TECHNISCHE UNIVERSITÄT DRESDEN

DRESDEN concept

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

Application Development for Mobile and Ubiquitous Computing

Dr.-Ing. Thomas Springer

FlatSharing

First Presentation

Group 7

Lucija Veljacic
Daniel Creanga
João Calado

Dresden, 3 November 2017

Application Idea

- We provide an easy solution for students who need to find accomodation during their studies.

- Just create an account, provide your student email, and start looking for your perfect place to stay!

- Find new flatmates from all over the world.

- If you have a flat to rent, just place the advertisement and find new tenants.

Challenge adaptation

Offline Challenge:

- Provide some available content to use in offline mode.

Context: Detect if there is a network connection available.

```
ConnectivityManager.TYPE_MOBILE || ConnectivityManager.TYPE_WIFI
```

Adaptation: Cachable content of the latest, following and own advertisements data on the client side, assuring data integrity for recovery mode.

Energy Challenge:

- Reduce battery consumption of the GPS.

Context: Detect user location with GPS from time to time.

```
oldLocation = LocationManager.GPS_PROVIDER
int ONE_MINUTE = 1000 * 60 * 1;
newLocation = LocationManager.GPS_PROVIDER
//compare
```

Adaptation: Use GPS tracking activity only when a user starts moving or searches for ads.

Challenge adaptation

Usability Challenge:

- Provide a good and intuitive UI to users with all types of devices (e.g. Tablet, Smartphone);
- Minimize user input interaction.

Context:

- Detect screen size and device type;

  Configuration.SCREENLAYOUT_SIZE_MASK
- Detect user location with GPS;

  LocationManager.GPS_PROVIDER && LocationManager.NETWORK_PROVIDER
- Notify user about new subscriptions, price changes, etc.

  Google Cloud Pub/Sub

Adaptation:

- App pages showing more/less ad information for different devices;
- Center map on user’s position;
- Usage of Google Cloud Pub/Sub system to notify users of desired info.

