



ADVANCED SOFTWARE ENGINEERING

# PROJECT DESIGN AND REPORT

*Presented by*

**Group – 6**

**Members:**

Rohan Kadam, Aman Thapa Magar, Junkai Ou, Tom Naccarato,  
Jawad Nasar Shah, Mya Lwin, Rimjhim Chakraborty, Veerpal

# Abstract

This report presents the study, design and analysis of a software project targeted at creating a web-based game application that consists of N-Queens problem, Polysphere Problem in 2 Dimension and Polysphere Problem in 3 Dimension. This project is developed as a part of the course module Advanced Software Engineering as a tool to learn software development in large scale IT industries and to understand distinct methodologies involved for the development. The project has been created by the collaborative and determined efforts of the team members of Group-6.

## Acknowledgement

The team members of Group – 6 expresses sincere gratitude to our teacher and module convenor Prof. Hsi-Ming Ho for his constant support and invaluable insights throughout the course of this project. His expertise and guidance has been crucial in achieving success through the journey of the project development.

Furthermore, a heartfelt thank you to all the members of Project Group 6. The dedicated hard work, sheer determination and collective effort of all the team members have played an important role in achieving the goals of developing this project. The commitment to excellence and working together has made this journey fulfilling and has provided a rewarding experience to all. Thank you all for your valuable contributions.

# Contents

Abstract.....	1
Acknowledgement .....	2
INTRODUCTION.....	4
Overview .....	4
Project Tasks .....	5
N-Queens Problem.....	5
Polysphere Problem - 2 Dimension.....	5
Polysphere Problem - 3 Dimension.....	5
DEVELOPMENT.....	6
Development Methodology.....	6
Software Tools .....	6
Implementation .....	7
TESTING.....	9
N-Queens Solver .....	9
Polysphere Pro .....	14
Polysphere Extreme .....	20
CONCLUSION.....	22
Limitations .....	22
Future Improvements .....	23

# INTRODUCTION

## *Overview*

This comprehensive report unfolds an in-depth study and analysis, delving into a purposeful software initiative aimed at the development of web-based game application. The primary objective of this application is to engage users in the interactive exploration and resolution of puzzles, specifically focusing on the N-Queens puzzle, the Polysphere Puzzle in a 2-dimensional grid, and the Polysphere Puzzle in a 3-dimensional pyramid. The central emphasis of this web application development project is to empower users to independently engage with and solve these intricate puzzles.

The N-Queens puzzle, the Polysphere Puzzle in 2-dimensional grid, and the Polysphere Puzzle in 3-dimensional pyramid – these games will be styled as “**N-Queens Solver**”, “**Polysphere Pro**” and “**Polysphere Extreme**” in the application respectively and the names will be used as such interchangeably throughout the report.

This web application is developed to offer users the capability to actively participate in the puzzle-solving process. It is designed to let users navigate through the N-Queens puzzle, strategically placing queens on a chessboard, and tackling the Polysphere Puzzles in both 2D and 3D environments by placing different shapes on the grids. Furthermore, the application is equipped to present solutions to these puzzles upon user request, to help users understand, compare and reevaluate their solutions.

The primary recipients of this report encompass the esteemed professor, referred to as the client, and the dedicated members constituting Project Group 6. The journey of the development of this project serves as an educational tool for an immersive and hands-on learning experience in software development procedures.

## *Project Tasks*

The project consisted of the systematic implementation of the three widely recognized puzzles - N-Queens Problem, the Polysphere Puzzle in a 2D grid, and the Polysphere Puzzle in a 3D pyramid - undertaken in distinct stages. The following sections provide a detailed briefing on each of the puzzles incorporated into the project.

### **N-Queens Problem**

The N-Queens problem requires the strategic placement of N number of Queen chess pieces on an  $N \times N$  chessboard so that no two Queens attack each other. This means that the user must place the Queen pieces in such a way that no two Queens can be placed facing each other in a straight line or diagonally. This classic puzzle is implemented in the project for Chessboard sizes  $4 \times 4$  to  $10 \times 10$  for 4 Queens and 10 Queens respectively.

### **Polysphere Problem - 2 Dimension**

In the 2D Polysphere Problem, the user will be presented with a white grid board and a selection of puzzle pieces, each uniquely fashioned with a specific arrangement of spheres and distinct colours. The challenge lies in the user's ability to strategically place each puzzle piece on the board, ensuring that every shape effectively covers all the white spaces while also preventing any overlap between two puzzle pieces.

### **Polysphere Problem - 3 Dimension**

Similar to the 2D Polysphere Problem, the user 3D Polysphere Problem in will be presented with a selection of puzzle pieces, each uniquely fashioned with a specific arrangement of spheres and distinct colours. However, unlike the 2D Polysphere Problem, here the user will be given a 3 dimensional pyramid board. Here the challenge is that the user must strategically position each puzzle piece on the pyramid without distorting its shape, effectively covering all the white spaces without overlap between two puzzle pieces.

