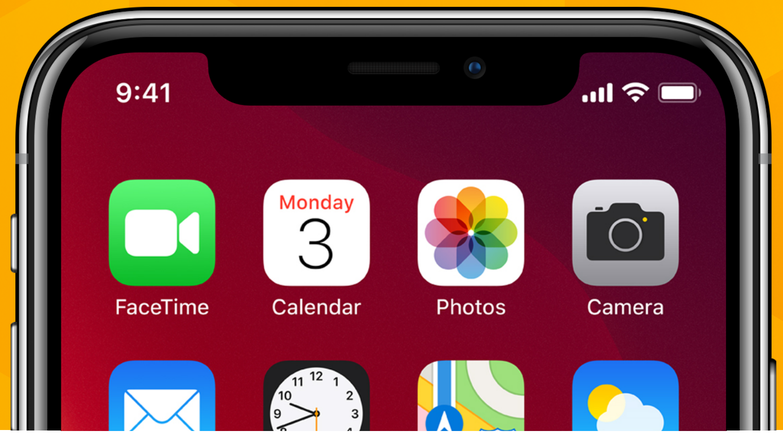


iOS Application Development

Lecture 1: Introduction

Prof. Dr. Jan Borchers
Media Computing Group
RWTH Aachen University

Winter Semester ’24/’25

hci.rwth-aachen.de/ios

RWTH AACHEN UNIVERSITY

Team

Prof. Dr.
Jan Borchers

Kevin
Fiedler

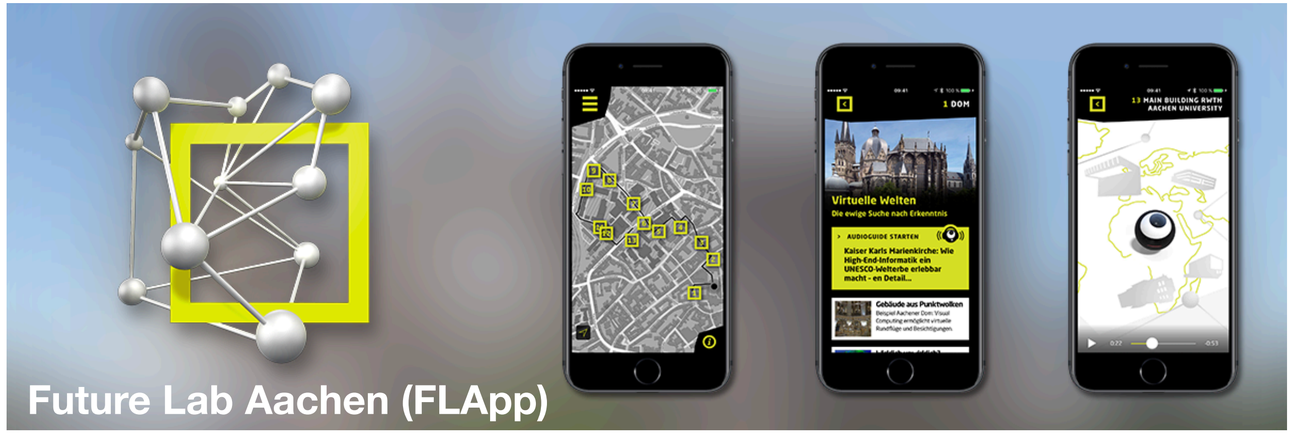
kfiedler@cs.rwth-aachen.de
E-Mail Subject: [iOS]

Class Goals

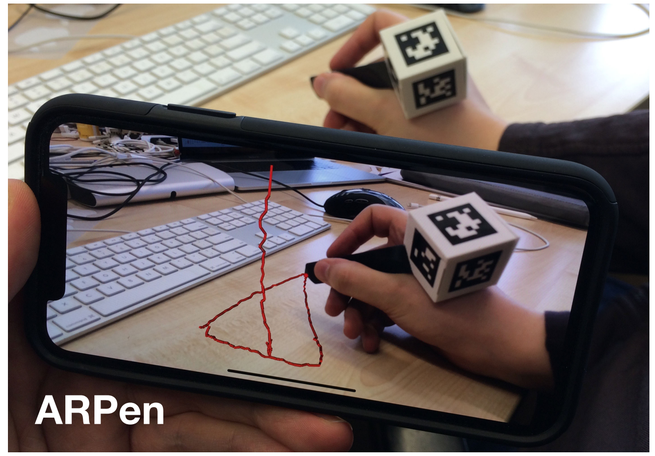
- Understand the differences between desktop and mobile development
- Learn to write iOS Apps (UIKit & SwiftUI)
- Learn how to use advanced iOS frameworks

- During your project: extra focus on UX and usability
- Do lots of coding!

