Design Document

Title:

Ultimate Tic-Tac-Toe

Participants:

Jose Alejandro Sarmiento and Ksenia Myakisheva.

Overview

The Ultimate Tic-Tac-Toe app offers single-player mode against AI and local multiplayer mode for two players on the same device. It is compatible with both iOS and Android devices, as it is built using the Expo framework in React Native.

Our objective is to enhance the user's gaming experience through an intuitive and visually appealing UI. This includes color personalisation, display of in-game information and the ability to replay previous games.

Additionally, interactive tutorials will be provided to assist users in understanding game mechanics.

Context

Our game addresses the challenges associated with playing Tic-Tac-Toe on paper.

Tic-Tac-Toe's popularity makes it a widely recognized game, but playing it on paper can lead to issues such as losing track of available moves. Our game addresses this problem by illuminating the region where the player can play next, thus keeping the user informed of the progress of the game.

Additionally, our game includes an AI mode, allowing users to enjoy the game even when they don't have someone to play with. This makes it possible for our users to play whenever and wherever they want.

Another difficulty is learning how to play the game. If our users don’t have anyone to explain it to them or they don’t have access to the Internet, we offer the rules inside the app.

If, however, the user is not in the mood for reading, our app also offers an interactive tutorial.

It also adds the possibility to watch previous games, something impossible without a digital solution. This feature can be seen in other famous game platforms such as Chess.com.

Lastly, our customizable user interface is very appealing, and the user can play as many times as they want, since they won’t be running out of paper. This also makes our game environmentally friendly.

Goals

- ☑ Make the game easy and fun to play.
- ☑ Solve common problems like forgetting where to move next.
- ☑ Offer different ways to play (multiplayer and single player modes.
- ☑ Ability to watch and store past games.
- ☑ Customizable color options for personal preference or accessibility needs, such as accommodating color blindness.
- ☑ Teach users how to play the game.

Non-Goals

- ☐ Online Multiplayer Mode.
- ☐ Making IA too smart.
- ☐ Any feature that requires ejecting from Expo.

- ☐ Avatar/ Picture personalisation

Milestones

Start Date: January 3, 2024

Milestone 1 - Implement Game Screen: Develop a working version of Ultimate Tic-Tac-Toe supporting single-player and local multiplayer modes: January 15, 2024

Milestone 2 - Implement Statistics Screen: Create a system for storing and reviewing previous games.: January 28, 2024

Milestone 3 - Implement Settings Screen: Integrate functionality for users to customize app themes and colors within the application: February 2, 2024

Milestone 4 - Develop Tutorial and Rules Screens: Design screens to provide users with explanations and instructions for playing the game: February 5, 2024

Milestone 5 - Implement Adaptive Layouts and Integrate External Services: integrate Firebase and Sentry: February 7, 2024