# DEVELOPMENT

## *Development Methodology*

To replicate the authentic experience of a rigorous software development project, the development approach involved the adoption of the Agile methodology. This strategic choice allowed all parties to proactively respond to the evolving needs and preferences of the client by delivering requested features and valuable feedback in a timely manner.

GitHub will be the platform for all aspects of development tracking in collaboration with the client. GitHub provides an efficient environment that can be used to maintain communication, version control, and project management throughout the development process. By using GitHub, it is possible to maintain transparency and collaboration, and also the client has the access to be actively involved in monitoring and providing feedback on the project's progress.

## *Software Tools*

The project utilized following set of tools to support various aspects of development:

- **PyCharm IDE:** This integrated development environment (IDE) served as the primary platform for coding and overall development.
- **Python:** The project was implemented using the Python programming language, which was employed for coding and algorithmic development.
- **HTML, CSS, and JavaScript:** These web technologies were utilized for frontend development and the creation of the puzzle board animation.
- **Three.js and Orbit Controls:** These libraries played a crucial role in implementing the 3D pyramid animation within the project.
- **MySQL:** The MySQL database management system was chosen to store and manage puzzle solutions efficiently.
- **Ubuntu Minimal Linux OS:** The project was hosted on the Ubuntu Minimal Linux operating system, providing a robust and secure environment for the application.

The project used the following Python libraries:

- **Django**: Chosen as the framework of choice, Django provided a robust foundation for building and structuring the application.
- **pip**: This package installer was used in managing and installing the required Python packages.
- **mysql-connector** and **mysqlclient**: These libraries provided the connection to the database server for efficient interaction with MySQL.
- **OCI** and **Oci-Cli**: Utilized for accessing the bucket storage in the Oracle Cloud, these libraries played a pivotal role in integrating cloud-based resources into the project.
- **Pillow**: Employed for image processing related to the polysphere puzzle, Pillow enhanced the project's capabilities in handling graphical elements with efficiency and versatility.

## *Implementation*

The project's implementation unfolded through a series of structured tasks, each contributing to the overall development and refinement of the application. Here's an elaboration on each task:

### Task 1:

In this phase, the project plan was devised to create a primary structure of the project.

### Task 2:

Within this phase, we employed the Backtracking algorithm to tackle the N-Queens Puzzle. To manage computation time complexity and resource-intensive operations, we set a constraint of 10 queens. Recognizing the potential for further enhancement, a roadmap was established for future optimization, proposing an extension of the N-Queens puzzle to 15 queens using the DLX algorithm. This forward-looking initiative was identified as a feature request, paving the way for continued improvements.



### Task 3:

The task focused on addressing the Polysphere puzzle with a strategic shift towards Donald Knuth's Algorithm X, opting for its expedited solution generation—80,445 in total. This approach, which excluded rotation and reflection images for heightened computation efficiency, significantly outpaced the traditional backtracking method. Notably, our DLX approach yielded all solutions within a mere 39 minutes, a stark contrast to the 7 hours required by the backtracking method on medium-graded microprocessors. Furthermore, a comprehensive plan was devised to print and store solutions in object storage in the cloud. This initiative aimed to accelerate user access to solutions during gameplay and content creation, minimizing unnecessary computations and enhancing user experience.

### Task 4:

Building on the success of the DLX algorithm, Task 4 refined the process further. By incorporating optimizations into the cover and uncover functions of the algorithm's logic, we achieved a notable improvement. The DLX algorithm, applied to this task, yielded a total of 26,720 solutions within an impressive 34-minute timeframe—demonstrating a considerable acceleration compared to previous implementations.

### Task 5:

In the final phase, the project underwent consolidation and refinement. Adopting a portal-hub design approach, we integrated all project components seamlessly. User authentication pages were implemented, and game levels were introduced to accommodate players of various ages. The inclusion of team descriptions, along with direct links to the project's source code and technical documentation, not only added transparency but positioned the project as an inspirational resource for those seeking reference and insight. This holistic approach not only ensured a user-friendly and engaging experience but also solidified the project's accessibility and relevance.

# TESTING

Following sections lists the test cases and results for each game to evaluate the software functionality is as expected and meets the specified functional requirements. The goal of this phase is to ensure the functionality and reliability of the implemented code and algorithm fulfilling its intended purpose.

