Mobile APP

Non-Functional Requirements:

- NFR01: All users must be authenticated with Facebook;
- NFR02: The APP must be developed in Java Language;
- NFR03: The APP must use SQLite Database;
- NFR04: All distances must be expressed in Metrics Unit.

Functional Requirements:

- FR01: Request user to authenticate using the Facebook API after the internal system authentication;
- FR02: Allows users to locate friends utilizing the same APP;
- FR03: Allows users check the profile of others users with the APP, requesting info about the users from the Facebook API and the Web service Database.
- FR04: Allows users post comments on their Facebook wall about the locating he is currently, gathering the info about the local from the GPS and Google Maps Geo-Coding API automatically.
- FR05: Allows users check the locations he was by period, utilizing the info that was sent to the Web service while he was utilizing the APP.
- FR06: Allows users find unknown persons nearby him with the same affinity utilizing the Facebook Like system.
- FR07: Allows users check his friends and his own most visited places, if allowed.
- FR08: Allows users to stop sending his location to the Web service for the historical purpose, allowing him be incognito temporary
- FR09: Allows users choose who can and can't see his current position and his location history;
- FR10: Allows users to create a Google Maps path to some friend or someone with the same affinity;
- FR11: Allows users to enable and disable sound effects and notifications;

- FR12: The APP must warn the users utilizing sounds effects and notifications about people who have the same affinities and is nearby, and easily allows the user to see that person profile (Facebook and System profile);
- FR13: Allows users exchange his Facebook profile by another profile he owns;
- FR14: Allows users to use Single Sign On, allowing him to automatically log-in utilizing his Facebook Profile
- FR15: Shows the user a Heat Map with the info about him or his friend visited location, if he is allowed to;
- FR16: The APP must check if his position was changed before sending the info to the Web service, avoiding sending the same location if he is in the same place;
- FR17: The APP must use the least possible data usage, utilizing data compressing sending info to the Web service
- FR18: The APP must send periodically and on launched a request to the Web service to check if its running, if not, show an error message
- FR19: The APP must ask which locating he really is, allowing the user to choose on a list of all nearby locations
- FR20: Allow the user to chat with his friends, utilizing any means, Facebook chat, or an internal chat.
- FR21: The AP must allow the user to see his own Facebook likes and other people Facebook likes, if the other person is nearby.

Use Cases:

- Use Case "Login": The user will fill his credentials, login and password, and request the APP to authenticate with the Web service.
- Use Case "View location history": The user can see his own or friends most visited locations, and can see a Heat Map with this info.
- Use Case "View profile from other users": The user can see the Facebook profile from nearby users, utilizing the GPS to determine the location. All the profile must be show on the APP.
- Use Case "View friends location": The user can see the location of his friends, if the friend allows him to;

- Use Case "Post comments of the location": The user can post comments of the location he is on his own wall, posting automatically the location name utilizing Geo-Coding.

- Use Case "Trace route to other user": The user can trace routs with the shortest path to the position of another user, utilizing the Google Maps API.

- Use Case "Change Facebook profile": The user can change the profile he is utilizing on his account by another one, authenticating with the Facebook Graph API.

- Use case: "Change privacy and usability settings": The user can change his privacy settings, for example, if he can be seen by others users or save his history on data base. Or change his settings about the notifications from the APP.

