UDC 004.946

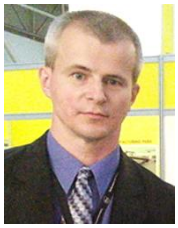
HEXAGONAL STRATEGY IN VR GAMING: A CHESS-INSPIRED APPROACH

N.A. Avchinnikov
Student of the National Children's Technopark, student of the Gymnasium No. 3 of Mogilev

M.C. Zamatay
Student of the National Children's Technopark, student of the School No. 7 of Volkovysk

M.S. Ilyasova
Assistant Professor, Department of Engineering Psychology and Ergonomics BSUIR, M.Sc.
m.iliasova@bsuir.by

P.V. Usenko
Software engineer, Department of Engineering Psychology and Ergonomics BSUIR, M.Sc.
f.usenko@bsuir.by



A.M. Prudnik
Associate Professor, Department of Engineering Psychology and Ergonomics BSUIR, Ph.D.
aleksander.prudnik@bsuir.by

N.A. Avchinnikov
Studies at gymnasium No. 3 in Mogilev. The area of scientific interest is virtual and augmented reality technologies.

M.S. Zamatay
Studies at secondary school No. 7 in Volkovysk. The area of scientific interest is the development of virtual and augmented reality applications.

M.S. Ilyasova
Earned her bachelor's degree from the Belarusian State University of Informatics and Radioelectronics, then proceeded to obtain a Master's degree, marked by the successful defense of her thesis. Her academic focus is on the intersection of Information Technology and Industrial Psychology.

P.V. Usenko
Upon graduating from the Belarusian State University of Informatics and Radioelectronics with a bachelor's degree, he advanced his education by securing a Master's degree, highlighted by the defense of his thesis. His scholarly interests are rooted in the fields of Information Systems, with a specific focus on user modeling and interface development.

A.M. Prudnik
Graduated from the Belarusian State University of Informatics and Radioelectronics. The area of scientific interest is related to human-computer interaction, information system interfaces, user interfaces, and front-end web development.

Abstract. The objective of the research was the development and evaluation of a virtual reality application, specifically a strategy game set on a hexagonal field. This game, reminiscent of chess, was conceptualized with the intention of augmenting the strategy genre and fostering the development of decision-making skills among players. The game unfolds on a hexagonal field, introducing a unique spatial element to strategic planning. It

incorporates elements characteristic of chess, including strategic planning, and the concepts of defense and attack. Players are required to think ahead, anticipate their opponent's moves, and devise strategies for both defense and attack. This interactive and immersive environment deepens the strategic experience, providing a platform for players to develop their decision-making skills. The development process encompassed the selection of appropriate tools, creation of 3D models of environments and characters, and the establishment of a development environment for the integration of these models. The mechanics of the game were implemented, and the project underwent rigorous testing. The outcomes of this research include the successful development of a unique strategy game that leverages virtual reality technology, and the evaluation of its effectiveness in augmenting the strategy genre and fostering decision-making skills among players. The results achieved and the solutions implemented were meticulously documented in a comprehensive report. This work contributes to the field of computer science by demonstrating the practical application of virtual reality technology in the context of strategic games.

Keywords: virtual reality, strategic game, game theory, strategic thinking, resource management, decision-making, cognitive perception, intellectual skills, immersive experiences, dynamic environment, uncertainty, analysis, forecasting.

Introduction. In the rapidly evolving landscape of Virtual Reality (VR), the integration of this technology into various facets of our lives offers unique opportunities for devising innovative research methodologies. This paper presents the project of a Strategic Game in VR, which aims not only to create an immersive gaming environment but also to provide a platform for analyzing the influence of virtual environments on strategic thinking and decision-making processes.

