Mobile Game Analysis

By Liad Traube

TL; DR

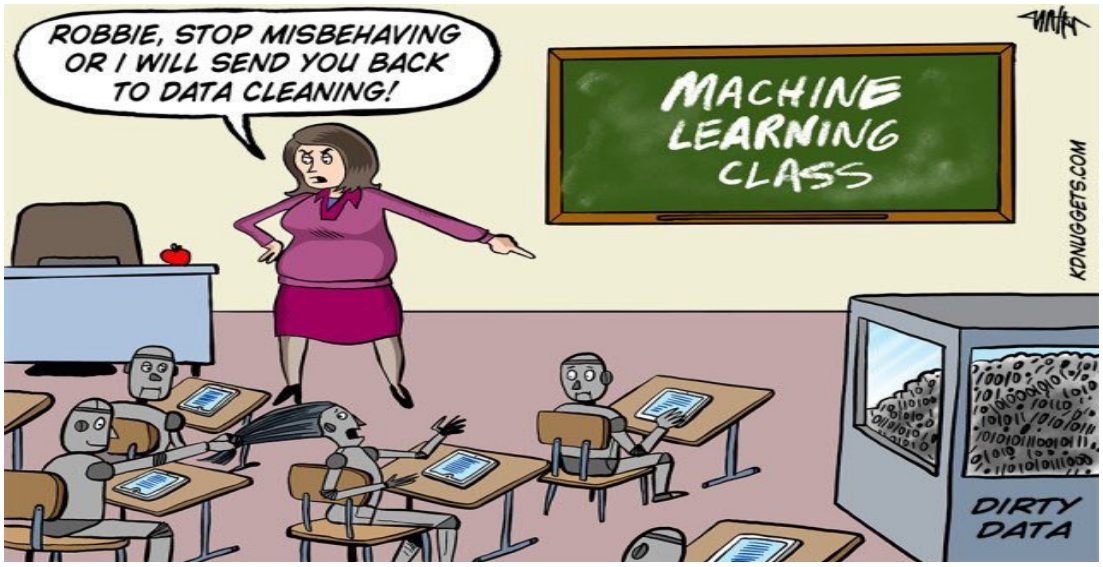
This project goal is to simulate a **mobile game database** and an analysis that is based on data from that database, including **KPI's and game economy.**

1. Introduction

For many days I wandered around web-sites like "Kaggle" and "Maven Analytics" searching for interesting datasets that I could use to create a big and impressive project, downloading files, loading the datasets in MSSQL and Tableau, preparing queries in SQL but It was all for vain.

Liad Traube | LinkedIn

none of the datasets were good enough for my needs, from my point of view and it seemed to me a lot the datasets in Kaggle were tailored for data-science and machine learning projects and some of the datasets were simply not "rich" enough in data or did not have what I needed for the kind of project that I had in mind.



As a result of my frustration in finding a proper dataset I decided that I have to address this challenge and build a database myself.

As a consequence of having the creative freedom with the mission of building a database by myself I decided to choose a subject that I find interesting and I want to study more from the "developers" side.

Mobile Game Analysis

As you will see in this project, **I tried to simulate a real mobile game database** as best as I could, using randomly generated data thanks to website like "generate data" and various functions in Microsoft Excel.

The database contains data about a mobile game called "Empire Of Crabs" (I tasked chat-GPT to generate me 10 names for mobile games and this was my favorite name).

*This image was randomly found in google and was added to illustrate the idea, this image doesn't belong to me

Game description:

Embark on a crustacean adventure like no other in "Empire of Crabs"! Dive into a captivating underwater world where you take command of a vibrant and resourceful crab colony. Your mission? Build, strategize, and conquer to create the ultimate crab empire!

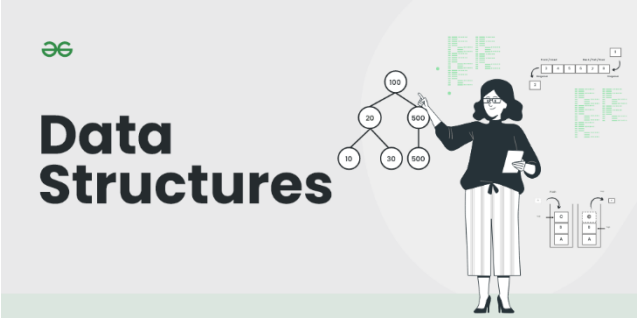
As the wise and ambitious leader of your crab civilization, you'll navigate the challenges of the ocean depths, from treacherous currents to rival crab factions. Harness the unique abilities of your crab subjects, each with their own strengths, to gather resources, construct intricate habitats, and defend your territory against predators.

Explore a visually stunning and immersive aquatic realm, where vibrant coral reefs, mysterious shipwrecks, and ancient ruins await your discovery. Collect rare treasures, unlock hidden secrets, and uncover the history of your crab civilization as you expand your dominion.

But it's not all peaceful tides and clear waters. Engage in epic battles against rival crab colonies for dominance over coveted underwater territories. Strategize your tactics, deploy your crab forces, and outmaneuver opponents in real-time battles that will test your leadership skills and wit.

game_id	game_name	published_date	age_ratings	languages	price	genre 1	genre 2	genre 3	size_mb
1	Empire Of Crabs	2019-08-10 00:00:00.000	4+	English, Hebrew	0	strategy	simulation	NULL	8493

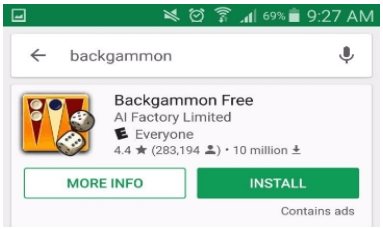
2. Data Base Structure



2.1 Database Tables

The data is stored in 9 different tables

1. GameInstall



Holds data about all the users that **installed** the game at least once

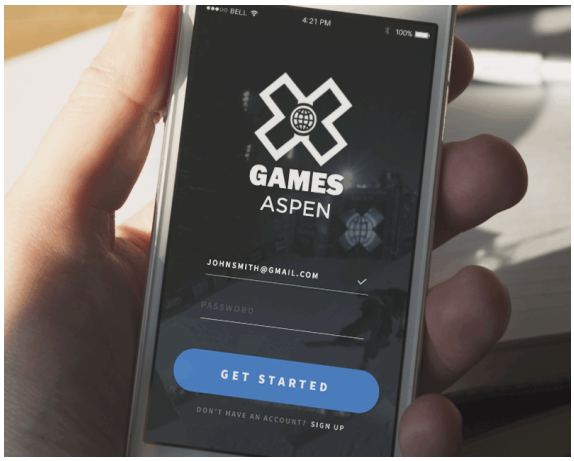
InstallID	user_id	install_date
1	6530	2020-11-07 00:00:00.000
2	11285	2020-01-28 00:00:00.000
3	848	2020-12-21 00:00:00.000
5	7396	2019-11-04 00:00:00.000
6	10694	2020-02-07 00:00:00.000