Screen Prototypes
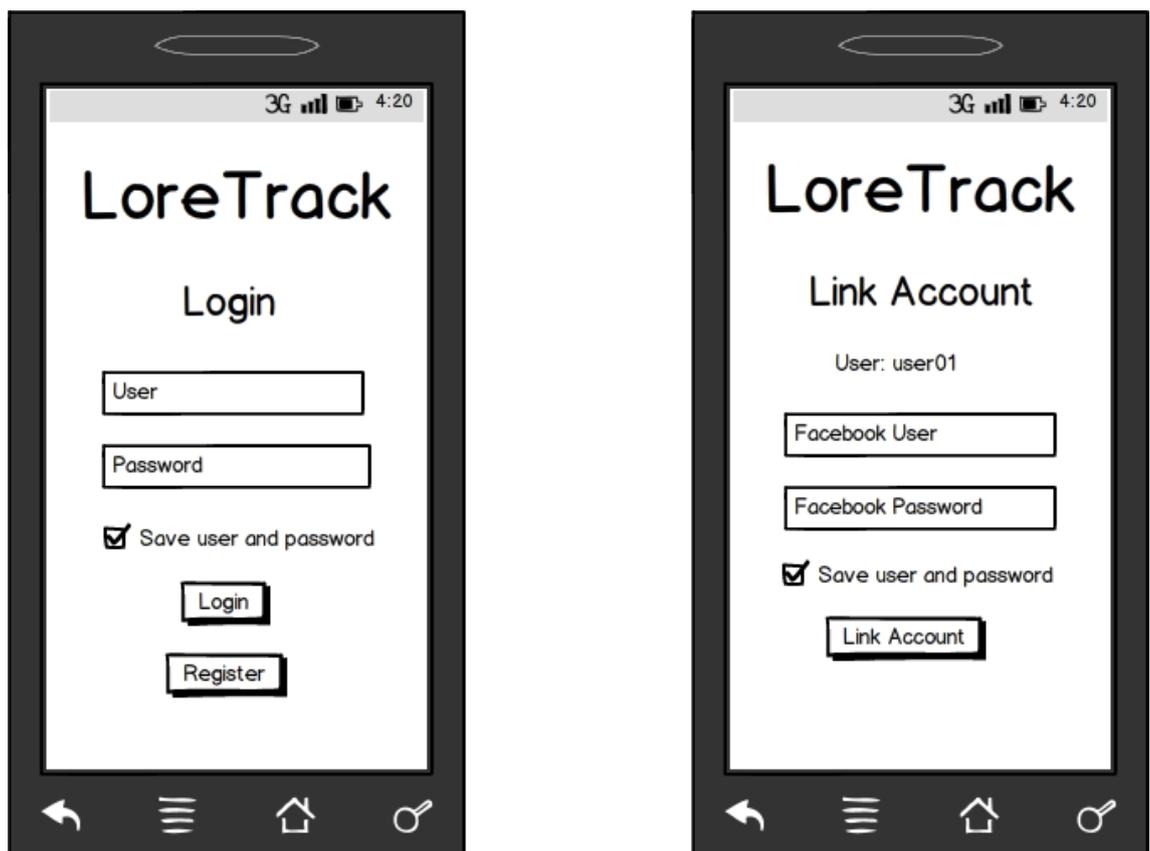
The login screen the user must enter his credentials registered on the system (not his facebook account), or choose the option to register a new account, both the option will send the user to the Facebook Graph API to login with the Facebook account, linking his Facebook account to the system account.

The menu screen is the main window of the APP, on this window he can open all others APP windows, and even quickly set the option to enter incognito mode.

