

# Shake Sudoku

Benjamin Horton

Benjamin Romalis

Mateos O'Connor

## Overview

Sudoku is a very popular puzzle game that is easily accessed in our life, in almost every newspaper, the subjects of entire books dedicated to this pastime, and importantly, in apps. While a variety of apps have covered the standard version as well as its variants(the most known being Samurai Sudoku). However, it is depressing and anger-inducing to not be able to figure out the solution, and in online-cases, lose all progress after too many mistakes. However, we decided that it should be up to the user to decide the difficulty of the Sudoku board, thus letting them gauge their skill level with our app.

## Goals

At the start, they are given a hard Sudoku puzzle immediately. For experts, this should be enough to start playing. For more casual players, all they have to do is look at the board, and if they do not believe they can do it with ease or at all, they are free to shake their iPhone to make the difficulty easier.

## How does it work?

Initially we were to use the shaking gesture to clear mistakes or reset the board, but we didn't want the user to be shaking their phone in the middle of the game and ruin their focus. Therefore, we chose to implement this action at the start of the game! All the player has to do is shake their iPhones *before any user input(i.e. selecting a number for any time)*, and the board will update, giving more numbers for the player to work with. We had decided to give about five numbers each difficulty, with a total of 21 numbers maximum when starting a Sudoku game with our app. Just in case there are any unintentional shaking during the start of the game, there is an alert that asks if this is the correct action you wanted. Since there are only three difficulty options, we didn't feel like it was annoying to the user, and created a more personal connection with the immersion of the game.

## How will I know if I am doing well?

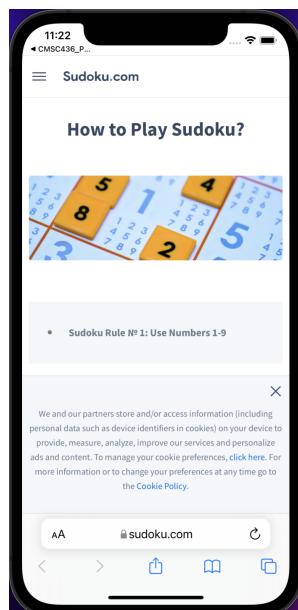
To ensure the player has no confusion as to which tile is selected, which needs to be corrected, etc. We have chosen specific coloring of each cell to make sure the player only has to concentrate on their ability to solve the board rather than figuring out the controls.

## User Interactions

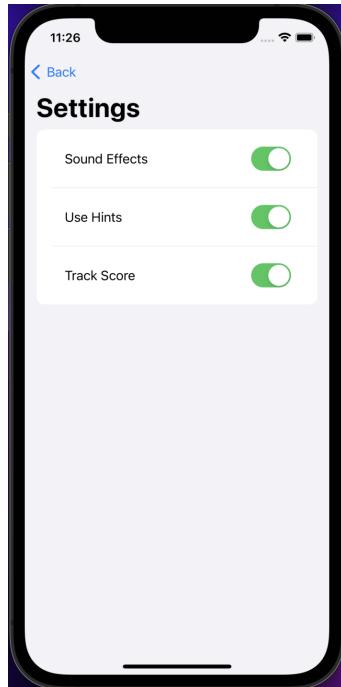
When the app starts, if you have no current games you are working on, it will look like the screen on the left. If there is a current game that is being tracked, it will look like the screen on the right instead:



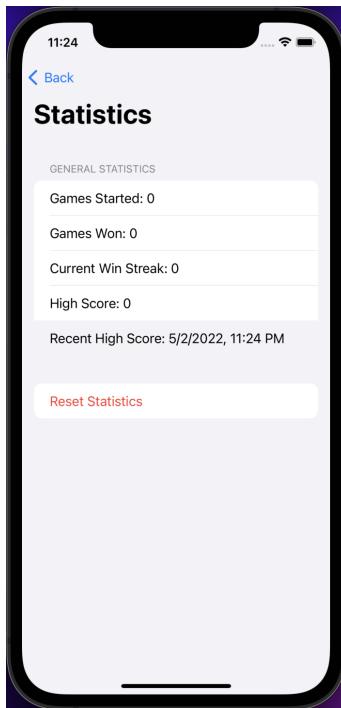
If you are unsure how to play Sudoku, the “Question-Mark” symbol in the top-right toolbar will lead you to this site on the rules of the game:



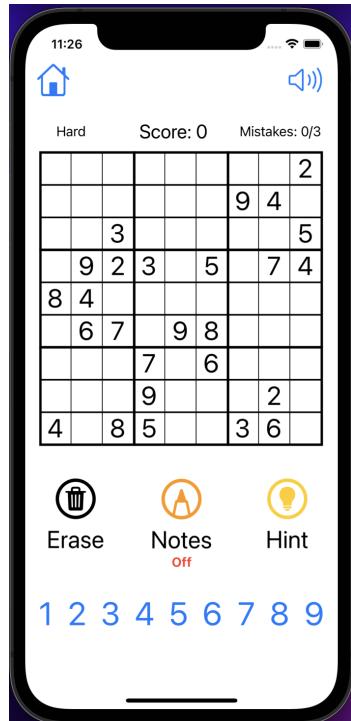
For the different User Defaults from this app, the “Gear” symbol on that same toolbar will lead you to the User Settings, where you can toggle whether or not you want hints, SFX, or your score to be tracked:



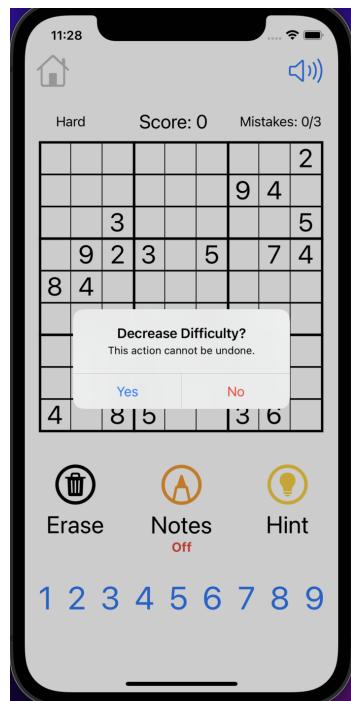
There is also a “Notebook/Whiteboard” symbol on the same toolbar, that will present player statistics while playing the app:



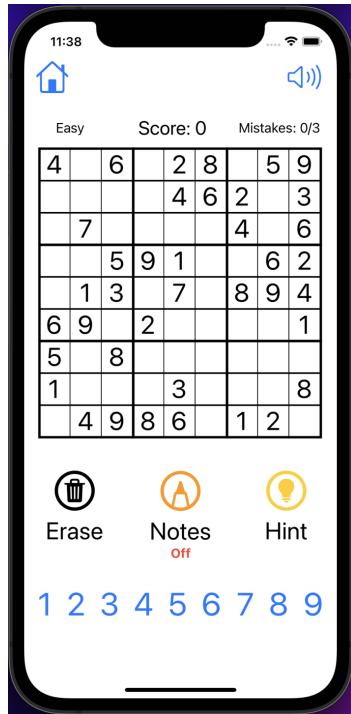
Once you start the game, you will be given a partially-filled board, starting on the hardest difficulty:



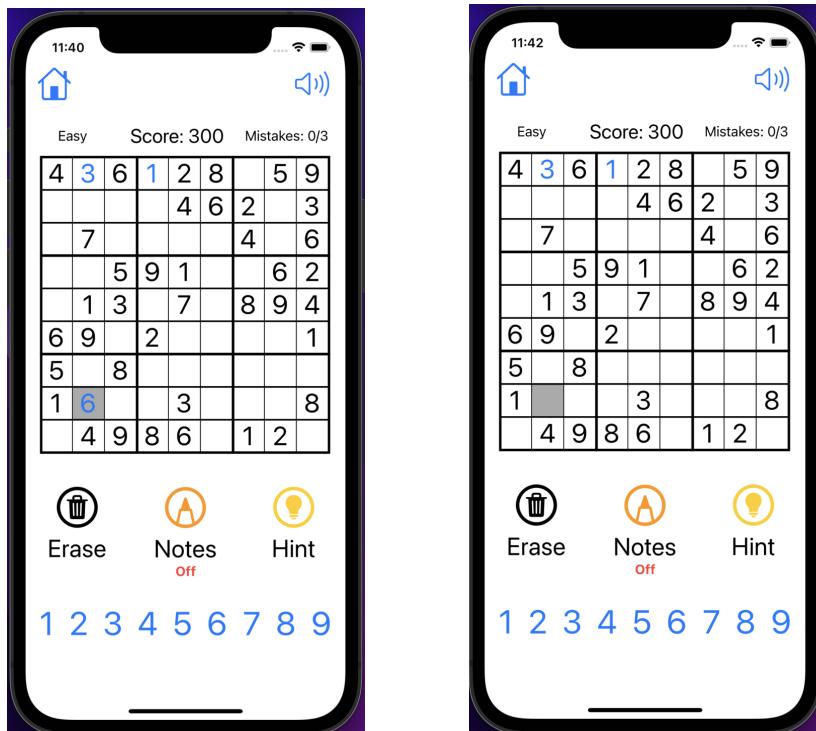
To decrease the difficulty, you shake the phone **before inputting any values to an empty tile**, and you are prompted with an alert to confirm your choice:



If you confirm it, you are given more numbers, and the difficulty in the top left of the board will decrease, until the difficulty turns to “Easy”, where it will not give you more tiles nor allow you to make it easier by shaking it:



To delete a tile, you must select a tile that *you changed yourself, not a given tile*, and then click the erase button to delete the cell:



To use a hint, press the “lightbulb” symbol in order to be given a correct input for an empty cell **in exchange for a score penalty**; if you have used all of your hints(4 total) nothing happens when the lightbulb is pressed going forward:



## Development Process

From the beginning of the project, we met once a week either in-person or via zoom. Each week we worked together on a new phase of the project and then assigned tasks for each group member to work on until our next meeting. We fell a bit behind leading up to milestone 1 as we tried to develop our own board generator and solver. The board generator was easily developed; however, the solver was very difficult for us to implement. As a result, we shifted our focus to the rest of the sudoku app and simply used pre-generated puzzles found online.

We ran into some issues with a couple important issues, one of which we were able to overcome. For the User Interface of the app, there was initially some confusion on the handling of NavigationView, leading to an overlap of View instances. This interfered with overlapping buttons and frames that conflicted with the overall appearance and game-feel. Luckily, that was a quick-fix, which involved taking out the second Navigation while in the main game struct, CurrentGame().

The biggest hurdle that we were faced with was the generation of the board, which we had spent every week trying to find a consistent while random approach to getting one solution. Creating a completed board was simple to implement, but in order to create a starting point, there must be only one path when selectively taking out individual cells at a time. We failed to implement this, due to the difficulty of handling the different permutations of solutions from this task, and resorted to puzzles that have already been published. In the future, we plan on creating one path for this consistently, and using it to create more diversity in the player experience.

Lastly, the “notes” that were planned on being taken involved the coloring of a selected cell to be yellow, and another set of numbers(much smaller) that would be in the tile to track notes. There were many UI issues that also conflicted with the appearance of the cells themselves, especially when taking into account the user’s ability to see the numbers without it leaking through another cell. In addition, there were issues that came from different objects that would come from registering if it was a selected/wrong/empty/noted cell. While we have a tracker to see the noted numbers for each cell, we fell short when trying to visualize it for our users.

### **Potential Future Direction**

For the future, we need to tie up loose-ends, especially with some UI tweaks. From the start, I wanted to support different appearances, such as the standard light-dark mode setting, but with more elements that would treat the user to enjoyable appearances that to their liking, such as a custom background, and perhaps different fonts and colors(within reason) to give the user more incentive to continue playing. However, the most important element that we need to work on is board generation for the app, so in the future we intend to create a database that can hold and extract from a larger pool. Another future tool that can be a selling-point for this app would be the use of the camera to scan a written puzzle, so that it can be solved through the app. Lastly, with an additional timer for the game, I wanted to make the game competitive, in order to encourage replayability by challenging friends through the share action, which would send over the seed for that specific puzzle, and once the second user finishes, the comparison is shown for both score and time.