

Application Development for Mobile and Ubiquitous Computing
Seminar

LO{O}CKOUT

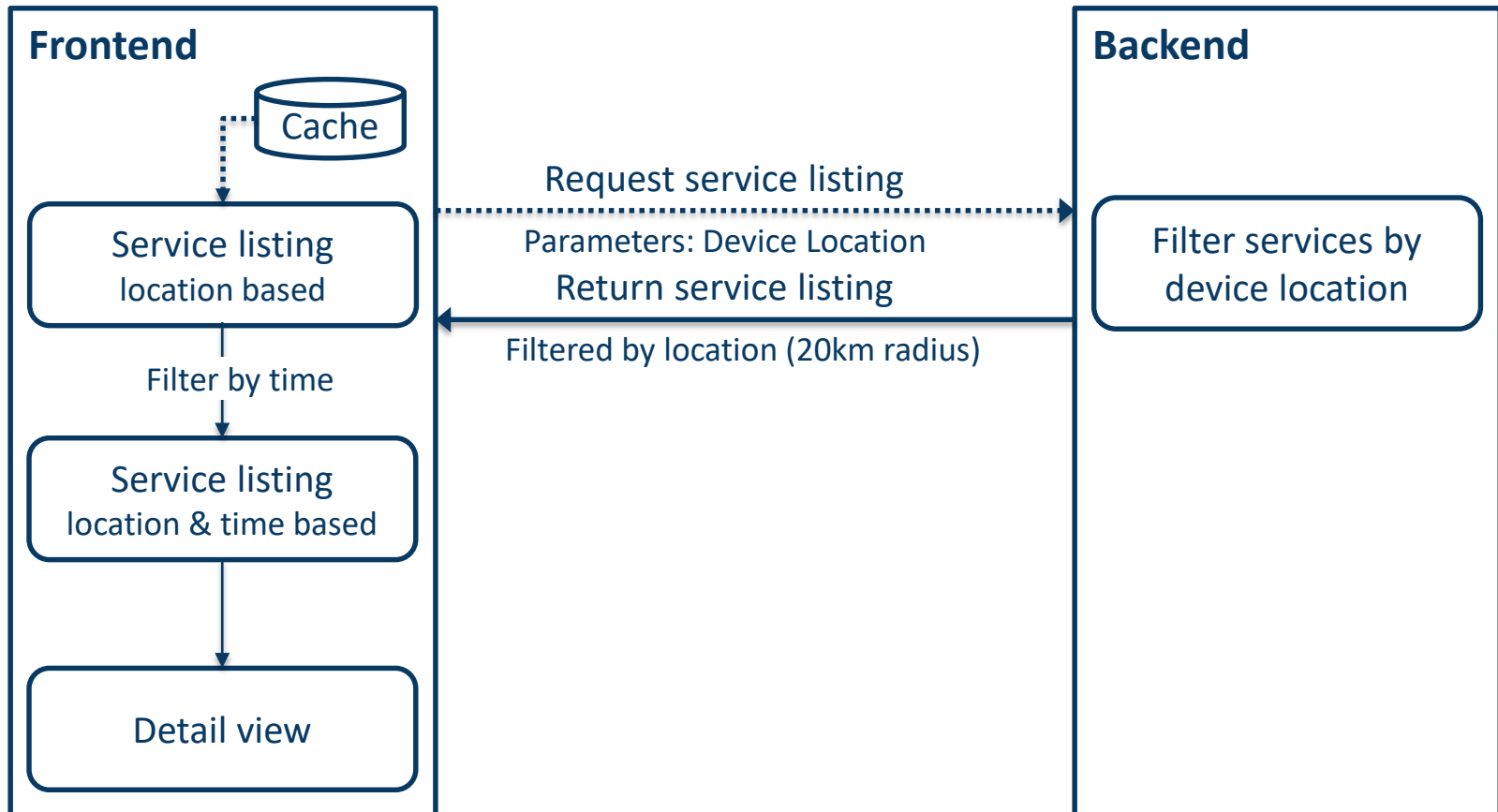
Final Presentation

Christian Deussen, Michael Gruschwitz
Dresden, 27 January 2017

HMMM...WHAT WAS IT ABOUT?

