SE 3XA3: Software Requirements Specification
DNA Says

Team 10, Team Name: DNA
Kareem Abdel Mesih (abdelk2)
John-Paul Dakran (dakranj)
Shady Nessim (nessimss)

December 2, 2016

Contents

1 Project Drivers **1**
- 1.1 The Purpose of the Project . . . 1
- 1.2 The Stakeholders . . . 1
  - 1.2.1 The Client . . . 1
  - 1.2.2 The Customers . . . 1
  - 1.2.3 Other Stakeholders . . . 1
- 1.3 Mandated Constraints . . . 2
  - 1.3.1 Solution Constraints . . . 2
  - 1.3.2 Partner or Collaborative Applications . . . 2
  - 1.3.3 Budget Constraints . . . 2
  - 1.3.4 Scheduling Constraints . . . 2
  - 1.3.5 Enterprise Constraints . . . 2
- 1.4 Naming Conventions and Terminology . . . 2
- 1.5 Relevant Facts and Assumptions . . . 3
  - 1.5.1 Relevant Facts . . . 3
  - 1.5.2 Assumptions . . . 3

2 Functional Requirements **4**
- 2.1 The Scope of the Work and the Product . . . 4
  - 2.1.1 The Context of the Work . . . 4
  - 2.1.2 Work Partitioning . . . 5
  - 2.1.3 Individual Product Use Cases . . . 5

3 Functional Requirements **7**

4 Non-functional Requirements **10**
- 4.1 Look and Feel Requirements . . . 10
  - 4.1.1 Appearance Requirements . . . 10
  - 4.1.2 Style Requirements . . . 12
- 4.2 Usability and Humanity Requirements . . . 12
  - 4.2.1 Ease of Use Requirements . . . 12
  - 4.2.2 Personalization and Internationalization Requirements 13
  - 4.2.3 Learning Requirements . . . 13
  - 4.2.4 Understandability and Politeness Requirements . . . 14
  - 4.2.5 Accessibility Requirements . . . 14
- 4.3 Performance Requirements . . . 15

4.3.1 Speed and Latency Requirements . . . 15
4.3.2 Safety Critical Requirements . . . 15
4.3.3 Precision of Accuracy Requirements . . . 16
4.3.4 Reliability and Availability Requirements . . . 16
4.3.5 Capacity Requirements . . . 16
4.3.6 Scalability Requirements . . . 17
4.3.7 Longevity Requirements . . . 17
4.4 Operational and Environmental Requirements . . . 17
4.4.1 Expected Physical Environment . . . 17
4.4.2 Release Requirements . . . 17
4.5 Maintainability and Support Requirements . . . 18
4.5.1 Maintenance Requirement . . . 18
4.5.2 Supportability Requirements . . . 18
4.5.3 Adaptability Requirements . . . 18
4.6 Security Requirements . . . 19
4.6.1 Privacy Requirements . . . 19
4.7 Cultural Requirements . . . 19
4.8 Legal Requirements . . . 20

5 Project Issues **20**
5.1 Open Issues . . . 20
5.2 Off-the-Shelf Solutions . . . 20
5.3 New Problems . . . 20
5.4 Tasks . . . 20
5.5 Migration to the New Product . . . 21
5.6 Risks . . . 21
5.7 Costs . . . 21
5.8 User Documentation and Training . . . 21
5.9 Waiting Room . . . 21
5.10 Ideas for Solutions . . . 21

6 Appendix **22**
6.1 Symbolic Parameters . . . 22

List of Tables

1 **Revision History** . . . iii

2 List of Terminology 3
3 List of Events 5

List of Figures

1 Context of Work Diagram 4

Table 1: **Revision History**

Date	Version	Notes
2016/10/10	1.0	Completion of sub-section 1 & 2
2016/10/10	2.0	Completion of sub-section 4
2016/10/11	3.0	Completion of sub-section 3
2016/12/02	4.0	Revision 1

1 Project Drivers

1.1 The Purpose of the Project

Video games have always been one of the top choices with regards to entertainment. They are also named as one of the great ways to overcome boredom. This project is a redevelopment of the famous digital game Simon Says, with a slight modification that makes DNA Says unique while keeping the integrity of the game consistent with the original version. This interactive game serves the purpose of allowing people of all ages, whether bored or simply having a break, to enjoy a fun and an interactive game. The main basis of Simon Says is to remember a given pattern, and iterate it back. In addition, this project will aid in the enhancement of ones visual and auditory memory.

1.2 The Stakeholders

1.2.1 The Client

This pogram is developped as the final project for McMaster University's Software Engineer 3XA3 - Software Project Management. Therefore, the client for this project is Dr. Spencer Smith, the Professor of that course.

1.2.2 The Customers

The customers for this project are the general public who will operate the game DNA Says. A typical customer will be any person ranging from five years of age and older, who can access and operate a computer.

1.2.3 Other Stakeholders

- The Development Team - Kareem, John-Paul and Shady.
- Previous and future developers as they possess the power to modify and publish this program as they desire.

1.3 Mandated Constraints

1.3.1 Solution Constraints

Description: This game is OS independant. It is compatible with Windows, Mac OS X, and Linux operating systems.

Rationale: The client will be using any of the operating systems listed above.

Fit Criterion: During the user testing phase, all of the operating systems mentioned above were tested.

1.3.2 Partner or Collaborative Applications

This project is a redevelopment of the digital game Simon Says in which its Python open source code is available online. The new game DNA Says supports the current game's user platform.

1.3.3 Budget Constraints

The operating budget of the project is $0. All resources needed to develop this game are currently owned by the developers.

1.3.4 Scheduling Constraints

The project must be fully completed by December 7, 2016. This includes the implementation, testing and documentation.

1.3.5 Enterprise Constraints

This game is free and accessible to all users who have exposure to a computer.

1.4 Naming Conventions and Terminology

.

Table 2: List of Terminology

Term	Definiton
OS	Short for operating systems.
Windows	Microsoft's operating system.
Mac OS X	Apple's operating system.
Linux	A Unix operating system.
Python	A programming language.
IDLE	Integrated development environment
LaTeX	A document preparation system.
Mode	Different subsections of the game.
GUI	Graphical user interface.
Gantt Chart	Chart outlining the timeline of the project.

1.5 Relevant Facts and Assumptions

1.5.1 Relevant Facts

Python is used to develop this project. It runs in its basic IDLE Version 3.5. Framework is tested in an automated fashion to validate the different cases and outcomes of the game. Family and friends have tested the overall functionality and performance of the game, as well as its non-functional requirements. LaTeX is to generate required documents.

The previous implementation of this game has approximately two hundred and fifty lines of code. That implementation only has one mode, however the version implemented has three different modes and a menu. Therefore the number of lines of code is greater than the original version's. The original implementation has no licenses that need to be acquired by the team or McMaster University.

1.5.2 Assumptions

It is assumed that the user has downloaded any version of Python along with its corresponding Pygame version.
It is also assumed that the user has basic understanding of operating a computer. The user must be able to open an application and follow simple instructions to interact with the GUI.

Finally, the user's computer must have enough processing speed and storage to effectively run and host the application.

2 Functional Requirements

2.1 The Scope of the Work and the Product

2.1.1 The Context of the Work

.

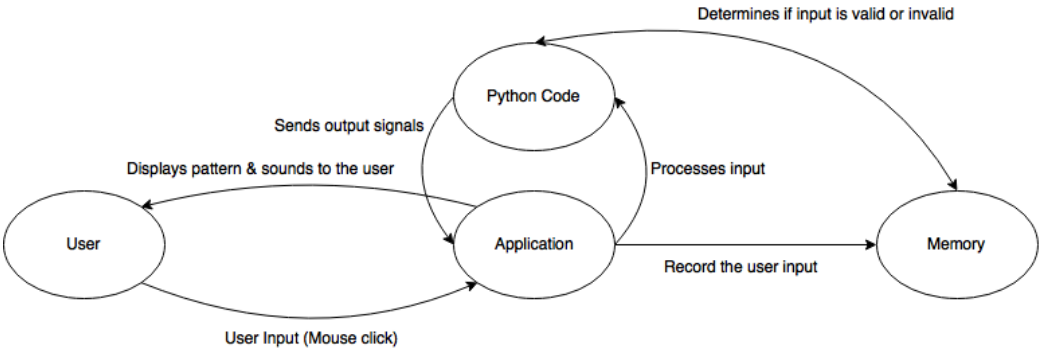

Figure 1: Context of Work Diagram

2.1.2 Work Partitioning

Table 3: List of Events

#	Event	Input	Output
1.	DNA Says Creation	Developer code	Executable file
2.	DNA Says Audio	None	Audio output device
3.	DNA Says GUI	Developer code	Monitor
4.	Open the file	User input	New window
5.	Select a mode	User input	Buttons appear
6.	Click a correct disk	User input	Light & sound
7.	Click an incorrect disk	User input	Light & sound
8.	Repeat pattern successfully	User input	Addition to the pattern
9.	Exit to main menu	User input	Main menu appears
10.	Exit game	User input	Window termination

2.1.3 Individual Product Use Cases

- Use Case #1
  - Name: Open the executable file.
  - Trigger: The user selects to open the file.
  - Precondition: The DNA Says icon must be available on the desktop.
  - Postcondition: The main menu will open.
- Use Case #2
  - Name: Select a mode.
  - Trigger: The user selects to choose one of the three modes.
  - Precondition: The user must be in the main menu.
  - Postcondition: The user will be able to view the three buttons and begin the game.
- Use Case #3
  - Name: Click a correct button.

  - Trigger: The user selects a button that was part of the pattern displayed in the correct playing order.
  - Precondition: The user must be in a mode and the computer has displayed the pattern.
  - Postcondition: The button will light up and make a sound.

- Use Case #4
  - Name: Click an incorrect button.
  - Trigger: The user selects a button that was not part of the pattern displayed.
  - Precondition: The user must be in a mode and the computer has displayed the pattern.
  - Postcondition: The game will make a specific sound indicating an incorrect move and the screen will flash.

- Use Case #5
  - Name: Successfully repeat the pattern.
  - Trigger: The user selects the series of button that composed the pattern displayed in order.
  - Precondition: The user must be in a mode and the computer has displayed the pattern.
  - Postcondition: The next pattern will be displayed to the user.

- Use Case #6
  - Name: Exit to the main menu.
  - Trigger: The user selects main menu icon.
  - Precondition: The user must be in a mode.
  - Postcondition: The user will leave a mode and the main menu will open.

- Use Case #7
  - Name: Exit game.

  - Trigger: The user selects the exit game icon
  - Precondition: The user must be in the main menu.
  - Postcondition: The appllicaion will be terminated

3 Functional Requirements

- Requirement #1
  - Description: The user will be able to open the executable file.
  - Rationale: The user must be able to open the program.
  - Fit Criterion: A new window will open on the user's computer screen.
- Requirement #2
  - Description: The interface will open in a new window.
  - Rationale: The program will be operated in a separate window.
  - Fit Criterion: A new window will appear on the user's computer screen
- Requirement #3
  - Description: The game will have three separate modes - Kareem Says, JP Says and Shady Says.
  - Rationale: The game is designed to have three distinct modes.
  - Fit Criterion: The three different modes will be displayed on the main menu of the game.
- Requirement #4
  - Description: The user will be able to select one of the three modes to play.
  - Rationale: The user must be able to play one mode at a time.
  - Fit Criterion: The user will be able to select one of the three modes displayed in the main menu of the game.
- Requirement #5

  - Description: The main menu will display the three different modes.
  - Rationale: The user must be able to view which mode they wish to select.
  - Fit Criterion: Three distinct icons will be displayed in the main menu.

- Requirement #6
  - Description: If Kareem Says is selected, then a piano will be displayed on the screen, otherwise nine squared buttons will show up for JP Says, and four for Shady Says.
  - Rationale: The game is designed to have different interfaces for each mode.
  - Fit Criterion: When a user selects a mode - accordingly, a piano will show up, nine buttons or four buttons.

- Requirement #7
  - Description: Each button will light up and produce a different sound when clicked.
  - Rationale: This gives the user the ability to detect the pattern that will be displayed.
  - Fit Criterion: When the user clicks a button, the button will light up and produce a sound.

- Requirement #8
  - Description: The user will be able to exit the game at any time and go back to the main menu.
  - Rationale: The user must have a means of exiting an ongoing game and return to the main menu.
  - Fit Criterion: When the user clicks the main menu button, they will find their screen in the main menu window.

- Requirement #9
  - Description: Every time a user passes a level, the score goes up by one point.

  - Rationale: A record of a user's score must be kept.
  - Fit Criterion: At level N, the score = N.

- Requirement #10
  - Description: Every time a user fails a level, the score is reset to zeo.
  - Rationale: When a user fails a level, the game must restart from level one.
  - Fit Criterion: Whenever the user makes a mistake, the score text will reset to zero.

- Requirement #11
  - Description: There will be a score icon in the top right corner.
  - Rationale: The user must be able to view their score.
  - Fit Criterion: When the user selects a mode, the score icon will be set to zero.

- Requirement #12
  - Description: At level N, a random pattern of N disks will light up and be displayed to the user.
  - Rationale: The pattern's length will increase as the levels progress.
  - Fit Criterion: During level one, one random button will light up and sound.

- Requirement #13
  - Description: The user cannot click the button while the pattern is being displayed.
  - Rationale: The pattern must be displayed to the user in full effect.
  - Fit Criterion: The program will not record clicks the user inputs during this time.

- Requirement #14
  - Description: The user will be able to click the buttons once the pattern has been displayed.

  - Rationale: The user must repeat the pattern correctly to pass the level.
  - Fit Criterion: The program will monitor the user's input clicks to determine if the entry is correct or not.

- Requirement #15
  - Description: A level is passed if the user repeats the pattern correctly.
  - Rationale: The user will be able to progress through the game.
  - Fit Criterion: The score will be increased by 1 when the user is successful.

- Requirement #16
  - Description: If the user fails, the game will restart - I.e. N = 1.
  - Rationale: The user must restart from the beginning of the game when a mistake is made.
  - Fit Criterion: Whenever a mistake is made, the user will be directed to level one.

4 Non-functional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

- Requirement #1
  - Description: The product shall have an appealing colorful appearance.
  - Rationale: The display should always be engaging so as to keep user interested in game. The product must be aesthetically pleasing and easy to use to benet the end-users
  - Originator: Shady Nessim
  - Fit Criterion: Stakeholder satisfaction regarding the appearance, user attraction to game.

  - Priority: High
  - History: Created October 5, 2016

- Requirement #2
  - Description: The buttons must be well designed and colored.
  - Rationale: The game revolves around pressing buttons in a pattern. It is the main entity of the game and must thus be aesthetically pleasing to attract user interest
  - Originator: Shady Nessim
  - Fit Criterion: User reaches high levels as a result of uniqueness and beauty of buttons.
  - Priority: High
  - History: Created October 5, 2016

- Requirement #3
  - Description: The product shall have attractive sound patterns. The associated sounds with buttons must be well constructed and notes must follow harmonically.
  - Rationale: The user follows a pattern based on colors and sounds, the sounds must thus be well designed to be easy to follow. When user hears an attractive pattern, naturally they are inclined to repeat it.
  - Originator: Shady Nessim
  - Fit Criterion: User shall be invested in game and spend a lot of time playing the game.
  - Priority: High
  - History: Created October 5, 2016

4.1.2 Style Requirements

- Requirement #4
  - Description: The product shall have enough buttons to keep game engaging but not too many as to make the screen feel cluttered. DNA Says will appear to be a bright upbeat game.
  - Rationale: The game must induce a style and feel to the user that will be a driving factor to use the game if the user likes the style of the game
  - Originator: Shady Nessim
  - Fit Criterion: Stakeholder satisfaction regarding the style, user attraction to game.
  - Priority: Medium
  - History: Created October 5, 2016

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

- Requirement #5
  - Description: The product shall be easy to use for people of all ages, including children.
  - Rationale: The game involves no reading or writing, it does not involve intelligence either. The game involves short term memory. As such it should be easy to use for all people to improve their short term memory.
  - Originator: Shady Nessim
  - Fit Criterion: User figures out how to play the game within the first couple of minutes of use.
  - Priority: High
  - History: Created October 5, 2016
- Requirement #6
  - Description: The product shall be easy to install for all users.

- Rationale: This product is simply a game so the user will probably not go through the trouble of downloading and installing the game if it is not an easy process.
- Originator: Shady Nessim
- Fit Criterion: User easily downloads and installs the game in a timely manner.
- Priority: High
- History: Created October 5, 2016

4.2.2 Personalization and Internationalization Requirements

- Requirement #7
  - Description: The product shall operate with the English language.
  - Rationale: The application is intended for use by English and non-English speakers, however with minimal required text use, this game can easily be figured out and used by non-English speakers
  - Originator: Shady Nessim
  - Fit Criterion: User easily understands objective of game and how to play.
  - Priority: Medium
  - History: Created October 5, 2016

4.2.3 Learning Requirements

- Requirement #8
  - Description: The application shall not require a tutorial and shall be clear and simple enough in early levels to communicate to the user how the game is played.
  - Rationale: The application is intended for use by people of all ages. Must thus be easy to understand.
  - Originator: Shady Nessim
  - Fit Criterion: User easily understands objective of game and how to play.

  - Priority: Medium
  - History: Created October 5, 2016

4.2.4 Understandability and Politeness Requirements

- Requirement #9
  - Description: The application shall not produce ugly sound patterns or offensive visual patterns to respect all users.
  - Rationale: The application is intended for entertainment and as a cure for boredom, if user feels uncomfortable or offended they will not use the game.
  - Originator: Shady Nessim
  - Fit Criterion: User feels good about game and patterns are appealing and attractive.
  - Priority: Medium
  - History: Created October 5, 2016
- Requirement #10
  - Description: The product shall produce a friendly indication when user loses or wins a level.
  - Rationale: The application is intended for entertainment and as a cure for boredom, if user feels uncomfortable or offended they will not use the game.
  - Originator: Shady Nessim
  - Fit Criterion: User feels good about level progression and is encouraged to play again.
  - Priority: Medium
  - History: Created October 5, 2016

4.2.5 Accessibility Requirements

- Requirement #11

  - Description: The product shall produce patterns both visually and auditory so as to accommodate for users with visual or auditory problems that they can use an alternative pattern means.
  - Rationale: The application is intended all users, should be easy to use for someone by just following visual patterns or just following auditory patterns.
  - Originator: Shady Nessim
  - Fit Criterion: User with auditory or visual problems feel comfortable playing the game.
  - Priority: Medium
  - History: Created October 5, 2016

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

- Requirement #12
  - Description: The application should be able to recognize whether the user has entered the right pattern as soon as they finish pressing the last button.
  - Rationale: The user should not have to wait for the application to calculate whether their input was correct or not.
  - Originator: Shady Nessim
  - Fit Criterion: Application should respond immediately to user input and the upcoming pattern should start soon after user input ends.
  - Priority: High
  - History: Created October 5, 2016

4.3.2 Safety Critical Requirements

There are none applicable to this project.

4.3.3 Precision of Accuracy Requirements

- Requirement #13
  - Description: The application must be specific to each button press. Button press confusion or mistake must not be tolerated.
  - Rationale: The purpose of the game is to produce exact same pattern shown by application. If program does not detect a mistake even if it is just one wrong button, then that defeats the fairness and purpose of the game.
  - Originator: Shady Nessim
  - Fit Criterion: Application should perceive and evaluate user pattern input impeccably.
  - Priority: High
  - History: Created October 5, 2016

4.3.4 Reliability and Availability Requirements

- Requirement #14
  - Description: The application must be available at all times.
  - Rationale: The purpose of the game is to defeat boredom which may come at any time and thus the game must be available at all times.
  - Originator: Shady Nessim
  - Fit Criterion: User should be able to play the game whenever they are bored.
  - Priority: High
  - History: Created October 5, 2016

4.3.5 Capacity Requirements

- Requirement #15
  - Description: The application must be able to produce and receive patterns as long as 25 buttons.

  - Rationale: In order to make the game challenging enough, patterns including but not limited to 25 in length should be produced and received by program.
  - Originator: Shady Nessim
  - Fit Criterion: User should be able to reach level 25 in each mode.
  - Priority: Medium
  - History: Created October 5, 2016

4.3.6 Scalability Requirements

There are none applicable to the project.

4.3.7 Longevity Requirements

There are none applicable to the project.

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

- Requirement #16
  - Description: The product should be able to be used on laptops and desktops.
  - Rationale: The clients will use the product from these devices.
  - Originator: Shady Nessim
  - Fit Criterion: User should be able to run the game on any laptop or desktop.
  - Priority: High
  - History: Created October 8, 2016

4.4.2 Release Requirements

- Requirement #17

  - Description: The product will be revised yearly and updated according to changing demands and needs of the client. The product will undergo maintenance upon realization of any errors in gameplay behavior.
  - Rationale: The game has to stay updated and problems have to be handled in order to maintain user interest and usage.
  - Originator: Shady Nessim
  - Fit Criterion: App should be updated at least annually.
  - Priority: Medium
  - History: Created October 8, 2016

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirement

- Requirement #18
  - Description: The source code for the application shall be visible to the public.
  - Rationale: This enhances the ability to monitor and maintain the system.
  - Originator: Shady Nessim
  - Fit Criterion: Source code is available in a public repository.
  - Priority: Low
  - History: Created October 10, 2016

4.5.2 Supportability Requirements

None applicable for this project.

4.5.3 Adaptability Requirements

- Requirement #19
  - Description: The product shall run on Windows, Linux and Mac OS X environments.

- Rationale: The users may be using any of these operating systems.
- Originator: Shady Nessim
- Fit Criterion: The product works on listed platforms in the test groups.
- Priority: Medium
- History: Created October 10, 2016

4.6 Security Requirements

4.6.1 Privacy Requirements

- Requirement #20
  - Description: The application shall not store, transmit, or upload any user data.
  - Rationale: This is required in order to protect the privacy of users.
  - Originator: Shady Nessim
  - Fit Criterion: No functionality to perform these tasks is implemented in the application.
  - Priority: Low
  - History: Created October 10, 2016

4.7 Cultural Requirements

- Requirement #21
  - Description: The application shall not contain any imagery or text that can be reasonably foreseen as potentially oensive to users of all cultures, backgrounds and ethnicities.
  - Rationale: User satisfaction will be greatly reduced if they notice any offensive patterns in the game.
  - Originator: Shady Nessim
  - Fit Criterion: Application does not contain offensive patterns or references.
  - Priority: Medium
  - History: Created October 10, 2016

4.8 Legal Requirements

There are none applicable to this project.

5 Project Issues

5.1 Open Issues

There has been no open issues in the duration of this project. The undocumented code has been carefully anlyzed and each line has been assessed and a solid understanding of the program has been gained.

5.2 Off-the-Shelf Solutions

In general, there are many games that share the similar nature and purpose. However, with respect to Simon Says, there is the original oral game in which one designated person speaks out an action for the others to do, having them only do the action if they say ?Simon says? before it. As for digital versions, there exists multiple ones online.

5.3 New Problems

The only problem that could arise from this project is addiction. As there could be individuals that instead of enjoying this game during their free time or as a short break from their schedule, they would consume their other priorities? time to play. This game could potentially be the reason behind a missed deadline, or anything in that manner.

5.4 Tasks

All tasks that need to be accomplished are covered within this group's Gantt Chart, including their start and end dates found here:

- Gantt Chart

5.5 Migration to the New Product

There will not be issues transferring from another version of this game to this current version. Only the user's preferences matter for this, and that will be discussed as a risk in the section below.

5.6 Risks

The initial risk has been that the user's preferences might conflict with the team's preferences that the game was built upon. That risk has been taken into consideration and to counter it, the team has developed a user survey. That survey was taken by various users that played the game and their opinions were analyzed and the game has been updated accordingly.

5.7 Costs

There are no monetary costs included in this project.

5.8 User Documentation and Training

Instructions are always available at the bottom left corner of the screen. The game is very simple and does not require more than one line of explanation per mode.

5.9 Waiting Room

At this point, the team is improving all documentation for the next round of marking.

5.10 Ideas for Solutions

To make sure that most of the users will enjoy this game, a survey was conducted to collect different thoughts and preferences as to what the users would like to see in this game, what they are looking forward to, and what they expect. That survey included the desired colors, sounds, interface and functionality. The current implementation of this project is modified to suit those preferences.

6 Appendix

This section contains no related information for this document.

6.1 Symbolic Parameters

N represents any integer and does not have an upper bound. It is used throughout the document to represent level numbers, the number of elements in a given pattern and the score.