Milestone 6 - Conduct Testing : February 8, 2024

End Date: February 8th, 2024

User Stories

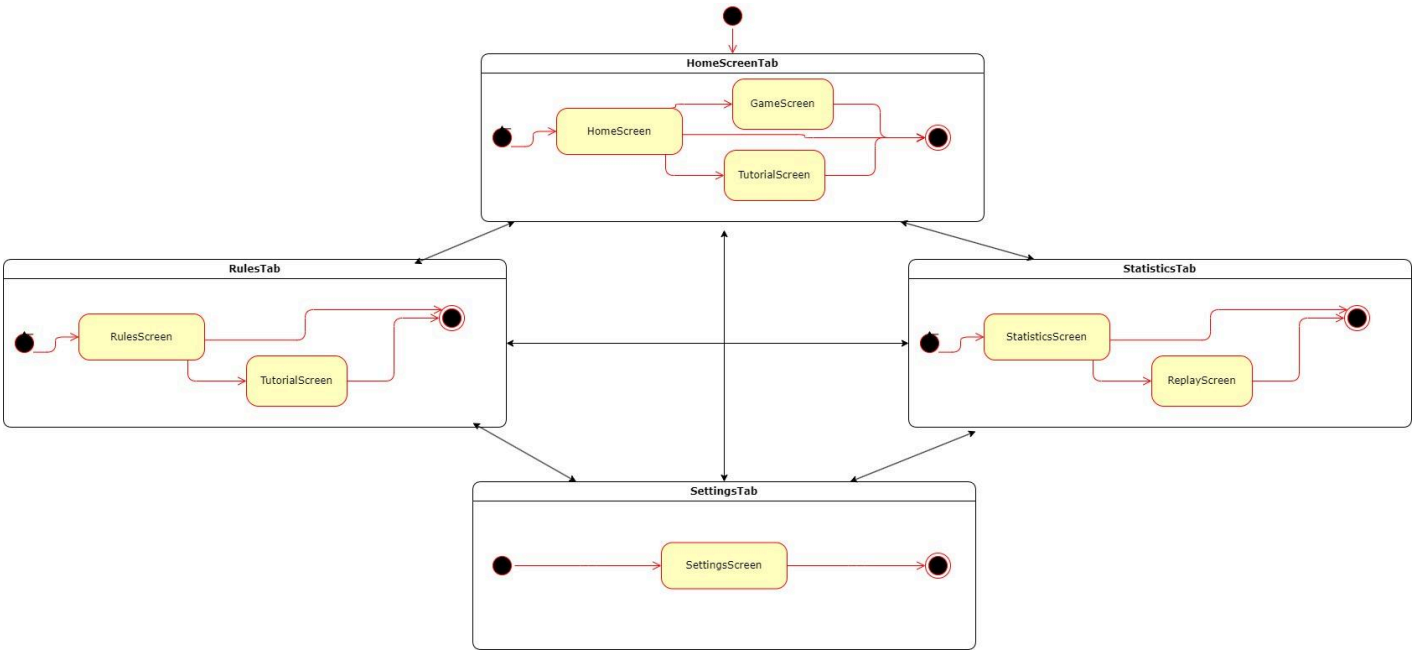
US1 - As a user, I want to create an Ultimate Tic Tac Toe game against my friends locally so that we can play whenever we are together.
US2 - As a user, I want to create an Ultimate Tic Tac Toe game against an AI so that I can play alone whenever I want.
US3 - As a user, I want to have a tutorial available so that I can learn how to play.
US4 - As a user, I want to have the rules of the game available so that I can consult them if I forget something.
US5 - As a user, I want to continue a game I didn't finish earlier so that I can finish it.
US6 - As a user, I want to see the number of moves a game took so that I know if it was a fast or slow game.
US7 - As a user, I want to see how many victories the player playing X has and how many the player playing O has so that I can get an idea who has an advantage.
US8 - As a user, I want to see the date of my finished games so that I can know when it took place
US9 - As a user, I want to replay finished games so that I can see all the movements in order of said game
US10 - As a user, I want to customize the colors of the X and O symbols so that I can make the application look however I want.
US11 - As a user, I want to change the theme of the application so that it matches the one of my device or not.
US12 - As a user, I want to login so that I can access the cloud storage functionalities.
US13 - As a user, I want to save my finished games to the cloud so that I can either delete them from my local storage or save them on another device.
US14 - As a user, I want to sync my cloud games with my device games so that I can have all of them together.
US15 - As a user, I want to delete my games locally so that I have a cleaner history or to save space.

US16 - As a user, I want to delete my cloud games so that I have a cleaner history on the cloud.
US17 - As a user, I want to logout so that I can log in the application with another account.

Proposed Solution

The application is built on React Native Expo, which handles platform specific operations. It communicates with Firebase for authentication and storage and with Sentry for error alerting and monitoring. The application has 6 main screens in which all functionalities are handled. These are Home, Gameplay (and Tutorial), Rules, Statistics and Settings.

The following is a FlowChart of the possible navigations between screens:

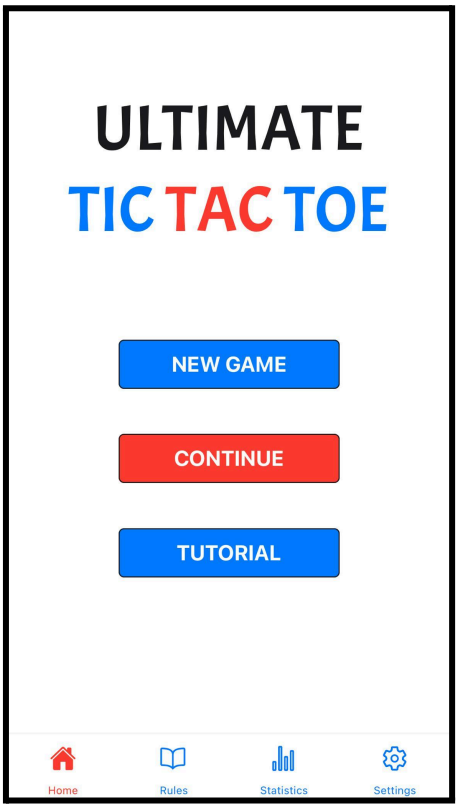


Resulting Solution

A detailed description of user experience is presented below.

