|  |  |  |
| --- | --- | --- |
| **Tester: Adviser** | | |
| **Success criteria** | **Testing results: Met, Not Met, or Partially Met** | **Comments** |
| **Visual design and user’s interaction with the GUI** | | |
| Users can log-in, sign-up, play, and see the output (prompts, movements, and game results) on the GUI | met |  |
| Users are allowed to switch between the scenes (between the login scene and sign-up scene, from login scene to game scene, from game scene to result scene, and quit result scene) and quit the application using buttons  Enter / Sign-up / Back / Next / Quit | met | You need to add user instructions on how to play and add to one of the appendices. This includes keyboard strokes |
| **Multiple accounts can be stored** | | |
| Registration: users can create accounts if they don’t have one using usernames and passwords | met | You need to improve the design of your signup and login page |
| Authorization: Only users with an account are allowed to play the game (the account is used for ranking), to make sure that the ranking is correct and to prevent cheating | met | Recommendation is to incorporate the use of email account to register |
| Updating: Each account will have a playing history in the database, where wins and losses can be updated after each round played | met | Needs to be shown in the video, Does this include a timestamp of games played? |
| Sorting: A rank of the top five players could be shown in the result page | met | Improve the GUI display for this information |
| **A 3D shape of the cube** | | |
| Mouse and keyboard action listeners: keyboard typing and mouse motion, dragging, scrolling, and clicking; all the cells can be selected | met | This would be good if the game can be played in each individual player’s screen even over LAN |
| **User-friendly** | | |
| Multiple ways to turn and zoom the cube | met |  |
| The indication of the selected cell of cubelet is clear (who took this cell)  A See-through diagram to help users to understand the rule is provided, which can be a 2D representation of the three layers, and the actions on the cube are also shown on the diagram | met | See-through is really seeing through the whole cube |
| **Checking algorithm** | | |
| A winner should be identified when three cells in a line are occupied by the same person.  The three cells in a line could be horizontal, vertical, and diagonal in the same layer or across the layers | met | Top 5 players display |

|  |  |  |
| --- | --- | --- |
| **Test Plan** | | |
|  | Action to be tested | Testing method and Expected output |
| 1 | Open the application and display the first scene | Click run file in Netbeans - a window is shown in the middle of the desktop; all GUI components are displayed in the correct place with the correct size |
| 2 | Functions of the GUI components in the login scene | Type texts into the textFields and passwordFields - Text can be typed into the fields. The text in the textFields will be shown, but the text in the passwordFields will be hidden as ••••••  Click “enter” and “sign-up” buttons - The enter button and sign-up button can be clicked with the intended functions |
| 3 | Login features in the login scene (for both player1 and player2) | Enter pre-registered user account - the textFields and “enter” button will be hidden, an image of correct is displayed  Enter incorrect username/password or unregistered username - the texts in the fields are deleted, a label shows that the username or the password is wrong |
| 4 | Sign-up a new account | Register with a pre-registered user’s username - the text in the passwordFields will be deleted; a label will show that the username already exists  Enter different text in the password and confirm password fields - the text in the passwordFields will be deleted; a label will show that the two passwords are not the same  Register with a new username (the password and confirmed password are the same) and view the database table - after clicking “enter” button, the login scene is shown again, and a new row is added into the database table with the correct username and password (other information are 0)  Click back - back to the login scene |
| 5 | Turning and scrolling the cubic board | Drag the cube using a mouse - the cube will turn to the direction where the mouse moves to  Scrolling the mouse - the cube will move closer or further from the screen (bigger or smaller) in a range  Drag the cube using WASD on the keyboard - the cube will turn to the intended direction |
| 6 | Clicking the cells | Bothe the players click the cells - after one player double-clicks the cell, the cell will become blue, another player is red |
| 7 | The 2D representation of the cubic board | Place the mouse on the cube and move the mouse: the selected cell will be highlighted; the related cell on the 2D representation will be highlighted |
| 8 | Select the middle cell | Player 1 click the “click the middle cell” button: on the 2D representation, cell 14 change to red  Player 2 click the button: cell 14 change to blue |
| 9 | Winning condition | The first player connects his/her three cells in a horizontal line (number 1 2 3): a button is shown, on which it displays the username of player 1  Vertically (cell 1 10 19): see above  Diagonally (cell 1 11 21): see above  Diagonally in different layers (cell 1 14 27): see above |
| 10 | The result scene | Click the “go to result page” button: A table with three rows and three columns is displayed. Player 1’s “wins” cell is added by one, Player 2’s “losses” cell is added by one. |
| 11 | Quit the application | Click the “quit” button: the window is closed |