

CSC470: Software Engineering Final Report  
TESS: The Extraordinary Sudoku Solver

Team: #1  
David Koval and Joseph Mammo  
Instructor: Dr. Ahyoung Lee

4/24/17  
v1.1

## Table of Contents

<b>1</b>	<b>Summary</b>	<b>3</b>
<b>2</b>	<b>Introduction</b>	<b>3</b>
2.1	Purpose . . . . .	3
2.2	Scope . . . . .	3
2.3	Definitions, acronyms, and abbreviations . . . . .	3
<b>3</b>	<b>Goals</b>	<b>3</b>
<b>4</b>	<b>Specific Requirements</b>	<b>3</b>
4.1	Functional Requirements . . . . .	3
4.2	Non-Functional Requirements . . . . .	5
<b>5</b>	<b>System Design</b>	<b>5</b>
5.1	Design Overview . . . . .	5
5.2	Realistic Constraints and Professional Standards . . . . .	5
5.3	Alternative Designs and Design Choices . . . . .	5
<b>6</b>	<b>System Implementation</b>	<b>6</b>
<b>7</b>	<b>System Testing</b>	<b>7</b>
7.1	Test Plan . . . . .	7
7.2	Test Results . . . . .	7
<b>8</b>	<b>Conclusions</b>	<b>7</b>

# 1 Summary

High level summary with 1 page. the project goals, motivation or problem issues (requirements). Design considerations and choices to solve the problems to achieve the goals. Implementation, validation and testing plans.

!!!

## 2 Introduction

Overall introduction

### 2.1 Purpose

### 2.2 Scope

### 2.3 Definitions, acronyms, and abbreviations

Term	Definition
DESC	Description
Grid	This is where all the values are stored for the player to see.
ID	Identification
PW	Priority Weights
User	Whoever will be using the app

## 3 Goals

This is the goals section.

## 4 Specific Requirements

This is the specific requirements section.

Be sure to include numbering scheme including Identifier (RQ1, RQ2, and RQn) and PW (the priority weights, may be the highest priority = 5 and the lowest priority = 1) to allow traceability.

Provide a high-level use case diagram for the high-level system models and a traceability matrix for the requirement validation.

### 4.1 Functional Requirements

#### ID:R1

TITLE: Play Game

DESC: The user first opens up the app, they should be able to choose the option to play a Sudoku puzzle. They user should be able to stay as long as they want to on this screen.

#### ID:R2

Add priority weights

**TITLE:** Select Difficulty

**DESC:** The user should be able to select a difficulty setting that better suites needs at any time. There should be 5 difficulty options for the user to choose from. 1 being the easiest all the way down to 5 being the hardest.

**ID:R3**

**TITLE:** Back Option

**DESC:** The user should be able to return to the main screen from the select difficulty screen if they don't select a difficulty. The user can remain on the select difficulty as long as they want to.

**ID:R4**

**TITLE:** Continue Game

**DESC:** The user should be able to continue a game that has been previously started whether or not the app has been closed. All of the user's input should be saved so they could be brought up again should the user want to continue a game.

**ID:R5**

**TITLE:** Get Puzzle

**DESC:** When the user selects a difficulty, a puzzle with the selected difficulty should be retrieved from the database for the user to be able to play it and enjoy the game.

**ID:R6**

**TITLE:** Solver Option

**DESC:** On started, the user should be able to select the Solver option in the app to go straight to the solver part of the app.

**ID:R7**

**TITLE:** Input Sudoku Puzzle To Solve

**DESC:** The user needs to be able to input any sort of Sudoku puzzle that the user has, with or without any extra numbers the user wishes to input.

**ID:R8**

**TITLE:** Check Current Board

**DESC:** The user should be able to check the status of the current puzzle they are working on. The user should be able to view the inputs that are conflicting with each other. They should be marked in some way for the user to be able to see them clearly.

**ID:R9**

**TITLE:** Delete an Input Value

**DESC:** The user should be able to delete an input value that they put in previously or a value that the system put in automatically.

**ID:R10**

**TITLE:** Clear Board

**DESC:** The user should be able to clear the entire board easily and effortlessly.

**ID:R11**

**TITLE:** Solve Current Board

**DESC:** The user should be able to have the option to solve the current board that they are working on. It should use the input the system put in as well as the inputs the user decided to add.

**ID:R12**

TITLE:

## **4.2 Non-Functional Requirements**

**ID:RQ1**

TITLE: System Availability

DESC: The system needs to be available to the user 99.9% of the time, whether or not the system is being used. The system can only be down for 0.1% of the time for updates or maintenance.

**ID:RQ2**

TITLE: Memory Space

DESC: The app should have a small footprint. The entire app should take up less than 5MBs of memory, this is including with future updates as well.