1. **HomeScreen**

In the existing version of the app, users launch the application and are presented with the Home Screen. The menu screen is one of the four tabs that the user can reach through the tab bar. The tab bar is located at the bottom of the screen. The Home Screen contains a menu consisting of three buttons, as seen below:

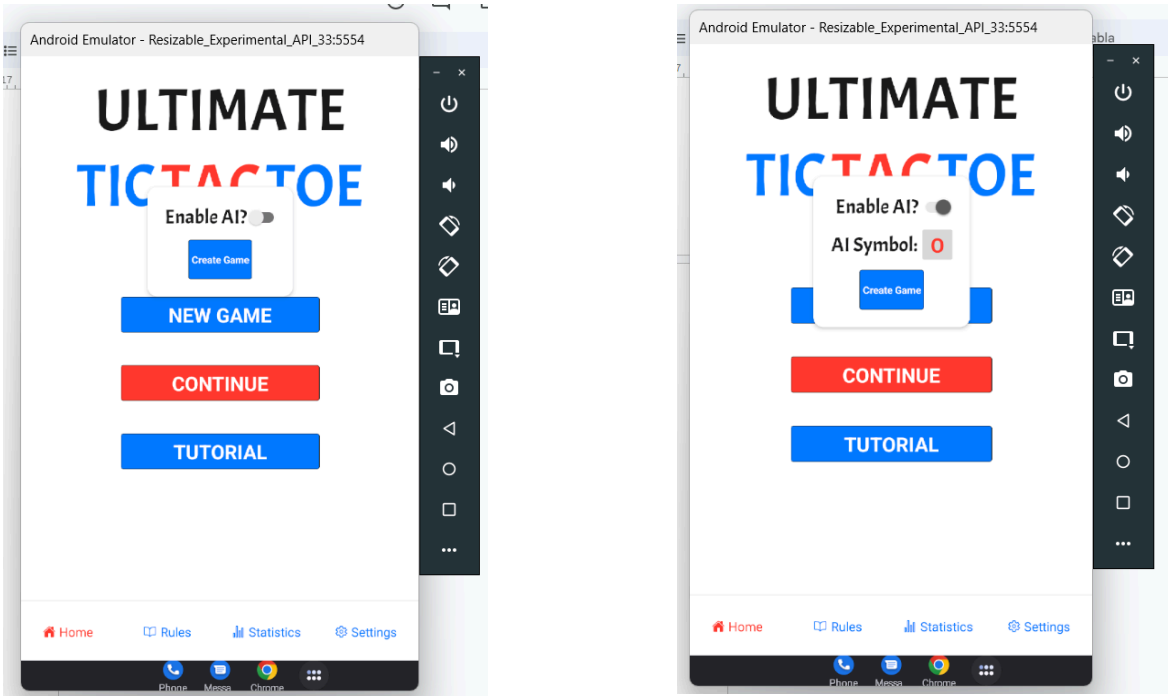


Home Screen, Iphone 8 plus

The user gets a choice between starting a new game, resuming a game he previously started but not finished or accessing the tutorial.

- **Start a new game**

When the user tries to start a new game, a pop-up appears. Here the user can choose between creating a game in single-player mode, by enabling the IA, or two-player mode. If IA is enabled, the user may choose which symbol he wants the IA to have. X is always the starting symbol.



Home Screen, Creating a Game
Android EmulatorResizable Experimental API 33, Tablet Mode

After pressing on the create game button, the user is taken to a newly generated Game Screen.