## *N-Queens Solver*

S.No.	Test Case	Expected Result	Results
1	To test the successful launch of N-Queens Server	Successful Launch of Server	Server launched successfully.[Fig-1]
2	To test the display of chess board is in accordance with the size selected.	The Chessboard is displayed when range is between 4-10 size. In case of 'out of range' input by the user, the server displays a pop up indicating incorrect range input. Display of chessboard should be accurate.	The results are as expected.[Fig-1][Fig-2]

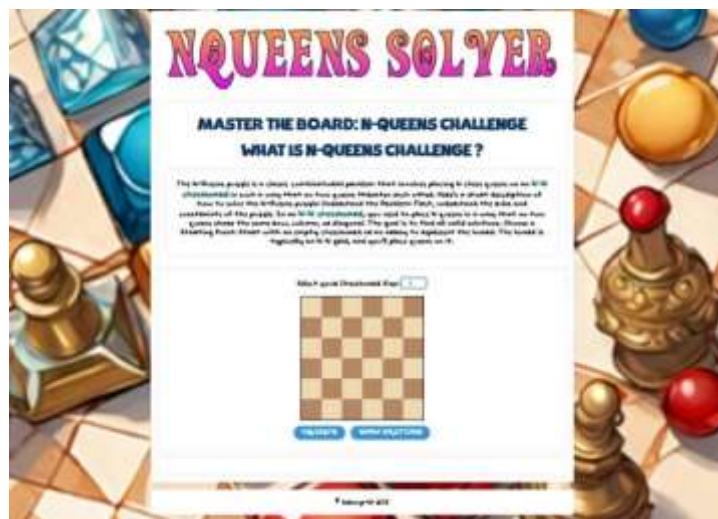


Figure 1



Figure 2

S.No.	Test Case	Expected Result	Results
3	To test the validation of solution presented to user	<b>Case 1:</b> Display pop up to the user if the validation of their solution is correct.	Results are presented in [Fig-3]
		<b>Case 2:</b> Display pop up to the user if the validation of their solution is incorrect.	Results are presented in [Fig-4]



Figure 3



Figure 4

S.No.	Test Case	Expected Result	Results
4	To test placement and removal of Queen Pieces by clicking.	First click on chessboard should place the Queen piece on that position and Second Click on the Queen piece should remove the said piece.	Each expected action is performed successfully. [Fig-5][Fig-6]



Figure 5

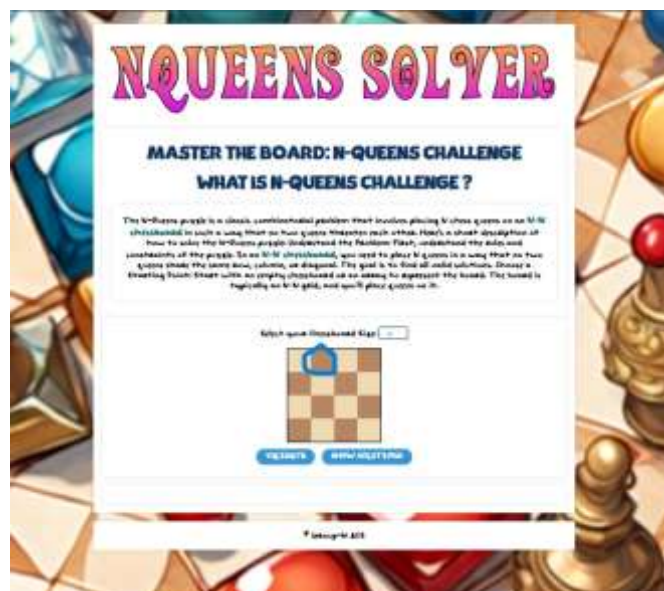


Figure 6

S.No.	Test Case	Expected Result	Results
5	To test the number of Queen pieces placed is not more than the corresponding size of the Chessboard.	To ensure if user attempts placing Queens more than the size of the Chessboard, an error pop up is displayed to the user.	The user is displayed the error pop up message as expected.[Fig-7]



Figure 7

S.No.	Test Case	Expected Result	Results
6	To verify the accuracy of the presented solutions	Accurate solutions are displayed after clicking “Show Solution” button according to the partially filled or empty chessboard and solutions are hidden after clicking “Hide Solution” button.	The presented solutions are accurate as per the rules of the game.[Fig-8][Fig-9]



Figure 8



Figure 9



## Polysphere Pro

S.No.	Test Case	Expected Result	Results
1	To test the successful launch of Polysphere Puzzle -2D Server	Successful Launch of Server	Server launched successfully.[Fig-10]

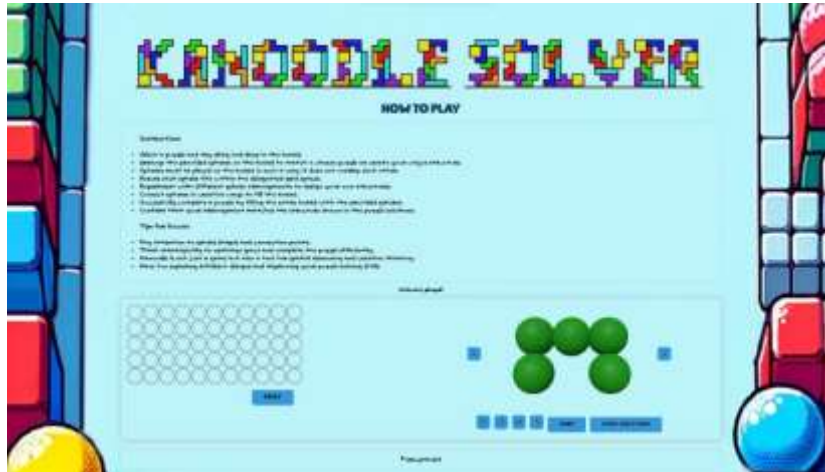


Figure 10

S.No.	Test Case	Expected Result	Results
2	To test the placement of polysphere pieces into the board.	Accurate placement of the pieces by dragging and dropping polysphere pieces onto the board as per user's expectation and removal of the last placed piece by clicking "undo" button.	Dragging and dropping action is as expected [Fig-11][Fig-12] as well as "undo" action. [Fig-13][Fig-14]

