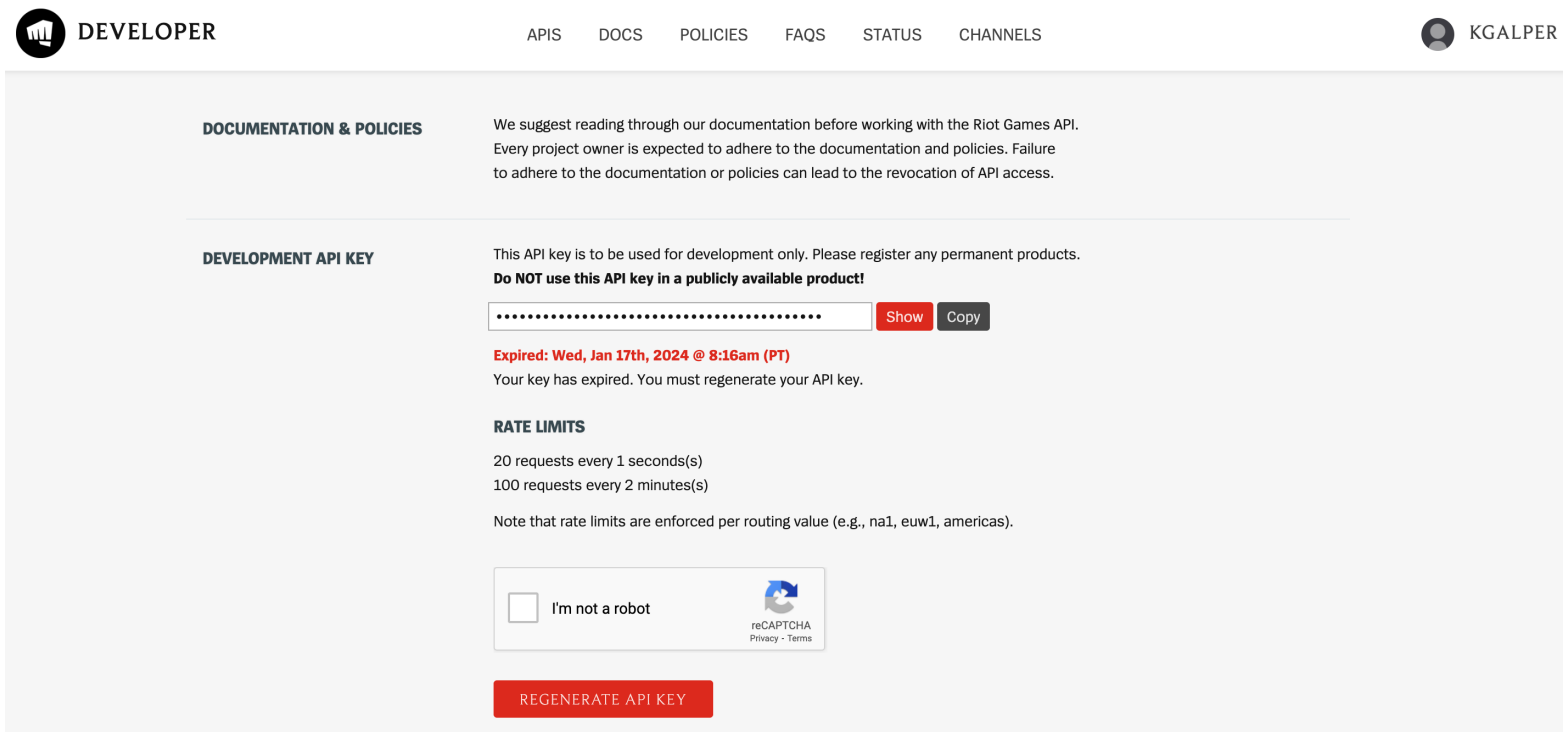
My League of Legends Playtime

Riot API

We need a daily API key.



Dealing With the Request Rate Limits

- We need an iteration to deal with the rate limits.