The map screen allows users to see his actual position, his friends positions, or even make a comment of his position and post automatically on his Facebook Wall, posting automatically his location, utilizing Google API Geo-Coding
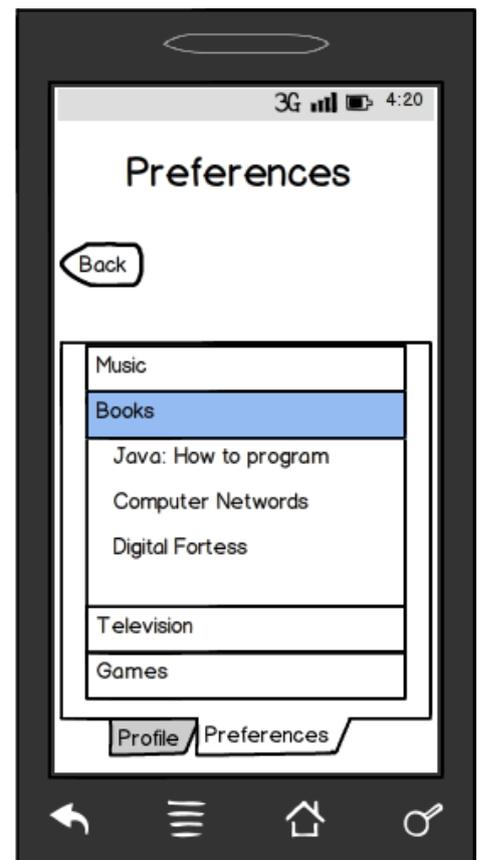
The profile and preferences screens are subdivided in tabs. The profile allows the