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

[1. Project Name 4](#_Toc188169524)

[2. Background 4](#_Toc188169525)

[2.1 General description 4](#_Toc188169526)

[2.2 System goals and objectives 7](#_Toc188169527)

[3. Current Market State and Challenges 7](#_Toc188169528)

[3.1 Current market state 7](#_Toc188169529)

[3.2 Challenges in the current market 9](#_Toc188169530)

[4. What We Intend to Renew or Enhance 11](#_Toc188169531)

[5. System and functional requirements 12](#_Toc188169532)

[5.1 System requirements 12](#_Toc188169533)

[5.2 Functional requirements 12](#_Toc188169534)

[6. Expected problems during development and solutions 13](#_Toc188169535)

[6.1 Problems description 13](#_Toc188169536)

[6.2 Possible solutions 13](#_Toc188169537)

[7. Selected Technology Solution 14](#_Toc188169538)

[7.5 Division into programs and modules 14](#_Toc188169539)

[10. Description of the algorithmic - computational component 15](#_Toc188169540)

[10.1 The problem we aim to solve 15](#_Toc188169541)

[The problem: 15](#_Toc188169542)

[The solution: 15](#_Toc188169543)

[10.2 Data collection and statistical analysis 15](#_Toc188169544)

[13. Work Plan and Stages for Project Implementation 16](#_Toc188169545)

# 1. Project Name

Poker Mate – Play poker, Settle debts, Track your progress.

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 help you when you’re craving a game of poker or need help settling some in game debts.

**Our slogan:**

Created by poker players for poker players!

You’re in charge, play the way you play at home.

# 2. Background

## 2.1 General description

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

Some facts about Poker before we begin :)

* Poker is the most popular card game in the world
* There is an annual increase of 34% in new poker players worldwide.
* The online poker player count is massive, with a **100 million** player base.
* The global online poker market was valued at ~$90 billion in 2023 and is expected to be worth **$170 billion** by 2030.

[Global Gambling Statistics & Trends 2024](https://gamblingindustrynews.com/global-gambling-statistics/)

[https://playtoday.co/blog/stats/online–poker–statistics/](https://playtoday.co/blog/stats/online-poker-statistics/)

Hey there! I’m Giora, a passionate poker enthusiast. When the pandemic (COVID–19) struck the world, many of us, including myself, turned to poker as a form of entertainment. The lockdown resulted in a significant 43% growth in online poker, which also affected physical poker play. As my friends and I continued to play more frequently, I began to notice a recurring issue.

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.

## 2.2 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. **Solve problems** – A utility tool inside of Poker Mate that allows the user to use 2 different functionalities.
   1. **PokerSolverPro** – Can be accessed independently or after ending a game in the “Play” functionality. It’s used to settle debts from a game of poker and handle any issues that may arise (e.g., total buy–in ≠ total buy–out).
   2. **Chip Chop** – Used for splitting X pots to Y players with a care for remainders based on the minimum game chip.
3. **Track your progress** – Games you played (online & games you’ve inputted your name (after confirmation) PokerSolverPro (Solve Problem) function.

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

# 3. Current Market State and Challenges

## 3.1 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.
* 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 levelling 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.

## 3.2 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.

# 4. 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)

Currently, there isn’t a single app that offers all these features. There are apps that are solely focused on tracking player statistics, while others allow you to play with friends, like “Chips of Fury,” which enables players to play together and control game settings. However, there are missing functionalities, and the app requires one player to have a subscription for a complete experience.

*Shortened version.*

We are dedicated to completing the missing puzzle piece in the poker app marketplace with an application that will cater to the fundamental needs of an enthusiast poker player.

Currently, there is not a single app in the Poker/Casino apps market that offers all the functionalities of Poker Mate (our app).

Refer to [System goals and objectives](#_2.2_System_goals).

# 5. System and functional requirements

## 5.1 System requirements

1. **Firebase DB** – A DB to store user data and statistics for the application.
2. **Local Server** – Having a local server for players to be able to play together on
3. **iOS / Android Phone** – to run the application.
4. **Unique user ID generation** – so the app won’t crash when a lot of people are trying to sign up.
5. **Ability to save in game name on device** – So that a user will have its username saved on device and won’t require him to log in.
6. **App responsiveness** – App must be responsive and not lag.