Some of our iOS Projects



iEat



Centre Guide

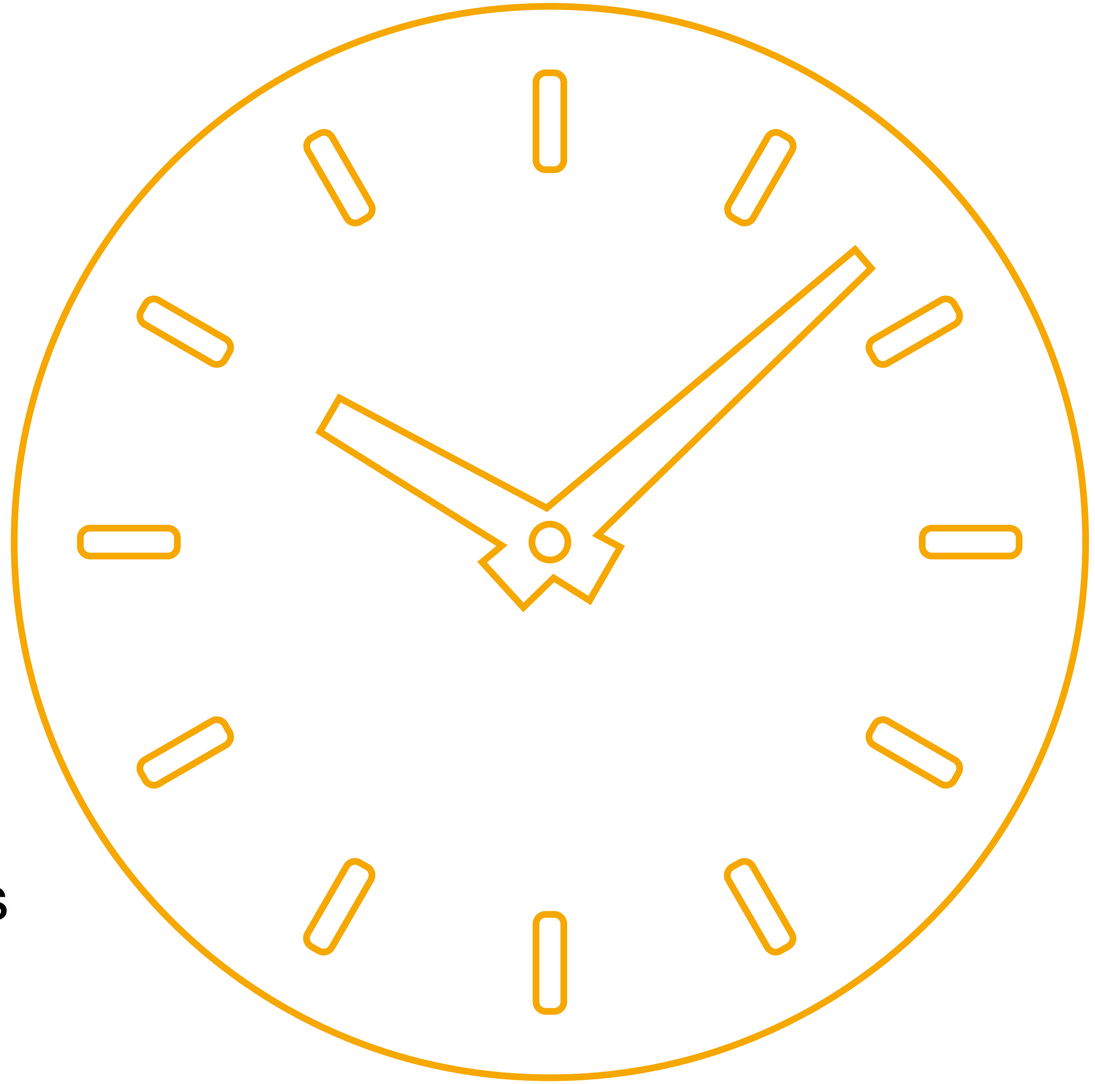
Corona

ForcePicker

Aixplorer

Administrivia

- 6 credit points
- Class times
  - Mondays (12:30–14:00),
    i10 seminar room 2.73
  - Tuesday (10:30–12:00),
    i10 seminar room 2.73
  - Already over 95 registrations, but 42 seats
- hci.rwth-aachen.de/ios



Course Requirements

- Required
  - OOP experience
  - Apple Developer ID
  - You must be registered for this course in RWTHonline