2. Users

Holds data about all the users that also **opened an account** (the other option is to sign in as "Guest")

user_id	date_registered	country	birth_date
10001	2021-12-29 00:00:00.000	Russian Federation	1992-09-01 00:00:00.000
10002	2020-08-10 00:00:00.000	Chile	2003-01-11 00:00:00.000
10003	2020-03-23 00:00:00.000	France	1967-09-12 00:00:00.000
10004	2020-01-30 00:00:00.000	Chile	1970-10-05 00:00:00.000
10005	2022-06-26 00:00:00.000	India	1967-04-04 00:00:00.000
10006	2021-09-02 00:00:00.000	Mexico	1966-09-18 00:00:00.000

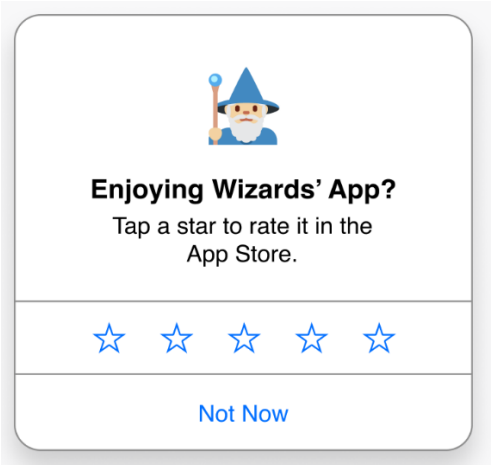
3. Log_In



Holds data about when different users **logged in** and used the app and for **how long** (in minutes)

log_id	log_in_date	log_off_date	minutes played	user_id
1	2022-04-05 00:00:00.000	2022-04-05 01:40:00.000	100	17969
3	2021-02-10 00:00:00.000	2021-02-10 01:20:00.000	80	10384
4	2021-10-22 00:00:00.000	2021-10-22 00:40:00.000	40	14386
5	2020-05-03 00:00:00.000	2020-05-03 01:37:00.000	97	8862

4. GameRatings



Holds data about what **rating** each user gave to the app (if the user chose to do so)

rating_id	user_id	rating_date	rate
33	7260	2020-09-23 00:00:00.000	3
37	1997	2020-03-19 00:00:00.000	4
42	6747	2020-06-15 00:00:00.000	2
43	5500	2020-04-30 00:00:00.000	1

5. Marketing

Hold data about all the money that the company use to **advertise** the game

marketing_id	campign	start_date	end_date	cost_per_month
1	facebook_ads	2020-01-07 00:00:00.000	2020-03-06 00:00:00.000	10
2	creative	2020-01-07 00:00:00.000	2021-01-05 00:00:00.000	5
3	app_store_fees	2020-01-07 00:00:00.000	2021-01-05 00:00:00.000	3

6. PremiumItems

Holds data about the special items or features that the user can **purchase with real life money**

premuim_item_id	premuim_item_name	item genre	cost
1	Green Crab	skin	2.3
2	Crown	power-up	3
3	Aurora Shell Shimmer	skin	4
4	Oceanic Elegance	skin	6.5
5	Enigmatic Moonstone Claw	power-up	2.3
6	Cursed Relic	power-up	4.99
7	Crystal Cavern Chateau	house	5.3

7. Purchases

Holds data about the **purchases users made** (using real life money)

purchase_id	purchased_item_id	user_purchased	purchase_date
1	15	12164	2020-06-05 00:00:00.000
25	15	5274	2020-08-20 00:00:00.000
50	15	5071	2020-05-28 00:00:00.000
52	15	4485	2020-03-01 00:00:00.000
55	15	6931	2020-07-09 00:00:00.000
57	15	4812	2020-01-09 00:00:00.000

7. game economy

Hold data about the digital in app **currency inflows and outflows**.
the digital currency we can call "CrabCoins", so the "CrabCoins" that were generated by various actions made and milestone achieved by the users actions in game and the "CrabCoins" that were used by the users and disappeared into the void (or sink, whatever term you like).

action_id	user_id	action_date	action_desc	effect
1	10001	2020-01-01 00:00:00.000	first_log_in reward	25000
2	10001	2020-01-01 00:00:00.000	battle	5000
3	10001	2020-01-01 00:00:00.000	purchased blue crab	-10000
4	10001	2020-01-01 00:00:00.000	purchased iron sword	-2500
5	10001	2020-01-01 00:00:00.000	battle	7500
6	10001	2020-01-01 00:00:00.000	purchased blue crab	-10000
7	10001	2020-01-01 00:00:00.000	battle	1250
8	10001	2020-01-01 00:00:00.000	battle	5000
9	10001	2020-01-01 00:00:00.000	milestone - tier 2 clan	50000
10	10001	2020-01-01 00:00:00.000	battle	7500
11	10002	2020-02-01 00:00:00.000	purchased blue crab	-10000
12	10002	2020-02-01 00:00:00.000	milestone - tier 3 clan	75000
13	10002	2020-02-01 00:00:00.000	found magic sea-shell	10000

2.2 Data Cleaning and Validation

As consequence of the data being randomly generated, I had to build queries in SQL to make sure that data make sense,
for example:

a user cannot have data in the log-in table on a date that is **earlier than the date he installed the app** for the first time.

