

Design rationale

Choice of API-level

For Eventify we have chosen a minimum API-level of 16 which is the first API-level of Android Jelly Bean. By having a minimum API-level of 16 we are able to cover approximately 95,2% of the total amount of Android users. With Eventify we had a goal of everyone being able to use our app while providing as much functionality we had in mind as possible, API-level 16 satisfies both of these goals.

External dependencies

Eventify is using several external dependencies in order to provide the functionality we wanted. Some of these dependencies are open source and tested by the community while the rest is developed by major companies like Facebook. The external dependencies used in Eventify are as follows:

- <https://github.com/Clans/FloatingActionButton>
- <https://github.com/SundeepK/CompactCalendarView>
- <https://github.com/bumptech/glide>
- Firebase
- Facebook API
- Google Gson
- Microsoft Bing Translate API
- Google Maps API

Database structure

Our initial prototype of Eventify did not involve any external database. Since most of Eventify's targeted users will have limited availability to an internet connection, we decided that we wanted to store as much data as possible on the user's phone, instead of relying on an internet connection to constantly retrieve new events and other data.

The idea behind Eventify is that the user will have internet access when the app is downloaded and installed. New events will then be collected at installation and saved locally, and if the user later moves to a location without internet access, the user is still able to see events that were retrieved the last time access was available. The only time new events will be collected from the Facebook API is when the user requests an update by scrolling down the event list. If the user at that time has access to internet, the update is performed.

One may now wonder why Firebase is used despite our decision of not having any external database. The reason for this is that the Facebook API requires an access token in order to pull data from Facebook's servers. This access token is created when a user uses the Facebook login function that is implemented in Eventify. The user's Facebook account credentials generate a unique access token which is then used to reach the Facebook servers. Since we in the planning phase decided that we also wanted to target users who do not have a Facebook account, a major problem surfaced. To solve this problem, we had to generate an access token using an account that was created for this purpose. It is considered bad practice to store access tokens of this kind locally due to security reasons, which is why we made the decision to store the access token in Firebase.

If the user decides to log in as a guest, Eventify retrieves the access token from Firebase instead of letting the Facebook API do it for us. This enables us to use the Facebook API and its servers even without a user's own Facebook account.