user to see and modify some info, changing also on his Facebook account. The preferences allows the user to see his own Facebook likes, subdivided in categories.
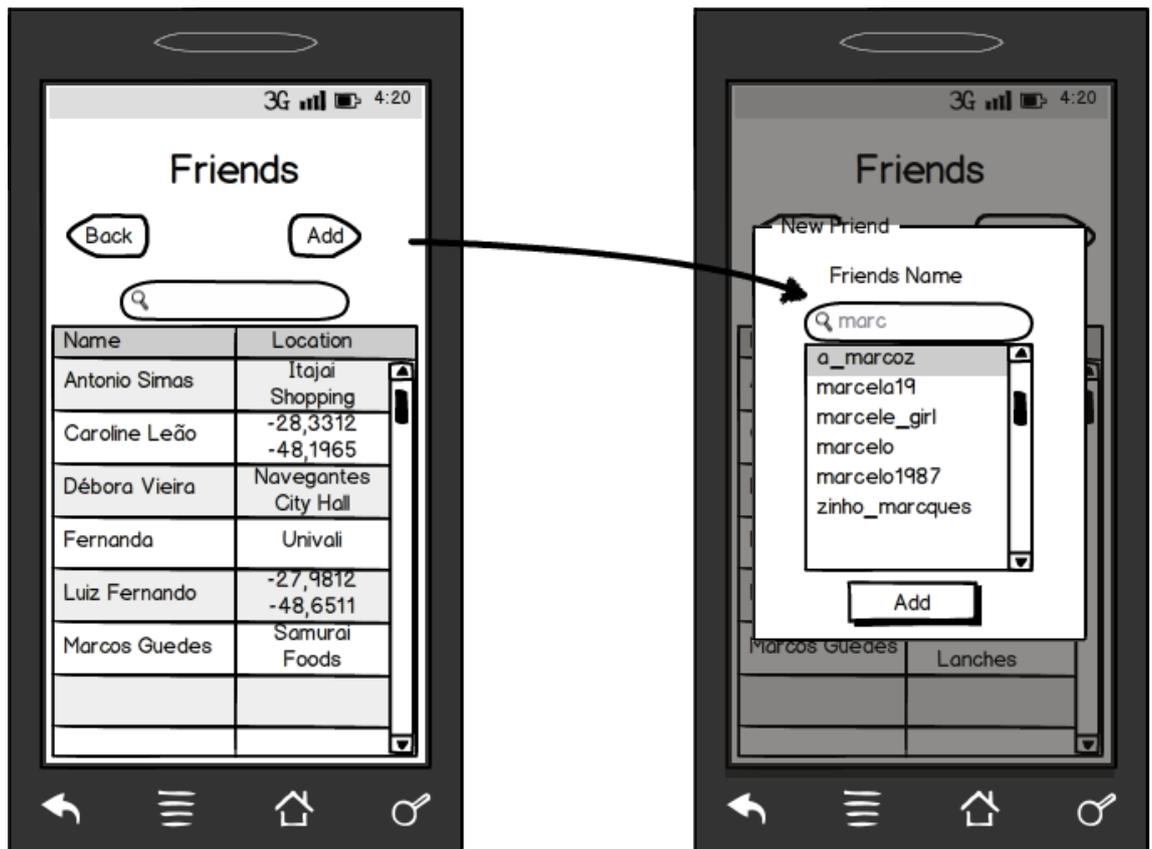
The friends screen allows the user check his friends' actual positions, giving the possibility to filter by real name or nickname, and ask to add someone