Installs table -

InstallID	user_id	install_date
1	6530	2020-11-07 00:00:00.000
2	11285	2020-01-28 00:00:00.000
3	848	2020-12-21 00:00:00.000

Log_in table -

	user_id	log_in_date	log_off_date	minutes played
1	6530	2021-11-29 00:00:00.000	2021-11-29 00:28:00.000	28
2	6530	2019-12-20 00:00:00.000	2019-12-20 00:12:00.000	12
3	6530	2019-12-21 00:00:00.000	2019-12-21 00:32:00.000	32
4	11285	2021-02-23 00:00:00.000	2021-02-23 00:10:00.000	10
5	11285	2022-04-12 00:00:00.000	2022-04-12 00:43:00.000	43
6	11285	2020-12-22 00:00:00.000	2020-12-22 00:10:00.000	10

Query to validate the data

```
WITH cte_first_install
AS
        (
        SELECT user_id, min(install_date) AS "FirstInstall"
        FROM GameInstalls
        GROUP BY user_id
        )
DELETE FROM Log_In WHERE log_id IN
    (
    SELECT L.log_id
    FROM cte_first_install FI JOIN Log_In L ON FI.user_id = L.user_id
    WHERE FI.FirstInstall > L.log_in_date
    )
```

*There were 8 other queries to validate the data but I did not include them in this file

3. Analysis

The database contains dates from the years 2019-2022 but the analysis will be based on data from the year **2020 only,** a yearly analysis you could say.

3.1 Descriptive Statics

I will start the analysis section with some simple analysis to give a little snapshot about the mobile app performance in 2020.
just to clarify I will add and say that the app was made by 1 person and the only expenses are spent on marketing.

Users Statics

Number Of Distinct Users That Installed The App:
32,424
Number Of Users That Opened An Account:
3,420
Number Of Diffrent Countries (Registered Users Only):
35
Average Age (Registered Users Only):
18

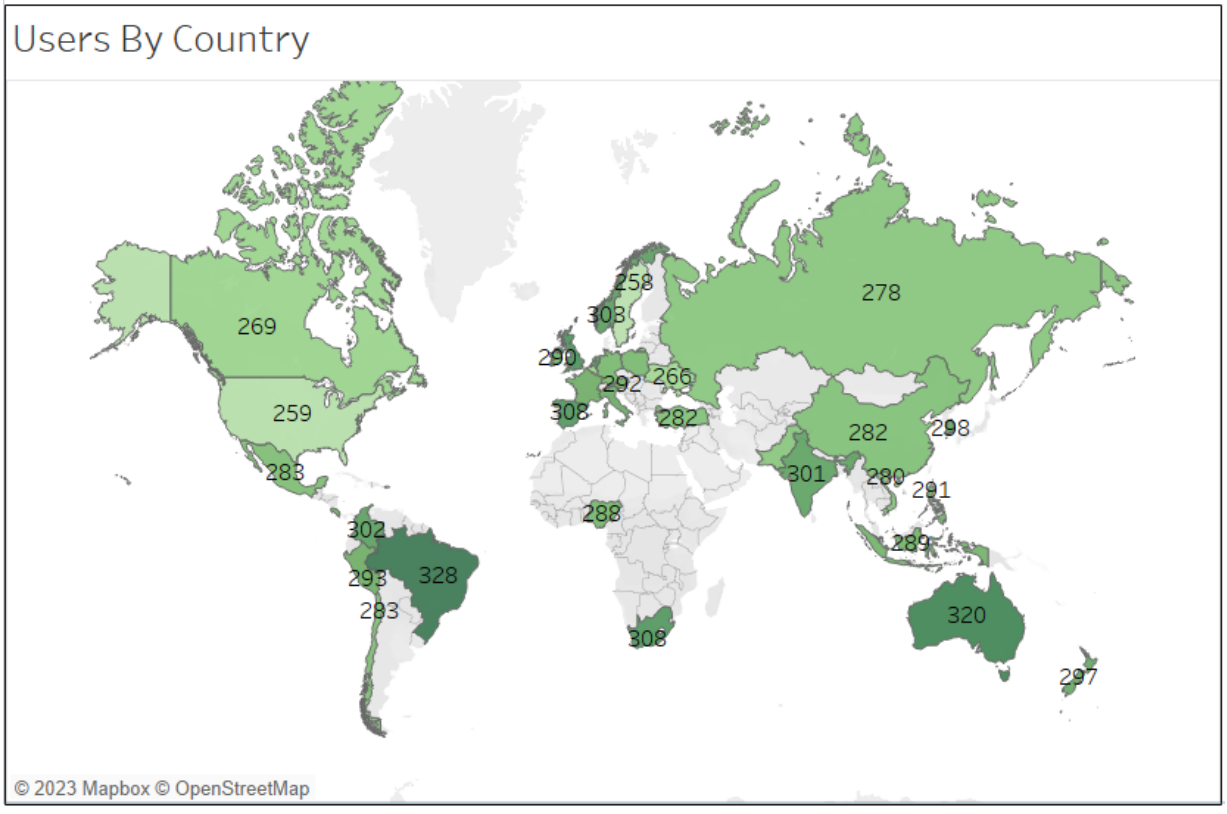
Ratings Statics

Average Rating:
2.9 ★
Number Of Ratings:
368

Revenue Statics

Total Revenue:
$1,332
Total Expenses:
$116
Number Of Purchases:
166
Avg Purchase Amount:
$8.03