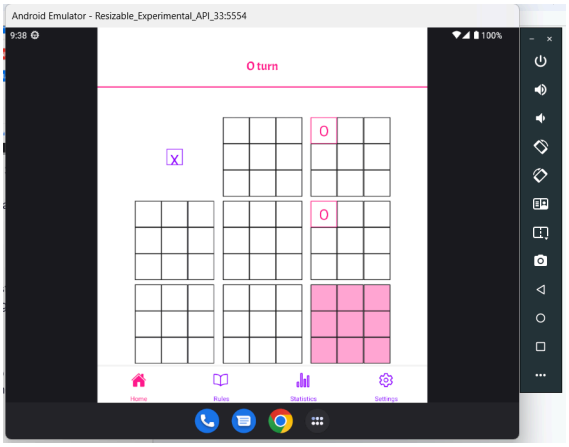
- **Resume a game**

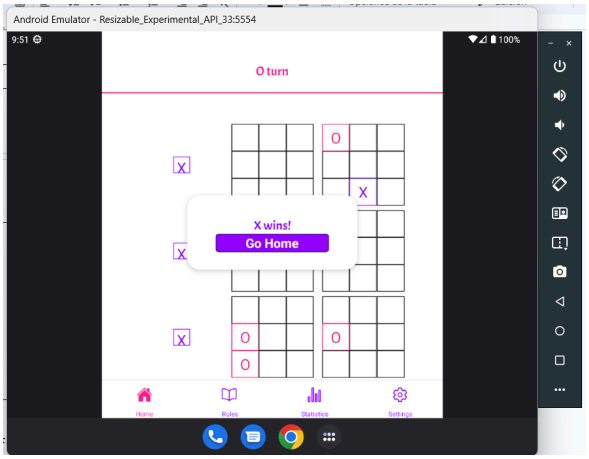
User is taken to a Game Screen with data about a past game. If no past game were to be present, it generates an empty game.

- **Start Tutorial**

User is taken to a Tutorial Screen, which is identical to a Game screen, except for the addition of instructions explaining the game.

2. Game Screen





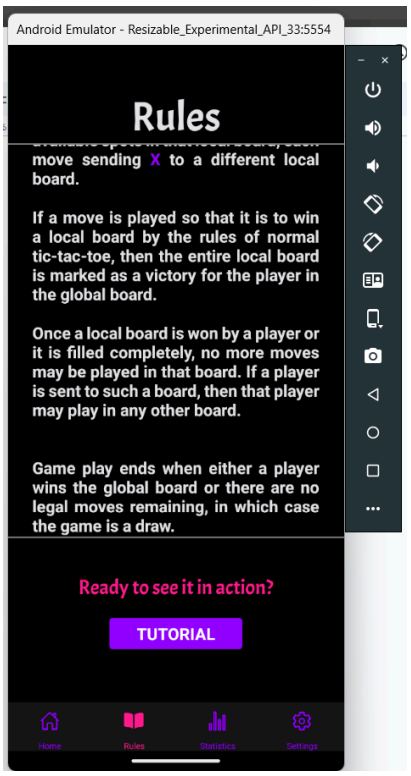
Android EmulatorResizable Experimental API 33, FoldableMode
Game Screen (Color Preferences Changed)

Game Screen provides the following information:

- Whose turn is it
- Where can they move next
- Who won on each board
- The colors of each opponent
- Displays the winner

3. Rules Screen

By going to the second element of the tab, the user can access the Rules screen. This screen contains explanations about how to play the game, through which the user can scroll, and gives direct access to the tutorial.

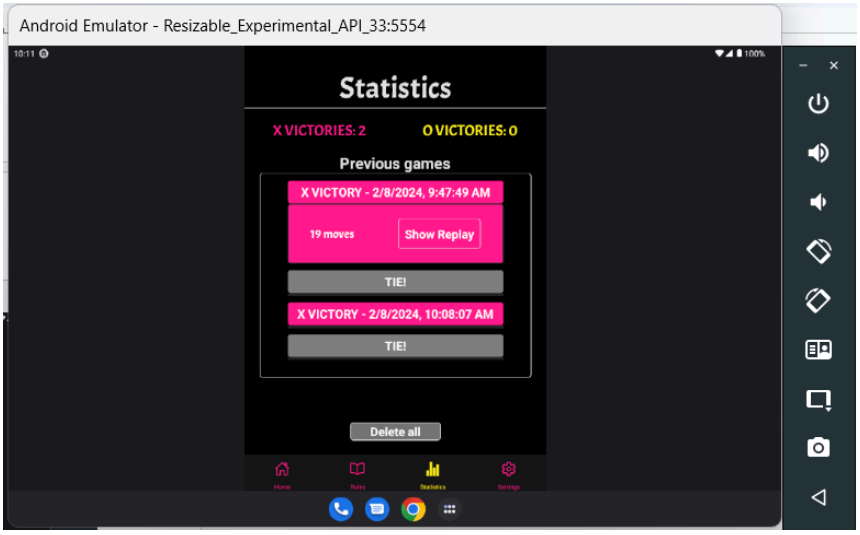


Rules Screen, Android EmulatorResizable Experimental API 33, Phone Mode (Dark Mode)

4. Statistics Screen

One can access the Statistics Screen by pressing the third element of the tab. This screen contains the number of wins and loses of each symbol.