The utilization of VR as a platform for strategic games is an emerging research area. These strategies enable a reevaluation of the interaction experience within virtual space. The gameplay is predicated on the principles of strategic thinking and resource management. Each player's move is analyzed within the framework of game theory, facilitating the assessment of both immediate and long-term effects of strategic decisions. This game offers a comprehensive platform for studying the impact of virtual strategic scenarios on the enhancement of intellectual skills and decision-making in uncertain situations.

Furthermore, the project fosters a learning environment that encourages the development of strategic skills such as analysis, forecasting, and decision-making under conditions of limited information. In this project, VR serves as a tool that broadens cognitive perception capabilities and cultivates the application of strategic thinking in a dynamic environment.

Overview of the Current Practice and Potential of Combining Virtual Reality and Strategy Games. VR applications offer various forms of entertainment and learning, ranging from simulations to puzzles [1, 2, 3]. Among them, strategy games are a popular genre that allows users to experience immersive and interactive gameplay in VR [4, 5]. This paper aims to analyze several VR strategy games and their features, as well as the factors that influence the demand for such games.

Strategy games are games that require planning, decision making, and resource management to achieve a certain goal. They often involve elements such as combat, exploration, and diplomacy. In VR, strategy games can leverage the capabilities of VR devices to create realistic and engaging scenarios that challenge the user's spatial and strategic thinking [6]. Some examples of VR strategy games are *I Expect You To Die*, a puzzle game where the user plays as a secret agent; and *Brass Tactics*, a real-time strategy game where the user commands an army of steampunk units.

To analyze the VR strategy games, we will use the following criteria:

– Material interaction, which involves interacting with physical objects that are digitally augmented or connected.

– Cooperative interaction, which involves collaborating with other users or artificial intelligence agents.

– Multimodal interaction, which involves using multiple input modes at the same time, such as voice and gestures.

– Hybrid interaction, which combines several of the above methods to provide the most immersive experience.

These criteria will help us evaluate how VR strategy games use VR technology to create interactive and immersive gameplay, as well as how they develop the user's cognitive skills.

The demand for VR strategy games is influenced by several factors, such as the availability and quality of VR devices, the investments and support from leading gaming companies, and the content and design of VR strategy games. We will examine how these factors affect the market dynamics and user preferences for VR strategy games, and how the developers can adapt their products to meet the user requirements and expectations.

We expect that the demand for VR strategy games will continue to grow as VR technology improves and the VR content becomes more diverse and innovative [8]. We also expect that the VR strategy games will have a positive impact on the VR application market, as they provide a unique and compelling form of entertainment and learning.

Development Tool Selection. Before the development of the VR strategy game on a hexagonal field, careful consideration was given to the selection of development tools. This crucial step set the foundation for the subsequent stages of the project.

Unity was selected due to its robust toolset, cross-platform capabilities, and compatibility with multiple VR devices. Unity's intuitive GUI, comprehensive resources, and active developer community facilitate VR learning and development. As the project expanded, Unity's robust architecture ensured scalability without compromising performance.

C# was chosen for scripting in Unity. As a statically typed language, C# facilitates error detection and performance optimization, crucial for VR applications. Its broad developer community provides a rich knowledge base and resources, accelerating development. Being cross-platform, C# enables the creation of applications for various platforms, providing flexibility and wider audience reach.

Blender was utilized for the creation of 3D models due to its comprehensive suite of tools for 3D graphics, cost-effectiveness, and compatibility with other software. Its support for various data formats allows seamless integration with Unity, streamlining the workflow. Python, Blender's official language for extensions and scripts, enables custom tool creation and process automation. Blender's various rendering engines facilitate the creation of high-quality renders, crucial for VR applications. In essence, Blender's extensive functionality and open-source nature make it an ideal choice for this project.

Purpose of the Research. The primary aim of this research is to create a unique and innovative platform that enhances decision-making skills through a strategic game on a hexagonal field. An additional goal is to enrich the strategy genre by leveraging the new opportunities provided by VR, thereby deepening the strategic experience.