  - Mac capable of running Xcode 16*
- Optional (but helpful)
  - iOS device running iOS 18* (simulator sufficient for many tasks)
  - Previous experience with Swift or Objective-C
- This course is only for students of:
  - B.Sc. and M.Sc. Computer Science
  - SSE / MI / DS / MTIK (TK)

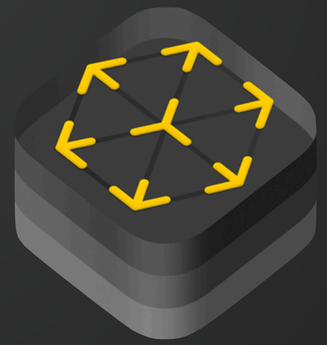
* We have a **FEW** time-share devices

Class Syllabus

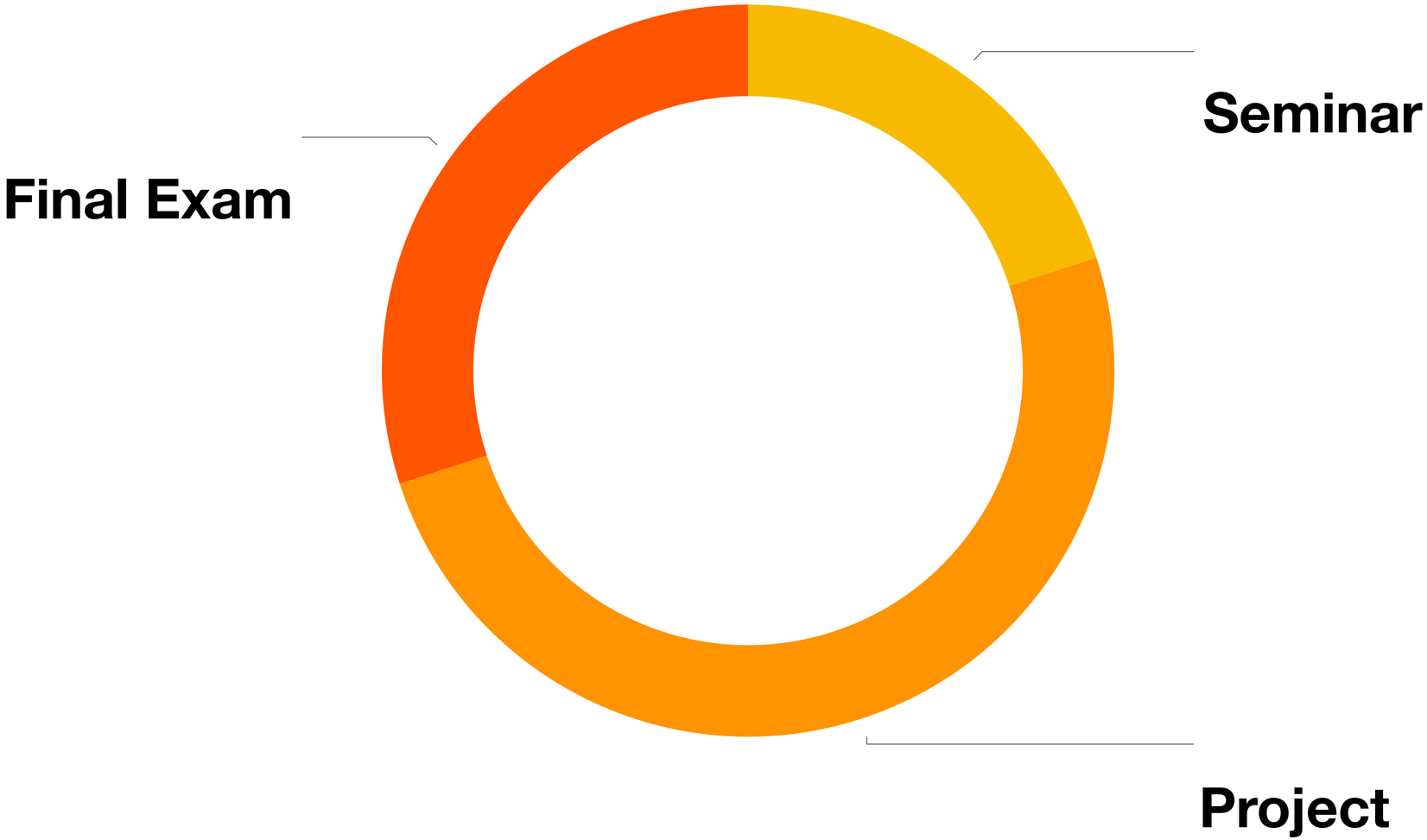
- Part 1
  Lectures: Introduction into Swift and iOS

