UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

GDHelper: Tools to Help the Diagnosis of Gaming Disorder

Daniela da Costa Jorge

Mestrado em Informática

Trabalho de Projeto orientado por:
Profª. Doutora Ana Paula Pereira Afonso
Prof. Doutor Manuel João Caneira Monteiro da Fonseca

2024

To gamers and professionals, working together for a balanced digital world.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my advisors, Dra. Ana Paula Afonso and Dr. Manuel Fonseca, for their crucial guidance, mentorship, and support throughout the course of this project. Their expertise and encouragement have been instrumental in shaping this research and my academic journey.

I extend my sincere gratitude to Joana Cardoso, whose psychological insights and reliable support have significantly contributed to the direction and progress of this project.

To my family, I am deeply grateful for your unwavering patience, loyalty, and support throughout this journey. Your understanding and encouragement have been a constant source of strength, from engaging in discussions about my academic journey to celebrating each milestone. Your presence has been a motivation throughout this journey.

I would also like to thank my friends for their companionship and encouragement. Whether indulging in gaming sessions of Counter-Strike or supporting me through the various stages of this project, your presence has provided a source of comfort and inspiration during this journey.

Finally, I would like to express my gratitude to everyone who has contributed to this project, whether directly or indirectly, through their time, feedback, or encouragement. Your efforts have made a significant impact on the outcome of this research, and I am deeply grateful for your support.

Resumo

O fenómeno dos esports (desportos eletrónicos) tem vindo a crescer e, com este, também o interesse por vídeo jogos online, por parte de jogadores amadores, profissionais e espetadores. Entre esses vídeo jogos, o Counter-Strike 2 (CS2) destaca-se como um dos mais populares, contando com uma base de jogadores ativa e interessada. Entretanto, junto com esse crescimento, surgiram preocupações com os impactos que o jogo excessivo pode causar na vida de algumas pessoas. Uma das principais preocupações no campo da saúde mental é a Perturbação de Jogos da Internet, conhecida como Internet Gaming Disorder (IGD), que foi formalmente reconhecida na quinta edição do Manual Diagnóstico e Estatístico de Transtornos Mentais (DSM-5, Diagnostic and Statistical Manual of Mental Disorders) e na 11ª Revisão da Classificação Internacional de Doenças (ICD-11 International Classification of Diseases).

A Perturbação de Jogos da Internet foi incluída no DSM-5 em 2013, na secção de condições para estudo futuro, com o intuito de motivar a discussão e investigação da mesma. Sendo a sua origem e a manutenção ainda alvo de debate, os profissionais de saúde mental ainda encontram dificuldades no processo de diagnóstico e intervenção desta perturbação mental. Segundo o DSM-5, pode ser diagnosticada recorrendo a uma lista de nove critérios de diagnóstico, sendo necessário a ocorrência de pelo menos cinco num período de, no mínimo, 12 meses. Alguns exemplos destes critérios, são: sintomas de abstinência, preocupação constante com os videojogos, perda de interesse noutras atividades, mentir/enganar pessoas ou profissionais de saúde sobre o tempo despendido a jogar, entre outros. Atualmente, o diagnóstico desta perturbação realiza-se a partir de questionários de auto-relato e/ou avaliações clínicas. Porém, devido à inadequação de alguns critérios de diagnóstico a identificação desta perturbação revela-se difícil, sendo necessária a procura de alternativas a estes métodos de avaliação tradicionais. O diagnóstico do IGD apresenta uma série de desafios. Embora a pertubação tenha sido reconhecida oficialmente, ainda há uma considerável falta de consenso sobre os critérios diagnósticos específicos que devem ser utilizados. O DSM-5 e a ICD-11 fornecem diretrizes gerais e amplas e podem ser interpretadas de maneiras diferentes por diferentes profissionais de saúde. Além disso, a maioria das ferramentas disponíveis atualmente para o diagnóstico de IGD depende fortemente de questionários de autorrelato, que são instrumentos subjetivos e que podem ser influenciados pela percepção do indivíduo sobre o seu próprio comportamento. Muitas vezes, os pacientes minimizam ou negam a gravidade do seu problema, o que pode dificultar a obtenção de uma visão clara e objetiva da situação para os profissionais de saúde mental. A disponibilização de dados recolhidos durante o jogo (dados

telemétricos) por parte das empresas criadoras dos jogos (p.e. Riot, Valve entre outas) permite que sejam realizadas análises para vários fins. Embora a análise dos dados de telemetria se centre principalmente no comportamento do jogador/equipa e no desempenho do jogo, também podem ser utilizados para analisar o jogador de outras perspetivas e ajudar os profissionais de saúde a detectar comportamentos problemáticos do jogador. No âmbito de uma tese de mestrado anterior foram identificadas as métricas (p.e. tempo de jogo e número de sessões por ano, mês, dia entre outras), as variações de emoções de jogadores de Counter-Strike, e desenvolvidas visualizações que permitem ajudar os profissionais de saúde na análise de determinados critérios de diagnóstico do IGD. Para enfrentar esses desafios e fornecer uma solução mais objetiva e confiável para o diagnóstico do IGD, este projeto apresenta o GDHelper, uma aplicação web interativa criada para auxiliar profissionais de saúde mental no diagnóstico e tratamento dessa perturbação. O GDHelper é uma ferramenta inovadora que se destaca por integrar dados de telemetria do jogo Counter-Strike 2 com avaliações psicológicas tradicionais, proporcionando aos profissionais de saúde uma visão mais detalhada e precisa sobre o comportamento de jogo dos seus pacientes.

A telemetria é a recolha automática de dados durante as sessões de jogo, e no caso do GDHelper, esses dados incluem métricas essenciais como a duração das sessões de jogo, a frequência com que o indivíduo joga ao longo de diferentes períodos (diário, semanal, mensal e anual), além de outras informações pertinentes que podem ser úteis para uma análise aprofundada do comportamento do jogador. Com essa integração, o GDHelper oferece aos profissionais de saúde a capacidade de monitorizar o comportamento de jogo dos pacientes de forma contínua, algo que os métodos tradicionais, baseados apenas em autorrelatos, não são capazes de proporcionar.

O principal objetivo do GDHelper é fornecer uma ferramenta dinâmica que auxilie a precisão e a confiabilidade do diagnóstico do IGD, minimizando a subjetividade associada aos questionários de autorrelato. A interface da aplicação foi desenvolvida com o intuito de ser amigável e de fácil utilização, permitindo que os profissionais de saúde possam aceder e analisar os dados de telemetria. Através do GDHelper, os profissionais têm a possibilidade de visualizar dados atualizados sobre o comportamento de jogo dos pacientes, o que facilita o acompanhamento da evolução do tratamento e permite intervenções mais rápidas e eficazes.

A aplicação permite que os profissionais acedam às métricas de diferentes jogadores de forma individual e analisem de acordo com os padrões existentes ou conhecidos. Isso pode ser particularmente útil para contextualizar o comportamento de um paciente dentro de um panorama maior, ajudando o profissional a identificar se o comportamento observado não é normal ou se está dentro de padrões aceitáveis de uso.

O desenvolvimento do GDHelper foi realizado em colaboração com especialistas em psicologia e comportamento de jogo, garantindo que a aplicação está de acordo com as necessidades reais dos profissionais que trabalham com pacientes que apresentam sintomas de IGD. A psicóloga clínica Joana Cardoso, atualmente doutoranda em Psicologia na Universidade da MAIA, e investigadora no Centro de Psicologia da Universidade do Porto, desempenhou um papel fundamental na validação do projeto, fornecendo conhecimentos valiosos sobre as métricas mais relevantes para o diagnóstico e ajudando a garantir que a interface era intuitiva e de fácil uso. Além disso, o feedback de outros profissionais de saúde que participaram da fase de testes foi extremamente positivo, destacando a eficácia da aplicação em fornecer informações úteis e de fácil interpretação.

Outro recurso importante do GDHelper é o suporte para a atualização regular dos dados de telemetria. A aplicação foi projetada para ser flexível e permitir que os dados sejam atualizados conforme necessário. O profissional de saúde pode optar por atualizar as informações de um jogador específico ou de todos os jogadores existentes na aplicação. Isso garante que os dados mais recentes estejam sempre disponíveis para consulta, o que é fundamental para análises clínicas informadas e para o acompanhamento da evolução do paciente ao longo do tempo.

Além de auxiliar no diagnóstico, o GDHelper também oferece suporte no desenvolvimento de planos de tratamento mais eficazes. Com uma visão clara e detalhada dos hábitos de jogo de cada paciente, os profissionais de saúde podem adaptar as suas abordagens com base nos dados recolhidos. Isso permitirá criar estratégias de tratamento que sejam realmente eficazes e voltadas para as necessidades específicas de cada indivíduo, aumentando as chances de sucesso no processo terapêutico.

Do ponto de vista da investigação, este projeto contribui de forma significativa para o campo de estudo do IGD. Ao integrar dados de telemetria com os métodos tradicionais de avaliação psicológica, o GDHelper representa uma abordagem nova e inovadora para o diagnóstico e tratamento dessa perturbação. A combinação de dados objetivos e subjetivos permite uma compreensão mais profunda do comportamento de jogo dos pacientes, ajudando a distinguir entre o uso saudável e recreativo dos jogos eletrónicos e o uso problemático que caracteriza o IGD.

Os resultados obtidos com o GDHelper sugerem que a aplicação tem o potencial de ajudar os profissionais de saúde no processo de diagnóstico assim como no tratamento do IGD, oferecendo uma visão mais objetiva e atual dos hábitos de jogo dos pacientes. Esse avanço pode ter implicações importantes para a saúde mental, uma vez que permite que os profissionais de saúde tomem decisões mais bem informadas e com base em dados concretos, permitindo melhorar os desfechos do tratamento e a qualidade de vida dos indivíduos afetados pelo IGD. Foi efectuada uma avaliação de usabilidade com três psicólogos, que revelou uma elevada usabilidade para tarefas básicas e forneceu indicações sobre áreas a melhorar.

Em resumo, a popularidade e crescimento dos desportos eletrónicos associado ao facto de surgirem alguns comportamentos problemáticos entre os jogadores justifica o desenvolvimento de ferramentas eficazes para o diagnóstico e tratamento do IGD. O GDHelper representa um avanço nesse sentido, fornecendo uma solução inovadora que combina dados de telemetria com avaliações psicológicas, permitindo uma análise mais completa e precisa do comportamento de jogo. Ao fornecer dados objetivos e atualizados, o GDHelper facilita intervenções mais eficazes, contribuindo para melhorar a saúde mental e o bem-estar dos indivíduos afetados pela perturbaçãoo de jogo na internet.

Palavras-chave: Perturbação de Jogos da Internet, Telemetria de jogos, Aplicações Web, Counter-Strike 2, Esports

Abstract

The rise of esports has led to an increased focus on Internet Gaming Disorder (IGD), a condition recognized in the DSM-5 and ICD-11. Despite its growing prevalence, diagnosing IGD remains challenging due to the subjective nature of self-report questionnaires and the lack of consensus on diagnostic criteria. This thesis presents GDHelper, an interactive application designed to aid health professionals in diagnosing and treating IGD by integrating telemetry data from Counter-Strike 2 with psychological assessments.

The primary goal of GDHelper is to provide a dynamic tool that enhances the accuracy and reliability of IGD diagnosis. The application builds on previous research by incorporating key metrics such as gameplay duration and session frequency into a user-friendly interface. The data automatically collected from online game platforms allows us to obtain objective in-game player behaviour, a historical view of gaming behaviour, and analysis of the evolution of the player behaviour and compare with behaviour patterns. This approach reduces the reliance on subjective self-reports and offers a more objective view of a player's gaming habits.

The development of GDHelper follows a user-centered methodology involving collaboration with experts in clinical psychology and gaming behavior, ensuring the tool's relevance and effectiveness in clinical settings. The application supports regular data updates, facilitating informed discussions during clinical follow-ups. By providing a comprehensive analysis of gaming behavior, GDHelper aims to help more accurate diagnoses and effective treatment plans for individuals with IGD. A usability evaluation was conducted with three psychologists, revealing high usability for basic tasks and providing insights into areas for improvement.

In conclusion, this thesis contributes to the ongoing research on IGD by offering a novel approach to its assessment. The integration of real-time game data with traditional assessment methods provides a more holistic understanding of gaming behaviors. The findings highlight the potential of GDHelper to improve the diagnosis and treatment of IGD, ultimately contributing to better mental health outcomes for affected individuals.

Keywords: Internet Gaming Disorder, Telemetry Data, Web Applications, Counter-Strike 2, Esports

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This section provides an overview of the context and motivation behind this project. It discusses the growing concern of Internet Gaming Disorder (IGD) and the challenges associated with its diagnosis and treatment. It also outlines the project's main goals and objectives, which aim to develop an interactive application that integrates telemetry data and psychological assessments to support the diagnosis and treatment of IGD, as well as the contributions of the project and how this document is structured.

1.1 Motivation

The rise of esports (electronic sports) has captivated a growing audience of amateur and professional players, as well as spectators, around the globe [4]. Coinciding with this upsurge, the term Internet Gaming Disorder (IGD) surfaced in the Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition (DSM-5), in 2013 [5]. It was cataloged under conditions meriting further study, to initiate further academic dialogue and investigation. Given the etiology and maintenance of IGD are still a matter of debate and contested among experts [6], mental health professionals still find it challenging to diagnose and intervene in this mental disorder. It's worth noting that ”internet gaming” in this context exclusively denotes the activity of playing video games.

Video games are a widespread and evolving form of entertainment, impacting billions of people globally [2, 7]. While they can positively influence physical, mental, and social well-being [8, 9], some problematic gaming behaviors have emerged in recent years. The excessive use of video games can escalate to pathological and even addictive levels, as highlighted by research on gaming disorder [10, 6, 11]. The worldwide prevalence of IGD is around 3%, with variations from 0.8% to 11.8% in Europe [12, 13]. The recent international policy decisions, such as the inclusion of IGD in the DSM-5 and International Classification of Disease (ICD-11), reflect efforts to address scientific uncertainties regarding the potential impacts of video games on players' well-being [5, 14].

According to the DSM-5, IGD can be diagnosed using a list of nine diagnostic criteria, requiring at least 5 symptoms within a 12-month timespan to posit a diagnosis [5]. Some of these criteria can be experiencing withdrawal symptoms, constant preoccupation with video games, loss of interest in other activities or hobbies, and lying/deceiving people or health professionals about the time spent playing, among others. Typically, diagnoses are based on self-report questionnaires and clinical assessments [15]. However, the reliance on self-reported data introduces subjective biases, calling for more objective methods [15].

Many of these tools interpret the criteria freely and one of the challenges in the field is the lack of consensus on the criteria used for screening and diagnosing this condition [16]. The diagnosis can be challenging due to the inadequacy and subjective nature of some of the diagnosis criteria and the lack of consensus on assessing and measuring gaming behavior [17]. The approaches based on the self-reports are inaccurate as subjective responses of the player hardly capture the real player behaviors, as the player may not remember or not mention some relevant information (e.g., year playtime, last week playtime, emotional states).

The limitations of traditional assessment techniques have encouraged the pursuit of more precise and dependable methods for diagnosing Internet Gaming Disorder. A previous master's thesis [3] has explored and identified valuable metrics such as gameplay duration, and the number of sessions per year, month, and day experienced by players of Counter-Strike [18].

Visualizations based on those metrics, obtained from telemetric data, were developed to help health professionals analyze certain diagnostic criteria for IGD. Game telemetry refers to data collected in real-time using sensors and other tools within video games [19]. This data records everything from general game information to specific player actions and events, offering valuable insights into how players interact with games [19, 20]. However, the current visualizations have some drawbacks. First, they are static, which means they are based on fixed game data and do not reflect the changes in gaming behavior over time. Second, they display data for only a single player, not allowing for comparisons with other players or the general population. These limitations highlight the need for an interactive tool that can support the diagnosis of IGD, enabling health professionals to monitor telemetric data over time and complement questionnaires and monthly clinical follow-ups with players. The integration of digital data collection, notably telemetry data that records in-game behaviors such as play durations and session frequencies, has proven a viable path for objectively studying IGD.

The main goal of this project is to develop an interactive application that integrates the previously studied metrics and visualizations, accesses regular game data (allowing health professionals to update their patient's data whenever necessary), and provides a tool to help health professionals analyze the player's gaming behavior before each consultation and include previous statistics.

Joana Cardoso, Clinical Psychologist and Ph.D. candidate at Universidade de Maia, plays an important role in this project. As a member of the Portuguese Federation of Esports (FPEsports) [21] and her extensive experience in psychology and gaming behavior, her expertise has been instrumental in shaping this project. She contributed to the literature on Internet Gaming Disorder, helped develop key metrics, and critically analyzed the data visualizations, which greatly enhanced the findings of this project. Additionally, her input was crucial in validating the GDHelper tool and recruiting users for the study. Her contribution shows the benefits of combining different fields of study in IGD research.

1.2 Goals

The primary goal of this project is to develop an interactive application, named GDHelper, that helps the diagnosis and treatment of Internet Gaming Disorder (IGD) by providing health professionals with a dynamic and comprehensive tool for analyzing gaming behavior specifically related to Counter-Strike 2. This application builds upon the foundational work of a previous master's thesis [3], which identified key metrics and developed visualizations related to gameplay duration and session frequency. However, the existing solutions are static and limited in scope. To address these limitations, this project aims to achieve the following goals:

1. **Integrate existing metrics and visualizations into an interactive platform:**
    - Use the previously defined metrics, such as gameplay duration, number of sessions per year, month, and day, specifically for Counter-Strike 2, and incorporate them into a dynamic, user-friendly interface.
    - Ensure the application supports regular data updates, allowing for the continuous monitoring of gaming behavior.
2. **Enhance data accessibility:**
    - Develop features that allow telemetry data collected from online game platforms to provide objective in-game player behavior, a historical view of gaming behavior, and precise insights into gamer behavior to discover or corroborate patterns of behavior.
3. **Provide regular access to updated game data:**
    - Implement mechanisms to regularly retrieve and integrate new game data, ensuring that the analysis remains relevant and reflective of the player's current gaming behavior.
4. **Support health professionals in clinical settings:**
    - Design the application to be a practical tool for health professionals, enabling them to review and analyze a player's gaming behavior before consultations. This will assist in informed discussions during clinical follow-ups and support more accurate diagnoses.
5. **Facilitate objective assessment of IGD:**
    - Develop the application to complement traditional self-report questionnaires by offering objective, telemetry-based insights into the player's gaming behavior, thereby reducing the dependency on subjective data.
6. **Collaborate with experts in psychology and gaming behavior:**

- Collaborate with experts like Joana Cardoso, Clinical Psychologist and Ph.D. candidate, to ensure the application meets the needs of healthcare professionals and is based on the latest research on IGD.
- Continuously validate and refine the metrics and visualizations through collaboration with experts, ensuring their relevance and effectiveness in clinical practice.

By achieving these goals, the project will provide a valuable tool for the diagnosis and treatment of IGD, helping health professionals gain a more accurate and comprehensive understanding of their patients' gaming behaviors.

1.3 Contributions

In this section, we outline the key contributions of our project:

- **Literature review and analysis:** Conducted a comprehensive review of existing literature on Internet Gaming Disorder and its diagnostic challenges. Identified the strengths and limitations of previous research, particularly focusing on the integration of telemetry data and psychological assessments. This review helped to position our work as a complement to and enhancement of existing methodologies.
- **Identification of gaps in current screening protocols:** Analyzed current diagnostic criteria and tools used for diagnosis, highlighting their limitations, particularly the reliance on self-reported data. Proposed solutions that leverage objective data sources, such as gameplay telemetry, to address these gaps and improve the accuracy and reliability of IGD assessments.
- **Integration of Counter-Strike 2 telemetry data:** Developed a method to access, retrieve, and use real-time telemetry data from Counter-Strike 2. This data serves as the foundation for defining relevant metrics that provide insights into gaming behavior, such as playtime, session frequency, and gaming intensity. This integration enables the continuous monitoring of player behavior, allowing for a more dynamic analysis over time.
- **Design and development of GDHelper tool and interactive visualizations:** Developed the GDHelper tool, which integrates a set of interactive visualizations with additional functionalities. These visualizations, building upon static ones from previous research, are designed to help health professionals quickly and effectively analyze a player's gaming behavior.
- **Collaboration with experts in psychology and gaming:** Worked closely with experts like Joana Cardoso to ensure the application's metrics and visualizations accurately reflect real-world clinical scenarios and meet the needs of healthcare professionals. This collaboration was crucial for refining the tool to ensure it is practical and valuable in a clinical setting.

- **Enhancement of IGD diagnosis through objective data:** Provided a different approach to IGD diagnosis by integrating telemetry data from Counter-Strike 2 with traditional assessment methods. This approach reduces the reliance on subjective self-reporting, offering a more objective view of a player's gaming behavior. The developed tool can be used alongside questionnaires and clinical evaluations to offer a more comprehensive understanding of the player's condition.

- **Contribution to the ongoing IGD research and practice:** By developing a tool that integrates regular game data with psychological assessments, this project contributes to the broader field of IGD research. This tool provides a new way to objectively evaluate gaming habits, with potential to be applied to other games and situations in the future.

This project not only builds on previous research but also introduces significant innovations in how gaming behavior is analyzed and understood, offering a more robust tool for the diagnosis and treatment of IGD.

1.4 Structure of the document

This document is organized as follows:

- **Chapter 2: Background -** Provides essential context for understanding Internet Gaming Disorder (IGD), including its definition, diagnosis criteria, risk factors, comorbidities, and the relevance of Counter-Strike 2 in this work.

- **Chapter 3: Related Work -** Reviews existing literature and previous research related to game telemetry analysis, screening tools, and user interface design for health professionals.

- **Chapter 4: GDHelper Analysis and Design -** Defines the proposed solution, outlines the software development methodology, and details the data workflow and system architecture.

- **Chapter 5: GDHelper Implementation -** Describes the development environment, application API, user interface, and considerations for responsiveness and accessibility.

- **Chapter 6: GDHelper Validation -** Discusses the methodology and results of the validation process, including user feedback and task-specific insights.

- **Chapter 7: Conclusion and Future Work -** Summarizes the findings, contributions, and potential future directions for the project.

Chapter 2

Background

This chapter provides essential context for understanding the problem of Internet Gaming Disorder (IGD) and its implications for mental health. It aims to clarify the definition, diagnosis, and criteria of IGD, as well as the risk factors, comorbidities, and consequences associated with it. It also introduces the game Counter-Strike 2, which is the focus of this study, and the web development framework Node.js, which is the tool that will be used to create the application. Each section explores these topics in more detail, offering relevant background information and a literature review.

2.1 Internet Gaming Disorder

According to the DSM-5, the diagnosis of Internet Gaming Disorder can be posited using a list of nine diagnostic criteria requiring the occurrence of at least 5 symptoms within 12 months [5]. The nine diagnostic criteria are:

1. Preoccupation with online/offline gaming.
2. Experience withdrawal symptoms when gaming is taken away.
3. The need to spend more time engaged in games.
4. Unsuccessful attempts to reduce time in games.
5. Loss of interest in hobbies and entertainment except for gaming.
6. Continuation use of games despite knowledge of psychosocial problems.
7. Deceiving family members, health professionals, or others regarding the amount of gaming.
8. Use of games to escape reality or relieve negative emotions.
9. Jeopardizing or losing a significant relationship or job because of gaming.

While the DSM-5 from the APA lists Internet Gaming Disorder (IGD) as a condition meriting further research [5], the World Health Organization (WHO) included the term Gaming Disorder in their 11th revision of the International Classification of Diseases (ICD-11) [14]. Gaming Disorder is characterized when a player presents a pattern of persistent or recurrent gaming behavior, which can be online or offline [14]. Gaming Disorder in ICD-11 is characterized by:

1. Impaired control: The individual shows impaired control over gaming (frequency or duration).
2. Priority given to gaming: The increasing priority given to gaming over other life interests.
3. Continuation of gaming despite negative consequences: The continuation or escalation of gaming despite negative consequences.

According to the ICD-11 [14], those struggling with gaming disorder frequently fail to reduce their time spent on gaming despite efforts, especially when persuaded by others. They may increase the time spent gaming or game difficulty to keep feeling excited. In addition, if they decrease their time gaming or even stop it, they could go through mood changes or neglect healthy routines like eating, sleeping, or exercising [14].

To diagnose IGD, a set of symptoms needs to be evaluated, as certain behaviors like high frequency of play or improving abilities in a game do not directly indicate a disorder [14]. It's important to consider factors such as age, gender, cultural norms, and peer influences [14]. The consequences differ, for example, young males are often more affected by IGD. Adolescents might struggle with anger, emotional instability, and low self-worth, whereas adults tend to feel anxious and depressed [14].

2.2 Risk factors in IGD

IGD is a disorder that can affect anyone, regardless of their age or demographic. However, certain circumstances may increase the chances of being diagnosed with IGD [16]. Biological tendencies like addiction and neurotransmitter imbalances, personality traits such as emotional instability, and environmental elements including difficult family or educational environments are key to determining IGD vulnerability [16]. Personal crises and major life transitions also contribute to the risk of IGD. Identifying these elements is crucial for developing effective preventive and intervention strategies [16].

Torres-Rodriguez et al. [16] listed prominent risk factors for IGD:

- Biological risks: Addiction tendencies, neurotransmitter issues, and psychiatric conditions.
- Personality vulnerabilities: Emotional instability, identity issues, low self-esteem, poor self-discipline, low resilience, search for intense sensations, and weak social skills.
- Environmental concerns: Family troubles like poor communication, lack of supervision and unity; low academic performance; and poor social settings.
- Stress factors: Grief, major personal crises, or life changes [16].

