

Tic-tac-toe App

Sayuri Karunanayake

Content

01. Description
02. Flowchart
03. Use case diagram
04. Test cases
05. Selenium automation testing results
06. Screenshots of UI

01. Description

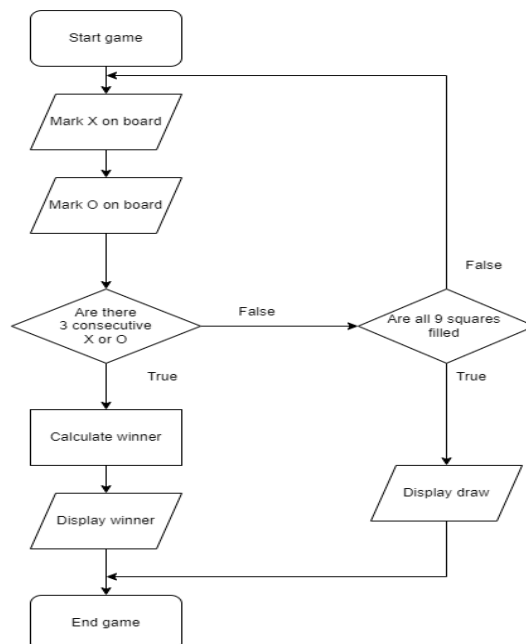
Technology: react.js

This is a simple web application developed using react.js to play the tic-tac-toe game. Two players can play in the game as X and O. Players can start the game by clicking on “Start game” button and go back to any previous moves by undoing the moves they made. Players can see the game’s final result at the top right near the board. As well as the game can be restarted at any point by simply clicking on “Start game” button. Players can see who makes the next move at the top right. After the game is over other remaining empty boxes cannot be marked and once a box is marked it can not be clicked again unless you undo to that move. There can be 3 statuses at the end of the game those are,

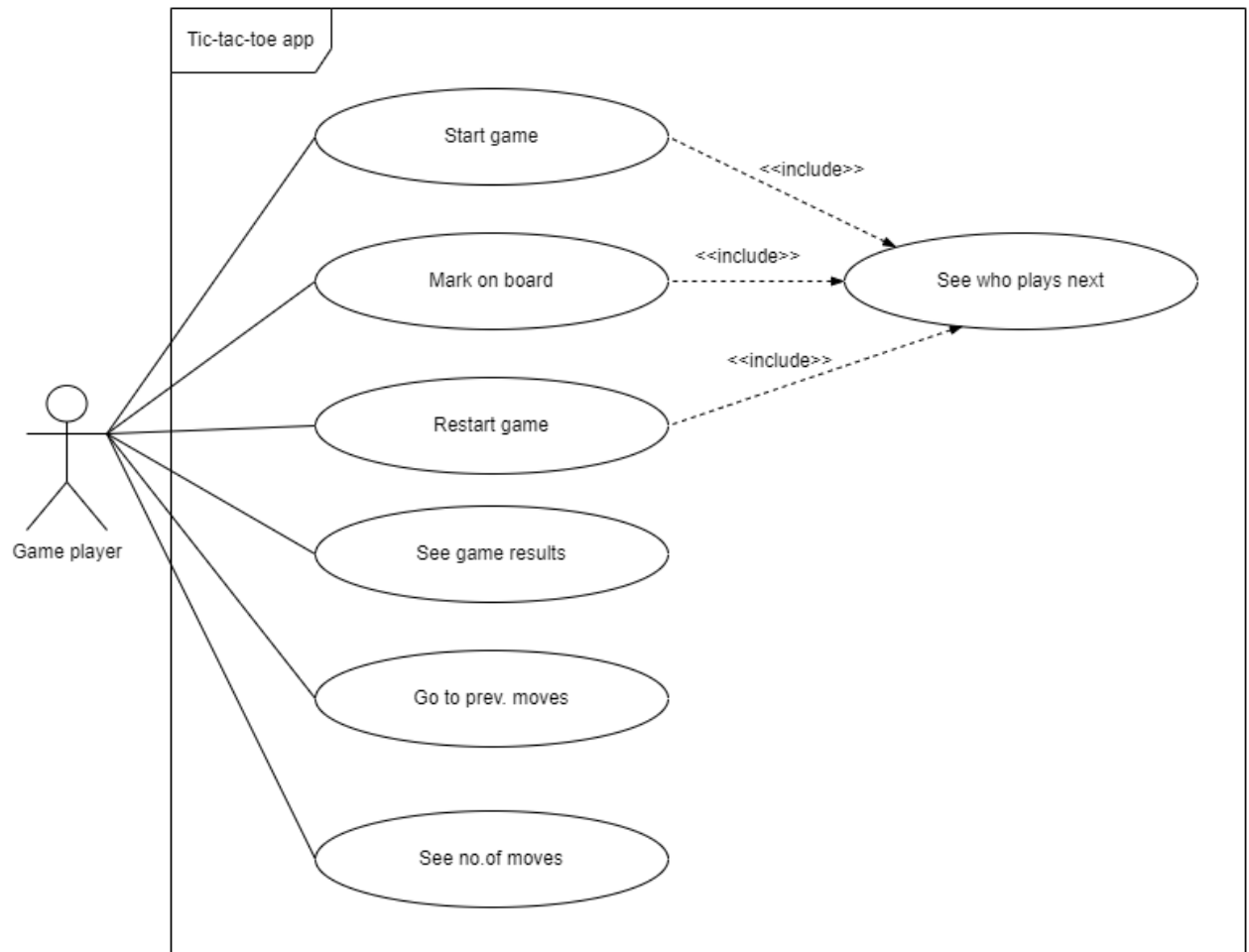
- X is the winner
- O is the winner
- Draw

