Storyteller

Julian Fuchs
Dominik Florencki

Idea

Tell **location based** stories ...
    ... to create a **city tour guide**
    ... to create a **fantasy adventure**

Feels like ...
    ... a text-based adventure game

Adaption concept - Technical context

Challenge: Offline

Contextual information: Network
　　　ConnectivityManager.TYPE_MOBILE
　　　ConnectivityManager.TYPE_WIFI

Adaption: Download data and give hints
　　　WIFI - Download new stories without asking the user; use GoogleMaps for hints
　　　MOBILE - Ask the user if he wants to download new stories; use GoogleMaps for hints
　　　OFFLINE - No download of new stories possible; use text-hints only

Adaption concept - Technical context

Challenge: Battery draining issue

Contextual information: Battery
       BatteryManager.BATTERY_PROPERTY_CAPACITY

Adaption: Reduce GPS usage when battery is low
       100% until 31% - Use GPS as intended
       30% until 0% - Use GPS less frequently without reducing of usability and fun

Adaption concept - Physical context

Challenge: Location

Contextual information: GPS, Network
      LocationManager.GPS_PROVIDER
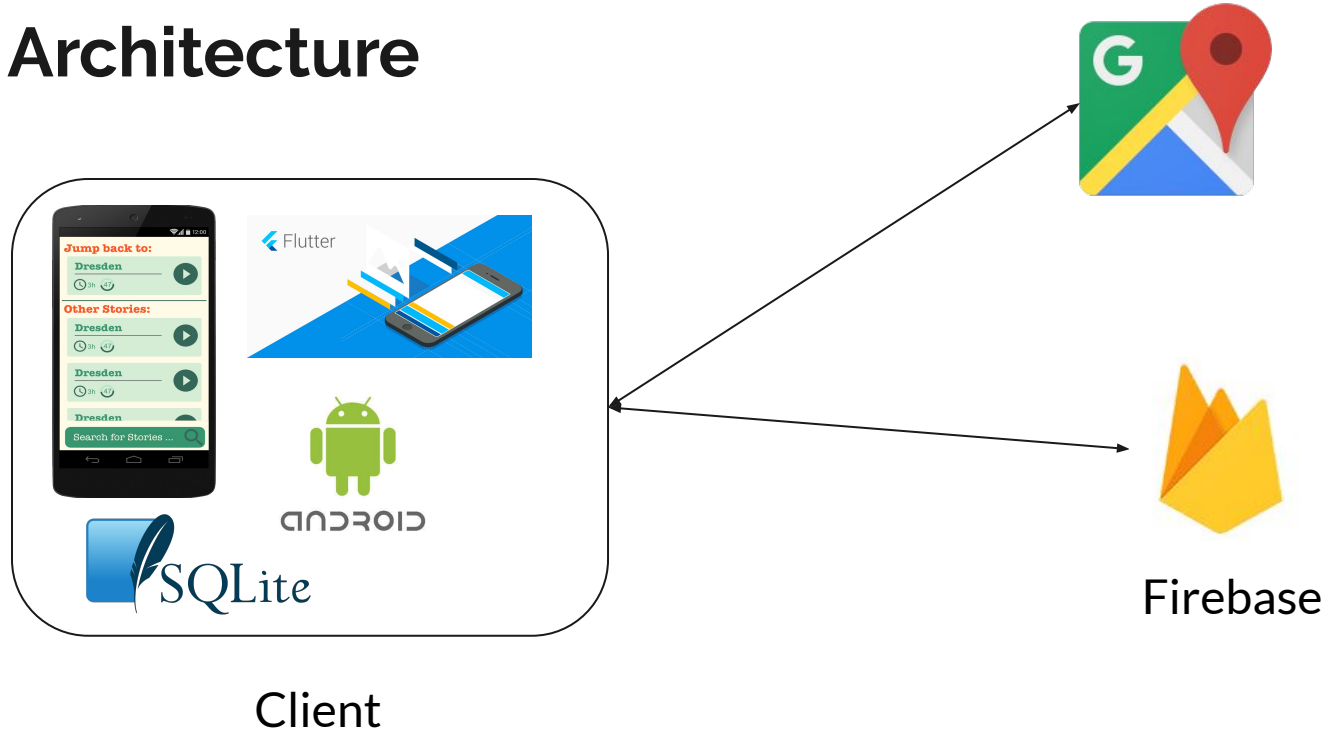   LocationManager.NETWORK_PROVIDER

Adaption: Reduce GPS usage when battery is low or use Network-location instead
      100% until 31% - Use GPS as intended
      30% until 0% - use only the Network Location (instead of GPS) and reduce the rate of
location updates
      30% until 0% - If Network is not enabled use GPS

Architecture

Client

Firebase

Technologies

OS: Android
Language: Dart (Flutter-Framework)
IDE: Android Studio
VCS: Git (GitHub)
**Data(base):** SQLite (local), JSON-File (server)
Maps API: Google Maps
Server: Firebase

Storing Data

- All Stories are stored on **Cloud Firestore** (from Firebase and Google Cloud Plattform)
- Every user has the stories on his device he already downloaded (stored in local **SQLite**)
- The user sees only the Title of the stories in the app and more data is downloaded on demand when the user tries to open a story he hasn't downloaded yet:
  - **WiFi:** Story-Data is downloaded
  - **Mobile:** the story-data is downloaded after user approvement of using mobile data, otherwise it will be stored in the queue waiting for the wifi connection
  - **Offline:** The user cannot download any content, he can look through already cached data and choose which one add to downloading queue. Stories he already downloaded are still available for him (because they are stored locally in his SQLite-Database)

Screenshots
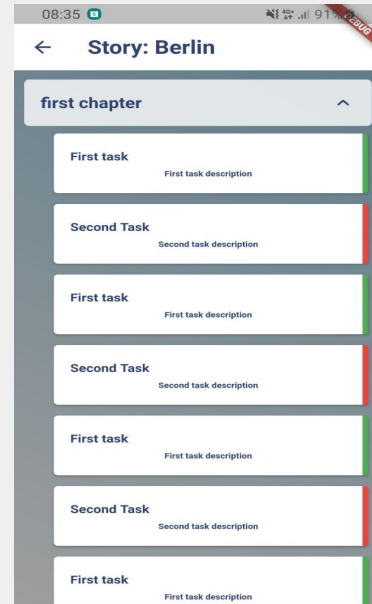
Thank you for your attention.

by
Julian Fuchs
Dominik Florencki