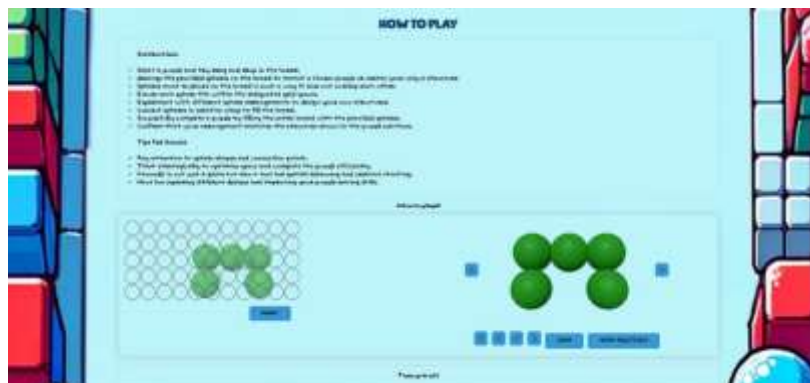


Figure 11

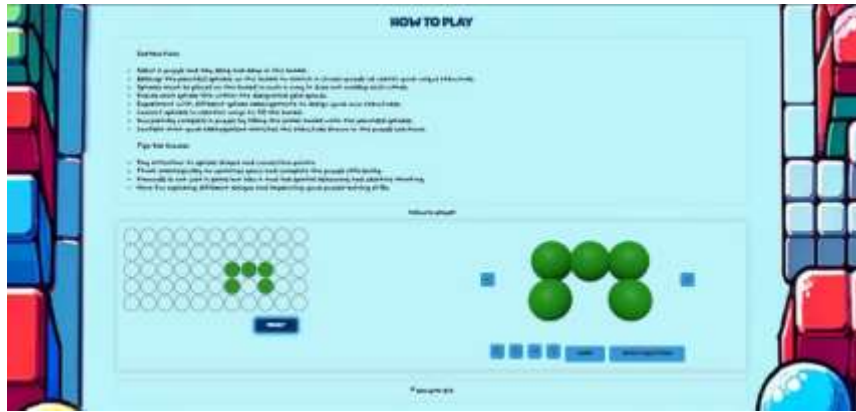


Figure 12

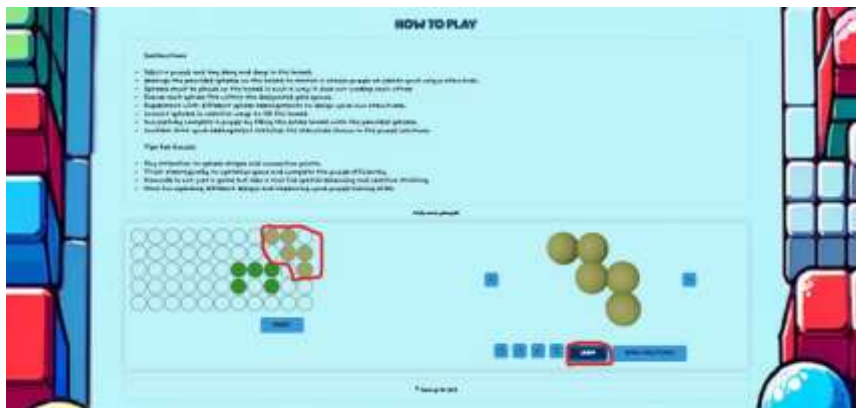


Figure 13

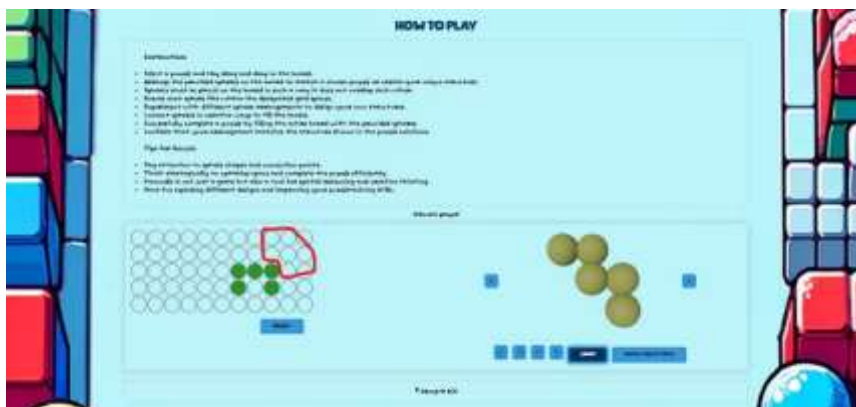


Figure 14



S.No.	Test Case	Expected Result	Results
3	To test the accuracy of rotation and restructuring of the polysphere pieces using the provided buttons.	<p><b>Case 1:</b> Verification of left and right rotate symbol buttons.</p> <p>90 degrees left or right Rotation of the polysphere pieces by click of each respective left or right rotate symbol button.</p>	<p>Button functions are as expected. [Fig-15][Fig-16] [Fig-17]</p>
		<p><b>Case 2:</b> Verification up-down and side-to-side flip symbol buttons.</p> <p>180 degrees' flip of the polysphere pieces by click of up-down or side-to-side flip symbol buttons.</p>	<p>Button functions are as expected. [Fig-18][Fig-19] [Fig-20]</p>