Link to visit Tic-tac-toe app : <https://sayuriKarunanayake.github.io/Tic-tac-toe-app/>

02. Flowchart



03. Use case diagram



04. Test cases

| Test case ID | Description | Expected result | Status |
|--------------|---|--|---------|
| 1 | X makes moves in order to align 3 X s in the grid | X wins | Success |
| 2 | O makes moves in order to align 3 O s in the grid | O wins | Success |
| 3 | Both players, X and O make moves without aligning 3 X s or 3 O s | Draw | Success |
| 4 | The game draws so players want to restart the game | Game restart when "Start game" clicked | Success |
| 5 | One player makes a wrong move mistakenly so wants to go back to the previous move | Go back to the previous move when that move is clicked | Success |

05. Selenium automation testing results

Test case 1

The screenshot displays the Selenium IDE interface for 'Tic-tac-toe app testing'. The 'Tests' pane on the left shows 'Test case 1' as the selected test. The main pane shows a table of commands and their targets for 'http://localhost:3000'.

| Command | Target | Value |
|----------------------|---|-------|
| 1. ✓ open | / | |
| 2. ✓ set window size | 835x824 | |
| 3. ✓ click | css=div:nth-child(1) > btn:nth-child(1) | |
| 4. ✓ click | css=div:nth-child(3) > btn:nth-child(2) | |
| 5. ✓ click | css=div:nth-child(2) > btn:nth-child(2) | |
| 6. ✓ click | css=div:nth-child(1) > btn:nth-child(3) | |
| 7. ✓ click | css=div:nth-child(3) > btn:nth-child(3) | |

Below the table, the 'Command' field is set to 'open', 'Target' to '/', and 'Value' is empty. The 'Log' pane at the bottom shows the execution of 'Test case 1' with a timestamp of 21:38:28, indicating it completed successfully.

Test case 2

The screenshot displays the Selenium IDE interface for 'Tic-tac-toe app testing'. The 'Tests' pane on the left shows 'Test case 2' as the selected test. The main pane shows a table of commands and their targets for 'http://localhost:3000'.

| Command | Target | Value |
|-----------------|---|-------|
| 4. ✓ click | css=div:nth-child(1) > btn:nth-child(1) | |
| 5. ✓ click | css=div:nth-child(2) > btn:nth-child(3) | |
| 6. ✓ click | css=div:nth-child(2) > btn:nth-child(2) | |
| 7. ✓ click | css=div:nth-child(1) > btn:nth-child(2) | |
| 8. ✓ mouse over | css=div:nth-child(1) > btn:nth-child(2) | |
| 9. ✓ mouse out | css=div:nth-child(1) > btn:nth-child(2) | |
| 10. ✓ click | css=div:nth-child(3) > btn:nth-child(1) | |

Below the table, the 'Command' field is empty, 'Target' is empty, and 'Value' is empty. The 'Log' pane at the bottom shows the execution of 'Test case 2' with a timestamp of 21:39:10, indicating it completed successfully.

Test case 3

Selenium IDE - Tic-tac-toe app testing

Project: Tic-tac-toe app testing

Tests +

Search tests...