During the research process, the following issues need to be identified and addressed:

– Define the requirements for a reality visualization application in the context of a strategy game.

– Identify a set of interactive elements to be incorporated into the application to increase user engagement and improve decision-making skills.

Application Development. The gameplay of a VR application, designed as a strategic and educational game, can be segmented into several stages. The user materializes in a thematic room, adjacent to a table featuring a hexagonal field. During this phase, the user becomes familiar with the environment, as well as the characters and tokens that will be utilized throughout the game (Figure 1).

Figure 1 represents a VR environment designed for a strategy game. The setting is reminiscent of a medieval room, characterized by stone walls and floors. The room is equipped with two tables, one of which is set up for chess gameplay, signifying the gaming area. The other table, although vacant, is highlighted by overhead lighting. The room also features arched

wooden shelves embedded into the stone walls, providing a sense of authenticity to the medieval theme. The lighting conditions are dim, with spotlights strategically placed over the tables, thereby creating an intense and focused gaming atmosphere. This immersive and atmospheric setting is anticipated to enhance the strategic elements of the VR game, offering players a unique and engaging gaming experience. The development of such an environment underscores the potential of VR in transforming traditional game formats into more immersive experiences.

Figure 1. Location: A Medieval-Themed Room Designed for the VR Strategy Game

Characters. The figures in the game are designed in a low poly style, which is a technique that uses a limited number of polygons to represent shapes (Figure 2). This style is often used in 3D modeling and animation for its simplicity and efficiency, and it gives the figures a stylized, abstract look that is both modern and visually striking.

The Pyromancer is depicted as a mysterious figure in golden robes, with a hood that obscures the face. The orange glow of the staff it holds adds a dynamic element to the figure, highlighting the Pyromancer's elemental power. The Cleric is portrayed as a wise and serene figure in white robes, with an orange collar peeking out from underneath. The blue orb it holds contrasts with the overall color scheme, drawing attention to the Cleric's magical abilities. The Samurai is represented as a disciplined warrior in traditional attire, complete with a conical hat and a sword. The figure's pose suggests readiness for combat, emphasizing the Samurai's martial prowess.

Overall, the low poly style, combined with the distinct color schemes and poses of each figure, creates a visually appealing and immersive gaming experience. The figures are not only functional game pieces but also integral parts of the game's aesthetic and narrative. They are drawn with a level of detail that brings out their unique characteristics while maintaining the minimalist appeal of the low poly style. This style contributes to the game's overall visual identity and enhances the player's engagement with the game world.

Figure 2. Low Poly Character Design: The Pyromancer, Cleric, and Samurai

The user is then introduced to the hexagonal field. This unconventional field shape was chosen due to its complexity: hexagons offer a greater variety of moves on the field compared to a standard square grid. The unique combinations offered by the hexagonal shape allow for more directions and combinations for placing pieces on the field (Figure 3).

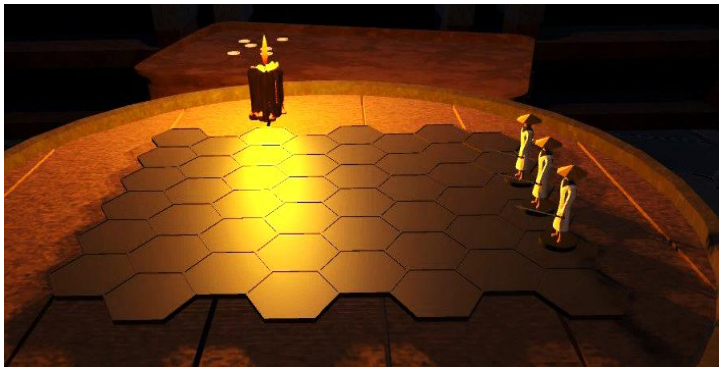


Figure. 3. Playing Field: A Hexagonal Battlefield for a Game

Subsequently, the user employs tokens to spawn characters on the playing field. At this juncture, the user becomes acquainted with the mechanics of character spawning, the limitations of their placement, and the turn-based nature of the game.