## Functional requirements

**Players**

1. **Play** – Having all the basic functionalities of poker (Check, Bet, Raise, Fold, Straddle), playing with correct poker logic and intuitively doing actions.
2. **Chat** – Having a chat where players can talk with one another via text.
3. **Statistics** – Having the option to see basic stats in game while playing (Buy-in, P/L of player, Number of hands played)
4. **Settings** – Having a settings page when **creating a game** where you can tweak different settings such as (blinds, poker type, pot limit / no limit / dynamic, Time limit, Amount of time banks) and more.
5. **Saving a game setting** – Previously made games can be saved in order not to make a new one each time.
6. **Personal user DB** – A DB that will track the user’s games and show different data about the user.

**Administrator**

1. **Advanced features** – Has the functionalities of Players + more.
2. **Handle players** – Admin can oversee players and approve/remove if needed.
3. **Ability to transfer Admin** – Each game will have an admin / or 2 and each admin should be able to transfer his role to a regular player in the game.
4. **Change settings** – Can change game settings while in game.
5. **Pause game** – Can pause game.

# 6. Expected problems during development and solutions

Our biggest challenge in terms of the project is learning a new environment called Unity, which we haven’t covered in our courses. We’ve spoken to friends and lecturers, and they all emphasized the importance of learning and using this tool for our “play” mechanic, as it would make all the animation and general game mechanics simpler as a lot of them are built in into the unity engine.

It’s a significant challenge because we have no prior knowledge of Unity and need to start from scratch to build a relatively complex project.

Another challenge is to integrate a DB into our system and find the logical way for it to work with users, do we want the data on device, do we want the data on our (creators) DB?

Another challenge is to implement cross-play between iOS and Android. We learn as of now how to code in Android using Android Studio, but we will have to learn how to implement our app to iOS platforms as well and to ensure that both will be able to play together.

## 6.1 Problems description

1. Learning and adapting to the new environment (Unity).
2. Understanding a finding out the logical way to implement a DB in the project.
3. Implementing cross play between iOS and Android.

## 6.2 Possible solutions

1. Learning the new environment required for the development of the project.

* Watching YouTube videos regarding the engine.
* Reading unity documentation.
* Designing and planning the layout and functionalities of the app prior to programming them.

1. Implementing the DB requires planning what data we want to keep, and we want to save it (on device / DB).
2. Learning how to build the game for both Android and iOS using Unity and use a networking solution to manage multiplayer connections across platforms.