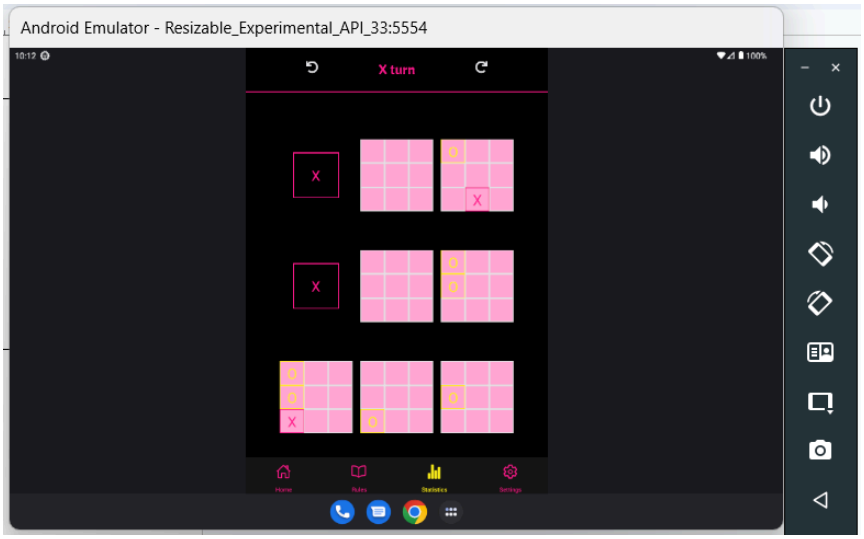
It also contains a list of all the games played, which can be expanded to see the number of moves and the replay button, which takes the user to the Replay screen . This list can also be deleted with the Delete All Button.



Statistics Screen, Android EmulatorResizable Experimental API 33, Tablet Mode (Turned)

5. Replay Screen

This screen can be accessed from the Statistics Screen. It shows the user a replay of a previous game. The user can go backwards or forward with the two arrows. The replay starts at the end of the game.

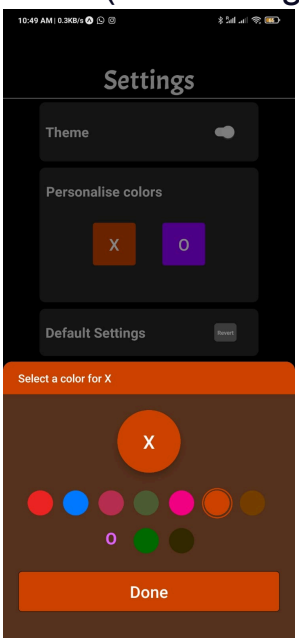


Replay Screen, Android EmulatorResizable Experimental API 33, Tablet Mode (Turned)

6. Settings Screen

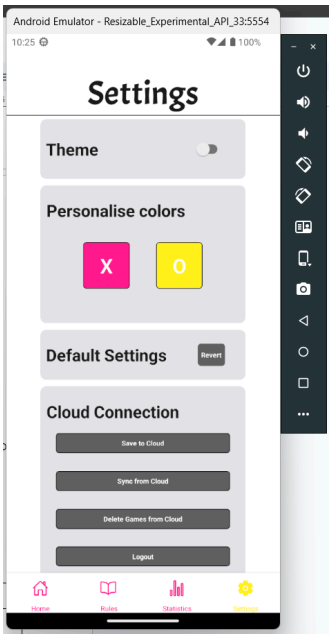
This screen has the following functionalities (listed in order):

- Change Theme (Light/Dark Mode)
- Customize X and O colors (also changes some custom buttons)