Users By Country



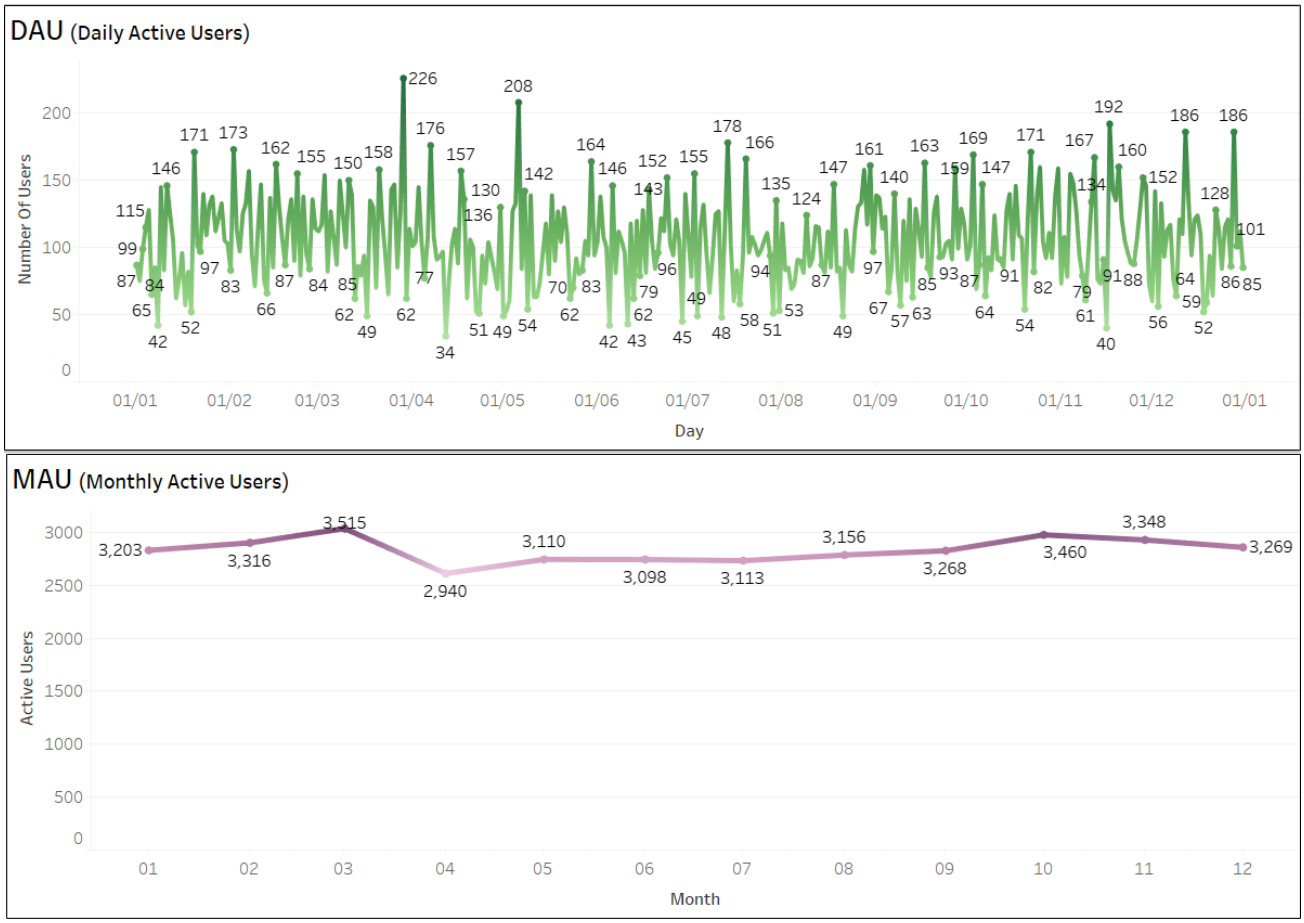
3.2 User Engagement

DAU

daily active users graph shows the number of **distinct** users that logged in for each **day** in 2020.

MAU

monthly active users graph shows the number of **distinct** users that logged in at least once for each **month.**

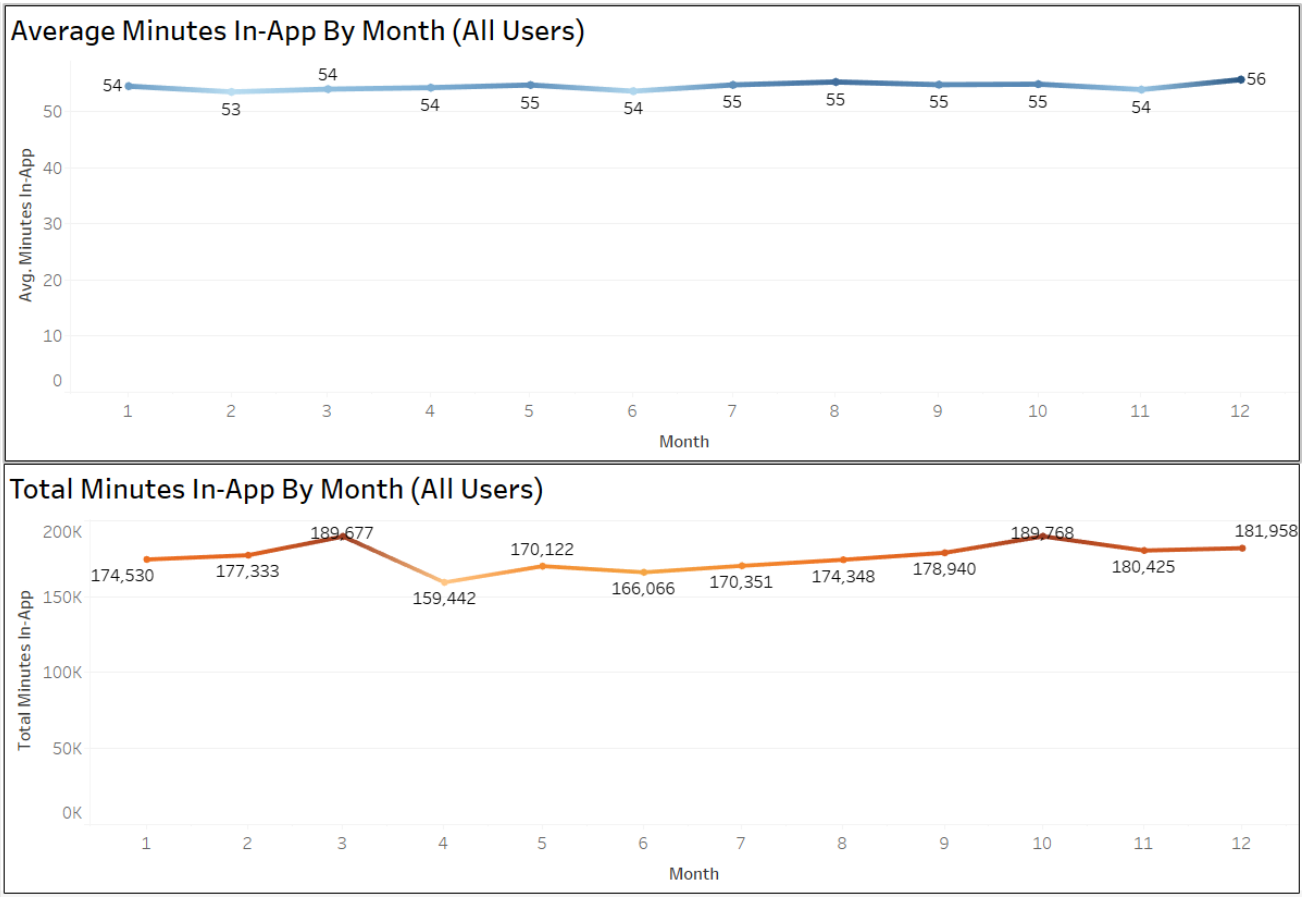


From the DAU graph we can notice that there is **high scattering in term of daily active** users across different days.