to his friends list. Allows also to check his friends Facebook profile, by just clicking in it.
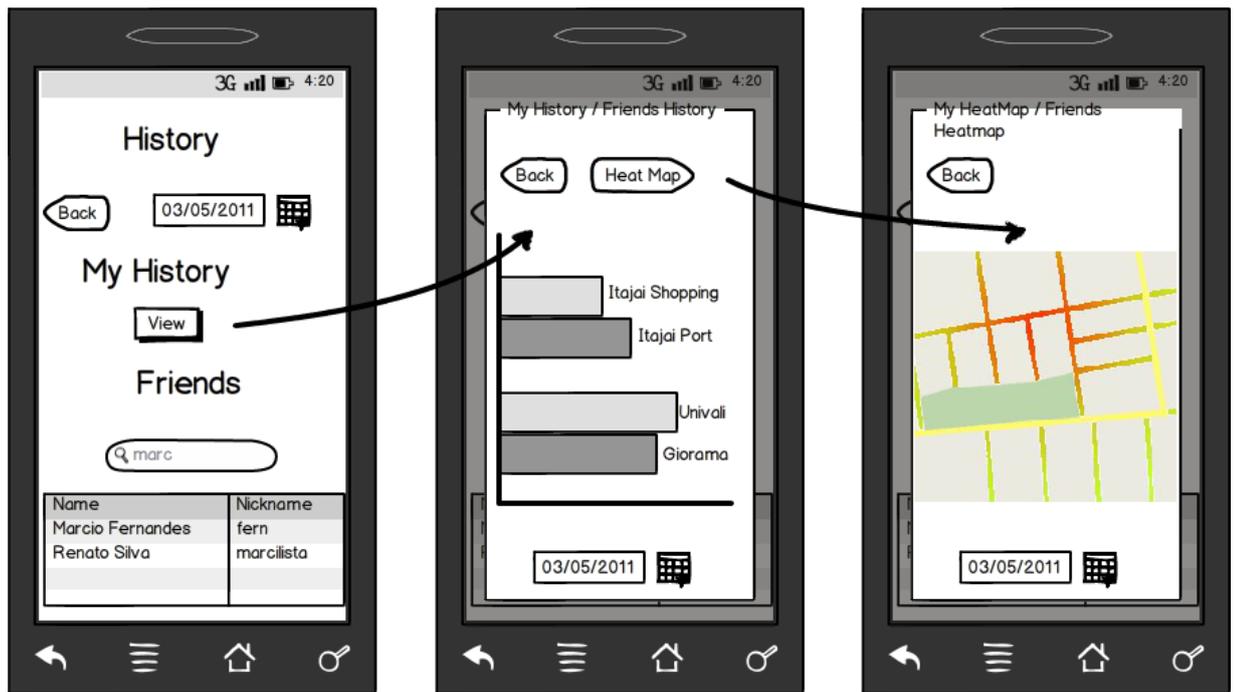
The history screens allows the user to see his owns location history or their friends, allowing him to filter by day, bringing the most visited locations and the heat map on the current period.