Used Technologies

Client-Side Technologies

android

Android Studio

POSTMAN

Request/Reply mechanisms

{JSON}
JavaScript Object Notation

{ REST }

Cloud Database

Google Cloud Platform

Server-Side Technologies

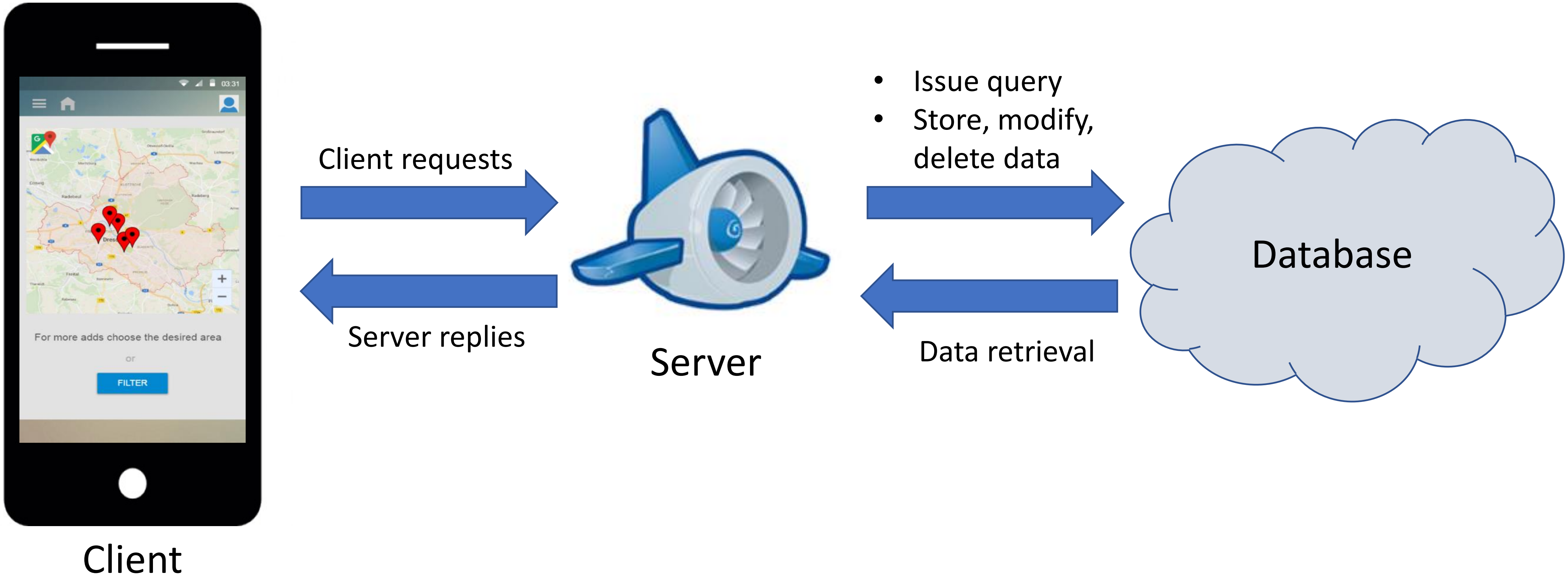
eclipse

Java

Google app engine

Apache maven

System Architecture



For more adds choose the desired area

Client

Client requests

Server replies

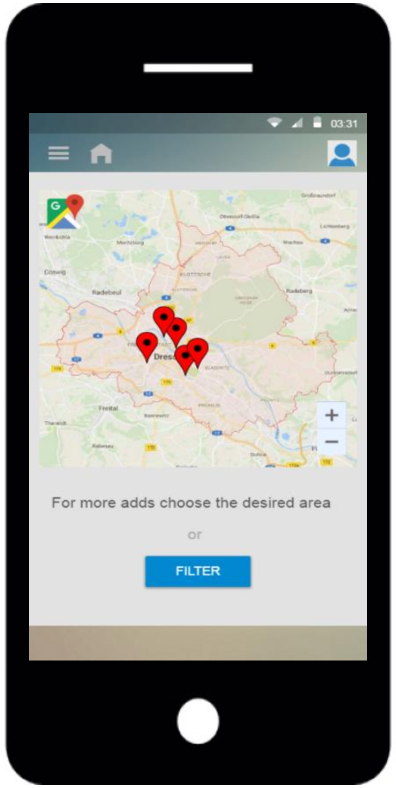
Server

- Issue query
- Store, modify, delete data

Data retrieval

Database

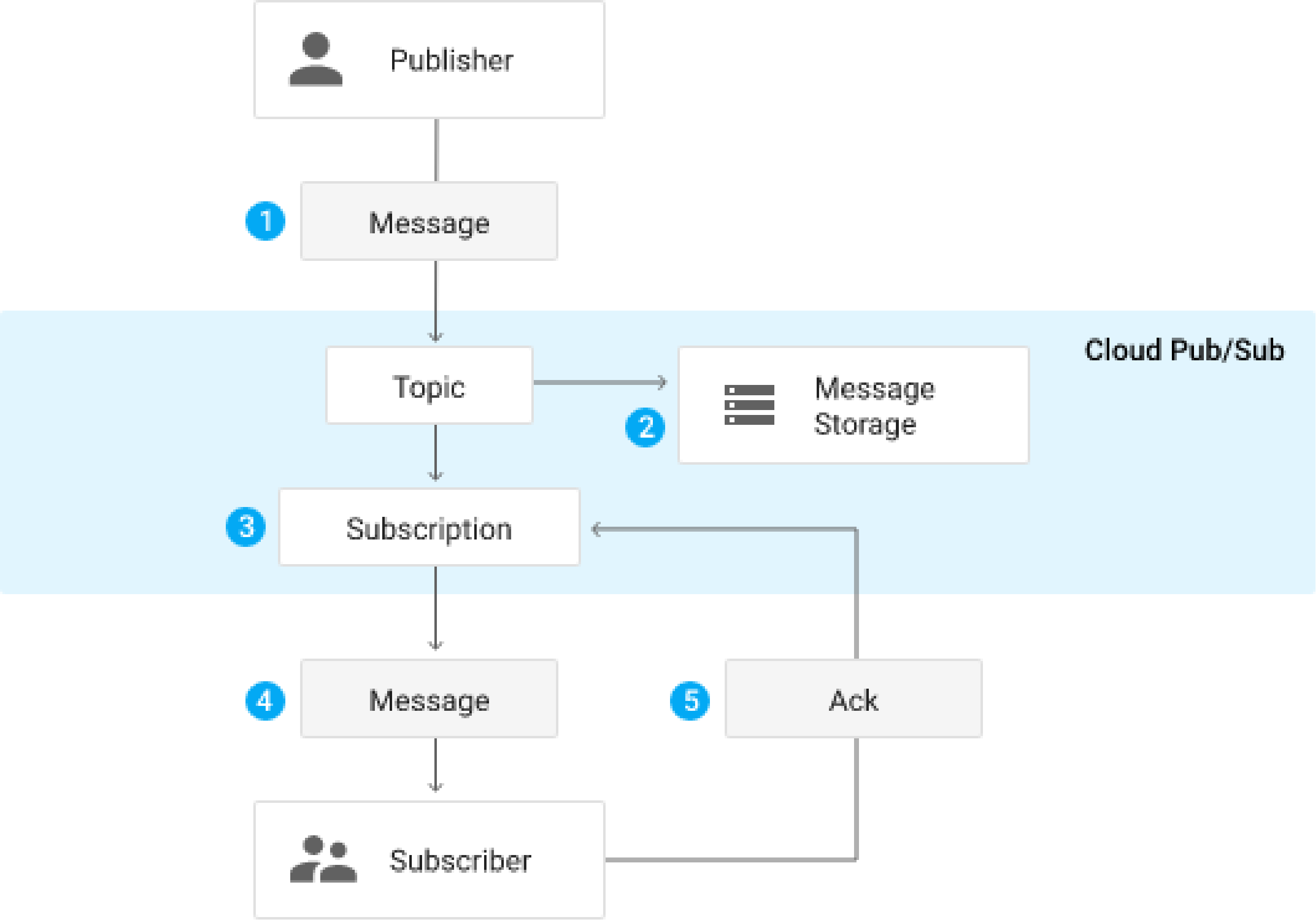
System Architecture – Pub/Sub



For more adds choose the desired area

or

Client



Summary of Server Requests/Replies

REST API

HTTP methods

- GET
  - /user/{email}
  - /ad/{address}
  - /ad/list →
    - /?minCost={…}
    - /?maxCost={…}
    - /?lat={…}
    - /?lng={…}
    - /?radius={…}
    - /?cursor={…}
    - /?page_size={…}
    - /?submitter={…}
- POST
  - /register
  - /login
  - /ad/post
- PUT
  - /user/{email}
  - /ad/delete/{address}/{email}
  - /ad/{address}
  - /ad/book/{address}/{email}
- DELETE
  - /user/delete/{email}
  - /logout
  - /ad/delete/{address}

Workplan

03.11.2017: First presentation

November ✓

- Discuss the server modeling (e.g. methods and database modeling)
- Backend development

15.12.2017: Adaptation concept presentation

December

- Frontend development
- Discussion and implementation of the subscription mechanism
- First functional prototype

January

- User Interface implementation
- Testing and bug fixes
- Final version

26.01.2018: Final presentation

Thank you for your attention!