The following stage involves the movement and attack of characters. During this phase, the user learns about the varying ranges of character movement and attack. Movement and attack are executed by maneuvering the characters via VR (Figure 4). When an allied character moves to a tile occupied by an enemy character, an attack is initiated. This stage necessitates strategic and logical thinking from the user (Figure 5).

Having familiarized themselves with the hexagonal field, the token system, and the basic mechanics of character movement and attack, the user is now ready to delve deeper into the strategic aspects of the game. The following section elucidates the specific rules governing the game, providing a comprehensive understanding of the gameplay dynamics. These rules encompass the unique attributes and abilities of each character type (Pyromancer, Cleric, Samurai), the health management system, and the conditions for victory. By mastering these rules, the user can effectively strategize and make informed decisions during gameplay, enhancing their overall gaming experience.

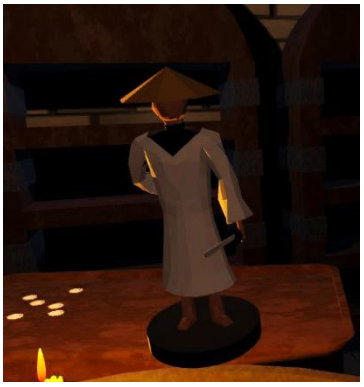


Figure 4. Character Movement

Figure 5. Gameplay

Let's proceed to the detailed game rules.

1 Start the Game: Initialize a 7x7 hexagonal game board.

2 Each player has 4 identical tokens.

3 All pieces start with 100% health.

4 Player 1's Turn Begins:

- Player 1 chooses a token.
- Player 1 chooses a piece type (Pyromancer, Cleric, Samurai).
- Player 1 places the chosen piece on the board within 2 hexagons from their edge.

5 Player 1 Moves a Piece:

- Player 1 chooses a piece to move.
- The piece moves according to its type. Samurai and Cleric can move 2 hexagons. Pyromancer can move 1 hexagon.

6 Player 1 Performs an Action:

- Player 1 chooses a piece to perform an action.
- If the piece is a Samurai or Pyromancer, it attacks an opponent's piece. The attacked piece's health decreases by 25%.
- If the piece is a Cleric, it heals an allied piece. The healed piece's health increases but cannot exceed 100%.

7 Player 2's Turn Begins:

- Player 2 chooses a token.
- Player 2 chooses a piece type (Pyromancer, Cleric, Samurai).
- Player 2 places the chosen piece on the board within 2 hexagons from their edge.

8 Player 2 Moves a Piece:

- Player 2 chooses a piece to move.
- The piece moves according to its type. Samurai and Cleric can move 2 hexagons. Pyromancer can move 1 hexagon.

9 Player 2 Performs an Action:

- Player 2 chooses a piece to perform an action.

10 Check Game Status: If a player has no pieces left on the board, the game ends. The other player wins. If not, return to step 4.

This sequence provides a high-level overview of the game flow and can be used as a basis for a flowchart. Please note that this sequence does not include all the details. These details would need to be added to the flowchart as necessary. Also, this sequence assumes that players can only use one token, move one piece, and perform one action per turn. If your game allows multiple actions per turn, you would need to adjust this sequence accordingly.

The game mechanics of the VR application, a strategic game on a hexagonal field, foster user engagement and motivation in the following ways:

- Learning through interaction.
- Unconventional gameplay.
- Achievement and reward.

Learning through interaction implies that users learn and develop through active engagement with the virtual world. They utilize strategic warfare techniques, which enables them to apply and enhance their skills. The unconventional gameplay stems from the varying outcomes of each game, due to the different ratios of available characters, as well as the user's experience and skill level. Achievement and reward suggest that with each victory, the player derives satisfaction from a tangible improvement in their skills, and each victory is recognized by the application itself.

