O_HAI 4 Games - D4.4. Case Study 4 - HoloGame

Schatten, Markus

O_HAI(4)Games

Orchestration of Hybrid Artificial Intelligence Methods for Computer Games

Case Study 4 - HoloGame

This project was funded by the Croatian Science Foundation

Principal investigator:

Markus Schatten

lab

PUBLISHED BY ARTIFICIAL INTELLIGENCE LABORATORY,
FACULTY OF ORGANIZATION AND INFORMATICS, UNIVERSITY OF ZAGREB

HTTP://AI.FOI.HR/OHAI4GAMES

Some of the results presented in this deliverable have been published in [35].

Technical Report No. AIL202401 – First release, February 2024, edit April 2024

Document compiled by: Markus Schatten with inputs from other project team members

This work has been supported in full by the Croatian Science Foundation under the project number IP-2019-04-5824.

We would like to especially thank students which participated in the development and testing of parts the console during the Game Development Platforms course at the Faculty of Organization and Informatics. Details about their participation and work can be found on the following GitHub repositories: https://github.com/AILab-FOI/PRRI-HoloGameV2023/wiki and https://github.com/AILab-FOI/PRRI-HoloGameV2024/wiki/Projektni-plan.

Contents

1. Project Description

1.1 Abstract

Hybrid artificial intelligence (HAI) methods, which can be defined as the orchestration of complementary heterogeneous both symbolic and statistical AI methods to acquire more precise results, are omnipresent in contemporary scientific literature. Still, the methodology of developing such systems is in the most cases ad-hoc and depends from project to project. Computer games have always been connected to the development of AI. From the earliest chess minmax algorithm by Claude Shannon in 1949 to the more recent AlphaGo in 2015, computer games provide an ideal testing environment for AI methods. Similarly, AI has always been an important part of computer games, which have often been judged by the quality of their AI and praised if they used an innovative approach. Computer games allow us to test AI methods, not only for fun and leisure, but also for numerous other fields of human activity through the fields of serious games and gamification. The project proposes to establish a new framework for the orchestration of hybrid artificial intelligence methods with a special application to computer games. Therefore an ontology of hybrid AI methods as well as a meta-model shall be developed that would allow for creating models (ensembles) of hybrid AI methods. This meta-model would be implemented into a modular distributed orchestration platform which would be further enriched with a number of modules to be tested in four gaming related environments: (1) MMORPG games, (2) gamified learning platform, (3) serious game related to autonomous vehicles and (4) a game for a holographic/volumetric gaming console which would also be developed during the project.

1.2 Introduction

The application of HAI which can be defined as the orchestration of heterogeneous artificial intelligence (AI) methods including both statistical and symbolic approaches in various domains is omnipresent in current scientific literature. It is largely overlapping with the term hybrid intelligence (HI) that has been defined as *"the combination of complementary heterogeneous intelligences (...) to create a socio-technological ensemble that is able to overcome the current limitations of (artificial) intelligence."* [10]. HI lies at the intersection of human, collective and artificial intelligence, with the intent of taking the best of each.

There have been numerous studies recently addressing issues related to HAI and HI methods in a multitude of application domains including but not limited to land-slide prediction [22], drug testing [8], forecasting crude oil prices [48], prediction of wildfire [17], evaluation of slope stability [20], modeling of hydro-power dam [6], wind energy resource analysis [13], industry 4.0 and production automation [3], airblast prediction [1], heart disease diagnosis [24] and these are just a few references from 2018 until the time of writing this proposal. Most of these and such studies report building HAI systems by combining various AI methods to acquire better and more precise results. However, when it comes to methodology of the actual orchestration of HAI methods the usual approach is ad-hoc and depends from project to project. The lack of methodology in orchestrating HAI shall be addressed in the proposed project.

In a previous project sponsored by the Croatian Science Foundation (Installation Project No. HRZZ-UIP-2013-11-8537 entitled Large-Scale Multi-Agent Modelling of Massively Multi-Player On-Line Playing Games - ModelMMORPG - see [42] for details) a comprehensive methodology for modelling large-scale intelligent distributed systems has been developed that includes a graphical modelling tool and code generator (described in [41] and in more detail in [27]). The implemented toolset allows for modelling complex multi-agent organizations and could be applied to numerous applications domains [37, 38]. Herein, we would like to apply and incorporate this methodology to the development of the HAI orchestration platform.

Computer games have always been connected to the development of AI. From the earliest chess minmax algorithm by Claude Shannon in 1949 to the more recent AlphaGo™ in 2015, computer games provide an ideal testing environment for AI methods. Similarly, AI has always been an important part of computer games. Computer games have often been judged by the quality of their AI and praised if they used an innovative approach like the ghosts in Pacman™ which had individual personality traits (1980), Creatures™ which used neural networks for character development (1996), Black & White™ which used the belief-desire-intention (BDI) model (2000), F.E.A.R.™ which used automated planning algorithms (2005) and many others (see [50, pp. 8–15] for a very detailed overview). Artificial intelligence in games is not only used for non-player character (NPC) or opponent implementation, but also for various other parts of games [50, pp. 151–203] including but not limited to **generation of content** (graphics including levels and maps, sound, narratives, rules and mechanics or even whole games like the Angelina game-generating system [9]), **player behaviour and experience modeling** [50, pp. 203–259], as well as **bot development and automated game testing** [50, pp. 91–151]. Due to their complex nature and endless possibilities of creative design, computer games present us with an excellent opportunity to study the orchestration of HAI in various scenarios – not only for fun and leisure but also for other domains in form of serious games and/or gamification.

In the previously mentioned ModelMMORPG project, we have already used an open source massively multi-player on-line role-playing game (MMORPG) called The Mana World (TMW) for which we have implemented a high-level interface to test intelligent agents playing the game. Additionally a number of connected game quests have been developed for various scenarios which allowed us to build an automated game testing system [40]. Herein we would like to use this interface to test orchestrated HAI methods, but also develop other testbeds for the planned platform.

Therefore, the main contribution of the proposed project shall be: (1) a comprehensive framework for the orchestration of hybrid artificial intelligence methods for computer games allowing to define models of HAI for various purposes, (2) an open source distributed cloud platform that will allow to implement such models based on existing HAI methods and connect them directly from game development platforms, (3) a set of best practices in developing HAI ensemble models tested in at least four specific testbeds.

1.3 Team Members

Markus Schatten (Principal investigator)
Head of Artificial Intelligence Laboratory,
Faculty of Organization and Informatics,
University of Zagreb

Jaime Andres Rincon Arango
Grupo de Tecnología Informática e Inteligencia Artificial (GTI-IA),
Departamento de Sistemas Informáticos y Computación,
Universitat Politècnica de València

Bogdan Okreša Đurić
Artificial Intelligence Laboratory,
Faculty of Organization and Informatics,
University of Zagreb

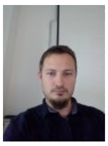
Carlos Carrascosa
Grupo de Tecnología Informática e Inteligencia Artificial (GTI-IA),
Departamento de Sistemas Informáticos y Computación,
Universitat Politècnica de València

Damir Horvat
Department of Quantitative Methods
Faculty of Organization and Informatics,
University of Zagreb

Vicente Julian
Grupo de Tecnología Informática e Inteligencia Artificial (GTI-IA),
Departamento de Sistemas Informáticos y Computación,
Universitat Politècnica de València

Tomislav Peharda
Artificial Intelligence Laboratory,
Faculty of Organization and Informatics,
University of Zagreb

Glenn Smith
College of Education,
University of South Florida



Igor Tomičić
Center for Forensics, Biometrics and Privacy
Faculty of Organization and Informatics,
University of Zagreb

Neven Vrček
Department of Information Systems Development
Faculty of Organization and Informatics,
University of Zagreb

2. HoloGameV

In this chapter we present the development of a holographic / volumetric (Ho/Vo) gaming console, integrating gesture recognition and cognitive agents. The work outlined is the culmination of the O_HAI 4 Games project, showcasing a novel console design that incorporates a retrofuturistic-style casing, a holographic fan display, a microcontroller, and various peripherals including speakers, a microphone, and a camera. Central to this innovation is the cognitive agent B.A.R.I.C.A., which demonstrates advanced human-computer interaction through gesture-based gameplay. The development process adheres to a design science approach, reflecting a synergy of holographic technology, gesture recognition, and artificial intelligence in gaming. This chapter details the development journey, technical specifications, and potential impacts of this novel console in the realm of interactive gaming.

2.1 Introduction

The domain of interactive gaming has continually evolved, driven by advancements in technology and innovative approaches to gameplay. In this study, we are trying to add a small leap in this evolutionary path. This chapter discusses the development and implementation of a holographic gaming console, a key outcome of the O_HAI 4 Games project. Central to this endeavour is the integration of Beautiful ARtificial Intelligence Cognitive Agent (B.A.R.I.C.A.), a cognitive agent previously detailed in [34, 43, 45] and developed at the Artificial Intelligence Laboratory of the Faculty of Organization and Informatics, with holographic and gesture recognition technologies.

The concept of quasi-holographic (holographic) displays in gaming is not new and has been explored in various capacities as highlighted in [35]. However, the integration of these displays with cognitive agents and gesture recognition systems presents a novel approach to enhancing user experience and interaction. Our development follows a design science methodology, similar to those described in [36], ensuring a systematic and user-centred approach in console creation.

Gesture recognition technology plays a pivotal role in our console, allowing for a more immersive and intuitive user interface. This integration is part of a broader trend in gaming technology, aiming to create more natural and engaging user experiences.

This chapter aims to detail the development process, the technical challenges overcome, and the potential impacts of the holographic gaming console on the gaming industry and human-computer interaction. The significance of this development is not only in its technological innovation but also in its potential to redefine gaming experiences, as we step into a new era of interactive entertainment.

The rest of this chapter is organized as follows: firstly we provide an overview of related work and provide an introduction to holographic technology, gesture recognition and cognitive agents. Afterwards we describe the development process and present the architecture of the developed console. Then we discuss our experiences and lessons learned. In the end we draw our conclusions and provide guidelines for future research.

2.2 Related Work

The advancement of interactive gaming technologies has been significantly shaped by pioneering research and development in the fields of holographic displays, gesture recognition, and cognitive agents. This section reviews the existing literature and past work that provides a foundation for the current project. Each subsection provides specifics of the technologies integrated into our holographic gaming console, identifying key contributions and how they influence our approach.

2.2.1 Holographic Technology

Holographic technologies have seen varied applications in different domains, from advertising to user interfaces, providing a depth of interaction that traditional displays cannot. A hologram is a 3D visual representation of objects that are visible from all angles. Currently, there are a variety of promising technologies [5, 7, 11, 15, 23, 30, 47]. Commercially, several technologies mimic this holographic effect.

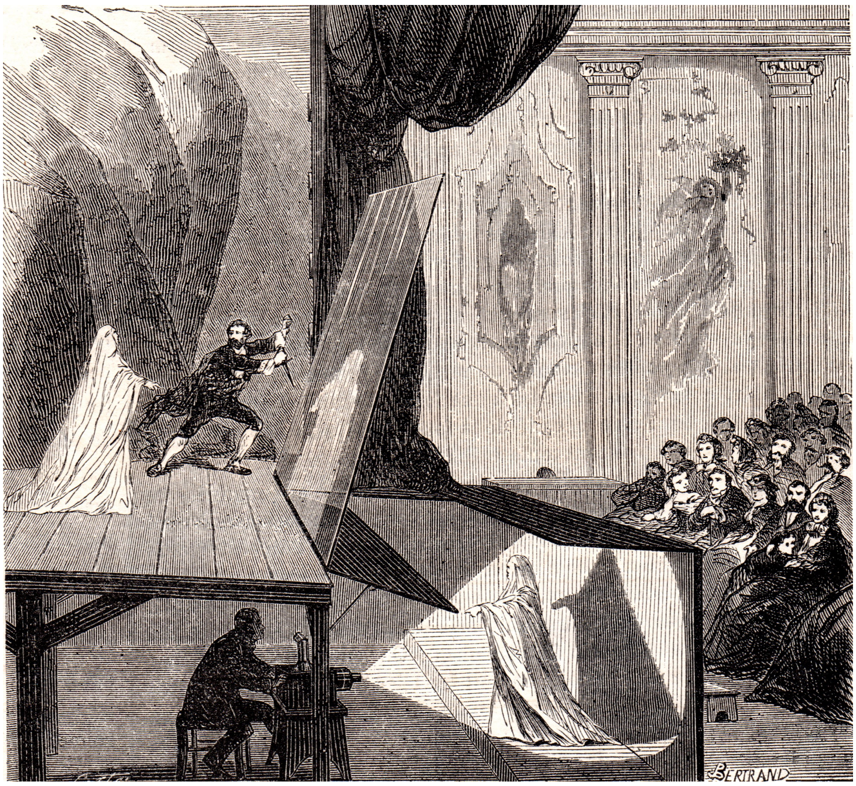


Figure 2.1: By Le Monde Illustré, Public Domain

One example is the Pepper's Ghost technique, named after the English scientist John Henry

Pepper (1821–1900), depicted in Figure 2.1. This method uses a light display to project an image onto a transparent surface, creating the illusion of a hologram.

An advanced version of this system employs a four-sided glass or acrylic pyramid. The display is positioned above this pyramid, allowing the holographic image to be visible from each side.

Voxel technology offers a different approach, where each voxel represents a volumetric pixel in three-dimensional space, effectively creating floating pixels. Displays utilizing this technology are known as volumetric displays and represent true holographic experiences [49].

Fan-based holography is also gaining interest. This technology projects light onto a spinning fan. At a distance, the fan's motion is imperceptible, but the light projection creates the appearance of floating objects [31].

2.2.2 Gesture Recognition

As an integral part of human-computer interaction, gesture recognition technologies have evolved to allow more natural and intuitive user interfaces. Gesture recognition involves the identification of significant motion expressions made by a human, including movements of the hands, arms, face, head, or body [26]. In the realm of gaming, this technology has evolved from a novel feature to a central gameplay element, enhancing interactive experiences and immersion. The concept of gesture-based gaming interfaces utilizes advanced computer vision techniques to interpret player movements and translate them into game actions.

Recent advancements have introduced more sophisticated algorithms and technologies, such as the Haar Cascade classifier, to improve object detection and gesture recognition within virtual environments [44]. These systems enable more intuitive interactions, allowing players to control game elements through natural body movements. For example, in the game "Flying Wing Mario," players can control the game entirely through hand gestures, demonstrating a significant shift from traditional control schemes to more engaging and physically active gaming experiences [44].

The integration of gesture recognition in gaming simplifies user interaction and introduces a new layer of realism and engagement. By employing technologies such as real-time hand tracking and object recognition, games can offer a seamless blend of the virtual and real worlds, where player actions directly influence the virtual environment [32].

Moreover, the application of these technologies is not limited to traditional gaming platforms. Mobile devices, equipped with sophisticated sensors and cameras, now support gesture-based games, providing a versatile and accessible platform for such innovative gameplay. The PicoLife game, for example, uses a computer vision-based gesture recognition system on Android devices to control 3D characters, highlighting the potential of mobile platforms to support complex gesture-based interactions [25].

The continuous development of gesture recognition technology promises to further revolutionize the gaming industry, making games more immersive, accessible, and enjoyable. Future research could explore more efficient algorithms that minimize latency, improve gesture detection accuracy, and reduce the computational load on devices, thus broadening the potential applications of gesture recognition in gaming and other interactive systems.

2.2.3 Cognitive Agents

Cognitive agents are a form of artificial intelligence that leverage a variety of AI methodologies, including but not limited to machine learning (ML), natural language processing (NLP), BDI models, knowledge bases (KBs), system automation, speech to text (STT), and text to speech (TTS) to facilitate interaction and learning from humans [21]. These agents have found applications across multiple fields, from industrial settings employing Internet of things (IoT) and fog computing [12] to mental health therapy [46], and notably in interactive educational systems [4].

In gaming, the implementation of cognitive agents has evolved to enhance player experience by adding depth to gameplay through dynamic interactions and intelligent non-player characters (NPCs). For instance, the Crystal Space 3D engine has been used to simulate an embodied cognitive agent, enhancing the realism of player interactions within 3D gaming environments [33]. This agent, designed on a three-layer architecture, demonstrates how cognitive agents can perform complex tasks like navigation and interaction within a simulated world, fundamentally increasing the immersion and interactivity of games.

Furthermore, a recent study discussed the integration of AI chatbots into video games, highlighting their potential to revolutionize player engagement through more personalized and non-linear storylines [19]. These chatbots, driven by advancements in NLP and machine learning, contribute to creating dynamic narrative elements that respond to player choices, thereby enhancing the gaming experience.

However, implementing cognitive agents for direct interaction with users presents challenges, particularly in maintaining seamless, realistic interactions without compromising the responsiveness or behavior of the game environment. Addressing these challenges involves not only refining the agents' ability to process and react to user inputs in real-time but also ensuring they can operate within the complex constraints of interactive gaming platforms without detracting from the user experience.

2.3 System Architecture and Implementation

We have utilized a design science methodology to develop a holographic gaming console that integrates gesture recognition and cognitive agents. Design science, a prevalent research method within the realms of information systems and technology, focuses on creating innovative artifacts, including models, methods, and systems, aimed at aiding individuals in the creation, utilization, and maintenance of IT solutions [18]. This methodology is applicable across various domains beyond merely information systems and technology.

The process of design science (refer to figure 2.2) typically encompasses several key phases. These phases are not necessarily linear and can be revisited repeatedly throughout a design science research project. Below, we detail these phases and describe their implementation in the creation of our gaming console:

1. **Problem Identification and Motivation:** This phase focuses on recognizing and establishing the significance of a specific problem that requires resolution. It is essential to fully comprehend the problem and clearly express the rationale for addressing it.

   In our context, the challenge identified aligns with the goals outlined in the O_HAI 4 Games project, to which our gaming console development contributes significantly. The core concept was to create a publicly accessible gaming console using Ho/Vo technology that operates without traditional controllers, enabling interaction through mobile devices or by gestures and voice commands. Moreover, it is designed to serve as an interactive display element in various exhibitions to demonstrate the potential of these cutting-edge technologies.
2. **Define the Objectives for a Solution:** Once the problem has been identified, it is essential to specify the objectives that the solution aims to achieve. These objectives must be realistic and directly tackle the identified issue.
   The main goals for developing our gaming console are set as follows:
   (a) *Support for Retrofuturistic Gaming:* The console should accommodate a variety of classic games reimagined for a holographic display, offering an experience that merges traditional gaming with modern nostalgia.
   (b) *User Accessibility:* The console will be designed to connect to users through a WiFi access point, enabling interaction via their mobile devices which can also act as con-

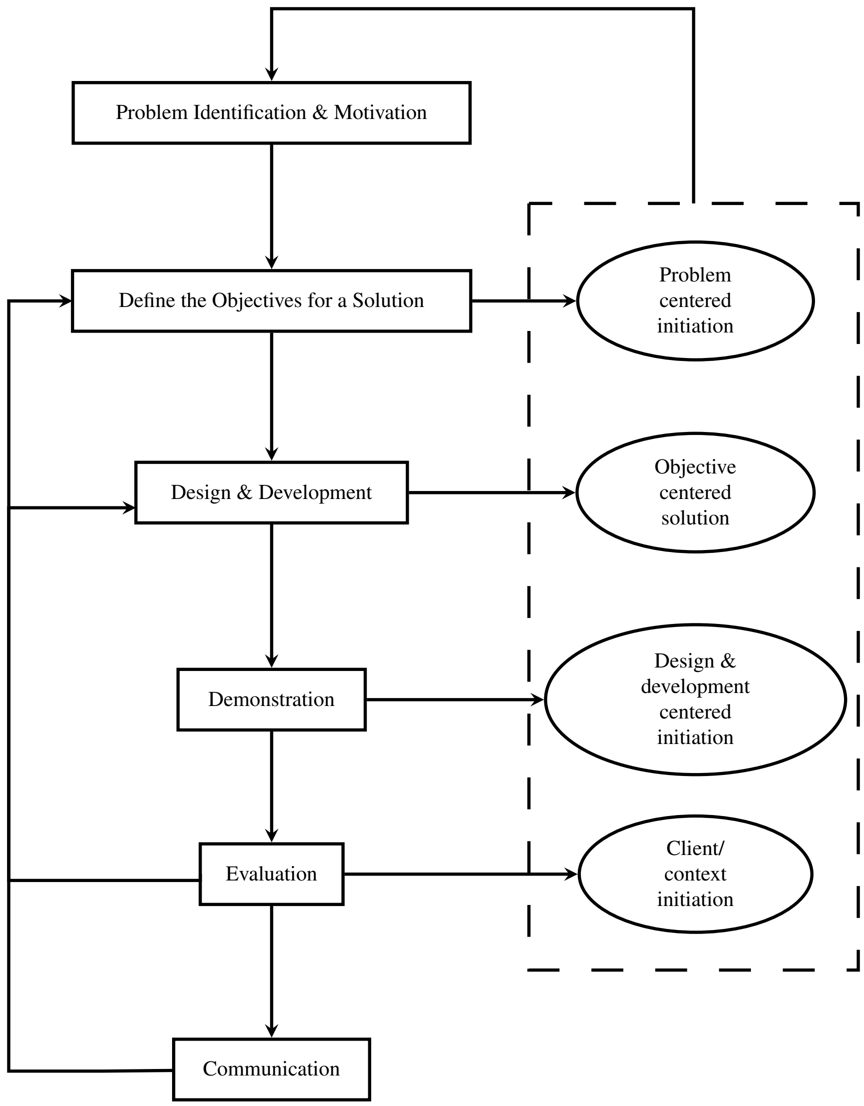


Figure 2.2: Adaptation of the design science methodology [2], originally based on [29]

trollers.