RATE LIMITS

20 requests every 1 seconds(s)

100 requests every 2 minutes(s)

```python
def make_request(url, params):
    while True:
        response = requests.get(url, params=params)

        if response.status_code == 200:
            return response.json()
        elif response.status_code == 429:  # Rate limit exceeded (100 request in every 2 minutes)
            print("Rate limit exceeded. Waiting for 10 seconds.")
            time.sleep(10)  # Wait for 2 minutes before retrying
        else:
            print(f"Request failed with status code {response.status_code}. Exiting.")
            break
```

Dealing With the Request Rate Limits

```python
def get_recent_matches(puuid, startIndex):

    count = 50
    match_url = f"https://europe.api.riotgames.com/lol/match/v5/matches/by-puuid/{puuid}/ids?start="+ str(startIndex)+"&count="+str(count)
    match_url += f"&api_key={api_key}"

    matches = make_request(match_url, {})

    return matches

player_id = input("Please enter your player ID: ")
summoner_url = find_summoner_url(player_id)
summoner_data = make_request(summoner_url, {})
startIndex = 0
if summoner_data:
    player_puuid = summoner_data["puuid"]
    matches = []
    while startIndex < 1000:
        matches += get_recent_matches(player_puuid, startIndex)
        startIndex += 50
```

league_of_legends_dataset.csv

- For modularity, I needed to create my own data set with the Riot API.

1	GameStartTime	GameEndTime	GameDuration	DateOfGame
2	2024-01-16 22:36:38.905	2024-01-16 23:08:27.499	1908	2024-01-16
3	2024-01-16 21:59:13.664	2024-01-16 22:28:08.934	1735	2024-01-16
4	2024-01-16 21:27:31.718	2024-01-16 21:54:08.575	1596	2024-01-16
5	2024-01-16 21:20:32.622	2024-01-16 21:22:17.681	105	2024-01-16
6	2024-01-16 20:28:12.074	2024-01-16 21:07:33.509	2361	2024-01-16
7	2024-01-16 19:43:50.963	2024-01-16 20:07:21.613	1410	2024-01-16
8	2024-01-16 19:13:40.748	2024-01-16 19:41:21.947	1661	2024-01-16
9	2024-01-16 18:12:34.771	2024-01-16 18:38:01.740	1527	2024-01-16
10	2024-01-15 20:25:47.120	2024-01-15 20:59:23.577	2016	2024-01-15
11	2024-01-15 19:47:36.955	2024-01-15 20:19:35.209	1918	2024-01-15
12	2024-01-15 18:51:40.888	2024-01-15 19:37:21.169	2740	2024-01-15
13	2024-01-15 18:09:09.454	2024-01-15 18:43:36.545	2067	2024-01-15