As a result of learning through the application, users exhibit a significant enhancement in skills and knowledge in the realm of strategies. The application's gameplay ensures constant user interaction with the virtual world, promoting active learning and development. Users learn to apply a range of skills and techniques in battle, thereby improving their experience and knowledge in the field. User engagement in the gameplay is ensured through the mechanics of

learning through interaction, unconventional gameplay, and the satisfaction derived from achieving visible results and receiving recognition from the application.

In conclusion, the strategy game serves as an effective tool for learning and developing skills in the field of logical and strategic thinking.

The application leverages a variety of resources and the graphical capabilities of the Unity engine to ensure a high degree of visual realism. This contributes to the immersive interaction effect when using VR glasses.

Code Development. This section outlines the various scripts utilized in the project. These scripts are broadly classified into two categories: those facilitating direct user interaction with objects and the surrounding environment, and auxiliary scripts that support these interactions.

The BaseTicket script is a token script that manages user interactions with game objects in a VR environment (Figure 6). It is necessary for providing a natural and intuitive interface for users to interact with game objects, particularly in the context of placing characters on a game map.

The script works by detecting when a game object is being interacted with (i.e., picked up or dropped) by a user's hand. When the game object is dropped, the script checks the distance between the game object and all tiles in the game. If the game object is close enough to a tile, it attempts to spawn a character on that tile. If there is already an object on the tile, it breaks the loop and does nothing. Otherwise, it spawns a character and destroys the game object. This mechanism allows for intuitive placement of characters on a game map.

src > main.cpp

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using Valve.VR.InteractionSystem;

public
class BaseTicket : MonoBehaviour {
private
  bool isInteracting = false;
  [SerializeField] GameObject characterPrefab;
  [SerializeField] string characterName;
protected
  virtual void OnAttachedToHand(Hand hand) { isInteracting = true; }

protected
  virtual void OnDetachedFromHand(Hand hand) {
    isInteracting = false;

    foreach (var tile in Main.tiles) {
      string tileName = tile.Key;
      GameObject tileGO = tile.Value.gameObject;
      if (Vector3.Distance(this.transform.position, tileGO.transform.position) <
          0.15) {
        string[] pos = tileName.Split(",");
        int tempX = int.Parse(pos[0]);
        int tempY = int.Parse(pos[1]);

        GameObject objOnTile =
            Main.tilesObjects[tempX.ToString() + ", " + tempY.ToString()];
        if (objOnTile != null) {
          break;
        } else {
          Main.SpawnCharacter(characterPrefab, characterName, false, false,
                              tempX, tempY, "Ally");
          Destroy(this.gameObject);
          break;
        }
      }
    }
  }
}
```

Figure 6. Token script

The BaseTicket script is a helper script that implements some basic features and checks (Figure 7). It is necessary for managing interactions with game objects in a VR environment. The script is designed to work with the Valve VR Interaction System, which allows for natural interactions with virtual objects.

src › main.cpp

```
using Valve.VR.InteractionSystem;

public
class BaseTicket : MonoBehaviour {
private
  bool isInteracting = false;
  [SerializeField] GameObject characterPrefab;
  [SerializeField] string characterName;
protected
  virtual void OnAttachedToHand(Hand hand) { isInteracting = true; }

protected
  virtual void OnDetachedFromHand(Hand hand) {
    isInteracting = false;

    foreach (var tile in Main.tiles) {
      string tileName = tile.Key;
      GameObject tileGO = tile.Value.gameObject;
      if (Vector3.Distance(this.transform.position, tileGO.transform.position) <
          0.15) {
        string[] pos = tileName.Split(",");
        int tempX = int.Parse(pos[0]);
        int tempY = int.Parse(pos[1]);

