We chose to use Python Flask as the back-end of our project architecture because it is a good choice for smaller projects, is very straight-forward and easy to use, and allows for a high degree of modularity in its code structure. Flask supports convenient extensions to simplify forms and form validation, session handling, and database management, so it was a good choice for an app that will be providing login, search, and preference storage services. We will be using an SQL database, through the SQL-Alchemy extension for Flask. This will provide us with easy, reliable, and safe access to a database through an ORM layer. In addition, the wide variety of high quality Python libraries readily available makes Flask a very powerful framework. The front-end will use HTML/JavaScript because these are conventional, effective ways at interpreting and displaying data for the user.

The other architecture that we looked at was PHP. We decided not to use PHP because it is less developer-friendly than Python and more prone to bugs, edge cases, and security vulnerabilities. PHP is most commonly connected to a MySQL database, so there also a bit less flexibility in this regard. Also, PHP code is often highly connected to the markup of the pages, making it a lot less modular than Flask, and much harder to organize.