Listing Playtime in Hours

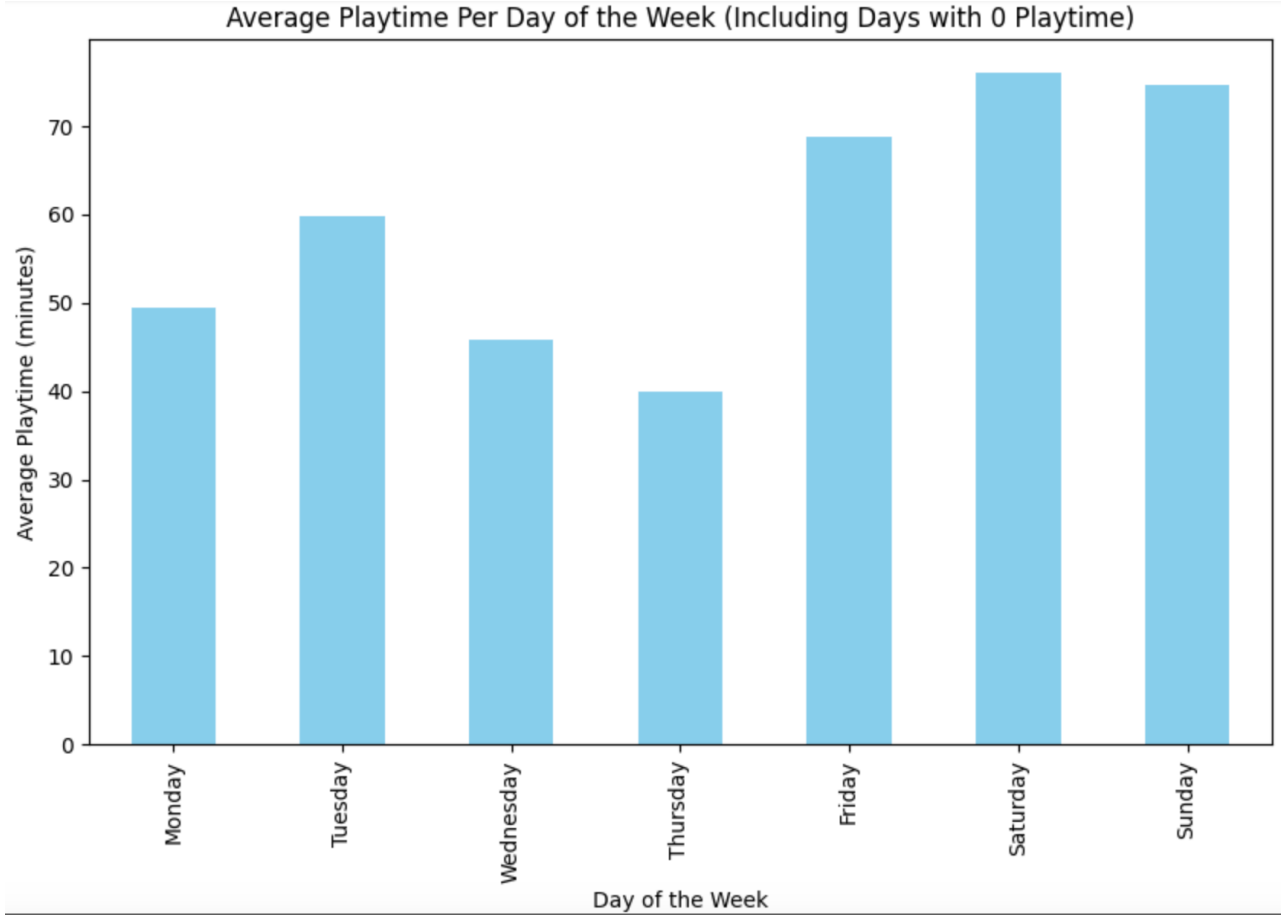
- The dates that I didn’t play the game are not included in the dataset, thus we need to modify the data set when we are using it.
- Fill these dates with zero game duration.

```
16 # Replace NaN values with 0 for playtime
17 merged_df['GameDuration'] = merged_df['GameDuration'].fillna(0)
18
```

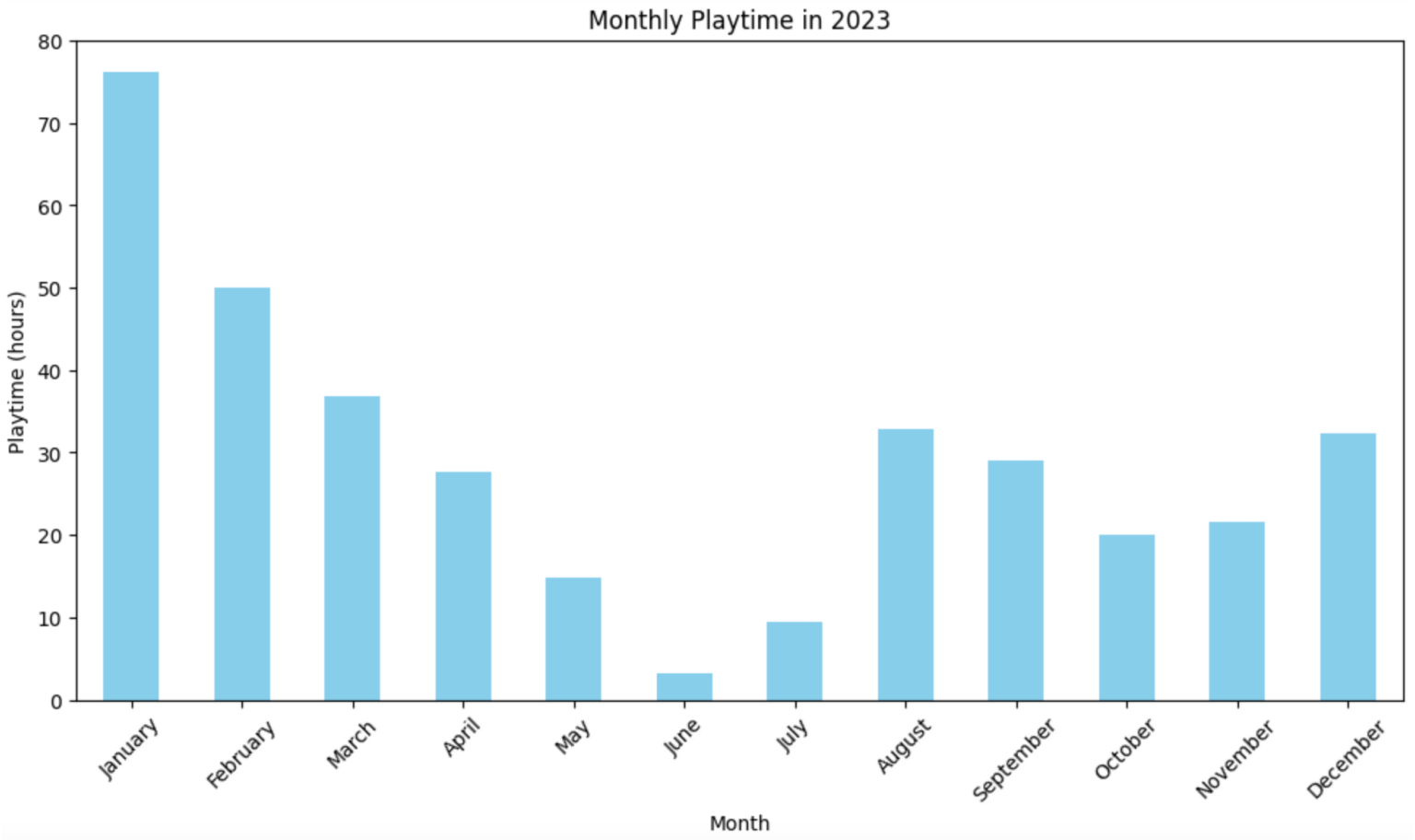
Listing Playtime in Hours

```
2023-11-14: 1.07 hours
2023-11-15: 0.0 hours
2023-11-16: 0.0 hours
2023-11-17: 1.03 hours
2023-11-18: 3.95 hours
2023-11-19: 2.52 hours
2023-11-20: 0.95 hours
2023-11-21: 0.79 hours
2023-11-22: 0.0 hours
2023-11-23: 0.0 hours
2023-11-24: 0.0 hours
2023-11-25: 0.0 hours
2023-11-26: 0.0 hours
2023-11-27: 0.0 hours
2023-11-28: 0.0 hours
2023-11-29: 0.76 hours
2023-11-30: 0.98 hours
2023-12-01: 0.0 hours
2023-12-02: 0.38 hours
2023-12-03: 0.39 hours
2023-12-04: 1.57 hours
```

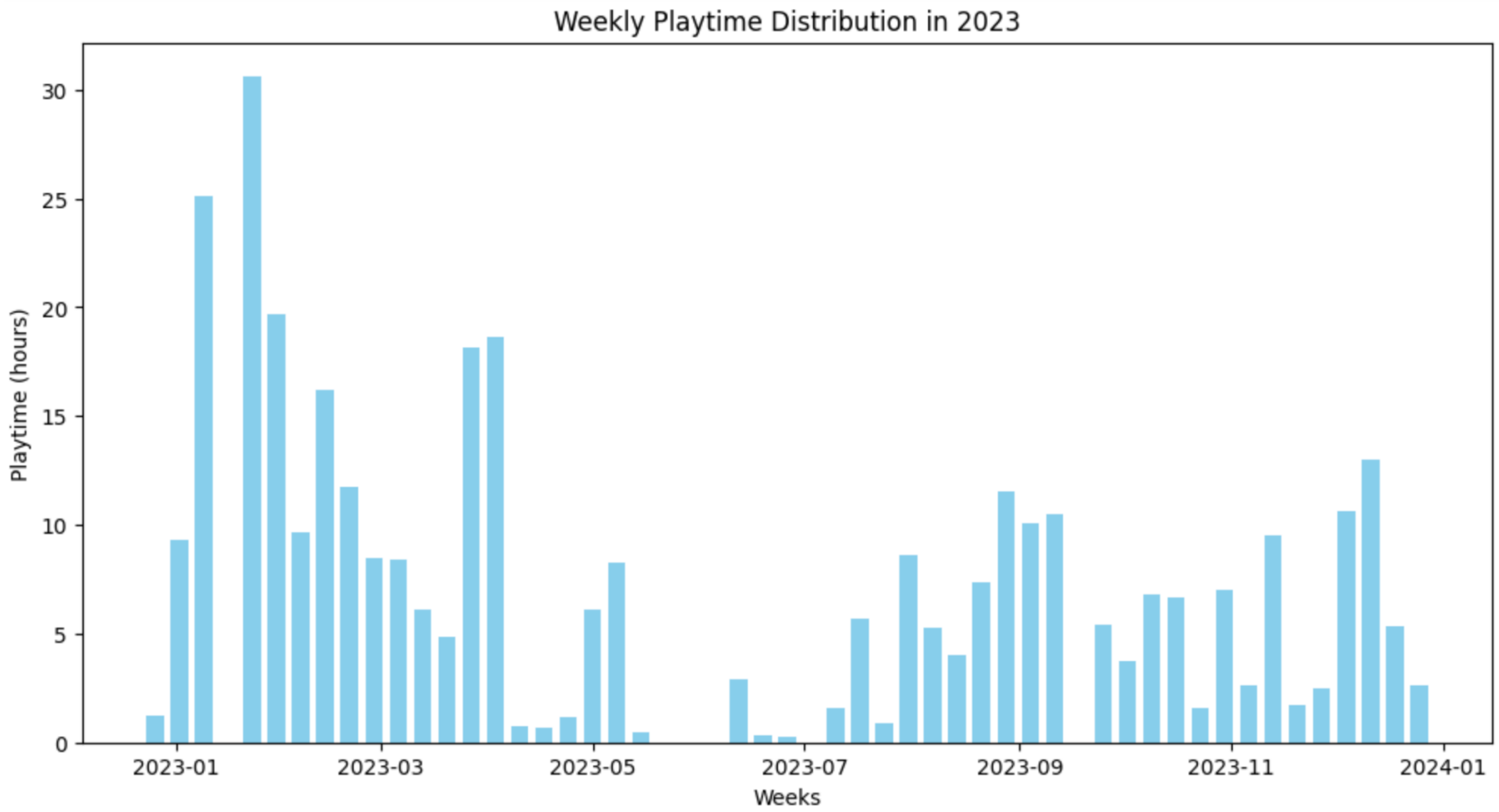
I have more time to play in weekends, clearly



January and Feburary are very high since it is after the final exams



I played a lot in the beginning of the new year and in semester breaks, however I did not spend too much time in summer



In the conclusion

My hypothesis is that in 2023, my playtime in League of Legends is smaller than 1 hour. Which is true due to my calculations.

```
Mean Daily Playtime: 0.97 hours
```

I hope you liked it

Thank you for your time