(c) *Gesture-Based Interaction:* Implement machine vision technology for gesture recognition to allow players to engage with games through intuitive physical movements.

(d) *Modularity and Contextual Adaptability:* The console will feature a modular design, allowing it to switch roles between a gaming platform and a host for the cognitive agent B.A.R.I.C.A., facilitating diverse user interactions in various settings.

(e) *Exhibition and Display Capability:* The console is to be crafted as a versatile display item for various exhibitions, showcasing its innovative features and the seamless integration of advanced technologies.

3. **Design and Development:** This phase involves the creation of an artifact, such as a model, a method, or an instantiation, that meets the previously defined objectives. The development of our gaming console, referred to as HoloGameV, included several distinct stages:

(a) *Preparation Phase:* Careful planning and detailed modeling of the HoloGameV system were undertaken to ensure a solid foundation for further development.

(b) *Casing Design:* A modular casing was developed to facilitate the installation of the fan display and interchangeable front design plates. The casing is comprised of a cabinet-like structure that consists of an upper and a lower part which can be disconnected for transport. Additionally, the upper part can be used as a standalone casing if it is positioned of a desk for presentation purposes. In the inside of the casing is a metalic rod in the middle on which the Ho/Vo display is mounted. Bellow are two shelves on which the other hardware is to be mounted. The two front plates are made of plexiglass and can be removed and changed when necessary. Two distinctive designs for the front plates were created (shown on figure 2.3:

- *RetroFuturistic Console:* Inspired by pixel-art games, utilizing the color scheme of our faculty.
- *Cognitive Agent:* Themed as an elegant ball gown for the cognitive agent B.A.R.I.C.A., depicted as a sophisticated lady.

Designs were crafted using GIMP with free and available design assets.

Figure 2.3: Designs of fron plates: left - Cognitive agent design, right - Retrofuturistic front casing

(c) *Hardware Design:* An overview of the hardware architecture is shown on figure 2.4.

The architecture revolves around a microcontroller which server both as the gaming hardware and connectivity server for WiFi, gesture and voice input. Components were strategically arranged within the casing as follows:

- *Ho/Vo Display:* Positioned at the upper front behind the transparent part of the design plates to enhance image integration.
- *Microcontroller:* A Raspberry Pi 4 microcontroller installed with RaspbianOS, concealed within the casing.
- *Camera and Motion Sensor:* Installed at the front to facilitate image capture and motion detection for gesture recognition.
- *Speakers:* Positioned on both sides of the casing to deliver audio output.
- *Microphone and Auxiliary Input:* Located at the front for audio input, with auxiliary inputs concealed within the casing for maintenance purposes.

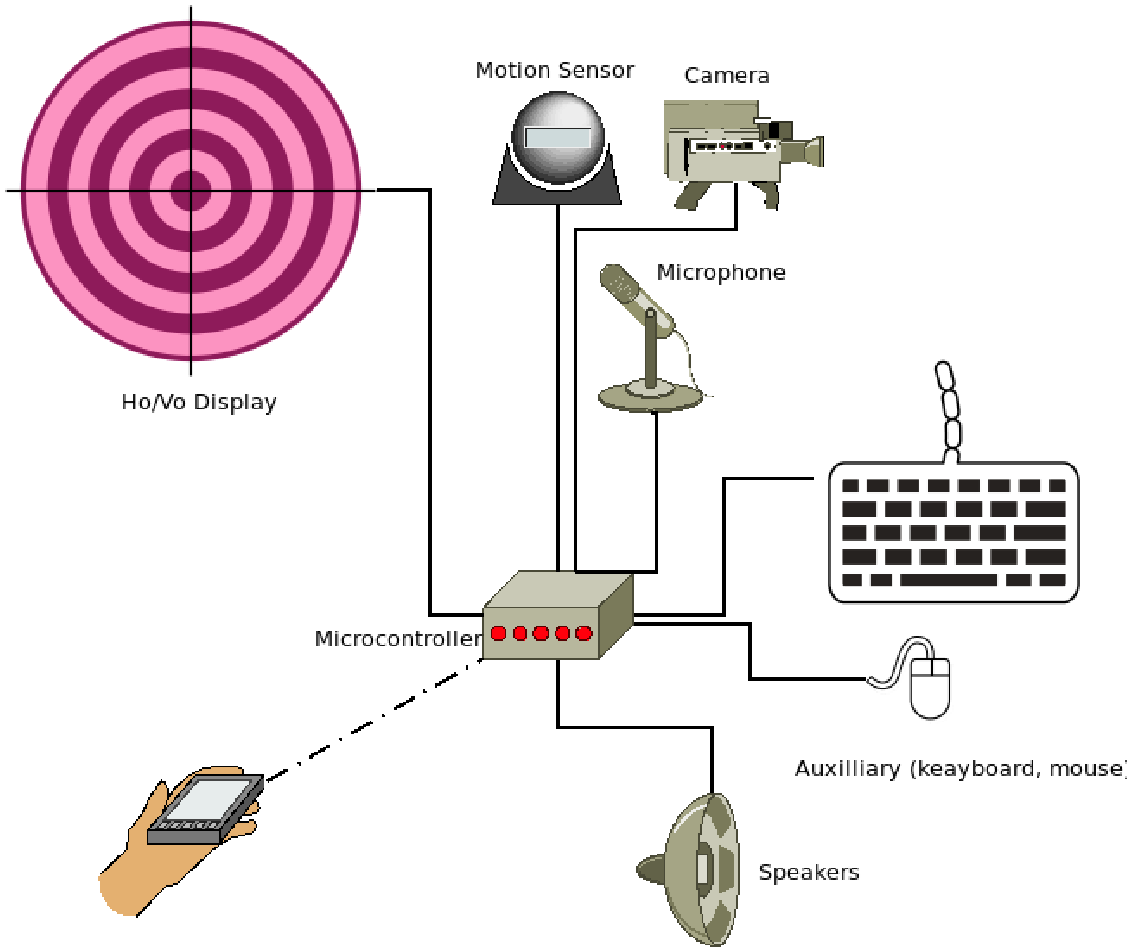


Figure 2.4: Hardware architecture overview

(d) *Software Implementation:* The software architecture of the console is comprised of several components.

- *WiFi AP:* In order to allow users to connect to the console using their mobile devices a WiFi access point has been installed. For this purpose we have utilized an open-source solution (WiHotspot). It acts similar to a captive portal which forwards user's browser requests to a specific web application that shows a mobile controller (shown on figure 2.6).
- *Mobile Controller:* The mobile controller is a web application developed using

Figure 2.5: Console with the retrofuturistic front casing

Python Flask and PyAutoGUI for the backend as well as JavaScript for the frontend.[1] The Flask server broadcasts a controller imagemap to mobile devices, which interact via websockets. On the backend side the incoming websocket requests are mapped to keyboard presses which are simulated using PyAutoGUI. In this way, due to low latency of the WiFi network, the user can directly interact with the game on-screen as if they use a connected game controller.

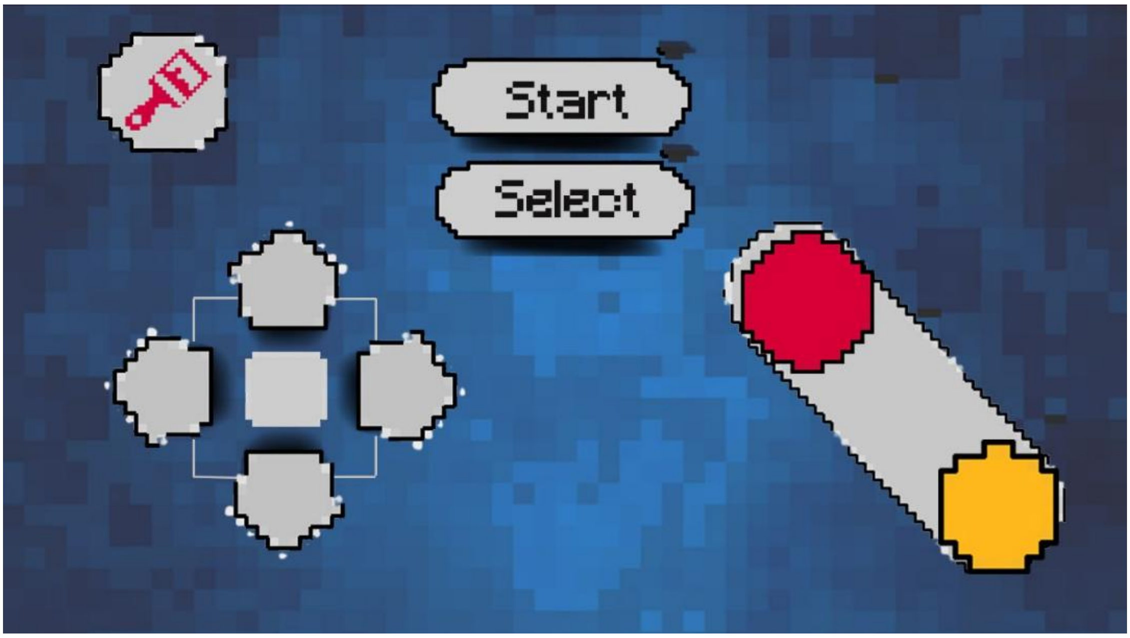


Figure 2.6: Controller interface (design made by students of the Faculty of Organization and Informatics during a Game Platform Development course)

- *Gesture Recognition:* A gesture recognition system has been implemented using Google Teachable Machines,[2]. The trained model has been exported and integrated using the previously mentioned Python server to simulate keyboard commands upon gesture detection.
- *Game Selection System:* The game selection system was implemented as a web application as shown on figure 2.7. The design is intentionally round due to the specific round display. The game selection system includes a number of free and open source retro games that have been carefully selected to be playable given the designed interface and the available controllers.
- *Cognitive Agent* The cognitive agent system, shown on figure 2.8, is based on the B.A.R.I.C.A. architecture [43, 45]. It features (1) a Python backend based on the chatterbot module that allows us to train custom regressive models for text recognition, (2) a JavaScript frontend that makes use of the SpeechRecognition API that allows users to converse with the agent in natural language and (3) an animated graphical user interface (GUI) that consists of a video sequence of prerendered

[1] A concurrent implementation of the server part in NodeJS has been implemented by students of the Faculty of Organization and Informatics which is available here `https://github.com/AILab-FOI/PRRI-HoloGameV2023/tree/main/GamePad`. This implementation has not been used for the final console due to a number of shortcomings, namely the inability to integrate the server with other components including the gesture recognition system.

[2] Available at `https://teachablemachine.withgoogle.com/`; The initial gesture recognition system has been implemented by student Noa Midžić and is available here `https://github.com/AILab-FOI/PRRI-HoloGameV2023/tree/gestures-sensor/GamePad/js`.

Figure 2.7: Game Selection Interface (design made by students of the Faculty of Organization and Informatics during a Game Platform Development course)

speech animations that are controlled via JavaScript based on the selected answer from the regression model.

- *Integration:* In order to allow for smooth operation a number of tasks have been automated through a series of bash scripts, which allow for managing system startup and server operations. Depending on the context, the system can be adjusted to either load the gaming system, with or without gesture recognition, or the cognitive agent interface.

4. **Demonstration:** The artifact is demonstrated to address one or more specific instances of the identified problem through practical applications such as experiments, simulations, case studies, or other relevant activities.
The console was prominently featured at the Faculty's day and the University's fair over two consecutive years, where it was extensively used by various attendees. This provided significant real-world exposure and valuable feedback. During the Faculty's day, the cognitive agent system was highlighted, with the agent B.A.R.I.C.A. engaging guests as they arrived. At the University's fair, the focus was on demonstrating the gaming system capabilities, as illustrated in figure 2.9.
5. **Evaluation:** The artifact undergoes evaluation to ascertain its effectiveness in solving the identified problem. This can include user observations, experimental testing, or analytical assessments.
Beyond the previously mentioned showcases, the console underwent extensive testing over two consecutive semesters by students enrolled in the Game Development Platforms course at the Faculty of Organization and Informatics. The feedback gathered was analyzed, leading to several enhancements in the implementation. Notable suggestions from students included redesigning the controller interface, which was subsequently realized by the students themselves, rectifying bugs in controller key press and release events, developing additional games suited for the gesture recognition system, and enhancing the integration of the cognitive agent within the console.
Looking forward, we see several potential areas for further improvement. For instance, enhancing the noise cancellation capabilities would significantly improve interactions with the cognitive agent, particularly since the console operates in noisy public spaces and the Ho/Vo

Figure 2.8: Cognitive agent B.A.R.I.C.A. in the casing during Faculty's day

Figure 2.9: Gaming console at the University's fair

display generates additional noise. A promising approach could involve utilizing mobile devices to interact with the agent, leveraging their in-place noise cancellation technologies.

6. **Communication:** The results of the project are disseminated to appropriate audiences through various means. This includes the publication of research papers, presentations at scientific and industry conferences, and the creation of detailed technical reports.
   In addition to this chapter, the console has been demonstrated at various public events as previously described. Furthermore, to foster community engagement and collaboration, all software developed for the console has been made open-source. This software is readily accessible on the Github repositories of the Artificial Intelligence Laboratory at the Faculty of Organization and Informatics.[3] We hope that this approach will promote transparency and encourage contributions from other developers and researchers, enhancing the ongoing development and refinement of the console. Additionally, the project and its outcomes have been featured in several academic and industry newsletters, further broadening the reach and impact of our research.

2.4 Discussion

The development of the holographic gaming console presented in this chapter represents a significant advancement in the integration of holographic technology and interactive gameplay with cognitive agents. This section discusses the key contributions of this development, its limitations, and implications for future research and application.

Our console, leveraging both holographic and gesture recognition technologies, provides an immersive user experience that is markedly distinct from traditional gaming systems. The ability to interact with games through natural gestures and without physical controllers is a noteworthy progression towards more intuitive human-computer interaction. This system enhances user

[3]For more information, visit: `https://github.com/AILab-FOI`

engagement ant tries to expand accessibility, making gaming more inclusive for users who may have limitations with traditional gaming interfaces.

The integration of the cognitive agent B.A.R.I.C.A. further augments this experience by adding a layer of interactive intelligence that can respond to and learn from user interactions. This integration is not just a technical enhancement but also adds a unique character to the gaming console, potentially increasing user attachment and engagement through personalized interactions.

Despite the successful implementation of the console, several challenges were encountered. Firstly, the noise interference from the holographic display poses a significant challenge in terms of ensuring clarity in voice recognition, crucial for interacting with the cognitive agent. This issue underscores the need for advanced noise-cancellation technologies or alternative design solutions to minimize background noise.

Secondly, the gesture recognition system, while effective, still requires fine-tuning to handle a broader range of natural user gestures. The current system's reliance on predefined gestures can sometimes limit the spontaneity and fluidity of interaction, which are critical in gaming.

The research presents numerous opportunities for future work, particularly in improving the robustness and versatility of the interaction modalities. For instance, enhancing the cognitive agent's capabilities to handle more complex interactions and to learn from user behavior over time could significantly personalize the gaming experience.

Additionally, the modular nature of the console's design suggests potential beyond gaming; it could be adapted for educational purposes or specialized training simulations, where holographic interactions could provide substantial benefits.

The development of this holographic gaming console contributes to the field of interactive gaming by demonstrating the practical application of holographic displays in a consumer product. Furthermore, it adds to the academic discourse on the potential of combining cognitive computing with real-time user interaction technologies. The feedback from showcasing this console at public events and the incorporation of user feedback into iterative design improvements underscore the practical relevance and adaptability of this research.

2.5 Conclusions & Future Research

This chapter has detailed the development of a holographic gaming console integrating gesture recognition and cognitive agents, highlighting its innovative design and potential impact on interactive gaming. The holographic gaming console developed through this research exemplifies a successful application of holographic display and gesture recognition technologies combined with cognitive interaction. The console offers a unique gaming experience by enabling intuitive user interactions through gestures and engaging players with a responsive cognitive agent. The project's adherence to design science methodology ensured a systematic approach to development, which can serve as a model for future technology design projects.

Moreover, the practical demonstrations and user testing provided valuable insights into user engagement and system performance, affirming the console's appeal and functionality. The integration of advanced technologies in a user-friendly format presents a significant step forward in making sophisticated interactive technologies more accessible and enjoyable for the general public.

Building on the foundations laid by this project, future research can explore several promising avenues:

- **Enhanced Interaction Capabilities:** Future iterations of the console could incorporate more advanced AI capabilities to enhance the cognitive agent's responsiveness and adaptability to individual users. This could involve more sophisticated machine learning models that allow the agent to learn from each interaction and tailor responses more effectively.
- **Expanded Gesture Recognition:** Expanding the gesture recognition system to include a wider array of gestures and possibly combining this capability with other forms of user input

could enhance the interactivity and accessibility of the gaming console.

- **Noise Reduction Solutions:** Research into more effective noise-cancellation technologies or design modifications to reduce interference from the holographic display would improve voice interaction quality, crucial for interactions with the cognitive agent.
- **Scalability and Commercialization:** Further research could also focus on the scalability of the console's design for mass production and commercialization, including considerations of cost reduction and manufacturing feasibility.

The development of this holographic gaming console represents a notable advancement in interactive technology, merging aesthetics with functionality and pushing the boundaries of user interaction in gaming. The continued exploration of these technologies holds the promise not only for enhancing entertainment but also for applications in education, training, and beyond, signaling a bright future for holographic and cognitive interaction technologies.

3. HoloGameP

In this chapter we will focus on the development of another quasi-holographic console called HoloGameP which is based on a Pepper's Ghost pyramidal display.

3.1 Introduction

A game engine is a computer program that enables the development and management of all aspects and computing resources required to run a game (e.g., graphics, sound, user interface, scripts, events, data storage, networking, artificial intelligence, physics, etc.). The user of a computer game engine is a computer game programmer. In this chapter we are on our journey to develop a game engine that would allow games to be (1) streamed over a network and (2) displayed on a holographic display device.

Game engines are complex systems consisting of numerous subsystems (see figure 3.1). Our objective herein is not to *reinvent the wheel* and reimplement all parts of a game engine from scratch, but to focus on two particular aspects of game engines which will allow us to achieve the above stated objectives: game streaming and holographic display of games.

Thus, we have chosen to use an existing game engine (namely UPBGE - a community based continuation of the Blender Game Engine[1] and extend it with: (1) a game streaming architecture, (2) a display transformation to allow display on a holographic display.

The rest of this chapter is organized as follows: firstly in section 3.2 we provide an overview of game streaming technology. In section 3.4.1 we describe the prototype system that we have developed. In the end in section 3.7 we draw our conclusions and provide guidelines for future research.

3.2 Game Streaming

The advancement of technology allows most households to own at least one personal computer (PC). Such PCs can usually suffice when playing video games is considered, most modern games included.

[1] For further information, see `https://upbge.org/`

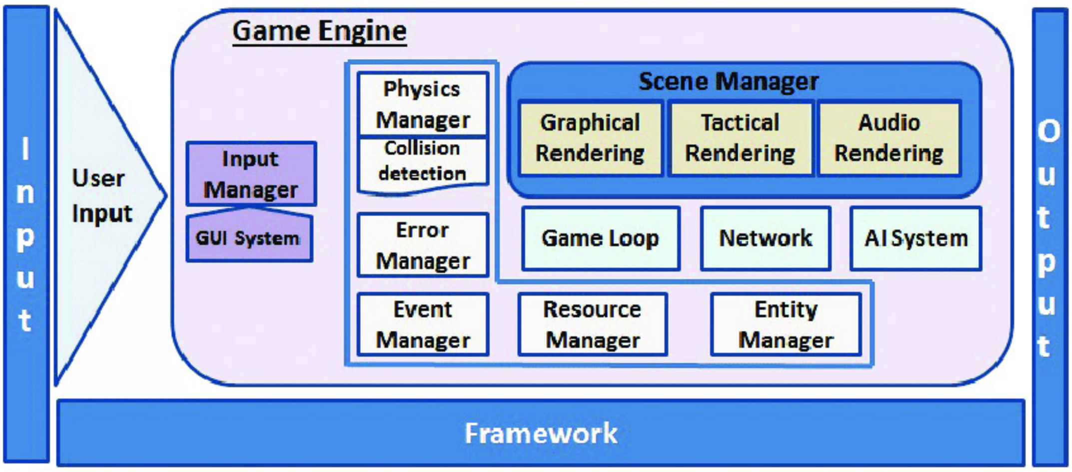


Figure 3.1: A game engine's architecture [51]

The key and the most observable element of enjoying a game is observing rich and smooth graphics provided by the game designers, especially when combined with smooth and fluid storytelling and game mechanics, provided by the games' developers and other artists. The most important building block of a PC, in the context of providing enjoyable graphics, is graphics hardware, i.e. external graphics cards (as opposed to on-chip integrated graphics hardware). The main problem in the domain of graphics cards is their price, which soared during the last decade, and made high-tier PCs almost unavailable to average targeted consumers in e.g. the EU, North America, or Far East. This obstacle, in turn, makes it harder for gamers, i.e. variously skilled players of games, to enjoy the full potential of games and their developed graphics.

With the advent of access to the Internet, and the increasing speeds available in increasingly large areas and portions of the world, e.g. access to high-speed optical fibre across the EU, an average user can rely more on the constant availability of information, and the speed of recovering it whenever needed. Constant availability of the Internet rendered music and video storage, such as a CD-ROM, a DVD-ROM, a BluRay disks, or a USB memory drive, or even a portion of a disk with stored music files, almost obsolete. Reliable access to the sources that provide multimedia files as streams of data shifted the focus of an average user, shifted the behaviour towards legal use of various types of multimedia, and created new habits when multimedia consumption is considered. Even though they are dispersed amongst many streaming services, a rich collection of music, movies, TV shows, cartoon series, etc. is available in a matter of seconds to a great number of users throughout the world.

The trouble with streaming video games is that video games are an interactive type of multimedia entertainment. Videos and music can be streamed safely, but they are being streamed one way only: from the service provider to the user, and not much interaction is expected except for maybe starting a stream, pausing, moving the time line and other simple instructions. In streaming video games, the game is expected to behave according to the input of its player in real time. For example, if the player moves by pressing a key on the keyboard, the in-game character is expected to move almost instantly; if the player clicks on a menu item, the interface is expected to change accordingly; if the player's avatar is being attacked, or is found in a situation where speed is of the essence, the game is expected to react to the player's input almost instantly, etc. The described scenarios are only a few, a needle in a haystack of playing video games, that require almost instant reaction of the game – moreover, they require that the player receives an almost instant feedback that the game instantly reacted to the received player's input. Usually, establishing such a quick feedback loop is not a problem in PCs, but it does present a challenge in video game streaming services.

There is a caveat that should be addressed though. Not all games require feedback loops that are (as much as possible) instantaneous. Some games (usually single-player games) can be played with an occasional slight delay, because such a game, for example, does not rely on the speed of its player's reaction, or that particular feature is of low significance.

The benefit of combining the ideas of video games and multimedia streaming services is manifested in the opportunity to play video games without having necessarily to own the hardware that can run those video games. Usually, the time spent on gaming, i.e. playing video games, is only a fraction of the time spent on using the PC for other purposes. Therefore, owning hardware that will provide the resources for playing a video game can be expensive, especially when time, utility, and use are considered. On the other hand, a video game streaming service can be used whenever needed, and (in some cases) on virtually any device available, if the appropriate Internet speed and game-controlling hardware (e.g. a game controller, a keyboard, a mouse, a wheel, etc.) are at the player's disposal.

There are technologies and approaches that are used for further reducing the expected feedback loop duration, and providing smoother playing experience, such as predicting a player's move and rendering possible actions before they actually happen, or rendering only specific sets of pixels that change in consequent frames, etc. [39]

Game streaming systems are usually complex and consist on numerous components that deal with various aspects of networking, game interaction as well as graphics rendering (see figure 3.2 for an example architecture based on virtual machines proposed by [16]).

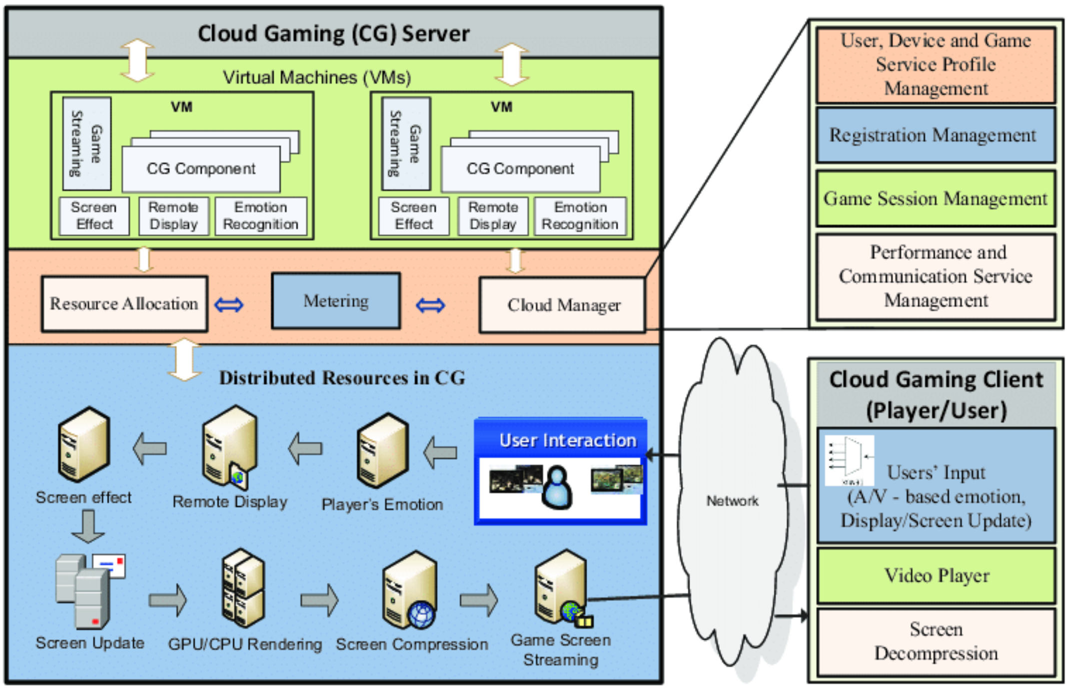


Figure 3.2: A game streaming architecture [16]

Many video game streaming platforms exist today [14], each of which uses a different method of providing games to be streamed, some of which are: NVIDIA GeForce NOW[2], Amazon Luna[3], Sony PlayStation Now[4].

[2]For further information, see `https://play.geforcenow.com`
[3]For further information, see `https://www.amazon.com/Luna/`
[4]For further intofmation, see `https://www.playstation.com/en-us/ps-now/`

The initial test case scenario for the holographic platform described in this chapter will consist of a game that does not rely on the speed of its player, rather it will be used to showcase the possibilities of playing a game developed for, and played on, such a platform.

3.3 Hardware Architecture

For the construction of the physical hardware we have conducted initial calculations for the size of the plexiglass pyramid and then modeled the rest of the cabinet according to these calculations, available display sizes and the necessity that the console allows for human eye line playing. We have constructed a detailed Computer Aided Design (CAD) model using FreeCAD[5] which is included in the appendix of this document.

Figure 3.3: HoloGameP Cabinet

According to this model, a wooden and metallic cabinet has been constructed as shown on figure 3.3. In the upper metallic compartment a display is mounted beneath which we positioned the plexiglass pyramid. In the lower wooden compartment the hardware including a gaming PC, a router and auxiliary mouse and keyboard are stored.

[5]Available at `https://www.freecad.org/`

3.4 Software Architecture

The software needed to run our console HoloGameP consists of a (1) streaming system, (2) game engine with camera transformation and (3) controller application programming interface (API) implementation. The system can run either in streamed mode or with a locally installed game. Tests have shown that the locally installed version provides much better performance due to network latency issues, which is why in the end we have focused on implementing proof-of-concept implementations for the locally installed version.

3.4.1 Game Streaming System

In order to provide a working proof-of-concept holographic game engine we have designed and implemented a game engine prototype for streaming holographic content. During design and implementation we have used multiagent systems (MASs) as a natural way of developing intelligent distributed systems. The architecture of the system is shown on figure 3.4. Additionally, we have used a gaming console as a metaphor for the implementation. The system consists of a pool of existing game agents which comprise containers called cartridges that represent the games that are supported by the engine (implemented by actual game developers). Each cartridge is a microservice container (in our case Docker[6] container) which consists of a minimalized Linux operating system that includes a graphical user interface and a game engine (in our case UPBGE).

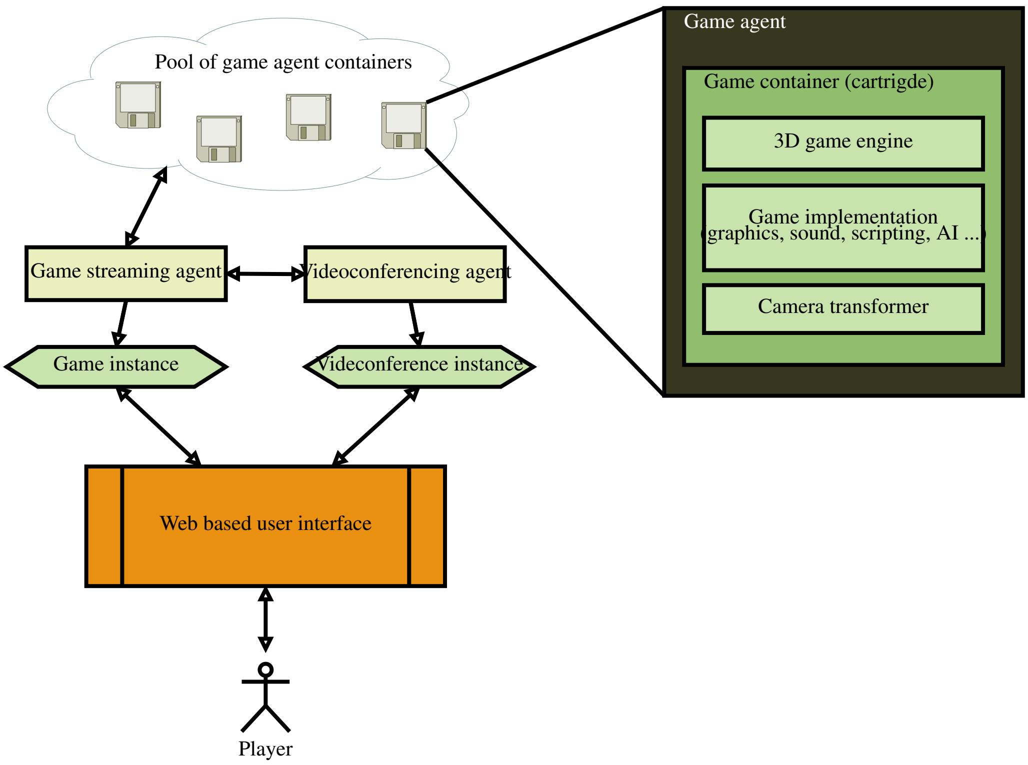


Figure 3.4: Prototype Architecture

3.4.2 Game Engine with Camera Transformation

In order to display a 3D game on a holographic pyramidal display, the actual game implementation had to include a camera transformation. We have implemented this transformation in UPBGE by

[6]For further information, see `https://www.docker.com/`

using a simple technique. In order to get the state of the 3D world for each of the four sides of the pyramid we have used four cameras which display the world in real time from four perspectives. Additionally, we have created a custom viewport that shows all four camera outputs on the same display at once, which allows us to use one display above the pyramid to show all four perspectives. The implementation of this script is shown in the following listing:

```
from bge import logic, render
camList = logic.getCurrentScene().cameras

cont = logic.getCurrentController()
own = cont.owner
camn = camList[ own[ 'camn' ] ]
cams = camList[ own[ 'cams' ] ]
camw = camList[ own[ 'camw' ] ]
came = camList[ own[ 'came' ] ]

a = render.getWindowWidth()
b = render.getWindowHeight()

x = int( ( a - b ) / 2 )
y = int( b / 4 )
z = int( b / 2 )

# setViewport( left, bottom, right, top )
camn.setViewport( x+y, y+z, x+y+z, b )
cams.setViewport( x+y, 0, x+y+z, y )
camw.setViewport( x+y+z, y, x+2*y+z, y+z)
came.setViewport( x, y, x+y, y+z )

camn.useViewport = True
cams.useViewport = True
camw.useViewport = True
came.useViewport = True
```

For example figure 3.5 shows the UPBGE interface for a simple world in which we have put an interactive 3D model of a dragon[7] on a simple plane. The dragon can be manipulated using the usual keyboard and mouse inputs in real time.

By using the script shown above, the world is rendered as shown on figure 3.6. As one can see, four cameras show the dragon from four sides (north, east, south, west) at the same time.

By using a transparent pyramid and placing the center of it in the center of the display by aligning the sides with the sides of the display, one can get a hologram-like appearance as shown on figure 3.7. In this way, any game implemented in UPBGE with the given transformation can be transformed in a holographic game. Of course, this transformation is not suitable for all existing games, but only for those for which a holographic display can or should be used. We envision that various new types of games, which make use of the holographic nature of the display in their core game mechanics can be implemented that go beyond games based on traditional non-holographic displays.

In addition to game agents, there are two other important types of agents: (1) game streaming agents as well as (2) videoconferencing agents. The gane streaming agents are envisioned as

[7]Available at `https://3dmdb.com/en/3d-model/bge-dragon-20/1005315/?free=True&q=bge+dragon`

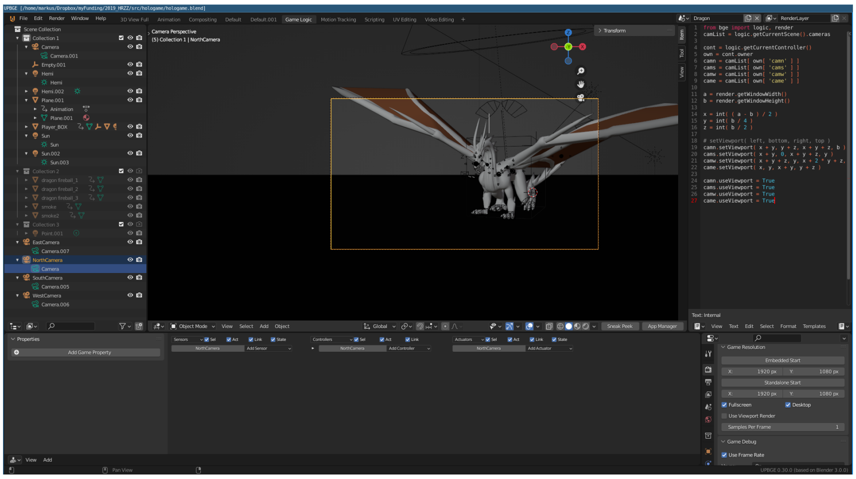


Figure 3.5: UPBGE interface

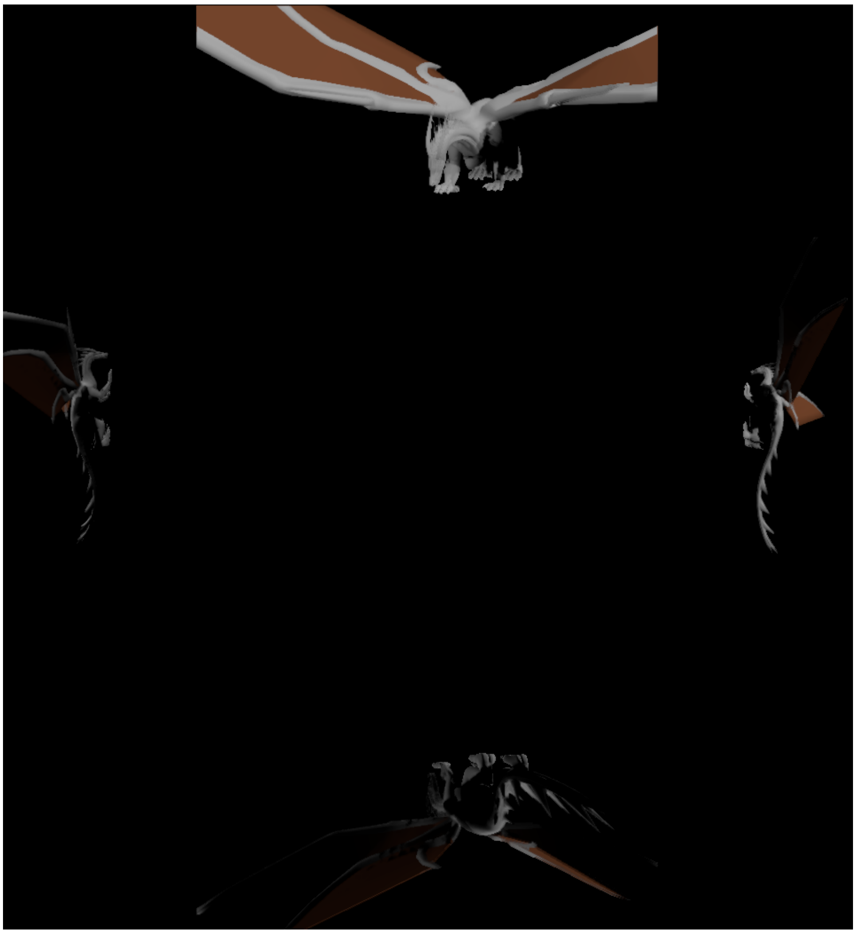


Figure 3.6: Display with four parallel cameras

orchestrators. They get requests from clients that would like to play a game from the pool of available games, allocate resources in the could (which could be any orchestration platform like Kubernetes of Docker Swarm) and instruct a game agent to start it's cartridge as a container instance on the allocated end system. This allocation is considered a game instance in which an instance of a game agent directly communicates with an end-user's web based interface to which a game

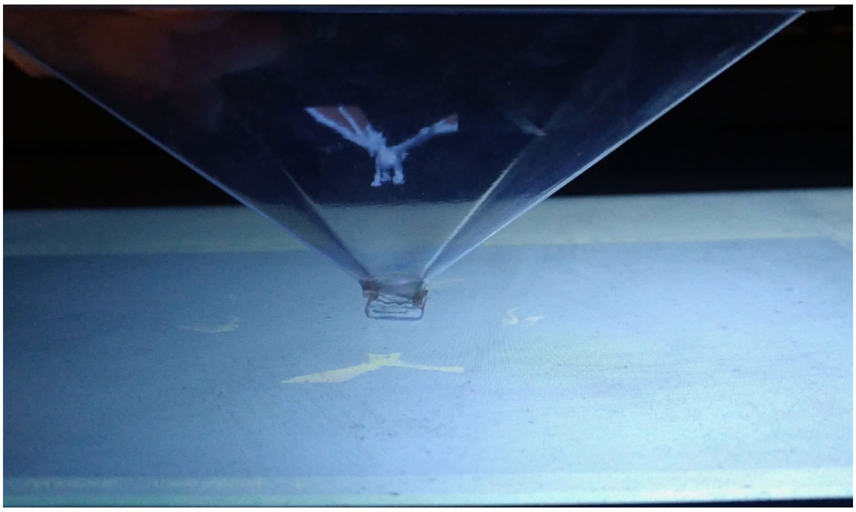


Figure 3.7: Hologram-like appearance

is streamed. The streaming is implemented by using XPRA[8] and a X11VNC[9] server which are contained in each game cartridge.

In addition to game streaming agents, videoconferencing agents are implemented as well that allow audiovisual and text communication between players (in multiplayer games for example) or live streaming of a players video and audio feeds (for live events for example). The game sgtraming agents instructs a videoconferencing agent to open an new room for each game that is being streamed and this new videoconference instance then directly communicates with the player's web based user interface. Videoconferencing is implemented using Janus WebRTC server[10].

In the end, the web based user interface features a VNC client that displays the streamed game and allows user interaction as well as a WebRTC client to allow for video and audio streaming.

All agents are implemented using Python, especially the SPADE platform [28]. The web based user interface is an adaptation of noVNC[11] with additional WebRTC client implementation in JavaScript.

