Technologies

We have chosen **Python** as the main programming language for our backend application because we are familiar with it, it is compatible with multiple platforms and systems, the code is readable and the standard library comes with a wide range of in-built modules that help us write our code (like the json module, for instance).

**Flask** is a very popular light web framework for Python. It’s a simple tool that helps you build a web application, making development faster by offering in-built methods for interacting with a database, URL parameters, files, and more.

We used **HTML & CSS** for creating the design of our web application because of their simplicity and homogeneity with Python and Flask framework.

**Javascript** was also used to make our web pages dynamic and interactive, by implementing custom client-side scripts.

APIs

**Google Cloud Source Repositories** was an excellent choice for a place to develop and securely manage our code. Here, we created a common repository in order to always have our code updated and well organized. We collaborated easily and it was effortless to connect to other Google Cloud tools like App Engine, Datastore, and Logging.

**Google Places** was essential for our application since the orders presented to the user are the ones closest to him, and it is also used for the user to get nearby markets’ location.

**Google App Engine** is the platform used to host our application. It is a Platform as a Service (PaaS) product that provides us with easy system administration and deployment commands.

**Google Cloud Datastore** is utilized to host our database entities. It is NoSQL, highly scalable, and easy to manage from the Cloud Platform.

**Google Sign-in** was a facile method of implementing the registration and login of our users. It manages the OAuth 2.0 flow and token lifecycle, and it is easy to integrate with Google APIs.

**Google Geocoding** is a service that provides geocoding and the possibility to get a user’s physical address based on latitude and longitude.

**Google Cloud Functions** is the platform used to maintain the code for our functions that make calls to Google Places API and Geocoding API. It is a serverless, event-driven solution that worked best for our project.