Studies highlight the role risk factors play in IGD's emergence and maintenance. Following that, below are some examples of those studies.

Rodriguez et al. [22] described the PIPATIC program (Programa Individualizado Psicoterapéutico para la Adicción a las Tecnologías de la información y la comunicación), which translates to Individualized Psychotherapeutic Program for Addiction to Information and Communication Technologies. This program is specifically designed as a Cognitive-Behavioral Treatment for adolescents aged 12 to 18 with Internet Gaming Disorder. This study evaluated the PIPATIC program's effect on teenagers with IGD and took note of familial and educational difficulties pre-treatment, alongside comorbid conditions like depression, social phobia, autism and ADHD (Attention-deficit/hyperactivity disorder). In the evaluation, reductions in gaming and IGD symptoms were observed in the participants post-treatment [22].

Celine et al. [23] correlated non-problematic gaming with cohesive families, while problematic gamers often come from conflicted homes. In female gamers, parents restricting gaming is linked to increased IGD levels. These findings highlight the influence of family relationships on IGD and point to the need for prevention programs that focus on the family's role [23].

Melina et al. [24] examined the connection between parental gaming disapproval, self-evaluation, and IGD. Surprisingly, while parental disapproval shows a slight positive correlation with IGD, lower self-esteem is more substantially associated with increased IGD scores [24].

Wartberg et al. [25] found in their study that hyperactivity, attention deficits, and low self-esteem are indicators for developing IGD, which may also affect adolescent mental health [25].

2.3 Comorbidities in IGD

Comorbidity refers to the presence of more than one disease or condition in a person at the same time and the interaction between such conditions can complicate their management [26]. In the case of IGD, it is known to accompany various mental health issues [27, 28], making it essential to understand these related psychopathologies for effective treatment.

The DSM-5 outlines anxiety disorders as characterized by fear, worry, or excessive apprehension, including conditions like social and panic phobia, and depression as a persistent and intense sadness affecting daily life [5]. Meanwhile, Attention-Deficit/Hyperactivity Disorder is marked by difficulties with focus, concentration, and behavior, and Autism Spectrum Disorder (ASD) involves challenges in social interaction and communication. Additionally, Obsessive-Compulsive Disorder (OCD) includes unwanted thoughts and repetitive behaviors [29].

Research shows a strong association between IGD and mental disorders, particularly anxiety, depression, hyperactivity, and social phobia [27]. There's debate over whether IGD can predict these issues, particularly depression. Jeong et al. [28] recognized a bidirectional link between IGD and depression, calling for integrated prevention and treatment strategies for children.

Furthermore, studies show that anxiety, depression, and hyperactivity symptoms correlate significantly with IGD, with social phobia also strongly linked [27]. Because of these associations, a simultaneous examination of IGD and mental health is suggested. Fazeli et al. [30] found that depression and anxiety mediate the relationship between IGD, insomnia, and quality of life, emphasizing the need for families to monitor gaming habits and support children facing stress.

ADHD has also been identified as a risk factor for developing IGD [31], and there is evidence that individuals with ASD show a significant tendency toward IGD [32]. Laconi et al. [33] observed that specific game genres correlate with self-esteem and depression levels, and found depressive symptoms were a consistent predictor of IGD across genders suggesting a complex relationship between gaming behavior and psychological factors.

Beyond mental health, studies suggest that excessive gaming can negatively impact sleep and diet, contributing to a sedentary lifestyle, though there are mixed findings on this topic. For example, some studies suggest a general disruption of sleep and physical activity in IGD cases [34, 14], while research conducted in Portugal found that gamers could still maintain healthy lifestyles and regular physical activity [35].

Overall, it's generally accepted that exploring IGD alongside mental health conditions is important, given their strong interrelations [27, 30]. However, the direction of the relationship between IGD and issues like anxiety and depression remains uncertain, with some debate over whether one condition directly causes the other [36].

2.4 Counter-Strike and FACEIT

Counter-Strike 2 (CS2) [37], previously known as CS:GO, is a first-person shooter (FPS) game that focuses on team-based tactics and player combat. It was originally developed as a modification for Valve's Half-Life by Minh "Gooseman" Le and Jess Cliffe in 1999. Valve later took over the development rights, resulting in various updated versions being released [38].

The choice of Counter-Strike 2 for the research is based on several important criteria to ensure the relevance and effectiveness of the data collected for analyzing gaming behaviors, particularly those related to IGD:

1. Relevance to IGD: A recent study [39] indicated that playing massively multiplayer online role-playing games, first-person shooters, like CS2, and real-time strategy games/Multiplayer Online Battle Arena is associated with more time spent gaming and higher endorsement of IGD symptoms.

2. Rich gameplay mechanics: CS2 offers rich gameplay mechanics that expose players to diverse emotional conditions, which is important for studying the psychological effects of gaming behavior.

3. Simplicity and accessibility: The game's simplicity makes it accessible even to players with limited familiarity with the genre, providing a more uniform experience among research participants.

4. Popularity and engagement: CS2 is a popular and widely played game, making it more likely to attract participants and increase engagement during gaming sessions.

5. Supportive community: The CS2 community is active and accessible, open to collaboration on research.

6. Telemetry data access: CS2 is hosted on platforms that provide services for accessing telemetry data with automated and publicly accessible methods, facilitating data collection.

Lastly, the involvement of psychologist Joana Cardoso, a therapist and member of the Portuguese Federation of Esports, who has connections with CS2 players, adds credibility to this choice.

The game sets two teams against each other: the Terrorists (T) and the Counter-Terrorists (CT). The main objective of the two sides varies depending on the game mode, but in the most widely-played mode, Bomb Defusal, the objective of the Terrorists is to plant a bomb at a specified location and guarantee it detonates, while the Counter-Terrorists seek to prevent the bomb from being planted, defuse it if it has been planted, or eliminate the terrorist team [18, 40].

CS2 features a variety of weapons, including pistols, submachine guns, rifles, shotguns, machine guns, and grenades. Each of these weapons has different in-game mechanics and usage strategies associated with it. The game is also known for its wide range of maps, each with unique layouts and strategic considerations. While the specific number of official maps can vary with updates and differences between game versions, CS2 offers some official maps. Community-created maps, available through the Steam Workshop, add countless additional maps to the game [18, 41].

A typical round in Counter-Strike 2 follows this flow:

1. Preparation: Players begin a round with a short period called ”freeze time,” during which they cannot move but can purchase weapons and equipment using in-game currency earned from their performance in previous rounds.

2. Objective Pursuit: After freeze time, players execute their strategic plans. For the Terrorists, this often involves moving to designated bomb sites (A or B) to plant the bomb. Counter-Terrorists may either spread out to defend both sites or stack a particular site to counter the Terrorists' strategy.

3. Combat: Players engage with the enemy, trying to eliminate the opposing team while also pursuing their objective.

4. Bomb Planting (Terrorist): If the Terrorists plant the bomb, the dynamic of the round changes, putting pressure on the Counter-Terrorists to retake the bomb site and defuse the bomb before its timer expires.

5. Resolution: A round ends when one team eliminates the other, the Terrorists successfully detonate the bomb, or the Counter-Terrorists defuse the bomb. Rounds can also end if the time limit expires before the Terrorists can plant the bomb, which results in a win for the Counter-Terrorists.

The overall match is won by the team that wins a set number of rounds, typically 13 in competitive play, out of a maximum of 24 [40].

FACEIT

FACEIT [42] is a competitive gaming platform that allows players to engage in Counter-Strike 2 matches outside of Valve's standard matchmaking system [43]. It offers a free-to-use service with optional premium features for enhanced gameplay. Players can connect their Steam [18] accounts, use advanced anti-cheat measures and participate in various game modes like solo, group, or team play. FACEIT uses an ELO (a rating system to calculate the relative skill level of players in games [44]) points system to rank players across 10 levels, with a special Challenger rank for the top 1000 players [43]. The platform also provides opportunities for players to advance to professional esports through its Pro League.

The platform offers several advantages, including free access to basic features without the need for a subscription, an advanced anti-cheat system that ensures a fair gaming environment, skill-based ranking using the ELO system for balanced matchmaking, and a "Path to Pro" feature that provides opportunities for players to join professional leagues and teams.

This platform plays a crucial role in our project by providing a **free-to-use API** that grants access to a lot of player information and demofiles which are files that provide detailed information about a certain match (for example, health points and weapons purchased). While other platforms such as Steam and HLTV [45] were initially explored, they were considered unsuitable due to restrictions and data limitations. The FACEIT API, however, allows us to retrieve detailed data on players, teams, competitions, and game statistics, all formatted as standard REST-ful JSON resources [46]. The availability of such comprehensive data enabled us to analyze player performance, track progress, and enhance our overall research and development process.

2.5 Web Development Frameworks

The main goal of this project is to develop an interactive application to help health professionals and selecting an appropriate web development framework is crucial for building efficient applications. This section analyzes some frameworks for developing web applications and presents the motivation for the chosen one.

These frameworks, which can be client-side (front-end) or server-side (back-end) [47], offer a collection of pre-written, standardized code in the form of libraries and APIs. Thanks to this capability, developers can construct resilient web applications without starting from zero. Frameworks have structures and features that can impact how things are developed, how well they work, the development process, performance, scalability, and ultimately the success of web projects. For the development of the front-end, frameworks like React [48], Vue.js [49], and Angular [50] focus on the user-facing elements of a web application [51]. On the server side, there are frameworks like Express.js [52], Django [53], Ruby on Rails [54], and ASP.NET [55] that handle different server programming needs. They come with different levels of abstraction, connecting to databases, and support for middleware [51]. In the context of the proposed application, we used Next.js [56], a full-stack JavaScript framework that combines React on the front-end with

Node.js [57] on the back-end. This approach offers several advantages aligned with the project's goals. By using a single language (JavaScript) throughout the application, development and integration between the front-end and back-end are significantly simplified. Many applications use this full-stack JavaScript combination, such as MERN (MongoDB [58], Express, React, Node.js) and PERN (PostgreSQL [59], Express, React, Node.js), making it a popular and efficient choice. The details of how Next.js is implemented in this project will be discussed in more detail in future chapters.

Next.js and Node.js are known for the following benefits:

- **High performances:** Node.js is great for making really fast apps that get results quickly. It can handle many tasks at once, making it useful for web applications. Its design allows it to process lots of requests simultaneously without slowing down the computer. This is because of its special way of handling events and not blocking other tasks while doing one thing [60].
- **Scalability:** Node.js provides flexibility for developers to expand applications both horizontally and vertically. Horizontal scaling involves adding more nodes to the current system, allowing for increased capacity and performance. In contrast, vertical scaling in Node.js enables the addition of extra resources to individual nodes, enhancing scalability. This adaptability makes Node.js a highly scalable and superior choice compared to other JavaScript servers [60].
- **Easy to learn:** Since many frontend developers are familiar with JavaScript, incorporating Node.js on the backend becomes easier for them. Understanding Node.js is more straightforward, and the learning curve is shorter, making it a quicker process to work with [60].
- **Reduces loading time using caching:** Node.js caching module simplifies the process for developers to reduce task workload and avoid unnecessary code re-execution, a significant advantage of Node.js. This feature involves caching the initial module of a web application in the in-app memory upon access. As a result, users can navigate online sites swiftly without enduring prolonged waiting times [60].
- **Improves response time and boosts performance:** Node.js employs a single-threaded event-loop approach, featuring a non-blocking asynchronous design. This design, using fewer resources by avoiding the creation of additional threads, enhances application responsiveness, effectively managing multiple concurrent users [60].
- **Large community support:** The Node.js community is active and constantly contributing to its rich ecosystem of modules available through the npm (node package manager) [61], which can accelerate the development process [60].
- **Cost-effective development:** Using a full-stack JavaScript approach, combining React and Node.js, streamlines development and reduces the need for separate resource teams for the front-end and back-end, leading to significant savings in time and costs [60].

For the database system, MongoDB [58] will be used. MongoDB is a document database that stores data in JSON-like documents with flexible and dynamic schemas. It's a NoSQL database, which means it differs from traditional relational databases that use SQL and tables of rows or columns [62]. MongoDB offers distinct advantages that align with the project's goals. It is known for its:

- **Complete developer data platform:** MongoDB offers a collection of services that integrate with the database, such as performance optimization, full-text search, data visualization, and multi-cloud deployment [63].
- **Flexible document model:** MongoDB allows for the storage and manipulation of data in a format that is close to the objects used in programming languages and supports schema validation and dynamic changes [63].
- **Advanced features query API:** MongoDB enables to query data in various ways, from simple lookups to complex analytics pipelines, without requiring joins or transactions [63].
- **Scalable and distributed:** MongoDB is designed to handle large and high-throughput collections, with real-time replication and sharding capabilities [63].

In the development stages, the familiarity and advantages of Node.js and MongoDB played a crucial role in ensuring the effectiveness of the interactive application. As we progress into implementation, Node.js and MongoDB stand as a reliable foundation for achieving the goals of this research project.

2.6 Summary

This section offers a comprehensive overview of the foundational concepts necessary to understand the study of Internet Gaming Disorder and its associated mental health implications. It begins by defining IGD, detailing its diagnostic criteria according to the DSM-5 and ICD-11, and discussing its associated risk factors and comorbidities, such as anxiety, depression, and ADHD. The section also introduces the game Counter-Strike 2 (CS2) and the FACEIT platform, which are central to the study, explaining their relevance to the research. Lastly, it covers the web development tools chosen for the project, specifically the Node.js framework and MongoDB database, highlighting their technical benefits and suitability for the application being developed. This background lays the groundwork for understanding the study's focus, methodology, and technological approach.

Chapter 3

Related Work

This chapter reviews the key areas relevant to this study, including game telemetry analysis, screening tools for Internet Gaming Disorder (IGD), and user interface design for health professionals. It discusses the various questionnaires developed to diagnose IGD, examines existing research on how game telemetry is used to analyze player behavior and performance, and highlights best practices in designing healthcare interfaces to improve decision-making and workflow efficiency. Each section provides critical insights that inform the design and implementation of tools to understand better and address gaming behaviors and mental health.

3.1 Screening tool questionnaire

Various self-report questionnaires are used in research and in clinical settings to diagnose Internet Gaming Disorder, although they are only used when a therapist suspects a patient may have IGD [17]:

- **Internet Gaming Disorder Scale—Short-Form (IGDS9-SF):** Created by Pontes and Griffiths (2015) [64], this short questionnaire includes nine items that evaluate several aspects of gaming behavior, such as preoccupation with games, withdrawal symptoms, and the impact on other areas of life. This scale has been validated in different languages, including Portuguese [65] and Spanish [66].

- **Problematic Online Gaming Questionnaire (POGQ):** Demetrovics et al. [67] developed the POGQ, which has 18 items covering six dimensions of problematic online gaming, as identified through their factor analysis.

- **Internet Gaming Disorder Scale (IGDS):** Lemmens et al. (2009) [68] created the IGDS, a much longer scale with 27 items that explore different aspects of gaming behavior. A shorter 9-item version is also available.

- **Gaming Addiction Scale for Adolescents (GASA):** This questionnaire, developed by Lemmens and Valkenburg [69], uses 21 items to measure seven factors related to gaming addiction (like salience and mood change).

- **Video Game Addiction Test (VAT):** This test was created by Van Rooij and Schoenmakers (2012) [70] to identify video game addiction symptoms.
- **Gaming Addiction Identification Test (GAIT):** This test was created by Vadlin et al. [71] and is a screening tool designed for detecting gaming addiction in teens.

In the work developed before, the IGDS9-SF was used since it is one the most common questionnaires used in research and clinical practice to assess IGD. It consists of 9 questions that reflect the nine criteria previously defined by the DSM-5 [5].

Below are the nine questions of the questionnaires by Pontes and Griffiths [64]:

1. Do you feel preoccupied with your gaming behavior? (Some examples: Do you think about previous gaming activity or anticipate the next gaming sessions? Do you think gaming has become the dominant activity in your daily life?)
2. Do you feel more irritability, anxiety or even sadness when trying to reduce or stop your gaming activity?
3. Do you feel the need to spend an increasing amount of time engaged in gaming to achieve satisfaction or pleasure?
4. Do you systematically fail when trying to control or cease your gaming activity?
5. Have you lost interest in previous hobbies and other entertainment activities as a result of your engagement with the game?
6. Have you continued your gaming activity despite knowing it was causing problems between you and other people?
7. Have you deceived any of your family members, therapist, or others because of the amount of your gaming activity?
8. Do you play to temporarily escape or relieve a negative mood (e.g., helplessness, guilt, anxiety)?
9. Have you jeopardized or lost an important relationship, job, or educational or career opportunity because of your gaming activity?

This questionnaire quantifies how the patient feels according to the nine DSM-5 criteria for IGD. The answer to these questions uses a Likert scale from Never to Very Often. Each question gives points from 1 to 5, so the scale produces final scores between 9 and 45.

The key differences among IGD screening tools include their scope, target population, language and cultural adaptations, criteria for assessing problematic gaming, and scoring and interpretation methods [17]. For example, the IGDS9-SF aligns its criteria with the DSM-5, while other tools introduce unique criteria, leading to variations in scoring and interpretation that can influence how clinicians or researchers assess the severity of IGD [64]. Furthermore, self-report approaches present unavoidable limitations, including potential biases in a recall, tendencies towards denial or defensiveness, and a lack of insight [17]. A recent survey conducted by King et al. [17] underscores the research community's persistent uncertainty and divergent opinions on the best screening and assessment practices. These tools vary in scope, population, language, cultural adaptations, and scoring criteria, which can influence IGD severity assessments [72].

This uncertainty is reflected in the continuous introduction of new assessment tools characterized by diverse scopes and content, reflecting a lack of consensus in the field. Király et al. [73] also emphasized the need for cross-cultural validation in IGD tools, such as their work on the Ten-Item Internet Gaming Disorder Test (IGDT-10), which was validated across multiple languages.

Despite the range of IGD screening tools, their limitations indicate the need for more objective methods. These tools may not fully capture gaming behavior complexity and variability, and may not keep up with the evolving gaming industry. Consequently, there's a need for methods to provide more information on gaming behavior and its impact on mental health. One such method is the use of game telemetry data, which can record and analyze various aspects of gaming activity, such as playtime, session frequency, in-game events, and performance [3]. By using game telemetry data, mental health professionals can obtain more information about a player's gaming behavior, and identify patterns or indicators of IGD that may not be evident from self-reports.

3.2 Game telemetry analysis

Video games have become increasingly complex and now include technical advancements that make gathering game telemetry easy and common [74, 19]. This data, known as game telemetry, is collected in real-time using sensors and other tools [19]. It records everything from general game data to specific player actions and events, providing valuable insights into how players interact with the game, improving game usability and difficulty balancing [20, 19].

This information serves multiple purposes. It helps game designers understand how players use and interact with the game environment and its objects, which can help them design more intuitive and enjoyable games [20]. Additionally, it helps analyze player and team performance to improve tactics and strategies. Telemetry can track details like a player's movement, accuracy, and reaction time, which coaches can use to enhance player skills [2, 75]. In team esports, this data is even more critical, as it provides real-time insights that can influence coaching decisions during competitions [2, 75].

Beyond gameplay mechanics, game telemetry has emerged as a potential tool for studying player behavior from a health perspective. It allows researchers to analyze problematic gaming behaviors, including those associated with IGD [76]. Drachen [76] and Deterding et al. [77] argue that telemetry data can identify problematic behaviors that may not be evident through self-reports. Although its potential remains not sufficiently explored, recent studies have integrated telemetry data with psychological measures to understand better the relationship between gaming and well-being [78]. Vuorre et al. [78] analyzed in-game events alongside self-reported ratings on six psychological measures (autonomy, competence, enjoyment, focus, immersion, and well-being), showing that game telemetry can contribute to the measurement of psychological outcomes.

Zendle et al. [79] analyzed telemetry data from mobile games to investigate playing habits across different countries, thereby identifying regional variations in gaming behavior. They did find notable differences, but they didn't explore IGD. Meanwhile, Padman, Redma, and their team [80] also looked at gaming data, but for investigating health, specifically focusing on childhood obesity linked to gaming on mobile apps, not IGD.

Furthermore, Parry's work [81] shows that when people report their gaming habits, the numbers don't always match up with actual gaming data logged by devices. This is especially true for social media use, where there's a weak link between self-reports and real figures. They found that gamers usually report playing 1.26 hours less per week than they actually do. Kahn's study [82] is the only one out of 106 that looks into gaming, highlighting an important point: people might not give the most accurate accounts of their own gaming habits.

Supporting this claim, Andrews [83] examined smartphone use by comparing self-reported habits with actual usage data. The smartphone study found that people's self-reports don't always line up with how long and how often they actually use their phones.

Adding to this, Thompson et al. [84] analyzed game data to understand how players learn complex skills. They noticed that the key factors in their machine learning models changed with the player's skill level, implying that using different datasets is crucial to get a real sense of how people learn and play games.

Although several studies have explored aspects of gaming behavior, few have specifically explored telemetry data to extract detailed insights into Internet Gaming Disorder. This highlights the need for further research into the potential of game telemetry as a tool for identifying and understanding problematic gaming behaviors more objectively and comprehensively.

3.3 User interface design for health professionals

The design of user interfaces is important in all areas, but it plays a particularly crucial role in healthcare applications, where effective communication and interaction between healthcare professionals and patients is essential. A well-designed interface promotes a smooth and easy exchange of information, which is critical for the successful operation of healthcare applications [85, 86]. Given the complexity of healthcare data, user interfaces must be both functional and user-friendly, ensuring that health professionals can easily interpret data and make informed decisions promptly.

Although the design of UI for these applications follows the general guidelines, several studies highlighted that healthcare applications need to focus on clear data presentation and productivity enhancement to enable quicker decision-making by professionals [85, 86]. A simple and minimalistic design approach ensures that users can navigate through complex medical data without feeling overwhelmed, which is essential in high-pressure environments like hospitals and clinics. This minimalist design also helps reduce cognitive load, allowing professionals to focus on patient care rather than struggling with complex interfaces [85, 86].

Another critical aspect of user interface design is personalizing the interface to meet the diverse needs of various users, including doctors, nurses, and administrative staff. Designing the interface for specific user groups enhances accessibility and usability, making the application more effective across different roles [85, 86]. This personalization can result in higher user engagement and satisfaction, ultimately contributing to better health outcomes through more efficient workflows and improved patient monitoring.

Moreover, the design of user interfaces in healthcare should also incorporate cognitive science principles to account for the mental processes and limitations of healthcare professionals. Patel et al. [87] emphasize the need to consider the cognitive capabilities and limitations of end users. This includes applying insights from human reasoning and decision-making to create interfaces that are intuitive and support the complex decision processes involved in medical diagnosis and treatment. An iterative design process that integrates feedback from actual users can significantly improve usability, ensuring that interfaces meet the practical needs of health professionals in real-world settings.

Tang et al. [88] similarly argue that the requirements of end-users must be carefully considered during the design of user interfaces for healthcare professionals. Poorly designed interfaces can disrupt workflows and decrease the utility of healthcare workstations. To avoid this, Tang et al. propose user-centered design methodologies such as ethnographic studies and cognitive analyses to accurately capture the needs of healthcare workers. Understanding these needs is vital for designing interfaces that align with professional tasks and the way health professionals process and utilize information.

Enhancing the interaction between health professionals and their digital workstations involves several key improvements, including the optimization of input/output devices, development of patient context-sensitive interfaces, and intelligent information presentation [88]. These features help simplify interactions, making systems more intuitive and efficient. Additionally, the ability for users to customize interfaces based on their preferences further supports productivity and enhances user satisfaction by allowing health professionals to tailor the system to their workflow [88].

Overall, the design of user interfaces for healthcare applications must balance simplicity, personalization, and cognitive considerations to ensure they meet the needs of health professionals while enhancing their ability to make quick and informed decisions. Through a user-centered design process and the integration of real-time feedback, healthcare UIs can evolve to support both routine patient care and the complex decision-making processes required in healthcare environments.

In order to follow these guidelines, this work adopts a user-centered design approach, incorporating feedback from healthcare professionals and close collaboration with domain experts to ensure that the interface meets the specific needs of users and improves their workflow efficiency.

3.4 Summary

This chapter explored key areas relevant to understanding and addressing Internet Gaming Disorder through game telemetry, screening tools, and user interface design.

Various screening tools, such as the IGDS9-SF, POGQ, and GASA, offer ways to diagnose IGD based on self-reports, though these methods are subject to biases and inaccuracies. The chapter highlighted the need for more objective assessment tools, such as the integration of game telemetry with psychological measures, to provide a more comprehensive understanding of gaming behavior.

Game telemetry analysis has proven valuable for understanding player behavior, enhancing gameplay, and offering insights into gaming behaviors related to mental health, including IGD. While telemetry provides objective data, its potential remains not very used in health contexts, though research highlights its relevance.

User interface design for healthcare professionals was also discussed, emphasizing the importance of usability, personalization, and cognitive considerations in designing effective healthcare applications. By prioritizing intuitive designs and user-centered approaches, healthcare interfaces can significantly improve decision-making and workflow efficiency.

Together, these insights form a foundation for the design and development of tools that address gaming behaviors and support mental health professionals in diagnosing and treating IGD.

Chapter 4

GDHelper Analysis and Design

This chapter goes into the comprehensive analysis and design of the GDHelper application, a tool aimed at assisting in the diagnosis and treatment of Internet Gaming Disorder (IGD). By integrating telemetry data with psychological assessments, GDHelper aims to provide health professionals with a dynamic and interactive platform to monitor and analyze gaming behavior. It outlines the proposed solution, detailing the software development methodology, data workflow, and system architecture. This chapter also explores the components and interactions within the system, ensuring data flow and security, and conclude with a summary of the design considerations and decisions made throughout the development process.