3.5 Controller Implementation

In order to allow players to interact with the console, we have taken a similar approach as with the HoloGameV platform. A WiFi hotspot is set up (again using the open source solution WiHotspot) that allows players to connect their mobile devices to the console and use them as a gamepad. We have developed a WebSocket based interface that allows game developers to implement their own controller which can be interactive, meaning that not only it allows players to control the game by pressing various buttons, but also to communicate with the server and interactively show the game state (like for example acquired points, available hit points - HP, statistics or any game related data). In this way any game can have its own type of controller opening up new and unseen possibilities of game implementation like new types of game mechanics or player-game interactions.

[8] For further information, see `https://xpra.org/`

[9] For further information, see `https://launchpad.net/ubuntu/+source/x11vnc`

[10] For further information, see `https://janus.conf.meetecho.com/`

[11] For further information, see `https://novnc.com/`

3.6 Demonstration

In order to showcase some of the possibilities of the built console, we have implemented a draft prototype game for it. Additionally, students on the Game Platform Development Course have or are implementing additional games for the platform.[12]

3.6.1 Game Prototype - Chromospheres

Since the HoloGameP console offers exciting new possibilities of designing games some of which include the possibility to play from four different viewpoints at the same time as well as designing the interplay between the controller and the actual game on the pyramidal display, we have tried to come up with interesting game mechanics that would allow to show some of these capabilities. The first game prototype that we have partially implemented is called Chromospheres. It is a puzzle oriented 3D role-playing game (RPG) game in which up to four players can play in parallel. We have decided to use the following fictional story for the game:

> In the steampunk world of Cyndonia, a land of brass gears, roaring steam engines, and endless skies, an ancient mystery awaits discovery. Dr. Ezekiel Thorne, a reclusive scientist, has uncovered a forgotten relic—a chromosphere—that powers an ancient portal device, revealing a gateway to unknown realms.
>
> Summoning four brave adventurers, each with their unique skills, Dr. Thorne entrusts them with the mission of a lifetime. Captain Amadeus, a wise and seasoned airship captain, leads the team with his strategic mind and unwavering calm. Inventor Maribel, a young prodigy, brings her innovative gadgets and unquenchable curiosity to the expedition. Mechanic Dorian, a hands-on expert, stands ready to fix and tinker with any mechanical challenge that comes their way. And Pilot Brandy, a daring and fearless aviator, navigates the skies with unmatched skill and a thirst for adventure.
>
> Together, they must traverse twelve mysterious planets, solve intricate puzzles, and uncover the secrets of an ancient network of pathways. What lies beyond the portal is unknown, but legends speak of hidden treasures, mechanical marvels, and unimaginable dangers. Chromospheres - A Steampunk Adventure begin here, with a journey that will test their courage, ingenuity, and teamwork. Will they unlock the mysteries of the ancient mechanism and uncover the wonders of the once brotherly realm? The adventure awaits!

We have used a game engine combination to implement the prototype: (1) UPBGE - for the 3D world displayed on the pyramidal display (shown on figure 3.8) and (2) RPG Maker MV - for the controller displayed on players' mobile devices (shown on figure 3.9).

One game mechanic that would take advantage of the 3D display that we chose to implement was spherical gravity.[13] By taking inspiration of Antoine de Saint-Exupéry's novel "*The Little Prince*" we have chosen to put the players on small planets which rotate so that players have either to collaborate from different viewpoints or to change their position relative to the console to see various elements of the game world. The players have to solve puzzles on such planets to acquire so called chromospheres which open portals to other planets. An example planet is shown on figure 3.10.

On the other hand, the controller was implemented to allow for (1) login and registration, (2) character selection, (3) a lobby system for multiplayer as well as (4) a game controller as shown on figure 3.12.

[12] Available here `https://github.com/AILab-FOI/PRRI-HoloGameP2023/` and here `https://github.com/AILab-FOI/PRRI-HoloGameP2024/`

[13] The ingenious implementation of *wkk* was used `https://blenderartists.org/t/`

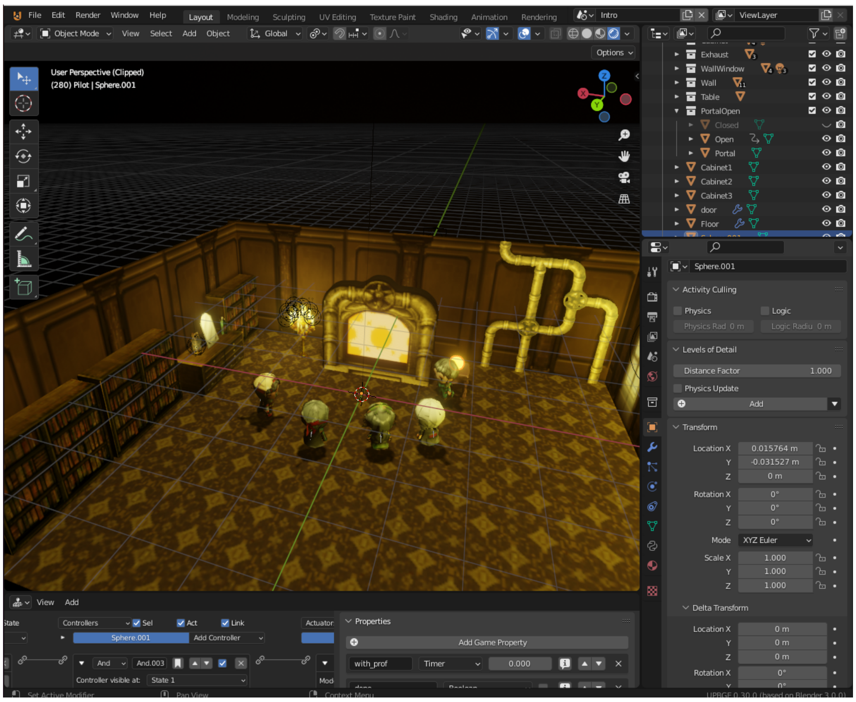


Figure 3.8: UPBGE development interface

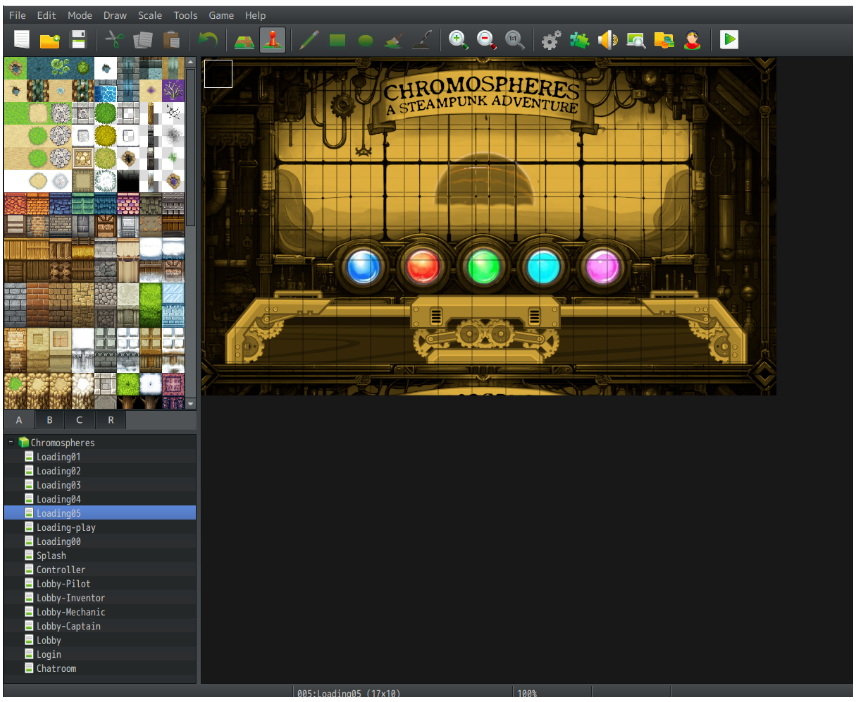


Figure 3.9: RPG Maker MV development interface

3.7 Conclusion & Future Reasearch

In this chapter we have showcased the implementation of a streamed 3D holographic game engine. We have developed a proof-of-concept game streaming platform that allows us to stream almost

`how-to-make-spherical-planetary-gravity-in-the-bge/675908/9`

Figure 3.10: Example of an in-game planet

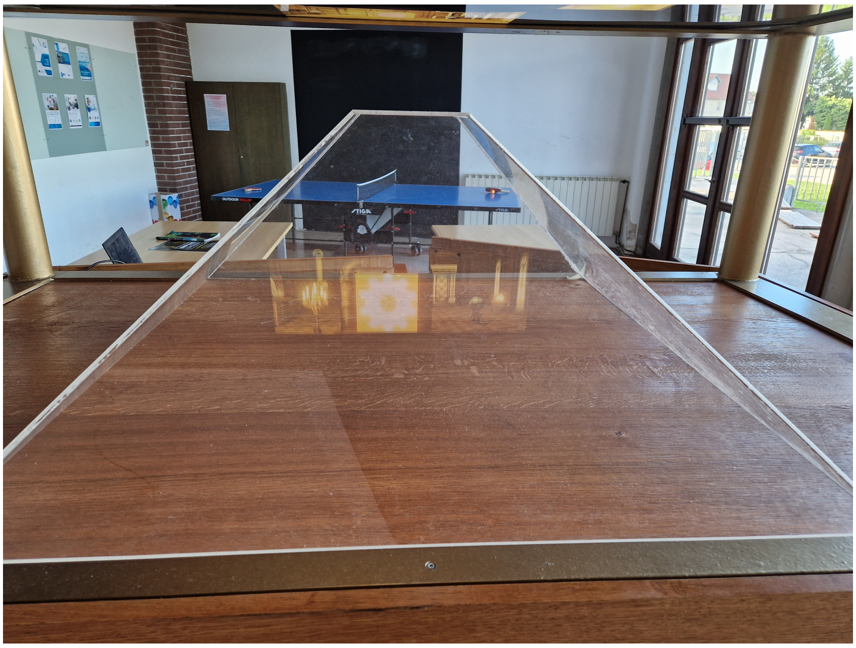


Figure 3.11: Example of an game intro scene

any game developed for the Linux X Window System to a web based VNC client. Additionally, we have extended the UPBGE game engine with a output transformation script that allows us to display games developed in the engine on a pyramidal holographic display device based on Pepper's Ghost technology. We have also developed a controller API that allows game developers to implement their own controllers to be used with mobile devices. In the end, we have developed a game prototype that showcases some of the possibilities of this console.

Our future research is aimed towards researching this technology further by implementing various games and consequently new game mechanics for this console as well as enhancing this

Figure 3.12: Game controller implemented in RPG Maker MV (login/registration, character selection, lobby system and game controller

technology with possibly other 3D game engines that allow for multiple camera views that could be embedded into game cartridges as described above.

Bibliography

[1] Danial Jahed Armaghani et al. “Airblast prediction through a hybrid genetic algorithm-ANN model”. In: *Neural Computing and Applications* 29.9 (2018), pages 619–629 (cited on page 2).

[2] Julius Azasoo and Kwame Boateng. “A Retrofit Design Science Methodology for Smart Metering Design in Developing Countries”. In: June 2015. DOI: 10.1109/ICCSA.2015.23 (cited on page 9).

[3] Aydin Azizi. “Hybrid artificial intelligence optimization technique”. In: *Applications of Artificial Intelligence Techniques in Industry 4.0*. Springer, 2019, pages 27–47 (cited on page 2).

[4] Amy Baylor. “Intelligent agents as cognitive tools for education”. In: *Educational technology* (1999), pages 36–40 (cited on page 7).

[5] Pierre-Alexandre Blanche. “Towards a Modular and Scalable Holographic Display”. In: *Light: Science & Applications* 11.1 (Dec. 2022), page 100. ISSN: 2047-7538. DOI: 10.1038/s41377-022-00786-9 (cited on page 6).

[6] Kien-Trinh Thi Bui et al. “A novel hybrid artificial intelligent approach based on neural fuzzy inference model and particle swarm optimization for horizontal displacement modeling of hydropower dam”. In: *Neural Computing and Applications* 29.12 (2018), pages 1495–1506 (cited on page 2).

[7] Eun-Young Chang et al. “360-Degree Color Hologram Generation for Real 3d Objects”. In: *Applied Optics* 57.1 (Jan. 1, 2018), A91. ISSN: 1559-128X, 2155-3165. DOI: 10.1364/AO.57.000A91 (cited on page 6).

[8] Wei Chen et al. “Novel hybrid artificial intelligence approach of bivariate statistical-methods-based kernel logistic regression classifier for landslide susceptibility modeling”. In: *Bulletin of Engineering Geology and the Environment* (2018), pages 1–23 (cited on page 2).

[9] Michael Cook, Simon Colton, and Azalea Raad. “Inferring Design Constraints From Game Ruleset Analysis”. In: *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE. 2018, pages 1–8 (cited on page 2).

[10] Dominik Dellermann et al. “Hybrid intelligence”. In: *Business & Information Systems Engineering* (2019), pages 1–7 (cited on page 1).

[11] Fazliaty Edora Fadzli et al. “Real-Time 3D Reconstruction Method for Holographic Telepresence”. In: *Applied Sciences* 12.8 (Apr. 15, 2022), page 4009. ISSN: 2076-3417. DOI: `10.3390/app12084009` (cited on page 6).

[12] Fotis Foukalas. “Cognitive IoT platform for fog computing industrial applications”. In: *Computers & Electrical Engineering* 87 (2020), page 106770 (cited on page 7).

[13] Tonglin Fu and Chen Wang. “A hybrid wind speed forecasting method and wind energy resource analysis based on a swarm intelligence optimization algorithm and an artificial intelligence model”. In: *Sustainability* 10.11 (2018), page 3913 (cited on page 2).

[14] Will Greenwald. *The Best Game Streaming Services for 2022*. PCMag. 2022. URL: `https://www.pcmag.com/picks/the-best-game-streaming-services` (visited on 06/18/2022) (cited on page 21).

[15] Abid Haleem et al. “Holography and Its Applications for Industry 4.0: An Overview”. In: *Internet of Things and Cyber-Physical Systems* 2 (2022), pages 42–48. ISSN: 26673452. DOI: `10.1016/j.iotcps.2022.05.004` (cited on page 6).

[16] M Shamim Hossain et al. “Audio–visual emotion-aware cloud gaming framework”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 25.12 (2015), pages 2105–2118 (cited on page 21).

[17] Abolfazl Jaafari et al. “Hybrid artificial intelligence models based on a neuro-fuzzy system and metaheuristic optimization algorithms for spatial prediction of wildfire probability”. In: *Agricultural and Forest Meteorology* 266 (2019), pages 198–207 (cited on page 2).

[18] Paul Johannesson and Erik Perjons. *An introduction to design science*. Volume 10. Springer, 2014 (cited on page 8).

[19] Natalia Khan and Nosheen Sabahat. “The Role of AI Chatbots in Revolutionizing Gaming Experiences-A Survey”. In: *VFAST Transactions on Software Engineering* 12.1 (2024), pages 93–104 (cited on page 8).

[20] Mohammadreza Koopialipoor et al. “Applying various hybrid intelligent systems to evaluate and predict slope stability under static and dynamic conditions”. In: *Soft Computing* (2018), pages 1–17 (cited on page 2).

[21] In Lee. *Encyclopedia of E-business Development and Management in the Global Economy*. IGI Global, 2010 (cited on page 7).

[22] Mengshan Li et al. “Prediction of pKa values for neutral and basic drugs based on hybrid artificial intelligence methods”. In: *Scientific reports* 8.1 (2018), page 3991 (cited on page 2).

[23] Ziyin Li et al. “Portable Autostereoscopic Display Based on Multi-Directional Backlight”. In: *Optics Express* 30.12 (June 6, 2022), page 21478. ISSN: 1094-4087. DOI: `10.1364/OE.460889` (cited on page 6).

[24] Gunasekaran Manogaran, R Varatharajan, and MK Priyan. “Hybrid recommendation system for heart disease diagnosis based on multiple kernel learning with adaptive neuro-fuzzy inference system”. In: *Multimedia tools and applications* 77.4 (2018), pages 4379–4399 (cited on page 2).

[25] Mahesh Babu Mariappan, Xiaohu Guo, and Balakrishnan Prabhakaran. "Picolife: A computer vision-based gesture recognition and 3D gaming system for android mobile devices". In: *2011 IEEE International Symposium on Multimedia*. IEEE. 2011, pages 19–26 (cited on page 7).

[26] Sushmita Mitra and Tinku Acharya. "Gesture recognition: A survey". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 37.3 (2007), pages 311–324 (cited on page 7).

[27] Bogdan Okreša Đurić. "Organizational Modeling of Large-Scale Multi-Agent Systems with Application to Computer Games". PhD thesis. Faculty of Organization and Informatics, University of Zagreb, 2018 (cited on page 2).

[28] Javier Palanca et al. "SPADE 3: Supporting the New Generation of Multi-Agent Systems". In: *IEEE Access* 8 (2020), pages 182537–182549 (cited on page 26).

[29] Ken Peffers et al. "A design science research methodology for information systems research". In: *Journal of management information systems* 24.3 (2007), pages 45–77 (cited on page 9).

[30] Marián Piliar. "Collaborative use of virtual and augmented reality in selected domains". In: *Proceedings of the Central European Seminar on Computer Graphics for students (CESCG)*. 2022. URL: `https://cescg.org/wp-content/uploads/2022/04/Piliar-Collaborative-use-of-virtual-and-augmented-reality-in-selected-domains-2.pdf` (cited on page 6).

[31] Mauricio Xavier Prado Ortega et al. "Application of the technical - pedagogical resource 3D holographic LED-fan display in the classroom". In: *Smart Learning Environments* 7 (2020) (cited on page 7).

[32] Siddharth S Rautaray and Anupam Agrawal. "Interaction with virtual game through hand gesture recognition". In: *2011 International Conference on Multimedia, Signal Processing and Communication Technologies*. IEEE. 2011, pages 244–247 (cited on page 7).

[33] John F Santore and Stuart C Shapiro. "Crystal cassie: Use of a 3-d gaming environment for a cognitive agent". In: *Papers of the IJCAI 2003 Workshop on Cognitive Modeling of Agents and Multi-Agent Interactions*. 2003, pages 84–91 (cited on page 8).

[34] Markus Schatten, Bogdan Okreša Đurić, and Rikardo Protrka. "Conceptual Architecture of a Cognitive Agent for Telemedicine based on Gamification". In: *Central European Conference on Information and Intelligent Systems*. Faculty of Organization and Informatics Varazdin. 2021, pages 3–10 (cited on page 5).

[35] Markus Schatten, Tomislav Peharda, et al. "Towards a Streamed Holographic 3D Game Engine". In: *Central European Conference on Information and Intelligent Systems*. Faculty of Organization and Informatics Varazdin. 2022, pages 17–22 (cited on pages 2, 5).

[36] Markus Schatten, Marínela Schatten, and Ljiljana Inkret-Martinčević. "Development of Gamified Cognitive Agents for Primary Schools-A Case Study". In: *Central European Conference on Information and Intelligent Systems*. Faculty of Organization and Informatics Varazdin. 2023, pages 3–8 (cited on page 5).

[37] Markus Schatten, Jurica Ševa, and Igor Tomičić. "A roadmap for scalable agent organizations in the internet of everything". In: *Journal of Systems and Software* 115 (2016), pages 31–41 (cited on page 2).

[38] Markus Schatten, Igor Tomičić, and Bogdan Okreša Đurić. "A review on application domains of large-scale multiagent systems". In: *Central european conference on information and intelligent systems*. 2017 (cited on page 2).

[39] Markus Schatten, Igor Tomičić, and Bogdan Okreša Đurić. “Towards Application Programming Interfaces for Cloud Services Orchestration Platforms in Computer Games”. In: *2020 Central European Conference on Information and Intelligent Systems*. Varaždin, Croatia: Faculty of Organization and Informatics, 2020, pages 9–14 (cited on page 21).

[40] Markus Schatten et al. “Agents as bots–an initial attempt towards model-driven mmorpg gameplay”. In: *International conference on practical applications of agents and multi-agent systems*. Springer. 2017, pages 246–258 (cited on page 2).

[41] Markus Schatten et al. “Automated MMORPG Testing–An Agent-Based Approach”. In: *International conference on practical applications of agents and multi-agent systems*. Springer. 2017, pages 359–363 (cited on page 2).

[42] Markus Schatten et al. “Large-Scale Multi-Agent Modelling of Massively Multi-Player On-Line Role-Playing Games–A Summary”. In: *Central European Conference on Information and Intelligent Systems*. 2017 (cited on page 2).

[43] Markus Schatten et al. “A cognitive agent’s infrastructure for smart mobility”. In: *Transportation Research Procedia* 64 (2022), pages 199–204 (cited on pages 5, 13).