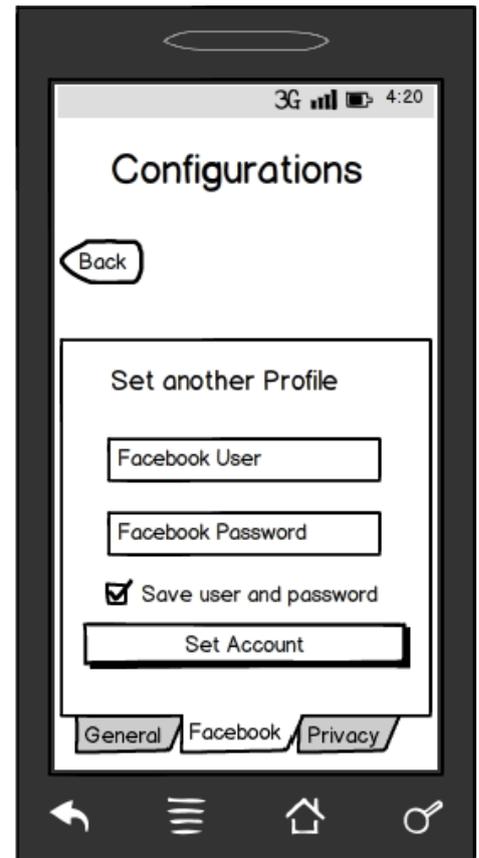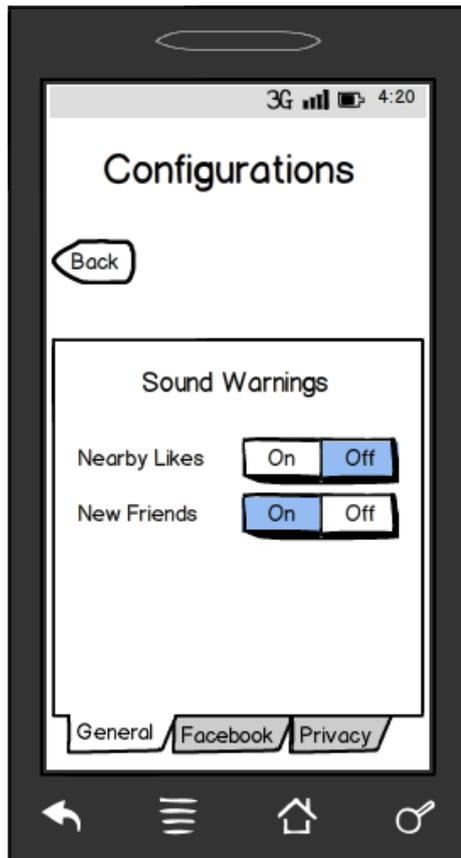
The tab general in configuration allows the user modify some options about how he is notified about some user with the same likes near him, or warning about someone wanting to be his friend
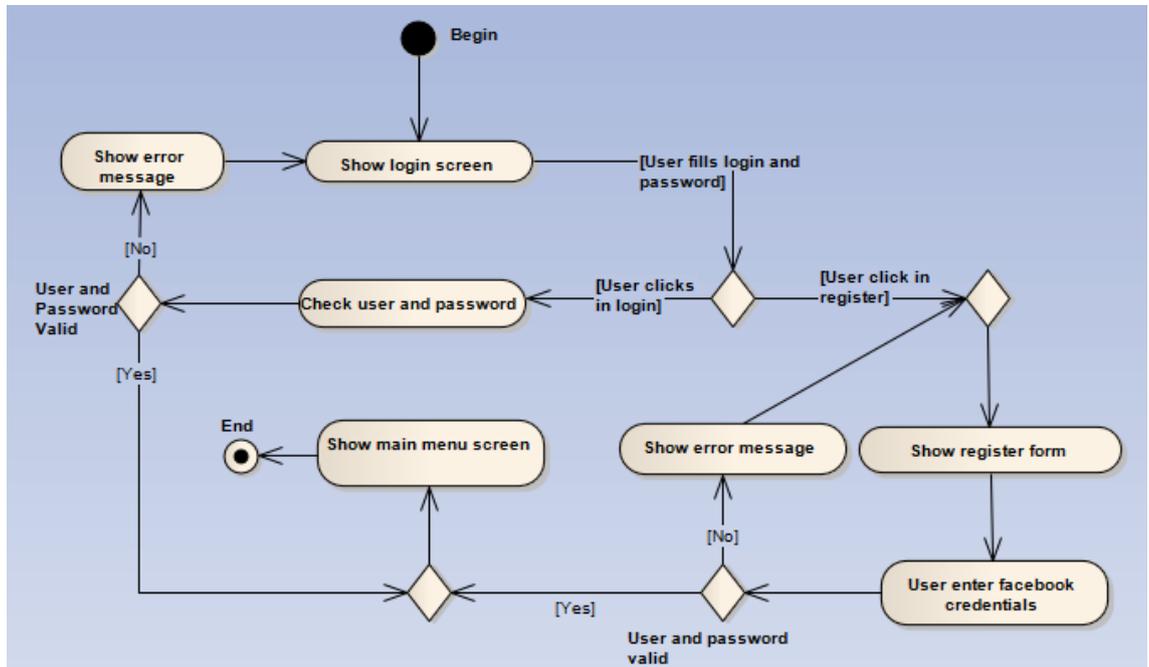
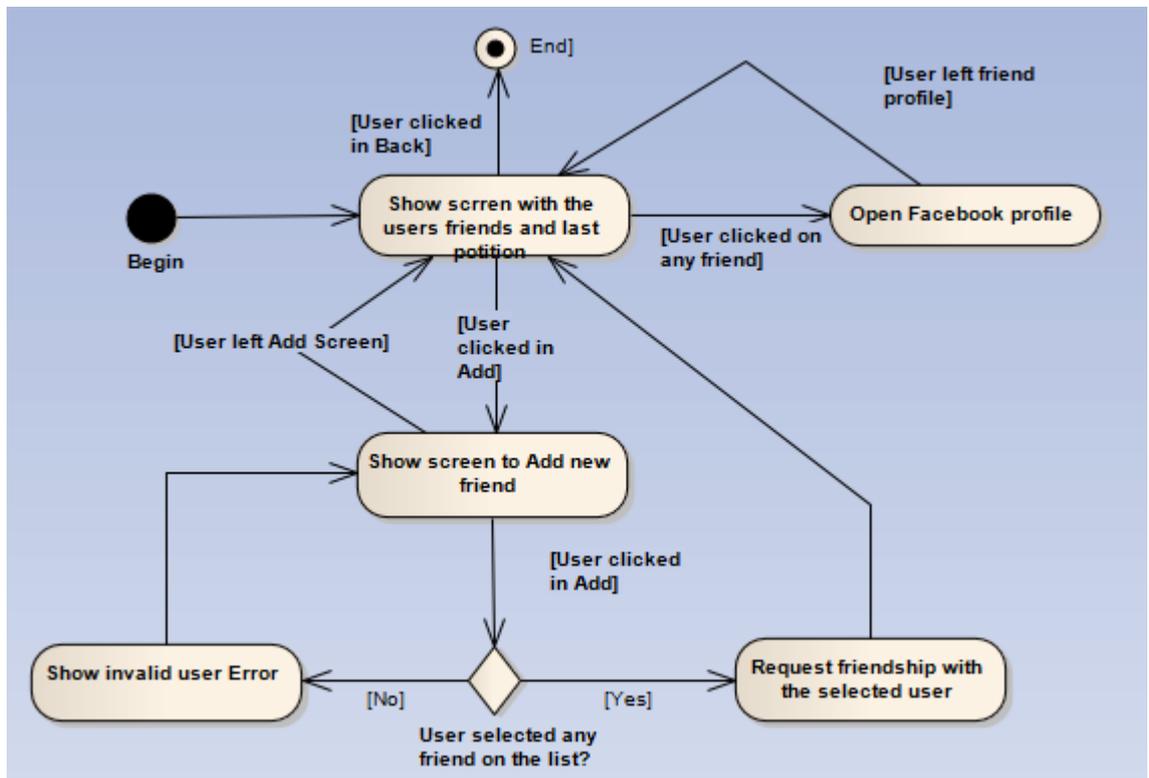The Facebook tab allows the user to change his Facebook profile.