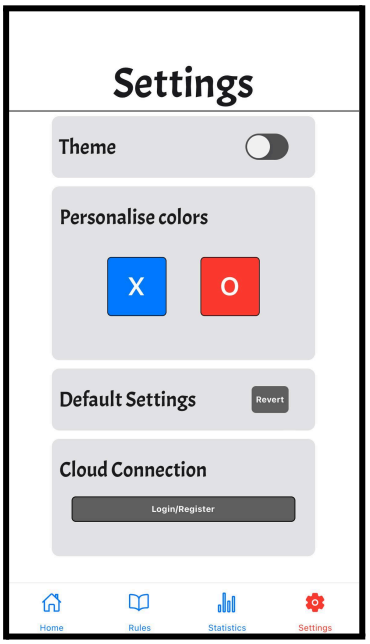
Settings Screen, Redmi Note 8 Pro, color selection sheet

- Revert to the default theme and colors of the app
- A logged-in user may save, retrieve and delete his previously played games from CLOUD.

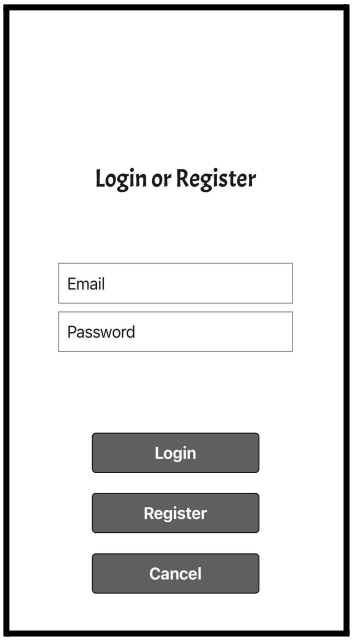


Settings Screen, Android EmulatorResizable Experimental API 33, Phone Mode (authenticated user)

A non-identified user can login or register an account with a button visible only for non-authenticated users. When the user presses the button, he can login/register with the following ‘pop-up’ (Modal in react terms):



Settings Screen, Iphone 8 plus (non-identified user)