4.1 Definition of the proposed solution

Various IGD diagnostic approaches are explained in this document. Most studies reviewed rely on interviews, surveys, and similar analog methods. However, these can be problematic for therapists, as patients may provide inaccurate information about their gaming habits, such as time spent playing [82].

The previous master's thesis [3] aimed to assist in the IGD diagnosis and treatment by leveraging gaming telemetry. Gaming patterns were analyzed - when, how often, and for how long patients play - to track behavior intensity over time. Following data collection, metrics, indicators and measurements were developed to assist therapists throughout treatment. They explained how crucial it is for therapists to review recent gaming activity before each session. Specific indicators were created for this purpose, allowing healthcare professionals to assess gaming intensity and emotional fluctuations since the last appointment.

The metrics and visualizations were designed to address the nine IGDS9-SF questions. For instance, by examining gaming time trends, it's possible to determine whether gaming has become the patient's primary daily activity or if they've stopped other hobbies for gaming - key DSM-5 [5] diagnostic criteria. This approach enables therapists to identify discrepancies between questionnaire responses and telemetry data. Such mismatches could indicate deception or time perception issues relevant to DSM-5 criteria.

The limitations of traditional assessment techniques discussed in this document have encouraged the pursuit of more precise and dependable methods for diagnosing Internet Gaming Disorder (IGD) [15]. As identified by the the previous master's thesis [3], the valuable metrics such as gameplay duration, and the number of sessions per year, month, and day experienced by players of Counter-Strike created visualizations to help health professionals analyze certain diagnostic criteria for IGD.

However, these visualizations have some drawbacks. First, they are static, which means they are based on fixed game data and do not reflect the changes in gaming behavior over time. Second, they display data for only one player, not allowing comparisons with other players or the general population.

These limitations highlight the need for an interactive tool to support the diagnosis of IGD, enabling health professionals to monitor telemetric data over time and complement questionnaires and monthly clinical follow-ups with players. As a result, GDHelper, our interactive application was developed to overcome these limitations while integrating the previously studied metrics and visualizations, accessing game data as frequently as the user requires, and providing a tool to help health professionals analyze the player's gaming behavior before each consultation and including previous statistics.

In the context of established IGD treatment protocols, such as the widely used cognitive-behavioral program PIPATIC [16], GDHelper can assist the process of diagnosing IGD. In traditional treatment methods, therapists conduct initial assessments using tools like the IGDS9-SF questionnaire, supplemented by interviews and follow-up sessions, to monitor gaming behavior progression over time. GDHelper integrates into this workflow by providing objective, regular data on a patient's gaming habits, which therapists can use alongside traditional self-reported information to get a more accurate understanding of the patient's behavior between sessions.

Furthermore, GDHelper allows therapists to review recent gaming activity before each session, which aligns with the need for constant monitoring in treatment. The tool supports health professionals in identifying any significant changes in gaming behavior since the last appointment, so the therapist can plan the next steps and adjust treatment strategies accordingly. By providing visualizations that track gaming intensity over time, GDHelper contributes to the decision-making process, enabling therapists to better understand the patient's gaming pattern and whether they are progressing or regressing in their treatment.

To ensure that GDHelper meets the needs of healthcare professionals in diagnosing and treating IGD, several key functional requirements have been identified:

- User management: Health professionals must be able to create new users (patients) in the system, allowing them to track gaming behavior over time.
- Data management: The system should enable health professionals to update player data as necessary and keep track of players' gaming behavior between sessions.
- Deletion of players: Health professionals should be able to delete users from the system once treatment is complete or if they are no longer relevant to the study.

- Gameplay monitoring: The tool should allow regular access to telemetric data, so health professionals can examine changes in behavior over time.
- Interactive visualizations: The tool must include visualizations that are dynamic and reflect updated data, enabling professionals to quickly interpret and make decisions based on real-time or recent data.

In addition to functional requirements, several non-functional requirements are critical to ensure the success and usability of GDHelper:

- Security: The tool must ensure secure access, including authentication protocols to protect the user's privacy.
- Accessibility: The tool should be accessible to a wide range of users, including those with disabilities.
- Responsiveness: The system should be responsive, providing quick access to data and visualizations.
- Scalability: The tool must be scalable to accommodate multiple users simultaneously and handle increasing amounts of game telemetry data over time.

In conclusion, the proposed solution is a web application that builds upon previously created visualizations. The application will focus on the practical application of the metrics discussed in this chapter, in conjunction with the IGDS9-SF questionnaire, the primary diagnostic tool for identifying IGD. Unlike traditional approaches that focus on a single player, this web application will offer visualizations of any desired player, using regular data. This will provide a more comprehensive understanding of a player's gaming pattern, providing a more dynamic and interactive platform to monitor and analyze gaming behavior for healthcare professionals.

Moreover, alongside meeting functional requirements like user management and gameplay monitoring, the tool ensures secure access, responsiveness, and accessibility, making it a reliable and effective tool for healthcare professionals in diagnosing and treating IGD.

4.2 Software Development Methodology

In the field of software engineering, a software development methodology provides a structured framework that guides the process of planning, designing, developing, and testing software projects [89]. Software methodologies are essential for organizing a team's approach to project management and can significantly impact the efficiency and success of software development efforts because they formalize communication and determine how information is shared within the team [89]. Selecting the right methodology can greatly improve the chances of achieving successful software in terms of cost-effectiveness, meeting deadlines, client satisfaction, software reliability and robustness, and reducing the costs of unsuccessful projects [90].

Numerous methodologies exist, each linked to different types of projects and team dynamics. Among the most widely recognized are the Waterfall model, known for its linear and sequential approach that focuses on collecting requirements and designing the software architecture before doing development and testing [90]; the Agile method, known for the ease of adaptation to changing requirements and for its iterative and incremental process [90]; the Spiral model, noted for its focus on identifying risks [91]; and the V-Model, which focus on testing stages [91].

For this project, we employed an Iterative and Evolutionary Development approach. This approach focuses on developing the software in a series of repeated cycles, typically short and fixed-length, called iterations [1], which allows for continual refinement and evolution of the software [1]. Iteration by iteration, the software grows incrementally over time [1]; for that reason, this methodology is also known as iterative and incremental development. Moreover, as feedback and adaptation drive the evolution of specifications and design, this approach is commonly referred to as iterative and evolutionary development (Figure 4.1).

Additionally, our approach prioritizes a user-centric design [1] to address the needs of healthcare professionals. Our team includes a Clinical Psychologist who works with gaming-related issues in clinical and research settings, ensuring a comprehensive understanding of their needs, and she is also a gamer. She played a crucial role in defining attributes, validating metrics, and analyzing the visualizations during the previous master thesis and in this work in the validation of the tool. Leveraging her firsthand gaming experience, she provided valuable insights that enriched the design process and ensured that the tools effectively meet the needs of healthcare professionals.

Practical Implementation of the Iterative and Evolutionary Development Approach:

The methodology in our project was characterized by weekly (or bi-weekly, depending on specific deadlines) meetings. These meetings served as checkpoints where the current iteration of the application was presented, which was then subject to critique and feedback. This feedback loop was essential for the continual refinement and evolution of the software.

Iterative Process

Each iteration began with a focused set of objectives, which were typically determined based on the feedback received in the previous meeting. These objectives could range from implementing new features to refining existing ones, addressing bugs, or improving user interface elements. By setting clear, manageable goals for each iteration, the development process remained organized and productive.

- **Requirements analysis:** Before each iteration, we would review the feedback from the previous meeting, reassess the requirements, and make any necessary adjustments to the project's scope and priorities. This step ensured that the software development remained aligned with the needs of mental health professionals and the specifics of Internet Gaming Disorder (IGD).

- **Iterative design and development:** A key feature of the iterative approach was the rapid development of a basic version of the application during each cycle. Initially, we created a minimal viable product that covered the most critical features. Subsequent iterations were focused on progressively enhancing the software by adding new features, refining the user experience, and optimizing performance based on feedback.

- **Testing:** Each iteration included a thorough testing phase where new features were tested for functionality, and any bugs identified during previous iterations were fixed. This ongoing testing ensured that the application maintained a high-quality standard throughout its development.

- **User feedback and refinement:** While developing the application, valuable feedback was received from my supervisors, which guided the early stages of design and functionality. Later, we conducted usability testing with psychologists, the intended end-users, to gather insights on the application's effectiveness. This iterative process allowed us to make improvements and ensure the application meets the practical needs of healthcare professionals involved in diagnosing and treating IGD.

Iterative Adaptability and Stakeholder Involvement

The iterative and evolutionary development approach promoted a high degree of adaptability. By working closely with the team and incorporating feedback at every stage, we were able to respond to changing requirements and unexpected challenges efficiently. This methodology also encouraged active stakeholder involvement, ensuring that the final product was closely aligned with the initial vision and user needs.

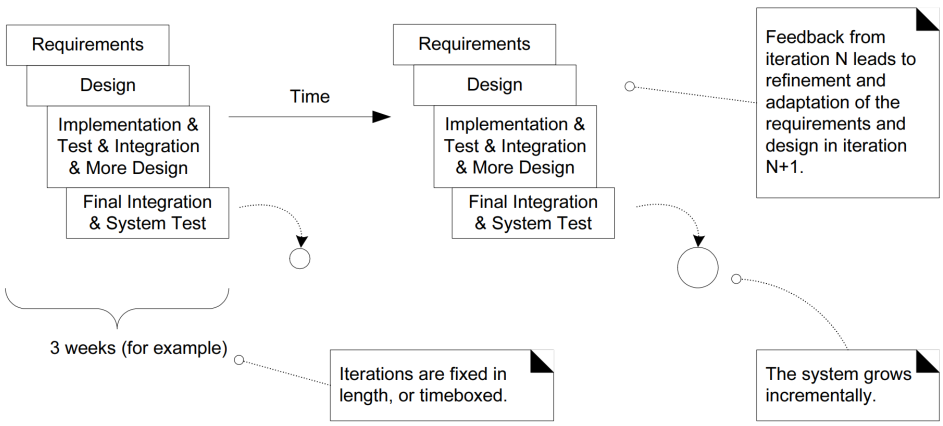


Figure 4.1: Iterative and evolutionary development. This image is reproduced from Larman (2004) [1]

The Iterative and Evolutionary Development method was instrumental in delivering a robust, user-centered software application, enabling continuous improvement and fostering collaboration throughout the project lifecycle. By allowing for regular feedback and iterative refinement, this approach ensured that the application evolved in alignment with user needs and project goals, ultimately leading to a more polished and effective final product.

4.3 Data Workflow

In this section, we explain the process we followed to collect, manage, and analyze gaming data. We used a structured approach called the gaming data life cycle [2], which helps organize the different steps involved in handling data, from defining what data to collect to analyzing the results.

The gaming data life cycle includes steps such as collecting data from players, cleaning it, analyzing it, and then using it to find patterns or insights. We followed this process to ensure that the data we gathered from platforms like FACEIT was accurate and useful for our analysis.

In the following subsections, we will describe how we collected and processed the data, defined important metrics, and analyzed the results to better understand player behavior.

4.3.1 Gaming data life cycle

Before presenting the GDHelper tool, we will explain the fundamental steps in managing and using gaming telemetry data. The gaming data life cycle refers to the stages involved in managing and using data telemetry effectively for analysis and decision-making purposes (see Figure 4.2). According to Seif El-Nasr et al [2], the gaming data life cycle consists of eight steps: attribute definition, data acquisition, data preprocessing, metrics development, analysis and evaluation, visualization, reporting and knowledge deployment (see Figure 4.2). We will focus on every step of the cycle except for the Metrics development since they were previously developed [3].

- **Attribute definition:** This initial and very important step involves defining the objectives, variables, and tracking strategies for the data analysis. While some KPIs and metrics had already been defined, we had to conduct further research to implement them effectively in GDHelper. This process included identifying relevant information to collect and refining the features, metrics, and Key Performance Indicators (KPIs) that would be used for future analysis. KPIs are carefully chosen metrics that help determine whether specific objectives are being met. To draw meaningful conclusions, KPIs require a historical context to provide a basis for comparison [2]. An example of a tracking strategy is monitoring the number of matches a player participates in per day (see Figure 4.3).

- **Data acquisition:** This is the step where it's collected the relevant data from the game telemetry system, which records all the gameplay logs or system logs [2]. The data collected in this step can include in-game events, match dates and duration and even player behaviour. This requires a database to store all the information collected and subsequently written code for this effect. For this study, our information is going to be gathered through the FACEIT API (more details in section 2.4).

- **Data preprocessing:** This is the step where the data is checked and cleaned before analyzing it [2]. For example, removing empty values, errors, outliers or even irrelevant information that may affect the results [2]. Before any further analysis, it is essential to complete this stage, which involves storing the collected data in databases or data files, either locally or in the cloud. Fortunately, the quality of our main data source which was the FACEIT API was high, eliminating the need for a data cleaning process.

- **Metrics development:** This is the process where it's constructed meaning or abstractions from the data [2]. In other words, new and more complex features are developed [3]. Although the metrics were initially developed in a previous master's thesis [3], in this work, we had to re-implement and adapt them within the GDHelper system to ensure compatibility and functionality. We detail the specific metrics used in section 4.3.3. For example, the concept of a session has been created, which stands for a sequence of matches of a player that are separated by no more than 30 minutes.

- **Analysis and evaluation:** This step typically involves using statistical or machine learning methods to test hypotheses, build models, or find patterns in the data [2]. However, in our work, the primary techniques applied in this step were statistical inference and visualization approaches. These methods allowed us to reveal patterns in player behavior, which we detail further in section 4.3.3. The goal was to gain a deeper understanding of gaming habits and trends by leveraging these techniques to validate hypotheses and create meaningful visual representations of the data.

- **Visualization:** This is the step where it's used graphical tools to display the data or the results of the analysis. For example, techniques such as tables, graphs, charts and heatmaps. The visualization step makes it easier to understand the data extracted and to demonstrate it to stakeholders and have the added benefit of enabling the identification of patterns, trends, and anomalies with greater ease [2]. Moreover, they can even be used during the data preprocessing stage to detect outliers or missing data, thereby enhancing the overall quality of the analysis [2].

- **Reporting/knowledge deployment:** This is the step where the results and knowledge discovered throughout the cycle it's presented to the relevant stakeholders [2]. These results are typically presented clearly and concisely, using plots and tables, and are often deployed in dynamic dashboards to facilitate easy interpretation and understanding. This final step of the cycle often catalyzes a new iteration, sparking a fresh cycle of data-driven discovery and decision-making. For example, using plots or tables to show the key metrics and insights for a player.

This project has successfully navigated every stage of the gaming data life cycle, leveraging it as a guiding framework. Some phases, like Reporting/knowledge deployment and visualization, were initially addressed in the previous thesis [3], but in our work, we adapted these stages to the context of GDHelper. For example, the Reporting/knowledge deployment phase, which involves communicating results to stakeholders, was previously completed but executed again during the GDHelper evaluation process to ensure that the system's results were properly communicated and visualized within the web application. Similarly, Visualization was implemented in the earlier work, yet we recreated the visual representations, this time implementing them directly into the GDHelper website to make them more accessible and interactive. In the following subsections, we will get into the specifics of how we applied the data acquisition and preprocessing stages, defined metrics and analyzed and interpreted the results, organized by topic and source.

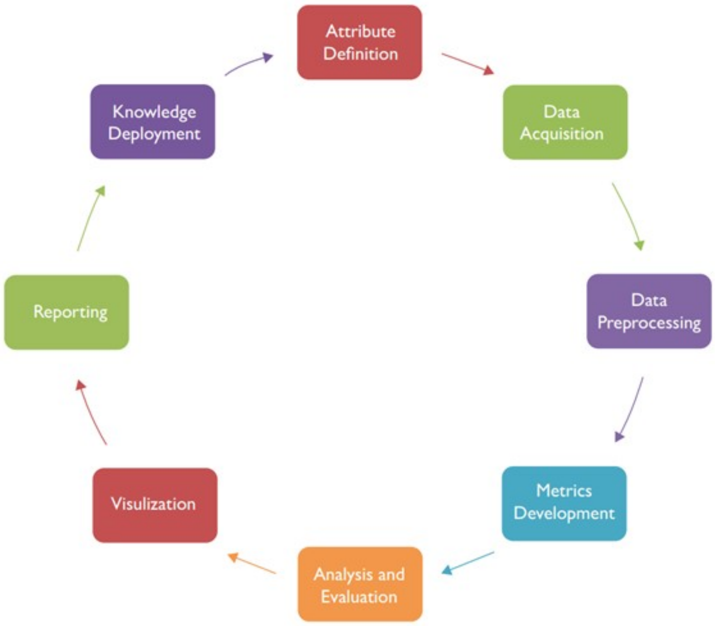


Figure 4.2: Gaming data life cycle. The figure is reproduced from Seif El-Nasr et al. [2]

4.3.2 Data collection

This section outlines the process of Attribute Definition, Data Acquisition, and Data Preprocessing as part of the data life cycle. The primary focus is on gathering telemetry data related to gaming performance, particularly for players of Counter-Strike 2 on the FACEIT [42] platform. The data includes match statistics, player performances, and various in-game events that are instrumental in analyzing player behavior and performance trends.

Unlike previous studies [3] that relied on a fixed group of participants, our approach leverages the flexibility of our application, which allows any user to analyze the data of any player with a FACEIT account. Users can input the FACEIT nickname of any player they wish to analyze, and our application retrieves the relevant data directly from the FACEIT API [46]. This feature enables large-scale data collection, providing a more comprehensive analysis of gaming behaviors across different skill levels and regions. It is worth noting that we only used publicly available gaming data platforms.

To obtain the gaming data, we explored several potential sources, including Steam [18], HLTV

[45], and FACEIT [42]. However, these platforms were considered unsuitable for our needs due to restrictions and limitations associated with Steam and HLTV.

- **Steam:** While Steam is a widely used platform for managing and playing Counter-Strike 2, its API does not provide the detailed telemetry data necessary for our analysis. Moreover, accessing certain data, like login history, requires player permission and is not scalable for large datasets.
- **HLTV:** HLTV offers rich telemetry information but is limited to official competitions. This restricts access to a player's complete gaming history, making it insufficient for our analysis, which requires data from all matches played.
- **FACEIT:** Ultimately, we selected FACEIT as the primary data source due to its comprehensive API, which offers detailed telemetry data for every match played by a player. FACEIT's API provides access to various data points, such as player statistics, match outcomes, and in-game events. This data is critical for analyzing player performance over time and across different contexts. Additionally, FACEIT provides demofiles for all matches, which are structured data files similar to JSON. These files are organized into multiple levels and contain detailed information about in-game events, including player health and income for specific rounds. While these demofiles offer a rich source of data, our approach did not require their information, as it is more suited for analyses focused on emotional player behavior during gameplay.

The data collection process using the FACEIT API involves the following steps:

1. **Player ID retrieval:** Users provide the FACEIT nickname of the player they wish to analyze. Our application then uses the Search endpoint of the FACEIT API to retrieve the player's unique ID and then stores it in our own database.
2. **Player information collection:** With the player ID, we retrieve general information about the player, including their region, skill level, and historical gaming performance, using the Player endpoint.
3. **Match data retrieval:** The Matches endpoint allows us to collect data for all matches the player plays. This includes match IDs, start and end times, team compositions, and detailed player statistics such as kills, deaths, assists, and kill-death ratio.

The FACEIT API is well-suited to our needs due to its flexibility and the granularity of the data it provides. This setup allows us to collect and analyze data at both the individual and community levels, offering a comprehensive view of gaming behaviors. Telemetry data can be collected from any data span as long as the timestamp wanted is provided to the API. Data retrieval from players has been available since 2018 because it was the year the FACEIT platform began, enabling the analysis of long-term trends and seasonal variations in player performance.

4.3.3 Metrics and Visualizations

In the context of gaming data analysis, FACEIT [42], a competitive gaming platform, provides two primary sources of information: the API and Demofiles. The API offers general information such as the match's final score, date, and time. In contrast, the Demofiles provide more granular data for each match, including the exact moment a player eliminates another.

To analyze the gaming data collected from FACEIT, we need to define some metrics that can capture the relevant aspects of gaming behavior. These metrics originate from the ones proposed by Liniers (2023) [3] which are based on the criteria for IGD. The metrics used are:

- **Monthly hours:** The total number of hours played by each player per month, calculated by summing up the duration of each match.
- **Daily hours:** The total number of hours played by each player per day, calculated by summing up the duration of each match.
- **Monthly hours by time of day:** The distribution of the monthly hours by morning (from 6 am to 1 pm), afternoon (from 2 pm to 9 pm), and evening (from 10 pm to 5 am), to analyze the gaming habits and preferences of each player.
- **Monthly hours by weekday/weekend:** The distribution of the monthly hours by weekday (from Monday to Friday) and weekend (Saturday and Sunday), to examine the impact of gaming on the players' daily routines and obligations.
- **Number of matches:** The total number of matches played by each player per month or days, obtained by counting the number of matches in the API data.
- **Session:** Represents a series of consecutive matches played by a gamer, starting when they begin playing and ending when they stop. Consecutive matches are those played within 30 min of each other. Each match belongs to only one session, but multiple matches may be part of one session. Multiple sessions can occur within a day [92].
- **Number of sessions:** The total number of sessions played by each player per month or days, defined as a group of matches that are separated by no more than 30 minutes. This metric can indicate the frequency and intensity of gaming, as well as the difficulty of stopping playing.
- **Session duration:** The average duration of each session in minutes, calculated by dividing the total session time by the number of matches in the session. This metric can also reflect the intensity and persistence of gaming.
- **Session frequency:** The average number of sessions per day, calculated by dividing the number of sessions by the number of days in the month. This metric can show how often the players engage in gaming.

- **Hours per session ratio last 7 days:** This metric indicates the average time spent playing games per session over the last 7 days. A higher ratio may suggest that the player is engaging in longer gaming sessions, which could be an indicator of increased gaming intensity or addiction.

- **Difference hours/session from last week or last month:** This metric shows the percentage change in the average duration of gaming sessions compared to the previous week or month. An increase in this metric could indicate that the player is spending more time playing games per session, while a decrease could suggest a reduction in gaming intensity.

- **Hours played difference from last week or month:** This metric represents the percentage change in the total hours spent playing games compared to the previous week or month. An increase in this metric could indicate an overall increase in gaming activity, while a decrease could suggest a reduction in gaming frequency or duration.

These metrics are then used to create visualizations that can help the health professionals to understand the gaming patterns of each player, and to compare them with other players or with the general population. The visualizations include tables, charts, and plots that show the monthly, weekly, and daily variations of the gaming data, as well as the correlations between different metrics. In Figure 4.3 and Figure 4.4, we can see examples of the metrics being used and the visualizations that were proposed for the analysis of the data (visualization step in Figure 4.2).

In Figure 4.3, the vertical axis displays the *number of matches* (represented by blue bars) and *sessions* played (represented by orange bars) per day, 12 days before the date the information was extracted from. The *number of matches and sessions* is calculated by counting the occurrences of these events for each day. Through accessing the FACEIT API, it's possible to extract information about the start and end time of each match, as well as the players involved. The sessions, as mentioned before, are the *total number of sessions* played by the player per day, defined as a group of matches that are separated by no more than 30 minutes. Having the list of matches per day of the player makes it possible to calculate the *number of sessions*. By analyzing this visualization, we can see the player had only 3 matches on most days, equivalent to 2 hours per day, sometimes in more than one session. This can help to see the fluctuations of the player's gaming lifestyle.

In Figure 4.4, the vertical axis displays the *monthly hours during the afternoon* (represented by blue bars) and the *monthly hours during the evening* (represented by orange bars). Through accessing the FACEIT API, it's possible to extract information about the start and end time of each match which then will be classified as an afternoon or evening according to the time the match took place. By analyzing this visualization, we can see that the player plays mostly in the evening. Typically, it spends 30 or more hours playing each evening every month. However, it didn't play in August and an explanation for that can be being on vacation at that time since it's a common month people take vacation in. The evolution of the average hours played per month is quite stable, except for the last months of the year when it has reduced the total playtime.