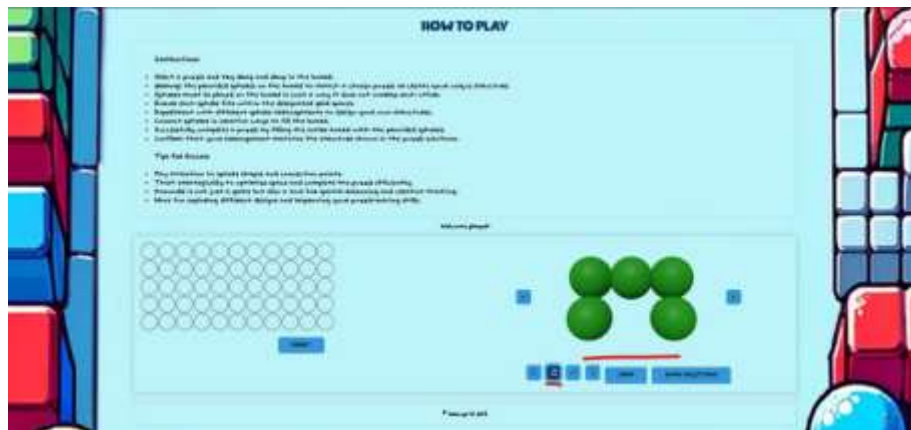


Figure 15

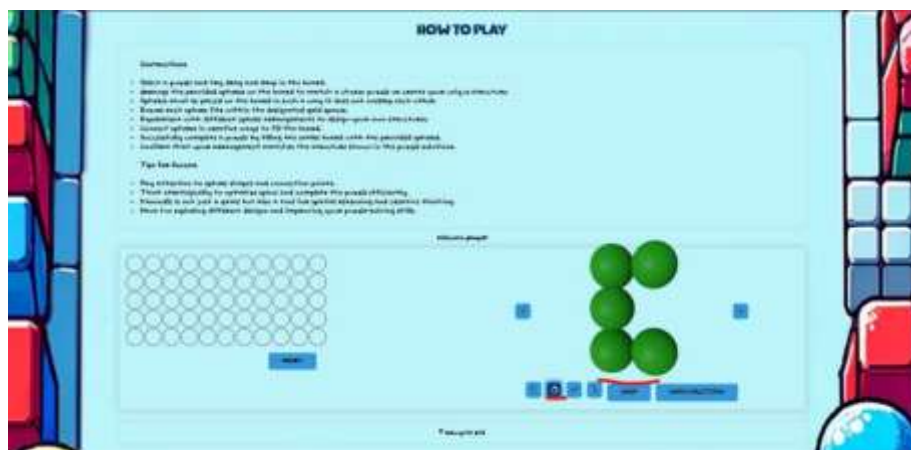


Figure 16

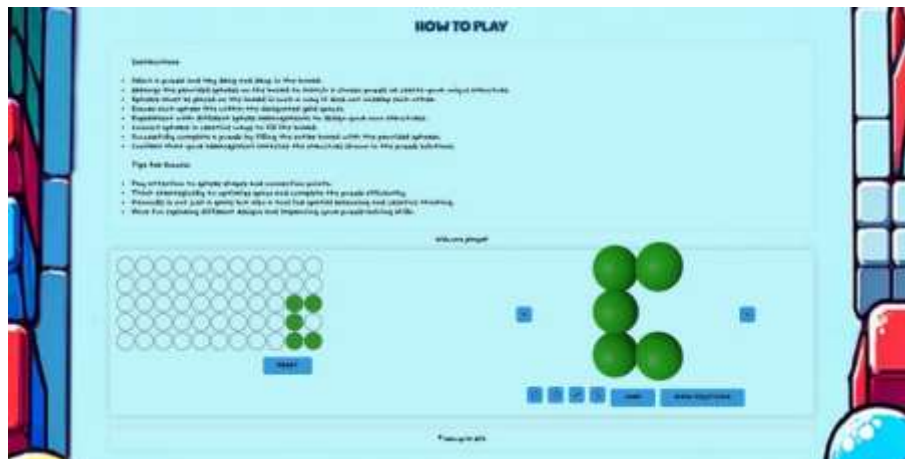


Figure 17

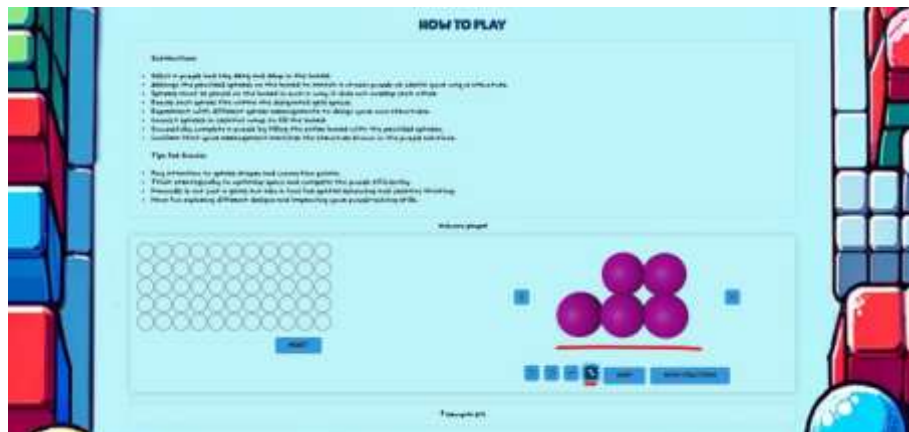


Figure 18

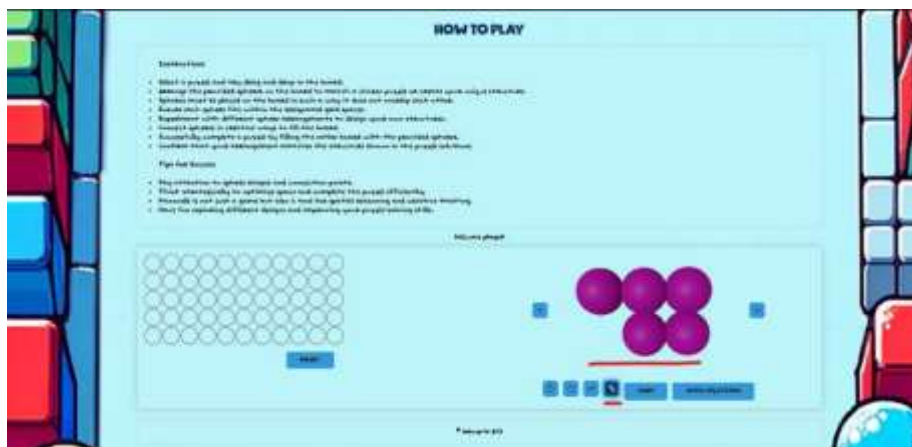


Figure 19

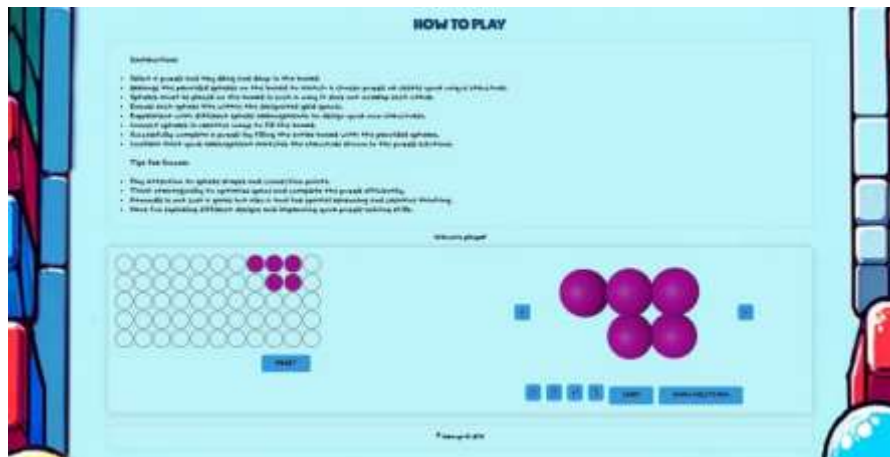


Figure 20

S.No.	Test Case	Expected Result	Results
4	To test the accuracy of the presented solutions.	<p><b>Case 1:</b> To verify functionality of “Show Solution” Button</p> <p>Solutions are displayed after clicking “Show Solution” button and in case of no possible solutions for user’s piece arrangement on board, a message is displayed</p>	Button function is as expected. [Fig-21]
		<p><b>Case 2:</b> To verify the accurate solution is presented.</p>	Accurate solution is presented.[Fig-22]

