Charlie McClatchy
Mcclatc4
CSC301 A1 mobile app report

In this assignment our goal was to create a checkout price calculator with basic functionality. This calculator would correct for tax and any possible deals as well as allow for the users to add items themselves for a fully customizable checkout experience. For this section our task was to develop a mobile app version of the assignment. While this was similar in effect to the web app version, producing this checkout calculator for mobile results in its own particular series of challenges and choices to make.

This project includes technologies in the following tech stack. Examples of various solutions as well as our assignment selection are given for each. Finally in each category a considerations block is added to explain the reasoning for picking our particular technologies.

Frontend:

Possibilities: Angular, React native, Swift
Chosen: React Native
Considerations:

Many considerations went into choosing which frontend framework and language to go with. When creating a mobile app you have to start with the choice of what operating system you would like to run on. We decided it would be best for our app to be able to run one both ios and android, as this would be the most convenient for any customers looking to use our product. This means that while both Angular and React Native would be applicable, Swift would not be a good option for us as it only works on ios. With this is mind we also considered the domains covered by these languages. All three of these options have mobile app frontend development in their domains, with Swift being the most specific and narrowed in option. As a last consideration we thought about the popularity and maturity of the language. Here we have to give the edge to Angular and React Native again because javascript is the most popular technology among developers in 2019 according to stackoverflow and the language is very mature with a plethora of libraries.

Backend:

Possibilities:Parse, Back4app, Django
Chosen: Parse
Considerations:

Choosing the language and framework for the backend is a very important task which comes hand in hand with choosing the frontend. As you will see many of these decisions come from the knowledge that we will be using React Native as our actual frontend technology. In this category there are conflicting considerations for using the different options and it will be shown that there is no true right answer. For the first consideration we can say that Django will be the easiest for me to use by far, as I have lots of Python experience but little javascript experience. On the other hand we have the domain of the language and the libraries available. While Python can be used for backend app development, that is not in its primary domain

whereas javascript is the go to language for this kind of thing. Moreover, having the whole frontend and backend in various javascript frameworks makes things much cleaner and easier for development.

Testing infrastructure:

Possibilities: Jest, Mocha, Jasmine

Chosen: Jest

Considerations:

A good testing framework is vital in the production of good code, as time wasted on inefficient bug testing and error reports might as well be wasted time if you don't have proper unit tests. Jest, Mocha, and Jasmine are all very common javascript testing frameworks and are all quite useful. We chose only javascript testing frameworks because both our frontend and backend are in javascript. Our first consideration for this section is popularity of the software. This is because with something straightforward like this it makes sense that the common and frequently used frameworks will be quite useful for general purposes. This is why we chose Jest as it is the default unit testing framework for React Native.

Database:

Possibilities: SQLite, MongoDB, MySQL

Chosen: MongoDB

Considerations:

For our last group of considerations we chose our database once we had already made all of the prior choices. This left us with less room for the database and for the proper choice to functionally fall into place. Because we chose Parse as our backend we were required to go with MongoDB, as is stated by the Parse website. However, there are still possible considerations to be made for the database in our assignment. While these are all well respected and highly effective database options, sqlite is most considered a choice for mobile development and local data storage.