        GameObject objOnTile =
            Main.tilesObjects[tempX.ToString() + ", " + tempY.ToString()];
        if (objOnTile != null) {
          break;
        } else {
          Main.SpawnCharacter(characterPrefab, characterName, false, false,
                              tempX, tempY, "Ally");
          Destroy(this.gameObject);
          break;
        }
      }
    }
  }
}
```

Figure 7. Helper script

The script works by detecting when a game object is being interacted with (i.e., picked up or dropped) by a user's hand. When the game object is dropped, the script checks the distance between the game object and all tiles in the game. If the game object is close enough to a tile, it attempts to spawn a character on that tile. If there is already an object on the tile, it breaks the loop and does nothing. Otherwise, it spawns a character and destroys the game object. This mechanism allows for intuitive placement of characters on a game map.

HexTileMapGenerator is a script that generates a hexagonal field according to specified dimensions (Figure 8). It is necessary for creating a grid-based game environment where each hexagon represents a tile that can be interacted with. The script works by instantiating a hexagonal prefab object and positioning it in a staggered pattern based on the map width and height. It also assigns a name and a parent transform to each hexagon for easy identification and organization. The script can be attached to any game object and executed in the Start() method.

src > main.cpp

```
GameObject hexTilePrefab;
[SerializeField] int mapWidth = 25;
[SerializeField] int mapHeight = 12;

float tileXOffset = 0.166f;
float tileZOffset = 0.15f;

Vector3 realPos;
void Start() { CreateHexTileMap(); }

void CreateHexTileMap() {
  realPos = this.gameObject.transform.position;

  for (int x = 0; x <= mapWidth; x++) {
    for (int z = 0; z <= mapHeight; z++) {
      GameObject tempGameObject = Instantiate(hexTilePrefab);

      if (z % 2 == 0) {
        tempGameObject.transform.position =
            new Vector3((realPos.x + x) * tileXOffset, realPos.y,
                        (realPos.z + z) * tileZOffset);
      } else {
        tempGameObject.transform.position =
            new Vector3((realPos.x + x) * tileXOffset + tileXOffset / 2,
                        realPos.y, (realPos.z + z) * tileZOffset);
      }
      SetTileInfo(tempGameObject, x, z);
    }
  }
}