- Part 2
  Seminars: iOS Frameworks

- Part 3
  Project: App Development



Your Final Grade (6 ETCS)



Lectures

- Swift
- Xcode
- UIKit
  - Navigation, Animation, Customization
  - Collections, AutoLayout
  - Persistence and Networking
- SwiftUI

Textbooks

Available for free on Apple Books

Seminar

Sample Topics:

- CoreAnimation
- Haptics and Sounds
- SpriteKit
- Working with Files
- Combine
- Debugging in Xcode
- WidgetKit and Siri
- WKWebView
- MapKit
- UIPresentationController
- CoreML & CreateML
- CoreData
- watchOS Apps
- SceneKit
- Advanced SwiftUI Layout
- Metal

- 15-minute presentations
- 18.11.2024–9.12.2024
- Overview of the framework
- Example applications
- Attendance is mandatory
- Missing 1 seminar day (>10%) is a 5.0

Project

- Kickoff: 9.12.2024 (after the seminar talks)
- Topics announced at kickoff date
- Presentations: 27.01.2025
- 2h slot instead of 1.5h

- Submittable app
- Final presentation with live demo
- Code documentation!
- Attendance on Presentation Day is mandatory
- Missing Presentation Day (>10%) is a 5.0

Written Exam

- Questions about:
  - Lecture content
  - All seminars

- Exam dates:
  - 1st exam: Sat. 03.02.2025
  - 2nd exam: Wed. 05.03.2025
- 60 minutes

Limited Seating

- We only have **42 seats** available
- Attending the seminar and project presentations is **mandatory**
- **Only take this class if you are sure you can attend all classes**

How to Get a Seat

- **Register** in RWTHonline & send the **Declaration of Compliance** before **Wednesday** (09.10.2024) **13:00**
- We will invite everyone who did this to **Moodle** to **form groups**
- Form groups of **3 students** in Moodle (“Seminar Groups”)
  - There should be at least **2 students with a Mac** per group
- On **Friday 11.10.2024 13:00**, we will randomly select 14 **full** groups
  - Everyone else will be removed from RWTHmoodle

Lecture Slides and Videos

- All slides and lecture videos will be available on RWTHmoodle

Developing for Mobile Platforms

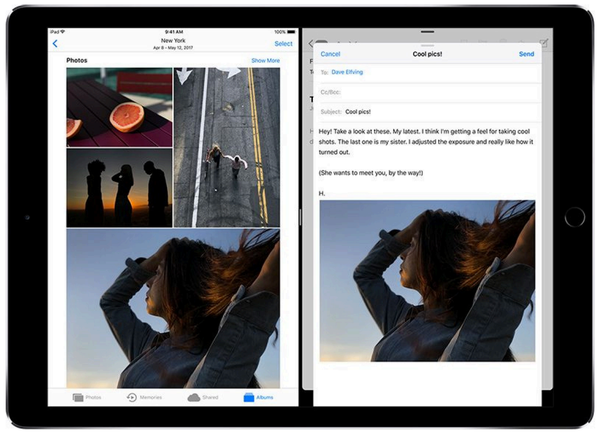

Finder File Edit View Go Window Help

VS.