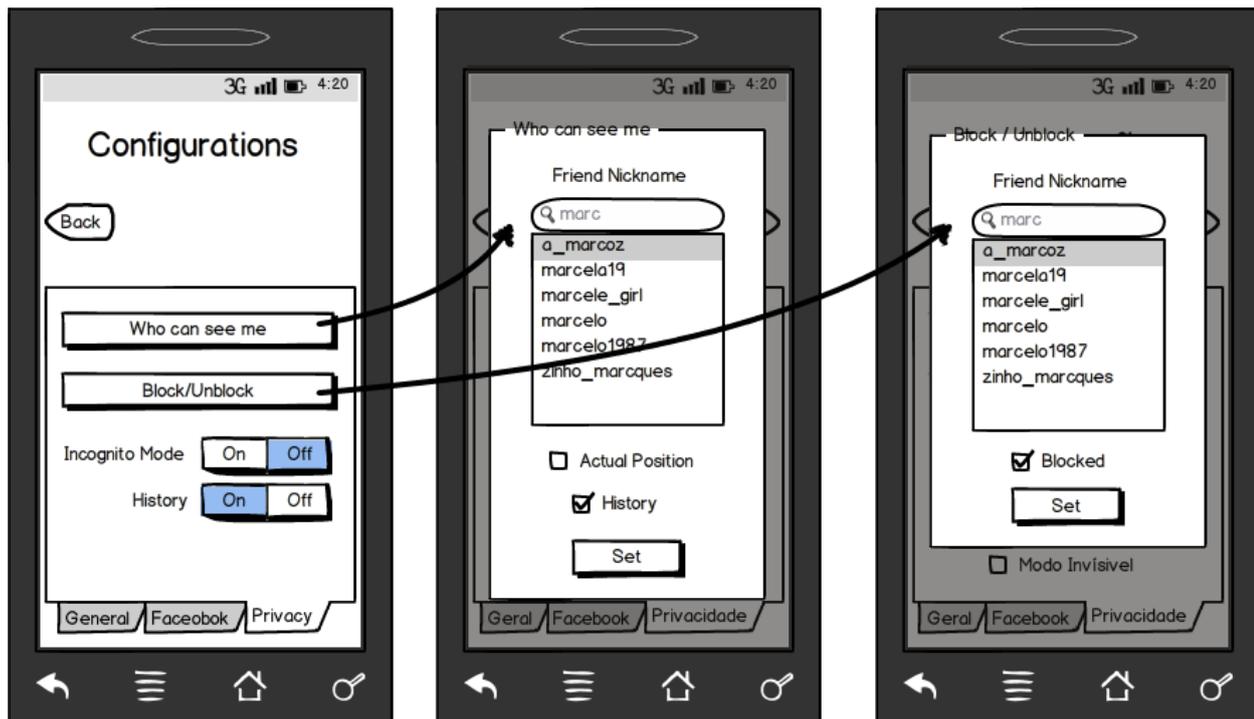
The privacy tab allows the user choose how others peoples interact with him, allowing him to change who can see his history and actual position and to block or unblock any user.
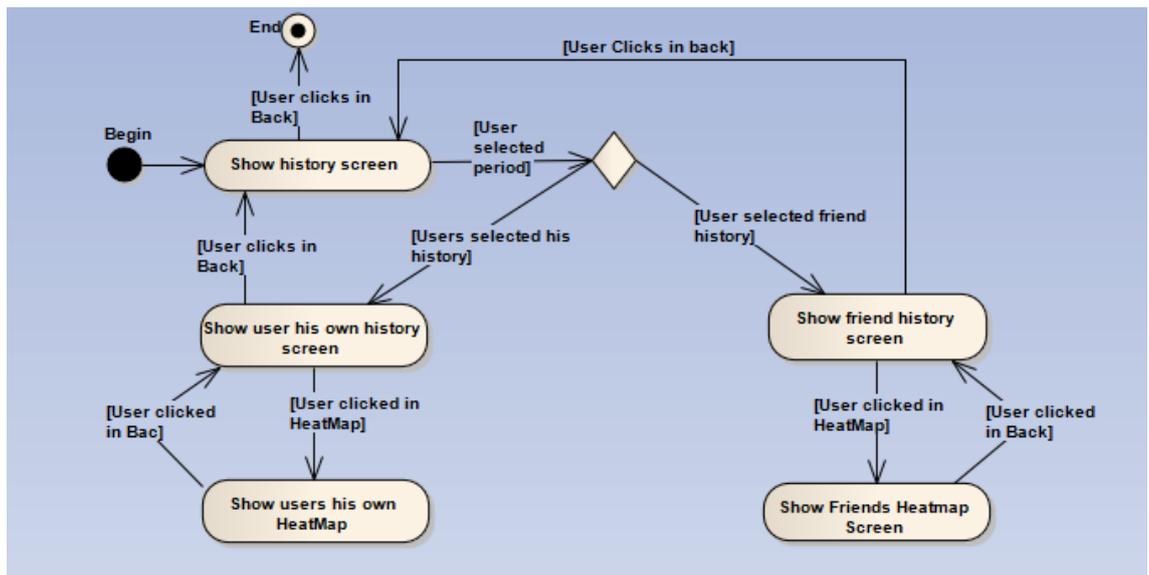
Activity Diagram

Login:

Friends:

History:

Privacy: