משרד התעשיה המסחר והתעסוקה

מה"ט המכון הממשלתי להכשרה בטכנולוגיה ובמדע.



Major: תוכנה ס'

Final Project

Student name: Giora Glovatsky

Student ID: 314956046

Student name: Netanel Masfin

Student ID: 318611985

**המנחה: עדי צימרמן**

**המנחה: שי אברהם**

Documents

Table of Contents

Table of Contents

[Project Name 4](#_Toc185179188)

[Background 4](#_Toc185179189)

[General description 4](#_Toc185179190)

[System goals and objectives 6](#_Toc185179191)

[Current Market State and Challenges 7](#_Toc185179192)

[Current market state 7](#_Toc185179193)

[Challenges in the current market 9](#_Toc185179194)

[What We Intend to Renew or Enhance 10](#_Toc185179195)

# Project Name

After careful consideration and deliberation, we have chosen the name “Poker Mate.” These two words hold significant meaning for the app.

“Poker” is a direct reference to the subject matter of the app, making it easier for new players to find an app that aligns with their interests.

“Mate”, a friend, reflecting our intention to create an app that will act as a friend, helping you with issues or playing with it.

# Background

## General description

We’d like to provide some basic context for the development of this app.

Hey there! I’m Giora, and I’m a poker enthusiast. When the pandemic (COVID-19) hit, me like many others, turned to poker for entertainment. As me and my friends kept playing more and more, I started to notice a bit of a problem that was reoccurring.

After a game was over, when we wanted to close it (summarize the wins and losses and transfer the money from the losers to the winners), I saw we had a few issues.

1. **Money is not adding up**: Sometimes the money didn’t add up buy-ins ≠ buy-outs.
2. **Suboptimal logic for who owes who money**: The logic of how best and most efficiently settle debts between players was hard.  
   In a small game this issue is straightforward to resolve. However, in a larger game with more players, optimally settling debts becomes a complex challenge.

(make as little transfers as possible)

1. **Disorganized money transfers**: Some players in a game decided to send money to specific winners privately, leading to others transferring money to the same winners. Some winners expected money from specific losers, while losers had already paid to different winners. Obviously, this caused confusion...

When I realized that I was playing more regularly and had to consistently deal with these issues (since I’m the one who usually organized the games), I decided I needed to find a tool to help me with that.

In today’s digital age, I was confident that I would find an app to assist me with these challenges, but unfortunately, I did not.

I explored various apps, but none of them met my specific requirements.

At that point, I became intrigued by programming and decided to attempt creating a simple program to address the issues I’ve mentioned above.

I picked up Python and, with the assistance of AI, I created a program that had several flaws but helped me organize the game debts (who owes who). However, to use it, I had to modify the code itself, which meant that every time I needed to input game data, I had to type it manually into the hard code.

Additionally, while I made the app suggest various solutions to the problems that arose, it didn’t implement any of them by itself which meant I had to go back to the hard code again to implement the suggested solutions.

After ending my first year in Ruppin, in the programming major I’ve decided to test my skills and transfer the program from python to C# using all the things I’ve learned and more as I was much more commutable with the language.

I have developed a Console Application from scratch, enhancing several aspects of the Python application.

Below is a detailed list of the improvements I have made to enhance the usability and enjoyment of the console application:

1. User-input string data instead of changing the hard code itself
   1. Format: Name, buy-in, buy-out (if no buy-out, then buy-out == 0)
   2. Smart name recognition (e.g.: “gio” == “Giora”, “ro” == “Ron” / “Liron” (will ask the user to choose), however if the user types an exact name no prompt will be asked to choose (e.g.: “ron” == “Ron” ≠ “Liron”)
2. Improved problem-solving mechanisms.
   1. Added more mechanisms.
   2. Added smart mechanisms. (Will suggest a solution based on the total buy-out of the players compared to the largest winner / loser depending on the problem)
   3. Added mechanisms implementation. (Will implement the solution and rerun the game with new data from solution)
3. Creation of text databases
   1. Game History
      1. GameID, Game day, Game location, Program RT (runtime), Game Resolved (was a solution used), Cash on table, player stats, game debts, records broken (if any)
   2. Milestones
      1. Most chips on table per game with game data
      2. Most win per player with player data and game data
      3. Most loss per player with player data and game data
   3. Profit/Loss Tracker
      1. Tracks the current gain or loss of a saved player (in the dictatory of the code)
4. Error detection and resolution throughout the program
   1. If data isn’t adding up, (e.g.: wrong format, data was too long, etc.…) a prompt will be shown to the user with the error and ask for new data.
5. Utilization of an efficient algorithm to settle debts (superior to the algorithm implemented in Python), The algorithm consists of 3 stages to sum up the game.
   1. Settle equal losers with equal winners (x loss == x win)
   2. Settle several losers with a single winner that won the amount of all losers using DFS (x loss + y loss + z loss == b (b = x+y+z) win)
   3. Settle the remaining players using the following logic (biggest losers => biggest winner)
6. Incorporation of additional sentences for entertainment purposes (A sentence that will be randomly chosen depending on the winning of the biggest winner and shown at the end of the program)
7. Pot settling method that will help divide the pot/s to x players.
8. Remainder handling while considering who gives / gets the remainder based on current profit / loss.

## System goals and objectives

There are 3 main functionalities for the application:

1. **Play** - Be able to play with friends (up to 9 people), Can work on a mutual network, one administration manages the game (controls the settings), Game can be saved to DBs after confirmation because sometimes you play for fun (friendly) if you do not want it saved if you want to track your “real game” (normal game).
2. **PokerSolverPro** - Be able to calculate an existing game of poker that you’ve played before with friends / other people to settle debts / After an active session in the “Play” functionality.  
   And **Chip Chop** a tool for splitting X pots to Y players.
3. **Track your progress** - Games you played (online & games you’ve inputted your name (after confirmation) PokerSolverPro function.

Why after confirmation? Because your name might be used for a different player.

# Current Market State and Challenges

## Current market state

The primary challenge in the current market for Poker apps is the lack of emphasis on home games and players who prefer playing without intermediaries like sites or casinos.

There are several advantages to playing a home game.   
Firstly, there are no fees involved.   
Secondly, you can play with familiar people.   
Thirdly, you have complete control over the game’s progression.

Additionally, as far as we’ve observed, there isn’t a single app specifically designed to settle poker debts among players. This is a niche topic because in a casino, the amount of chips you deposit into the cash window is the amount you’ll receive back. If you lose a chip or gain an extra one by mistake, no one will notice. However, in a home game, everything must be perfectly balanced. The amount of chips players bring into the game should be the same as the amount they leave with (all buy-ins should equal all buy-outs).

We’ve tested various apps on both Android and iOS devices to identify relevant apps in our market segment. Here are a few notable apps that are somewhat related to our app and could potentially be competitors:



**Main competition: Chips of Fury:**

**Pros:**

* Simple and intuitive user interface (GUI).
* Excellent user experience (UX).
* Offers a wide range of configurations.

**Cons:**

* Absence of crucial poker logic that is game-ruining and unacceptable in professional settings.  
  Specifically, the game lacks the ability to handle **some** all-in situations correctly. When there’s a round of raises, and the last player in the round goes all-in **and** his all-in is under the minimum bet, the game allows the player after them to re-raise even though they should only have the option to call or fold. This flaw creates an illegal betting round and gives players who are “trapping” an unfair advantage by potentially blowing up the pot before they’ve even seen any cards on the board.
* Absence of time chips: Another issue is the absence of the feature to give players “time chips” when there’s a timer for each action. Additionally, the game lacks support for additional game modes, such as Bomb pot Omaha 4/5 cards and Pineapple.
* Game-breaking bugs: For instance, when a player shows their card, in the next round, one of their entire cards is exposed to all the other players instead of being hidden.
  + \*\*UPDATE:\*\* As of 12/12/2024, this issue has been resolved.
* The app has transitioned to a subscription-based model, restricting free-to-play players from accessing several crucial features.
* Lastly, the game lacks a Debt Settlement feature, which means it doesn’t support calculating or tracking poker debts outside of the app.