9:41
FaceTime
Monday
3
Calendar
Photos
Camera
Mail
Clock
Maps
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
TV
Health
Home
Wallet
Settings

Mobile Device Characteristics

- Context is key: task, attention split, peripheral use, movement, interaction time
- Small screens
- Users interact with one screen at a time
  - And mostly one application at a time (iPadOS allows split screen)
- Onscreen help is minimal
- Input: direct touch, pen, camera, microphone, sensors



10 Golden Rules of Interface Design (see DIS 1)

1. Keep the interface simple
2. Speak the user's language
3. Be consistent and predictable
4. Provide feedback
5. Minimize memory load
6. Avoid errors, help to recover, offer undo
7. Design clear exits and closed dialogs
8. Include help and documentation
9. Offer shortcuts for experts
10. Hire a graphics designer

The iOS Family

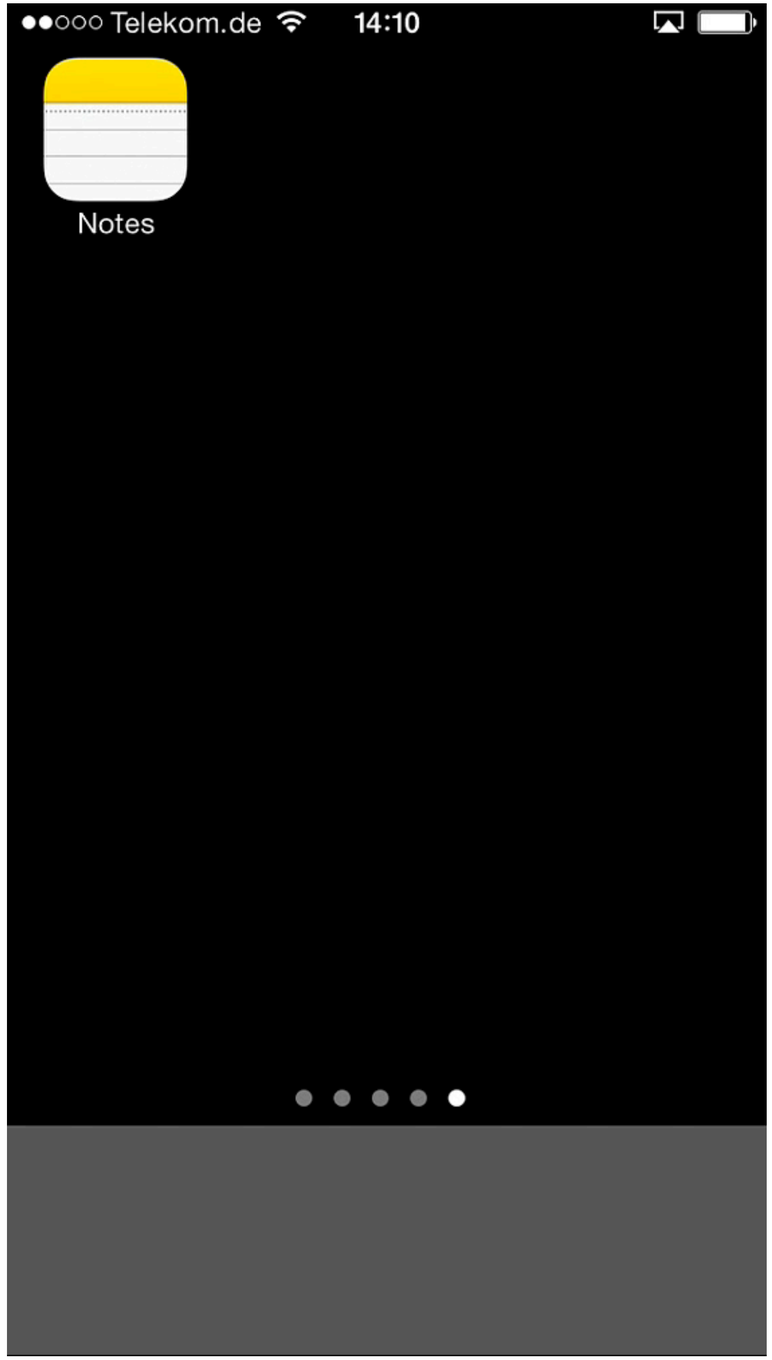
Life as an App

- iOS is an app-centric environment
- One app per task!
  - Define the task that users want to accomplish with your app
  - Do *one* thing, but do it well
- Data is stored per app
- Exchanging data between apps is difficult

Designing the UI

- Make it obvious how to use your application
- Sort information from top to bottom
- Use visual weight for relative importance
- Use alignment for groupings or hierarchy
- Use multiple views for different parts of the app

Example



Designing the UI

- Text should always be legible
- Avoid inconsistent appearances
- Provide fingertip-size targets (8 x 8 mm)
- Minimize text input

Interaction Design

- Smartphones are mostly used with only one hand

- Not everyone is used to multitouch interaction
  - Only use multitouch if you need it