[44] Swati Singhvi, Naman Gupta, and Shashank Mouli Satapathy. “Virtual Gaming Using Gesture Recognition Model”. In: *Advances in Distributed Computing and Machine Learning: Proceedings of ICADCML 2021*. Springer. 2022, pages 114–124 (cited on page 7).

[45] Tajana Šokec. *Modeliranje kontekstualno svjesnog agenta za razgovor na hrvatskom jeziku uz pomoć konačnog automata i strojnog učenja*. Rektorova nagrada Sveučilišta u Zagrebu. Mentor: Markus Schatten. 2019 (cited on pages 5, 13).

[46] Shinichiro Suganuma, Daisuke Sakamoto, and Haruhiko Shimoyama. “An embodied conversational agent for unguided internet-based cognitive behavior therapy in preventative mental health: feasibility and acceptability pilot trial”. In: *JMIR mental health* 5.3 (2018), e10454 (cited on page 7).

[47] Jun Wang et al. “Bidirectional Phase Compensation for Curved Hologram Generation in Holographic Display”. In: *IEEE Photonics Technology Letters* 34.11 (June 1, 2022), pages 579–582. ISSN: 1041-1135, 1941-0174. DOI: 10.1109/LPT.2022.3173439 (cited on page 6).

[48] Minggang Wang et al. “A novel hybrid method of forecasting crude oil prices using complex network science and artificial intelligence algorithms”. In: *Applied energy* 220 (2018), pages 480–495 (cited on page 2).

[49] Ling Yang, Haiwei Dong, and Abdulmotaleb El Saddik. “See in 3D: state of the art of 3D display technologies”. In: *Multimedia Tools and Applications* 75 (Oct. 15, 2015) (cited on page 7).

[50] Georgios N. Yannakakis and Julian Togelius. *Artificial Intelligence and Games*. Cham: Springer International Publishing, 2018. ISBN: 978-3-319-63519-4. DOI: 10.1007/978-3-319-63519-4 (cited on page 2).

[51] Anis Zarrad. “Game engine solutions”. In: *Simulation and Gaming* (2018), pages 75–87 (cited on page 20).

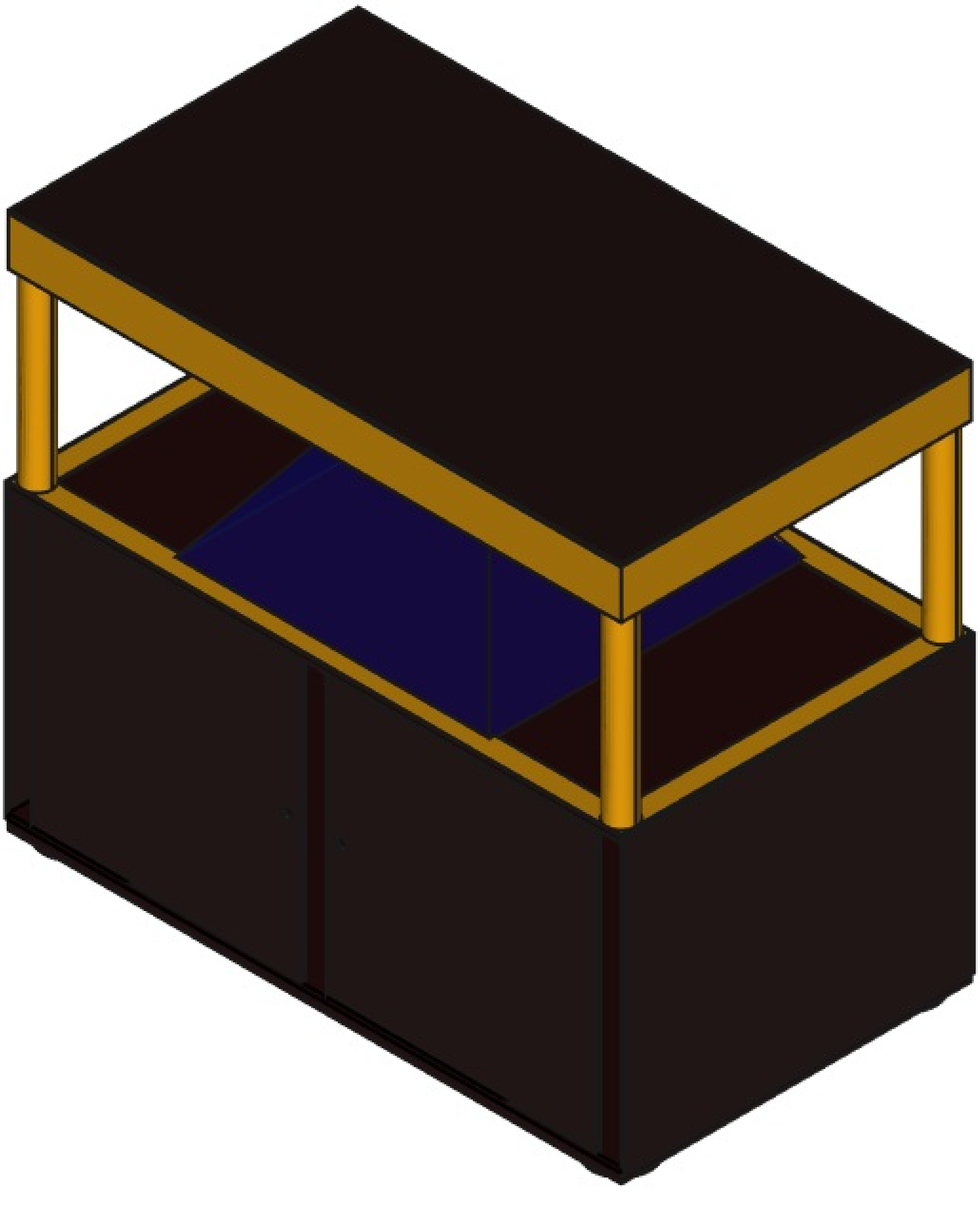
A. HoloGameP CAD Model

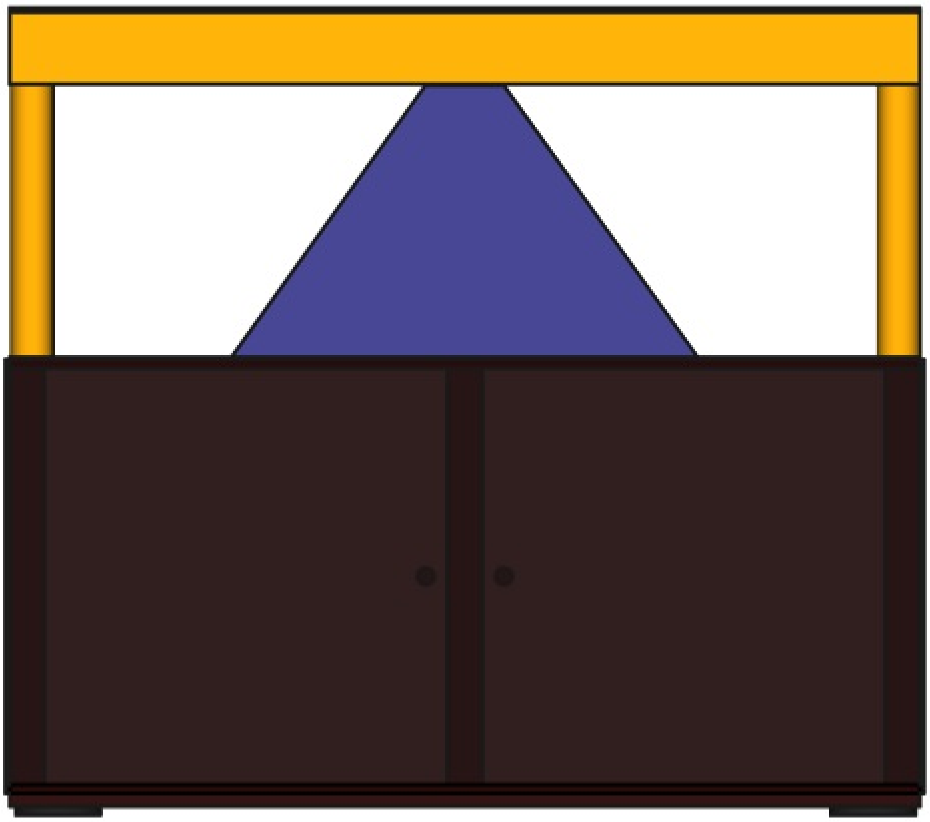
HoloPyramid – Hologramska igraća konzola

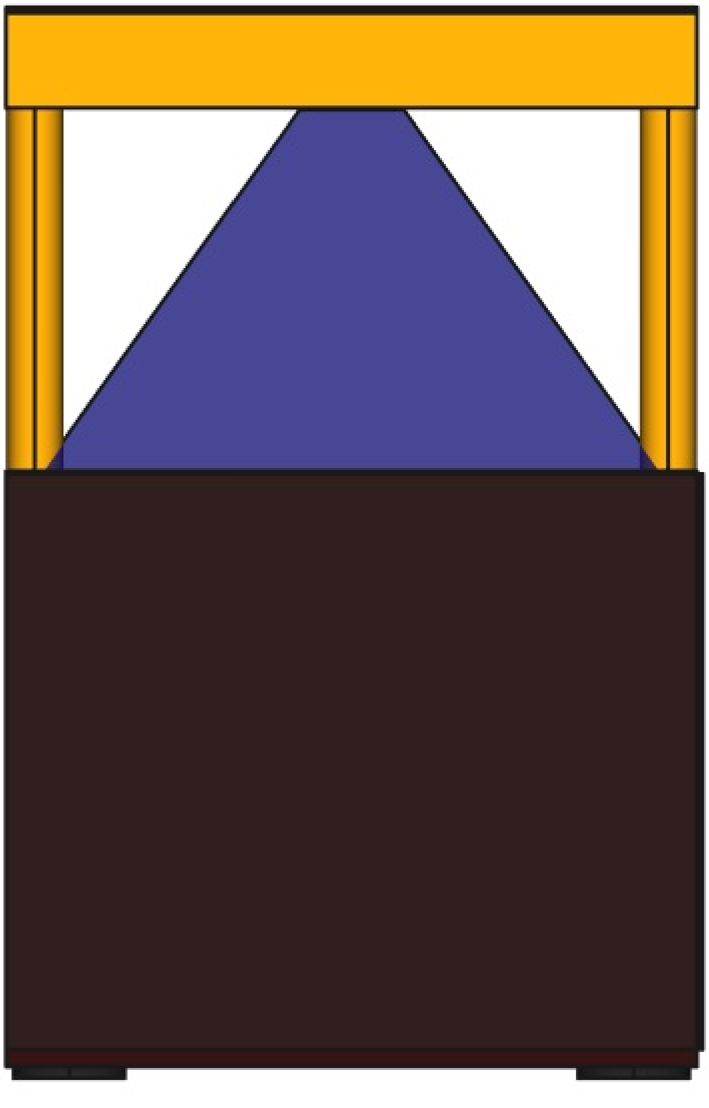
Konzola se sastoji od:

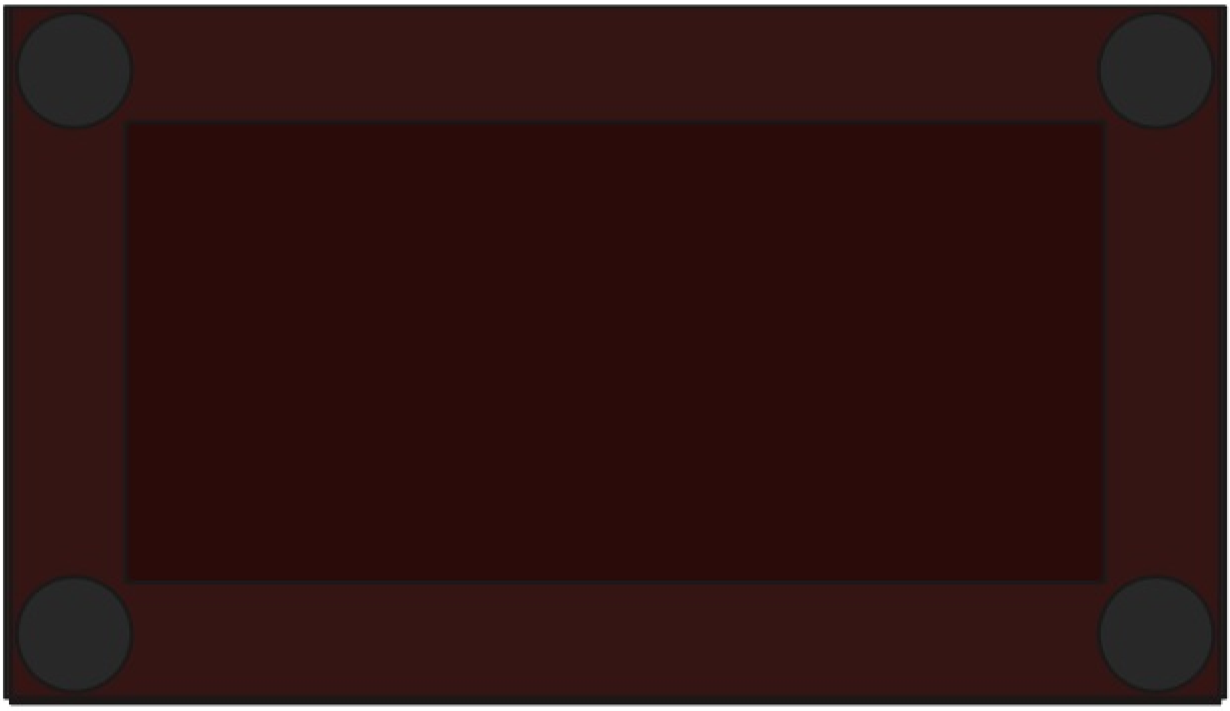
1. **Metalnog okvira** u koji će se položiti 85'' ekran (dimenzije ekrana 1892.8 x 1083.2 x 54.9 mm, težina oko 50 kg). Metalni okvir treba biti žuto cinčani (starinski izgled). Stalci trebaju biti šuplje cijevi kako bi se kroz njih mogli provući kablovi.
2. **Poklopca** kojim će se poklopiti ekran (drvo). Na poklopcu trebaju biti tokareni ukrasi koji mogu poslužiti kao ručke za skidanje poklopca, npr. Nešto oblika (starinski ormari)

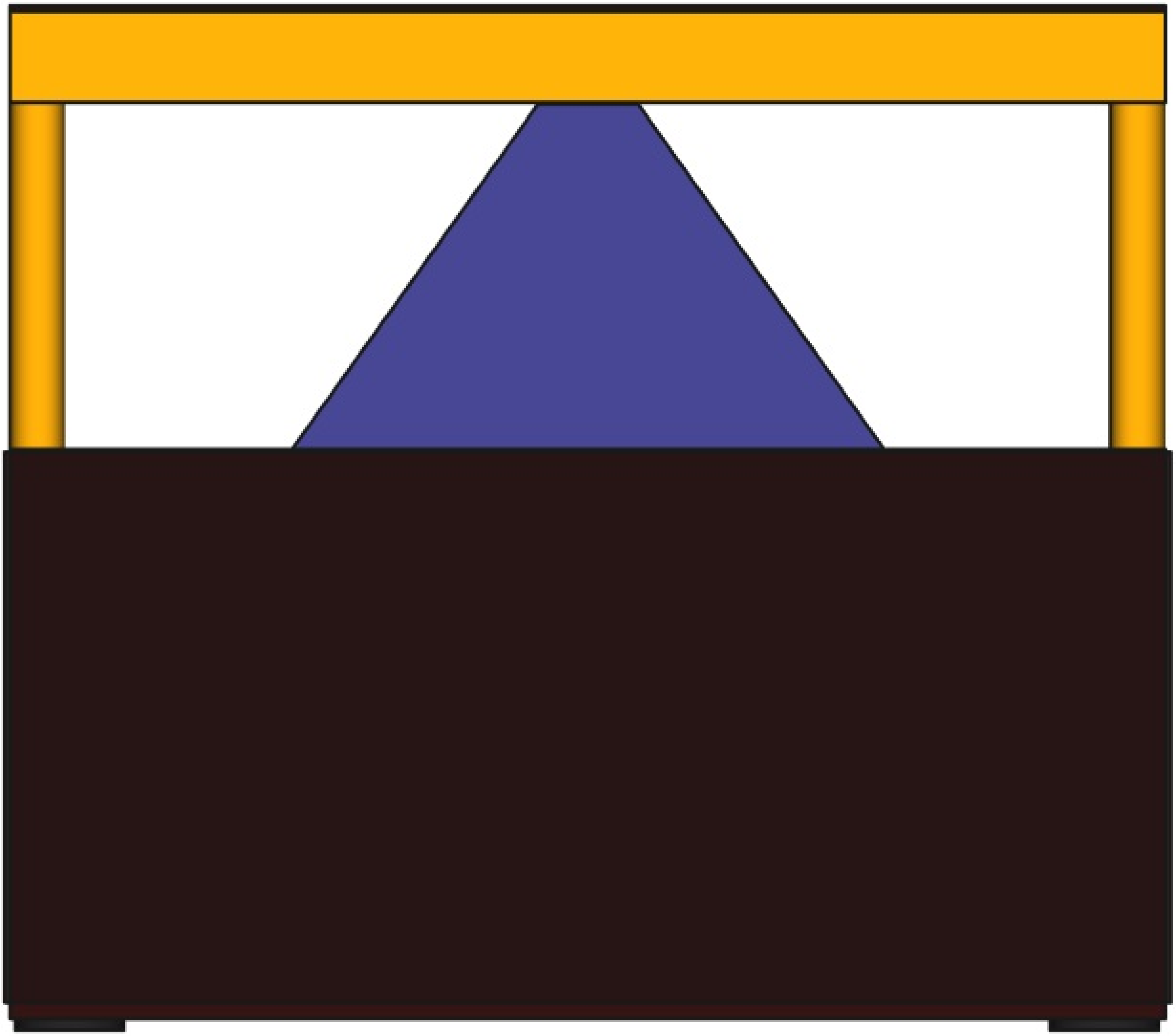
3. **Krnje piramide** od pleksiglasa na koju će se s ekrana projecirati slika (hologramska projekcija).
4. **Drvenog ormara postolja** na kojem će se montirati okvir i krnja piramida te u kojem će se pohraniti računalo koje će pokretati konzolu. Ormar treba imati vrata koja se mogu zaključati te biti ukrašen u starinskom stilu npr. (ne nužno s toliko detalja):