The remaining visualizations and views of the player's gaming behavior will be shown in the GDHelper UI presentation, which is detailed in Section 5.3

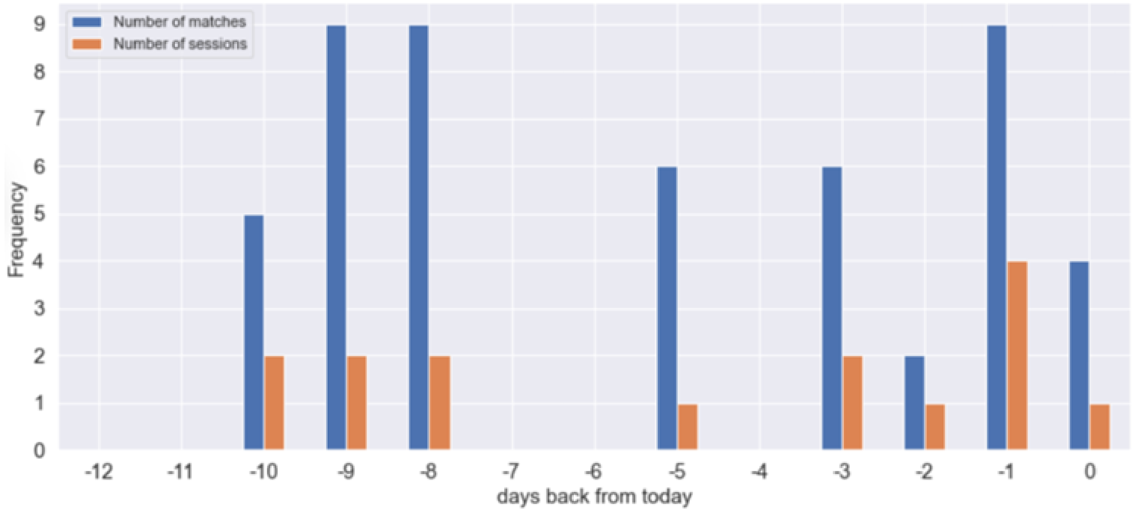


Figure 4.3: Player daily number of matches and sessions. This image is reproduced from Liniers [3]

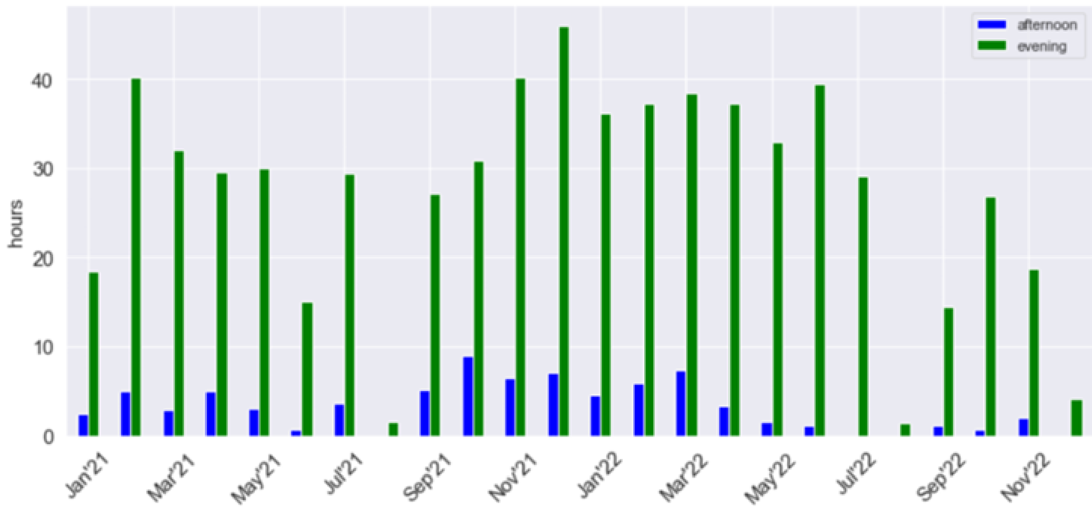


Figure 4.4: Player monthly hours during afternoon and evening. This image is reproduced from Liniers [3]

4.4 System Architecture

In modern software development, designing a robust and scalable architecture is important to the success of any application. Software architecture serves as the blueprint for a software system, outlining the key principles that guide its design and development [93]. It defines the overall structure, organization, and relationships between components, as well as the levels of abstraction and other essential aspects of the system. By establishing a clear architecture, developers can set project goals, create a roadmap for development, and ensure that the final product meets the required standards and user needs [93].

This section presents the architectural design of our application, which integrates various technologies to achieve consistent data collection and processing for Counter-Strike 2 players. By leveraging Next.js [56] for its UI and API capabilities, NextAuth.js [94] for secure authentication, MongoDB for efficient data storage, and the FACEIT API [46] for detailed player statistics, the system is structured to provide convenient insights and analytics. The architecture not only supports the core functionalities required by users but also ensures scalability and maintainability as the application evolves.

We will explore the interactions between these components, the flow of data, and the strategies employed to maintain security and efficiency. Figure 4.5 provides a visual representation of the architecture, illustrating how each component connects and collaborates to form a cohesive system.

4.4.1 Components and Interactions

The architecture comprises several key components that interact to provide a smooth experience for users and efficient data processing:

- **Next.js UI:** The User Interface (UI) is responsible for the presentation layer, providing an interactive and responsive experience for users. It communicates with the API layer to request data and trigger actions, such as retrieving player statistics or initiating authentication processes.

- **API Layer:** The API acts as the foundation of the application, handling business logic and serving as an intermediary between the UI, database, and external services. It processes requests from the UI, interacts with the MongoDB database to store and retrieve data, and integrates with the FACEIT API to obtain player telemetry data.

- **NextAuth.js:** This component manages user authentication, ensuring secure access to the application. The UI interacts with NextAuth.js to handle user login and session management, allowing the API to receive authenticated user information for personalized data access and operations.

- **MongoDB:** MongoDB serves as the database layer, storing user profiles and player data. It maintains collections such as `User` and `Player`, supporting advanced searches and efficient data retrieval. The API interfaces with MongoDB to update and fetch data as required.

- **FACEIT API:** The FACEIT API provides crucial data on player performance and match events. The API layer communicates with FACEIT to fetch real-time statistics and historical data, which is then processed and stored in MongoDB for analysis.

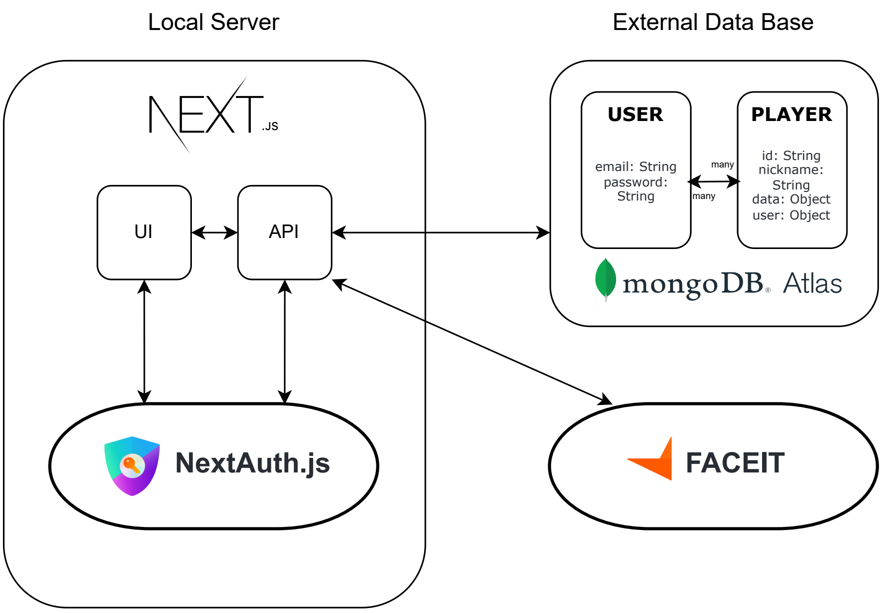


Figure 4.5: GDHelper Architecture

4.4.2 Data Flow and Security

The architecture ensures secure and efficient data flow throughout the system:

- **UI to API communication:** The UI sends requests to the API for data operations and user interactions. This setup allows for a clear separation of concerns, where the UI focuses on presentation and the API handles business logic.

- **Authentication management:** NextAuth.js facilitates secure authentication by managing login sessions and providing user identity to the API. This approach safeguards sensitive data and ensures that only authenticated users can access protected resources.

- **Data Storage and retrieval:** MongoDB efficiently handles data persistence for the application by supporting the storage and retrieval of user and player data. The data is organized into two key collections: User and Player, each designed to hold specific types of information relevant to the application's functionality.

  1. User collection:
     - email (String): This field stores the user's email address, which serves as a unique identifier for each user account.
     - password (String): This field stores the encrypted password associated with the user's account for secure authentication.
  2. Player Collection:

  - id (String): This field holds the unique identifier for a player, provided by the FACEIT API.
  - nickname (String): This field stores the player's nickname or username used in the game.
  - data (Object): This is an object that contains general player data, such as the nickname, ranks and total games.
  - user (Object): This field links the player to one or more user accounts. It stores information about the users who are tracking or monitoring this player. This allows the application to establish relationships between users and players.

- **External data integration:** The FACEIT API integration enables the application to leverage external data sources for enhanced functionality. The API layer securely communicates with FACEIT, retrieving data that is crucial for player performance analysis and feature implementation.

This architectural design provides a robust framework for the application, ensuring scalability, security and maintainability. By leveraging modern technologies such as Next.js, MongoDB, and NextAuth.js, the system effectively manages complex interactions and data flows, delivering valuable insights into player behavior and performance.

Some of the aspects mentioned here, including detailed data storage procedures, authentication mechanisms, and data integration, will be further detailed in Chapter 5.

4.5 Summary

This chapter provided a comprehensive analysis and design of GDHelper, a web application developed to assist in the diagnosis of Internet Gaming Disorder (IGD). GDHelper builds upon previously created visualizations, integrating telemetry data with psychological assessments to offer health professionals an interactive platform for monitoring and analyzing gaming behavior. This approach aims to overcome the limitations of static visualizations, providing a more dynamic and insightful tool.

The development of GDHelper followed an Iterative and Evolutionary Development methodology and a user-centered design, characterized by regular weekly or bi-weekly meetings, focused objectives for each iteration, and continuous refinement based on stakeholder feedback. This approach ensured adaptability and active involvement of all parties throughout the development process.

The chapter detailed the gaming data life cycle, containing attribute definition, data acquisition, preprocessing, analysis, visualization, and reporting. FACEIT was selected as the primary data source due to its comprehensive API, which provides detailed telemetry data for every match played, enabling a thorough analysis of gaming behavior.

To capture relevant aspects of gaming behavior, several metrics were defined. These include monthly and daily hours played, session duration and frequency, and changes in gaming patterns over time. These metrics form the basis for creating visualizations that assist health professionals in understanding and analyzing gaming patterns effectively.

The system architecture of GDHelper integrates various technologies. Next.js is used for both UI and API capabilities, while NextAuth.js provides secure authentication. MongoDB serves as the database for efficient data storage, and the FACEIT API is used to obtain detailed player statistics. This architecture ensures secure data flow, communication between components, and scalability for future enhancements.

Chapter 5

GDHelper Implementation

This chapter details the practical steps taken to develop the GDHelper application to life. It covers the development environment, the design and functionality of the application programming interfaces (APIs), and the user interface (UI). The chapter also addresses the importance of responsiveness and accessibility in the application, ensuring it meets the needs of diverse users. Each section extensively examines the methodologies and technologies employed, highlighting the challenges faced and solutions implemented during development.

5.1 Development Environment

The development environment of GDHelper involves multiple technologies that include the following details:

Programming Language

- **TypeScript [95]:** The primary language used for the entire project, both on the server-side and client-side. TypeScript is a strongly used programming language that builds on JavaScript and adds static typing [95]. Given the project's requirements, TypeScript's unique benefits including type safety, improved code quality, and seamless integration with the JavaScript ecosystem [96], made it the ideal choice for this project.

Frameworks and Libraries

- **Next.js:** A React framework used for building Server-side rendered (SSR) and statically generated web applications that uses the Node.js [57] runtime environment to run server-side code. It provides features like file-based routing, API routes, and automatic code splitting [97]. Behind the scenes, Next.js simplifies the development process by automatically handling and configuring essential tooling for React, including bundling, compilation, and other tasks [97]. This enables developers to concentrate on building their applications, rather than dedicating time to tedious configuration tasks [97]. In the context of our application, Next.js was chosen not only because of the advantages described above but also due to the familiarity with the framework on previous projects.

- **React [48]:** A JavaScript library for building user interfaces, particularly single-page applications where you can create reusable UI components. Since Next.js is built upon React, React components were used for rendering the user interface (UI) of this project [48].

- **Recharts [98]:** A composable charting library built on React components, used for creating data visualizations such as bar charts, line charts, and pie charts [98]. Because one of the main goals of this project is to create visualizations of the data, having a library that enables the creation of several types of charts so we can visually represent data is very important and Recharts facilitate the representation of complex data in a simple and visually appealing way [99].

- **NextAuth.js:** An authentication library for Next.js applications designed to be secure by default and encourage best practices for safeguarding user data [94]. To access the application, a tool for authentication that manages login sessions and provides user identity is needed.

- **Tailwind CSS [100]:** Tailwind CSS is an open-source CSS framework that works exceptionally well with Next.js [101].

- **Shadcn/ui [102]:** Beautifully designed components that join React, Radix UI and Tailwind CSS UI Components [102].

- **Mongoose [103]:** An Object Data Modeling (ODM) library for MongoDB and Node.js, facilitating data modeling and schema definition. It offers a simple and structured approach to modeling the application's data, complete with a range of features that simplify development. With Mongoose, applications can have built-in support for type casting, data validation, query construction, and custom business logic, all available right out of the box [103].

Database

- **MongoDB:** A NoSQL database was employed for storing and managing the project's data, ensuring flexibility and scalability.

- **Document-Oriented Data Model:** MongoDB is a NoSQL database that uses a document-oriented data model. Unlike traditional relational databases that store data in tables with rows and columns, MongoDB stores data in flexible, JSON-like documents [63]. This format is particularly advantageous for handling complex and hierarchical data structures, such as player profiles, game statistics, and in-game events, which are the foundation of this project. The document model allows for nested fields, arrays, and other rich data types, enabling more natural and intuitive data modeling that closely aligns with the structure of the data being analyzed.

- **Scalability and Performance:** MongoDB is designed to scale horizontally, making it an ideal choice for applications that need to handle large volumes of data and high-throughput operations [63]. As the gaming disorder analysis tool needs to process substantial amounts of telemetry data from multiple users over time, MongoDB's ability to distribute data across multiple servers ensures that the database can grow seamlessly with the application's demands. This scalability is critical for maintaining performance as data volumes increase.

- **Schema Flexibility:** One of MongoDB's standout features is its schema flexibility. Unlike traditional relational databases that require predefined schemas, MongoDB allows for dynamic schema design, meaning that the structure of documents can evolve over time [63]. This is particularly useful in an environment where data requirements might change as the project evolves, allowing for quick iterations without the need for complex database migrations.

  This flexibility was extremely helpful in our application because iteration after iteration the project's requirements would get refined or even changed which made the process of changing the structure of data easier.

- **Ease of Integration with JavaScript and Node.js:** MongoDB integrates easily with Node.js, the runtime environment used in this project. Through the use of Mongoose, an Object Data Modeling (ODM) library, MongoDB's integration with JavaScript becomes straightforward, allowing to define schemas, models, and data relationships directly within the application's codebase. This integration enables faster development and reduces the cognitive load on developers, as they can work consistently within the JavaScript ecosystem without needing to switch contexts.

- **Community and Ecosystem:** MongoDB has a large and active community, along with a vast ecosystem of tools and libraries [63]. This support network is valuable for resolving issues, finding best practices, and accessing a wealth of resources that can help simplify development. Additionally, MongoDB's comprehensive documentation and the availability of MongoDB Atlas, a cloud-based managed service, offer further incentives for using this database in both development and production environments.

Development Tools

- **Visual Studio Code [104]:** The Integrated Development Environment (IDE) chosen for writing, editing, and managing the project's codebase. It supports almost every major programming language like JavaScript, TypeScript, CSS, and HTML and has very useful extensions that help with writing and editing code.

- **ESLint [105]:** A static code analysis tool used to identify and correct linting issues in JavaScript, promoting code quality.

- **Prettier:** A code formatter that enforces consistent code style across the entire project.

Version Control

- **Git [106]:** Used for version control, enabling efficient tracking of changes and collaborative development [106]. For the development of the gaming disorder analysis tool, Git was selected as the version control system due to its robustness, flexibility, and widespread adoption in the software development industry.

- **GitHub [107]:** The platform is used for hosting the repository, facilitating code reviews, issue tracking, and collaboration [108]. GitHub is built around Git, the version control system used for this project, offering a smooth integration and an intuitive interface for managing repositories [108]. This close integration allows developers to push, pull, and clone repositories directly from GitHub, making it easy to synchronize changes between local and remote repositories. The platform's web-based interface also provides additional tools for managing branches, viewing commit histories, and merging pull requests, all within a user-friendly environment [107].

Package Management

- **npm (Node Package Manager) [61]:** In the development of the gaming disorder analysis tool, npm was selected as the primary tool for managing project dependencies ensuring that all required packages are easily installed and updated.

  Npm is the default package manager for Node.js and plays a crucial role in the efficient management and organization of the various libraries and tools used throughout the project [109]. It provides access to a vast ecosystem of open-source libraries and modules through its online registry, which is one of the largest in the software development world [109]. For the gaming disorder analysis tool, npm enables the incorporation of essential libraries such as Next.js, Mongoose, and NextAuth.js, among others.

Environment Configuration

- **Local Environment:** The application is currently being developed and tested within a local environment. This setup provides a controlled and accessible environment that allows for efficient testing, debugging, and iteration during the development process. Running the application locally ensures that any changes or new features can be rapidly tested and refined, maintaining consistent behavior throughout development.

  While tools like **Docker** [110] and **Webpack** [111] are commonly used in production environments for containerization and module bundling, respectively, their use is postponed in this phase. The focus is on building and perfecting the application's core functionality within the local environment before considering deployment strategies. This approach allows us to facilitate the development workflow, ensuring that the application is stable and fully functional before it is prepared for further projects.

5.2 Application API

The API of GDHelper is designed to efficiently handle user authentication, data retrieval, and data processing tasks. The API is structured into various routes that serve specific purposes, including user authentication, player data management, and the computation of statistical analyses related to gaming behavior.

5.2.1 Authentication API

The authentication API routes manage user authentication using the NextAuth.js framework, ensuring secure access to the application. These routes include (see Figure 5.1):

- **GET/POST: `/api/auth`** - This route handles the user login and logout processes. The route uses NextAuth.js with support for multiple authentication providers, including GitHub and custom credentials. It ensures that only authenticated users can access the application's core features.
- **Configuration: `authOptions.ts`** - The `authOptions` file configures the authentication providers and callbacks. For instance, it includes a custom credentials provider that checks user credentials against hashed passwords stored in the database, and a GitHub provider for OAuth authentication.

5.2.2 Player Data API

These API routes are dedicated to managing player information within the system. They include operations for retrieving, adding, updating, and deleting player data (see Figure 5.1) that implement the requirements mentioned in section 4.1:

- **POST: `/api/register`** - Handles new user registration by connecting to the database, checking for existing users, hashing passwords, and saving the new user data securely.
- **POST: `/api/players/[playerId]`** - Handles new player's statistics in a MongoDB database. It begins by extracting relevant data from the request payload, establishes a connection to the database, checks if the player already exists, and if so, updates the player's information with the new data. This ensures that the player's statistics are updated with the latest information.
- **GET: `/api/getAllPlayers`** - Retrieves a list of all players stored in the database. This route connects to MongoDB and returns the player data as a JSON response.
- **GET: `/api/getPlayer?player_id={id}`** - Fetches detailed information about a specific player using the player's unique ID.
- **DELETE: `/api/deletePlayer?player_id={id}`** - Deletes a player record from the database based on the provided player ID. It ensures that all associated data is removed, maintaining database integrity.

- **GET:** `/api/getPlayerId?nickname={nickname}` - Retrieves a player's ID from the Faceit API based on the provided nickname. This integration with Faceit allows for ideal data synchronization.

5.2.3 Statistical Analysis API

The statistical analysis routes are crucial for processing and analyzing gaming data. They perform complex computations to generate insights over various time frames (see Figure 5.1):

- **GET:** `/api/generalstats/matchesSessionsLast4Years` - Calculates the number of gaming sessions and matches over the last four years. The route processes historical match data, handling edge cases to ensure accurate session counts.
- **GET:** `/api/generalstats/matchDurationLast4Years` - Computes the average match duration for each year, providing a long-term view of gaming habits.
- **GET:** `/api/generalstats/totalHoursLast4Years` - Computes the total gaming hours for each year, providing a long-term view of gaming habits.
- **GET:** `/api/dayPartStats` - Analyzes match data to determine average hours and match percentages for different times of the day (morning, afternoon, evening) across each year. This data helps users understand their gaming patterns throughout the day.
- **GET:** `/api/hoursTimeOfDay/last7Days` - Retrieves and processes match data for the past 7 days, categorizing playtime by morning, afternoon, and evening. Similar routes are available for the last 15 days and the last 2 years.
- **GET:** `/api/last7Days` - This API route retrieves the player's history data from the Faceit API for the past 7 days and calculates the number of matches and sessions for each day. Similar routes are available for the last 15 days and the last 2 years.
- **GET:** `/api/hoursWeek` - Analyzes the distribution of playtime between midweek and weekends over a specific month, helping users identify trends in their weekly gaming patterns.
- **GET:** `/api/hoursPerYear` - Analyzes the hours played per month in the last 2 years, helping users identify trends by having side to side comparisons of the same month but different years.
- **GET:** `/api/monthlyStats` - Calculates average gaming sessions and hours for midweek and weekends over the course of a year, presenting a comprehensive monthly analysis.
- **GET:** `/api/sessionStats/avgHoursPerSess` - Computes the average hours played per gaming session over several years, offering insights into session lengths and frequency. Similar routes are available for matches instead of hours to have added insights.

- **GET: `/api/sessionStats/avgPerMonth`** - Retrieves the player's history data from the FACEIT API for the past 4 years and calculates the average number of sessions per month for each year. The decision to use a 4-year window was based on discussions with the therapist assisting in the project, who indicated that this period is sufficient to observe meaningful gaming patterns and trends in player behavior. Similar routes are available calculating the average number of sessions but per week instead to have added insights.

Figure 5.1: GDHelper's Simplified API Diagram

The API implements robust error-handling mechanisms to ensure the reliability of operations across all endpoints. Each route includes try-catch blocks to manage exceptions, and appropriate HTTP status codes are returned to the client. Additionally, the API employs data validation and sanitation techniques to prevent unauthorized access and protect user data.

It also integrates smoothly with the Faceit API to enhance the application's functionality. This integration allows for the retrieval of player statistics directly from Faceit, which are then processed and stored in the application's MongoDB database. Routes like `/api/getPlayerId` exemplify this integration, ensuring that data remains consistent and up-to-date.

5.3 Application UI

The User Interface (UI) of the gaming disorder analysis tool is designed to be intuitive and user-friendly, allowing users to navigate through the application's features easily. The UI is divided into three main pages: Playtime Analysis (highlighted in orange in Figure 5.2), Statistics (highlighted in green in Figure 5.2), and Players (highlighted in pink in Figure 5.2). Each page provides specific functionalities aimed at helping users analyze and manage gaming behavior.

5.3.1 Playtime Analysis

Before accessing the Playtime Analysis page, users must first select the player they wish to analyze. Once a player is chosen, the Playtime Analysis page visualizes and analyzes the player's gaming habits over different periods. This page is divided into three key sections: Last 7 days, Last 15 days, and Last 2 years (see Figure 5.2):

Last 7 days:

- This section of the Playtime Analysis displays the user's gaming activity over the past week. It provides bar charts showing the total playtime for each day, allowing users to assess the gamer's daily gaming habits quickly. These visualizations are especially useful for health professionals, who can compare the current week's playtime with previous weeks to understand patterns and prepare for each session. Users can hover over each bar to see exact playtime details, offering a detailed breakdown of games, sessions, or hours.
- It has two tabs that separate the analysis into different perspectives:
  - The first tab, **Matches vs Sessions**, (see Figure 5.2) allows the user to view the number of matches and sessions played over the last 7 days. It also provides key metrics such as the average duration of each gaming session and the percentage change in session duration compared to previous periods.
  - The second tab, **Hourly Distribution**, (see Figure 5.3) displays the total hours played each day, divided by morning, afternoon, and night. It also shows percentage changes in gaming hours compared to the previous week or month, enabling users to track variations in their playtime patterns.
- In addition to the charts, the section includes summary cards displaying total hours, games, sessions and key metrics like percentage changes in gaming duration. Each metric is accompanied by a tooltip, represented by an ”?” icon, which provides users with a brief explanation of the metric when hovered over it. This ensures that users, particularly health professionals, can easily understand the meaning and relevance of each key metric.