- Follow Apple’s examples if possible

- If you use complex gestures, help the user

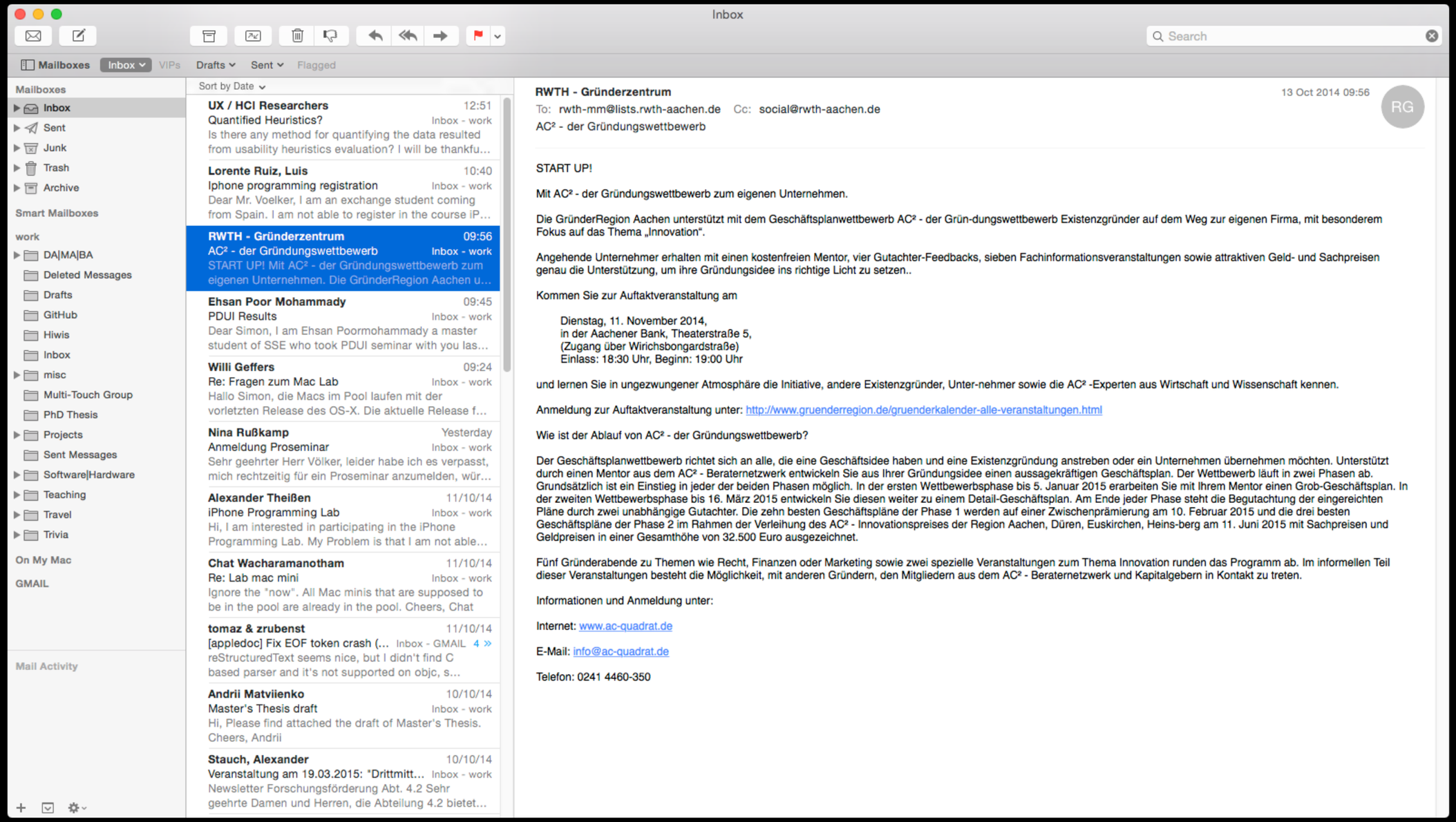
Standard Gestures

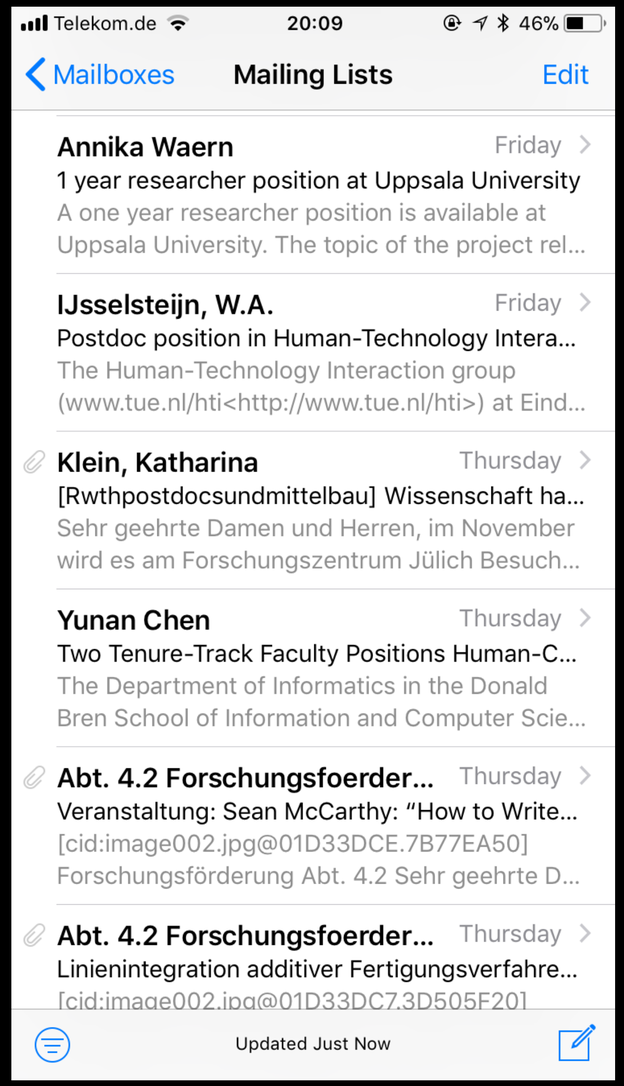
	Tap	To press or select a control or item (analogous to a single mouse click).
	Drag	To scroll or pan.
	Flick	To scroll or pan quickly.
	Swipe	In a table-view row, to reveal the Delete button.
	Double tap	To zoom in and center a block of content or an image. To zoom out (if already zoomed in).
	Pinch open	To zoom in.
	Pinch close	To zoom out.
	Touch and hold	In editable text, to display a magnified view for cursor positioning.

