Assignment 2:First Step in Building Modern Software

Group number: 42

Team members: Zijia(Sonny) Chen, Hongshou(Caules) Ge

Github Repository: https://github.com/csc301-fall-2022/assignment-2-42-sonnyffff-caulesge

Link to web app: https://checkoutsonnycaules.herokuapp.com/

UI/Frontend **1**
- React 1
- Vue 2
- HTML 2

Logic/Backend **2**
- Python 2
- Java 2
- C/C++ 2

Database **3**
- SQL server 3
- Oracle 3
- Flask 3

Summary of final decisions **3**

Appendix **4**
- Work Distribution 4

UI/Frontend

React

Pros: React is one of the most popular front-end frameworks in the industry that can develop UI for both website and mobile applications. From my previous experience, using React Components with JSX can create a multipage website conveniently. It is suitable for creating complex projects with many states.
Cons: Not friendly to people who have no JavaScript knowledge. Meanwhile, if we do not use JavaScript for the backend, it is hard to connect them: we need to use axios and implement a database, which is not necessary for this small-scale application

Vue

Pros: Vue is also one of the most popular front-end frameworks in the industry. Compared to React, it is easier to learn and get started.
Cons:Compared to React, it takes longer when the project contains many states. Similarly, connecting it with a non-JavaScript backend is complex.

HTML

Pros: HTML is very easy to learn and implement. It is suitable for a single page website. Besides, if we use Python to build the backend, we can use Django or Flask to show data on HTML easily.
Cons: Methods are very limited and can not be suitable for projects with many pages and complex methods.It cannot produce dynamic output alone, since it's a static language.

Logic/Backend

Python

Pros: Python is object-oriented and it contains rich and powerful libraries that makes Python code easy to extend and reuse. It is also easy for python to link various modules made in other languages such as C and Javascript. Last but not least, python has a simple and concise syntax that is easy to read and write.
Cons: Python execution is slow, compared to compiled languages like C and Java. Because of python's GIL(Global Interpreter lock), Python is poor at multithreading.

Java

Pros: Java is an object-oriented programming language that can write cross-platform applications and software. It has many variations that responsible for different platform, for desktop applications there is Java SE, for web apps there is Java EE, and Java ME for mobile applications
Cons: Code execution is relatively slower than C since it relies on Java virtual machines that do not directly execute machine code. Java is also memory consuming, Java's garbage collection mechanism helps to avoid memory leaks but reduces performance significantly.

C/C++

Pros: Since the running speed of the code generated by the C language is almost the same as the running speed of the code written in the assembly language, the C language is used as the system development language. C++ is the extension of the C language. Besides C's original features, C++ as an object-oriented language supports data abstraction, Object-oriented programming, principle-based design and other programming styles.
Cons: C is an imperative procedural language. It has weak data encapsulation, and it directly accesses memory that may lead to security problems. Unlike object oriented language, C code

is hard to reuse and makes extensions. C/C++ also has relatively difficult syntax which may take more time to learn.

Database

When we are choosing a database, we need to consider the target subject, functionality, stability and security and supporting operating system.

SQL server

Pros: SQL server has a graphical user interface that makes database management more intuitive. Furthermore, it has good scalability and can be used across platforms. It has a client-server architecture. A request is sent to MS SQL Server by the client application. The SQL Server accepts, processes, and returns the processed data in response to the request.
Cons: It can only run on Microsoft Windows. SQL Server performance will become very poor when there are many user connections, so it is not stable enough.

Oracle

Pros: Oracle runs on all popular operating systems and it is fully backward compatible. It has the highest level of certification of ISO standards, so it has a high security level. It also has a high performance and it is tested for a long time with low risks.
Cons: High hardware requirements. Management and maintenance is difficult and expensive. Requires high technical background because of the complicated operations.

Flask

Pros: Flask is considered the best framework for light web application serving, it is easy to learn and easy to implement. It can render HTML so we can use it to connect Python backend to the HTML implemented frontend easily.
Cons:Developing and maintaining a very large-scale application using Flask can be a complicated task. Meanwhile, It comes with a limited set of tools and functionalities.

Summary of final decisions

We first analyze the problem and decide our use cases to be adding to/removing from the shopping cart and checking out with possible tax and discount. We recognized that items would be the main entity in the program so we define that as an individual class. And in order to decouple frontend and back and follow SOLID principle, we created a counter class that is responsible for all calculation and instruction handling. So that interface should only depend on the method provided by the backend i.e. counter.
For the frontend we decided to use HTML and CSS. The reason that we do not use React or Vue is that this is just a light scale single page application and there is no need to use them. Besides, we have already decided to use Python for the backend so it is more convenient to use

Python Flask to render HTML.The HTML contains a form that can support the POST request which can send the input in the textbox to the backend.
For the backend we finally decided to use Python, the first reason is that Python is an object-oriented language which is suitable for the scope of this project. Another reason to choose Python is that Python as an interpreted language is easier to debug and test since it saves compilation time. Also for personal reasons, since we have done some projects with Python before, so we have more experience with it.
For the database, we decided to use Flask because it is based on Python and suitable for light scale projects. In this project we do not need to store much data or use complex data methods, we only need to save the data in the counter implemented in the backend..

Appendix

Work Distribution

Zijia (Sonny) Chen: Backend implementation and testing. Description and instructions on the backend. Backend and database in report.

Hongshou(Caules) Ge: Developed the frontend with HTML/CSS and used Python Flask to connect the backend to the frontend. Finished the frontend part of the report.