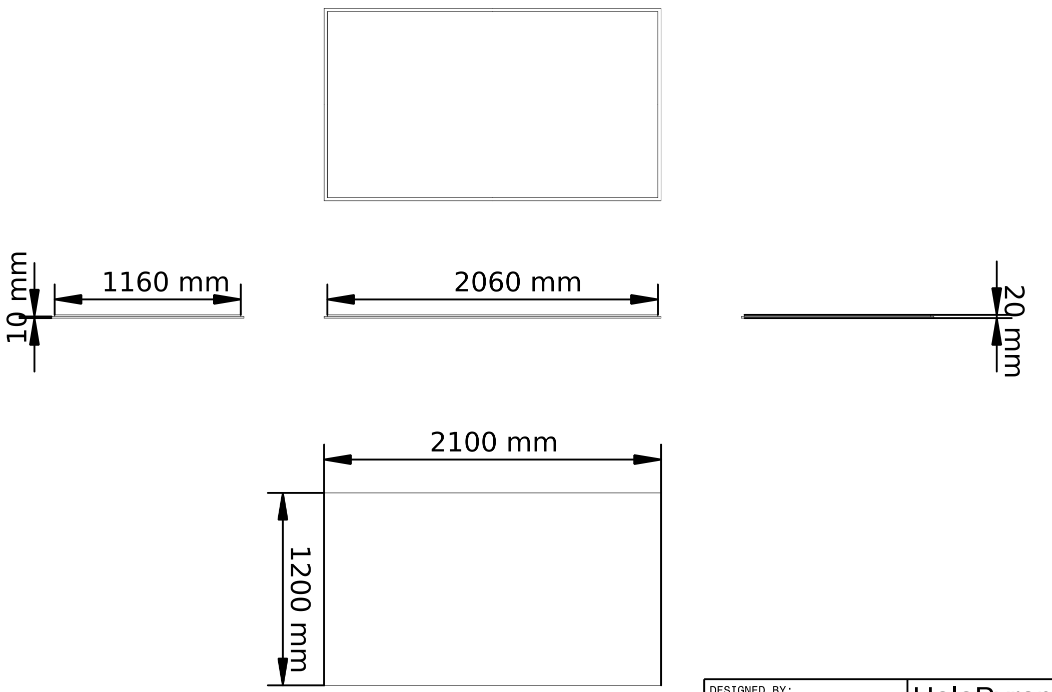












DESIGNED BY:
Markus Schatten

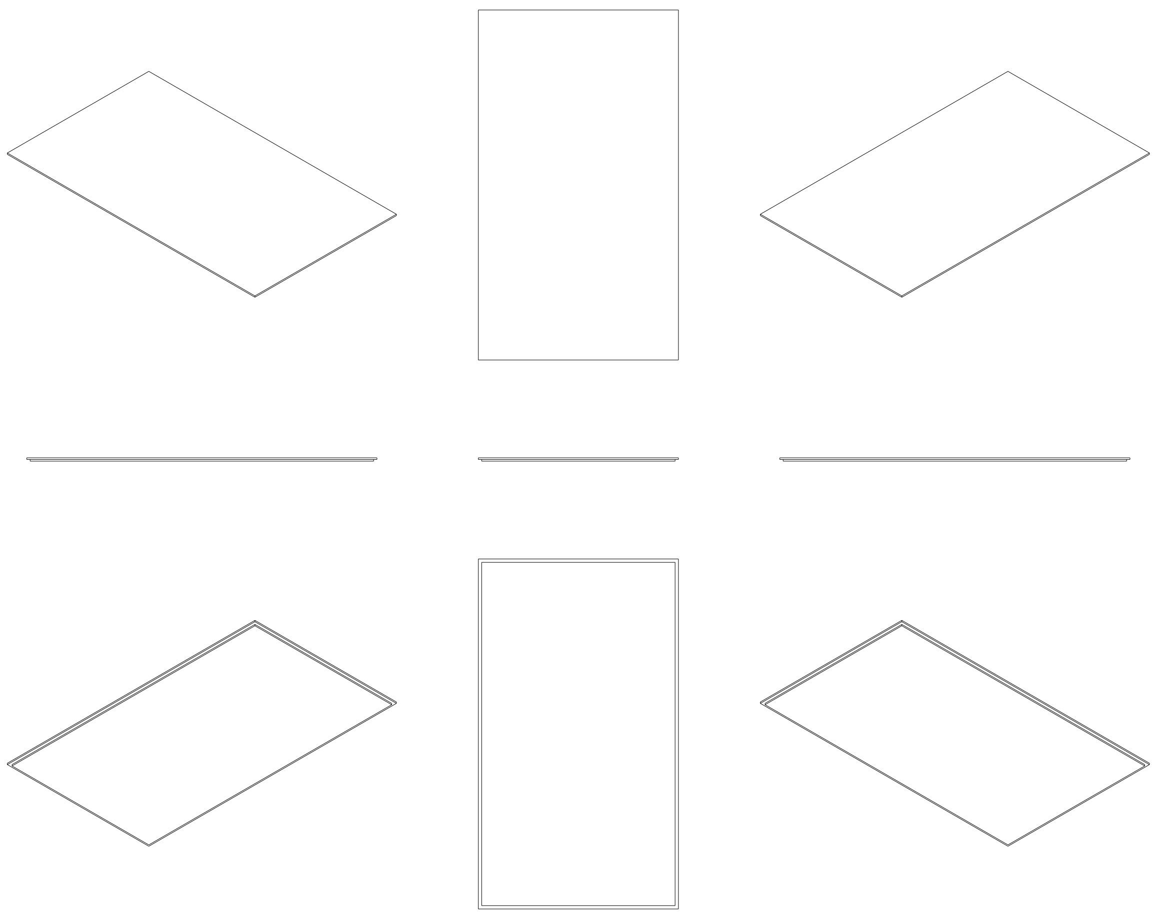
DATE:
24.11.2022.

HoloPyramid

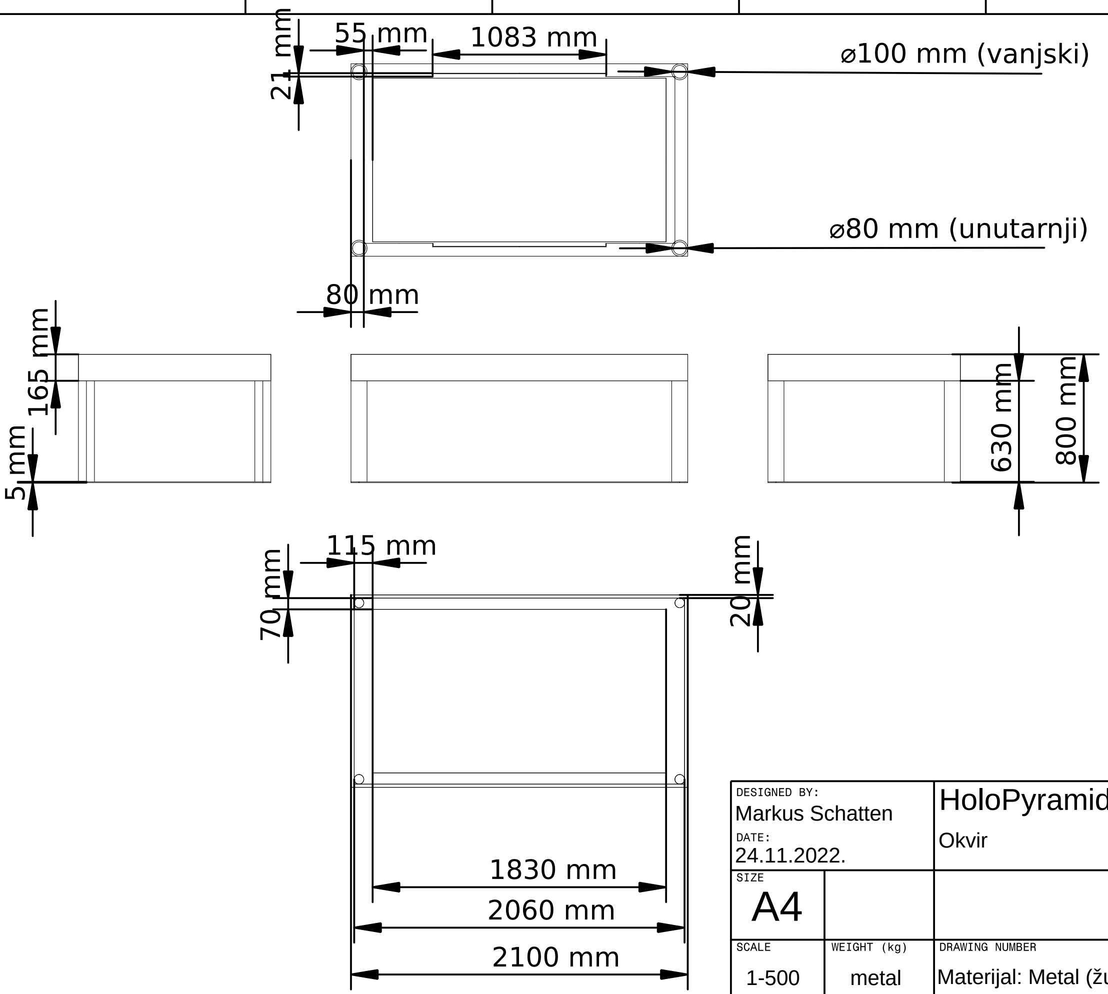
Poklopac

SIZE
A4

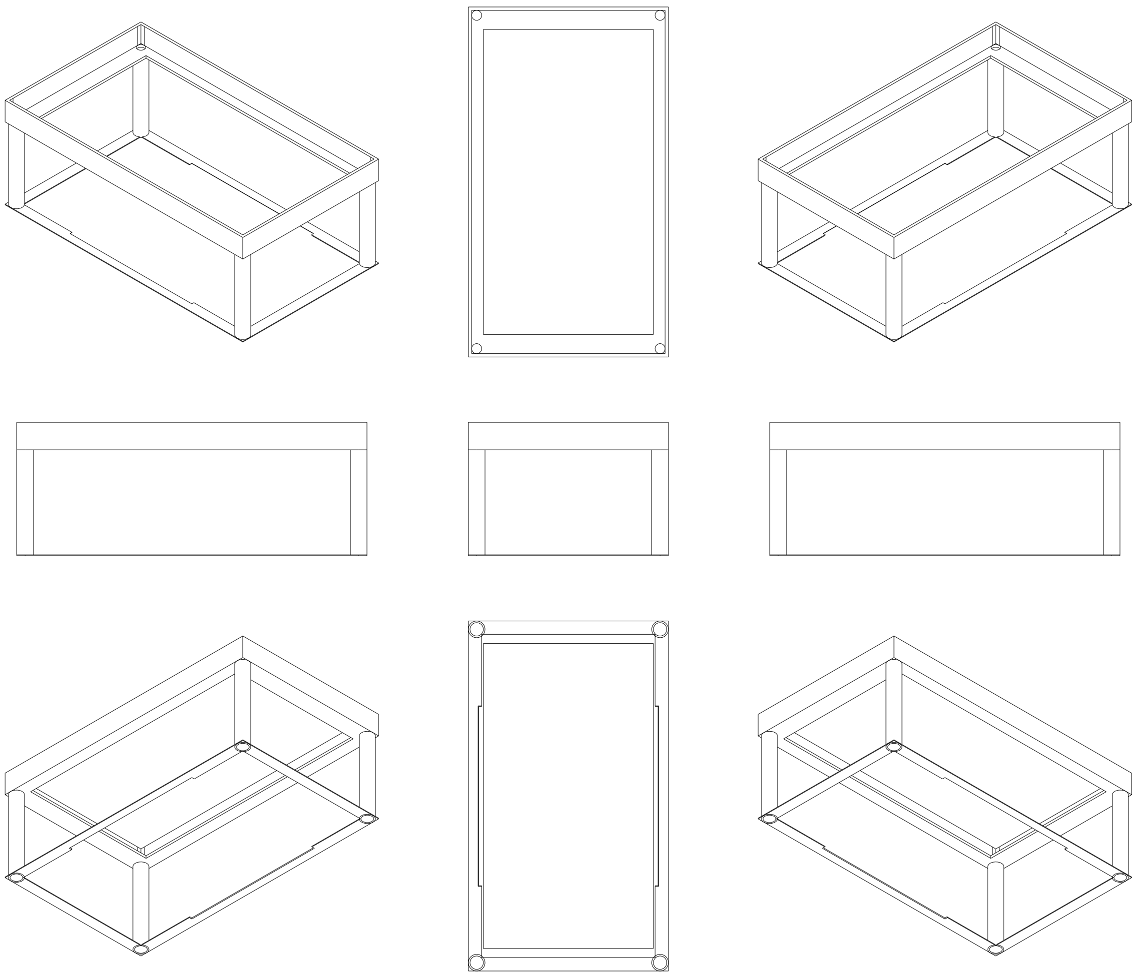
SCALE	WEIGHT (kg)	DRAWING NUMBER	SHEET
1-500	drvo	Materijal: Puno drvo	1



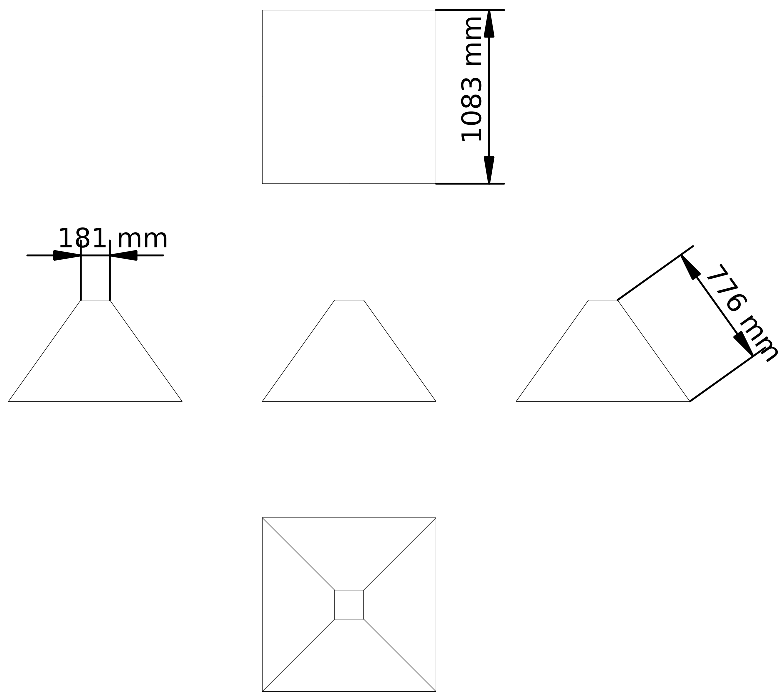
Created by: MARKUS SCHATTEN	Title: POKLOPAC			
Supplementary information: PUNO DRVO (RUCKE) PO POTREBI I MOGUCNOSTI UKRASI (JAJE)		Size: A0	Sheet: 1 / 1	Scale: 1:1.125
		Part number: PN005		
		Drawing number: DN005		
		Date: 28/11/2022		Revision: REV A



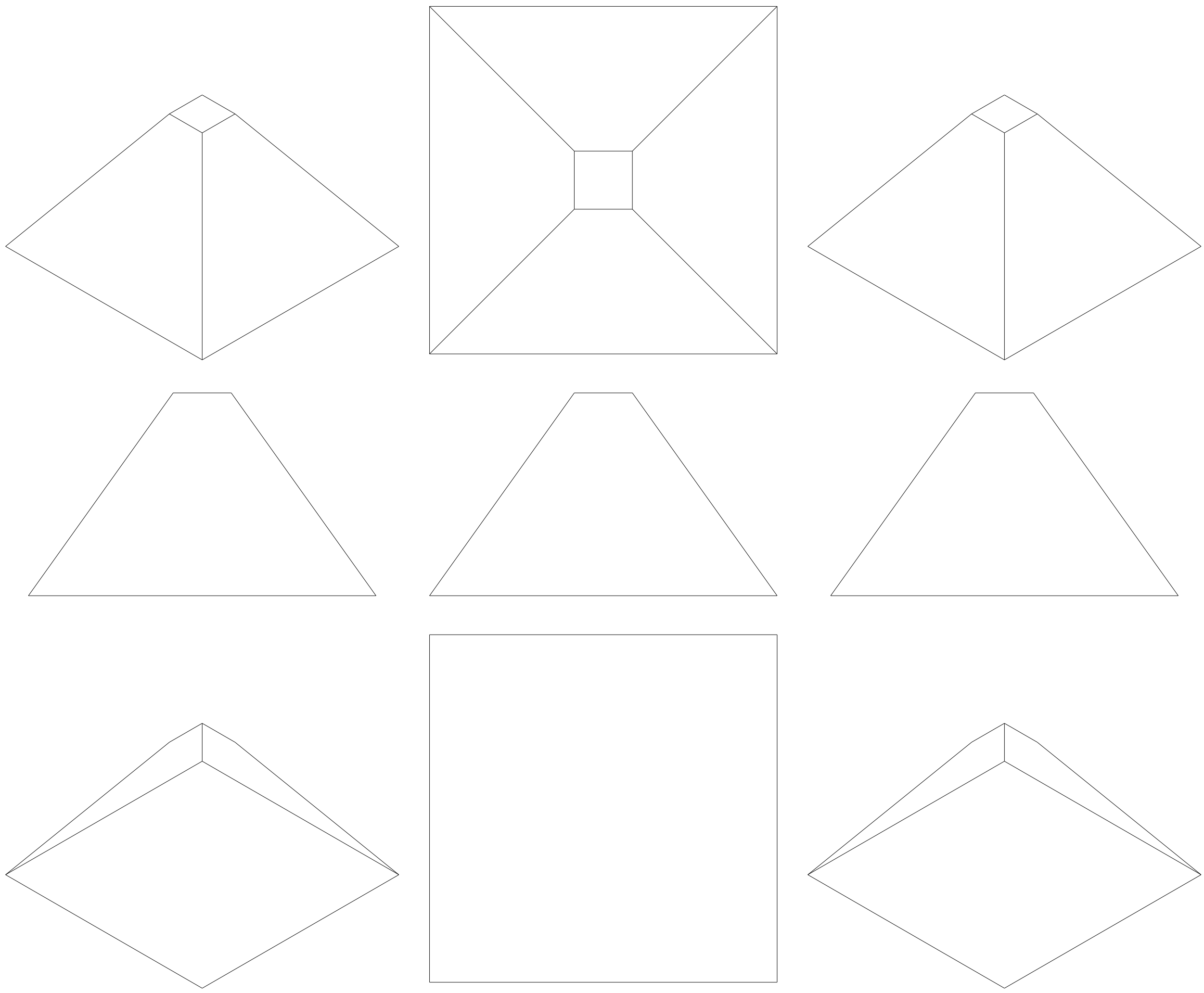
DESIGNED BY: Markus Schatten	HoloPyramid		
DATE: 24.11.2022.	Okvir		
SIZE A4			
SCALE 1-500	WEIGHT (kg) metal	DRAWING NUMBER Materijal: Metal (žuto cinčano)	SHEET 1



Created by: AUTHOR NAME	Title: DRAWING TITLE		
Supplementary information: FreeCAD DRAWING	Size: A0	Sheet: X / Y	Scale: SCALE
	Part number: PN		
	Drawing number: DN		
	Date: DD/MM/YYYY		Revision: REV A

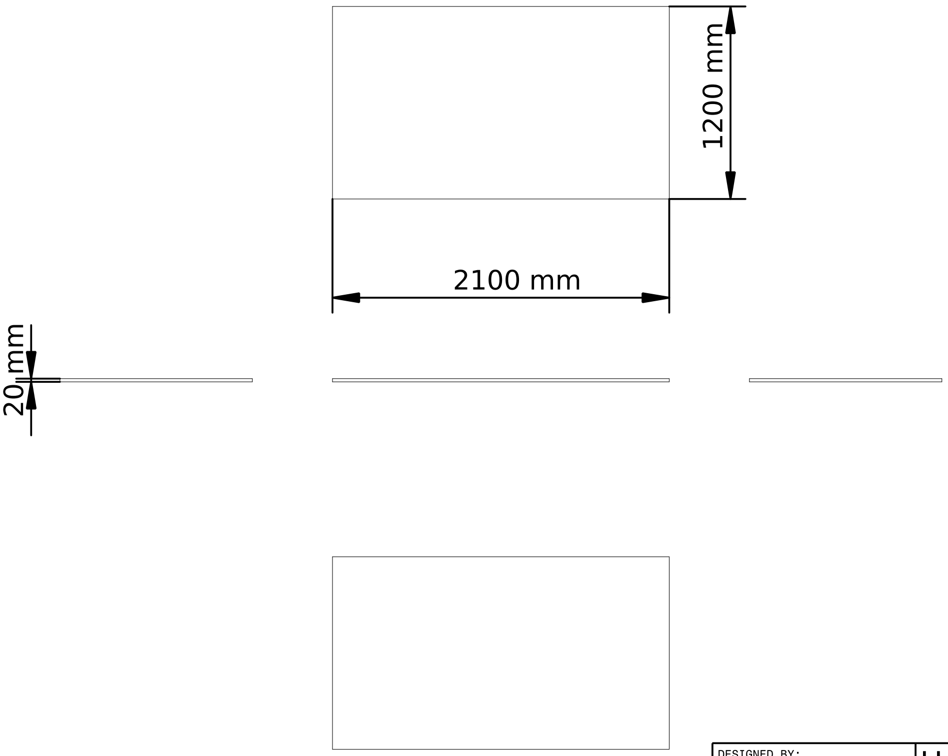


DESIGNED BY: Markus Schatten DATE: 24.11.2022.		HoloPyramid Krnja piramida		G _ F _
SIZE A4				E _ D _
SCALE 1-500	WEIGHT (kg) staklo	DRAWING NUMBER Materijal: Staklo / pleksiglas	SHEET 1	C _ B _
This drawing is our property; it can't be reproduced or communicated without our written consent.				A _