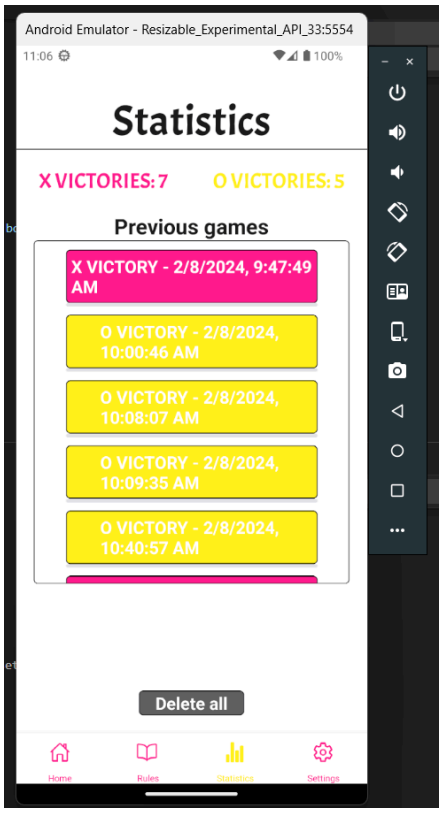


Authentication Modal, Iphone 8 plus

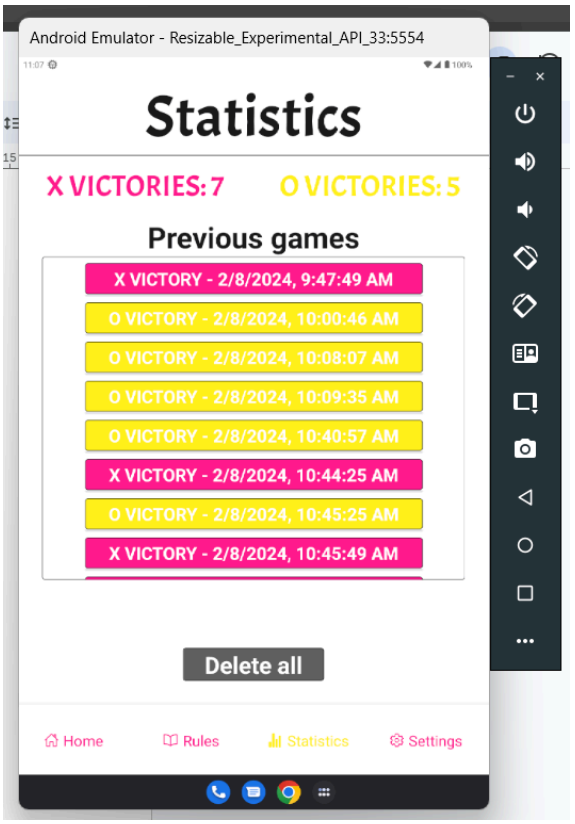
- **Adaptive Layouts**

As shown above, our app layout adapts to different types of screens and operating systems. This is possible due to a reduced use of fixed dimensions. Additionally, a special case was made for screens whose width exceeds 750 pixels (thought for tablets). In this case font sizes, and in some cases dimensions, get bigger (usually doubling). Another example of this is shown below:

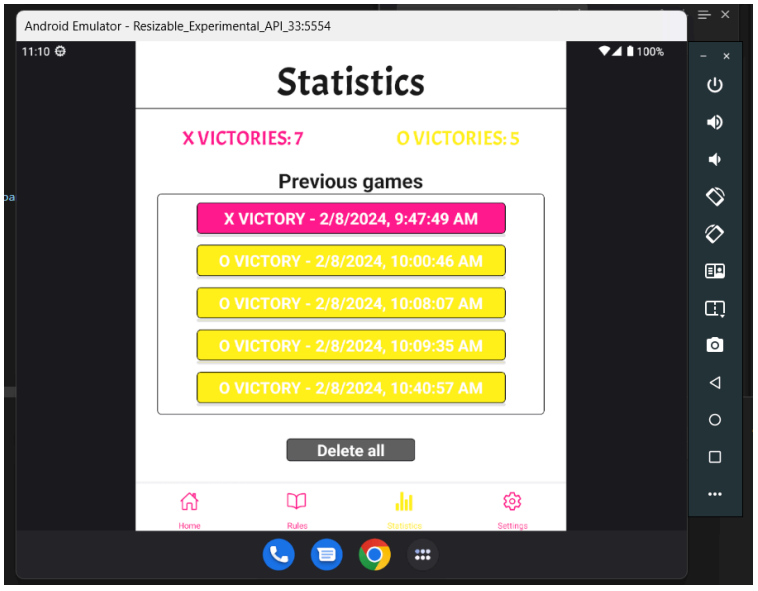
Comparison between Statistic Screens the three modes of Android EmulatorResizable Experimental API 33



