Website		Mobile	
Pros	Cons	Pros	Cons
● Cross-Platform compatibility ● Independent of which operating system is the user using ● Independent of version of environment the user is using ● The user's data is stored in a cloud server, which is more secured and in sync. ● Low development cost	● Require stable internet	● Can have a local database, independent of the internet ● Back-end programming languages have better performance	● Compatibility across all platforms ● Compatibility across all versions of systems ● The APP store may have specific regulations

Front-end		
Vue	React	Angular
● Best rated on git hub repository ● A teammate has previous experience ● Most intuitive ● UI and behavior are perfectly binded ● Highly customizable ● Two-way binding simplifies the codes, the state changes are synced. ● Good runtime performance ● Fast compile and runtime on DOM	● Most-used ● Good for large scale project ● Difficult to learn	● Size of library is the largest ● Longer loading time

Back-end		
Springboot (java)	Express(js)	Flask(Python)
• Using java, a well known, static, and stable language • IoC, AoP • Flexible dependence injection • Great efficiency • Large community and easy to get help	• Connected with front-end easier since it is using a front end language	• Lack of database support • Require more learning about the previous framework • Maintenance cost high

CI/CD		
Travis CI	CircleCI	Github Action
• Created earlier • Only free for open-source projects	• Runtime problem • Difficult to learn • Documentation is not easy to understand	• UI is easy to use • Since the codes are uploaded to Github, this tool is very convenient to use

Database		
PostgreSQL	MySQL	MongoDB
• A relational database management system that supports more data types • monolithic architecture • Uses SQL • Work well with a high number of client connections • supports materialized views • Have better performance for complex queries	• purely relational • support plans from Oracle are paid	• document database • distributed architecture • Uses BSON