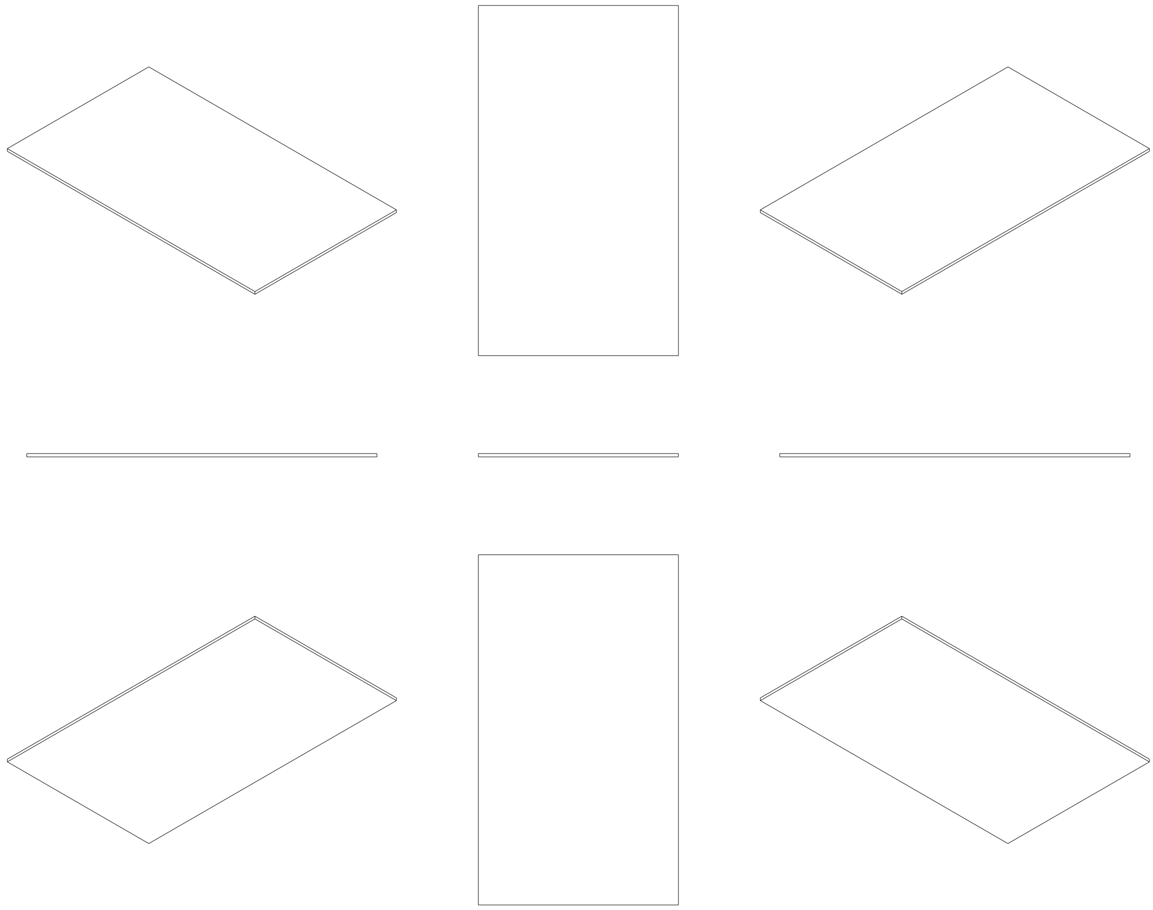
Created by:	Title:
MARKUS SCHATTEN	KRNJA PIRAMIDA

Supplementary information:	Size:	Sheet:	Scale:
STAKLO / PLEKSIGLAS	A0	1/1	1:200
	Part number: PN003		
	Drawing number: DN003		
	Date: 27/11/2011		Revision: REV A

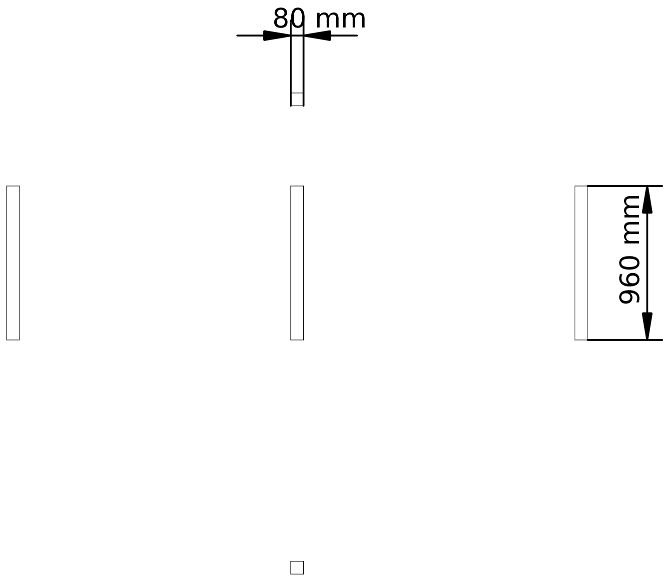


DESIGNED BY: Markus Schatten DATE: 28.11.2022.	HoloPyramid Donja i srednja ploča (2X)		
SIZE A4			
SCALE 1-500	WEIGHT (kg) drvo	DRAWING NUMBER Materijal: puno drvo	SHEET 1

G	_
F	_
E	_
D	_
C	_
B	_
A	_



Created by: MARKUS SCHATTEN	Title: DONJA I GORNJA PLOCA			
Supplementary information: PUNO DRVO 2X	Size: A0	Sheet: 1 / 1	Scale:	
	Part number: PN006			
	Drawing number: DN006			
	Date: 28/11/2022		Revision: REV A	



DESIGNED BY:
Markus Schatten
DATE:
29.11.2022.

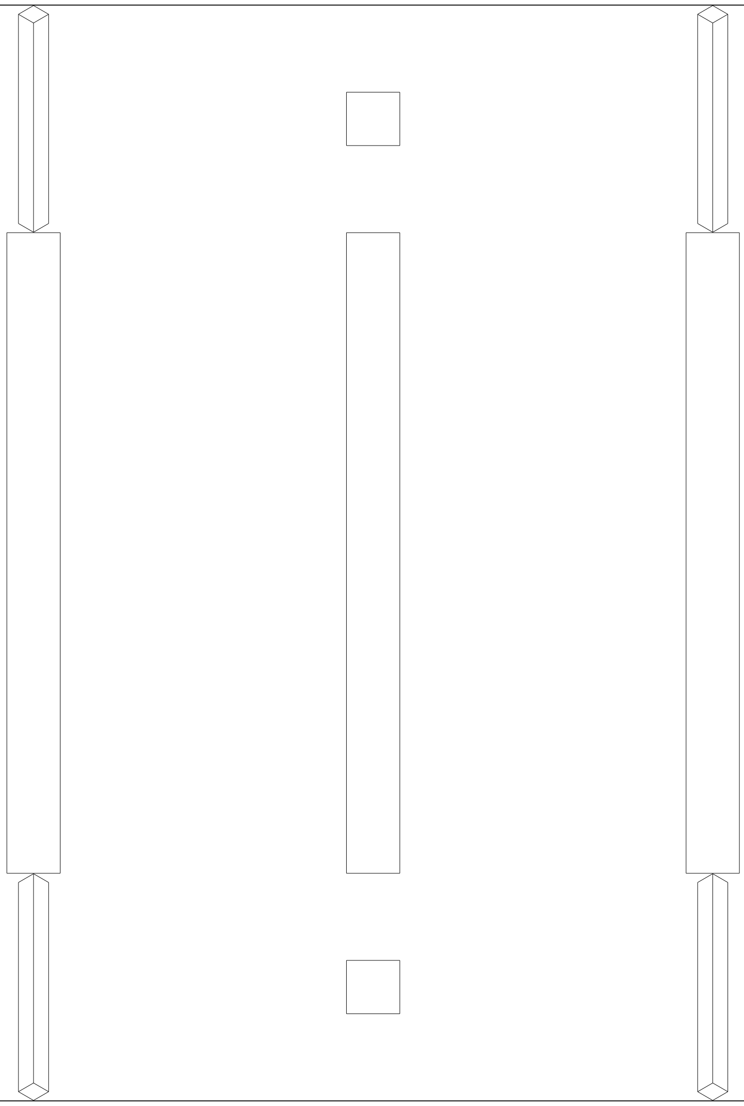
HoloPyramid

Daska (drzac) X6

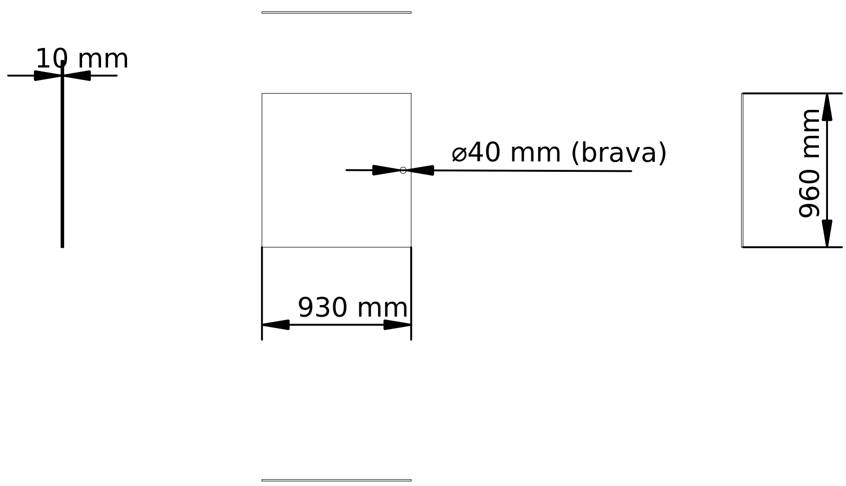
SIZE
A4

SCALE	WEIGHT (kg)	DRAWING NUMBER	SHEET
1-500	drvo	Materijal: puno drvo	1

G	_
F	_
E	_
D	_
C	_
B	_
A	_



Created by: MARKUS SCHATTEN	Title: DASKA X6			
Supplementary information: PUNO DRVO UNUTAR ORMARA	Size: A0	Sheet: 1 / 1	Scale:	
	Part number: PN006			
	Drawing number: DN006			
	Date: 28/11/2022		Revision: REV A	

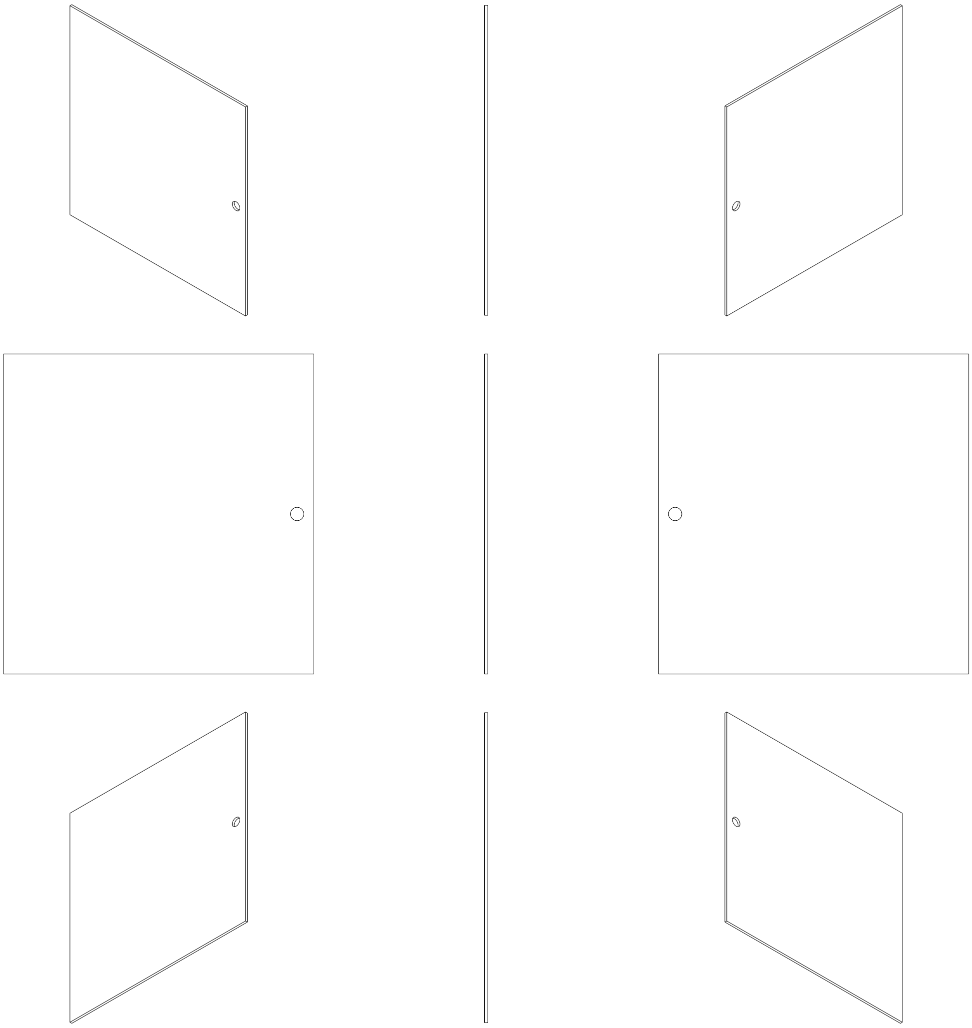


DESIGNED BY:
Markus Schatten
DATE:
28.11.2022.

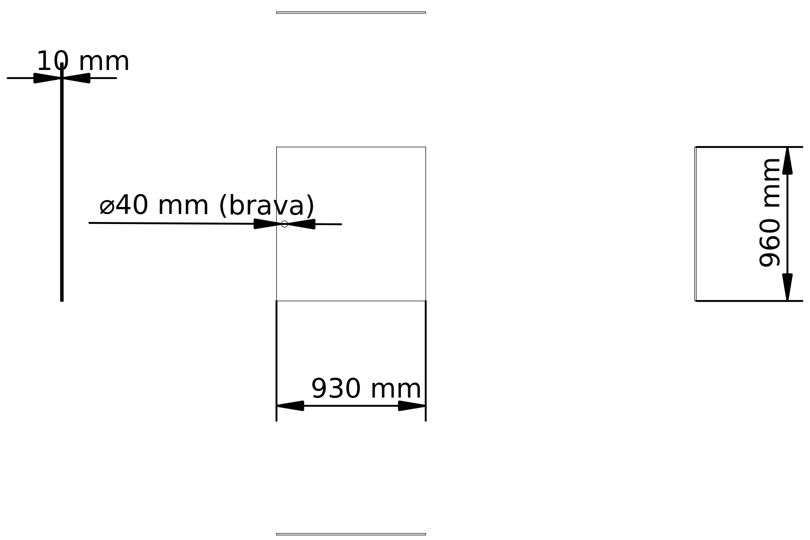
HoloPyramid

Lijeva vrata

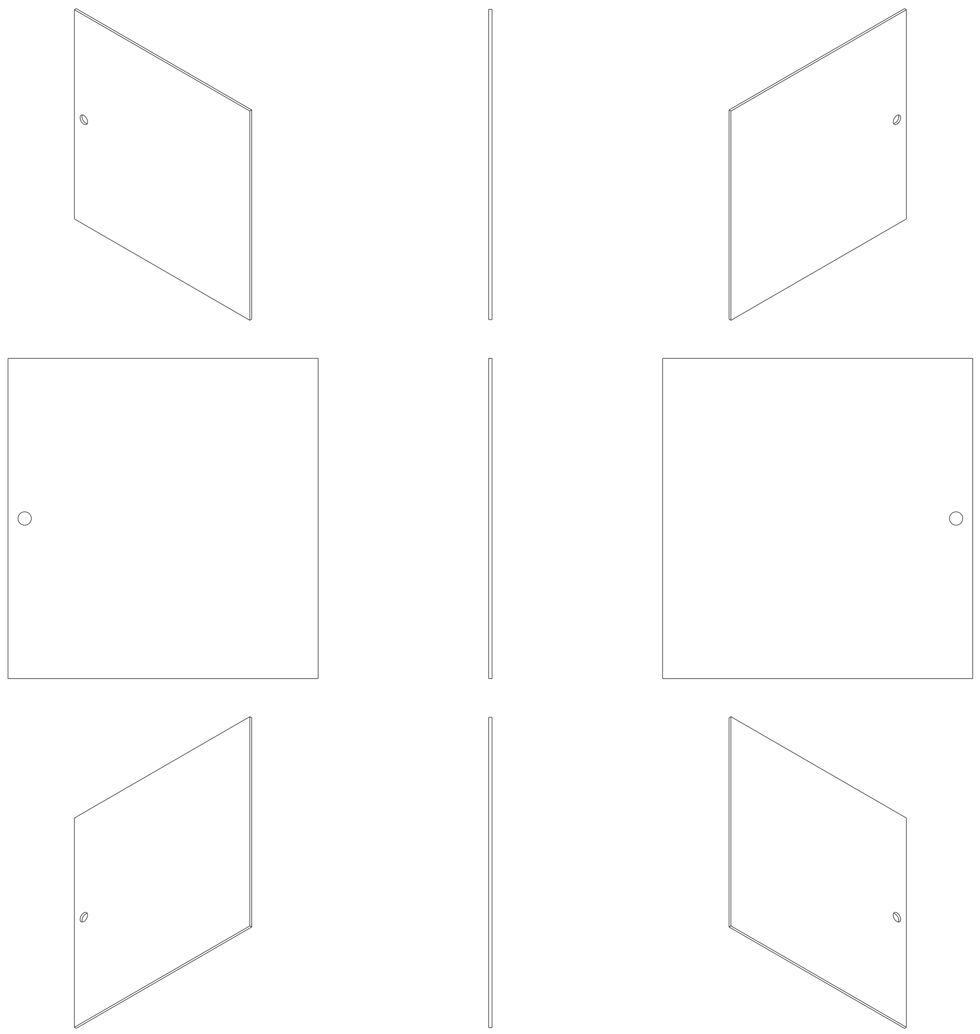
SIZE	SCALE	WEIGHT (kg)	DRAWING NUMBER	SHEET
A4	1-500	drvo	Materijal: puno drvo	1



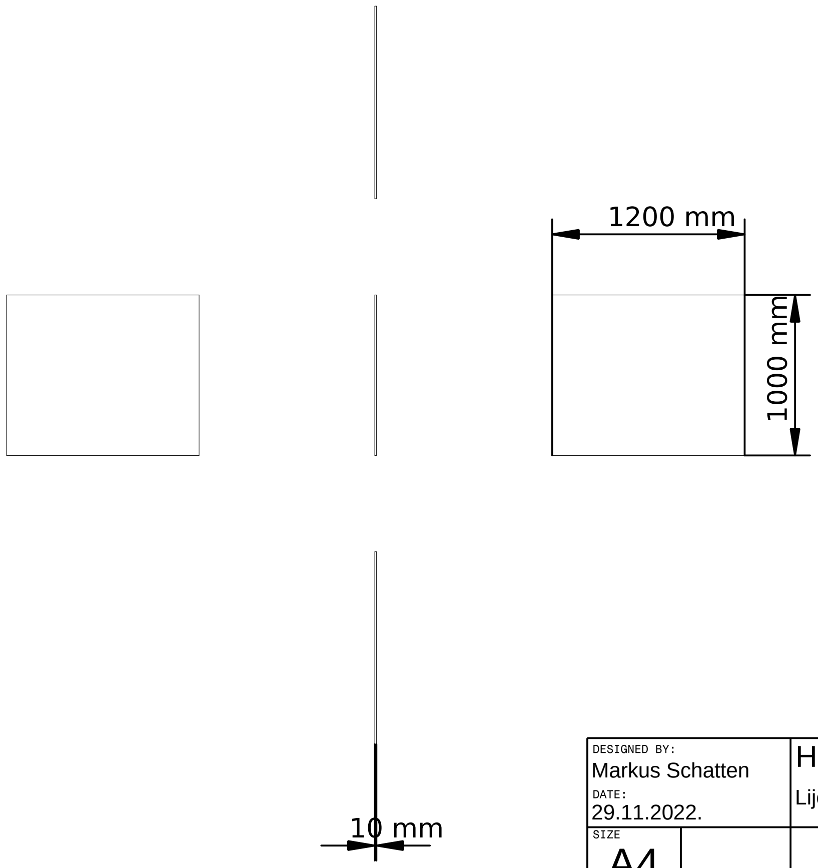
Created by:	Title:		
MARKUS SCHATTEN	LIJEVA VRATA		
Supplementary information: PUN DRVO eVENTUALNO UKRASI BRAVA (RUPA)	Size: A0	Sheet: 1 / 1	Scale:
	Part number: PN006		
	Drawing number: DN006		
	Date: 28/11/2022		Revision: REV A



