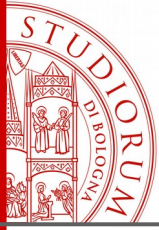


ANDROID PROJECTS

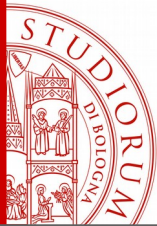
Proposals 2018

Federico Montori and Luca Bedogni



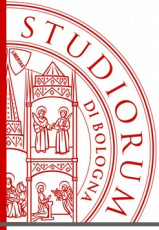
Introduction

- The following proposals must be considered **just hints**.
- All the main functionalities listed must be implemented (**minimal requirements** to have the project accepted).
- We strongly encourage to **expand/customize the proposal based on your creativity**.



Introduction

- Projects described in the following must be deployed by a **single student**. Group projects are not allowed.
- Project implementation must be **original and 100% student work** (no code share or reuse).
- Submit the project by email (lamprojects@cs.unibo.it) including all code, a technical report, and a short presentation (10-15 slides)

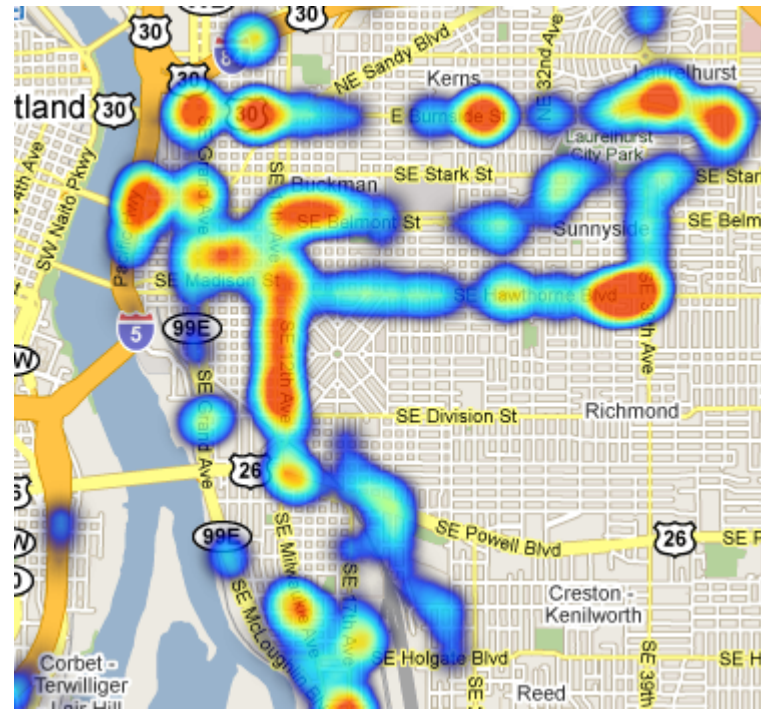


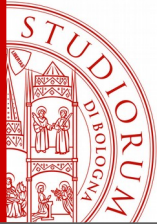
Introduction

- Read and follow the instructions about projects submission policies (deadlines, validity, etc) on the course website:
<http://www.cs.unibo.it/bononi>

PROJECT 1

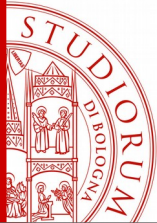
- Implement a Wireless Connectivity Map
 - Use the Google Maps API
 - Color the map based on the signal strength received (from red to green)
 - Monitor three different technologies (LTE, UMTS, WiFi).





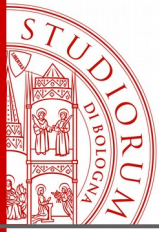
PROJECT 1

- Functionality 1: encode Google Maps Areas
 - Use the Google Maps API
 - Choose a valid encoding for representing areas instead of points (so no pure GPS).
 - Areas should be encoded in a way such that the whole space can be covered without “holes” (e.g. circles are not valid unless overlapping).



PROJECT 1

- Functionality 2: encode the RSSI
 - Visualize the connectivity strength (the RSSI) through coloring the areas with a color scale (e.g. red to green).
 - The data should be acquired again if a (settable) amount of time has elapsed.
 - Two (or more) measurements about the same area have to be visible in some way (e.g. the transparency of the color).



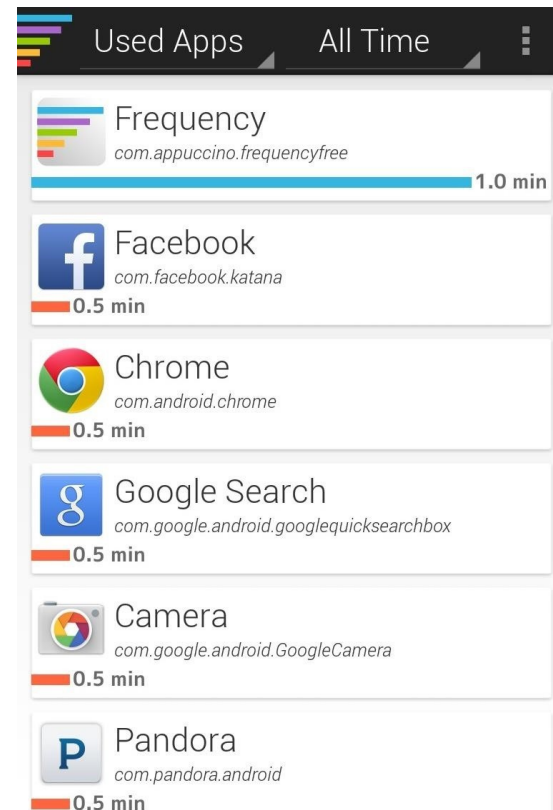
PROJECT 1

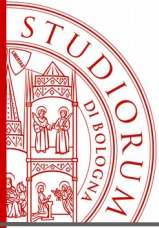
- Functionality 3: Three technologies
 - The applications should show AT LEAST three different maps depending on the technology used:
 - LTE (4G)
 - UMTS (3G)
 - WiFi