void SetTileInfo(GameObject gameObject, int x, int z) {
  gameObject.transform.parent = transform;
  gameObject.name = x.ToString() + ", " + z.ToString();
}
}
```

Figure 8. The script that generates a hexagonal field

Conclusion. The successful execution of this research project has effectively showcased the potential of VR applications in the realm of strategic skills development. Utilizing the Unity game engine and the C# programming language, an application was developed that not only enhances strategic expertise but also provides players with a unique platform for honing their decision-making skills in an interactive and immersive environment.

Employing Blender, a 3D object creation software, we managed to accurately recreate the environmental location and game models. This contributes to the creation of an immersive experience, providing users with a deeply engaging educational journey.

With the aid of this VR application, users can deepen their strategic experience and develop and refine their decision-making skills in an interactive setting, thereby contributing to the popularization of strategies in VR.

While the project's objectives have been met, there are several areas that could be improved. For instance, more sophisticated 3D modeling and animation techniques could be applied, and more complex physics simulations could be enabled to enhance the realism of the user experience.

In summary, this project signifies a substantial advancement in the utilization of VR for creating strategy games. It exemplifies how modern technology can be leveraged to popularize this genre by fostering the user's strategic experience and logical and strategic thinking.

References

[1] Whitton, Nicola. Digital Games and Learning: Research and Theory. New York: Routledge, 2014.

[2] Truman, Samuel, Nicolas Rapp, Daniel Roth, and Sebastian von Mammen. “Rethinking Real-Time Strategy Games for Virtual Reality,” FDG '18: Proceedings of the 13th International Conference on the Foundations of Digital Games. August 7, 2018. Pages 1–6. https://doi.org/10.1145/3235765.3235801.

[3] Checa, David, and Andres Bustillo. “A Review of Immersive Virtual Reality Serious Games to Enhance Learning and Training.” Multimedia Tools and Applications 79, no. 9-10 (December 5, 2019): 5501–27. https://doi.org/10.1007/s11042-019-08348-9.

[4] Lim, Russell, Wong Chung Wei, Tet-Khuan Chen, and Leo Gertrude David. “The Important and Relevance of Strategy in Role-Playing Games.” 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), December 2, 2022. https://doi.org/10.1109/icmnwc56175.2022.10032017.

[5] Kuan, Yen-Ting, Yu-Shuen Wang, and Jung-Hong Chuang. “Visualizing Real-Time Strategy Games: The Example of StarCraft II.” 2017 IEEE Conference on Visual Analytics Science and Technology (VAST), October 2017. https://doi.org/10.1109/vast.2017.8585594.

[7] Kim, Youngkwang, Yeo-Song Yoon, Tea-Gyeoung Oh, Yeung-Hwan HwangBo, and Jeong-Hee Hwang. “Real-Time VR Strategy Chess Game Using Motion Recognition.” Journal of Digital Contents Society 18, no. 1 (February 28, 2017): 1–7. https://doi.org/10.9728/dcs.2017.18.1.1.

[8] Squire, Kurt D. “Video Game–Based Learning: An Emerging Paradigm for Instruction.” Performance Improvement Quarterly 21, no. 2 (2008): 7–36. https://doi.org/10.1002/piq.20020.

author’s contribution

Authors to make an equivalent contribution.

ГЕКСАГОНАЛЬНАЯ СТРАТЕГИЯ В VR-ИГРАХ: ШАХМАТНЫЙ ПОДХОД

Н.А. Авчинников
Учащийся УО «Национальный детский технопарк», учащийся ГУО «Гимназия №3 г. Могилева»

М.С. Заматай
Учащийся УО «Национальный детский технопарк», учащийся ГУО «СШ № 7 г. Волковыска»

М.С. Ильясова
ассистент кафедры инженерной психологии и эргономики БГУИР, магистр

Ф.В. Усенко
инженер-программист кафедры инженерной психологии и эргономики БГУИР, магистр

А.М. Прудник
доцент кафедры инженерной психологии и эргономики БГУИР, кандидат технических наук, доцент

Аннотация. Основной целью данного исследования была разработка и оценка виртуального реального приложения, представляющего собой стратегическую игру на гексагональном поле. Эта игра, напоминающая шахматы, была разработана с целью обогащения жанра стратегий и развития навыков принятия решений у игроков. Игра разворачивается на гексагональном поле, вводя уникальный пространственный элемент в стратегическое планирование. Она включает элементы, характерные для шахмат, включая стратегическое планирование, и концепции обороны и атаки. Игрокам требуется думать наперед, предвидеть ходы противника и разрабатывать стратегии как для обороны, так и для атаки. Эта интерактивная и погружающая среда углубляет стратегический опыт, предоставляя платформу для развития навыков принятия решений у игроков. Процесс разработки включал выбор подходящих инструментов, создание 3D-моделей окружающей среды и персонажей, а также настройку среды разработки для интеграции этих моделей. Были реализованы механики игры, и проект прошел тщательное тестирование. Результаты этого исследования включают успешную разработку уникальной стратегической игры, использующей технологию виртуальной реальности, и оценку ее эффективности в обогащении жанра стратегий и стимулировании навыков принятия решений среди игроков. Достигнутые результаты и реализованные решения были задокументированы в отчете. Эта работа вносит вклад в область компьютерных наук, демонстрируя практическое применение технологии виртуальной реальности в контексте стратегических игр.

Ключевые слова: виртуальная реальность, стратегическая игра, теория игр, стратегическое мышление, управление ресурсами, принятие решений, когнитивное восприятие, интеллектуальные навыки, иммерсивный опыт, динамическая среда, неопределенность, анализ, прогнозирование.