Plotnikov Aleksandr

Development of an esports HUD for Counter-Strike 2

Bachelor's degree in Esports Business

Autumn/Spring 2024

KAMK • University of Applied Sciences

Abstract

Author(s): Plotnikov Aleksandr

Title of the Publication: Development of an esports HUD for Counter-Strike 2

Degree Title: Bachelor of Business Administration, Esports Business

Keywords: esports, HUD, Broadcast

Esports is in a stage of constant growth. Esports tournaments grow in scale, prize money, and the number of concurrent viewers. Every year the minimum requirements from various studios and tournament operators for tournament coverage and broadcasting increase. Although these requirements are not prescribed anywhere, every studio or tournament operator uses Head-Up Display (HUD) in its broadcast.

The aim of this thesis is to develop a working HUD specifically designed for broadcasting of Counter-Strike 2 (CS2) esports events. The development process involves a design phase where the HUD is conceptualized based on feedback gathered from a series of interviews with broadcasters and viewers. These interviews help to identify important elements that a broadcast HUD should contain, focusing on enhancing viewer understanding and engagement with the data of a CS2 esports match. The HUD integrates real-time game data using CS2's Game State Integration, ensuring accurate and timely display of information. Following the design phase goes coding, testing and evaluation stages.

The outcome of this thesis includes delivery of a functional and user-centered HUD which addresses viewers and broadcasters requirements.

Table of Contents

List of Symbols

HUD – Heads-Up-Display

CS2 – Counter-Strike 2

CS – Counter-Strike

UI – User Interface

UX – User Experience

GSI – Game State Integration

HTML – Hypertext Markup Language

JSON – JavaScript Object Notation

CSS – Cascading Style Sheets

OBS – Open Broadcaster Software

DOM – Document Object Model

FPS – First Person Shooter

HP- Health Points

IEM – Intel Extreme Masters

ADR – Average Damage per Round

1 Introduction

The Esports industry is increasing in popularity each year. Especially in the first-person shooter genre games like Counter-Strike 2 (CS2), for example according to Šimić (2024) The PGL Major Copenhagen tournament recorded a bigger audience reach with 1.85 million viewers in peak than previous Blast.tv Major Paris 2023 tournament with 1.58 million respectively. The development of an intuitive and informative at the same time HUD is critical for providing live game data of the match that is being broadcasted. It enhances viewer engagement and understanding, forming the overall user experience.

1.1 Goals and Objectives

The primary goal of the thesis is to deliver an effective HUD for CS2 esports broadcasts which is functional, accessible, and aesthetically appealing.

The objectives that are set to reach this goal are as follows: identify viewer requirements of an effective HUD, identify key HUD design principles, by using findings from interviews and implementing UX design theory, develop a prototype and design of a HUD, make a code for the prototype by utilizing game state integration of CS2 to retrieve live game data, test a usability of a HUD in a broadcasting software and evaluate the work.

1.2 Development task

The development task for this thesis includes the conceptualization, design, implementation, and evaluation phases of a Heads-Up Display (HUD) creation. This process encompasses these activities:

Conceptualization: conduct research to gather the information on needs and preferences of esports viewers from different segments. This involves review and analysis of existing HUDS and interviews with experienced and inexperienced viewers as well as broadcasters. Based on a collected insights determine principles that will be used in the HUD creation process. Shape an initial concept featuring set of elements that will be placed on HUD and their approximate location.

Design: Transform concept created in the previous stage into a further design. This involves creating detailed mockup and prototype that visualizes the HUD's overall layout, used color pallets and fonts. Apply interviewees insights on esports HUDs and established UX design theories to ensure that created overlay is intuitive and accessible.

Code Implementation: Development of the actual HUD software. Frontend coding of visual elements which are displayed on a web page by using HTML, CSS and JavaScript languages to create responsive and live updating interface. Integration of backend system, utilizing Game State Integration that retrieves live game data from CS2 match.

Evaluation: Conduct set of the HUD tests in a broadcasting software to ensure functionality and reliability of all the elements as well as integrity with broadcasting tool and compatibility with different resolutions. Gather a diverse group of viewers and broadcasters to hold interviews with , collect feedback on a created piece and analyze retrieved data to understand user satisfaction and potential are for improvement.

1.3 Limitations

Limitations of this study include several barriers. Technical restrictions of the existing GSI of CS2, which limits the amount of live match data that can be accessed, making it difficult to display all match statistics on a HUD. Diversity of esports audience, due to its' different levels of experience with the CS2 game it is challenging to design a HUD that fits the needs of beginner esports viewers and experienced spectators at the same time. Lack of studies and information on a topic of HUD development makes it harder to create a HUD from scratch. Subjectivity plays a big role in evaluation, due to effectiveness and intuitiveness assessment of a HUD being dependent on interviewees' opinion.

2 Theoretical Background

This chapter presents the theoretical background of the thesis which forms the basis for HUD development research. It includes brief history of esports and broadcasting, focusing on Counter-Strike discipline, HUD term explanation, differences between various interfaces and HUD development and trends throughout the years. This information serves as a foundation for development of the very own CS2 esports HUD.

2.1 Esports and broadcasting

According to Leroux-Parra (2020) Esports or electronic sports are video games that are played in a highly organized competitive environment. It is called this way because like in a traditional sport it consists of players, teams, coaches, analysts, and competitive matches in front of a crowd. (Hamari & Sjöblom, 2017, p. 211). The first recorded esports tournament was back in 1980, which was held within a Space Invaders discipline. (Sacco, 2016). Back then participants were trying to beat a high score in this game. However, the technologies were never standing still and during the years of computers improvement the games were also developing as well as a whole entertainment industry since it directly correlates. People started to play more competitive games against each other instead of trying to beat a high score in a single player game. In the 1990s with the arrival of the Internet and other personal computers (PCs) advancements the popularity of esports increased, due to the growing number of interested people and ability of tournament organizers to generate revenue, esports industry is intensifying and the events starting to be publicized. (Bousquet & Ertz, 2021) In the past esports were televised and that was the only way to watch esports if not attending the offline event. In 2011 the livestreaming platform Twitch was launched, and it gave esports a huge boost in popularity and opportunities. Esports enthusiasts could watch all their favorite tournaments from any point of the world.

2.2 Counter-Strike

Counter-Strike is a series of tactical first-person shooter games developed by Valve inc company (Ashley_Buckwell, 2024). The CS2 is the latest of them, it was launched on September 27th, 2023. According to Ashley Buckwell (2024) CS2 is built on a legacy of its predecessors with enhanced

graphics and refined gameplay, but keeping the core points which are genre, game modes and some of the mechanics. First-person shooter, also known as FPS, is a genre in which player sees the point of view of the character (Sandmann, 2024). CS2 is a tactical FPS, its' most popular and played game mode is 5v5 competitive match. Two teams are playing on a certain location which is called map. Teams consist of 5 players each, one team is Terrorists, and the other is Counter Terrorists, based on their teams, players have to reach different goals to win the round. Round is a period of time in which the main interaction between teams takes place, the match consists of certain number of rounds. Terrorists team, using strategy, has to plant bomb on one of two plant sites of the map and successfully explode it. Counter-Terrorists team should prevent the bomb plant or defuse the bomb until it is exploded. Each player has primary and secondary weapons which are knife and pistol and additionally they can buy better guns with the money which they receive per kills, deaths, round endings, lose or win streaks, bomb plants or defuses. In the middle of the match, teams switching sides and both teams starting to build its economy again from playing pistol round as in the beginning of the game (Morris, 2024).

2.3 Heads-up-display and its use in esports broadcasts

HUD is short for Heads-Up-Display, in video games it is a display area which contains characters important data, for instance character's current health, armor, ammo and other statistics. Initially HUDs were only used only for improving the player's experience in the game (Young, n.d.). In-game HUD which is demonstrated on figure 1 can be seen 5 parts: 1. Top bar or score bar, it displays team players, how many of them alive at the moment, round timer, match score. 2. Player bar showing current health which is now 100 Health Points (HP) and ammo in the clip of equipped gun - 7 and the total ammo - 35. 3. Radar with mini map of the area, player's and its teammates positions with generic name of the position. 4. Loadout of the player consisting of guns that player have in possession and a bomb or defuse kit if they have one. 5. Current money amount. (Counter-Strike 2, n.d.)

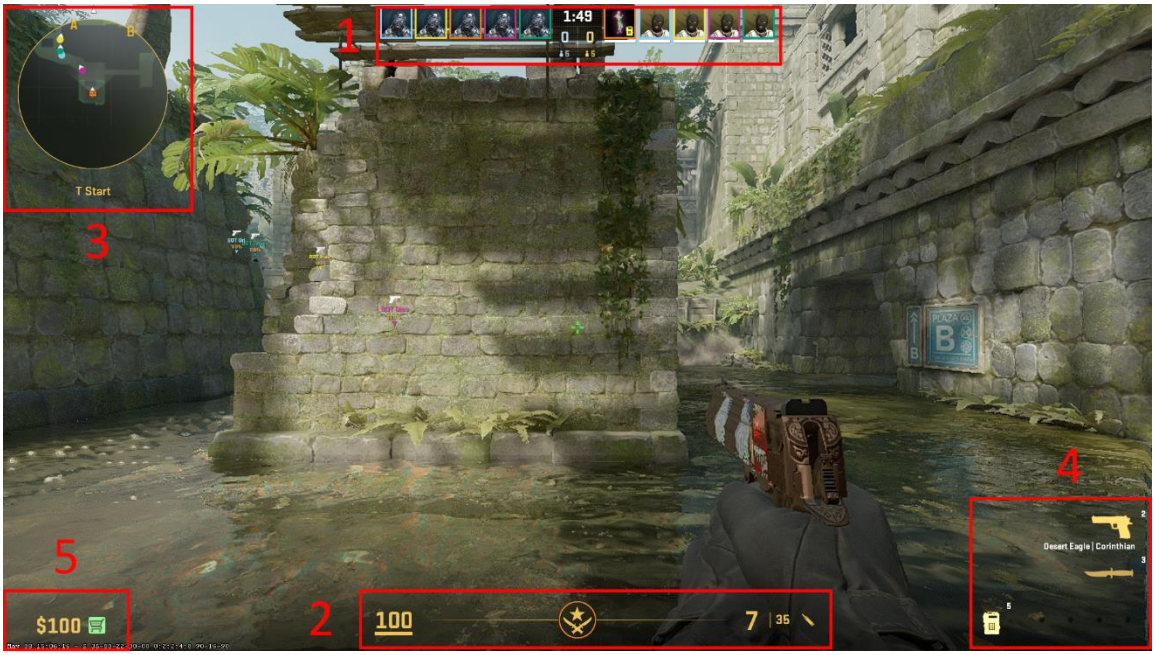

Figure 1. In-game HUD in CS2

As esports grew and tournaments became more popular, the demand for HUDs for spectators and broadcasters began to emerge. First of all was an observer HUD of spectator mode which can be seen once the spectator mode is enabled, it allows to see the gameplay of each of the players, their loadout and health, positions of all players on a map and switch point of views. With this data people can actually see what is going on in the game, it gives an insight of which team is leading (Carlsson & Pelling, 2015). On a figure 2 can be seen In-game spectator version of the HUD in the most recent Counter-Strike game, it is consisting of 1. Score bar including team players, how many of them alive at the moment, round timer, match score, players money and loadout. 2. Player's bar showing current health and ammo as well as additional statistic of ADR 3. Radar with mini map of the area, in a spectator mode it takes a square shape to display whole map and all players icons which have numbers, based on their observer slots. 4. Observed player's loadout consisting of guns that they have in possession and a bomb or defuse kit if they have one. 5. Current money amount (Counter-Strike 2, n.d.)

Figure 2. Spectator in-game HUD in CS2

For a long period of time basic in-game HUD with some small adjustments was used in the Major tournament broadcasts (Aman, 2022). Major tournaments are competitions which are held by game's developer – Valve, they are hosted usually twice per year (Young, n.d.). However, the idea of live game data visualization started progressing with the launch of the Counter-Strike Global Offensive – 6th game in the series. In 2017 on a PGL Major in Krakow was introduced a fully customized HUD for esports broadcast where all of the parts were customized and different, for example: the observed player bar and map has changed to increase viewer engagement and display vital information such as player health, armor, ammo, his statistics and nationality, before it was just indicating the name of observed player, the score bar was changed, displaying match score if it is a Best of 3 series, player bars on the sides were changed as well (PGL, 2017). The HUD can be seen on a figure 3.

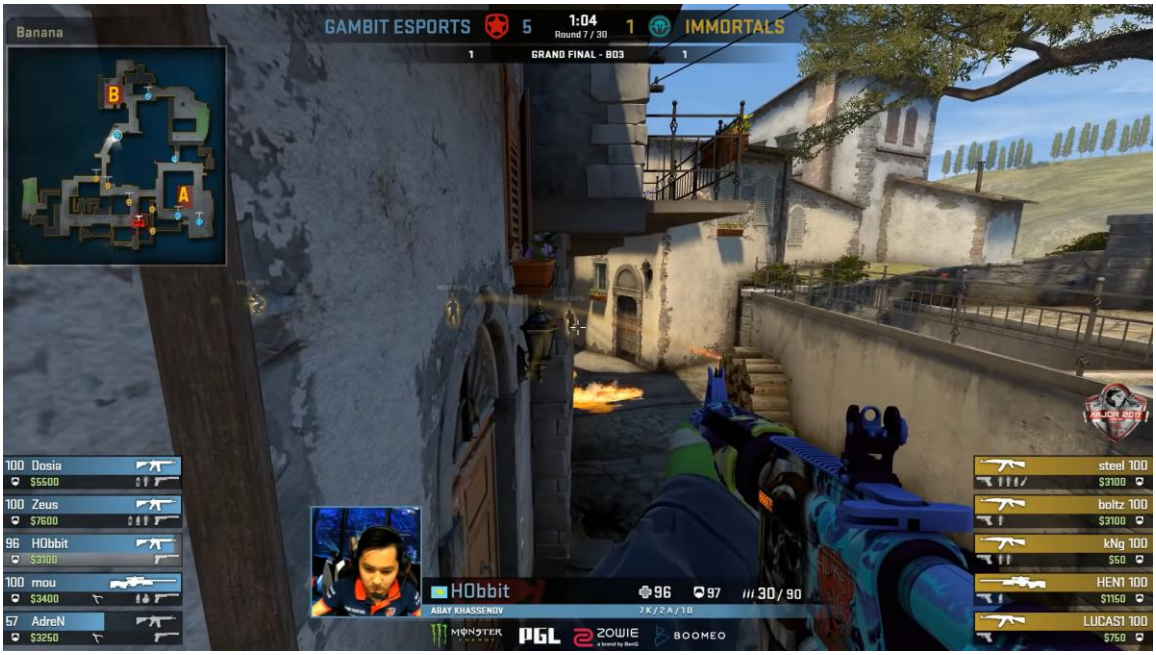

Figure 3. Fully customized HUD for esports broadcast. 2018. Adapted from [PGL,2017].

2.3.1 Existing trends and development throughout the years.

The development of HUDs for esports broadcasts of Counter-Strike has moved from simple and player oriented designs to more viewer-centered, data driven and complex overlays. During the time some of the details shaped standards for HUDs. For example, the player horizontal bars are located on top of each other and located on left and right sides of a screen. The colors which are used for team side identification, as in game, blue represents the counter-terrorists side and yellow – terrorists, however there are certain exclusions such as Blast Premier tournament where different approach have been used. HUDs began including team economy data, giving viewers a clearer understanding of team strategies and economic decisions. Other trends include radar enhancements, usage of graphical overlays and HUDs stylized in tournament colors.

3 Methodology

One of the objectives to reach the goal of the thesis is to conduct the qualitative research and gather insights which will act as a guidance in development of a HUD. In this chapter methodology and qualitative research process are discussed.

3.1 Research Design.

In the following chapter, the research design and methodology part of the thesis is discussed. The research design of this thesis combines qualitative research techniques with analysis of industry HUDs to develop and refine Heads-Up Display for Counter-Strike 2 (CS2) esports broadcasts. The qualitative approach emphasizes on understanding the needs of key stakeholders: viewers and broadcasters in particular, to ensure the HUD design aligns with their preferences.

Objectives of the Research Design:

O1: Identify key informational elements and design principles crucial for HUD development.

O2: Understand viewer and broadcaster preferences regarding HUD aesthetics and functionality.

O3: Evaluate the HUD prototype through feedback and practical testing.

3.2 Qualitative Research

In this thesis, author used qualitative research to get insights from the viewers and broadcasters on CS2 esports HUD. This type of research focuses on words rather than quantification in the collection and analysis of data (Bryman, 2008a: 366, as cited in Hammersley, 2013).

3.2.1 Interviews

Qualitative research was chosen to get detailed information from stakeholders about their experiences and expectations regarding esports HUDs. The study primarily relies on interviews with

esports viewers and broadcaster. To better understand spectators' expectations, preferred informational elements and aesthetics of the HUD, the viewers interviews were conducted. In the broadcaster interview, author focused on the HUD's usability, its visual appeal and design solutions as well as display of game data for efficiency in broadcasting.

The main goal of this research is to capture detailed insights and opinions on esports HUD from a wide range of perspectives. In this study, the non- random sampling method was used. According to Rai, N., & Thapa, B. (2015) non-random sampling method is a method where sample is not based on the probability but rather on experience, intention, and expertise of a sampler. In the research author used a combination of purposive and snowballing techniques. Purposive technique relies on the judgment of the researcher, while focusing on particular characteristics of a population that are of interest. Rai, N., & Thapa, B. (2015) Which in this case are various experiences with CS2 esports and HUDs. Snowballing technique refers to a method where study subjects recruit future subjects from among their acquaintances, colleagues or friends. Obilor, E. I. (2023). By using these sampling methods, author selected four participants, including three viewers with different experience and one CS2 esports broadcaster. By utilizing the snowballing method, author have chosen informants through his contacts to represent both experienced and inexperienced in the esports. For this reason, the choice was made in favor of interviewees, as one of them is inexperienced in esports, other one is an esports enthusiast and the third one have experience with CS2 esports.

3.2.2 Interviews procedure

In this research the author has conducted semi-structured interviews with participants. Semi-structured interview is more flexible than the structured, as it allows bringing forward additional questions in response to the interviewee. (Mashuri et al., 2022) This type of interviews was used in the research because of its relevance in gathering insights and detailed feedback. Interviews were conducted one-on-one. The interview questions were designed as open-ended to allow participants to talk freely about their opinion and experience as well as to capture a wide range of perspectives. The main topics that were discussed are: overall experience with CS2 esports, most valuable information on a HUD for viewers and broadcasters, characteristics of a HUD, importance of design solutions. Key informants were interviewed on a social platform Discord, utilizing voice call. Insights were grouped into categories such as "characteristics of an effective HUD", "essential HUD elements," "design preferences," and "technical requirements."

4 Findings

This chapter discusses findings from the qualitative research interviews, conducted with viewers and broadcaster regarding their experience with esports, CS2 and HUDs in general, as well as their expectations and requirements for an effective interface, enhancing experience of both stake-holders. The findings are divided respectively and grouped by the topic that was discussed.

4.1 Findings from the interviews

In this research author carried out semi-structured interviews with four key informants, three viewers and one esports broadcaster. Conducted interviews with viewers and broadcaster, gave valuable insights into the important characteristics, important data elements presented on a HUD, and design considerations required for a CS2 esports HUD. Key findings are outlined below.

4.1.1 Important characteristics of a HUD

For all viewer interviewees accessibility, clarity and readability were the main characteristics that a HUD should have:

“For me the most important thing in the interface is its simplicity and visuals. The interface should be accessible and easy to spend minimum of resources to understand what and where is located” (Participant A)

Additionally, all participants mentioned that HUD should be informative while having a room for the game and not be overloaded with unnecessary information. Broadcaster emphasized the ease of use and pleasing look of a HUD which also should not obscure key game elements.

4.1.2 Valuable data on HUD

One of the key topics that were discussed was valuable data that needs to be placed on HUD. For an experienced viewer the map score, radar, players’ loadouts are important.

“During the round itself, I want to see the score, just the map score, not the whole bo3 match, the mini-map, weapons and the player's loadout, armor and grenades, everything else I consider unnecessary. In between rounds you can show the economy of players and their KDA.” (Participant B)

However more inexperienced viewers highlighted kills, deaths, assists statistics, health points and loadouts of players as well as round distribution, players’ country flags and team affiliation.

Broadcaster interviewee highlighted: kills, deaths, ADR statistics, player economy, health points, utility, different layers of the map on the radar.

4.1.3 Importance of a HUD design

According to all participants design solutions play a vital role in the HUD perception.

“Very important! 70% of the user experience depends on it, so I would pay special attention to this issue” (Participant C)

Interviewees were highlighting that colors and design affect the "pleasantness" of the interface, it can help a lot in making information easier to understand, the picture should be visually beautiful and well put together. HUD should feature color combinations that are pleasing to the eye and informative at the same time, for example different teams displayed with different colors.

For broadcasting, participant D mentioned that:

“If the design is flawed and difficult to read, you cannot make fast paced comments and notes about the game. Colors need to have clear meaning behind them and need to be separate from each other, so I know what color means what.”

The findings show that an effective HUD for CS2 esports broadcasts must balance informative constituent with visual clarity and minimalism. The HUD should display various game statistics and data, such as kills, deaths, assists statistics, radar, score, players’ economy and utility. These insights form the basis for the design principles and were used in the concept creation phase.

5 Implementation

This chapter is discussing the implementation process of the thesis. It focuses on practical part: starting with concept creation after identifying key characteristics of a HUD, following with design creation in a interface design application, implementation of CS2 game state integration and afterwards coding HUD's base and animated elements.

5.1 Creation of a concept

The first step in the Implementation phase consists of concept creation based on the findings from the interviews and analysis of the HUD trends in CS2 esports. During the concept creation stage, it was decided that HUD should possess such structure:

1) Top bar or score bar, which includes team names and logos, round number, round timer and match score.

2) Players alive box, including amount of player alive in each team in the round.

3) Observed player bar, displaying player's nickname, picture, match statistics, health bar.

4) Side player bars, representing both teams' players in the game, their health points and statistics.

5) Map information bar, including format of the match and maps picked.

6) Radar

The initial concept can be seen on a Figure 4, black and grey bars represent potential size and placement of the elements.

Figure 4. Initial concept.

5.2 Creation of design

Following concept creation goes design implementation. Design was created in the interface design software called Figma. While creating visual interface author was guided by research findings and also implemented UX design, Multiple resources and data visualization theories. According to Jon Yablonski () user experience design is led by several laws which can be applied in creating interface. Jakob's law emphasizes usage of familiar design patterns to enhance user experience and reduce the learning curve, making HUD more accessible. By applying this law it was decided to use locations of the elements that are familiar to the potential user, this includes usage of blue and yellow shades representing terrorists and counter-terrorists sides, radar placement on the top left corner of the screen and location of the score bar on the top of the screen like in the standard game HUD, which can be seen on Figure 1, players' details were located by the sides one under another, following the accepted standard in esports broadcasting industry starting from the earlier versions of the Counter-Strike game series. The side bars and observed player bar displays only important game information such as player's kills, deaths, nickname, picture, health points, armor and loadout implying Hick's law, which suggests that decision time increases with the number and complexity of choices. The mentioned bars were placed and sized to not obstruct a vision of the most crucial part of the broadcast – actual gameplay. This goes in line with Wickens (2002) Multiple Resource Theory which states that different types of information should

be processed at different stages to minimize competition for the same cognitive resources. Another key point in designing is visual hierarchy, elements should be differentiated based on their priority, in the HUD observed player bar and score bar are prioritized; thus, they are biggest in size among other elements. Among the information that is being displayed the most important are team names, match score, players' health points, nickname and economy which were highlighted by using bigger font size and different color. The created concept design can be seen on Figure 5.

Figure 5. Design concept of a HUD

5.3 Game State Integration

This part includes implementation of CS2 Game State Integration to retrieve the live game data from the match. There is no CS2 GSI Documentation listed on a developers website, however the GSI from the previous version of the game, CS:Go is fully functional and documented. To receive the game data author created a configuration file which specifies which data should be retrieved and the destination where it should be sent. When CS2 runs, it checks for any GSI configuration files and starts sending files to specified endpoints, in this case it is a local Node.js server with port 3000. The server receives the data from the game as payloads in a JSON format. The game will send only payloads that were chosen in the configuration file. After receiving data in JSON format, it is written in the file and displayed on the server on another port which represents the

webpage of a HUD that can be viewed in a broadcasting software. On a figure 6 can be seen a configuration file example, including receivable data and explanations what does each chunk of information mean. Needed data is marked by '1'.

```
"Gamestate Integration Config File"
{
	"uri" "http://localhost:3000" //link where to send the data, it can also be a website.
	"timeout" "5.0" // Time, during which the game will wait for a response from the server.
	"buffer"  "0.1" // Time, during which the game will collect in-game events.
	"throttle" "0.1" // Time, during which the game will not send a load after receiving a response from the server.
	"heartbeat" "5.0" // The frequency in which the game will send a payload to the server, even when no game state change has occurred.
	"output"
	{
		"precision_time" "3"
		"precision_position" "1"
		"precision_vector" "3"
	}
	"data"
	{
		"map_round_wins" "1"          // Rounds won or lost.
		"map" "1"                     // Gamemode, current map, phase, score.
		"player_id" "1"               // SteamID of the observed player.
		"player_match_stats" "1"      // Scoreboard information of the observed player.
		"player_state" "1"            // State of the observed player, armor value, health points, etc.
		"player_weapons" "1"          // A list of all player's guns at his disposal and their state.
		"provider" "1"                // Information about the provider.
		"round" "1"                   // The phase of the round and the winning team.

		"allgrenades" "1"             // Grenades of all players.
		"allplayers_id" "1"           // SteamID of each player.
		"allplayers_match_stats" "1"  // Scoreboard data for all players.
		"allplayers_position" "1"     // Position of each player.
		"allplayers_state" "1"        // Player state of each player.
		"allplayers_weapons" "1"      // Weapons of all players.
		"bomb" "1"                    // Bomb information.
		"phase_countdowns" "1"        // Countdowns of all game phases.
		"player_position" "1"         // Position and directions of view of an observed player.
	}
}
```

Figure 6. GSI configuration file example

5.4 Combination of design and code

To display the updating data, which is received from the CS2 GSI, the web page was created by using HTML and CSS languages, implementing the JavaScript code to retrieve, and put needed information onto a page. The HTML is used to structure the content on a web page, CSS is utilized to format, style, and layout this content. For coding purposes was used Visual Studio Code software, which is a lightweight source code editor (Documentation for Visual Studio Code, 2021). Node.js is neither a library nor a framework; it is a runtime environment for executing JavaScript code outside of a web browser. It allows developers to use JavaScript to write server-side code, making it possible to build scalable network applications. Starting with JavaScript code, it firstly imports essential Node.js libraries and modules, such as express – web application framework, fs – file system module in order to read and write to the files, http module for creating a server, socketIo for real time communication between client and server.

```
const express = require('express');
const fs = require('fs');
const http = require('http');
const socketIo = require('socket.io');
const path = require('path');
```

Figure 7. Importing libraries

After importing libraries, the code initializes express application which is a framework that provides a set of features to develop web application (Express - Node.js Web Application Framework, n.d.). This application handles the get and post requests. The game state integration is sending the data payloads via POST request to the server, which is run on port 3000, when the game is played.

```
app.post('/', async (req, res) => {
    // Reading the request body
    const requestData = JSON.stringify(req.body, null, 2);

    // Writing data to a text file
    fs.writeFile('receivedData.txt', requestData, (err) => {
        if (err) {
            console.error('Failed to write to file', err);
            return res.status(500).send('Failed to save data');
        }
        console.log('Data written to file successfully');

        // Preparing the response
        const responseText = 'Accepted.';
        const buffer = Buffer.from(responseText, 'utf8');

        // Sending the response
        res.setHeader('Content-Type', 'text/plain; charset=utf-8');
        res.setHeader('Content-Length', buffer.length);
        res.send(buffer);
    });
});
```

Figure 8. POST request handling.

On a figure 8 can be seen a POST request handling, which reads the request body with the data from the match as a JSON file, converts it to a string format and writes it in receivedData text file. Afterwards the Socket.IO server watches the text file for changes, when this event is triggered, it converts the text into a JSON and emits it, so it can be referenced in the HTML code to display on

a web page. Following this, application handles GET requests and sends the HTML page with HUD layout to the server with port 8080.

```
<body>
    <div class="centered-rectangle">
        <div class="logo-container left-logo">
            <img src="" alt="Team 1 Logo" class="Team-T-logo">
        </div>
        <div class="logo-container right-logo">
            <img src="" alt="Team 2 Logo"class="Team-CT-logo">
        </div>
        <div class="team-score left-score"></div>
        <div class="team-score right-score"></div>
        <div class="side-rectangle left-rectangle">
            <div class="team-t"></div>
        </div>
        <div class="small-rectangle left-small-rectangle">
            <div class="match-score left-score1"></div>
            <div class="match-score left-score2"></div>
        </div>
        <div class="side-rectangle right-rectangle">
            <div class="team-ct"></div>
        </div>
        <div class="small-rectangle right-small-rectangle">
            <div class="match-score right-score1"></div>
            <div class="match-score right-score2"></div>
        </div>
        <div class="round-info">
            <div class="round-number"></div>
            <img src="/bomb.png" alt="Bomb Icon" class="bomb-icon" style="display: none;">
            <div class="round-timer"></div>
        </div>
    <div class="bomb-timer-container" style="display: none;">
        <div class="bomb-timer-background">
            <div class="bomb-timer-progress"></div>
        </div>
        </div>
    <div class="players-alive-container">
        <div class="players-alive-top">

        </div>
```

Figure 9. HTML structure code snippet

On a figure 9 can be seen HTML structure code snippet, each element on a HUD has its own division. These elements are stylized with CSS code and updated with JavaScript code which handles received data from the socketIO server.

The JavaScript code consists of the event listener that listens for DOMContentLoaded event and starts to receive data from socketIO server. DOM which stands for Document Object Model is an application programming interface for valid HTML document, it is defining the way how document is accessed and manipulated (Wood,L, et.al 1998). JavaScript can manipulate the DOM to

dynamically change the web page. DOMContentLoaded event is important in this scenario as it is triggered when the initial HTML document is loaded, without waiting for CSS styles and images (Document: DOMContentLoaded Event - Web APIs | MDN, 2024). The received data from socketIo is passed to the functions that are changing the DOM. For instance, the function which is shown at figure 10 is updating the HP bar of the currently observed player.

```
function updateHPBar(data) {
    if (!data || !data.observedPlayerHp || !data.observedPlayerTeam) {
            console.error('Invalid or incomplete player data for HP bar update');
            return;
        }
            const hpBar = document.querySelector('.hp-bar');
            const hpPercent =data.observedPlayerHp;
            hpBar.style.width = `${hpPercent}%`;
            hpBar.style.backgroundColor = data.observedPlayerTeam === 'CT' ? '#1C77FF' : '#DDBA00';
            document.querySelector('.observedplayerhp').textContent = data.observedPlayerHp;
            document.querySelector('.observedplayerarmor').textContent = data.observedPlayerarmor;
            const armorIcon = document.querySelector('.armor-icon');
            const armorHelmetIcon = document.querySelector('.armor-helmet-icon');
            if (data.observedPlayerarmor > 0 && data.observedPlayerHelmet) {
            armorHelmetIcon.style.display = 'block';
            armorIcon.style.display = 'none';
        } else if (data.observedPlayerarmor > 0) {
            armorIcon.style.display = 'block';
            armorHelmetIcon.style.display = 'none';
        } else {
            armorIcon.style.display = 'none';
            armorHelmetIcon.style.display = 'none';
        }
        }
```

Figure 10. Function updating the HP bar of the Observed player.

The function firstly is checking whether the received data is valid and have the necessary properties and if the data is invalid, it logs an error message and returns. If the data is valid, it selects the hp-bar element in the HTML document and applies the health points as a width percentage of the bar. The function checks the side of the observed player and changes bar's color to blue if the player is on counter-terrorists side and yellow if the player is on terrorists' side. Additionally, in this code the health points, armor points and armor icons are handled. To display the appropriate armor icon, it checks on a player's armor and helmet status. There are three conditions: player having both armor and helmet, player having only armor and player having none of the above. On a figure 11 can be seen the HP bar of the currently observed player with following conditions: player's health is 100%, player's armor is 100%, player has an armor and helmet.

Figure 11. HP Bar

This element's style is included in the CSS code presented on a figure 12.

```css
.hp-container {
    width: 485px;
    height: 40px;
    background-color: black;
    position: fixed;
    bottom: 87px;
    left: 717px;
}
.hp-bar {
    height: 100%;
    transition: width 0.5s ease;
    z-index: 1;
}
```

Figure 12. CSS code snippet of HP bar

Hp bar consists of two elements: hp container that is 485 pixels in width, 40 pixels in height, placed 87 pixels from the bottom side of the screen and 717 pixels from the left side respectively and has a black color background. Animated hp bar is placed within the container, on top of it and has a 0,5 second ease transition.

6 Evaluation

In this chapter the study discusses the evaluation that was implemented via HUD testing sessions in the broadcasting software and conducted interviews with key informants. It covers the process of HUD usability testing in one of the popular broadcasting softwares – OBS as well as interviews design, sampling method and interviews findings which will serve as a base for the discussion and conclusions.

6.1 Testing

During the development, HUD was overtaking a testing process to identify any potential technical issues and evaluate its' performance. This process included functional tests, ensuring that all data including player's health points, statistics, score, round timer and other elements display accurate live data and response if the data is changed during the match. The usability tests which assess the ease of use of a HUD in the broadcasting software in addition to readability and visibility of all the elements. The process was structured as follows: the servers were enabled, fetching the data from the game, in CS2 the demo of the esports match was played to simulate the realistic conditions of live data change. After this the web page with the HUD was added to the OBS broadcasting software setup, ensuring everything is working. The testing showed that the HUD is functioning well and displays all needed data which can be seen in Figure 7.

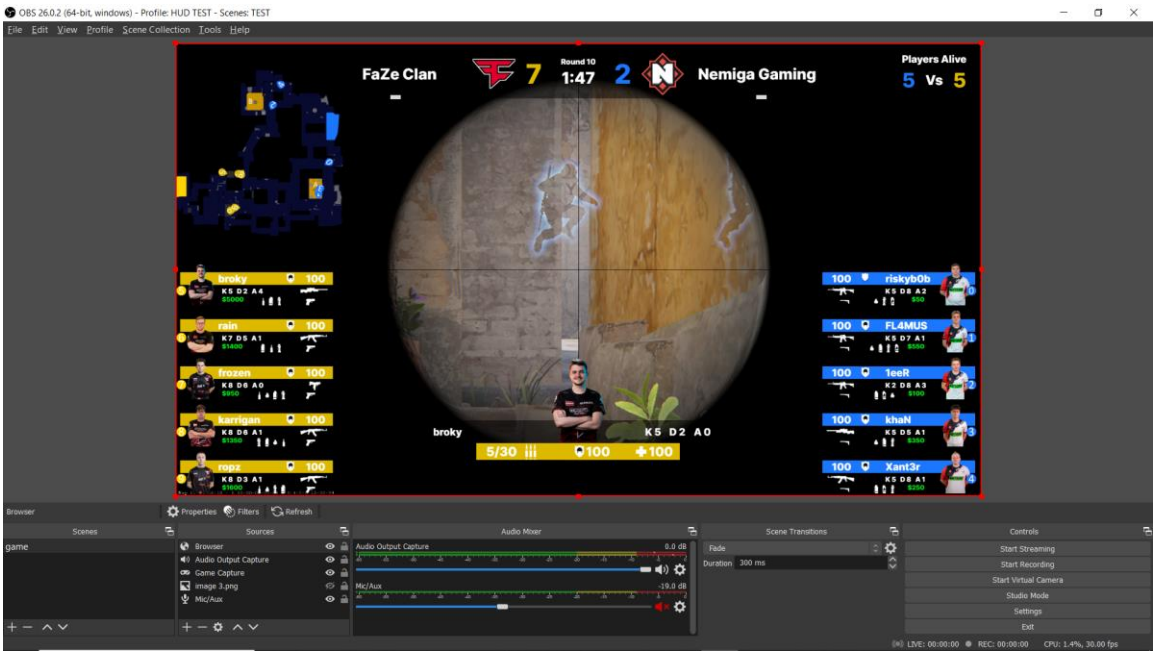

Figure 7. Testing in broadcasting software.

6.2 Interviews

After the testing phase, author conducted interviews and measured evaluation. The interviews were designed as structured. The sampling method was a combination of purposive and snowballing techniques. To identify the informants, author used a snowballing method, thus the participants were familiar to the author and were chosen specifically as being representative of a diverse esports audience, from beginners to professional commentators. The interviews were one-on-one. Questions were open-ended and were focused on gathering feedback on developed HUD from the viewers and broadcaster. Interviews were conducted on a social platform Discord, using voice call.

6.2.1 Findings from the interviews

The interviews provided qualitative feedback on how well the HUD was developed and if it meets the needs of esports viewers of any experience and broadcasters. All interviewees mentioned that the HUD is accessible. I like that the HUD is understandable and highlights important game data with font sizes and different colors" (Participant A). One of the participants stressed the color choice, mentioning that it is making information clearer and interface more intuitive.

On the other hand, participants expressed their opinion on future developments and what they would like to change. For instance, participant B mentioned removing the picture of observed player and kda statistics as well as map names that are being played in the match: "I would remove the kda and map picks, I don't understand why they are shown in the round, it is possible to display it in the end of the game. Another interviewee highlighted importance of the icons: "I would add the hp icon on the bar, because it would be more clear what do numbers represent".

7 Discussion and conclusions

This chapter discusses the entire research process, highlighting successes, limitations and explores potential areas for future development of a CS2 esports HUD.

Qualitative research interviews provided good insights which were utilized as guidelines in a development stage and the feedback received from the evaluation interviews indicates that the project objective was achieved. Interviewees highlighted that the developed HUD makes it easier to follow the esports broadcast and understand CS2 game without utilizing all cognitive resources. The design's minimalistic approach, emphasizing clarity and intuitiveness. Applying UX design theories was particularly effective in maintaining the balance between functionality and aesthetic appeal, which is important to keep viewer attention on a broadcast. This approach ensured that the HUD's design was familiar enough to be accessible to new viewers while still offering informative live match data. The positive responses from interview sessions highlight the success of these design choices. The integration of real-time game data through CS2's Game State Integration was a critical component of the HUD's functionality. This feature allowed the HUD to display live information accurately, which is essential for maintaining the integrity of the broadcast. The performance testing confirmed the HUD's reliability and responsiveness, crucial attributes for maintaining a seamless viewing experience. From the broadcasters' perspective, the HUD met the requirements of being easy to use and visually appealing enabling commentators to make fast paced comments about the game. The project question of How to develop effective HUD for esports CS2 broadcast was answered.

The future directions for development include addition of various extra animations and details on a HUD, for instance bomb timer color change as it reaches the end, grenades animations on a radar map. Technical implications may include optimization and better structurization of a code.

To conclude, the project question of How to develop effective HUD for esports CS2 broadcast was answered, the developed HUD for CS2 esports broadcasts has successfully met its initial objectives by enhancing viewer experience and supporting broadcaster needs. However, the findings from this project highlight the importance of continuous development in this area.

References

Aman. (2022, May 28). Every CS Major winning moment (2001-2022) [Video]. YouTube. https://www.youtube.com/watch?v=31T9JkLMZ3s

Ashley_Buckwell. (2024, April 1). The history of Counter-Strike. Acer Corner. https://blog.acer.com/en/discussion/1371/the-history-of-counter-strike

Beal, V. (2021, May 24). HUD – heads up display. Webopedia. https://www.webopedia.com/definitions/hud/.

Bkid. (2019, July 29). Game State Integration: A Very Large and In-Depth Explanation. https://www.reddit.com/r/GlobalOffensive/comments/cjhcpy/game_state_integration_a_very_large_and_indepth/.

Bousquet, J., & Ertz, M. (2021). ESports. In Advances in e-business research series (pp. 1–24). https://doi.org/10.4018/978-1-7998-7300-6.ch001.

Carlsson, C., & Pelling, A. (2015). Designing spectator interfaces for competitive video games. https://odr.chalmers.se/items/54f90e30-222b-405b-9554-3c676fde76bf

Counter-Strike: Global Offensive Game State Integration - Valve Developer Community. (n.d.). Developer.valvesoftware.com. https://developer.valvesoftware.com/wiki/Counter-Strike:_Global_Offensive_Game_State_Integration.

Denscombe, M. (2007). The good research guide: For small-scale social research projects. (3rd ed.). Birkshire: Open University Press.

Document: DOMContentLoaded event - Web APIs | MDN. (2024, January 15). MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/API/Document/DOMContentLoaded_event

Documentation for Visual Studio code. (2021, November 3). https://code.visualstudio.com/docs

ESL Archives. (2018, March 5). CS:GO - FaZe vs. Fnatic [Train] Map 5 - GRAND FINAL - IEM Katowice 2018 [Video]. YouTube. https://www.youtube.com/watch?v=pDFdhr1HIwc

eSports History: How it all began. (2024, April 16). ISPO.com. https://www.ispo.com/en/sports-business/esports-history-how-it-all-began.

Express - Node.js web application framework. (n.d.). https://expressjs.com/

Hamari, J., & Sjöblom, M. (2017). What is eSports and why do people watch it? Internet Research, 27(2), 211–132.

Hammersley, M. (2013). What is Qualitative Research? https://doi.org/10.5040/9781849666084

Leroux-Parra, M. (2020, August 2). Esports Part 1: What are Esports? Harvard International Review. https://hir.harvard.edu/esports-part-1-what-are-esports/.

Morris, D. (2024, April 16). Counter Strike 2 game modes explained – every mode in CS2. Esports.net. https://www.esports.net/news/counter-strike/cs2-game-modes/

Obilor, E. I. (2023). Convenience and purposive sampling techniques: Are they the same. International Journal of Innovative Social & Science Education Research, 11(1), 1-7.

PGL. (2017, July 23). PGL Major Kraków 2017 | Grand Final | Gambit vs Immortals | Game 3 [Video]. YouTube. https://www.youtube.com/watch?v=w2MdCEZSBtw

Rai, N., & Thapa, B. (2015). A study on purposive sampling method in research. Kathmandu: Kathmandu School of Law, 5(1), 8-15.

Sacco, D. (2016, December 6). A brief history of esports and video games. British Esports Federation. https://britishesports.org/the-hub/about-esports/a-brief-history-of-esports-and-video-games/.

Sandmann, L. (2024, May 1). What is a first person shooter (FPS) game? ExitLag. https://www.exitlag.com/blog/what-is-first-person-shooter/

Schwabish, J. A. (2021). Better data visualizations: a guide for scholars, researchers, and wonks. Columbia University Press.

Šimić, I. (2024, April 10). PGL Major Copenhagen 2024 records 1.8m viewers for debut CS2 Major. Esportsinsider. https://esportsinsider.com/2024/04/pgl-major-copenhagen-2024-viewership

Wickens, C. D. (2002). Multiple resources and performance prediction. Theoretical Issues in Ergonomics Science, 3(2), 159–177. https://doi.org/10.1080/14639220210123806.

Wood, L., Le Hors, A., Apparao, V., Byrne, S., Champion, M., Isaacs, S., ... & Wilson, C. (1998). Document object model (dom) level 1 specification. W3C recommendation, 1.

Yablonski, J. (2020). LAWS OF UX: design principles for persuasive and ethical products. (O'reilly Media, Inc, Usa).

Young, S. (n.d.). Professional Counter-Strike: An analysis of media objects, esports culture, and gamer representation. The Aquila Digital Community. https://aquila.usm.edu/dissertations/1886/

Appendix 1.

Interview questions.

Interview questions for viewers.

How familiar are you with esports and CS2?

Do you know what HUD is and the difference between in-game HUD and esports HUD?

What experience do you have with esports custom HUD?

What do you think are the most valuable characteristics of a HUD for you as a viewer?

Which data would you like to see on a HUD on CS2 broadcast?

How important color schemes and design solutions in an esports HUD for you?

What is your opinion on this HUD?

Does it meet the requirements of viewers?

What do you like in this and why?

What don’t you like and why?

What do you think can be developed and added in this HUD?

Interview questions for broadcaster.

How familiar are you with esports and CS2?

Do you know what HUD is and the difference between in-game HUD and esports HUD?

What experience do you have with esports custom HUD?

Briefly describe your experience.

What are the most valuable characteristics of a HUD for you as a representative of broadcasting company?

What information would you like to see on a HUD while broadcasting?

How important are colors and design solutions on a HUD for you?

What is your opinion on this HUD?

Does it meet the requirements of broadcast?

What do you like in this and why?

What don’t you like and why?

What do you think can be developed and added in this HUD?

Appendix 2.

Interview transcripts.

Interview with Participant A.

R – Researcher

PA – Participant A

R: Hello, I'm going to ask you some questions about HUD, your experience, preferences and important data.

R: Let's start with your experience, could you please tell me how familiar are you with esports, in particular with CS2 tournaments?

PA: I have almost no experience with CS2 esports. And I've watched esports broadcasts only a few times and a long time ago. I'm rather new to this and have only played the CS2 myself.

R: Okay, let's move on to CS2 Heads-Up-Display, do you have any experience with custom esports HUDs?

PA: As I said, I've seen custom HUDs a long time ago and only a few times, so I can't describe the experience, as there was practically none. But I can describe my personal experience of using the interface in CS2 during games.

R: Yes, that would be great. What's your opinion about the regular CS2 HUD?

PA: The basic interface seems to me clear and sufficient. It has everything important: what weapons you have, how much hp and armor you have, how much money, map with players and markings, game score, time until the end of the round. It also displays kills during the round, as well as quick messages in the chat, which can also sometimes help during the game. The positions of elements are also clear. In general, the interface of the game is looking well.

R: What do you think are the most valuable characteristics of a HUD for you as a viewer in general?

PA: For me the most important thing in the interface is its simplicity and visuals. The interface should be accessible and easy to spend minimum of resources to understand what and where is located and what different symbols and abbreviations mean.

R: That makes sense. Which data would you like to see on a HUD on CS2 broadcast?

PA: I've already said the important information that should be displayed, and it's well implemented in the basic CS2 interface. For broadcasts, though, the interface needs to change a little because viewers need information about all players in the match. Kills, deaths, assists, health, weapons, and money should be displayed for all players. I'd also add the ability to see which rounds each team won so viewers can follow the match progression.

R: Okay, I see, so you value clear and easy HUD?

PA: Yes.

R: I see, and what about design solutions? For example, colors of a HUD or icons are they important to you?

PA: It's really important because it affects how I perceive the information. Like everyone else, I don't want to waste time figuring out where information is or what does some symbol mean. Colors and design also impact the overall appealing of the interface. Colors can make information easier to understand, like using different colors for different teams.

R: Understood. Thank you for sharing your thoughts!

Interview with participant B.

R – Researcher

PB – Participant B

R: Thank you for agreeing on this interview about HUDs in esports. Could you tell me about your experience with CS2 esports?

PB: Yes, I have experience of watching a lot of tournaments.

R: So, you have seen the custom made esports HUDs?

PB: Yes, almost all broadcasting studios have their own custom HUD, which also changes from tournament to tournament.

R: Given your experience, what characteristics do you find most important in a HUD?

PB: I would like to see, first of all, a HUD that is not overloaded with unnecessary information. It should not interfere with the viewing of the game and should show only the necessary information. Also, an important point for me is the difference from the standard HUD in terms of design.

R: Can you specify what information you consider necessary during a match?

PB: During the round, I want to see the score, just the map, not the whole match if it's a best-of-three, for example, the mini-map, weapons, and the status of the player's health, armor, and grenades. Everything else I consider unnecessary. In between rounds, you can show the economy of players and their KDA.

R: Okay, so you would like to have focus on a gameplay itself, for example players' pov and their moves on a map?

PB: Yes, for me it is the most important thing.

R: Could you tell me if the design like color schemes, elements positions and shapes are crucial for you?

PB: As I said, for me, the design is important. The picture should be visually beautiful and well-composed. Colors are not so important, but of course, I would like to have color combinations that are pleasing to the eye.

R: Okay, got it, thank you for sharing your insights!

Interview with participant C.

R – Researcher

PC – Participant C

R: Hello, I'm going to ask you a few questions about HUD, your experience, preferences, and important data. Let's start with your experience, how familiar are you with esports, in particular with CS2 tournaments?

PC: I have no experience, did not watch broadcasts, only had a glimpse at some.

R: So, you have seen some of the broadcasts and HUDs which were there?

PC: When I enter the game, the first thing it offers is to watch the current broadcasts that are going on now. A couple of times I've seen how it looks like but didn't really look into it.

R: Do you have any specific preferences for HUD in esports broadcasts?

PC: No, since I don't watch them

R: Okay, if you were to get into watching esports broadcasts, what kind of HUD would you prefer?

PC: If I was into broadcasts, I would like to see stats from the game, but not an overloaded screen, where there is too much information on the screen and no room for the game.

R: What specific information do you think should be displayed on a HUD?

PC: A map with the location of the players, what stats on kills and deaths each player has, player info like country flag, team, and stuff like that.

R: How important do you think the design and layout of the HUD is?

PC: Very important! 70% of the user experience depends on it, so I would pay special attention to it.

R: Thank you for sharing your thoughts!

Interview with participant D.

R – Researcher

PD – Participant D

R: Hello, I'm going to ask you some questions about HUD, your experience, preferences, and important data. Let's start with your experience, how familiar are you with esports, in particular with CS2 tournaments?

PD: I have an experience with esports, I'm broadcasting and commentating CS2 matches at …

R: What are the most valuable characteristics of a HUD for you as a representative of broadcasting company? For example, ease of use or optimality of layout?

PD: Ease of use is a must as there are already so many other things that can go wrong. Second important thing is the visual outlook of the HUD and how pleasing it looks.

R: What information would you like to see on a HUD while broadcasting? It can be already on a usual HUD, or you can add any extra info which is not present.

PD: Kills, deaths, ADR, player money, HP, Utility, different layers of the map, if some are on top of each other, Detailed stats of a player live from the game that is being played as additional information.

R: How important are colors and design solutions on a HUD for you?

PD: Colors and design are extremely important as they play a massive part in the experience. If the design is flawed and difficult to read, you cannot make fast paced comments and notes about the game. Colors need to have clear meaning behind them and need to be separate from each other, so I know what color means what.

R: Thank you for the interview!

Evaluation interviews.

Interview with Participant E

R – Researcher

PE – Participant E

R: Hello, thank you for participating, I'm evaluating the new developed HUD, I am going to show you a HUD and ask you some questions about it. What can you say about this HUD?

PE: Hello, I think the HUD on picture is easy to understand. All the relevant information is shown like hp, map, players guns and score.

R: Could you tell me what do you think is good in this HUD and why so?

PE: The simplicity of design is good, unnecessary information doesn't clutter it, so it's easy to focus on the game.

R: What do you think about the design?

PE: I like that the HUD is understandable and highlights important game data with font sizes and different colors.

R: What do you think is bad in this HUD and why?

PE: In my opinion it could be more creative, also the colors are not that bright. Maybe the player statistics or basic info is not enough, I would add more player stats and details like country flags and team logos to the players bars.

R: Okay, moving forward, what can be developed in the future for this HUD?

PE: As I said I would add like ADR of the players or other detailed stats which are not present on a default HUDs and display the round distribution, like which team won which round, it would be helpful in my opinion. Also, I'd change colors to brighter ones.

R: Thank you for your feedback!

Interview with participant F.

R – Researcher

PF – Participant F

R: Hello, thank you for participating, I'm going to show you a picture of a new HUD and ask you a few questions about it. What are your thoughts on this HUD?

PF: I like it, looking good, it has clear layout and simple, but the white color on yellow is not really readable.

R: Can you tell me what do you think is done good in this HUD?

PF: I think that the best thing in this HUD is structure, I like that everything is divided into sections and well placed.

R: What do you think is bad in this HUD and why?

PF: I don't like the white text on a yellow background, and I would show players alive under the player details in the middle. Otherwise, it's looking pretty good.

R: What can be developed in the future for this HUD?

PF: I'd like to see player's national flags and tournament phase above the radar, adding any other things will be unnecessary.

R: Thank you for the interview!

Interview with participant G.

R – Researcher

PG – Participant G

R: Hello, thank you for participating in the interview, I'm going to show you a picture of a new HUD and ask you a few questions about it. What are your thoughts on this HUD?

PF: It doesn't look bad, the elements are well placed, especially the player info at the bottom, player pictures and nicknames make it easier to identify them, the score and round number are easy to read because of the font sizes.

R: Can you tell me what do you think is done good in this HUD?

PF: Simple and understandable design, with all the necessary information about the game is presented like round, time, kills, assists, weapons, money, hp, armor, colors are pleasant and eye-catching.

R: What do you think is bad in this HUD and why?

PF: I don't like the position of the players alive because it can be on top of the kills and overlap. I'd also frame the player I'm following, like on the sides, highlight the player whose gameplay is shown.

R: What can be developed in the future for this HUD?

PF: You can add more detailed information about economics, like players minimum money in the next round or equip value of each team, also you can add map picks if it is a bo3 match. And if possible, display the defusing kits if a player has it.

R: Okay, got it. Thank you for the interview!

Interview with participant H.

R – Researcher

PH – Participant H

R: Hello, thank you for joining, I will show you the new developed HUD and I would like to hear feedback from you! It looks like this. What do you think of it?

PH: Looks fine, everything needed for commentating is present and the screen is not overloaded.

R: Okay, let's move on to the usage, let me demonstrate the HUD setup in the broadcasting software, so here is the folder with the HUD, to run it, you need to install Node.js, put the config file into the game folder and open the executable file of the HUD and add the web page as a source in the broadcasting software. To add player pictures, they need to be placed into the public folder and that is it. Could you tell me if the process was easy and what are your thoughts about it?

PH: I think the setup looks pretty easy, other HUDs have kinda same setup, but there is also a customization window. For me it won't be a problem to install it, but the detailed guide is needed.

R: Understood, what do you think are the good points in this HUD?

PH: It's easy to read and looks familiar. Hud efficiently displays everything. The colors and fonts are good. In my opinion observed player details, hp and statistics were well designed.

R: Okay and could you please share your opinion on what is done badly in this HUD?

PH: I think minimap should be bigger and more contrast, so the map and players can be seen clearly, the player's stats and hp can be slightly bigger, if the player is dead, it can be displayed with the different color or hide the player picture, because now only the hp is decreasing and its hard to distinguish who is dead and who is alive.

R: What improvements or new features would you like to see in future?

PH: In my opinion ability to customize it would make the HUD better, for example changing colors and map picture or excluding some elements to change the style of it. The killfeed can also be changed to a custom one.

R: Thank you for your feedback!