**ID:RQ3**

TITLE: Solve Time

DESC: The app should be able to determine a solution, if possible, for a given Sudoku puzzle in under 5 seconds.

**ID:RQ4**

TITLE: Search Algorithm

DESC: The Suduko solver must utilize some search algorithm such as depth first search or breadth first search or any other searching algorithm.

**ID:RQ5**

TITLE:

## **5 System Design**

This is the system design section.

### **5.1 Desgin Overview**

Provide an overview of the design, including diagrams, key design subsections, and how they relate or connect to one another (e.g., Interaction, structural models).

### **5.2 Realistic Constraints and Professional Standards**

Identify and discuss realistic constraints on the problem, such that constraints may include economic, environmental, social, ethical, health and safety, manufacturability, policy issues, etc.

### **5.3 Alternative Designs and Design Choices**

Describe alternative designs that were considered during execution of the project. Discuss how design choices were guided by constraints and other factors. E.g.,

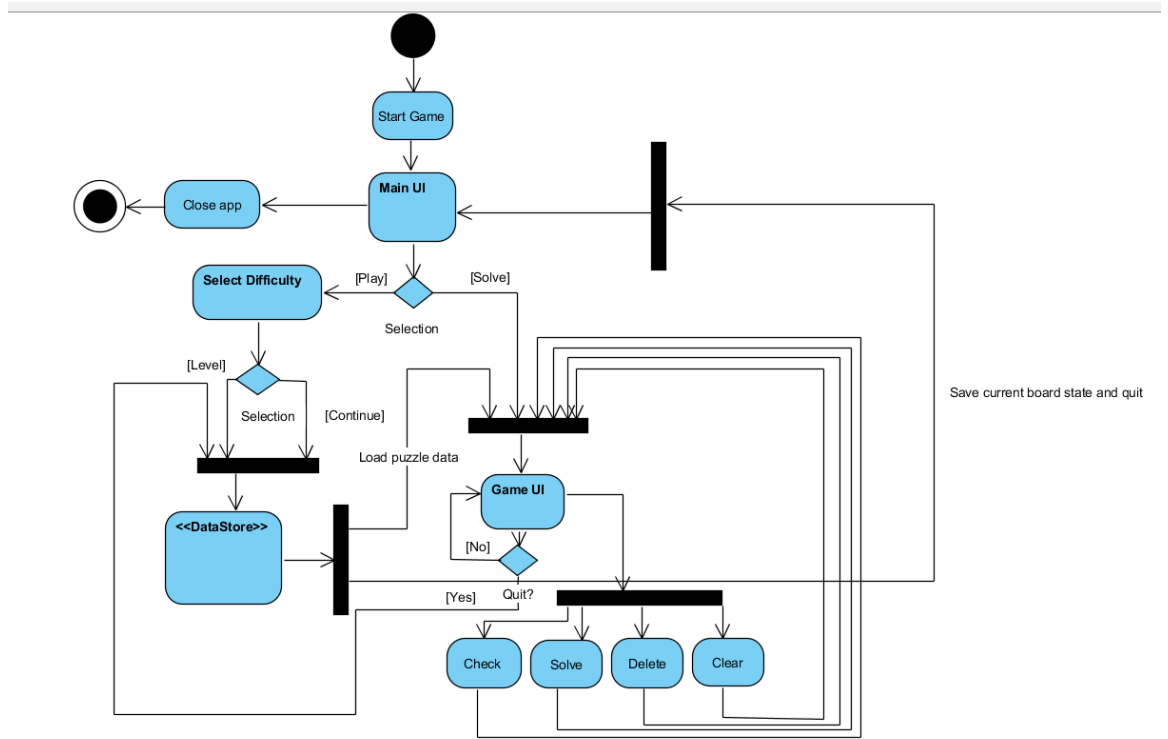


Figure 1: Activity diagram of a system.

architectural design models – Layered or Client-server and details shown using activity diagram as shown in Figure 1 (Context model), sequence diagram (Interaction model), class diagram (Structural model).

## 6 System Implementation

This is the system implementation section.

Describe the technical details for each of the subsystems or a the system-level and provide sequence diagrams or station/activity diagrams for your system implementation.

## 7 System Testing

This is the system testing section.

### **7.1 Test Plan**

Provide your test plan with unit testing (Black-box testing and White-box testing), integration testing (Top-down or bottom-up approach) and system testing.

### **7.2 Test Results**

Show your test results and evaluate them.

## **8 Conclusions**

This is the conclusions section.

Overall summary of design methodologies, key creative approaches and potential contribution/impact. [1]

## **References**

[1] J. Mammo, “Hahahahaha,” *Computer Networks*, vol. 71, pp. 1–30, 2017.