Unlike the DAU graph in the MAU graph the number of total distinct users each month is steadier and mostly balanced, with around 3,100 distinct users per month.

Session Duration

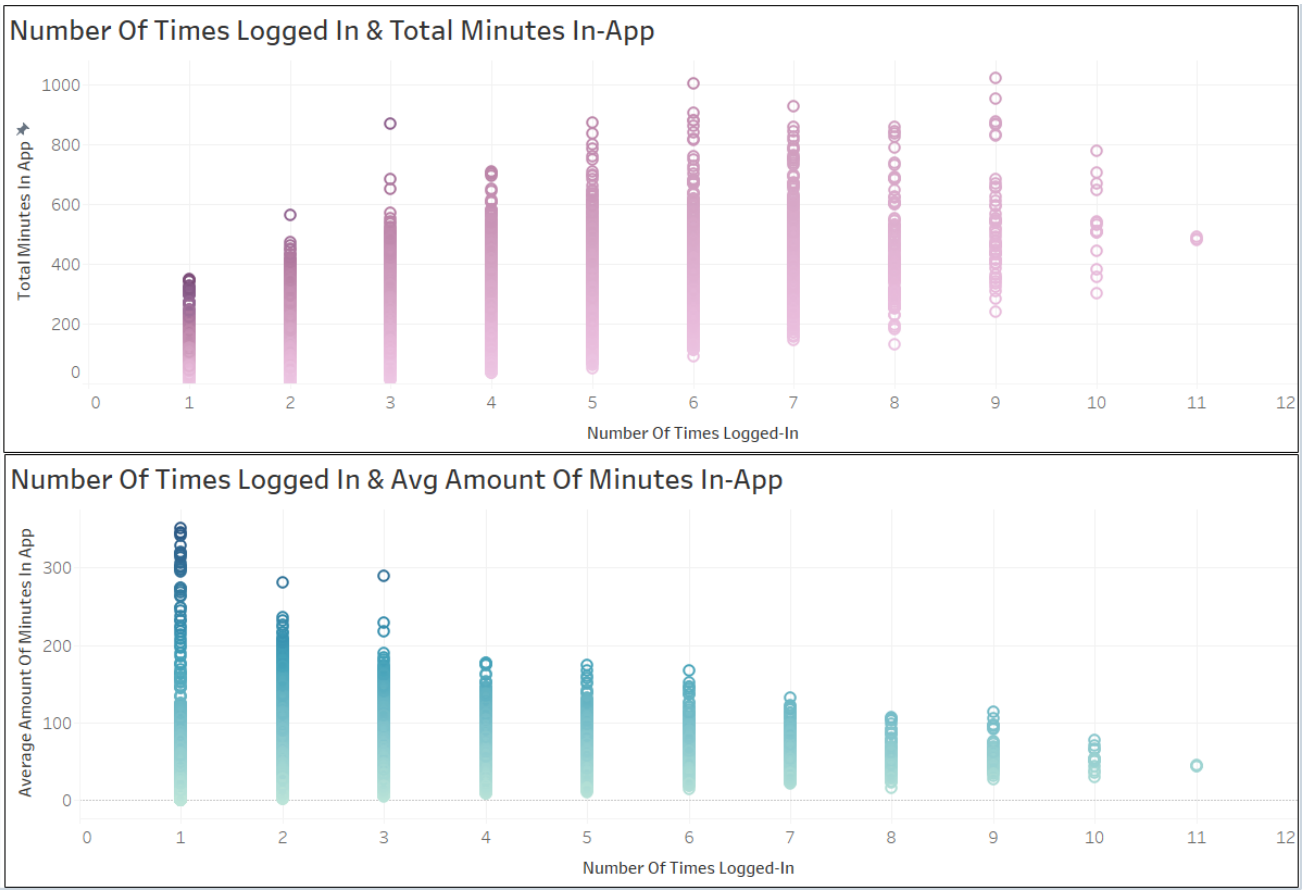
The amount of time a user spends actively using the app during a single session.



From these graphs we can note that are no significant changes in the average amount of time users spent in the app in different months, although there is one noticeable decline in total minutes between March and April, while the average minutes in-app did not change at all between these months we can assume from that a **decrease in the total number of distinct users between March and April**.

Frequency

How often users return to the app over a given period of time (in my project the period is one year).



From the first graph we can observe a slight trend – higher number of log in's (right side of the graph) result in more total minutes spent in-app which make sense but perhaps the difference is not significant enough.

