Demo Day Preparation

Prof. Dr. Dirk Riehle

Friedrich-Alexander University Erlangen-Nürnberg

AMOS H01

The AMOS Demo Day

- The demo day is the final day of the course
  - The demo day is organized as a **fair** ("Messe")
  - Student teams show the results of their project
  - Audience are industry partners and fellow students

AMOS Demo Day Process

- Introduction
  - General introduction
  - 1 slide, 1 min. introduction by each team (one speaker)

- Project demos
  - After the introduction, we split up into the projects
  - One presentation booth (in-person) / demo room (online) for each project
  - At least one person from each project needs to be ready to demo
  - Demo day participants (including students without booth duty) roam around
  - There are five demo sessions, and sessions restart every 20 min.

- Conclusion
  - At the end, we come together in the main room to say goodbye

AMOS Demo Day Schedule

- See **Course organization** doc, tab **Demo day schedule**

Presentation Booth / Virtual Demo Room

- If held **in-person**
  - Students are given a "presentation booth"
  - The booth is a table plus pinboards
  - Students demo their work at this table
  - Students explain their work using the posters
  - Students can create additional materials

- If held **online**
  - Students are given a virtual demo room
  - Students provide a back-up video
  - Students demo their work in this room
  - Students explain their work using slides
  - Students can create additional materials

Demo Posters / Slide Deck

- If held **in-person**
  - One product management poster with
    - Project and team name, team logo
    - Short project description
    - Key use cases

  - One software development poster with
    - Software architecture
    - Employed technology
    - Tooling and processes

  - These are **exact** requirements
    - We will print the posters

- If held **online**
  - A slide deck with
    - One product management slide
    - One software development slide
    - One team photo slide (can be screenshot)
    - Slide content like the in-person posters

  - These are **minimum** requirements
    - You can do more if you like

  - What happens in the demo room
    - Stays in the demo room
    - The slides are not deliverables

Poster 1

JOWBANA

Personalfragebogen 2.0
Personnel Questionnaire Automation

Personalfragebogen 2.0* is a personal data management software solution, supporting companies of any size in hiring new employees more efficiently.

The product improves the hiring process by automating the collection of personal data during the hiring procedure, and provides aid in managing the collected data.

Demo Day
Friedrich-Alexander Universität Erlangen
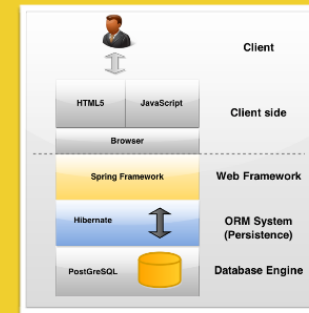**Wednesday, 15 July 2015 10.15-11.45**

*Personalfragebogen 2.0 is a joint project between FAU's OSR Group and DATEV eG.

Poster 2

JOWBANA

Personalfragebogen 2.0
Personnel Questionnaire Automation

Software Architecture

Technology

Name	Function
Spring Framework (4.1.6)	Java based Web Framework
Java SE (7u79)	Fundamental Platform
HTML5	Client-side core technology
Selenium (2.45.0)	UI Testing/Integration Testing
JUnit (4.12)	Java Unit Testing Framework
Hibernate ORM (4.3.9)	ORM System for persistence
PostgreSQL (9.4.1)	Database Management System
Tomcat 7.0.61	For local deployment

*Personalfragebogen 2.0 is a joint project between FAU's OSR Group and DATEV eG.

Use of Corporate Identities

- Please use university logo
- Please use your team logo
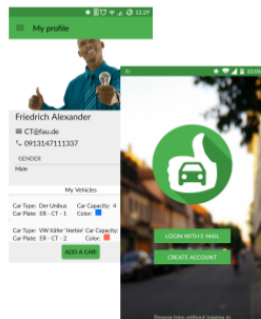- Please use industry partner logo, but ask first

CroudTrip!

sponsored by Elektrobit

The CroudTrip! application wants to revolutionize the car-ride-sharing market with its easy, user-friendly and highly automated way of organizing shared Trips!
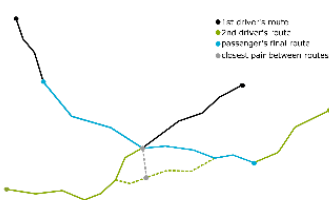
The Product

- Offer and join shared Trips at short-notice!

- For drivers: Easily find passengers on the way you are going anyway … and earn money with it!

- For passengers: Reach your destination comfortably!

- We will automatically match you to the best offer in real-time!

- Simply check-in and check-out of your Trips using NFC on your device!

- No direct Trips? No problem - Join a SuperTrip! with multiple drivers!

The Concept

SuperTrip!