- You are **locked out** of your home
- You need help from a key service nearby
- **lo{o}ckout** will help you to **find a reputable key service** nearby

ARCHITECTURE



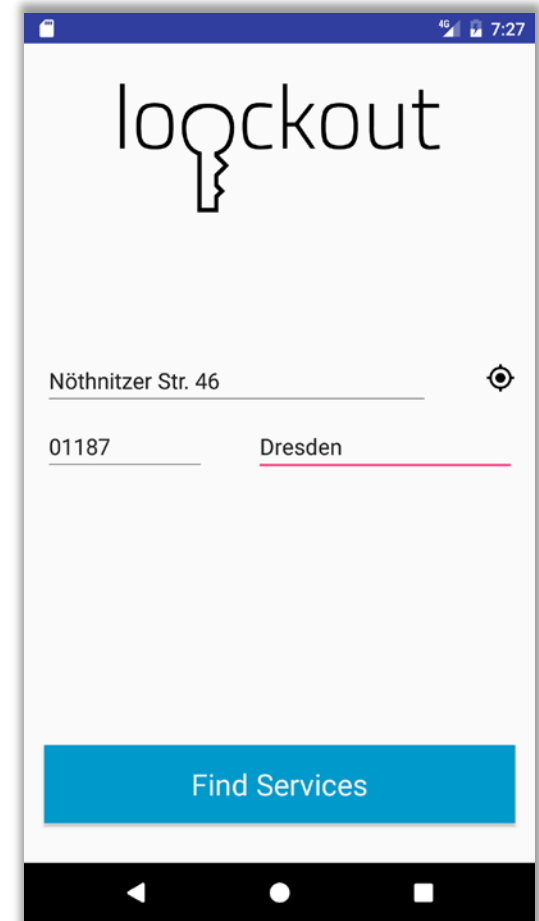
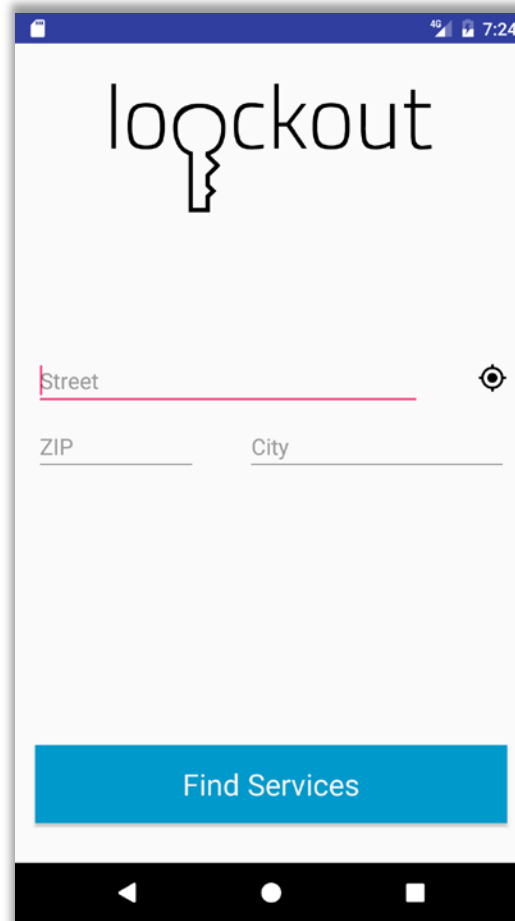
COMPONENTS & TECHNOLOGIES

- Main activity
 - Collect users address either by
 - reading users input
 - location detection using Google Play Services
- List activity
 - Check for cached services
if no cached services available for the address provided by the main activity
 - Get services using the backend API (filtered by location)
 - Put returned services to the cache
 - Filter services by date & opening hours of the services