# 7. Selected Technology Solution

## 7.3 Development Languages Chosen

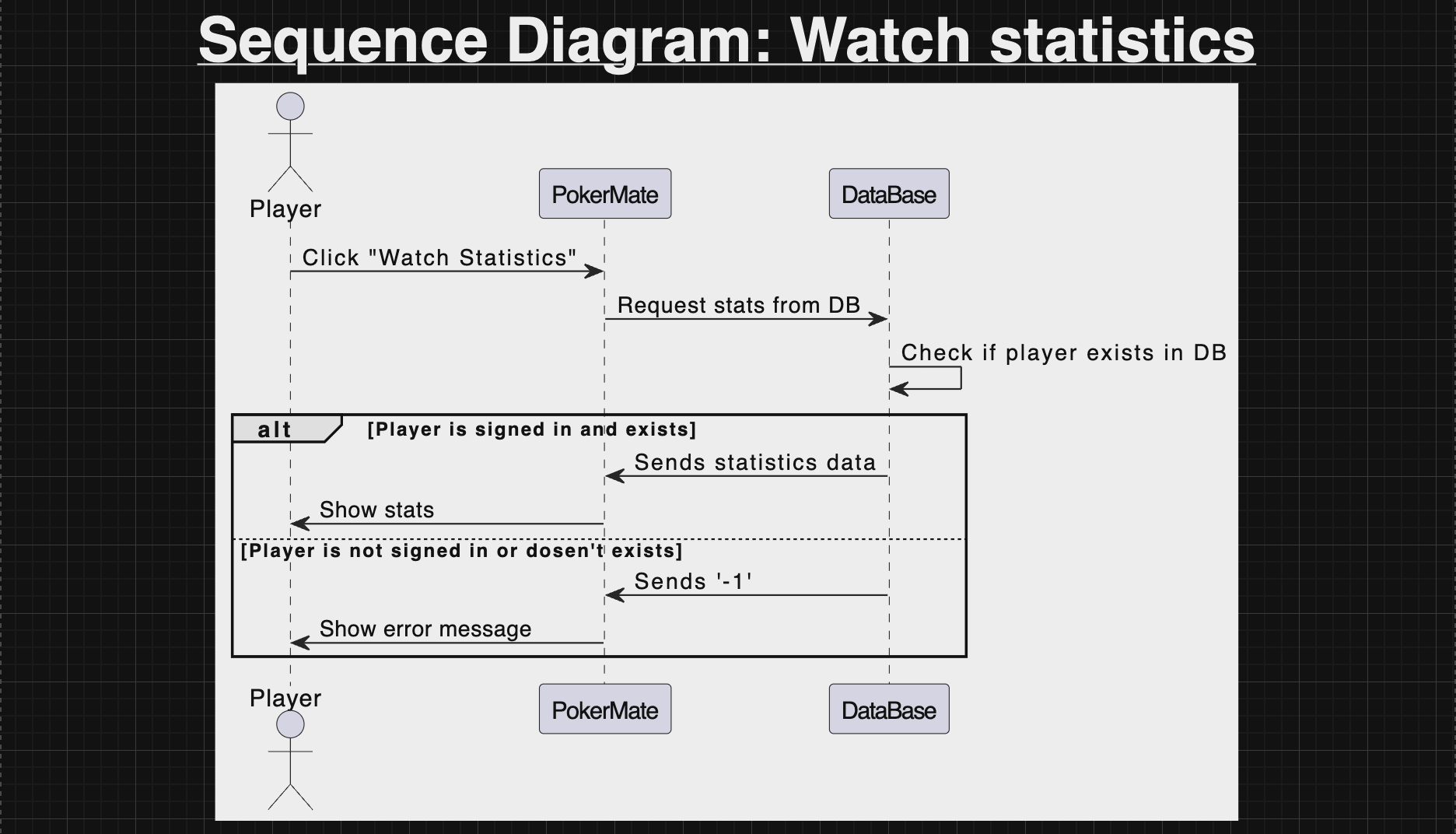
* C# – For backend logic
* JavaScript – React Native – For UI UX

