Parsons problem generator and solver

P1908345 Jane

Contents

Introduction

Challenges

Problem Analysis

Solution Design

Implementation Strategy

System Demo

Conclusion

Introduction

Parsons problem is a type of programming question to let students **drag and drop to reorder the unique predefined solutions.**

This convenient study tool only is used in introductory programming course (CS1), but not in Data Structures and Algorithms (CS2).

This project aims to build a web application to expand Parsons problem in CS2.

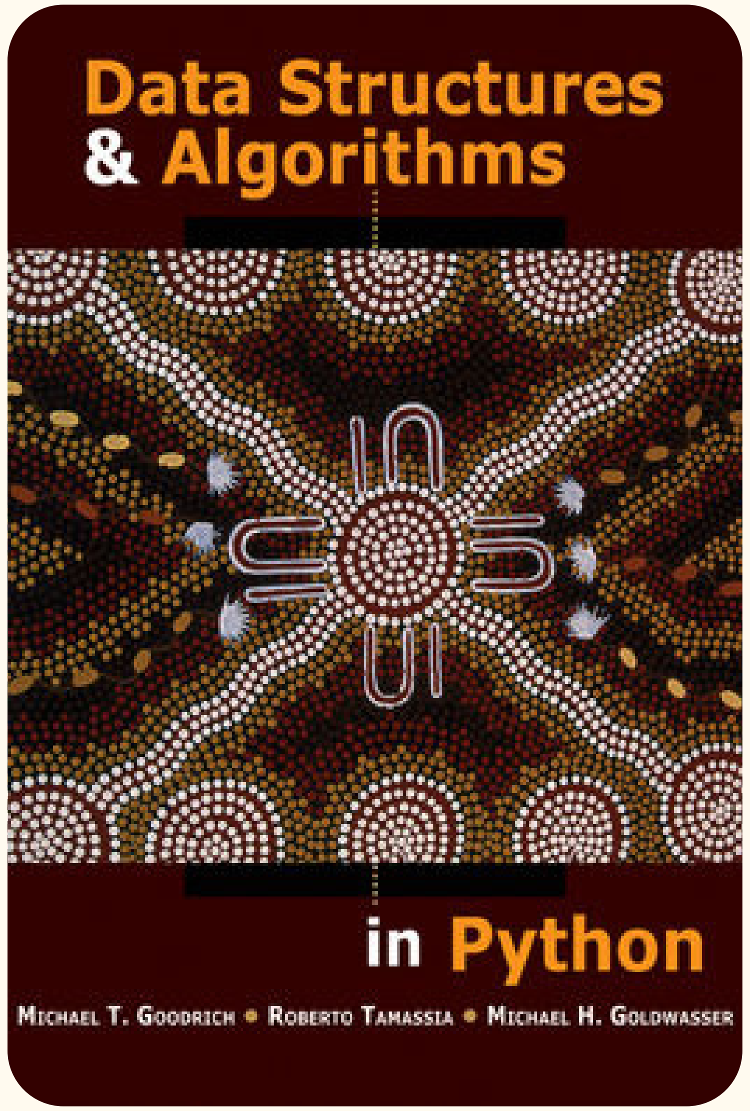
Challenges

It needs to study intensively the concrete exercises in CS2 to figure out the possible amelioration of the existing Parsons problem.

It needs to design and develop proper Parsons problems to fit the exercises in CS2.

Last challenge is to implement a 2D view for dragging and dropping the code blocks.

Problem Analysis


Data Structures & Algorithms
in Python
Michael T. Goodrich • Roberto Tamassia • Michael H. Goldwasser

It seems that the traditional Parsons problem can be applied to CS2 by only changing the input exercise instead of changing the structure of Parsons problem itself. However, it is not the truth.

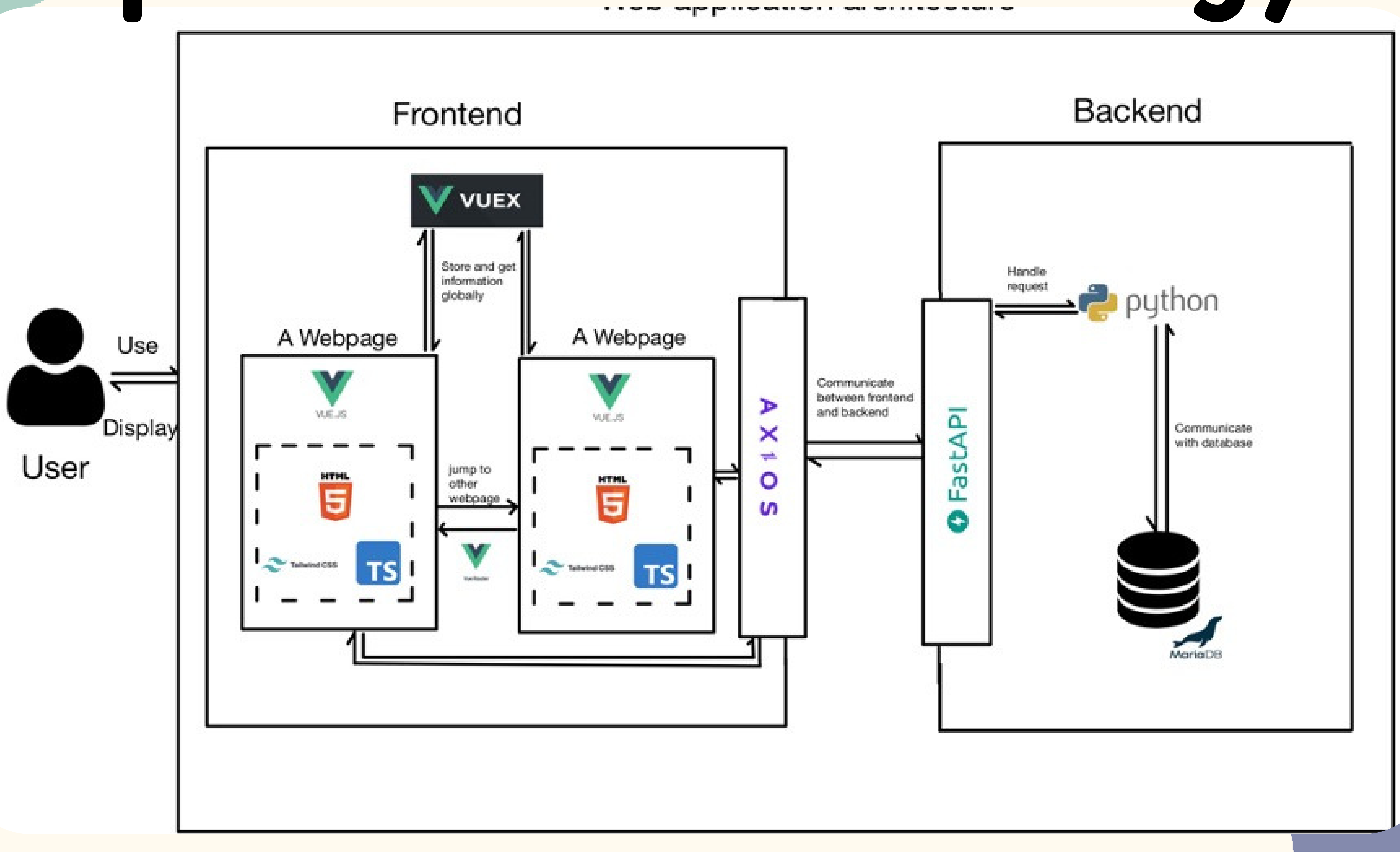
To show the reasons why not directly using it and explore the possible improvement of the traditional Parsons problem, the exercises in the famous textbook Data Structures and Algorithms in Python were analyzed. Among these exercises, three question types showing significant differences between CS1 and CS2 were identified:

- Object-oriented programming
- Recursion
- Comparison

Solution Design

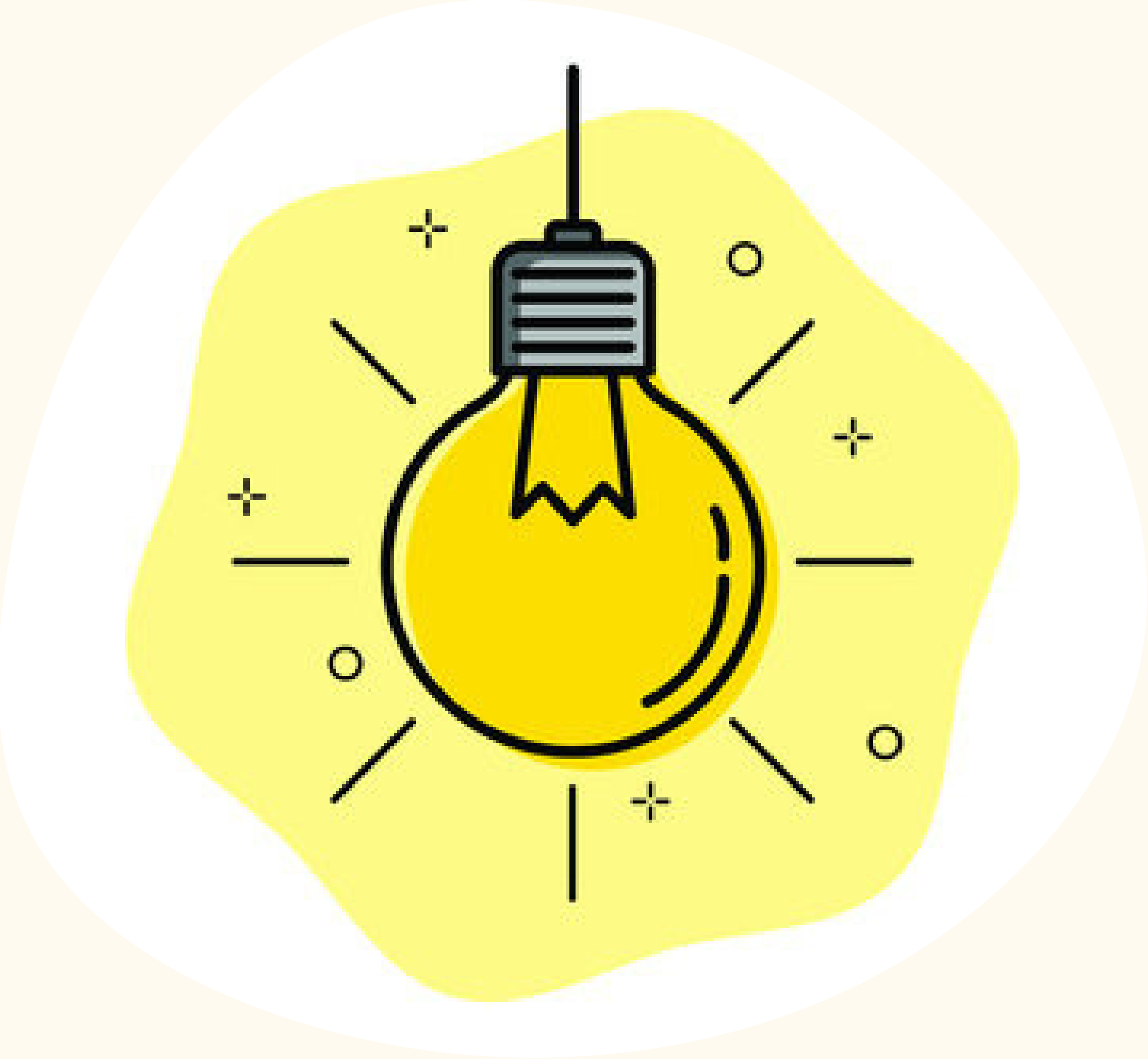
Problem	Why do we need to solve it?	Solution
Switching positions of methods is allowed in object-oriented programing.	It may cause multiple correct answers and have problems when comparing with the unique predefined answer	'Context' type of question is used to fix the order of methods to have unique correct answer
Recursion problem is difficult for students to understand.	Providing code fragments to read is not as effective as in CS1. It is almost like letting students write codes directly.	'Multiple step' type of question is used to provide a common idea to solve recursion problem.
There are a lot of similar or related codes in CS2.	It is worthwhile for students to distinguish these codes, and prevent students from using them mostly.	'Comparison' type of question is used to compare similar or related codes.
The difficulty level has been improved significantly in CS2	Students need more help when they do the exercise.	Multiple ways including inserting key code, pre-scaffold, and switching difficulty level are provided.

Implementation Strategy



System Demo

Home Page



Conclusion

The project's main contribution is:

- Provide an online platform for Parosn problem
- Improve students' engagement and reduce students' cognitive load
- Help students to consolidate knowledge

In the future,

- Further experiments will be conducted to show the effectiveness of this tool
- Expand survey to more students

Thank you!