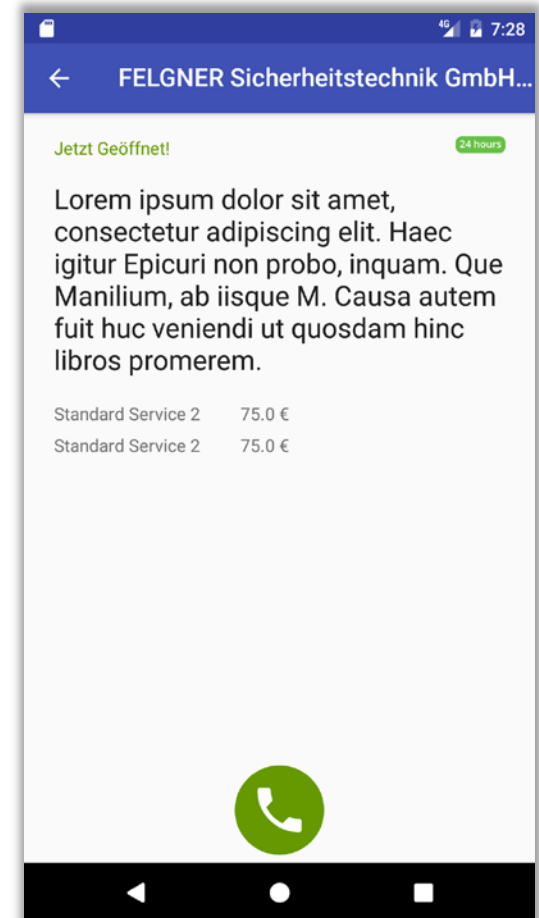
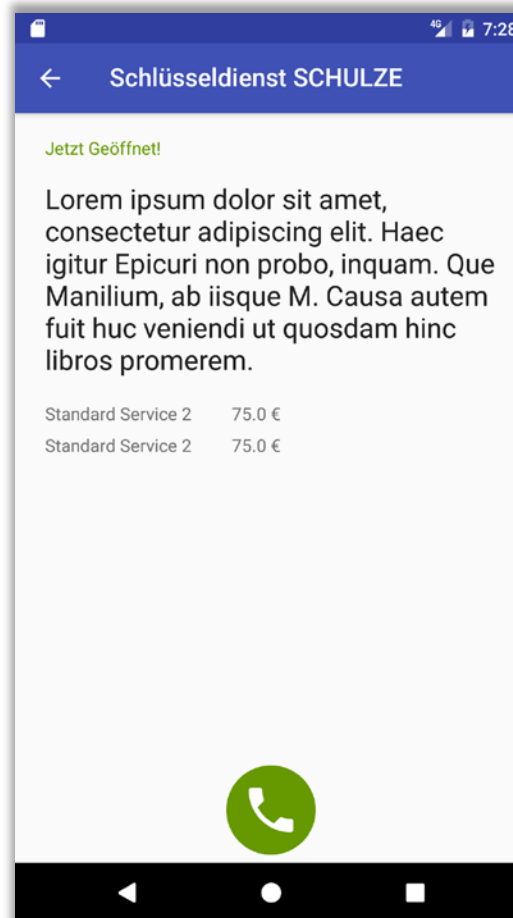
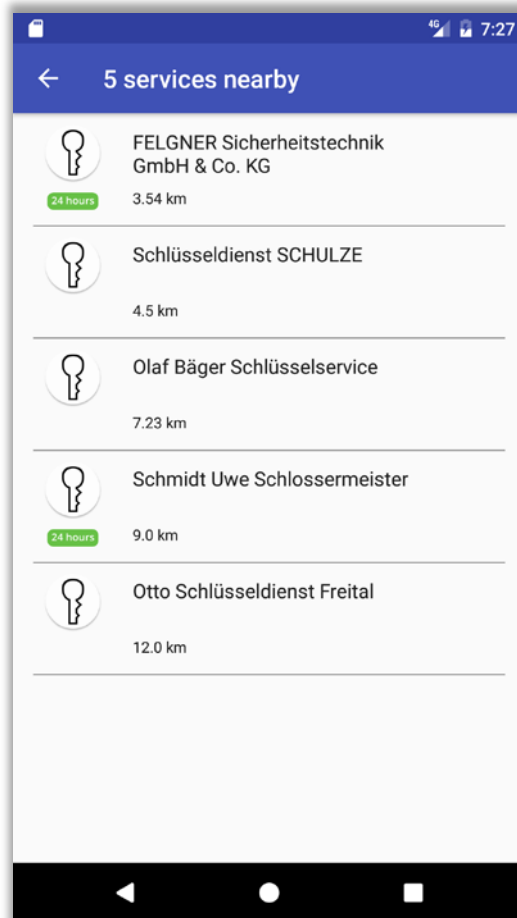
COMPONENTS & TECHNOLOGIES

- Detail activity
 - Show details of selected service
 - Allow user to call service directly

APPLICATION SCENARIO



APPLICATION SCENARIO



CONNECTIVITY CHALLENGE

- Challenge
 - Reduce amount of data to be transferred
- Context: GPS location
- Adaption
 - Reduce amount of services returned by backend API by doing server-side location based filtering
 - Cache services

USABILITY CHALLENGE

- Challenge
 - App should be usable with different screen sizes (phone, tablet, ...)
 - Show only relevant services (opening hours)
- Context: screen size & current time
- Adaption
 - Adapt layout for different screen sizes (ConstraintLayout)
 - Filter services by current time and their opening hours

POWER STATUS CHALLENGE

- Challenge
 - Reduce battery usage
- Context: battery status
- Adaption
 - Power cord connected: refreshing services based on current users location
 - Else: disable background refresh

OPEN ISSUES & LESSONS LEARNED

- Use constraint layouts instead of linear layouts in list & detail views
- „Android fragmentation“
 - KitKat Lollipop, Marshmallow, Nougat
 - Devices -> screen sizes
- Android Studio helps a lot to get started quickly (ready-to-use ecosystem)

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.1%
4.1.x	Jelly Bean	16	4.0%
4.2.x		17	5.9%
4.3		18	1.7%
4.4	KitKat	19	22.6%
5.0	Lollipop	21	10.1%
5.1		22	23.3%
6.0	Marshmallow	23	29.6%
7.0	Nougat	24	0.5%
7.1		25	0.2%