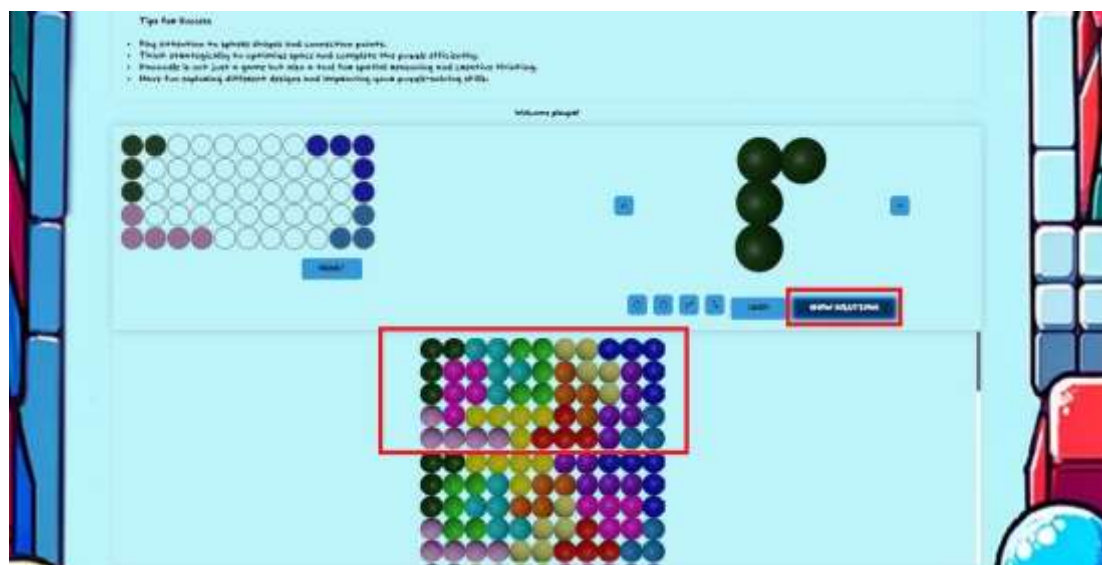


Figure 21



Figure 22

S.No.	Test Case	Expected Result	Results
5	To test the Reset button	The “Reset” button should clear all the pieces from the board.	Button function is as expected. [Fig-23][Fig-24]

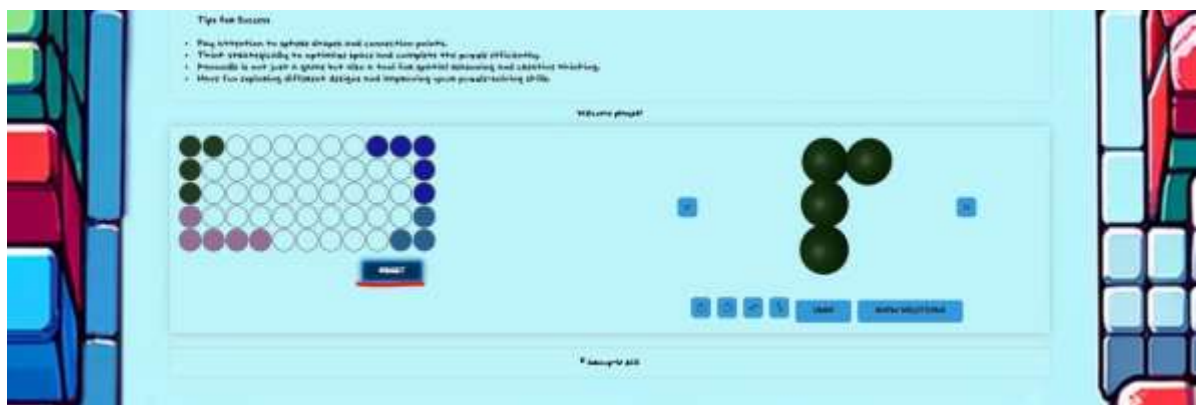


Figure 23

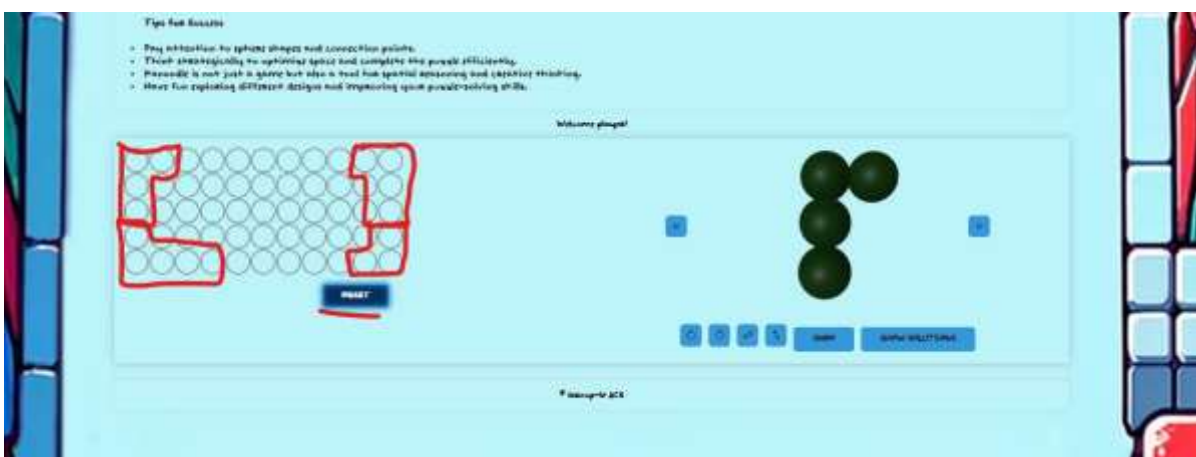


Figure 24

## *Polysphere Extreme*

S.No.	Test Case	Expected Result	Results
1	To test the successful launch of Polyshpere Puzzle -3D Server	Successful Launch of Server	Server launched successfully.[Fig]

S.No.	Test Case	Expected Result	Results
2	to test the 3D rotation of the polyshpere pyramid.	Dragging of the mouse should display the view of the pyramid from 3-dimensional perspective.	Successful implementation of functionality

S.No.	Test Case	Expected Result	Results
3	To verify selection of the shapes	Display of the choices of the shapes by selecting right or left swipe buttons.	Successful implementation of selection functions

S.No.	Test Case	Expected Result	Results
4	To test the colouring of the spheres in the pyramid according to shape selected by the user.	<b>Case 1:</b> Clicking on the spheres should colour it in the shade of the shape selected by the user	Successful colour functionality
		<b>Case 2:</b> Colouring of the spheres should be restricted to the shape selected. The number of spheres in the pyramid that can be coloured should be equal to the number of spheres in the shape displayed as selected by the user.	Successful implementation of restriction.

S.No.	Test Case	Expected Result	Results
5	To test the Solve, Clear and Stop functions.	<b>Case 1:</b> "Solve" button should give the number of solutions as well as visual representation of the solutions on the pyramid for partial and empty configuration.	Successful implementation
		<b>Case 2:</b> "Stop" button will stop the listing of solutions and Clear button should clear all the coloured spheres in the pyramid to reset the game.	Successful implementation

S.No.	Test Case	Expected Result	Results
6	To test the visual aspects of the pyramid.	The layer selection button should display the visual representation of the pyramid according to the number of layers selected in 2-D as well as 3-d layer wise.	Successful implementation

# CONCLUSION

In conclusion, this comprehensive report outlines a purposeful software initiative focused on the development of a web-based game application, emphasizing interactive puzzle resolution. The project's central objective is to empower users to independently engage with and solve intricate puzzles, including the N-Queens puzzle and the Polysphere Puzzles in both 2D and 3D dimensions.

The web application is designed to facilitate active user participation in the puzzle-solving process, allowing navigation through chessboard scenarios, strategic queen placements, and the placement of various shapes on grids for Polysphere Puzzles.

The project is implemented through agile methodology to provide requested features and feedback from the client. Utilized GitHub as the central platform for tracking all development activities in collaboration with the customer.

The project also underwent thorough testing to verify reliability and functionality of the web-based game application.

## *Limitations*

### **N-Queens Puzzle**

- The Chessboard size is limited to 10 Queen pieces due to computation complexities emerging with higher size boards as heavy resources are consumed when computing the solutions.

### **Polyspere Pro**

- While dragging of the shapes, image of the shape while being dragged cannot change according to rotations and flips performed by the user. Although the placement of the shape remains accurate.

### **Polyshere Extreme**

- Placement of the shapes can only be done unidirectional relative to the initial placement of the first sphere.

- Users cannot place the pieces diagonally as the shape of pieces cannot be skewed from its original shape.
- The user cannot drag and drop the pieces on the pyramid. The game relies on the User's understanding of the 3-d structures and visual knowledge of the shapes. The limitation is caused due to heavy reliance on libraries such as Three.js and Orbit Controls that are used to generate the 3-D elements of the Game.

## *Future Improvements*

To improve the puzzle-solving capabilities, the optimization of time complexity and the enhancement of algorithmic efficiency in the computation of solutions for each puzzle will be required in the project. The primary goal is to guarantee the time required for solving puzzles is minimized, and the implemented algorithms show efficiency in delivering accurate solutions. By improving the time complexity, the overall performance of the puzzle-solving mechanisms will be enhanced, providing users with a smoother experience in navigating through complex problems.

Furthermore, enhancement in both the user experience and the visual features of the system can also be implemented. This improvement can elevate the overall usability and aesthetics of the application. Enhancing the user experience requires improving the interface, navigation flow, and overall interaction design for a more intuitive, efficient, and satisfying engagement for the users.