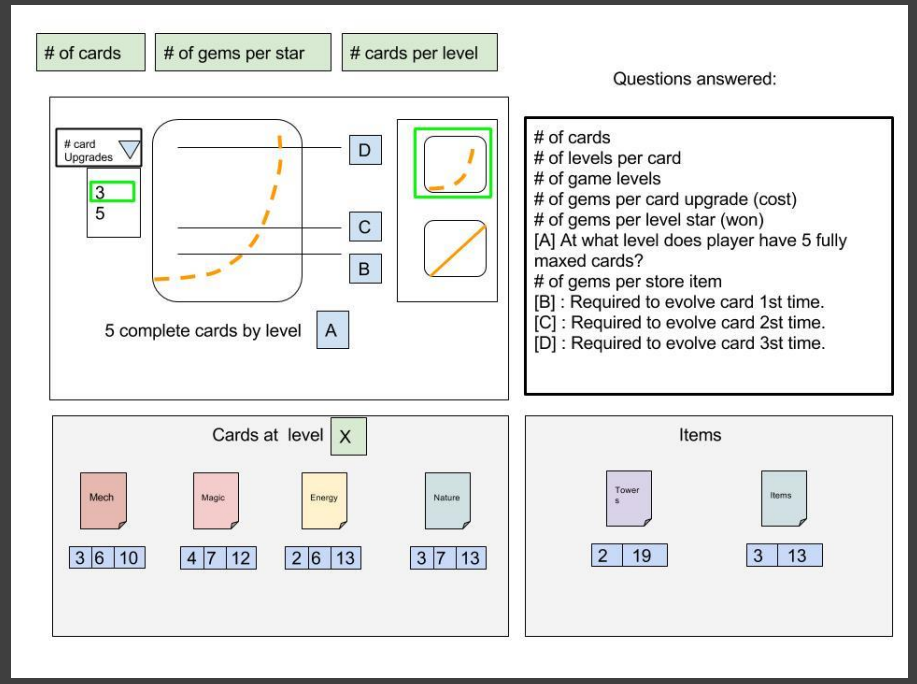


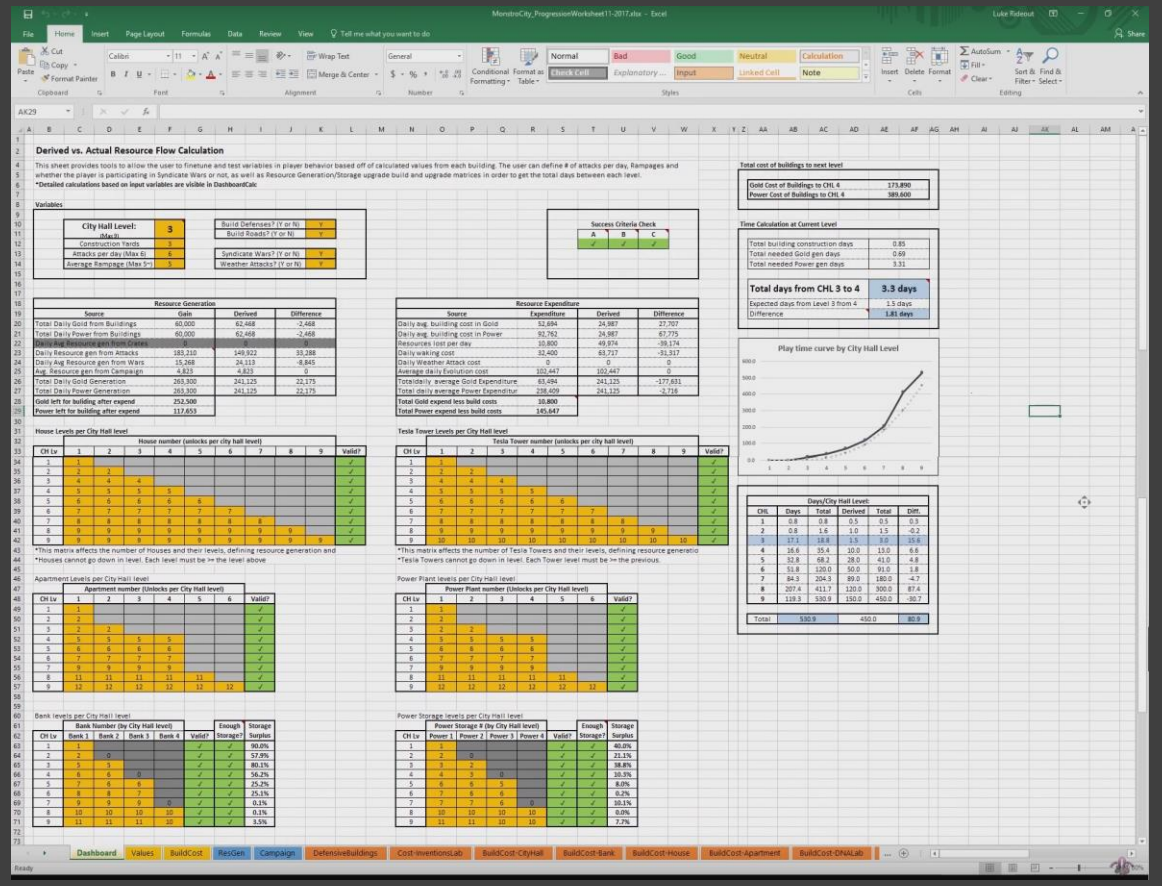
Seconds	Separation	Amount	wave1	Seconds	Separation	Amount	wave2	Seconds	Separation	Amount	wave 3	Seconds	Separation	Amount	wave4	Seconds
2	1	1	Zombie	1	1	1	Blind Tommy Science	1	1	1	Sickly Sue Susie	2	2	2	Root Reggie	2
3	2	2	Tenacious Zombie	2	2	2	Advanced Blind Tommy Science	2	2	2	Witch Sickly Sue Susie	3	3	3	Reggie 2	3
							Heavy Blind				Banshee					



Parallel Work:

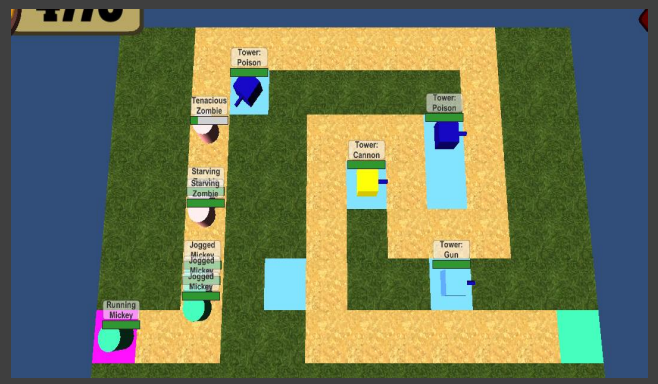
- **Design and Balance could now iterate and tweak on excel sheets without having to interrupt engineering.**





Separate Evolution:

- **Game Economy could get more complex on the excel side without interruption – the game still only used the same XML data exports.**



Art Last

- **The mockups and greyboxes built our game for us, informing us what we needed for the art. It was the last step, grown and informed by a solid internal core of game loop and rewards.**