

AlgoSenpai - Developer Guide

Table of Contents

1. User Profile	1
1.1. Target User Profile	1
2. Value Proposition	1
Appendix A: Extras	1
A.1. User Stories	1
A.2. Use cases	5
A.3. Non-Functional Requirements	6

1. User Profile

1.1. Target User Profile

- Wants to improve concepts in data structures and algorithms
- Wants to become faster at answering questions
- Dislikes learning concepts theoretically
- Prefers elements of fun in their learning
- Finds VisuAlgo boring, repetitive, and tedious to use

2. Value Proposition

Offer a more time efficient, and fun platform to learn and practice data structures and algorithms.

Appendix A: Extras

A.1. User Stories

Priorities: High (must have) - * * *, Medium (nice to have) - * *, Low (unlikely to have) - *

Priority	As a ...	I want to ...	So that I can...
* * *	New User	see a summary of the commands available in the program	Have a brief idea of how to navigate through the program

Priority	As a ...	I want to ...	So that I can...
* * *	New User	Start the quiz	Start learning as soon as possible
* * *	New User	Pause the quiz	To pause the game and save the current quiz records
* * *	New User	End the quiz	See the results for the current quiz
* * *	New User	Go back to the previous question	Review the question and change my answer if necessary
* * *	New User	Skip to the next question of the quiz	Move on to other questions that I am more confident in to avoid spending too much time on a single question
* * *	New User	See the results of the previous quizzes I attempted	Can see a tangible improvement in my quiz scores
* * *	Weak User	Choose to attempt a quiz on a certain chapter	Work on my weaker areas
* * *	New User	Continue the quiz which I have previously ended off with	Save time on trying to find out where I last ended off
* * *	User	Have an undo button	I can reverse my actions if I typed the wrong command

Priority	As a ...	I want to ...	So that I can...
* *	User	Share and spread this game to my friends	Invite them to learn together
* *	Average User	Get feedback on my performance	Identify my common misconceptions and work on those areas
* *	User	Provide feedback to developers	Give them suggestions on what they can improve on
* *	User	Receive notifications and reminders to complete a few quizzes a day	Be more consistent with my learning
* *	User	Experience a storyline	Learn in a less boring manner
* *	Weak User	Have visual aids for the questions	Understand the topic better
* *	Weak User	Have an explanation for any wrong answers	Learn faster from my mistakes
* *	Strong User	Have a time limit	Challenge myself to perform better
* *	Strong User	Have an arcade mode	Challenge myself to see how proficient I am in a topic
* *	Weak User	Have a multiple choice option rather than open-ended	Practice on my concepts before attempting harder questions

Priority	As a ...	I want to ...	So that I can...
* *	User	Have a reset option	Replay the game if I'm done with the storyline.
*	Weak User	Clarify certain concepts with a virtual agent	Build a stronger foundation
*	User	Invite friends to join me in a multiplayer quiz	Challenge them while progressing together
*	Advanced User	Get updates whenever the program has new levels	Keep up with the game and maintain my rank
*	Advanced User	Get updates whenever another user overtook me	Challenge myself to beat them
*	User	Customise my own questions	Test myself on questions I know I'm weak at.
* * *	Tutor	Track my students' progress	Cater my teaching style according to the weak chapters.
* *	Tutor	See which students are in dire need of help	Focus more attention on weaker students
*	Tutor	Set my own questions	Test my students' understanding on the current topic
*	Tutor	Set my own storyline	Make assignments more enjoyable.

A.2. Use cases

(For all use cases below, the System is **AlgoSenpai** and the Actor is the **User**, unless specified otherwise)

UC01: Play through a story

MSS:

1. User launches the game
2. System starts and displays a welcome message
3. User requests for the list of available stories
4. System displays the list of stories
5. User chooses a story to play
6. System starts displaying questions from the story
7. System displays the next question
8. User enters this answer
9. Steps 7 - 8 repeat until the game is over
10. System shows the result and returns to the main menu

Use case ends.

Extensions

- 8a. User enters an invalid command/answer.
 1. System shows an error message and prompts the user to input a valid command
 2. User inputs a new command
 3. Steps 1&2 repeat until the user has entered a valid command
 4. System resumes from step 7
- 8b. User enters a command (pause,undo,back etc)
 1. System executes the command
 2. System resumes from step 7

UC02: Get Help

1. User enters the **help** command
2. System shows a list of valid commands

Use case ends.

UC03: See previous results

1. User enters the **history** command
2. System shows the user's previous results

Use case ends.

UC04: Share the game

1. User enters the **share** command
2. System asks user to pick their choice of social media platform
3. User chooses one
4. System opens a browser window through which the user can share their results in the game on the social media platform.

Use case ends.

A.3. Non-Functional Requirements

- Should work on any mainstream OS as long it has Java 11 or above installed.
- A user with above average typing speed for regular English text (i.e. not code, not system admin commands) should be able to accomplish most of the tasks faster using commands than using the mouse.
- The randomly generated questions should have a consistent level of difficulty