



**TECHNISCHE
UNIVERSITÄT
DRESDEN**

Department of Computer Science Institute for System Architecture, Chair for Computer Networks

Application Development for Mobile and Ubiquitous Computing

**HandHirer
Final Presentation**

**Team 8
Maxime Thébault
Sebastian Kunze**

Dresden, 27.01.2017



**DRESDEN
concept**
Exzellenz aus
Wissenschaft
und Kultur

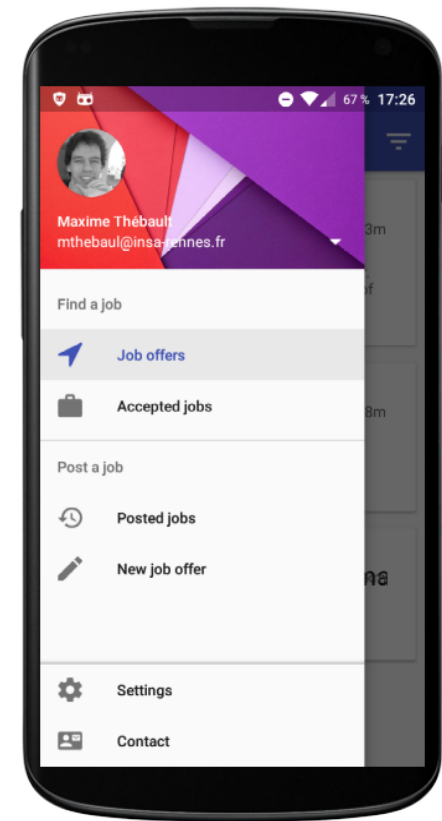
HandHirer

Idea

- quick way to offer job in current location
- look for jobs around you
- can also be used as a platform for classified ads

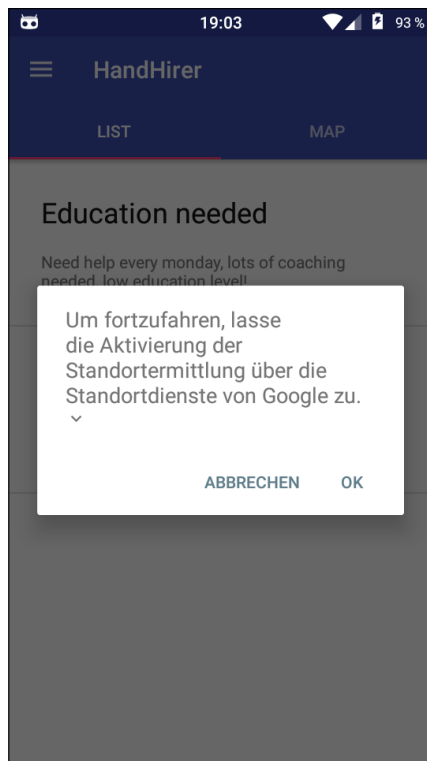
Scenario

- *create offer* with selectable or tracked gps location
- *look for offers* created by others near current position or via map and show details



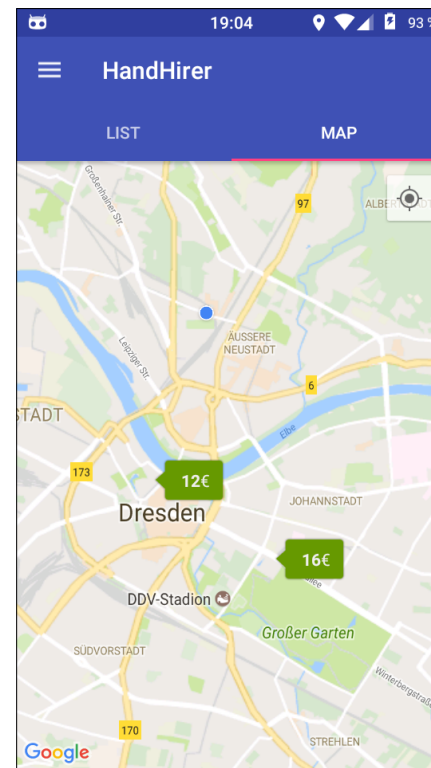
Demo

Starting up



27.01.2017

Browsing with map view

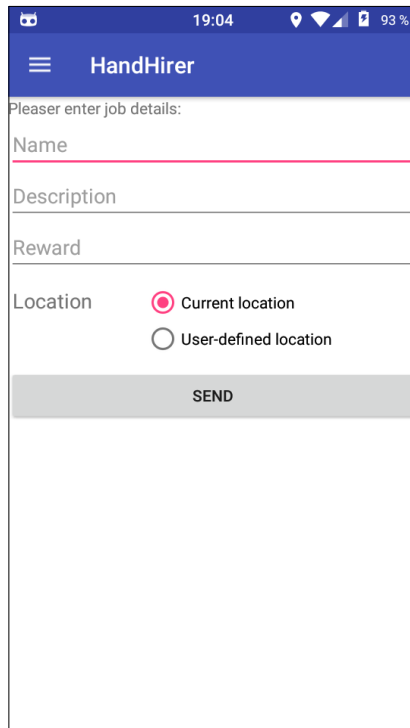


HandHirer – Final Presentation

Slide 3

Demo

Creating offer (location fetched via GPS)

