

# SE 3XA3: Test Report

## Sudoku Solver

Team 08, SudoCrew  
Rashad A. Bhuiyan (bhuiyr2)  
Kai Zhu (zhuk2)  
Stanley Chan (chans67)

April 12, 2022

### Contents

<b>1</b>	<b>Functional Requirements Evaluation</b>	<b>1</b>
<b>2</b>	<b>Nonfunctional Requirements Evaluation</b>	<b>1</b>
2.1	Look and Feel . . . . .	1
2.2	Usability and Humanity . . . . .	2
2.3	Performance . . . . .	2
2.4	Security . . . . .	3
<b>3</b>	<b>Comparison to Existing Implementation</b>	<b>3</b>
<b>4</b>	<b>Changes Due to Testing</b>	<b>3</b>
<b>5</b>	<b>Automated Testing</b>	<b>3</b>
<b>6</b>	<b>System Tests</b>	<b>4</b>
6.1	Functional Requirement Testing . . . . .	4
6.2	Homepage Display Testing . . . . .	4
6.3	User Interface Design Testing . . . . .	5
6.4	Ease of Use Testing . . . . .	5
6.5	Ease of Learning Testing . . . . .	5
6.6	Speed Testing . . . . .	6
6.7	Capacity Testing . . . . .	6
6.8	File Integrity Testing . . . . .	6
6.9	Access, Privacy, and Immunity Testing . . . . .	6
<b>7</b>	<b>Trace to Requirements</b>	<b>7</b>
<b>8</b>	<b>Trace to Modules</b>	<b>8</b>
<b>9</b>	<b>Code Coverage Metrics</b>	<b>8</b>

<b>10 Appendix</b>	<b>8</b>
10.1 Symbolic Parameters . . . . .	8

## List of Tables

1	<b>Revision History</b> . . . . .	iii
2	Test for FRH . . . . .	4
3	Test for FRIU . . . . .	4
4	Test for FRMI . . . . .	4
5	Test for FRP . . . . .	4
6	Test for SS-HD-1 . . . . .	4
7	Test for SS-HD-2 . . . . .	5
8	Test for SS-UID-1 . . . . .	5
9	Test for SS-EU-1 . . . . .	5
10	Test for SS-EU-2 . . . . .	5
11	Test for SS-EL-1 . . . . .	5
12	Test for SS-S-1 . . . . .	6
13	Test for SS-S-2 . . . . .	6
14	Test for SS-C-1 . . . . .	6
15	Test for SS-FI-1 . . . . .	6
16	Test for SS-AP-1 . . . . .	6
17	Trace Between Tests and Requirements . . . . .	7
18	Trace Between Tests and Modules . . . . .	8

## List of Figures

Table 1: **Revision History**

<b>Date</b>	<b>Version</b>	<b>Notes</b>
2022-04-04	0.0	Created Test Report
2022-04-12	1.0	Finished Test Report

This document describes the results of completing all of the tests listed in the Test Plan. This ensures that any person who reads this document is successfully able to reproduce the results of the test cases.

## 1 Functional Requirements Evaluation

Description: The purpose of these tests are to ensure that the user is able to use the software according to the specified functional requirements documentation. These tests include the following business events: arrival at the homepage where all pages load correctly, accurate results from an image upload, accurate results from manual input, and board generation for user to play a game of Sudoku.

1. **Test Suite Name: FRH**

Results: The user is able to view the homepage and all other pages without any issue.

2. **Test Suite Name: FRIU**

Results: The user is able to upload a valid image and get a solution.

3. **Test Suite Name: FRMI**

Results: The user is able to manually input a valid board and get a solution.

4. **Test Suite Name: FRP**

Results: The user is able to play a game of Sudoku at the specified difficulty level.

## 2 Nonfunctional Requirements Evaluation

### 2.1 Look and Feel

Description: The overall look and feel of the web application was tested by having a small group of McMaster students from streams outside of Software Engineering. This group of users better represent the wide array of our potential users. They were given simple feedback questions in 1-on-1 interviews where their responses were recorded.

#### Homepage Display Testing

1. **Test Name: SS-HD-1**

Results: Users gave feedback on button positions and were able to successfully access the desired web page based on the button clicked. Any flaws in connection were noted and changes were applied.

2. **Test Name: SS-HD-2**

Results: The web application was able to be clearly loaded and understood when users changed the browser size or loaded the web application on smaller screens.

#### User Interface Design Testing

1. **Test Name: SS-UID-1**

Results: The users gave feedback on the overall design of the web application, giving a rating out of 10 based on the visual appeal and accessibility of the web page.

## 2.2 Usability and Humanity

Description: The usability of the web application was tested by having a small group of users of varying technological backgrounds and ages, ranging from a middle school student to a professional engineer, attempt to use the functions of the web application without any prior explanation. Afterwards, each participant was interviewed 1-on-1 and feedback was noted about the overall usability of the web application.