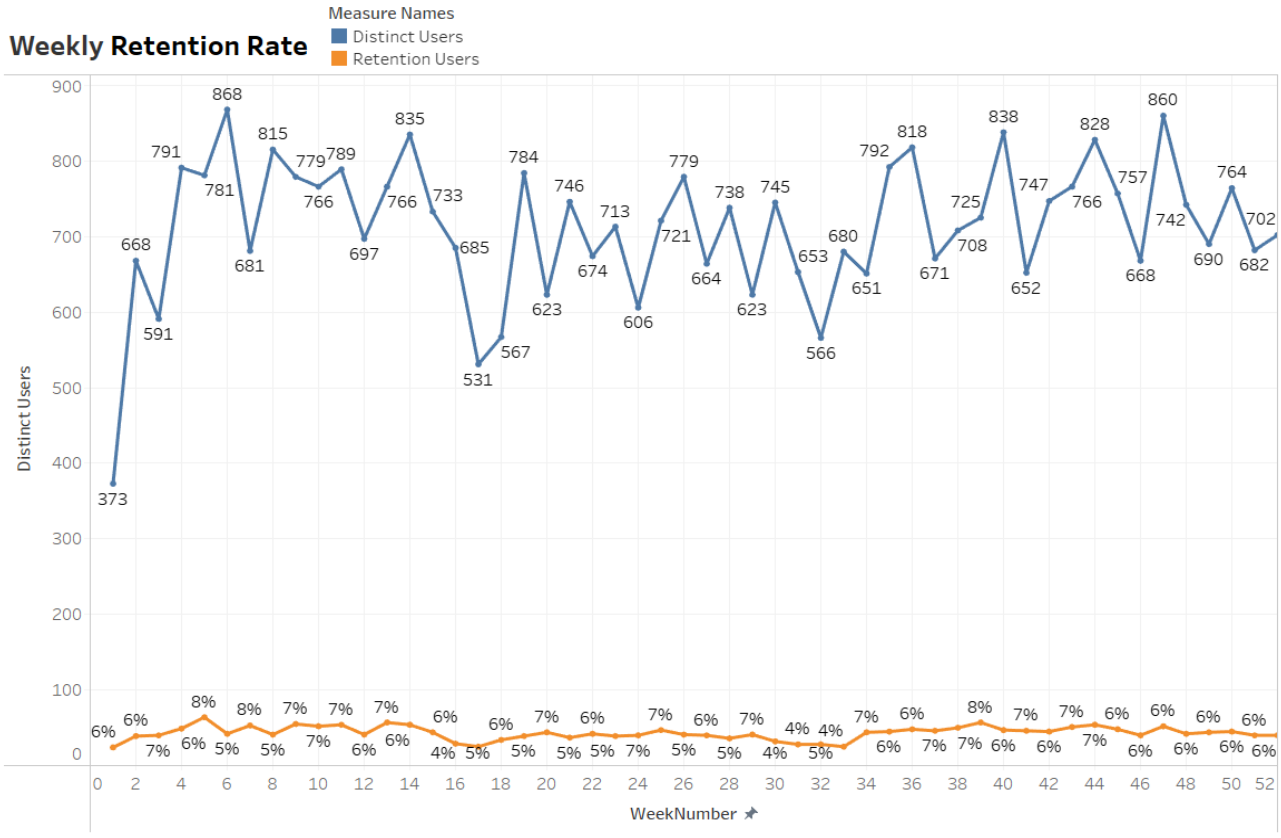
Contrary to the first graph, looking at the second graph we can observe a clear pattern – **users with higher number of log-in's has lower average amount of minutes in-app**.

Retention

Retention rate refers to the **percentage of customers or users who continue to engage with a product**, service, or platform over a specific period of time. It is commonly used as a key performance indicator (KPI) to measure the effectiveness of an organization's efforts in retaining its existing customer base.

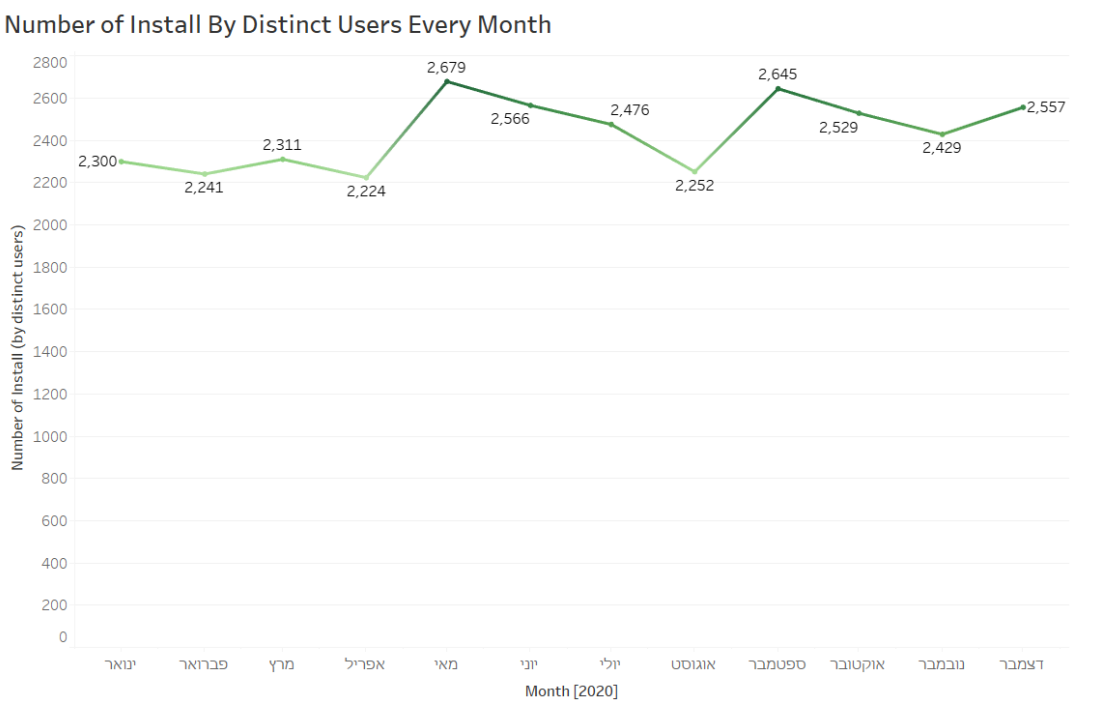
Retention rate can be calcuated based on user segments in terms of user activity patterns, for example:
a segment for daily log-in user, a segment for weekly log-in user and so on..
the higher number of days a "daily log-in" user does not log the higher chance he will become " churned customer " (Churn is the opposite of retention). For a weekly log in user it's the same idea but with weeks.

I decided to calculate without dividing the users for segments as I didn't have enough data and quality data (the data is randomly generted and does not represent a real situation) so I decided to calcuate it on **weekly basis for all users** - how many users logged in and logged in again the following week.



Observing the graph we can notice **high scattering in the number of distinct users every week**, but a balanced percentage in the weekly retention rate which may indicate a **percentage of loyal customers between 4% and 7%.**

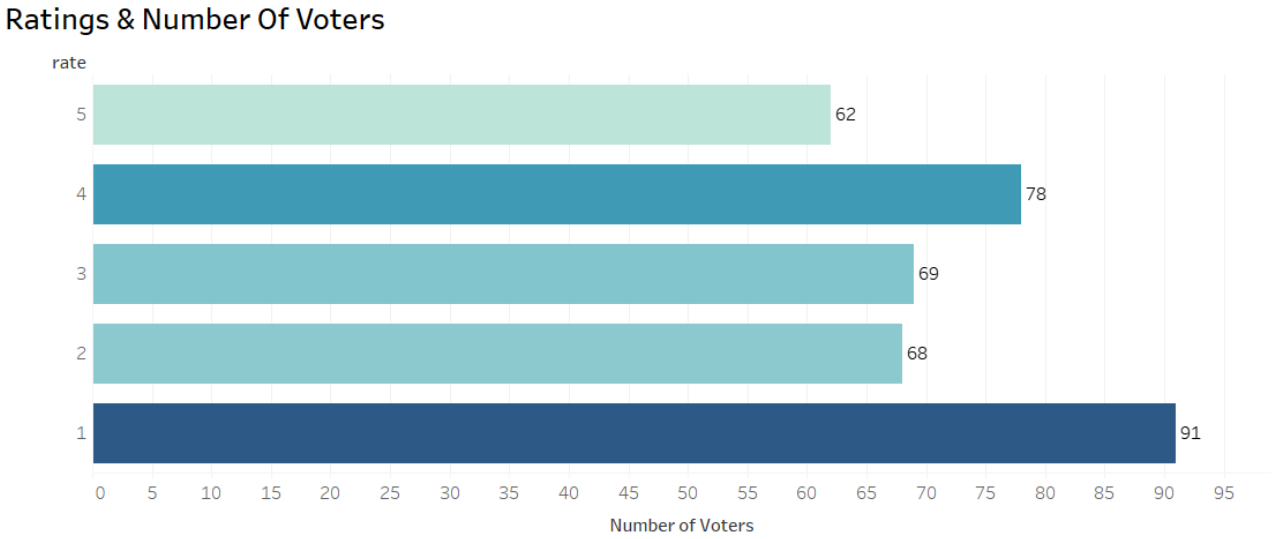
Game Installs



Rating

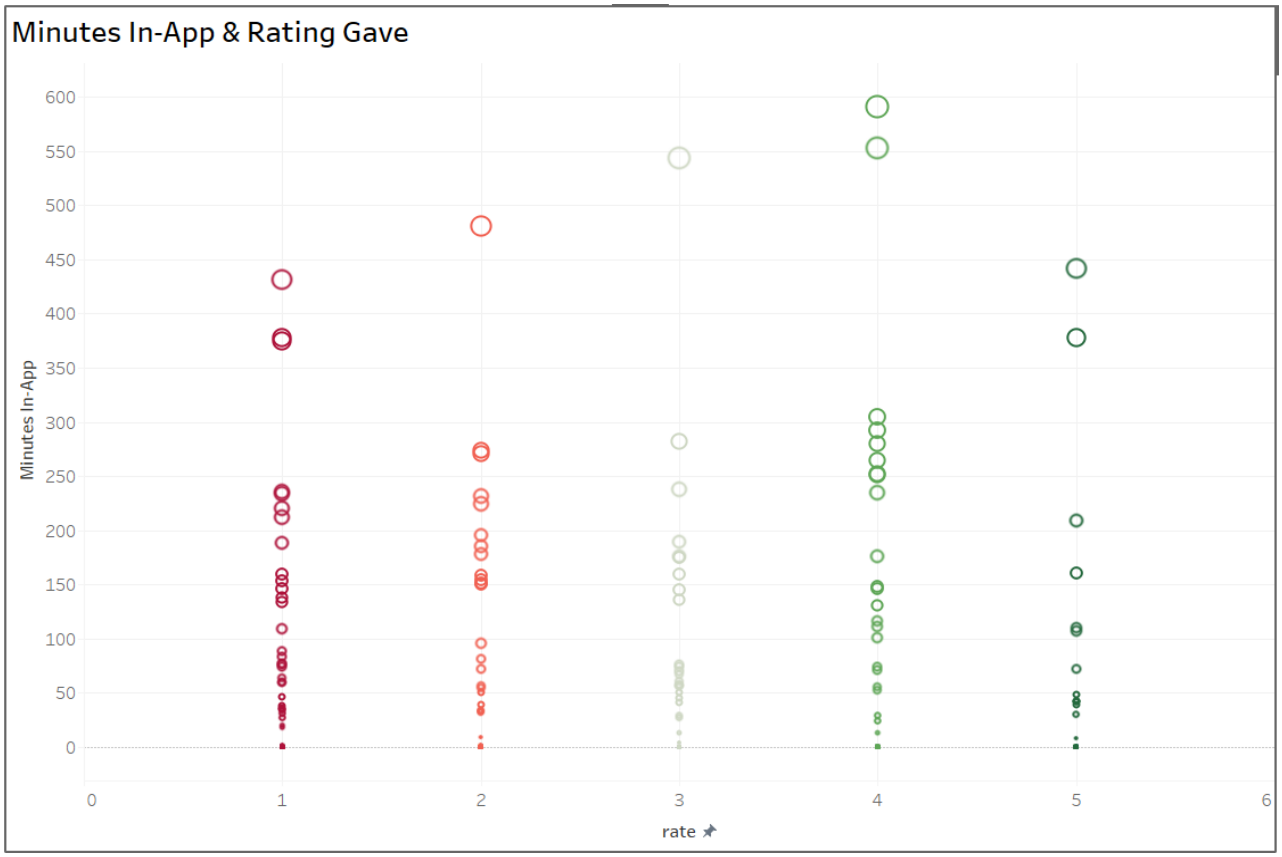
Mobile apps and online platforms often allow users to rate and review apps, products, or services. These user ratings help others decide whether to use the app or service.

The following graph shows the how users rated the app.



The graph is mostly balanced with one expectation which is the number of users that rated the app "1", clearly some users have serious a problem with the app and that should be looked into.

The following graph shows the rating each user gave compared to the total duration he spent in-app. **bigger circles indicate more based opinions**.