Test case 1 ✓

Test case 2 ✓

Test case 3 ✓

Test case 4

Test case 5

| Command | Target | Value |
|---------|---|-------|
| ✓ click | css=div:nth-child(1) > btn:nth-child(1) | |
| ✓ click | css=div:nth-child(1) > btn:nth-child(2) | |
| ✓ click | css=div:nth-child(3) > btn:nth-child(2) | |
| ✓ click | css=div:nth-child(3) > btn:nth-child(3) | |
| ✓ click | css=div:nth-child(2) > btn:nth-child(3) | |
| ✓ click | css=div:nth-child(2) > btn:nth-child(1) | |
| ✓ click | css=div:nth-child(3) > btn:nth-child(1) | |

Command

Target

Value

Description

Log Reference

19. mouseOver on css=div:nth-child(1) > btn:nth-child(1) OK 21:39:44

6. mouseOut on css=disabled:nth-child(2) OK 21:39:44

7. click on css=div:nth-child(1) > btn:nth-child(3) OK 21:39:44

8. click on css=div:nth-child(1) > btn:nth-child(2) OK 21:39:44

9. click on css=div:nth-child(3) > btn:nth-child(2) OK 21:39:44

10. click on css=div:nth-child(3) > btn:nth-child(3) OK 21:39:44

11. click on css=div:nth-child(2) > btn:nth-child(3) OK 21:39:44

12. click on css=div:nth-child(2) > btn:nth-child(1) OK 21:39:44

13. click on css=div:nth-child(3) > btn:nth-child(1) OK 21:39:45

Test case 3 completed successfully 21:39:45

Test case 4

Selenium IDE - Tic-tac-toe app testing

Project: Tic-tac-toe app testing

Tests +

Search tests...

Test case 1 ✓

Test case 2 ✓

Test case 3 ✓

Test case 4 ✓

Test case 5

| Command | Target | Value |
|--------------|---|-------|
| ✓ click | css=div:nth-child(1) > btn:nth-child(1) | |
| ✓ mouse over | css=div:nth-child(1) > btn:nth-child(1) | |
| ✓ mouse out | css=disabled:nth-child(1) | |
| ✓ click | css=div:nth-child(1) > btn:nth-child(3) | |
| ✓ mouse over | css=div:nth-child(1) > btn:nth-child(3) | |
| ✓ mouse out | css=disabled:nth-child(3) | |
| ✓ click | css=div:nth-child(3) > btn:nth-child(3) | |

Command

Target

Value

Description

Log Reference

19. mouseOver on css=div:nth-child(1) > btn:nth-child(1) OK 21:40:12

20. mouseOut on css=div:nth-child(1) > disabled OK 21:40:15

21. click on css=div:nth-child(1) > btn:nth-child(1) OK 21:40:15

22. mouseOver on css=div:nth-child(1) > btn:nth-child(1) OK 21:40:15

23. mouseOut on css=disabled:nth-child(1) OK 21:40:15

24. click on css=div:nth-child(1) > btn:nth-child(3) OK 21:40:15

25. mouseOver on css=div:nth-child(1) > btn:nth-child(3) OK 21:40:16

26. mouseOut on css=disabled:nth-child(3) OK 21:40:16

27. click on css=div:nth-child(3) > btn:nth-child(3) OK 21:40:16

Test case 4 completed successfully 21:40:16

Test case 5

Selenium IDE - Tic-tac-toe app testing

Project: Tic-tac-toe app testing

Tests +

Search tests...

Test case 1 ✓

Test case 2 ✓

Test case 3 ✓

Test case 4 ✓

Test case 5 ✓

| Command | Target | Value |
|-----------------|--|-------|
| 27 ✓ click | css=div:nth-child(3) > .btn:nth-child(1) | |
| 30 ✓ mouse over | css=div:nth-child(3) > .btn:nth-child(2) | |
| 31 ✓ mouse out | css=div:nth-child(3) > .btn:nth-child(2) | |
| 32 ✓ click | css=div:nth-child(3) > .btn:nth-child(3) | |
| 33 ✓ click | css=li:nth-child(9) > button | |
| 34 ✓ click | css=li:nth-child(8) > button | |
| 35 ✓ click | css=li:nth-child(7) > button | |

Command

Target

Value

Description

Log

Reference

27. mouseOver on css=div:nth-child(3) > .btn:nth-child(1) OK 21:40:50

28. mouseOut on css=div:nth-child(3) > .btn:nth-child(2) OK 21:40:50

29. click on css=div:nth-child(3) > .btn:nth-child(2) OK 21:40:50

30. mouseOver on css=div:nth-child(3) > .btn:nth-child(2) OK 21:40:50

31. mouseOut on css=div:nth-child(3) > .btn:nth-child(2) OK 21:40:50

32. click on css=div:nth-child(3) > .btn:nth-child(3) OK 21:40:51

33. click on css=li:nth-child(9) > button OK 21:40:51

34. click on css=li:nth-child(8) > button OK 21:40:51

35. click on css=li:nth-child(7) > button OK 21:40:51

Test case 5 completed successfully 21:40:51

06. Screenshots of UI