### Ease of Use Testing

1. **Test Name: SS-EU-1**

Results: Users were able to successfully upload an image to get a Sudoku board solution as well as manually input a board for a solution. Feedback was noted about the overall instructions behind each function and relevant changes were made to make it easier to use and understand.

2. **Test Name: SS-EU-2**

Results: Users were able to successfully complete a game of Sudoku of their chosen difficulty level and submit their answer. The difficulty level selected was based on their prior experience playing Sudoku.

### Ease of Learning Testing

1. **Test Name: SS-EL-1**

Results: Users were able to successfully understand the rules of Sudoku and play a game of Sudoku. Clarity of instructions were changed based on feedback given by novice Sudoku players.

## 2.3 Performance

Description: The overall performance of the web application was tested by having a small group of software engineering students from McMaster time the overall execution of various functions. These times were noted and compared to the maximum time allowed and changes were made to ensure that functions are completed within the time limit.

### Speed Testing

1. **Test Name: SS-S-1**

Results: Users were able to successfully time the upload of an image within the specified time frame. No changes were necessary.

2. **Test Name: SS-S-2**

Results: Users were able to successfully time the submission of a Sudoku board, however some users experienced a larger wait time than others. Minor changes were made in order to optimise the submission time.

### Capacity Testing

1. **Test Name: SS-C-1**

Results: Users were able to successfully upload an image of a Sudoku board. Upon submission, users noted that the image was not stored anywhere locally.

## 2.4 Security

Description: The security of the web application was tested by having a small group of software engineering students from McMaster upload images and change entries of the Sudoku board to check if users can cause harm to the web application.

### File Integrity Testing

1. **Test Name: SS-FI-1**

Results: Users were presented with an error message after attempting to upload an image that was either too large or too small.

### Access, Privacy, and Immunity Testing

1. **Test Name: SS-AP-1**

Results: Users were unable to edit the existing numbers when playing a game of Sudoku, or when checking the solution to a given image or manual submission.

## 3 Comparison to Existing Implementation

There were 4 test reports that associated to the existing implementation of the program, which are noted as follows:

- FRIU in Functional Requirements Evaluation
- FRMI in Functional Requirements Evaluation
- FRP in Functional Requirements Evaluation
- SS-S-2 in Nonfunctional Requirements Evaluation

After completion of each test, the existing implementation was refactored in order to meet the performance requirement. Correctness and validity of outputs were verified using manual and automated testing.

## 4 Changes Due to Testing

There were no changes required for completion of the functional requirements. Minor changes were made to the button layout and color scheme based on feedback given by the test group. Furthermore, changes to the Sudoku board generation and solution algorithm were implemented in order to decrease wait time for users. Finally, most of the changes were made to the instructions for each of the web-pages in order to be much more accessible to those with a less technological background.

## 5 Automated Testing

Automated testing was minimally used, focussing on the Sudoku board generation and solution verification. Most of the testing was completed manually as there was a focus on the user interface as well as use of machine learning. Automated testing for machine learning is currently out of our scope while the user interface can most accurately be tested through walkthroughs and exploratory testing by users.

## 6 System Tests

### 6.1 Functional Requirement Testing

<b>Test Name</b>	FRH
<b>Initial State</b>	Homepage of Sudoku Solver
<b>Input</b>	Button clicks to other pages
<b>Expected Output</b>	Each button redirects user to the relevant webpage

Table 2: Test for FRH

<b>Test Name</b>	FRIU
<b>Initial State</b>	Image upload page of Sudoku Solver
<b>Input</b>	Button clicks to upload image
<b>Expected Output</b>	Successful submission of an image where the solution is then provided

Table 3: Test for FRIU

<b>Test Name</b>	FRMI
<b>Initial State</b>	Manual input page of Sudoku Solver
<b>Input</b>	Button clicks to submit Sudoku board
<b>Expected Output</b>	Successful submission of a Sudoku board manually where the solution is then provided

Table 4: Test for FRMI

<b>Test Name</b>	FRP
<b>Initial State</b>	Play page of Sudoku Solver
<b>Input</b>	Button clicks to select difficulty
<b>Expected Output</b>	Arrival at gameplay page where users can play a game of Sudoku at the specified difficulty

Table 5: Test for FRP

### 6.2 Homepage Display Testing

<b>Test Name</b>	SS-HD-1
<b>Initial State</b>	Homepage of the Sudoku solver Flask app
<b>Input</b>	Users from testing group will view the homepage
<b>Expected Output</b>	Users will give feedback on button layout

Table 6: Test for SS-HD-1

<b>Test Name</b>	SS-HD-2
<b>Initial State</b>	Homepage of the Sudoku solver Flask app
<b>Input</b>	Users from the test group will change the browser size
<b>Expected Output</b>	The Homepage scales accordingly, making UI changes based on the dimensions of the window

