CS551Q SOLO Project Report

URL: dawanlamian.pythonanywhere.com

This project uses Django and Python to develop a database-driven online mall, simulating real-life e-commerce websites, such as Amazon. The report documents the design, development, implementation, installation, and use of the application, highlighting key aspects of each stage.

The software architecture of the application is designed using the MVC pattern and has three units: model, view and controller. The models.py file is created in the model unit, including the different parts of products, users, shopping carts, and orders. In order to create the database, xlrd is used to read the data in the data set and generate the database. Since this project uses a lot of view functions, a view folder is created and the view functions are classified according to different functions. All required html files are stored under the template folder, which is divided into registration part and shopping part according to functions.

The main functions implemented by this project are as follows. Visitors can browse the product list and add selected products to the shopping cart, but they cannot pay. If payment is required, they must register. Registered users can browse and purchase products normally, but neither visitors nor users are allowed to edit other user and product information. Administrators have the highest authority and can add or delete products, manage users, and edit user information. I adopted an Agile methodology, dividing tasks into sprints. For example, first implement the database establishment, then design the user login and logout functions, then the product list, order list functions, and finally the shopping cart function and payment function. By splitting tasks, the speed of development can be significantly improved.

The main difficulty of this project is that the logic is relatively complex, and users with different permissions need to implement different functions. In order to ensure security, some administrator functions cannot be opened to users, which made me very confused during the development process. In the subsequent development process, more attention should be paid to early demand analysis and functional design to avoid confusion during the development process.

This project uses behave for testing, and checks whether the website functions normally by simulating user behavior, including simulating users to log in, simulating adding products, and checking whether the products are added successfully.Throughout the development process, I used GitHub for version control and conducted regular code reviews to track changes and maintain the code. The project's repository is published on Pytonanywhere for everyone to use