- 1st driver's route
- 2nd driver's route
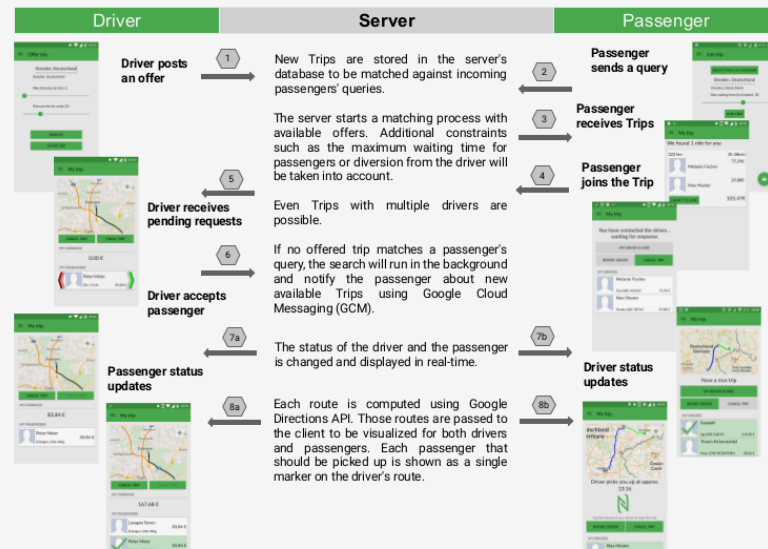- passenger's final route
- closest pair between routes

- Combine multiple offered routes to serve passengers even if there is no direct connection available
- Find routes which can pick up a passenger from his start position or drive to his final destination
- Subdivide those routes, compute the closest pair of those waypoints and use it as a "connection point"
- If the distance of the closest pair is too large, start a recursive matching process with these two waypoints

Multiple Passengers

- driver's original route
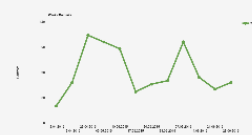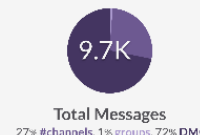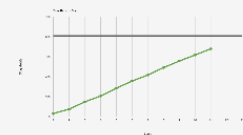- modified route by first passenger
- modified route by second passenger
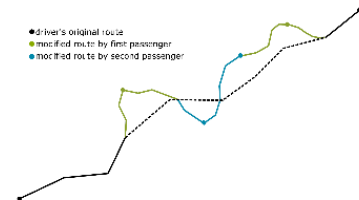
- Match multiple passengers with one driver who will pick them up and bring them to their destinations in an optimal order
- Optimal order is constrained by given internal order of each waypoint pair, because each passenger has to be picked up before the driver reaches his destination location
- Compute optimal order by solving the Travelling Salesman Problem via Brute Force (max. 4 passengers)

The Interactions

Driver	Server	Passenger

Driver posts an offer (1) — New Trips are stored in the server's database to be matched against incoming passengers' queries.

(2) **Passenger sends a query**

The server starts a matching process with available offers. Additional constraints such as the maximum waiting time for passengers or diversion from the driver will be taken into account.

(3) **Passenger receives Trips**

(4) **Passenger joins the Trip**

Driver receives pending requests (5) — Even Trips with multiple drivers are possible.

Driver accepts passenger (6) — If no offered trip matches a passenger's query, the search will run in the background and notify the passenger about new available Trips using Google Cloud Messaging (GCM).

Passenger status updates (7a) — The status of the driver and the passenger is changed and displayed in real-time.

(7b) **Driver status updates**

(8a) Each route is computed using Google Directions API. Those routes are passed to the client to be visualized for both drivers and passengers. Each passenger that should be picked up is shown as a single marker on the driver's route. (8b)

The Process

9.7K

Total Messages
27% #channels, 1% groups, 72% DMs

- Total # of story points: 370
- Development Speed: 30.9
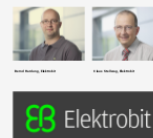- 13.23% of total effort used for bugfixing

- Slack as main communication tool
- Integrations for Travis CI, Github and Crashlytics

- Total # of Commits: 727
- Lines of Java Code: 15362
- Lines of Comments: 3938

The Team

The Sponsor

The University

FAU FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

Presentation Table

- If held **in-person**
  - Mandatory
    - Bring a laptop to demo your project
    - Bring anything else necessary

  - Optional
    - Be creative, do what works!
    - The goal is to explain your project

- If held **online**
  - Prepare for online presentation

Demo Preparation

- Dos
    - Have a clean user interface

    - Use domain terminology and examples
        – In the user interface (labels, titles)
        – In the stories you tell

    - Have a story to tell, for example,
        – A day in the life of …
        – A workflow example

    - Make the demo (data) re-entrant
        – You will have to start over several times
        – You always want to start at the same point

- Don'ts
    - Use "test" or "help" as labels
    - Not follow the advice on the left

Demo Execution

- Have two people at the presentation booth / virtual demo room
  - One talks to people in general
  - One demos to people

12

Thank you! Questions?

dirk.riehle@fau.de – **https://oss.cs.fau.de**

dirk@riehle.org – https://dirkriehle.com – @dirkriehle

DR

Legal Notices

- License
  - Licensed under the CC BY 4.0 International license

- Copyright
  - © 2010-2022 Dirk Riehle, some rights reserved