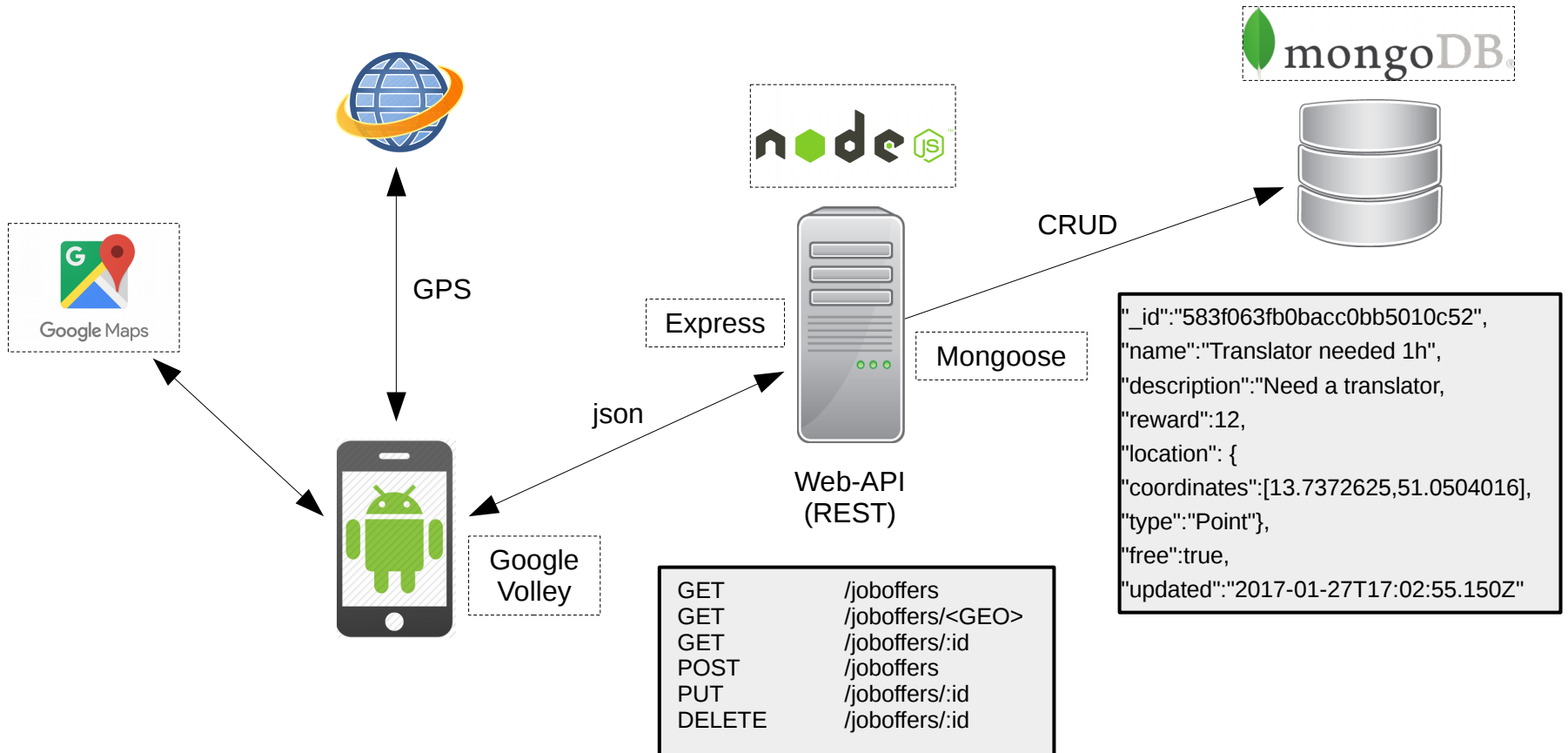


The screenshot shows the 'HandHirer' app interface. At the top, there is a blue header with a menu icon and the text 'HandHirer'. Below the header, the text 'Pleaser enter job details:' is displayed. The form consists of several input fields: 'Name', 'Description', and 'Reward'. Under the 'Location' section, there are two radio buttons: 'Current location' (which is selected) and 'User-defined location'. At the bottom of the form, there is a grey button labeled 'SEND'.

Creating offer (location selected manually)



Architecture



Context information

Physical

- GPS location gets obtained (or can be set manually)

Technical

- Permissions (GPS + internet) are checked
- Internet access is detected
- Device size and orientation

Adaptation mechanisms

Location

- Map zooms to current location
- Created offers get GPS coordinates of device
- Documents in DB GeoJSON conform
 - Use of MongoDB query operator „geoNear“ possible
 - Only offers near current position are shown

Permissions

- If GPS permission is denied, application displays prompt with possible activation

Adaptation mechanisms

Connectivity

- Reduced feature set if offline
 - User gets notified
 - Browsing functionality through cached data

Device properties

- Adapting to screen size
- Device orientation recognized and handled

Challenges of mobile computing tackled

Connectivity

- Queuing through volley adds to reliability

Offline

- Local storage enables basic functionality

Form factor

- Adapting to layout + orientation

Usability

- Material design guidelines aimed at
- Generally expected behaviour

Open issues

User authentication

- Login system not implemented

Managing offers

- Web API offers more functionality than implemented in application (deleting, updating)

Lessons learned & pitfalls

Learned

- Dealing with network requests within the Android framework thanks to Google Volley
- Usage of coordinator layout
- How to deal with configuration changes (e.g. screen rotation)

Challenges

- Organizing the code into a clear structure
- Not putting everything in the same Activity - decoupling concepts

Questions?

Thank you for your attention!