## 7.5 Division into programs and modules

# 9. Central System Diagrams

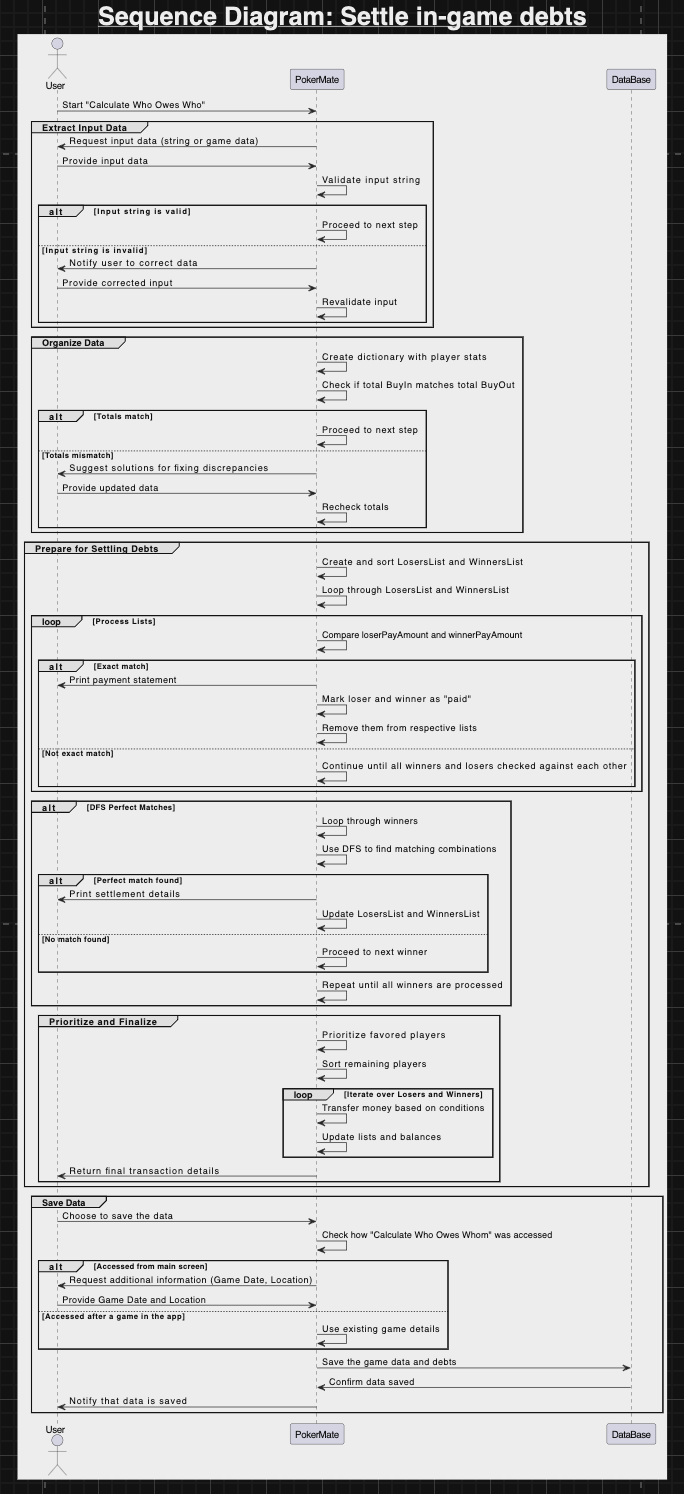
## 9.1Usecase

## 9.2 Sequence



A diagram of a game

Description automatically generated



## 9.3 Data Flow

# 10. Description of the algorithmic - computational component

## 10.1 The problem we aim to solve

### The problem:

The main problem we see in the current poker app world is that there isn’t a single app that gives an enthusiast poker player the “all in one” app that can do all the following things:

1. Play / Manage a poker game in a controlled and personal environment.
2. Track your games and subsequential data.
3. The ability to provide solutions for games that don’t add up based on logical and mathematical reasoning.
4. Ability to resolve in-game debts that occurred in app / outside of it using a multi part algorithm.

What sets our app apart is that there might be other apps that address one or two of the features we provide, but none can offer a comprehensive solution in a single app.

### The solution:

As previously discussed, our solution that we offer to the problem above is to develop an all-in-one application that users can utilize to create games, engage in gameplay with friends, solve problems in games, and track their games and progress.

## 

Our application prioritizes user privacy and is designed to collect only essential data for operational purposes.

We focus on gathering key metrics such as the number of users who have registered, quantity of games created, game types played, and amount of bugs reported.

We do not collect any additional user data beyond what is necessary to enhance user experience and fulfil the core functionalities of our application.

Our goal is to assist users in meeting their personal poker needs without infringing on their privacy, ensuring data collection is minimal and strictly aligned with service improvements.

* Number of games created – How many games users made in our app.
* Game types played – So we can analyze the data to see most popular game type (Texas Hold’em, Omaha 4, Omaha 5, Bomb pot 4, Bomb pot 5, Pineapple...)
* Bugs reported – How many bugs have been reported by users.

# 12. Resources needed for the project

## 12.1 Number of hours dedicated to the project, division of work between team members

**Giora** – Primarily working on the backend side of things, improving algorithms from edge cases.

Time dedicated – 4 hours a week as of 27/01/2025.

**Nati** – Focuses on UX UI and general design of the application.

Time dedicated – 4 hours a week as of 27/01/2025.

## 12.2 Equipment

* Computer
* Computer essentials (mouse, keyboard…)
* Android Phone for testing

## 12.3 Software

* Visual Studio Code.
* VSC extension – TODO Tree for code organization and tracking of bugs.
* VSC extension – C# kit.
* .NET Framework v.9.

## 12.4 New knowledge that is required to be learned in order to carry out the project

As previously discussed, The major knowledge required is the need to learn unity from scratch and implement the app on to it.

## 12.5 Literature and sources of information

TO BE DETERMINED

# 13. Work Plan and Stages for Project Implementation

* **Characterization** – will continue until 27/06/2024
* **System design and coding** – will continue until 31/08/2025
* **Testing and corrections** – during 09/2025
* **Project submission** – until 01/10/2025
* **Project defense** – during 10/2025

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**