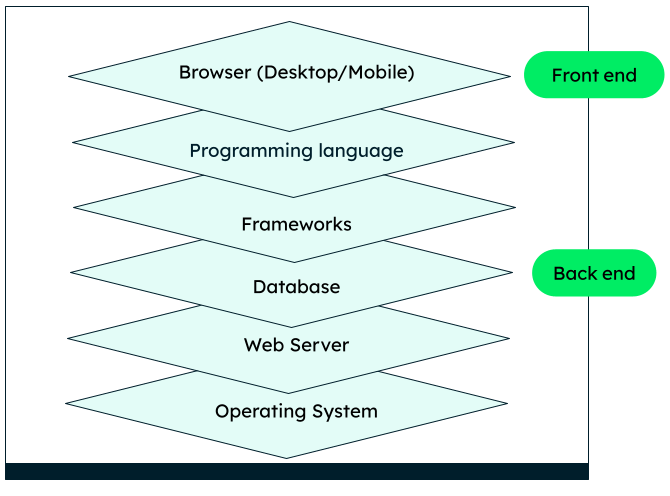
**Samsung**

Tech stack – Samsung pet app

Featured snippet from the web

A tech stack is the combination of technologies a company uses to build and run an application or project. Sometimes called a “solutions stack,” a tech stack typically consists of programming languages, frameworks, a database, front-end tools, back-end tools, and applications connected via APIs.



The application they plan on building should be fast and provide an interactive browsing experience for users. If users seek more information, the application must be able to quickly fetch and organize the information in an easy-to-view manner. Further, as your application grows in popularity, it should scale well, without adding any performance overhead to the application ecosystem.

**Back-end:**

Samsung uses Amazon Aurora as their database. 325 companies use Amazon Aurora as it is difficult and expensive to create your own database. 992.4 million Samsung phones were active in 2020. This means that the database needs to be huge in order to make every phone with an app with close to zero flaws and make sure it’s secure.

Samsung also uses 142 technologies for its compatibility in order to share Samsung Features across multiple platforms which are used by other technologies that Samsung owns. Samsung do this to make sure that if you own 2 different technologies they will still work.

The official language for Android development is Java. Large parts of Android are written in Java and its APIs are designed to be called primarily from Java.

For security, they can use the WPA3 protocol as it is the most secure wireless protocol as of now, therefore it should be implemented in the app, so it becomes a secure application for users to use.

**Front-end:**

Samsung uses 41 technology products and services including HTML5, Google Analytics, and jQuery, according to G2 Stack. This is used to build the website and other various features to make the website easy to read and user friendly.

Given java is the official programming language for most things android for the sake of consistency as well as compatibility we would use java for the programming of both front and back end.

The applications in the UI depend on the users and get verified using the play store by publishing their application using java as the main language requirement.

Both:

The Galaxy Store Developer API is a set of server-to-server APIs that allow you to manage apps, manage in-app items, and check app statistics. The set of APIs includes the Content Publish API, IAP Publish API, and GSS Metric and User APIs. We would use this to monitor the app and see how it's performing

Plan for implementation:

* Launch as BETA version to let customers test and see how the app is
* Use feedback to update the application
* Launch app to everyone
* Once the app is made with no flaws, make it as a default app for when a new phone is bought