Why Flutter?

Flutter is an open-source created by Google for developing mobile applications.

How is Flutter used?

This platform was created for helping developers while building a single app both on iOS and Android at the same time. With options like Hot Reload and large sets of fully-customizable widgets for building interfaces, Flutter is ideal for mobile development.

Who is using Flutter?

Flutter is used by companies all around the world. From Google Ads and Google Greentea to even Alibaba.com.

How Liquid Galaxy can utilise Flutter?

Liquid Galaxy can take advantage of this software for extending its mobile apps to both iOS and Android easier, instead of having to adapt the code for separate segments. On the other side, for this task will only be necessary a single team of creators.

Flutter can be used for incredibly fast rendering and expressive and flexible designs, which is an important factor for the Liquid Galaxy navigation tool.

Because this SDK is built in Dart, which is made by Google for web, server and mobile application, the developer can easily learn how to work in this environment. Dart is a cross-platform influenced by C#, Erlang, JavaScript, Smalltalk and Strongtalk, which are platforms that allow the app to work in a secure state.

Describing a use case:

If we think about the Liquid Galaxy for Education platform, Flutter can be used for managing a platform on both iOS and Android devices. For the questions and answers game, the application will receive a smooth running virtual image and a nice looking interface.

The buttons and the background will look better, in a Google style. The answer-button will be able to receive a nice pulse looking animation and some confetti falling on the screen if we want to take it to a new level. :) This thing can be done using the reactive input created by Flutter, which offers a variety of buttons and GestureDetectors classes. By using commands like “onDoubleTap”, “onLongPress” or “behavior” (for teaching the program a specific gesture) the application will offer accessibility.

For the free navigation segment, the 3D world will render fluently on the device, because of the flexible user interface rendering. The instructions placed at the bottom of the screen could have a more attractive look and the loading process when the destination is reached would be shorter.

Flutter uses Firebase, which is a back-end service used for hosting realtime database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for the files used inside the app. JSON and serialization will create the connection between the mobile and the web-server. All of those will allow the navigation feature to run faster and smoother on the device, because of the fast connection to the server. In the category of fast rendering takes part the Hot Reload, which works by injecting the updated source code files into the already running Dart VM.

I know this is not necessary an instruction set, but I like the design, the placement and the colors that were used. Also the animation of the first timer could be implemented in a check representation if the instructions were completed (ex. If the connection to the internet is available). For this

requirement, Hero animations are the way to go. This type of animations are used in moving images around the screen in a lot of online shopping apps. Flying an image from one screen to another is called a Hero animation in Flutter.