We divided the implementation process into three parts, front-end, middle layer, and back-end: the front-end system is a graphical user interface for users to use the Eatery platform more easily; the back-end system, which is not accessible to users, is mainly the implementation of the interface. middle layer, as the front and back-end data encapsulation and filter implementation.

**Front-end**

In front-end development, we mainly use HTML, CSS, JavaScript, and Webpack as the basic tools and technologies.

HTML (HyperText Markup Language) is the standard markup language for creating web pages. We use HTML to build the structure of a web page, defining the relationship of elements and content.

CSS (Cascading Style Sheets) is a language that describes how HTML elements are displayed on the screen. We use CSS to design and beautify pages and enhance the user experience.

JavaScript is a scripting language that is used to create dynamic web pages that increase interactivity. In this project, we use JavaScript for client-side programming to handle user interactions, such as clicking buttons or submitting forms.

Webpack is a module packager that is primarily used for JavaScript applications. We use Webpack to package all the modules into one or more static resources so that they can be loaded and run more efficiently.

We use react, a JavaScript library developed by Facebook, which is used to build user interfaces. We use React to build componentized user interfaces, provide faster page rendering, and deliver a better user experience.

Material-UI (Mui) is a UI framework for React that provides a rich and customizable set of UI components. We use Mui to design and implement user interfaces, reducing development effort while ensuring consistency and aesthetics.

Redux is a JavaScript state management library that is often used in conjunction with React. We use Redux to manage the state of our applications, providing a predictable state management model that makes it easier to track and manage changes in the state of our applications.

Tailwind CSS is a pragmatic CSS framework that provides an underlying, composable set of CSS classes for building customizable designs. We use Tailwind CSS to write styles, and its composability allows us to build interfaces more quickly and flexibly.

These are the main tools and techniques we use in the front-end section, which work together to provide a beautiful, easy-to-use and responsive interface for our users.

**Middle layer**

In the middle layer, we use Sanity, a real-time content infrastructure implemented on Node.js. Sanity provides a customizable content management system (CMS) and powerful APIs for data query and mutation. it is ideal for building, managing and delivering content to anywhere. It acts as a bridge between the front-end and back-end, taking requests from the front-end, processing the data, and returning the results to the front-end.

**Back-end**

For our back-end implementation, we primarily use Python as our development language, coupled with various database technologies for data storage and management. Our focus lies in crafting the back-end logic, handling user requests, and interacting with the database among other tasks.

To optimize our development process, we've opted to use Django, a Python framework. Django is a comprehensive and robust framework that includes most modules needed for web development, allowing us to quickly establish a complete web service.

**Database**

For the database, we chose MySQL, which is used to store and manage data so that when a user or system requests a service, we can retrieve the data from the database and process it as needed.