Happy Grocery 

Alessandro Castellani, Fabio Castello

Overall Description

The goal of Happy Grocery is making grocery shopping easy and fast! Through the app users will be able to scan barcodes, add them to their virtual shopping cart and pay.
From the instant users get into the mall Happy Grocery can be their only tool to ultimate the shopping experience.

Goal:

Users must be able to replicate through the app a complete shopping
Here are the different aims this project started with:

[G1] Login system must be fast and simple

[G2] Products are recognized through barcode scanning
ID of the shop is obtained through a QR code

[G3] Shopping cart is persistent

[G4] The app interacts with external shops' servers, obtaining info on product names, prices and discounts

[G5] The app shows consistently info on the cart: products are divided in categories

[G6] Info on past carts may be stored online

Assumptions:

1) In order to work, the app requires the target shop to actively support it.
   The shop shall grant access to an external RESTful interface with all info on its displayed products
2) Users must have an Android 5.0+ device with functioning camera and internet connection
3) In case the user owns a smartwatch it must be linked through Bluetooth to its master phone
4) External shops offer a unified REST interface and structure for the purposes of this app

Requirements:

[R.1.1] Login employs cellphone number and no more than one account shall be linked to each phone number
[R.1.2] Google Smart Lock automatizes the login experience after the initial registration
[R.1.3] An unregistered user is locked out of the application
[R.1.4] Firebase is the native solution to manage an efficient login

[R.2.1] Shops codes must be QR codes
[R.2.2] Products codes must be EAN-13 codes
[R.2.3] Codes are unique and saved to an external database
[R.2.4] Codes can be scanned multiple times in the same user flow

[R.3.1] Shopping data must be saved to a local database to avoid any interruption in the user's flow of actions (shutting down the app, crash, etc)
[R.3.2] Adding multiple times the same product will merge entries
[R.3.3] Database restore allows to repristinate the current status of the shopping in case an interruption happens
[R.3.4] Database is cleansed only upon shopping end

[R.4.1] External shops databases present a unified interface to access their content.
[R.4.2] The external DB coordinates are acquired upon scanning the QR code
[R.4.3] The external DB endpoints require only the barcode to release info on products
[R.4.4] External endpoints are persistent while shopping flow is live
[R.4.5] No external info coming from such servers can be in any way altered or modified, nor locally nor remotely

[R.5.1] Info on the shopping cart must be simple enough to be visualized and understood at first glance
[R.5.2] Categories assignment happens upon product recognition
[R.5.3] Categories are given by the external shopping DB and cannot be modified

[R.6.1] Past shopping carts are stored locally and uploaded
[R.6.2] Every time a user access old shopping info the app checks if there are further older transactions