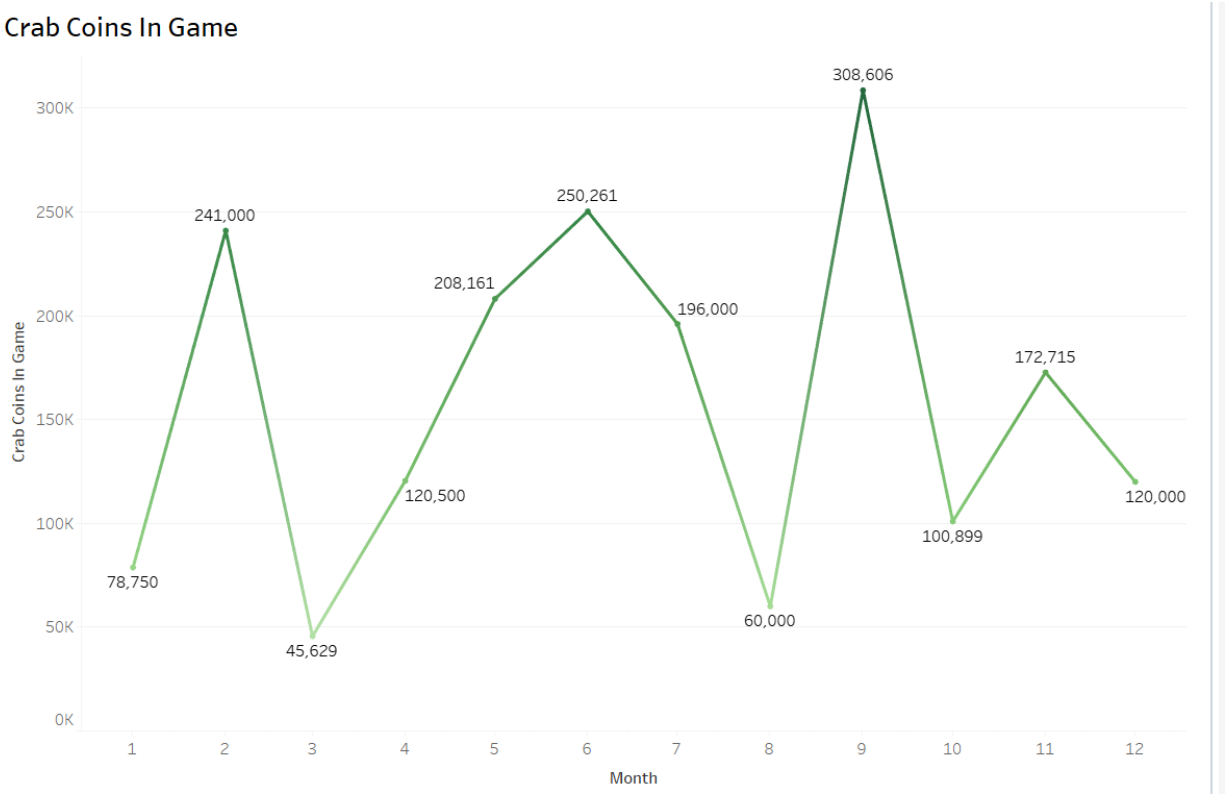


3.3 Game Economy

Effective mobile game economy design aims to create a virtuous cycle where players are motivated to engage, generate the game currencies / resources and enjoy spending it.

The currency in game should have meaning so we don't want to have it inflated too much, but at the same time we also should consider not frustrate the users by making the currencies / resources too scarce. The goal is to strike the right balance.

The following graph

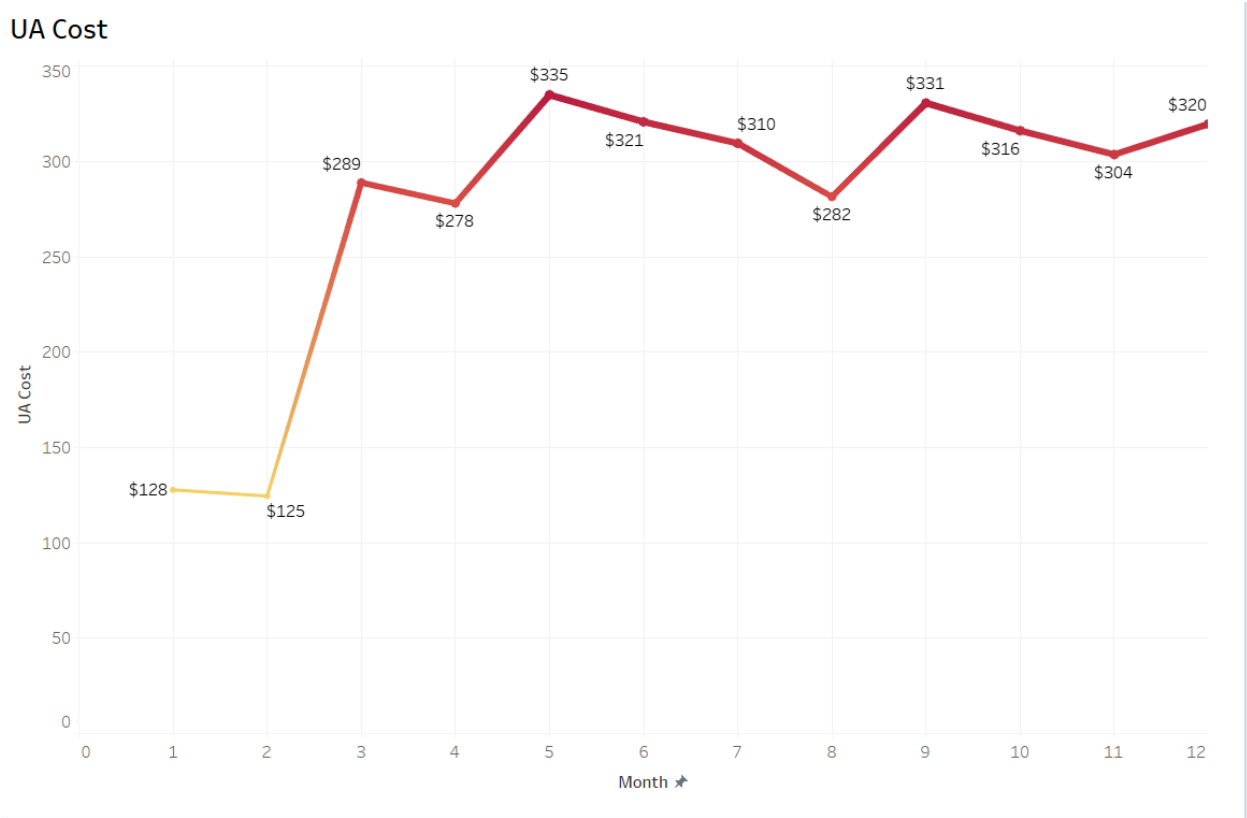


3.4 Expenses

UA Cost

User acquisition is a metric used in marketing and business to measure the cost of acquiring new customers, users, or clients for a product, service, or platform.

User Acquisition Cost = Total Marketing and Advertising Expenses / Number of New Customers Acquired



Judging the graph there is a **big jump in UA cost from February onward** that should be looked into.

As I mentioned earlier in this project, the game was developed by a single developer