Phone Mode



Tablet Mode

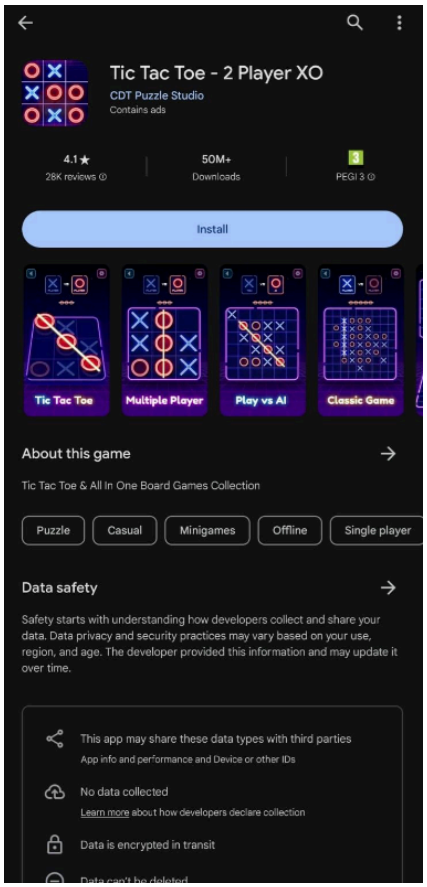


Foldable Phone Mode

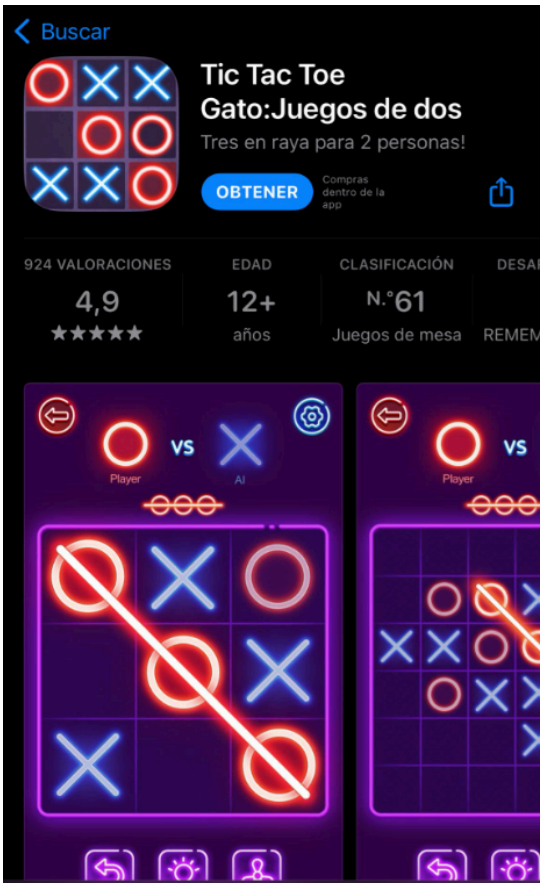
Alternative Solutions

There are many other Tic-Tac-Toe apps available on the market where users can choose between single-player and multiplayer modes.

However, the majority of them only offer the basic Tic-Tac-Toe game. Because of this, the most relevant ones are the following:



ANDROID



IOS

The main advantages of the first game are the different level difficulties for IA and its appealing UI.

The main advantages of the second game are its vast variety of features: online multiple player mode and interaction with other users, several languages, appealing UI, inclusion of other games and personalisation of the account (avatar and personal information).

However, these apps don’t provide any means to rewatch the games or customize colors inside the app.

Testability, Monitoring and Alerting

The project implements JEST tests for each of the screens of the application. These tests cover around 50% of all the code of the application and we will keep increasing the coverage of these. The main functionalities such as the Game itself, the AI, the HomeScreen, the Navigation and Settings are thoroughly tested.

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line
All files	51.54	48.05	50.87	50.74	
Ultimate-Tic-Tac-Toe	59.3	51.51	47.61	60.97	
App.js	0	0	0	0	8-36
RulesScreen.js	100	50	100	100	15-17
TutorialScreen.js	68.51	56	46.15	72.54	62-63,74,91,95
metro.config.js	0	100	100	0	1-6
react-native.config.js	0	100	100	0	1
Ultimate-Tic-Tac-Toe/AppLogic	0	0	0	0	
NavigationLogic.js	0	0	0	0	22-90
ThemeLogic.js	0	0	0	0	13-70
Ultimate-Tic-Tac-Toe/Firebase	0	100	100	0	
firebaseConfig.js	0	100	100	0	15-28
Ultimate-Tic-Tac-Toe/GameLogic	81.18	65.97	90.24	79.78	
Game.js	73.33	66.66	88.23	71.42	97,102,141-143
MonteCarloTreeSearch.js	92.68	63.63	91.66	92.95	19-22,42
Ultimate-Tic-Tac-Toe/GamePlayScreen	83.33	76.59	84.21	83.13	
BigBoard.js	100	100	100	100	
Cell.js	100	100	100	100	
GamePlayScreen.js	78.26	65	85.71	77.77	42,70-81,84
SmallBoard.js	100	100	100	100	
WinnerModal.js	75	66.66	50	75	18,32-45
Ultimate-Tic-Tac-Toe/HomeScreen	95.65	66.66	90	95.23	
HomeScreen.js	95.65	66.66	90	95.23	36
Ultimate-Tic-Tac-Toe/HomeScreen/homeComponents	75	0	66.66	75	
ButtonComponent.js	100	100	100	100	
HomeTitleComponent.js	100	100	100	100	
TabComponent.js	0	0	0	0	9-58
Ultimate-Tic-Tac-Toe/SettingsScreen	39.54	50.9	34.21	39.71	
AuthenticationSection.js	19.84	13.15	14.28	19.68	43,50-67,72-88
ColorPicker.js	70.58	83.33	62.5	71.87	47,89,92-125
ResetColorTheme.js	52.63	50	33.33	52.63	25-46
SettingsScreen.js	57.14	41.66	33.33	57.14	30-45
ThemePicker.js	100	50	100	100	20-52

For monitoring, the application integrates Sentry, which reports any error produced in the application besides monitoring user activity

around the application.

Besides that, firebase is not only useful for storage and authentication,but also for monitoring.