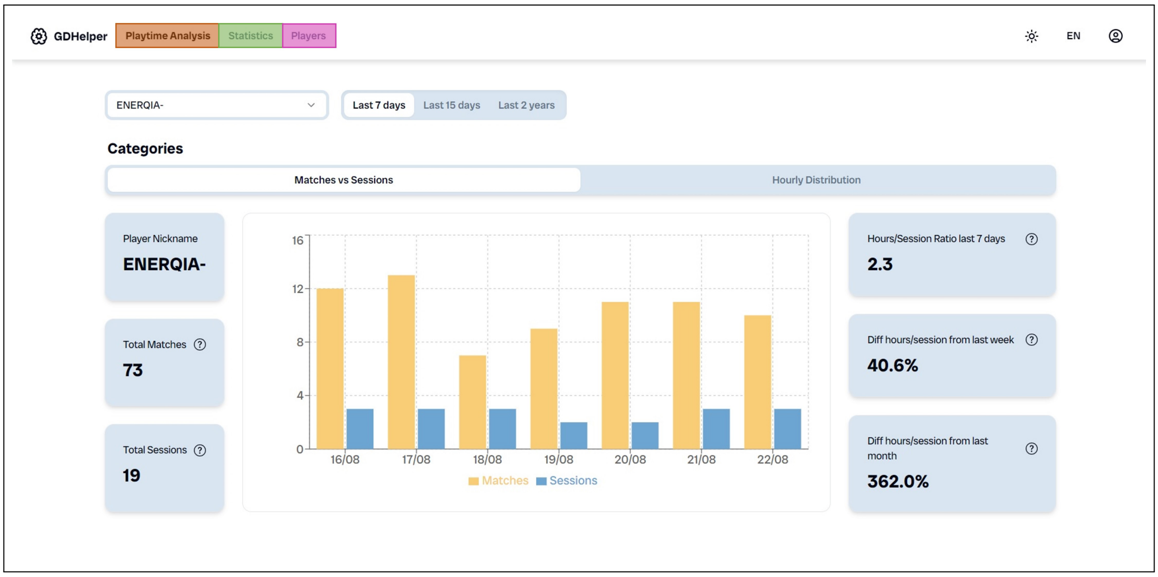


Figure 5.2: Number of sessions and matches in the last 7 days

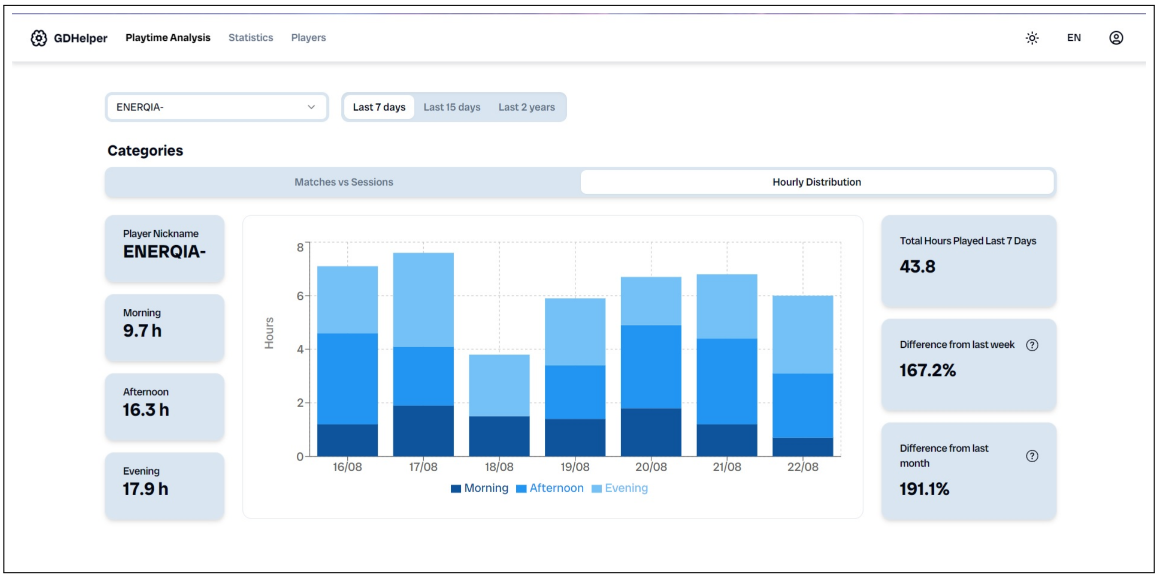


Figure 5.3: Hours distribution in the last 7 days

Last 15 days:

- The "Last 15 days" section expands the analysis to cover the user's gaming activity over the past two weeks, helping to identify longer-term trends or changes in behavior.
- The data is presented in bar charts with interactive elements that allow users to explore specific days in detail.
- This section is also divided into two tabs:
  - The first tab, **Matches vs Sessions**, (see Figure 5.4) displays the total number of matches and sessions played in the last 15 days, with a day-by-day breakdown.

– The second tab, **Hourly Distribution**, (see Figure 5.5) shows the hours played over the last 15 days, segmented by morning, afternoon, and night, providing a clearer picture of daily gaming habits.

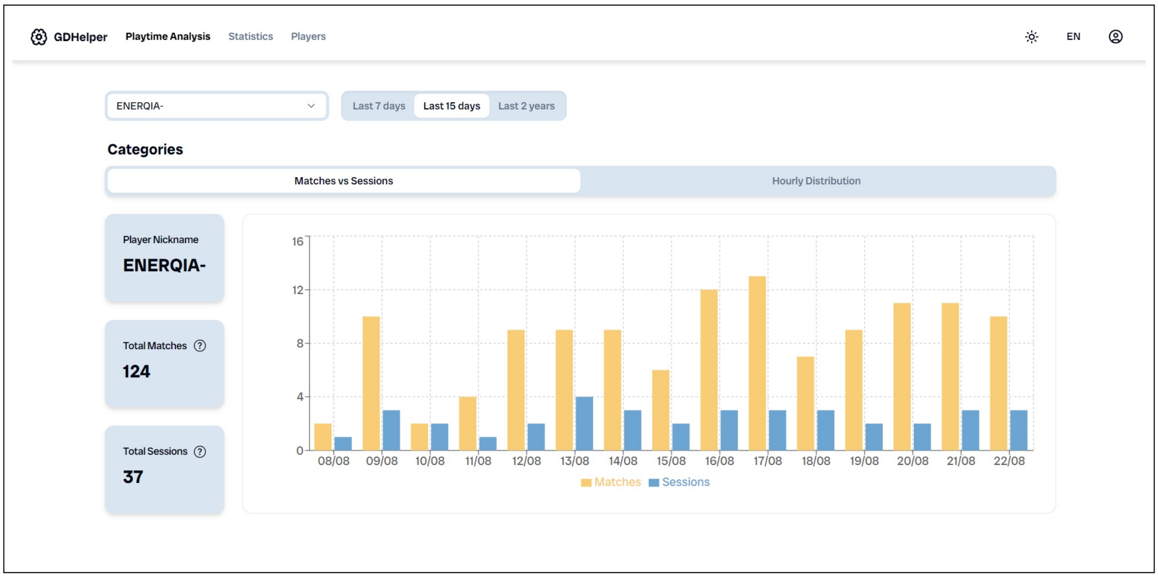


Figure 5.4: Number of sessions and matches in the last 15 days

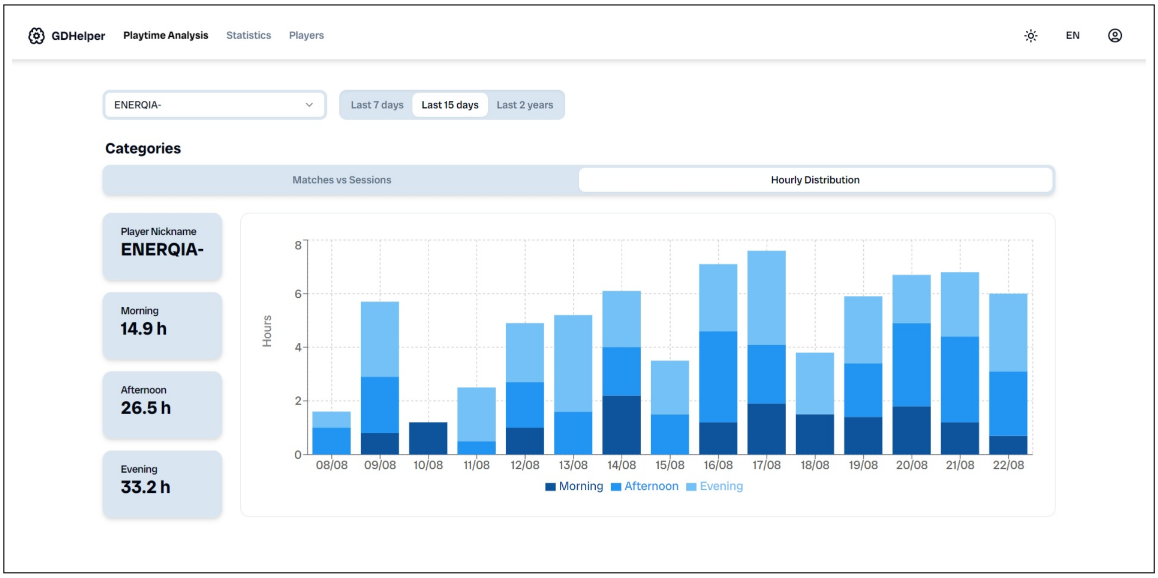


Figure 5.5: Hours distribution in the last 15 days

Last 2 years:

- The "Last 2 years" section offers a comprehensive long-term analysis of the user's gaming behavior, providing insights into patterns and trends over an extended period. This section is particularly valuable for identifying seasonal patterns, shifts in gaming intensity, or other significant changes in behavior over time.

- The analysis is divided into four distinct tabs, each offering a different perspective on the data:
  - **Matches vs sessions:** Similar to the previous sections, this tab provides the total number of matches and sessions played over the last 2 years (see Figure 5.6). It allows users to analyze this data monthly, identifying trends or changes in gaming habits over a longer period.
  - **Hourly distribution:** This tab presents the hours played during the last 2 years, divided by morning, afternoon, and night, with data available monthly (see Figure 5.7). It helps users understand gaming patterns over different times of the day across multiple years.
  - **Weekly distribution:** This tab offers insights into the user's gaming habits during weekends and weekdays over the last 2 years (see Figure 5.8). The data is presented monthly, allowing users to compare how the gaming frequency changes between midweek and weekend over time.
  - **Yearly distribution:** This tab provides a side-by-side comparison of monthly gaming hours over the last two years, helping users see how their gaming activity fluctuated throughout the year (see Figure 5.9). The comparison highlights significant changes or consistent patterns in gaming behavior across different months and years.
- Each tab allows for more extensive analysis by providing both detailed overviews and breakdowns. This comprehensive approach ensures that users can gain a complete understanding of the player's gaming behavior over the last two years.

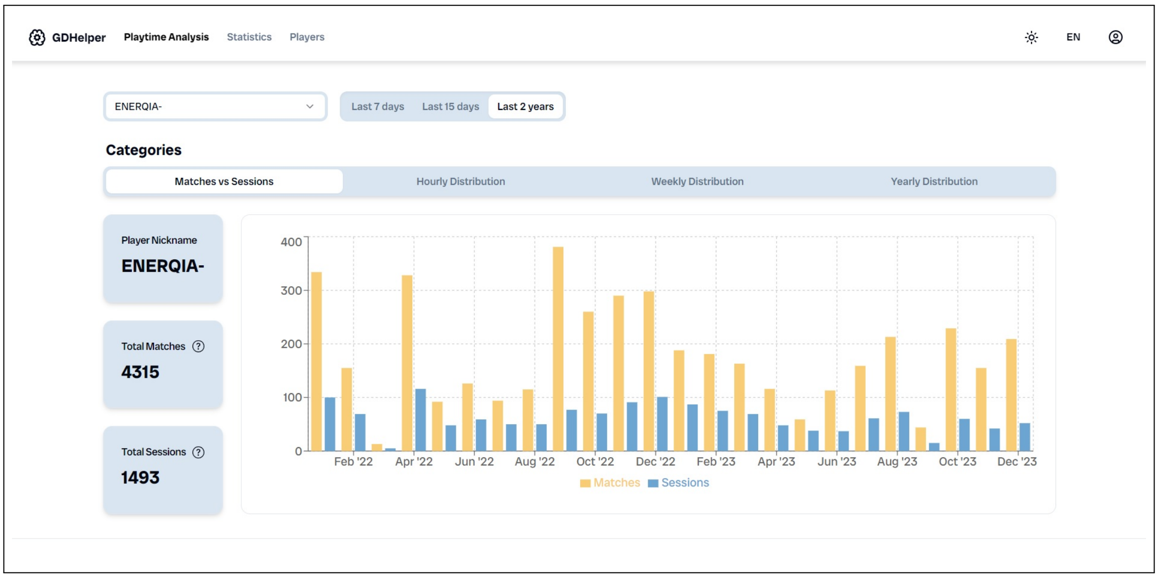


Figure 5.6: Number of sessions and matches in the last 2 years

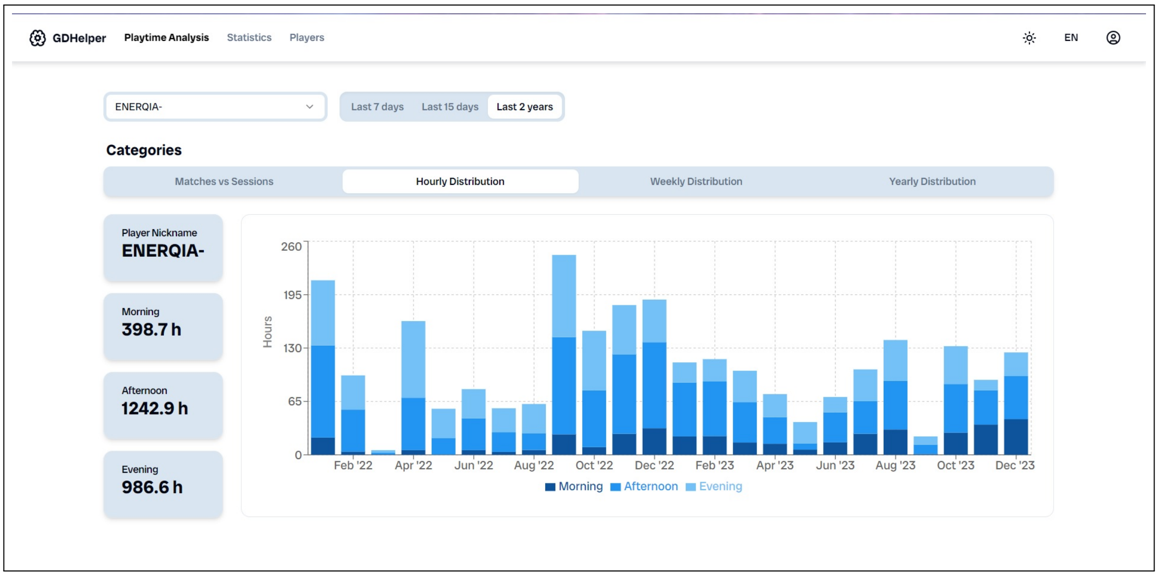


Figure 5.7: Hours distribution in the last 2 years

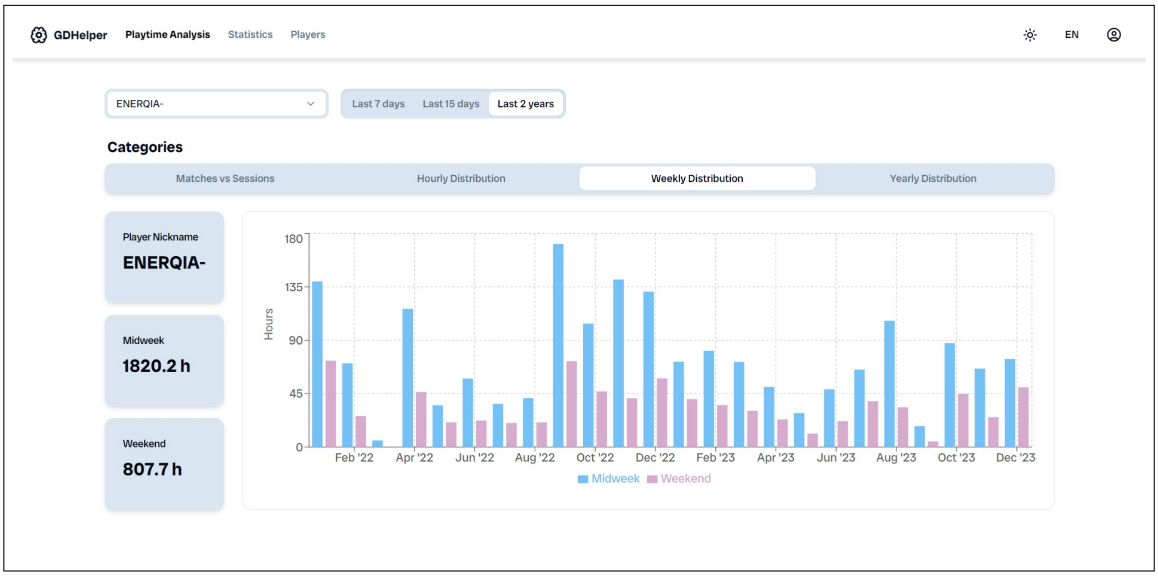


Figure 5.8: Weekly distribution in the last 2 years

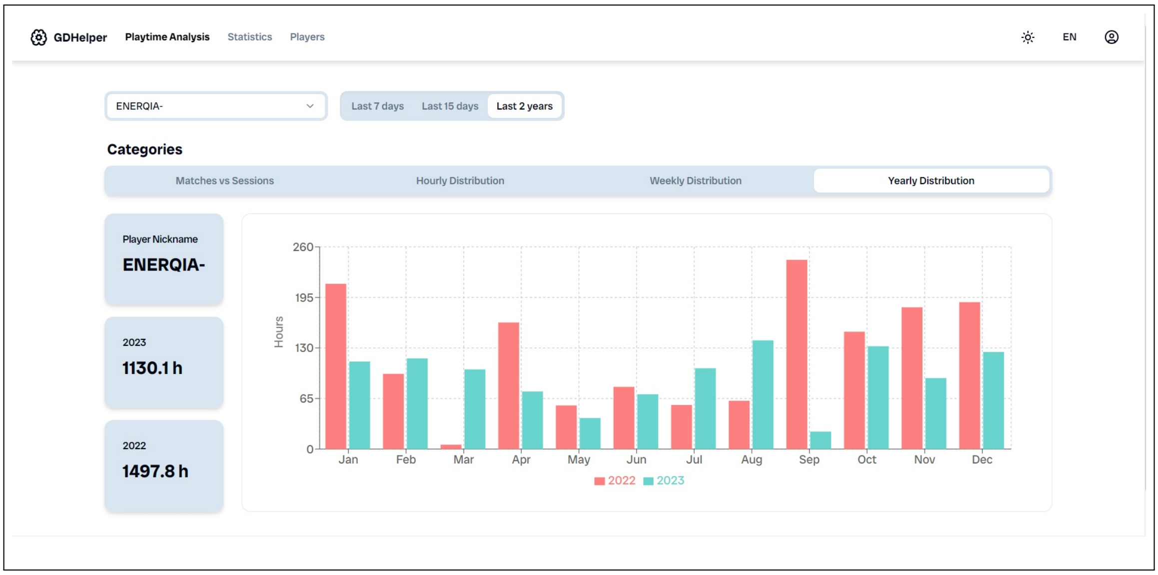


Figure 5.9: Yearly distribution in the last 2 years

5.3.2 Statistics

The Statistics page provides a comprehensive overview of the player's gaming metrics, organized into different categories. The data presented covers the last four years, and is divided into the following sections:

General:

- This section summarizes overall gaming metrics, such as (see Figure 5.10):
  - Number of played matches, Average match duration, Total hours played and Number of sessions played. It serves as a quick snapshot of the user's gaming activity.
- The information is displayed in a table format, with each metric clearly labeled for easy reference and a trend graph to facilitate the visualization of the tendency from year to year.

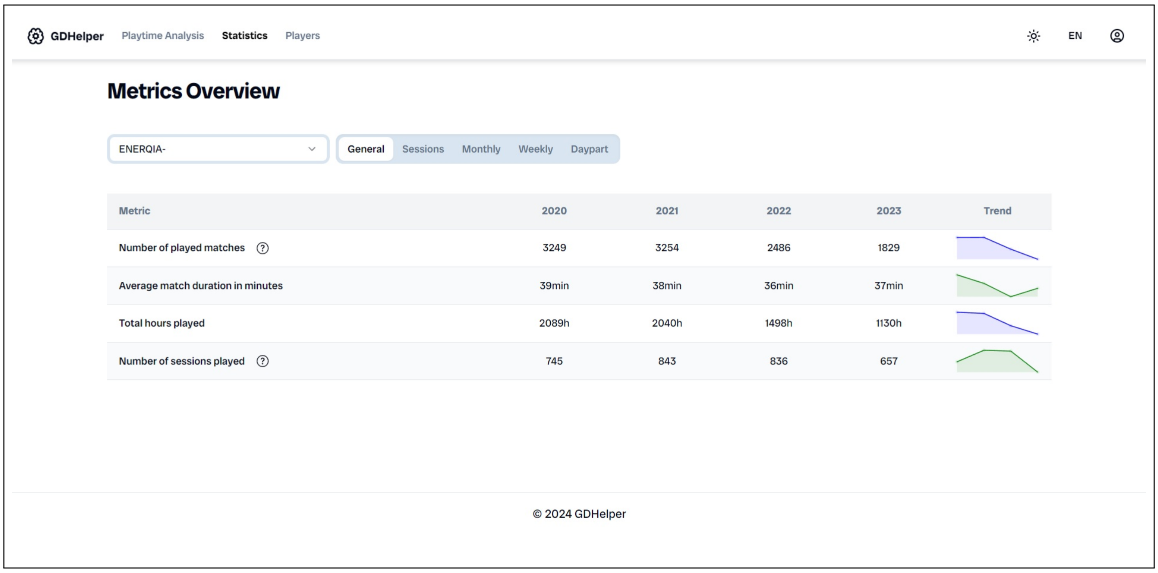


Figure 5.10: General statistics

Sessions:

- Focused on individual gaming sessions, this section breaks down metrics such as (see Figure 5.11):
- Average number of sessions per week, Average number of sessions per month, Average matches per session and Average hours per session. Users can analyze their gaming habits at a more granular level.
- The data is organized in a table, with each metric clearly labeled for easy reference.

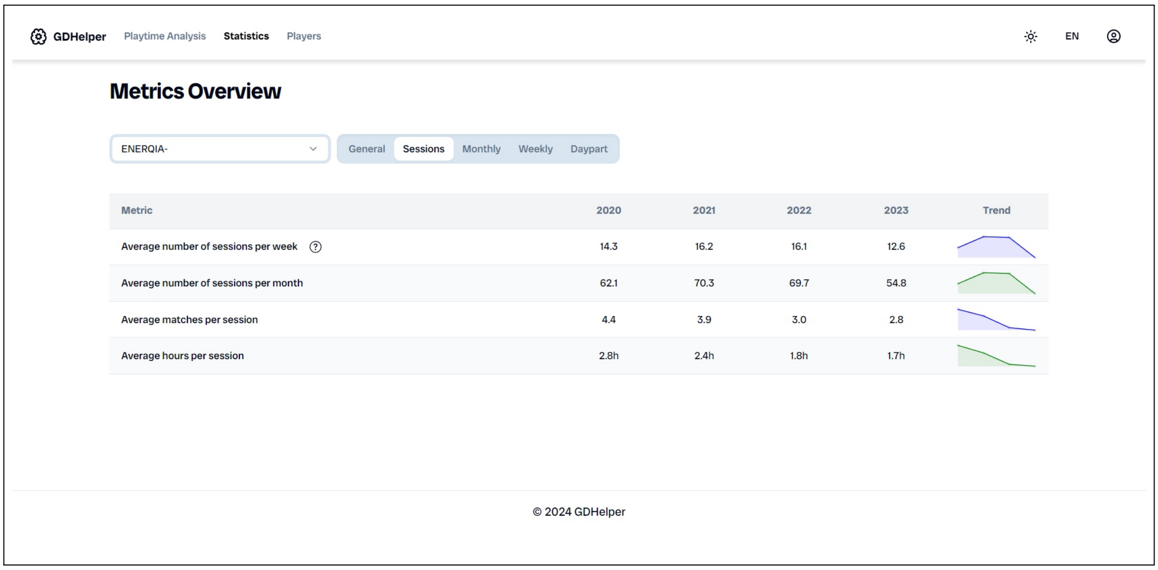
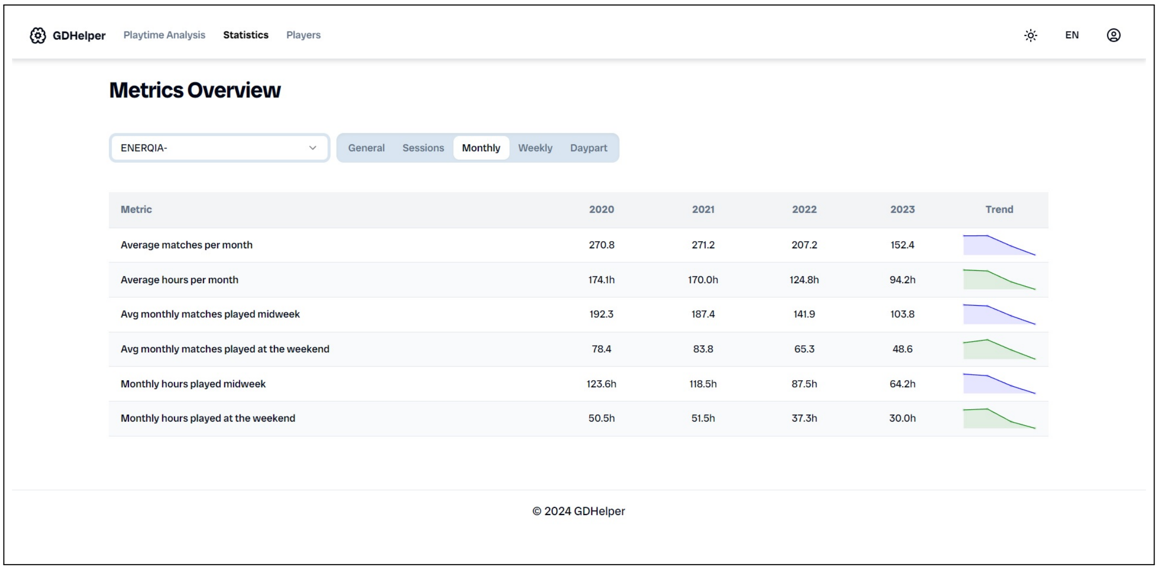


Figure 5.11: Sessions statistics

Monthly:

- This section aggregates data monthly, showing how gaming behavior is monthly in different years. It includes metrics like (see Figure 5.12):
  - Average matches/hours per month, Average monthly matches played midweek/weekend and Monthly hours played midweek/weekend.
- The information is presented in both tabular and graphical formats, allowing for easy comparison across years.



Metric	2020	2021	2022	2023	Trend
Average matches per month	270.8	271.2	207.2	152.4	
Average hours per month	174.1h	170.0h	124.8h	94.2h	
Avg monthly matches played midweek	192.3	187.4	141.9	103.8	
Avg monthly matches played at the weekend	78.4	83.8	65.3	48.6	
Monthly hours played midweek	123.6h	118.5h	87.5h	64.2h	
Monthly hours played at the weekend	50.5h	51.5h	37.3h	30.0h	

Figure 5.12: Monthly statistics

Weekly:

- Here, the data is further refined to show weekly trends. Users can see which how much time or matches the players spent during the weeks and compare their gaming habits across different years.

  It includes metrics like (see Figure 5.13):
  - Average weekly matches, Average weekly hours played and Number of weeks played and Midweek/Weekend match percentage.
- The UI presents this data in a format that is easy to interpret, with clear indicators of weekly highs and lows.

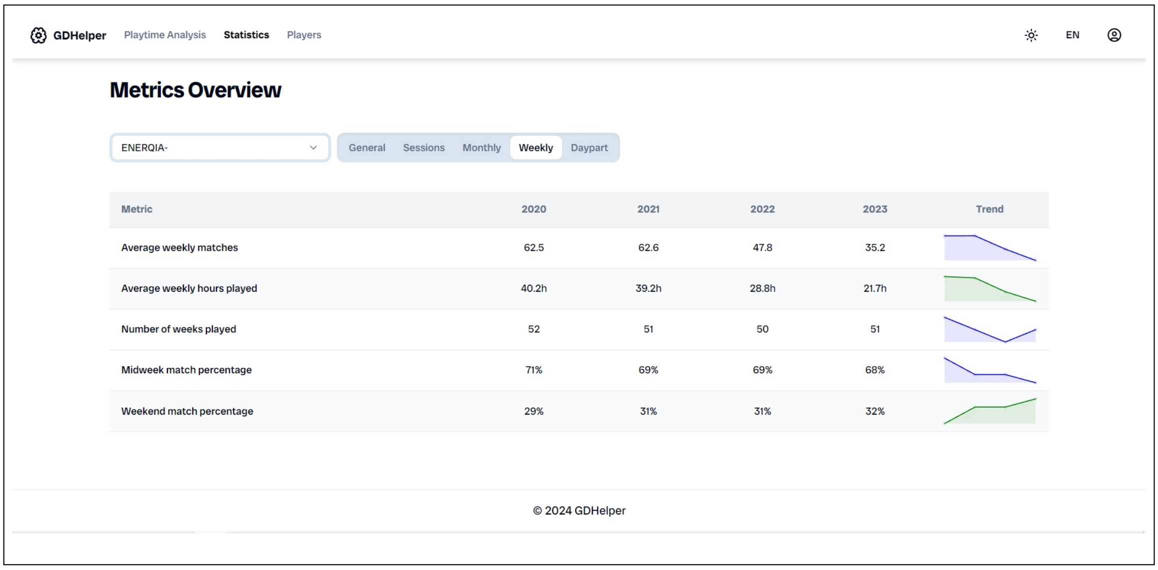


Figure 5.13: Weekly statistics

Daypart:

- This section divides playtime based on different parts of the day (e.g., morning, afternoon, evening, night). It helps users understand when they are most likely to engage in gaming and includes metrics like (see Figure 5.14):
  - Average daily morning/afternoon/night hours and Morning/Afternoon/Night match percentage.
- The data is in a table, offering a quick view of time-of-day gaming preferences.

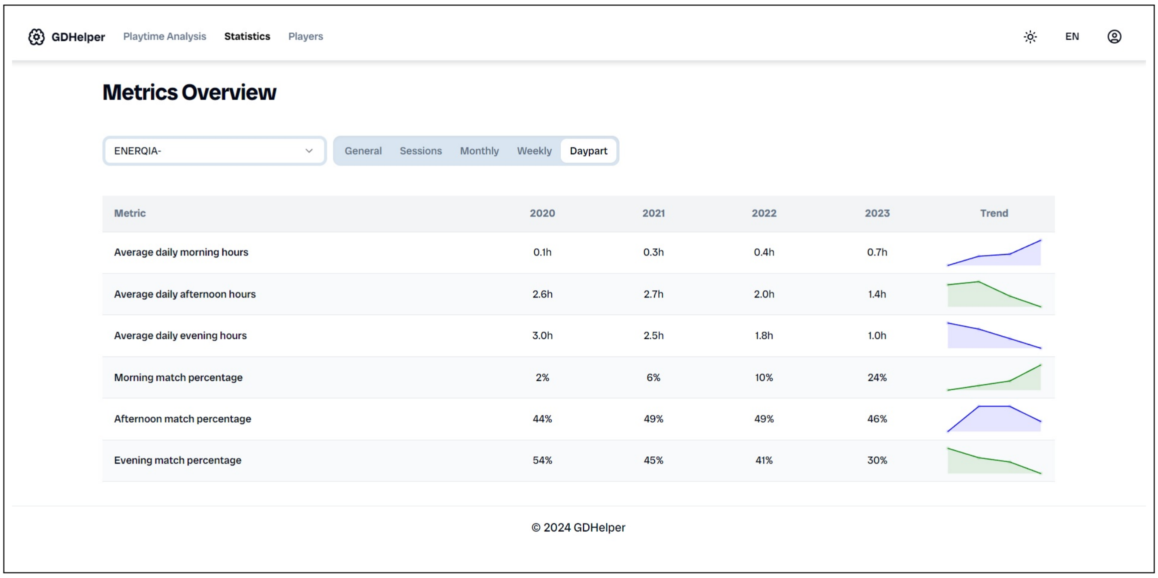


Figure 5.14: Daypart statistics

5.3.3 Players

The Players page is designed for managing player information within the application. Users can add, delete, and update player profiles, allowing them to tailor the analysis to different individuals. Key features include:

Adding Players:

- Users can create new player profiles by entering their FACEIT nickname (see Figure 5.15). This action will store the player in the database alongside their information from the last 4 years.
- The UI provides a simple form with fields for all necessary details, ensuring that adding new players is a straightforward process.

Deleting Players:

- The delete functionality allows users to remove player profiles from the application (see Figure 5.15). This is particularly useful for maintaining an updated list of players.
- A confirmation prompt ensures that users do not accidentally delete important data.

Updating Player Information:

- Users can update existing player profiles to reflect changes such as new gaming preferences or updated personal information (see Figure 5.15). This feature ensures that the analysis remains accurate and relevant over time.
- The UI lets the user choose to update a player of their choice or all the players the user has added before and that is stored on the database.

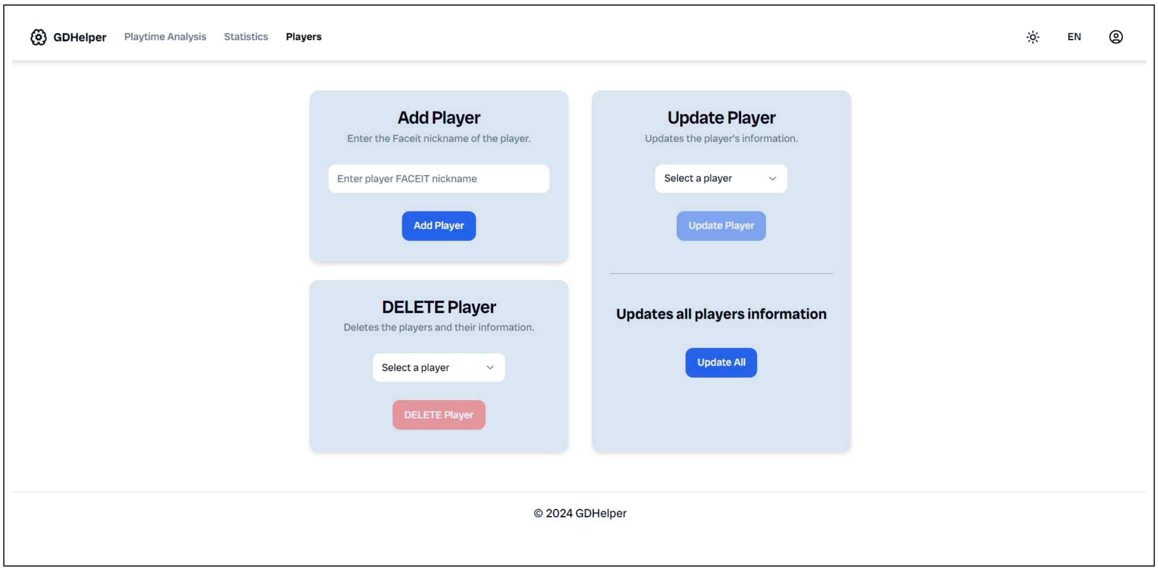


Figure 5.15: Players page

The UI of GDHelper is structured to provide users with a clear and efficient way to interact with the application's features. By organizing the data across Playtime Analysis, Statistics, and Players pages, the tool offers a comprehensive and customizable analysis of gaming behavior. The intuitive design ensures that users can easily navigate the application, access detailed insights, and manage player information effectively.

5.4 Responsiveness Design

The GDHelper application was designed with adaptability in mind to ensure a smooth user experience across a variety of devices, including desktops and tablets. Given the diverse ways in which users, especially healthcare professionals, may access the application, it was crucial to create an interface that functions effectively on devices commonly used in clinical settings, such as tablets, which are frequently used before consultations.

The application was developed using CSS media queries, flexible grid layouts, and scalable vector graphics (SVGs) to ensure that the interface adjusts appropriately to different screen sizes. These techniques allow the layout to respond fluidly to varying screen dimensions, maintaining usability and aesthetics on both larger screens like desktops and mid-sized devices like tablets.

To facilitate this adaptability, the application leverages modern frontend frameworks such as Next.js and React. These frameworks support the development of component-based UIs, which are easier to style and adapt for different screen sizes. Additionally, utility-first CSS frameworks like Tailwind CSS or Shadcn/UI provide built-in support for handling different screen resolutions, ensuring a smooth experience for users.

To verify the application's performance across different devices, a variety of testing methods were employed:

- **Browser developer tools:** Google Chrome's built-in developer tools were extensively used to simulate different device screen sizes, including desktops and tablets. This enabled real-time testing and adjustments to ensure an optimized experience across devices.

- **Manual testing on devices** Where possible, the application was manually tested on actual devices, particularly tablets, to confirm that the interface remains intuitive and user-friendly in environments like healthcare settings where tablets are often used before consultations.

Special attention was given to optimizing performance on tablet devices, where resources such as CPU and bandwidth are more limited, to ensure a smooth experience. Techniques such as lazy loading and minimizing the use of heavy JavaScript libraries were implemented to enhance performance.

By focusing on adaptability and thoroughly testing the application on different devices, including desktops and tablets, GDHelper ensures a consistent and reliable user experience. This commitment is particularly important for healthcare professionals who rely on tablets to access the application in preparation for patient consultations.

5.5 Accessibility

Accessibility was a key consideration in the development of the GDHelper application, ensuring that it is usable by a diverse range of users, including those with disabilities [112]. By implementing accessibility features such as color adaptation for colorblind users and careful attention to text readability, the GDHelper application is designed to be inclusive and usable by a general audience. This commitment to accessibility not only improves the user experience for individuals with disabilities but also improves the overall usability and effectiveness of the application.

To ensure that the visual components of the application are accessible to users with color vision deficiencies, special attention was given to color contrast and selection. For instance, the colors used in bar charts and other visual elements were chosen to be distinguishable by individuals with different types of color blindness, such as protanopia, deuteranopia, and tritanopia. Tools like Coloring for Colorblindness [113] were employed to simulate how the colors would appear to users with various forms of color blindness, allowing for adjustments to be made accordingly.

Text accessibility was prioritized, with a focus on ensuring that the font size, line spacing, and contrast ratios promote readability. While we were mindful of font sizes and line spacing to ensure legibility, we employed flexible font sizing in some places within certain headings, to allow for some adjustment across different screen sizes. While not fully responsive typography, these adjustments improve text readability on a range of devices.

To validate accessibility, a combination of automated tools and manual testing methods were employed. Automated testing using the WAVE [114] tool was important in identifying common accessibility issues such as contrast errors. Although the tool flagged some contrast issues between the background and the color of buttons that will be corrected in future iterations of the application, WAVE helped ensure the majority of the application meets accessibility standards. Furthermore, manual testing involved collaboration with team members affected by color blindness, ensuring that color choices were effective and inclusive.

5.6 Summary

This chapter details the implementation of GDHelper. The development process takes advantage of a tech stack, with TypeScript as the primary language, Next.js as the React framework, and MongoDB as the database, supplemented by libraries like Recharts and NextAuth.js to enhance functionality and design.

The application's API was meticulously structured to manage user authentication, handle player data, and perform statistical analyses. It contains routes for user registration, player data manipulation, and various statistical computations over different periods. The user interface is composed of three main pages: Playtime Analysis, Statistics, and Players. These pages offer visualizations of gaming habits, comprehensive metrics on gaming behavior, and tools for managing player profiles.

Responsiveness was a key focus in the development, ensuring the application functions smoothly across desktops and tablets through responsive design techniques and thorough testing. Accessibility features were also prioritized, with considerations for color vision disabilities and text readability. The implementation process included both automated and manual testing to ensure compliance with accessibility standards.

Overall, the implementation of GDHelper demonstrates a commitment to creating a user-friendly tool capable of analyzing gaming behavior while remaining accessible to a diverse user base, including healthcare professionals in clinical settings. In the next chapter, we present the evaluation conducted to assess whether GDHelper is an effective and valuable tool in the assessment of IGD. This evaluation aims to determine the tool's functional impact, especially in clinical settings, and its ability to aid healthcare professionals in analyzing gaming behavior.

Chapter 6

GDHelper Evaluation

After completing the development of GDHelper, it is critical to evaluate the application to ensure that it meets its intended goals, particularly in terms of usability, performance, and user satisfaction. This chapter outlines the process of evaluating GDHelper, detailing the methods and tools used to test its functionality and effectiveness for its intended users and healthcare professionals. The validation process helps to confirm that GDHelper provides useful data for gaming telemetry analysis and mental health monitoring.

6.1 Methodology

The validation process for GDHelper was conducted throughout its development to ensure that the application met its design goals, specifically targeting usability and effectiveness for healthcare professionals. Regular feedback iterations were established to monitor the application's progress and ensure alignment with user needs. In this section, we outline the methods used to gather feedback, conduct usability tests, and validate the application.

Throughout the development process, weekly meetings with GDHelper members were held, which provided ongoing feedback on the application's development and user interface design. These regular discussions helped identify potential improvements early in the development cycle.

In addition, we collaborated with clinical psychologist Joana Cardoso, who offered valuable insights from a healthcare perspective. Regular meetings with Joana were held to review the application's functionality and ensure that it aligned with the needs of clinical practitioners diagnosing Internet Gaming Disorder. Joana also tested the application during its development, providing real-world feedback on its usability in a clinical setting.

To validate GDHelper's usability and functionality, user testing was conducted with three participants: three healthcare professionals. This group allowed us to gather professional perspectives from the field even if not specialized in treating IGD, helping to ensure the application met its goals from both technical and user-interface standpoints.

User testing was conducted remotely via Zoom, where each participant was guided through a set of tasks designed to evaluate GDHelper's usability. Each session started with an introduction to the study, followed by a series of tasks that participants performed independently, without assistance. The specific tasks and detailed instructions provided to the participants can be found in the Google Forms survey, which is included in the Annexes (see Annex A for the full list of tasks and instructions).

To ensure consistency, all participants received the same form containing the task scenarios. The form was structured as follows:

1. **Informed consent:** Participants were first presented with an Informed Consent Form outlining the study's purpose, confidentiality, and voluntary participation.

2. **Initial questionnaire:** Participants completed a demographic survey and were asked about their prior experience with similar applications.

3. **Task-based usability testing:** Participants were instructed to complete a series of tasks using GDHelper, aimed at testing specific functionalities of the application.

4. **Post-task questions [115]:** After each task, participants were asked questions to evaluate their experience with that task, focusing on ease of use, understanding, and navigation within the application. The first question evaluated whether participants understood the information displayed on the screen while the second measured their satisfaction with the ease of completing the task, rated on a Likert scale from 1 (strongly disagree) to 5 (strongly agree).

5. **Final questionnaire (SUS):** At the end of the session, participants completed the System Usability Scale (SUS) questionnaire [116, 117] to provide an overall rating of the application's usability.

The tasks provided to participants were designed to simulate real-life use cases that healthcare professionals might encounter when using GDHelper. Each task focused on a key functionality of the application, such as logging into the system, retrieving gameplay data, and analyzing patient behavior. The specific tasks were as follows (see Annex A for the full list of tasks and instructions):

- **Task 1: Login and retrieve patient data -** Participants were asked to log in to the application using a test account and retrieve information about a specific patient, including gameplay hours and changes in playtime compared to the previous week.

- **Task 2: Analyze gameplay patterns -** The task involved analyzing the patient's gaming behavior over the past two years, identifying peak playing times, and determining whether the patient played more during the week or on weekends.

- **Task 3: Register a new patient -** Participants were instructed to register a new patient in the system and analyze the patient's gameplay statistics over the past four years, focusing on changes in gaming habits.

- **Task 4: Analyze gameplay intensity -** This task required participants to assess the intensity of gameplay for a new patient by analyzing the number of games played in the last 15 days.

- **Task 5: Delete patient data-** Participants were asked to delete a patient's data from the application, simulating a scenario where a patient has completed treatment and is no longer being monitored.

- **Task 6: Update patient information -** In a follow-up session, participants were instructed to update patient information and analyze recent gameplay data to detect changes in the number of games played and session duration over the past week.

Participants were asked a series of specific questions for each task to assess their experience. The number of questions varied by task, with tasks 1 to 3 including three questions each, task 4 including two questions, task 5 one question, and task 6 two questions. These questions focused on the clarity of instructions, ease of task completion, and any difficulties encountered during navigation.

Finally, after completing all tasks, participants filled out the SUS questionnaire to evaluate the system's usability. The SUS is a widely used tool for measuring usability, using a 5-point Likert scale to assess ease of use, efficiency, and overall satisfaction with the application.

Data from the usability tests were collected through the Google Forms [118] used to define the tasks and SUS questionnaire. The qualitative feedback from participants was carefully analyzed to identify common pain points, areas of confusion, and suggestions for improvement.

The SUS scores were averaged to provide a quantitative measure of GDHelper's usability, while the qualitative feedback was categorized into usability, interface design, and data visualization themes. This allowed us to identify specific areas for refinement and improve the application accordingly.

To improve the efficiency of our data collection process, we opted to use Google Forms as the primary platform for conducting our usability study. This choice offers several significant advantages that align with our research objectives. Google Forms allows us to easily share study materials with participants worldwide, ensuring accessibility [118]. It also simplifies the process of collecting and organizing responses, making data management more efficient.

The first step involved the submission of the user studies proposal to the Institutional Review Board, the Ethics Committee of Faculdade de Ciências da Universidade de Lisboa (CEC – Comissão de Ética de Ciências), which granted approval and provided recommendations on the ethical dimensions of the research proposal involving human beings.

The data collected from participants, including players and health professionals, and its processing will be carried out in accordance with the General Data Protection Regulation (GDPR) (EU 2016/679 of the European Parliament and of the Council), concerning the protection of individuals regarding the processing of personal data and the free movement of such data.

6.2 Results

As part of our study, we collected various demographic variables from participants, including three health professionals:

- Country - The study involved participants from Portugal (100%).
- Gender - The study revealed a significant gender disparity, with female participants making up 67% and males comprising only 33%.
- Age - Participants' age groups ranged from 25 to 44 years old, with the '25-34' group exhibiting the highest representation at 67%, followed by '35-44' at 33%.
- Education level - Participants had similar educational backgrounds, with 33% holding master's degrees and 67% with bachelor's degrees.
- Occupation - The participants were mostly Clinical Psychologists and University Professors.
- Experience - The participants reported having prior experience with web applications.
- Usage Frequency - 33.3% of participants reported very frequent use (5 on a scale of 1-5) of web applications for monitoring health or lifestyle, while another 33.3% reported frequent use (4 on the scale). The remaining 33.3% indicated infrequent use (1 on the scale).

Task completion and post-task results

An analysis of task completion and post-task responses collected through the usability test provides valuable insights into the ease with which users interacted with the GDHelper application. These findings help evaluate the system's usability and users' overall satisfaction. The data presented below reflect the tasks completed by the participants, including correct and incorrect attempts and task success rate, as well as their subjective ratings based on task difficulty and time required to complete each task.

Each task consists of a brief instruction with what the user is supposed to do alongside the actual questions. An example could be "Imagine you're a therapist and you're planning on checking how many hours your patient played in the last seven days" and the questions would be "how many hours did he play?" and "how many sessions he played?".

As shown in Table 6.1, participants demonstrated varying levels of success across tasks, with success rates ranging from 66.67% to 100%. The average task completion success rate was approximately 79.35%, which suggests that participants generally completed the tasks effectively. However, certain tasks proved to be more difficult than others, likely due to the complexity of the data being retrieved or specific cognitive challenges in interpreting task requirements.

Notably, tasks involving straightforward actions (e.g., patient registration and deletion) had the highest success rates (100%), while tasks requiring a more intensive analysis or data extraction (1, 4, and 6) resulted in lower success rates (66.67%). These lower success rates were due in part to specific questions that participants found challenging:

Task #	Correct Answer	Incorrect	Task success rate (%)
1 - Login and retrieve data	6	3	66.67
2 - Analyze gameplay patterns	7	2	77.78
3 - Register a new patient	9	0	100.00
4 - Analyze gameplay intensity	4	2	66.67
5 - Delete patient data	3	0	100.00
6 - Update patient data	4	2	66.67

Table 6.1: Task completion results for each task

- In **Task 1**, participants struggled with the question, "How many hours did the player play two days ago during the night?" One participant provided the hours for the morning instead of the evening, and another added hours from two days instead of providing the correct total.
- In **Task 4**, the question "How many sessions did the player play ten days ago?" caused confusion. Two participants provided incorrect answers by misinterpreting the time frame.
- In **Task 6**, the question "How many matches did the player play yesterday?" was answered incorrectly by one participant, reflecting a misunderstanding of the question.

In addition to these objective measures, participants' subjective perceptions of the system's usability were captured through two post-task questions following each task. The first question evaluated whether participants understood the information displayed on the screen while the second measured their satisfaction with the ease of completing the task, rated on a Likert scale from 1 (strongly disagree) to 5 (strongly agree).

The post-task feedback provided further insights into the user experience. Across all tasks:

- **Task understanding:** Most participants consistently reported understanding the information displayed, although a few struggled with more complex tasks (like 1, 4 and 6), particularly those involving multi-step data retrieval and analysis.
- **Ease of task completion:** Participants generally rated the ease of task completion positively, with most scores falling between 3 and 5 on the Likert scale. Tasks that involved more cognitive load or ambiguity in data interpretation (like 1, 4 and 6) saw slightly lower ratings, suggesting that improvements to clarity and interface design could enhance usability for these types of tasks.

Beyond task completion, we further evaluated the system's usability through participants' responses to the Post-task Questionnaire, which included a Likert scale rating for two key aspects of each task: perceived ease of use. The post-task results are summarized in Table 6.2, which provides the mean, median, and standard deviation for participants' ratings.

The post-task questionnaire results reveal that participants generally found the tasks easy to complete. The mean ease-of-use scores ranged from 3.33 to 4.67, with most tasks achieving a

Task #	Ease of Use (Mean)	Ease of Use (Median)	Std. Deviation
1 - Login and retrieve data	3.67	3.00	0.58
2 - Analyze gameplay patterns	3.67	4.00	0.58
3 - Register a new patient	3.67	4.00	0.58
4 - Analyze gameplay intensity	3.33	3.00	0.58
5 - Delete patient data	4.67	5.00	0.58
6 - Update patient data	4.00	4.00	0.00

Table 6.2: Results for post-task feedback

median score of 4, indicating general satisfaction. The higher standard deviation observed in Task 4 suggests some variability in user experience, potentially due to the complexity of the task. This variation aligns with the task's lower success rate, indicating that participants may have found it more challenging. In contrast, tasks such as Task 5, which involved simpler actions, saw the highest ease-of-use ratings and lowest variability.

The feedback indicated that while users found the system typically easy to use, some aspects, particularly tasks involving data analysis or detailed interpretations (like 1, 4 and 6), introduced some usability challenges. A common theme was the cognitive burden associated with interpreting data spread across multiple screens (for example, changing from the last 7 days screen to the last 15 days screen), which may have contributed to the occasional errors observed in task completion. However, overall satisfaction levels were high, reflecting the participants' comfort with the system.

In summary, the data reflects a generally positive user experience, with high completion rates and favorable subjective feedback. However, it also highlights areas for improvement, particularly in tasks requiring complex data interpretation. Future iterations of the application could benefit from better data presentation and improved clarity to further improve usability and reduce cognitive load on users.

User feedback and observations

After reviewing the quantitative results, we further examined the qualitative feedback provided by the participants. This section summarizes their impressions and observations gathered during the usability testing sessions, focusing on the strengths and areas of confusion they encountered while using the system.

Overall, the application itself received positive feedback. Most participants described it as easy to use and intuitive, with minimal dissatisfaction while completing tasks. The simplicity of the user interface allowed users to navigate smoothly through the features and accomplish the required actions without significant issues. Several participants highlighted that they found the system ”very easy to use” and that it helped them efficiently complete their tasks. This aligns well with the consistently high task success rates observed in the quantitative analysis.

However, participants provided mixed feedback regarding the post-task questionnaire. A common theme that emerged was confusion regarding the structure and clarity of some questions in the questionnaire. Several users mentioned that the phrasing of certain questions was unclear, leading to misunderstandings in their responses. Additionally, the form's layout and instructions could have been more user-friendly, as some participants expressed difficulty navigating through the questionnaire. This suggests that while the application itself was perceived positively, improvements could be made to improve the clarity and usability of the forms used for gathering feedback.

Despite these challenges, users claimed that their overall experience with the application was positive. They felt that, with slight refinements, future interactions would be even better. As one participant noted, "If I were to use the application again, the interaction would be much easier."

In more detail, we present the feedback from users, for each task:

- **Task 01** - For this task, participants had to check their patient's playtime over the last week and compare it with the previous week's data. The task was fairly simple, but some confusion emerged with understanding the specific periods. One participant misinterpreted the request, incorrectly summing playtime over multiple days instead of identifying specific hours played two days ago. The overall feedback for this task suggested that users found the interface intuitive, though they requested clearer instructions.

- **Task 02** - This task required users to identify the time of day a patient played the most and whether they played more during the week or on weekends. While most users completed the task, two participants struggled with interpreting the time-of-day data, resulting in incorrect answers. Nonetheless, the task interface was generally well-received, with participants commenting that the application provided a clear overview of the relevant data, making it easy to navigate and filter information. The confusion originated more from the interpretation of the task itself than from the application.

- **Task 03** - This task, which involved comparing patient playtime over the past few years, received positive feedback. All participants completed this task, reporting that the application made it easy to extract the required information. Users found the process of analyzing historical data intuitive and straightforward. They particularly appreciated the clear presentation of statistics, enabling them to complete the task efficiently without needing much assistance or clarification.

- **Task 04** - In this task, participants were tasked with reviewing the number of games played in the past 15 days. While users could easily identify the total number of games played, there was some confusion around identifying specific sessions played on certain days. Two participants struggled with finding the correct session data, which they attributed to the complexity of filtering by specific days. This feedback suggests that while the overall presentation was clear, adding more intuitive date-filtering options could enhance the user experience.

- **Task 05** - This task, which required participants to delete a patient from their list, was executed smoothly by all users. They found the interface for managing patients to be very user-friendly and appreciated how easy it was to complete administrative tasks such as deleting or updating patient records. The feedback for this task was extremely positive, with users expressing satisfaction with the simplicity and effectiveness of the application's patient management features.

- **Task 06** - This task involved updating a patient's information to reflect the most recent game session data. Although the task was generally well received, one participant misunderstood the request and failed to record the correct game session data. This was attributed to the phrasing of the task rather than any limitations in the application itself. Despite this, the majority of participants praised the application's ability to present up-to-date information clearly and concisely.

- **Post-task feedback** - Across all tasks, participants provided feedback on whether they understood the information displayed on the screen and their overall satisfaction with completing the tasks. Most users felt confident navigating the application and were satisfied with the task completion process. The simplicity of the interface and the ease of data access were recurring themes in the feedback. However, when asked if they understood all the information on certain screens, a few participants noted areas of confusion, particularly when navigating between different application sections, such as switching between tabs.

In conclusion, while the application's core functionality was praised for its ease of use, addressing the issues with the questionnaire's complexity would further improve the user experience. With clearer instructions and improved question phrasing, the data collection process can be improved and reduce participant confusion. These insights are valuable for future iterations of both the application and the feedback process.

Result for SUS questionnaire

The System Usability Scale (SUS) questionnaire provided insights into users' general impressions of the GDHelper application. Participants were asked to rate several statements related to usability on a scale from 1 (strongly disagree) to 5 (strongly agree). The results highlight areas of strength as well as potential areas for improvement.

Overall, participants rated the application positively in terms of usability. For example, when asked whether they would use the application frequently, two participants rated the statement a ”3” and one a ”4,” indicating a generally favorable reception. However, there was some variation in responses about the application's complexity. One participant strongly disagreed (rating of 1) that the application was unnecessarily complex, while another gave a more neutral score (2), and one gave a higher score of ”4,” suggesting a need for simplification in some areas.

In terms of ease of use, most participants rated the system favorably. Two participants rated GDHelper's ease of use a ”3,” while one rated it a ”4.” This reflects general satisfaction but also suggests that there may be small usability challenges. When asked if they would need technical assistance to use the app, two participants strongly disagreed (rating of 1), while one was more neutral (3), suggesting that most users found the system intuitive and user-friendly.

The integration of functions within the application received a favorable response, with two participants giving a ”4” and one a ”3.” However, responses to the question of whether the application had inconsistencies showed more variation, with ratings of ”1,” ”2,” and ”3.” This suggests that while most users did not experience significant inconsistencies, there may be areas where functionality can be facilitated.

Participants were also asked how quickly they believed users could learn to use the application. Two participants strongly agreed (5), and one agreed (4), indicating that the learning curve for GDHelper is minimal, and users can become proficient quickly. Confidence in using the application was also high, with two participants rating it a ”5” and one rating it a ”3.”

Lastly, all participants strongly disagreed (1) with the statement that they needed to learn several new things before using the application, indicating that previous knowledge or experience with similar systems was not necessary to navigate through GDHelper.

In summary, the calculated SUS score for GDHelper was 75.83 out of 100, which is above the average SUS score of 68, indicating above-average usability. This quantitative score aligns with the qualitative feedback, as the SUS results reflect a generally positive experience with the application. Users found it relatively easy to use, requiring little technical assistance, and felt confident navigating its features. Some minor areas of complexity and inconsistency were noted, which suggests options for improvement in future iterations of the application. However, overall, the system received a high rate for usability, with participants appreciating its ease of learning.

6.3 Discussion

The usability study of the GDHelper platform provides valuable insights into its effectiveness as a tool for monitoring and analyzing gaming behavior in a clinical context. This section discusses the key findings, their implications, and areas for future development.

Demographic factors

All participants in the study were clinical psychologists and/or university professors from Portugal. Though none of them work specifically with young people experiencing IGD, they are still experts in clinical practice. Their experience and expertise may have contributed to their ability to navigate the platform. It is important to emphasize, however, that future studies should include a more varied group of participants, both in terms of gender and age, to ensure the platform’s usability across a broader demographic. A study with more users, particularly those with experience in diagnosing and treating IGD, could offer additional insights.

Task completion and Post-task results

While the overall usability was high, there was notable variation in task success rates, ranging from 66.67% to 100%. Tasks involving straightforward actions like patient registration and deletion, were completed successfully by all participants, reflecting the system's ease of use for basic operations. The System Usability Scale results reinforce this finding, where participants expressed confidence in navigating basic tasks, with two out of three participants strongly agreeing that they felt confident using the system (rating of 5).

However, more complex tasks, particularly those requiring time-based data analysis (Tasks 1, 4, and 6), had lower success rates. For example, in Task 1 ("How many hours did the player play two days ago during the night?"), users misinterpreted the time frame or added data incorrectly. Similarly, Task 4 ("How many sessions did the player play ten days ago?") confused participants, with misinterpretation of the time frame leading to errors.

This finding is also supported by the SUS responses. While participants generally rated the system's integration of functions favorably, with two giving a score of "4" and one a "3," there was more variation in their perceptions of complexity. One participant gave the system a higher complexity rating of "4," which correlates with the lower success rates in more cognitively demanding tasks. Therefore, future iterations of GDHelper should simplify the presentation of time-based data, possibly by providing clearer visual indicators or step-by-step prompts for data extraction tasks.

It is worth noting that these participants, while clinical experts, may not be as familiar with gaming terminology or patterns of gaming behavior. This lack of familiarity might have contributed to the difficulty some users experienced with these tasks. Even though the application had tooltips to explain the metrics displayed, future iterations of GDHelper could explore adding more contextual help or clearer explanations for tasks involving data analysis to ensure that all users, regardless of their experience with gaming terms, can navigate the platform effectively.

User feedback and SUS questionnaire

The user interface was positively received, which is reflected in the overall SUS score of 75.83 out of 100 - a result that places GDHelper above the average usability score of 68. This above-average performance is particularly evident in aspects related to learnability, with participants praising the intuitive design and smooth navigation. Two participants strongly agreed that users could learn to use the application quickly (rating of 5), reflecting a short learning curve. However, some feedback suggested that the structure of the questionnaire was confusing, impacting participants' ability to fully express their experiences. For instance, questions related to time-based data were sometimes misunderstood, such as in Task 1 where participants added hours incorrectly, or in Task 4, where the time frame was misinterpreted. Confusion often occurred when users had to interpret data across different screens or filter it over specific periods.

While the SUS results showed minimal need for technical assistance (two participants rated it a "1") and demonstrated good overall usability, there is still room for improvement to reach an excellent SUS score (above 80.3). Future iterations should be refined to ensure that the questions are clear and user-friendly. Additionally, by improving the design of both the post-task questionnaire and the tasks themselves, GDHelper could potentially achieve higher task success rates and better

SUS scores in future usability studies.

Cognitive load and data interpretation

Some tasks required participants to interpret data or complete multi-step processes, which increased their cognitive load. This higher cognitive demand affected success rates for certain tasks. This was most apparent in the tasks involving detailed session data retrieval (e.g., Task 6: ”How many matches did the player play yesterday?”), where participants encountered difficulties despite their clinical expertise. The SUS results reflect this cognitive demand, with responses indicating slight inconsistencies in users' perceptions of the platform's complexity. For example, one participant gave a higher score (4) when asked about the complexity of the system, while others rated it lower, suggesting that for some users, the cognitive load was manageable, but for others, it posed a challenge. While the psychologists in this study are clinical experts, their unfamiliarity with specific gaming terms may have influenced the results. This suggests a need to refine the platform's presentation of gaming data by enhancing the clarity of time-related questions and providing guided analysis features, as users with different levels of expertise may require more intuitive support when interacting with gaming-specific data.

Questionnaire design

While the GDHelper platform itself received positive feedback, the structure and clarity of the questionnaire presented challenges. Some participants found certain questions confusing, which may have influenced the accuracy of their responses. Although the questionnaire was reviewed by the supervisors and by the psychologist, future iterations should include additional participants to better assess the clarity of the tasks of the questionnaire that obtained some errors. Improving the clarity of some tasks will not only improve the quality of feedback collected but also ensure that participants can accurately describe their experiences with the application. Following the feedback from the users who claimed that some of the questionnaire's questions were confusing and misleading, the improvement of the questionnaire should be considered in future user studies.

Task-specific insights

The varied performance across different tasks provides valuable insights for targeted improvements:

1. Time-based queries (Tasks 01 and 02) were hard for some users because they involved looking at data over time. We need to make it easier to find and understand that data.

2. Historical data analysis (Task 03) was easy for users, indicating that the application's long-term data presentation is good.

3. Specific session data retrieval (Task 04) proved difficult for some, highlighting an area to

make it easier to find and understand that data.

4. Administrative tasks like patient management (Task 05) saw perfect completion rates, affirming the application's strength in basic operations.

These insights will help in future development cycles where the focus will be on fixing the areas where users struggled while keeping the parts of the application that are already working well.

Limitations and future research

This study, while informative, has several limitations that should be addressed in future research:

- Limited geographic diversity (all participants from Portugal)
- Small sample size
- Narrow age range

This study has some limitations that must be addressed in future research. The participant pool was geographically limited to Portugal and consisted of a narrow age range, reducing the generalizability of the results. Future studies should aim for a more diverse sample to better understand how the platform performs across different user groups. Additionally, increasing the sample size will provide more conclusions about the platform's usability.

Implications for future development

Based on the task-specific insights and SUS feedback, several key areas for future development arise:

1. Guided analysis features to reduce cognitive load for complex tasks
2. Customizable dashboards to meet different user needs
3. Expanded help resources, such as contextual hints and tutorials

Prioritizing these areas in future iterations of GDHelper will likely cause significant improvements in overall usability and user satisfaction.

6.4 Summary

The GDHelper platform demonstrates potential as a clinical tool for monitoring gaming behavior, achieving a System Usability Scale score of 75.83 out of 100, which indicates above-average usability. The SUS scores indicate that users found the application easy to use, with high levels of confidence and minimal need for technical assistance, reflecting its solid foundation for usability. Additionally, the intuitive interface and high task completion rates for basic functions contribute to its practicality in clinical settings. However, the challenges identified in some tasks highlight areas for improvement. By addressing these usability concerns and building on the application's strengths, such as its simplicity and ease of learning, GDHelper can develop into an even more effective and user-friendly tool for healthcare professionals. Future development should focus on providing more help resources and guided analysis features, all while maintaining the ease of use that users already value.

Chapter 7

Conclusion and Future Work

This chapter presents the outcomes and future directions of the GDHelper application, a tool developed to assist health professionals in diagnosing Internet Gaming Disorder (IGD). We evaluate the successful integration of Counter-Strike 2 telemetry data and discuss the achievement of key project objectives, validated through testing with healthcare professionals. The chapter concludes by exploring potential improvements and expansions for future development, aiming to further improve the application's utility in clinical settings.

7.1 Conclusion

This project set out to develop an interactive application to assist the diagnosis of Internet Gaming Disorder, specifically focusing on Counter-Strike 2 gameplay. The primary goal was to provide health professionals with a dynamic and effective tool for analyzing gaming behavior. Reflecting on the initial goals and the outcomes achieved, we can draw the following conclusions:

1. **Integration of existing metrics and visualizations:** The project successfully incorporated previously defined metrics, including gameplay duration and session frequency (per year, month, and day), into a dynamic, user-friendly interface. This achievement represents a significant step forward from the static solutions previously available, offering health professionals a more interactive and comprehensive tool for analysis.

2. **Enhancement of data accessibility:** The application provides healthcare professionals with easy access to individual player statistics. Users can efficiently navigate through different sections to view detailed gameplay data for each patient. The platform's layout ensures that important metrics, such as gameplay hours and behavioral patterns, are accessible, allowing professionals to analyze individual gaming behaviors effectively. While this functionality supports specific patient assessments, future iterations of the application could further enhance data accessibility by streamlining the interface and optimizing data retrieval processes to allow quicker access to key insights.

3. **Regular access to updated game data:** Allowing users to update data for either specific players or all players simultaneously, successfully addresses the goal of providing regular

access to updated game data. This feature ensures that the application remains up-to-date with the latest player information and successfully meets the goal of providing regular access to fresh game data. This solution gives professionals control over when and how they update their data, making it both flexible and practical for clinical use.

4. **Support for health professionals in clinical settings:** The application's design prioritized ease of use for health professionals, a goal that was validated during the user testing phase. Feedback from health professionals confirmed that the application is indeed user-friendly and practical for clinical settings, suggesting achievement of this objective. The success of this design approach confirms that the application is suitable for clinical settings and could assist health professionals in their diagnostic processes.

5. **Facilitation of objective assessment of IGD:** By offering objective, telemetry-based insights into players' gaming behavior, the application successfully complements traditional self-report questionnaires. This achievement represents a significant step towards more accurate and comprehensive IGD assessments, fulfilling the goal of enabling health professionals to analyze players' behaviors through real-time data rather than relying on purely subjective data.

6. **Collaboration with experts:** The involvement of Joana Cardoso, a Clinical Psychologist and Ph.D. candidate, in both the development and testing phases of the application ensured that it met professional needs and standards. Her input was extremely important in shaping the application's design to meet the specific needs of health professionals working with IGD patients. Additionally, she participated in testing the application, helping to ensure that the tool was both relevant and effective in a clinical context. The collaboration was vital in validating the application's relevance and effectiveness in clinical practice.

In conclusion, the project has essentially succeeded in meeting its primary goals. The resulting application represents a significant advancement in tools available for IGD diagnosis and treatment, particularly for Counter-Strike 2 players. By providing an interactive, data-driven platform, will enable health professionals to make more informed decisions and conduct more meaningful clinical follow-ups.

The overall success of the project in creating a user-friendly, clinically relevant tool is evident. The positive feedback from health professionals during testing further validates the application's potential impact in clinical settings.

This project not only builds upon previous research but also opens new roads for the integration of game telemetry data in mental health assessments. As the field of IGD research continues to evolve, tools like this application will play a crucial role in bridging the gap between gaming behavior and clinical practice, ultimately leading to a better understanding and treatment of IGD.

7.2 Future Work

While the current iteration of the application has successfully met many of its initial goals, several avenues for future development and research could further enhance its utility in the diagnosis and treatment of Internet Gaming Disorder. These potential areas for future work include:

1. **Expansion to other games:**

    - Extending the application to cover other popular games beyond Counter-Strike 2 is important because it allows for a more comprehensive assessment of a patient's overall gaming behavior. Our results are based on telemetry data from CS2, which may not fully capture the broader gaming habits of patients with IGD, who often engage with a variety of game genres. By including more games, healthcare professionals can gain a better understanding of different gaming patterns and behaviors, helping to ensure that no significant trends or issues are overlooked.
    - Developing game-specific metrics and visualizations that account for unique aspects of different game genres would improve the relevance of the data, as different games require different skills and time commitments, and may affect behavior in different ways.

2. **Integration of IGD risk indicator:**

    - Development of an IGD risk indicator: GDHelper could implement an indicator that evaluates player behavior patterns using key clinical criteria for IGD. By analyzing the frequency, duration, and intensity of gaming sessions, the platform could calculate a risk score that flags players who show signs of problematic gaming behaviors.
    - Predictive analytics: Using machine learning algorithms, the platform could continuously learn from new data to refine this risk indicator. The application could then identify trends, such as increasing playtime during late-night hours or excessive session durations, and notify health professionals when a player's behavior aligns with established IGD risk factors (e.g., withdrawal symptoms, loss of interest in other activities, or failed attempts to reduce gaming).
    - Flagging potential IGD cases: By using the IGD risk indicator, GDHelper could automatically flag players whose behavior surpasses a predefined threshold, offering early warning signals to clinicians.

3. **Longitudinal studies:**

    - Conduct long-term studies to track the effectiveness of the application in supporting IGD diagnosis and treatment over time. These studies should include gamers of different profiles, such as professionals and amateurs, as well as individuals with both healthy and problematic gaming behaviors to ensure the results are representative of a wide range of gaming habits.

- Analyze how changes in gaming behavior correlate with treatment outcomes to refine the application's metrics and visualizations, ensuring they remain relevant and effective for diverse types of users.

4. **Enhanced data collection:**

   - Explore possibilities for more frequent or even real-time data updates to provide an even more current view of player behavior.
   - Investigate the integration of additional data sources, such as in-game chat logs or social interactions, to provide a full view of a player's gaming experience.

5. **User experience improvements:**

   - Conduct further usability studies to identify any remaining pain points in the user interface.
   - Develop customizable dashboards that allow health professionals to tailor the display of information to their specific needs or preferences.

6. **Integration with clinical tools:**

   - Explore possibilities for integrating the application with existing electronic health record systems.
   - Develop features that allow for easy export of data and visualizations for inclusion in clinical reports or research papers.

7. **Expansion of collaborative network:**

   - Engage with a wider network of mental health professionals, game developers, and researchers to continually refine and validate the application's metrics and interpretations.
   - Enable collaborative research projects that use GDHelper's data to advance knowledge of IGD.

8. **Ethical considerations and privacy enhancements:**

   - Conduct a thorough review of data privacy practices and implement any necessary enhancements to ensure compliance with evolving data protection regulations. It should be noted that all players analyzed will give their explicit consent before any data is collected or shared.
   - Develop features that give players more control over their data, including options to view their own statistics and control what information is shared with health professionals.

By pursuing these areas of future work, the application can continue to evolve, improving its effectiveness as a tool for assisting the IGD diagnosis, while also contributing valuable data and insights to the broader field of gaming and mental health research. The continuing development and refinement of such tools will be crucial in addressing the growing concerns around problematic gaming behaviors in an increasingly digital world.

Bibliography

[1] Craig Larman. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. Pearson Education (US), November 2004.

[2] Magy Seif El-Nasr, Truong-Huy D Nguyen, Alessandro Canossa, and Anders Drachen. *Game Data Science*. Oxford University Press, 2021.

[3] Beltrán Vázquez Liniers. Use of telemetry data and emotion analysis to complement the analysis of problematic gaming behaviors. Master's thesis, Universidade de Lisboa, 2023.

[4] Simone Kriglstein, Pejman Mirza-Babaei, Günter Wallner, Steven Schirra, Sven Charleer, Manfred Tscheligi, and Kathrin Gerling. Be part of it: Spectator experience in gaming and esports. `https://seegamesws.wordpress.com/`, 2020.

[5] American Psychiatric Association. *Diagnostic and Statistical Manual of Mental Disorders (DSM-5)*. American Psychiatric Publishing, Washington, DC, 2013.

[6] Daria J Kuss, Mark D Griffiths, and Halley M Pontes. Chaos and confusion in dsm-5 diagnosis of internet gaming disorder: Issues, concerns, and recommendations for clarity in the field. *Journal of Behavioral Addictions*, 6(2):103–109, 2017.

[7] Video Games Europe. EGDF key facts report from 2022, 2022. Accessed: 2022-12-05.

[8] C. J. Ferguson. The good, the bad and the ugly: a meta-analytic review of positive and negative effects of violent video games. *Psychiatr Q*, 78:309–16, 2007.

[9] Y. J. Halbrook, A. T. O'Donnell, and R. M. Msetfi. When and how video games can be good: a review of the positive effects of video games on well-being. *Perspect Psychol Sci*, 14:1096–104, 2019.

[10] W. A. Warburton, S. Parkes, and N. Sweller. Internet gaming disorder: evidence for a risk and resilience approach. *Int J Environ Res Public Health*, 19:5587, 2022.

[11] R. A. Mentzoni, G. S. Brunborg, H. Molde, H. Myrseth, K. J. M. Skouverø, J. Hetland, and et al. Problematic video game use: estimated prevalence and associations with mental and physical health. *Cyberpsychol Behav Soc Netw*, 14:591–6, 2011.

[12] K. Zajac, M. K. Ginley, R. Chang, and N. M. Petry. Treatments for internet gaming disorder and internet addiction: A systematic review. *Psychology of Addictive Behaviors*, 31(8):979, 2017.

[13] M. W. Stevens, D. Dorstyn, P. H. Delfabbro, and D. L. King. Global prevalence of gaming disorder: A systematic review and meta-analysis. *Australian & New Zealand Journal of Psychiatry*, 55(6):553–568, 2021.

[14] World Health Organization. *ICD-11: International Classification of Diseases 11th Revision*. Geneva, 2019.

[15] Xuechan Lyu, Tianzhen Chen, Zhe Wang, Jing Lu, Chenyi Ma, Haoye Tan, Runji Li, Peiyan Wang, Limin Ma, Hongwei Li, et al. The reliability and validity of a screening scale for online gaming disorder among chinese adolescents and young adults. *BMC psychiatry*, 22(1):1–9, 2022.

[16] Torres-Rodríguez et al. The treatment of internet gaming disorder: A brief overview of the pipatic program. *International Journal of Mental Health and Addiction*, 16:1000–1015, 2018.

[17] Daniel L. King, Samuel R. Chamberlain, Natacha Carragher, Joël Billieux, Dan Stein, Kathlen Mueller, ..., and Paul H. Delfabbro. Screening and assessment tools for gaming disorder: A comprehensive systematic review. *Clinical Psychology Review*, 77:101831, 2020.

[18] Steam. Counter-strike: Global offensive. `https://store.steampowered.com/app/730/CounterStrike_Global_Offensive/`. Accessed: 2023-11-24.

[19] Brian Bowman, Niklas Elmqvist, and TJ Jankun-Kelly. Toward visualization for games: Theory, design space, and patterns. *IEEE transactions on visualization and computer graphics*, 18(11):1956–1968, 2012.

[20] Günter Wallner and Simone Kriglstein. Comparative visualization of player behavior for serious game analytics. *Serious games analytics: Methodologies for performance measurement, assessment, and improvement*, pages 159–179, 2015.

[21] FPDE. Fpde - digital amalgamation of knowledge. `https://fpde.pt/`, 2023. Accessed: 2024-01-03.

[22] Alexandra Torres-Rodríguez, Mark D Griffiths, Xavier Carbonell, Núria Farriols-Hernando, and Encarna Torres-Jimenez. Internet gaming disorder treatment: A case study evaluation of four different types of adolescent problematic gamers. *International Journal of Mental Health and Addiction*, 17:1–12, 2019.

[23] Céline Bonnaire and Olivier Phan. Relationships between parental attitudes, family functioning and internet gaming disorder in adolescents attending school. *Psychiatry Research*, 255:104–110, 2017.

[24] Melina A Throuvala and Janikian et al. The role of family and personality traits in internet gaming disorder: A mediation model combining cognitive and attachment perspectives. *Journal of behavioral addictions*, 8(1):48–62, 2019.

[25] Lutz Wartberg and Kriston et al. A longitudinal study on psychosocial causes and consequences of internet gaming disorder in adolescence. *Psychological medicine*, 49(2):287–294, 2019.

[26] Centers for Disease Control and Prevention. Arthritis: Comorbidities. `https://www.cdc.gov/arthritis/data_statistics/comorbidities.htm`, 2021. Accessed: 2023-11-24.

[27] Vega González-Bueso, Juan José Santamaría, Daniel Fernández, Laura Merino, Elena Montero, and Joan Ribas. Association between internet gaming disorder or pathological video-game use and comorbid psychopathology: a comprehensive review. *International journal of environmental research and public health*, 15(4):668, 2018.

[28] Hyunsuk Jeong, Hyeon Woo Yim, Seung-Yup Lee, Hae Kook Lee, Marc N Potenza, Sun-Jin Jo, and Hye Jung Son. Reciprocal relationship between depression and internet gaming disorder in children: A 12-month follow-up of the icure study using cross-lagged path analysis. *Journal of behavioral addictions*, 8(4):725–732, 2019.

[29] Dan J Stein. Obsessive-compulsive disorder. *The Lancet*, 360(9330):397–405, 2002.

[30] Sara Fazeli, Isa Mohammadi Zeidi, Chung-Ying Lin, Peyman Namdar, Mark D Griffiths, Daniel Kwasi Ahorsu, and Amir H Pakpour. Depression, anxiety, and stress mediate the associations between internet gaming disorder, insomnia, and quality of life during the covid-19 outbreak. *Addictive Behaviors Reports*, 12:100307, 2020.

[31] Ashraf B Muzwagi, Fatima B Motiwala, Geetha Manikkara, Abid Rizvi, Mark A Varela, A John Rush, Muhammad K Zafar, and Shailesh B Jain. How are attention-deficit hyperactivity and internet gaming disorders related in children and youth? *Journal of Psychiatric Practice®*, 27(6):439–447, 2021.

[32] Alayna Murray, Arlene Mannion, June L Chen, and Geraldine Leader. Gaming disorder in adults with autism spectrum disorder. *Journal of Autism and Developmental Disorders*, pages 1–8, 2021.

[33] Stéphanie Laconi, Sophie Pirès, and Henri Chabrol. Internet gaming disorder, motives, game genres and psychopathology. *Computers in human behavior*, 75:652–659, 2017.

[34] Israel Oluwasegun Ayenigbara. Gaming disorder and effects of gaming on health: An overview. *Journal of addiction medicine and therapeutic science*, 4(1):1–3, 2018.

[35] Catarina N Matias, Joana Cardoso, Margarida L Cavaca, Sofia Cardoso, Rita Giro, João Vaz, Pedro A Couto, Artemisa Rocha Dores, Tiago B Ferreira, GM Tinsley, et al. Game on: A cross-sectional study on gamers' mental health, game patterns, physical activity, eating and sleeping habits. *Computers in Human Behavior*, 148:107901, 2023.

[36] Griffiths et al. A brief overview of internet gaming disorder and its treatment. *Australian Clinical Psychologist*, 2(1):20108, 2016.

[37] Valve Corporation. Counter-strike official website. `https://www.counter-strike.net/`. Accessed on: 2023-11-24.

[38] History of counter-strike. `https://history-computer.com/history-of-counter-strike/`, note = Accessed on: 2023-11-24,.

[39] F Rehbein, DL King, A Staudt, T Hayer, and HJ Rumpf. Contribution of game genre and structural game characteristics to the risk of problem gaming and gaming disorder: a systematic review. *Curr Addict Rep*, 8:263–81, 2021.

[40] Allen Rubin. Predicting round and game winners in csgo. In *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 2023.

[41] Steam workshop - counter-strike: Global offensive. `https://steamcommunity.com/app/730/workshop/`. Accessed: 2023-11-24.

[42] FACEIT. Faceit. `https://www.faceit.com/`. Accessed: 2023-12-17.

[43] Faceit cs2: Beginner's guide. `https://scope.gg/guides/faceit-en/`. Accessed: 2023-12-17.

[44] Chess.com. What is the Elo Rating? `https://www.chess.com/terms/elo-rating-chess#what-is-elo-rating`, 2022. Accessed: 2023-12-17.

[45] HLTV. Counter-strike: Global offensive community. `https://www.hltv.org/`. Accessed: 2023-11-24.

[46] Faceit. Faceit developer portal. `https://developers.faceit.com/`. Accessed: 2023-12-17.

[47] BrowserStack. Guide to web development frameworks. `https://www.browserstack.com/guide/web-development-frameworks`. Accessed on: 2023-11-30.

[48] React. `https://react.dev/`. Accessed: 2023-11-20.

[49] Vue.js. https://vuejs.org/. Accessed: 2023-11-20.

[50] Angular. https://angular.io/. Accessed: 2023-11-20.

[51] TechTarget. Web development framework. https://www.techtarget.com/searchcontentmanagement/definition/web-development-framework-WDF. Accessed on: 2023-11-30.

[52] Express.js. https://expressjs.com/. Accessed: 2023-11-20.

[53] Django. https://www.djangoproject.com/. Accessed: 2023-11-20.

[54] Ruby on rails. https://rubyonrails.org/. Accessed: 2023-11-20.

[55] Asp.net. https://dotnet.microsoft.com/pt-br/apps/aspnet. Accessed: 2023-11-20.

[56] Vercel. Next.js. https://nextjs.org/, 2023. Accessed: 2024-01-05.

[57] Node.js. https://nodejs.org/. Accessed: 2023-11-24.

[58] MongoDB. Mongodb. https://www.mongodb.com/. Accessed: 2023-12-17.

[59] The PostgreSQL Project. PostgreSQL. https://www.postgresql.org/. Accessed: 2023-11-24.

[60] Coding Ninjas. Advantages and disadvantages of node.js. https://www.codingninjas.com/studio/library/advantages-and-disadvantages-of-nodejs. Accessed on: 2023-11-30.

[61] Node package manager (npm). https://www.npmjs.com/. Accessed: 2023-11-24.

[62] MongoDB. Why use mongodb. https://www.mongodb.com/why-use-mongodb, 2023. Accessed: 2024-01-05.

[63] MongoDB. Advantages of mongodb. https://www.mongodb.com/advantages-of-mongodb, 2023. Accessed: 2024-01-05.

[64] Halley M Pontes and Mark D Griffiths. Measuring dsm-5 internet gaming disorder: Development and validation of a short psychometric scale. *Computers in human behavior*, 45:137–143, 2015.

[65] Halley M Pontes and Mark D Griffiths. Portuguese validation of the internet gaming disorder scale–short-form. *Cyberpsychology, Behavior, and Social Networking*, 19(4):288–293, 2016.

[66] Iván Sánchez-Iglesias, Mónica Bernaldo-de Quirós, Francisco J Labrador, Francisco J Estupiñá Puig, Marta Labrador, and Ignacio Fernández-Arias. Spanish validation and scoring of the internet gaming disorder scale-short-form (igds9-sf). *The Spanish Journal of Psychology*, 23:e22, 2020.

[67] Zsolt Demetrovics, Róbert Urbán, Katalin Nagygyörgy, Judit Farkas, Mark D Griffiths, Orsolya Pápay, Gyöngyi Kökönyei, Katalin Felvinczi, and Attila Oláh. The development of the problematic online gaming questionnaire (pogq). *PloS one*, 7(5):e36417, 2012.

[68] Jeroen S Lemmens, Patti M Valkenburg, and Douglas A Gentile. The internet gaming disorder scale. *Psychological assessment*, 27(2):567, 2015.

[69] Jeroen S Lemmens, Patti M Valkenburg, and Jochen Peter. Development and validation of a game addiction scale for adolescents. *Media psychology*, 12(1):77–95, 2009.

[70] Antonius J van Rooij, Tim M Schoenmakers, Regina JJM Van den Eijnden, Ad A Vermulst, and Dike van de Mheen. Video game addiction test: validity and psychometric characteristics. *Cyberpsychology, Behavior, and Social Networking*, 15(9):507–511, 2012.

[71] Sofia Vadlin, Cecilia Åslund, and Kent W Nilsson. Development and content validity of a screening instrument for gaming addiction in adolescents: The gaming addiction identification test (gait). *Scandinavian Journal of Psychology*, 56(4):458–466, 2015.

[72] Halley M Pontes, Bruno Schivinski, Cornelia Sindermann, Mei Li, Benjamin Becker, Min Zhou, and Christian Montag. Measurement and conceptualization of gaming disorder according to the world health organization framework: The development of the gaming disorder test. *International Journal of Mental Health and Addiction*, 19:508–528, 2021.

[73] Orsolya Király, Beáta Bőthe, Jano Ramos-Diaz, Afarin Rahimi-Movaghar, Katerina Lukavska, Ondřej Hrabec, Michal Miovsky, Joël Billieux, Jory Deleuze, Filip Nuyens, Laurent Karila, Mark D. Griffiths, Katalin Nagygyörgy, Róbert Urbán, Marc N. Potenza, Daniel L. King, Hans-Jürgen Rumpf, Natacha Carragher, and Zsolt Demetrovics. Ten-item internet gaming disorder test (igdt-10): Measurement invariance and cross-cultural validation across seven language-based samples. *Psychology of Addictive Behaviors*, 33(1):91–103, 2019.

[74] Günter Wallner and Simone Kriglstein. Visualization-based analysis of gameplay data–a review of literature. *Entertainment Computing*, 4(3):143–155, 2013.

[75] Ana Paula Afonso, Maria Beatriz Carmo, Tiago Gonçalves, and Pedro Vieira. Visualeague: Player performance analysis using spatial-temporal data. *Multimedia Tools and Applications*, 78:33069–33090, 2019.

[76] Anders Drachen. Debate: Games-based collaboration as a driver for massive-scale mental health research. *Child and Adolescent Mental Health*, 28(1):148–149, 2023.

[77] Sebastian Deterding, K. Mitchell, Rachel Kowert, and B. King. Games futures i. *ACM Games: Research and Practice*, 1(1):1–4, 2023.

[78] Matti Vuorre, Kristofer Magnusson, Niklas Johannes, James Butlin, and Andrew K Przybylski. An intensive longitudinal dataset of in-game player behaviour and well-being in powerwash simulator. *Scientific Data*, 10(622), 2023.

[79] David Zendle, Catherine Flick, Darel Halgarth, Nick Ballou, Simon Demediuk, and Anders Drachen. Cross-cultural patterns in mobile playtime: an analysis of ; billion hours of human data.

[80] Rema Padman, Disha Gupta, Bhargav Sri Prakash, Chelladurai Krishnan, and K Panchatcharam. An exploratory analysis of game telemetry from a pediatric mhealth intervention. In *MEDINFO 2017: Precision Healthcare through Informatics*, pages 183–187. IOS Press, 2017.

[81] Douglas A Parry, Brittany I Davidson, Craig JR Sewall, Jacob T Fisher, Hannah Mieczkowski, and Daniel S Quintana. A systematic review and meta-analysis of discrepancies between logged and self-reported digital media use. *Nature Human Behaviour*, 5(11):1535–1547, 2021.

[82] Adam S Kahn, Rabindra Ratan, and Dmitri Williams. Why we distort in self-report: Predictors of self-report errors in video game play. *Journal of Computer-Mediated Communication*, 19(4):1010–1023, 2014.

[83] Sally Andrews, David A Ellis, Heather Shaw, and Lukasz Piwek. Beyond self-report: Tools to compare estimated and real-world smartphone use. *PloS one*, 10(10):e0139004, 2015.

[84] Joseph J Thompson, Mark R Blair, Lihan Chen, and Andrew J Henrey. Video game telemetry as a critical tool in the study of complex skill learning. *PloS one*, 8(9):e75129, 2013.

[85] User interface design for healthcare applications. `https://digitalhealth.folio3.com/blog/user-interface-design-for-healthcare-applications/`. Accessed: 2024-09-17.

[86] UGEM Design. Ui design tips for healthcare apps. `https://ugem.design/blog/ui-design-tips-for-healthcare-apps`, 2024. Accessed: 2024-09-17.

[87] Vimla L. Patel and Andre W. Kushniruk. Interface design for health care environments: The role of cognitive science. *Cognitive Studies in Medicine, McGill University*, pages 29–37, 1998.

[88] Paul C. Tang and Vimla L. Patel. Major issues in user interface design for health professional workstations: summary and recommendations. *International Journal of Bio-Medical Computing*, 34:139–148, 1994.

[89] Roger S. Pressman. *Software Engineering: A Practitioner's Approach*. McGraw-Hill, 2014.

[90] David C. Young. Software development methodologies. *ResearchGate*, 2013. Accessed: 2023-12-15.

[91] Soobia Saeed, NZ Jhanjhi, Mehmood Naqvi, and Mamoona Humayun. Analysis of software development methodologies. *International Journal of Computing and Digital Systems*, 8(5), 2019.

[92] Ana Paula Afonso, Manuel J. Fonseca, Joana Cardoso, and Beltran Vasquez. Towards an automated approach for understanding problematic gaming. *Frontiers in Sports and Active Living*, 6:1407848, 2024.

[93] InterviewBit. System architecture. `https://www.interviewbit.com/blog/system-architecture/`, 2023. Accessed: 2024-09-18.

[94] Next-Auth.js Contributors. Next-auth.js. `https://next-auth.js.org/`, 2023. Accessed: 2024-09-18.

[95] Typescript. `https://www.typescriptlang.org/`. Accessed: 2024-09-18.

[96] 5 reasons to use typescript for your projects. `https://www.clickittech.com/developer/why-use-typescript/`.

[97] Nextjs docs. `https://nextjs.org/docs`.

[98] Recharts. `https://recharts.org/en-US/`. Accessed: 2024-09-18.

[99] Create charts using recharts. `https://refine.dev/blog/recharts/`.

[100] Tailwind css. `https://tailwindcss.com/`.

[101] Styling: Tailwind css — next.js. `https://nextjs.org/docs/app/building-your-application/styling/tailwind-css`. Accessed: 2024-09-18.

[102] Shadcn ui. `https://ui.shadcn.com/`. Accessed: 2024-09-18.

[103] Mongoose. `https://mongoosejs.com/`. Accessed: 2024-09-18.

[104] Microsoft. Visual studio code. `https://code.visualstudio.com/`. Accessed: 2023-12-17.

[105] Eslint. `https://eslint.org/`. Accessed: 2024-09-18.

[106] Git. `https://git-scm.com/`. Accessed: 2024-09-18.

[107] Github. `https://github.com/`. Accessed: 2024-09-18.

[108] An introduction to github. https://digital.gov/resources/an-introduction-github/. Accessed: 2024-09-18.

[109] 30 most popular npm packages for node js developers. https://www.turing.com/blog/top-npm-packages-for-node-js-developers. Accessed: 2024-09-18.

[110] Docker. https://www.docker.com/. Accessed: 2024-09-18.

[111] Webpack. https://webpack.js.org/. Accessed: 2024-09-18.

[112] World Wide Web Consortium. Accessibility fundamentals: Introduction. https://www.w3.org/WAI/fundamentals/accessibility-intro/. Accessed: 2024-09-22.

[113] Coloring for colorblindness. https://davidmathlogic.com/colorblind/#%23DA5526-%23F6893D-%23FEBC38-%23D8C6B4-%23697F90-%236CA5D2-%23000000-%23F9CD75-%23833C7F-%230DCEE1-%23F9CE00. Accessed: 2024-09-23.

[114] WebAIM. Wave web accessibility evaluation tools. https://wave.webaim.org/, 2024. Accessed: 2024-09-23.

[115] J. R. Lewis. IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1):57–78, 1995. https://garyperlman.com/quest/quest.cgi?form=ASQ.

[116] John Brooke. *SUS – a quick and dirty usability scale*, pages 189–194. Taylor Francis, 01 1996.

[117] Jeff Sauro. Measuring usability with the system usability scale (sus). February 2011. Accessed: 2024-10-25.

[118] Google forms: Online form creator. https://www.google.com/forms/about/. Accessed: 2024-09-26.

Appendix A

GDHelper's Usability Evaluation Questionnaire

Avaliação da aplicação GDHelper

A aplicação GDHelper é uma ferramenta destinada a auxiliar os profissionais de saúde na análise e diagnóstico da Perturbação de Jogo na Internet (IGD). Esta aplicação foi desenvolvida no âmbito de um projeto de tese de mestrado na Faculdade de Ciências da Universidade de Lisboa, em colaboração com a psicóloga clínica Joana Cardoso.

Este questionário tem como objetivo avaliar a funcionalidade e usabilidade da aplicação GDHelper. A sua participação fornecerá informações valiosas para melhorar a aplicação e o seu potencial impacto no diagnóstico e tratamento da IGD.

A participação é totalmente voluntária e pode desistir em qualquer altura, sem necessidade de indicar qualquer motivo. Esta avaliação consiste em várias tarefas que irá realizar utilizando a aplicação GDHelper.

1 - Inicialmente, será apresentado um Formulário de Consentimento Informado, seguido de um questionário inicial sobre dados demográficos e experiência de utilização.

2 - Depois, o participante irá realizar uma série de tarefas relacionadas com a aplicação. Após cada tarefa, ser-lhe-á pedido que responda a um conjunto de perguntas sobre a sua experiência.

3 - Após a
realização de todas as tarefas o participante responderá ao questionário final
(SUS) relativo à aplicação em geral.

Respeitamos todas as práticas éticas e legais, e todas as informações fornecidas serão tratadas com estrita confidencialidade. Para garantir o anonimato, os registos pessoais só serão acessíveis na sua totalidade ao investigador principal. Se os seus dados forem utilizados em publicações ou apresentações, serão completamente anonimizados, sem referências directas ou indirectas à sua identidade.

A avaliação centrar-se-á em vários aspectos da aplicação, incluindo o design da interface do utilizador, as funcionalidades de visualização de dados e a utilidade geral na análise dos padrões de comportamento de jogo.

Se tiver alguma dúvida ou questão sobre este estudo, por favor contacte a investigadora, Daniela Jorge (fc54989@alunos.fc.ul.pt), ou os orientadores do projeto, Professora Doutora Ana Paula Afonso (apafonso@fc.ul.pt) e Professor Doutor Manuel J. Fonseca (mjfonseca@fc.ul.pt).

Obrigado pelo seu tempo e valiosa contribuição para este projeto de investigação.

* Indicates required question

1. Por favor preencha o seguinte formulário de consentimento informado. *

Mark only one oval per row.

	Sim	Não
Confirmo que li e compreendi o folheto informativo associado ao projecto.	⬭	⬭
Foi-me dada a oportunidade de ler e considerar a informação apresentada, e fazer perguntas, asquais foram respondidas de forma satisfatória.	⬭	⬭
Compreendo que a minha participação é voluntária e que sou livre de desistir do estudo em qualquer altura, sem ter que dar quaisquer explicações e sem quaisquer consequências.	⬭	⬭
Compreendo que os dados recolhidos durante o estudo possam ser do conhecimento dos membros da equipa de investigação, sempre que necessário	⬭	⬭

para o estudo. Autorizo que os membros da equipa tenham acesso a esses dados.		
Compreendo que, caso esta investigação venha a ser publicada, todos os dados serão mantidos anónimos e nenhuma informação será identificável como sendo minha.		
Gostaria que me fosse enviado o relatório final do estudo.		
Gostaria de ser contactado para o endereço acima acerca de sessões ou estudos adicionais relacionados com este estudo.		
Declaro que não comuniquei nenhuma razão potencial de qualquer natureza que constitua um eventual factor de risco para a minha saúde ou integridade física		

ou integridade física

Declaro que participo neste estudo sem qualquer remuneração ou contrapartida, para além do ressarcimento das despesas em que tiver incorrido ou compensação simbólica pelo meu tempo.		
Declaro que tomo a minha decisão de forma inteiramente livre.		
Concordo em participar neste estudo.		

Declaro que participo neste estudo sem qualquer remuneração ou contrapartida, para além do ressarcimento das despesas em que tiver incorrido ou compensação simbólica pelo meu tempo.
Declaro que tomo a minha decisão de forma inteiramente livre.
Concordo em participar neste estudo.

2. Caso queira que lhe seja enviado o relatório final e ser contactado acerca de sessões ou estudos adicionais relacionados com este estudo **por favor indique o email** para o qual pretende ser contactado.

Questionário Inicial

3. País *

4. Género *

Mark only one oval.

- Feminino
- Masculino
- Prefiro não dizer
- Other: ________________

5. Idade *

Mark only one oval.

- < 18
- 18 - 24
- 25 - 34
- 35 - 44
- 45 - 54
- 55 - 64
- > 65
- Prefiro não dizer

6. Nível de ensino *

Mark only one oval.

- Não concluiu a escolaridade
- Diploma do ensino secundário
- Grau de bacharelato (por exemplo: BA. BS)
- Mestrado (por exemplo: MA, MS, MEng, MEd, MSW, MBA)
- Grau de doutoramento (por exemplo, PhD, EdD)
- Prefiro não dizer

7. Ocupação *

8. Qual é o seu grau de familiaridade com as aplicações Web? *

Mark only one oval.

	1	2	3	4	5	
Pou	◯	◯	◯	◯	◯	Muito familiar

9. Qual a frequência de utilização de aplicações Web para monitorizar a saúde ou o estilo de vida? *

Mark only one oval.

	1	2	3	4	5	
Pou	◯	◯	◯	◯	◯	Muito frequente

Tarefa 1

Aceda à aplicação.

Imagine que, como terapeuta, tem uma sessão de 30 minutos com o seu paciente. Utilize o GDHelper para entrar na sua conta "terapeuta1@gmail.com" com a palavra-passe “1234**passForte”. Veja quantas horas o seu paciente “ENERQIA-” jogou na última semana e qual a variação percentual das horas de jogo em relação à semana anterior.

10. Quantas horas é que jogou há dois dias durante a noite? *Exemplo: 3h* *

11. Quantas horas é que jogou? *Exemplo: 5h* *

12. Qual foi a variação percentual? *Exemplo: 18,1%* *

13. Compreendeu todas as informações apresentadas neste ecrã? *

Mark only one oval.

- Sim
- Não

14. Se respondeu "**Não**", explique o porquê.

15. De um modo geral, estou satisfeito com a facilidade de realização desta tarefa. *

Mark only one oval.

	1	2	3	4	5	
Disc						Concordo totalmente

Tarefa 2

Aceda à aplicação.

Para ter uma ideia dos padrões de jogo do seu paciente, pretende saber qual o período do dia em que "ENERQIA-" jogou mais nos últimos dois anos e se jogou mais durante a semana ou ao fim de semana. Utilize o GDHelper para recolher estes dados.

16. Em que período do dia é que este jogador jogou mais? *

Mark only one oval.

- De manhã
- À tarde
- À noite

17. Quantas horas durante a tarde jogou em setembro de 2022? *Exemplo: 37h.* *

____

18. Jogou mais ao fim de semana ou durante a semana? *

Mark only one oval.

- Fim de semana
- Semana

19. Compreendeu todas as informações apresentadas neste ecrã? *

Mark only one oval.

- Sim
- Não

20. Se respondeu "**Não**", explique o porquê.

____
____
____
____
____

21. De um modo geral, estou satisfeito com a facilidade de realização desta tarefa. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

Tarefa 3

Aceda à aplicação.

Imagine que tem um novo paciente na sua clínica, e pretende registá-lo na plataforma para poder ver as suas estatísticas nos últimos 4 anos. Use o "GDHelper" para analisar se o seu novo paciente "BENTEKE" jogou menos no último ano comparado com os anteriores.

22. Jogou mais no ano passado ou há dois anos? *

Mark only one oval.

- ◯ Ano passado
- ◯ Há dois anos

23. Em 2021, qual era a média de horas semanais? *Exemplo: 10.1h.*

____

24. Compreendeu todas as informações apresentadas neste ecrã? *

Mark only one oval.

- ◯ Sim
- ◯ Não

25. Se respondeu “**Não**”, explique o porquê.

____

____

____

____

____

26. De um modo geral, estou satisfeito com a facilidade de realização desta tarefa. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

Tarefa 4

Aceda à aplicação.

Para ter uma noção da intensidade de jogo do seu novo paciente “BENTEKE”, quer analisar quantos jogos e sessões ele jogou nos últimos 15 dias.

27. Qual é o número total de jogos que este jogador disputou nos últimos 15 dias? *
Exemplo: 10.

____

28. Quantas sessões foram realizadas há dez dias? *Exemplo: 5.* *

____

29. Compreendeu todas as informações apresentadas neste ecrã?

Mark only one oval.

- Sim
- Não

30. Se respondeu "**Não**", explique o porquê.

31. De um modo geral, estou satisfeito com a facilidade de realização desta tarefa. *

Mark only one oval.

	1	2	3	4	5	
Disc						Concordo totalmente

Tarefa 5

Aceda à aplicação.

Um dos seus pacientes terminou o seu tratamento, pelo que pretende eliminá-lo da sua lista de jogadores. Utilize o "GDHelper" para eliminar "BENTEKE" da sua lista de jogadores.

32. Compreendeu todas as informações apresentadas neste ecrã? *

Mark only one oval.

- Sim
- Não

33. Se respondeu “**Não**”, explique o porquê.

34. De um modo geral, estou satisfeito com a facilidade de realização desta tarefa. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

Tarefa 6

Aceda à aplicação.

Duas semanas depois, tem outra consulta com o jogador “ENERQIA-” e pretende registar o número de jogos que ele realizou nos últimos 7 dias e a variação percentual da duração média das sessões de jogo em relação à semana anterior. Comece por atualizar as informações do seu jogador para poder ver as novas informações da última semana.

35. Quantas partidas jogou ontem? *Exemplo: 4.* *

36. Qual foi a diferença horas/sessão? *Exemplo: 1.2%.* *

37. Compreendeu todas as informações apresentadas neste ecrã? *

Mark only one oval.

- Sim
- Não

38. Se respondeu “**Não**”, explique o porquê.

39. De um modo geral, estou satisfeito com a facilidade de realização desta tarefa. *

Mark only one oval.

	1	2	3	4	5	
Disc						Concordo totalmente

SUS (System Usability Scale)

40. Eu acho que gostaria de esta aplicação com frequência. *

Mark only one oval.

	1	2	3	4	5	
Disc						Concordo totalmente

41. Eu acho o GDHelper desnecessariamente complexo. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

42. Eu achei o GDHelper fácil de usar. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

43. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar a aplicação GDHelper. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

44. Eu acho que as várias funções da aplicação GDHelper estão muito bem integradas. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

45. Eu acho que a aplicação GDHelper apresenta muitas inconsistências. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

46. Eu imagino que as pessoas aprenderão a usar a aplicação GDHelper rapidamente. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

47. Eu achei a aplicação GDHelper complicada de usar. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Strongly Agree

48. Eu senti-me confiante ao usar a aplicação GDHelper. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente

49. Eu precisei de aprender várias coisas novas antes de usar a aplicação GDHelper. *

Mark only one oval.

	1	2	3	4	5	
Disc	◯	◯	◯	◯	◯	Concordo totalmente