PROJECT 2

- Implement an Application Monitor

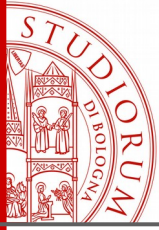
- Monitor the status and the usage of the applications in background
- The settings about what and how to monitor are customizable
- Send a report to the user upon his/her preference





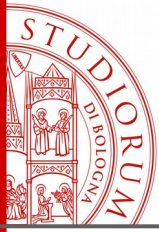
PROJECT 2

- **Functionality 1: Monitor the Apps in background**
 - Access to the usage data of the apps in background.
 - Provide at least three GLOBAL metrics in output (e.g. how many apps are running).
 - Provide at least three PER-APP metrics in output (e.g. for how long the app has been running).



PROJECT 2

- Functionality 2: interactive selection
 - Make the settings about usage customizable (e.g. which apps to monitor, for how long, which metrics to use, etc.) .
 - Make the settings about sending the report customizable (e.g. where to send the report, how often, if upon request etc.) .

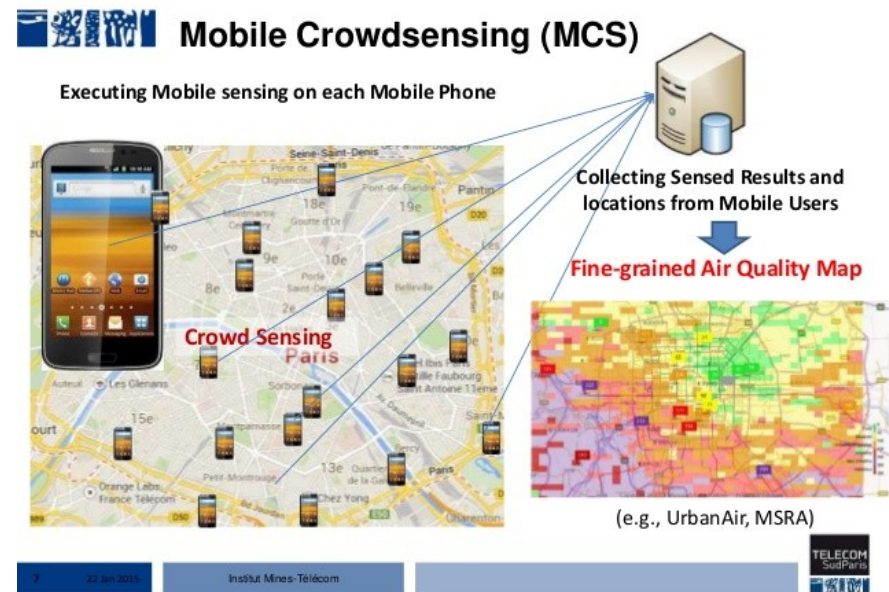


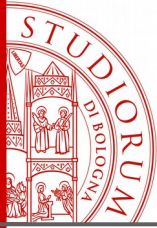
PROJECT 2

- Functionality 3: Sending the report
 - The report has to be sent where the user is able to check it (e-mail, Social media, an external server.)
 - The report has to be sent either:
 - Periodically (settable)
 - On demand

PROJECT 3

- Implement a Participatory Crowdsensing App for Citizen Science
- Notify the user of incoming tasks and accept/reject them
- Notify the user of the possibility of accomplishing a task.
- Guide the user in performing the task and send back the response.





PROJECT 3

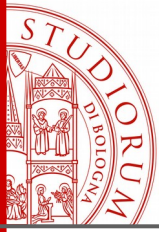
- THE TASK

The task is a twitter post containing the hashtag #LAM_CROWD18 (the twitter user @LAM_UNIBO_2018 posts periodically, or you can do yours).

- You have to use the twitter API.

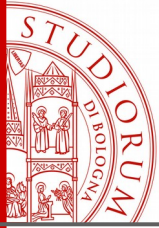
- The task has this format:

```
{  
  "ID" : "LookForKoalas"  
  "issuer" : "LAM_UNIBO_2018",  
  "type" : "picture",  
  "lat" : "-37.835309",  
  "lon" : "145.047363",  
  "radius" : "1.0",  
  "duration" : "5",  
  "what" : "eucalyptus_trees"  
}
```



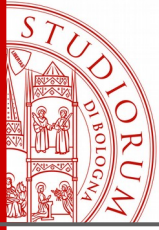
PROJECT 3

- **Functionality 1: Receiving the task**
 - The task has to be received through notification.
 - The user can accept or reject it manually.
 - The task can only be executed where specified and during the time lapse specified.
 - Non accomplished tasks should be kept in memory.



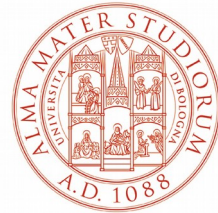
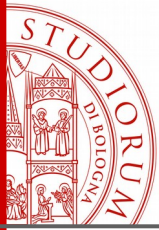
PROJECT 3

- **Functionality 2: Executing the task**
 - The possibility of accomplishing a task (time and space condition are met) should be notified to the user.
 - The user can accept or reject the task execution.
 - If the execution is accepted the application should interactively guide the user through the process of:
 - Taking a photo.
 - Perform a sensor read.



PROJECT 3

- **Functionality 3: Report the result**
 - The result of the execution of a task (a photo or a sensor read) should be posted as a reply of the original post.
 - Once executed, the task should be deleted from the list of pending tasks.



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Federico Montori

Unibo

federico.montori2@unibo.it