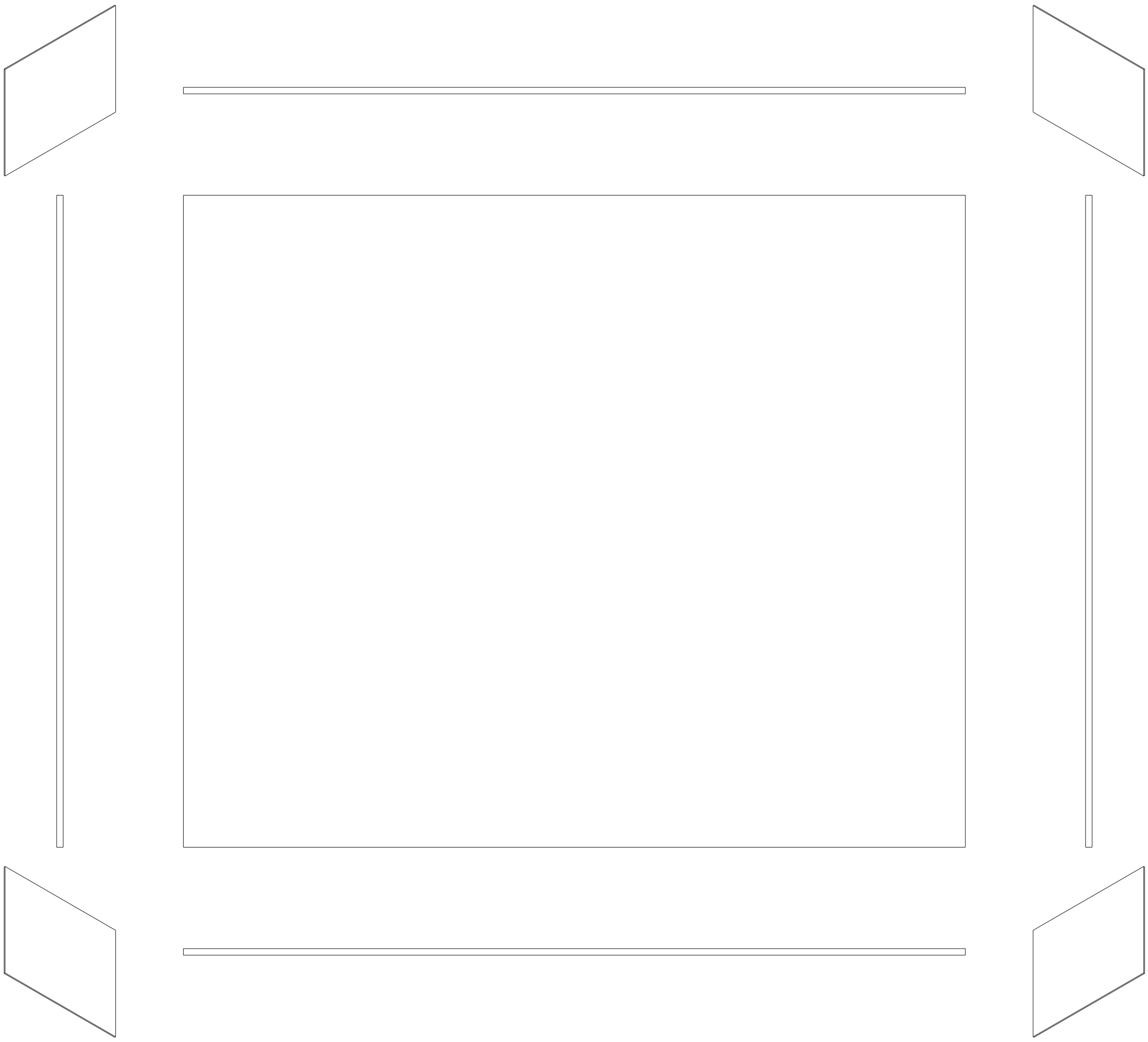
DESIGNED BY: Markus Schatten DATE: 28.11.2022.		HoloPyramid Desna vrata	
SIZE A4			
SCALE 1-500	WEIGHT (kg) drvo	DRAWING NUMBER Materijal: puno drvo	SHEET 1



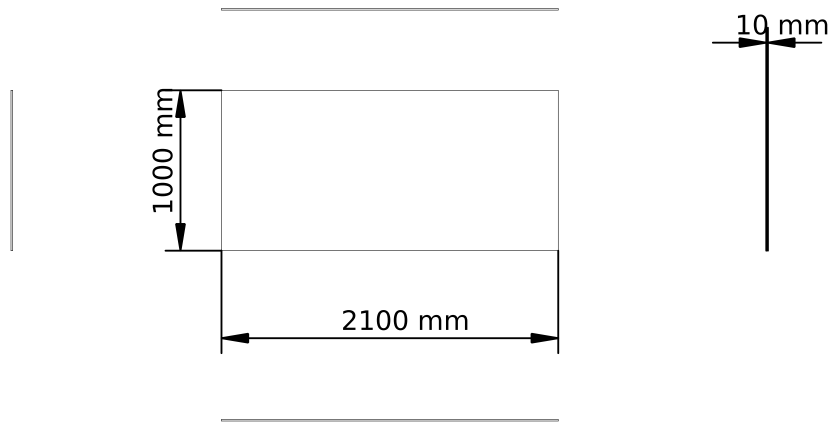
Created by:	Title:		
MARKUS SCHATTEN	DESNA VRATA		
Supplementary information:	Size: A0	Sheet: 1 / 1	Scale:
PUN DRVO eVENTUALNO UKRASI BRAVA (RUPA)	Part number: PN006 Drawing number: DN006		
	Date: 28/11/2022		Revision: REV A



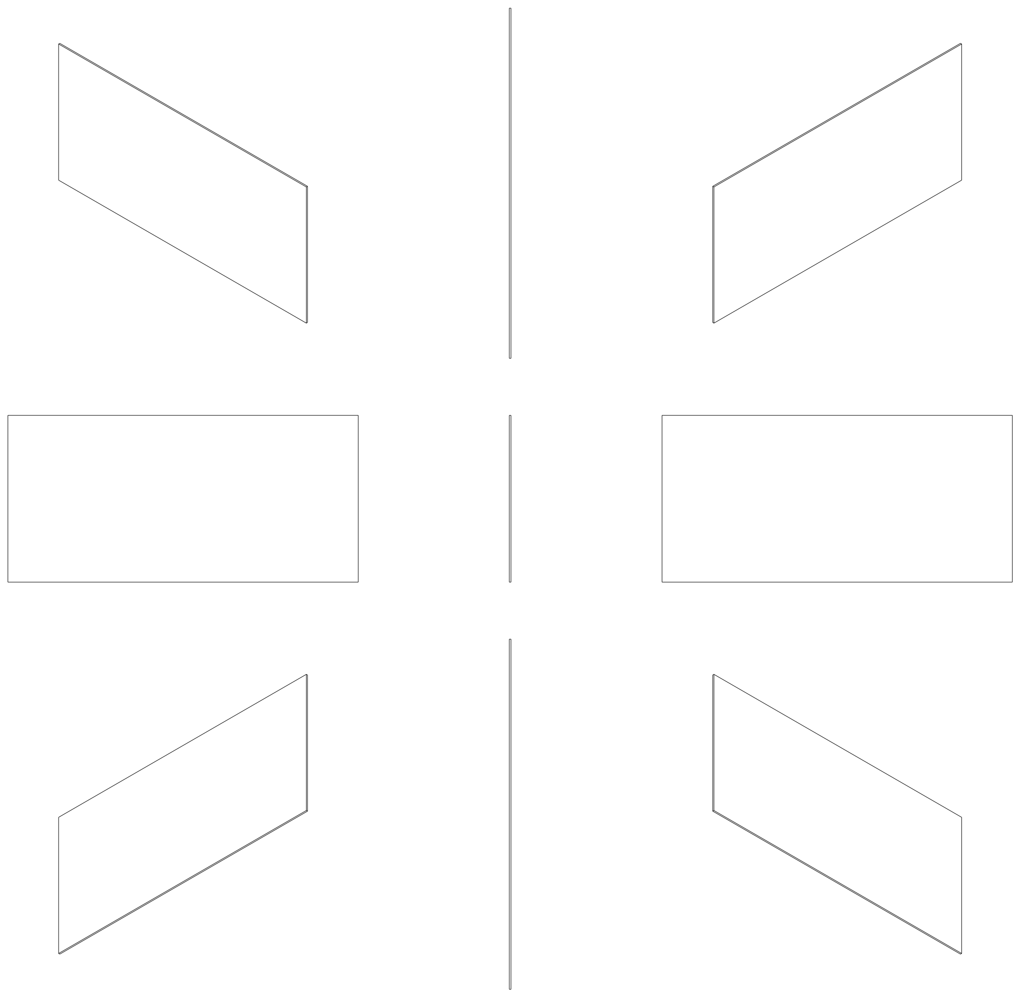
DESIGNED BY: Markus Schatten DATE: 29.11.2022.	HoloPyramid Lijeva i desna bočna strana (2X)	
SIZE A4		
SCALE 1-500	WEIGHT (kg) drvo	DRAWING NUMBER Materijal: puno drvo	SHEET 1



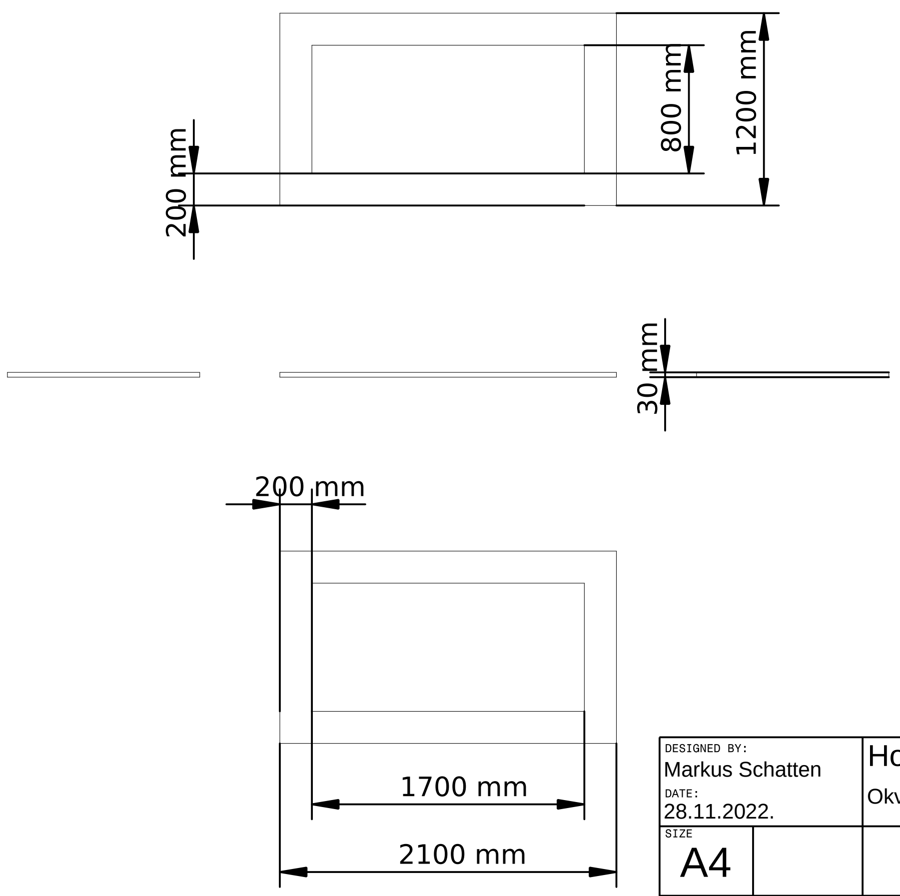
Created by:	Title:		
MARKUS SCHATTEN	LIJEVA I DESNA BOCNA STRANA		
Supplementary information: PUNO DRVO 2X	Size: A0	Sheet: 1 / 1	Scale:
	Part number: PN006		
	Drawing number: DN006		
	Date: 28/11/2022		Revision: REV A



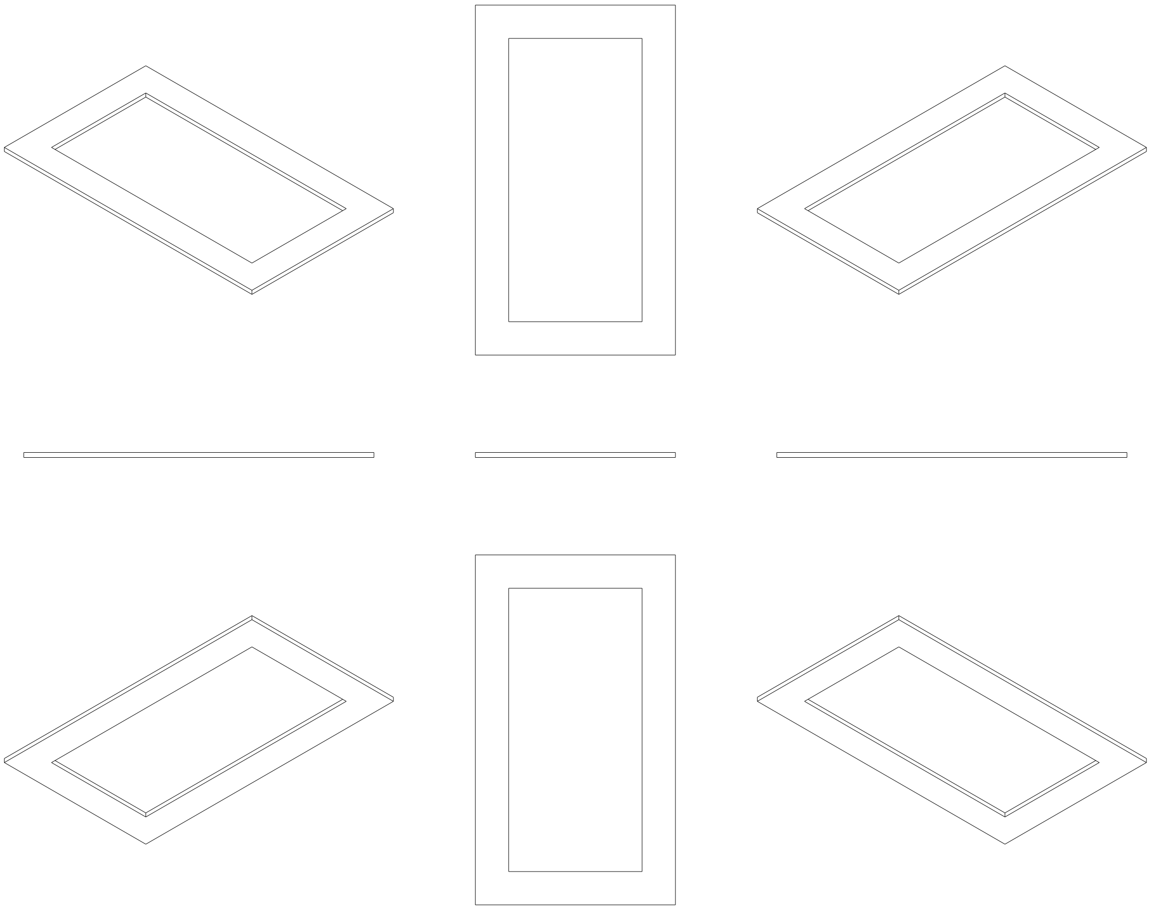
DESIGNED BY: Markus Schatten	HoloPyramid	G _
DATE: 29.11.2022.	Zadnja strana	F _
SIZE A4		E _
		D _
SCALE 1-500	WEIGHT (kg) drvo	C _
DRAWING NUMBER Materijal: puno drvo	SHEET 1	B _
		A _



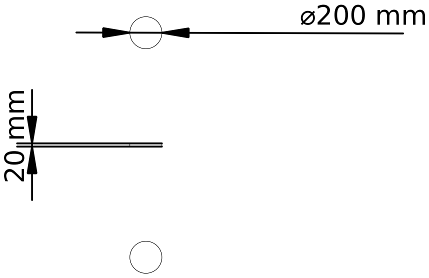
Created by:	Title:		
MARKUS SCHATTEN	ZADNJA STRANA		
Supplementary information: PUNO DRVO	Size: A0	Sheet: 1 / 1	Scale:
	Part number: PN006		
	Drawing number: DN006		
	Date: 28/11/2022		Revision: REV A



DESIGNED BY: Markus Schatten	HoloPyramid			G _
DATE: 28.11.2022.	Okvir donji			F _
SIZE A4				E _
				D _
SCALE 1-500	WEIGHT (kg) drvo	DRAWING NUMBER Materijal: puno drvo (daske)	SHEET 1	C _
				B _
				A _

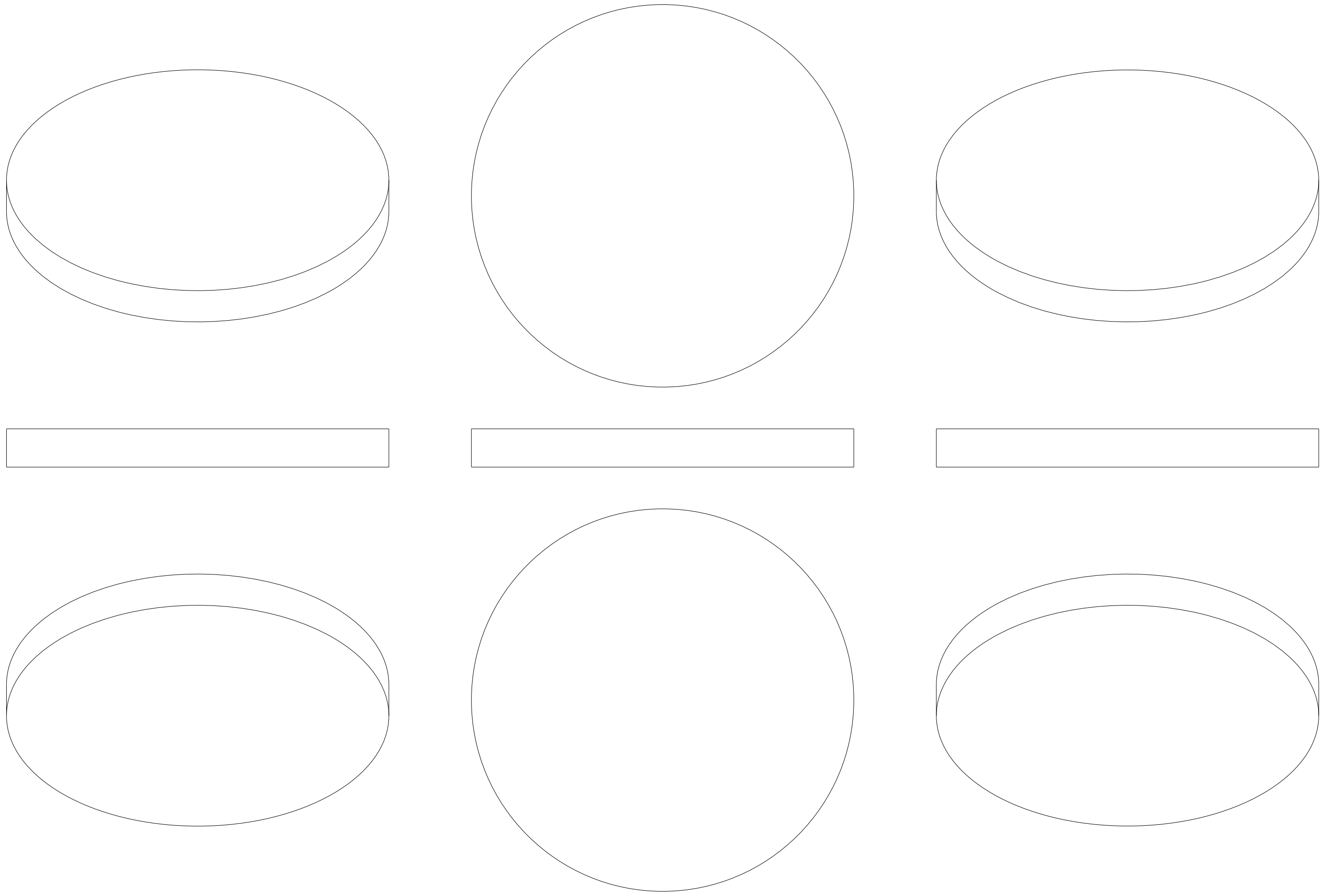


Created by:	Title:		
MARKUS SCHATTEN	OKVIR DONJI		
Supplementary information: PUNO DRVO ISPOD PL0010cE	Size: A0	Sheet: 1 / 1	Scale:
	Part number: PN006 Drawing number: DN006		
	Date: 29/11/2022		Revision: REV A



DESIGNED BY: Markus Schatten DATE: 28.11.2022.	HoloPyramid Nogica		
SIZE A4			
SCALE 1-500	WEIGHT (kg) drvo	DRAWING NUMBER Materijal: drvo / protuklizna guma	SHEET 1

G	_
F	_
E	_
D	_
C	_
B	_
A	_



Created by: MARKUS SCHATTEN	Title: NOGICA		
Supplementary information: X4 DONJA STRANA PROTUKLIZNA GUMA METAL ILI DRVO	Size: A0	Sheet: 1 / 1	Scale: 1:1.5
	Part number: PN004		
	Drawing number: DN004		
	Date: 28/11/2022		Revision: REV A