**Zynga Poker:**

**Pros:**

* Well-known app with a substantial player base.
* Generally bug-free.
* Offers a wide variety of online poker games.

**Cons:**

* Sometimes, the app can be extremely laggy.
* The user interface is overly saturated and has very high contrast, which can be eye-straining after prolonged use.
* There’s no option to play private games. You must play on an existing public server, which may force you to play with strangers even if you only want to play with your friends.
* The app is heavily advertised and is very pushy trying to make the player spend real money on the app to get in game chips that are required for play.

Although poker is a gambling game, some argue that it’s more of a skill game than luck. In Zynga Poker, aside from the poker itself, there are many elements of a “Casino” type that emphasize slots and leveling up your account by purchasing in-game currency. This can lead to addiction to gambling.

* Lastly, the game lacks a Debt Settlement feature, which means it doesn’t support calculating or tracking poker debts outside of the app.

## Challenges in the current market

As previously discussed in the background segment, the primary challenge in the current market of poker apps, is the absence of a crucial tool / app that is essential for individuals like us who occasionally enjoy a home game of poker, A app called **Poker Mate**.

With all the self-promotion aside, there isn’t a single app in the current market that offers all the features our app provides.

* Play with friends on your own terms.
* Settle games and resolve problems.
* Track your games using a database that provides valuable insights into your play.

While there are several session tracking apps available, they often require excessive fiddling and data entry, which we strive to eliminate.

We believe there’s a missing puzzle piece that can fill the current market gap in the world of poker apps.

# What We Intend to Renew or Enhance

We are committed to completing the missing piece in the poker app marketplace, an application that will fulfill the fundamentals of an enthusiast poker player.

* Play with friends in a controlled and friendly environment where you control what will happen.
* Track game and subsequential data
* Resolve problematic games that happened outside the app.
* Help settle debts (in / out the app)

1. **שם הפרויקט**
2. **רקע**

**2.1. תיאור ורקע כללי**

**2.2. מטרות המערכת**

1. **המצב בשוק והבעיות**
   1. **סקירת מצב קיים בשוק**
   2. **בעיות במצב הקיים**
2. **מה הפרויקט אמור לחדש או לשפר**
3. **דרישות מערכת ופונקציונאליות**
   1. **דרישות מערכת**
   2. **דרישות פונקציונאליות**
4. **בעיות צפויות במהלך הפיתוח ופתרונות** 
   1. **תיאור הבעיות**
   2. **פתרונות אפשריים**
5. **פתרון טכנולוגי נבחר**
   1. **טופולוגית הפתרון**
   2. **טכנולוגיות בשימוש**
   3. **שפות הפיתוח.**
   4. **תיאור הארכיטקטורה הנבחרת**

**7.5. חלוקה לתכניות ומודולים**

**7.6. סביבת השרת**

**7.7. ממשק המשתמש/לקוח – GUI**

**7.8. ממשקים למערכות אחרות/API**

**7.9. שימוש בחבילות תוכנה**

1. **שימוש במבני נתונים וארגון קבצים** 
   1. **מבני הנתונים**
   2. **שיטת האחסון**
   3. **מנגנוני התאוששות**
2. **תרשימי מערכת מרכזיים**

**9.1. Use Case**

**9.2. Sequence Diagram**

**9.3Data Flow**

1. **תיאור המרכיב האלגוריתמי - חישובי**

**10.1 הבעיה שבא לפתור**

**10.2 איסוף מידע וניתוחים סטטיסטיים**

1. **אבטחת מידע**
2. **משאבים הנדרשים לפרויקט**

**12.1 מספר שעות המוקדש לפרויקט, חלוקת עבודה בין חברי הצוות**

**12.2 ציוד נדרש**

**12.3 תוכנות נדרשות**

**12.4 ידע חדש שנדרש ללמוד לצורך ביצוע הפרויקט**

**12.5 ספרות ומקורות מידע**

1. **תכנית עבודה ושלבים למימוש הפרויקט**
2. **בדיקות**

**14.1 בדיקות תהליכיות**

**14.2 בדיקות יחידה**

1. **בקרת גרסאות**
2. **קטעי קוד**