Table 7: Test for SS-HD-2

### 6.3 User Interface Design Testing

<b>Test Name</b>	SS-UID-1
<b>Initial State</b>	Homepage of the Sudoku solver Flask app
<b>Input</b>	Users from the test group view the web app
<b>Expected Output</b>	Users give feedback on the overall UI

Table 8: Test for SS-UID-1

### 6.4 Ease of Use Testing

<b>Test Name</b>	SS-EU-1
<b>Initial State</b>	Manual input page or Image Upload page
<b>Input</b>	Users from test group follow instructions of respective page
<b>Expected Output</b>	System accepts provided Sudoku board

Table 9: Test for SS-EU-1

<b>Test Name</b>	SS-EU-2
<b>Initial State</b>	Sudoku gameplay page
<b>Input</b>	Users from test group start playing the game
<b>Expected Output</b>	Users play Sudoku and are given solution at the end

Table 10: Test for SS-EU-2

### 6.5 Ease of Learning Testing

<b>Test Name</b>	SS-EL-1
<b>Initial State</b>	How to Play Sudoku page
<b>Input</b>	Users from the test group read the page
<b>Expected Output</b>	Users understand rules of Sudoku

Table 11: Test for SS-EL-1



## 6.6 Speed Testing

<b>Test Name</b>	SS-S-1
<b>Initial State</b>	Upload image page
<b>Input</b>	User from test group submits an image
<b>Expected Output</b>	Loading icon will take MAX_UPLOAD_TIME to upload

Table 12: Test for SS-S-1

<b>Test Name</b>	SS-S-2
<b>Initial State</b>	Verify upload page
<b>Input</b>	User from test group submits an image
<b>Expected Output</b>	Loading icon will take MAX_UPLOAD_TIME to upload

Table 13: Test for SS-S-2

## 6.7 Capacity Testing

<b>Test Name</b>	SS-C-1
<b>Initial State</b>	Upload Image page
<b>Input</b>	User from test group uploads image
<b>Expected Output</b>	Image is not stored locally

Table 14: Test for SS-C-1

## 6.8 File Integrity Testing

<b>Test Name</b>	SS-FI-1
<b>Initial State</b>	Upload Image page
<b>Input</b>	User from test group uploads image that is too large
<b>Expected Output</b>	Error is given about file size

Table 15: Test for SS-FI-1

## 6.9 Access, Privacy, and Immunity Testing

<b>Test Name</b>	SS-AP-1
<b>Initial State</b>	Solution of Sudoku board
<b>Input</b>	User from test group attempts to edit shown solution
<b>Expected Output</b>	User cannot edit solution

Table 16: Test for SS-AP-1

## 7 Trace to Requirements

Test	Requirements
Functional Requirements Testing	
FRH	FR1-3
FRIU	FR4-10
FRMI	FR11-15
FRP	FR16-24
Nonfunctional Requirements Testing	
SS-HD-1	LF1, PR6
SS-HD-2	LF2
SS-UID-1	LF3, LF4, CR1
SS-EU-1	UH1
SS-EU-2	UH2
SS-EL-1	UH3, UH4, UH5
SS-S-1	PR1
SS-S-2	PR2, PR3, PR4
SS-C-1	PR5, SR4
SS-FI-1	SR1, SR2
SS-AP-1	SR3

Table 17: Trace Between Tests and Requirements

## 8 Trace to Modules

Test	Requirements
Functional Requirements Testing	
FRH	M8
FRIU	M3, M4, M5, M6, M7, M8
FRMI	M3, M7, M8
FRP	M2, M7, M8
Nonfunctional Requirements Testing	
SS-HD-1	M7, M8
SS-HD-2	M8
SS-UID-1	M8
SS-EU-1	M7
SS-EU-2	M1
SS-EL-1	M7, M8
SS-S-1	M4, M7
SS-S-2	M3, M4, M5, M6, M7
SS-C-1	M4, M7, M8
SS-FI-1	M7, M8
SS-AP-1	M3, M5, M7, M8

Table 18: Trace Between Tests and Modules

## 9 Code Coverage Metrics

The team of behind the Sudoku Solver web application was able to produce roughly 98% of code coverage metrics. These metrics were realized through the use of the testing plan. This number was approximated based on the results of testing and how each output can be traced to a specified requirement and module. Please refer to both the Requirements documentation and Module documentation for the relevant requirements and modules. This clearly shows the traceability between the testing plan, results of testing, requirements of the program, and the module guide.

## 10 Appendix

### 10.1 Symbolic Parameters

The definition of the requirements will likely call for SYMBOLIC\_CONSTANTS. Their values are defined in this section for easy maintenance.

*MAX\_UPLOAD\_TIME* = 8  
*MAX\_SOLUTION\_TIME* = 5