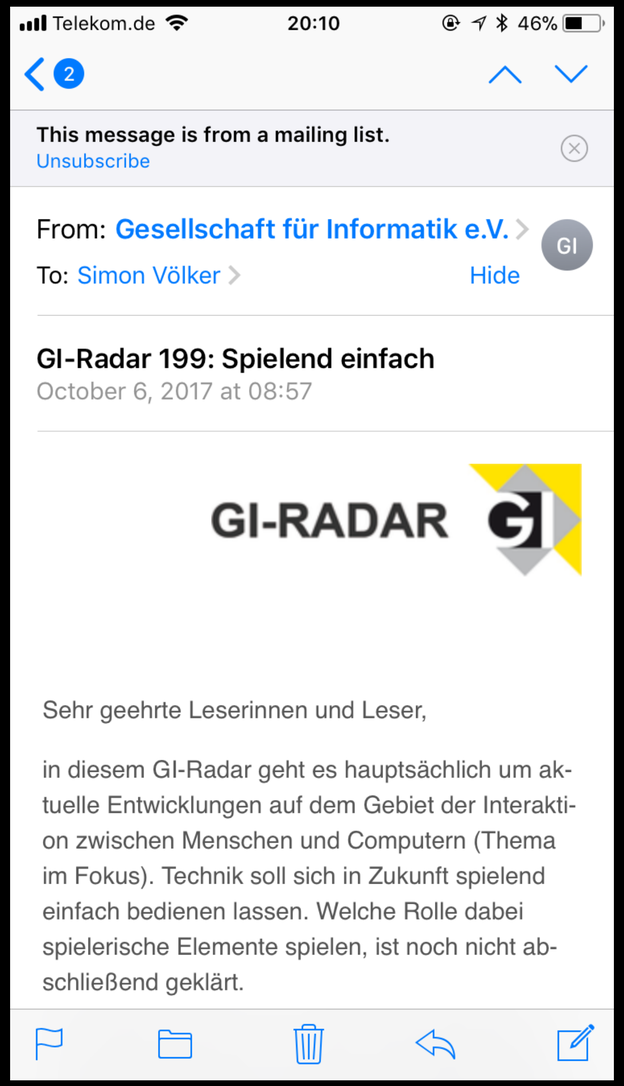
Designing the UI

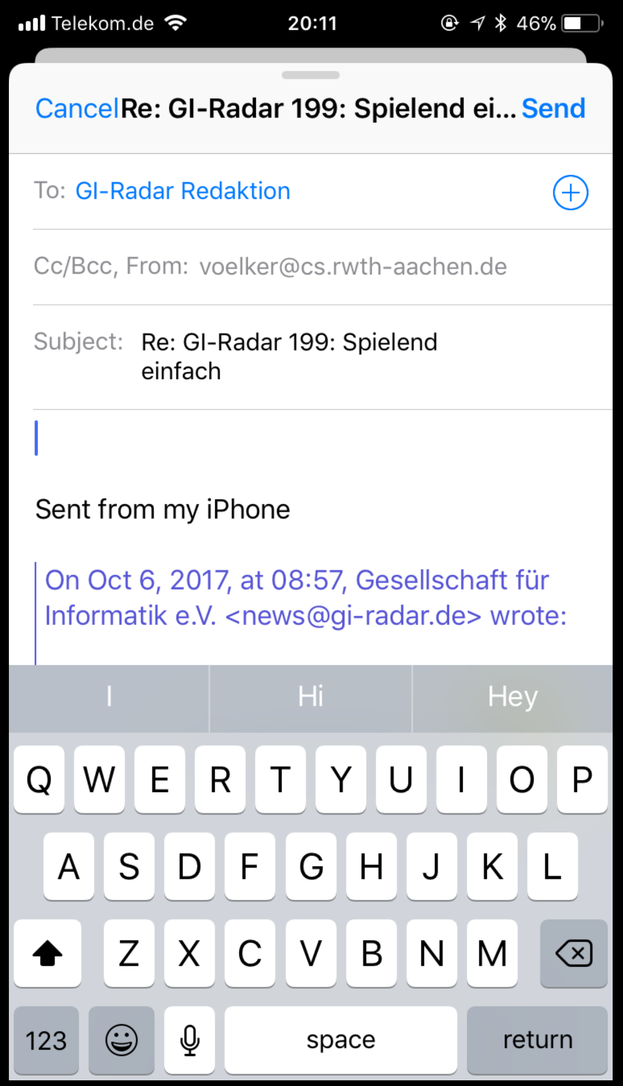
- Several resolutions, about four aspect ratios:

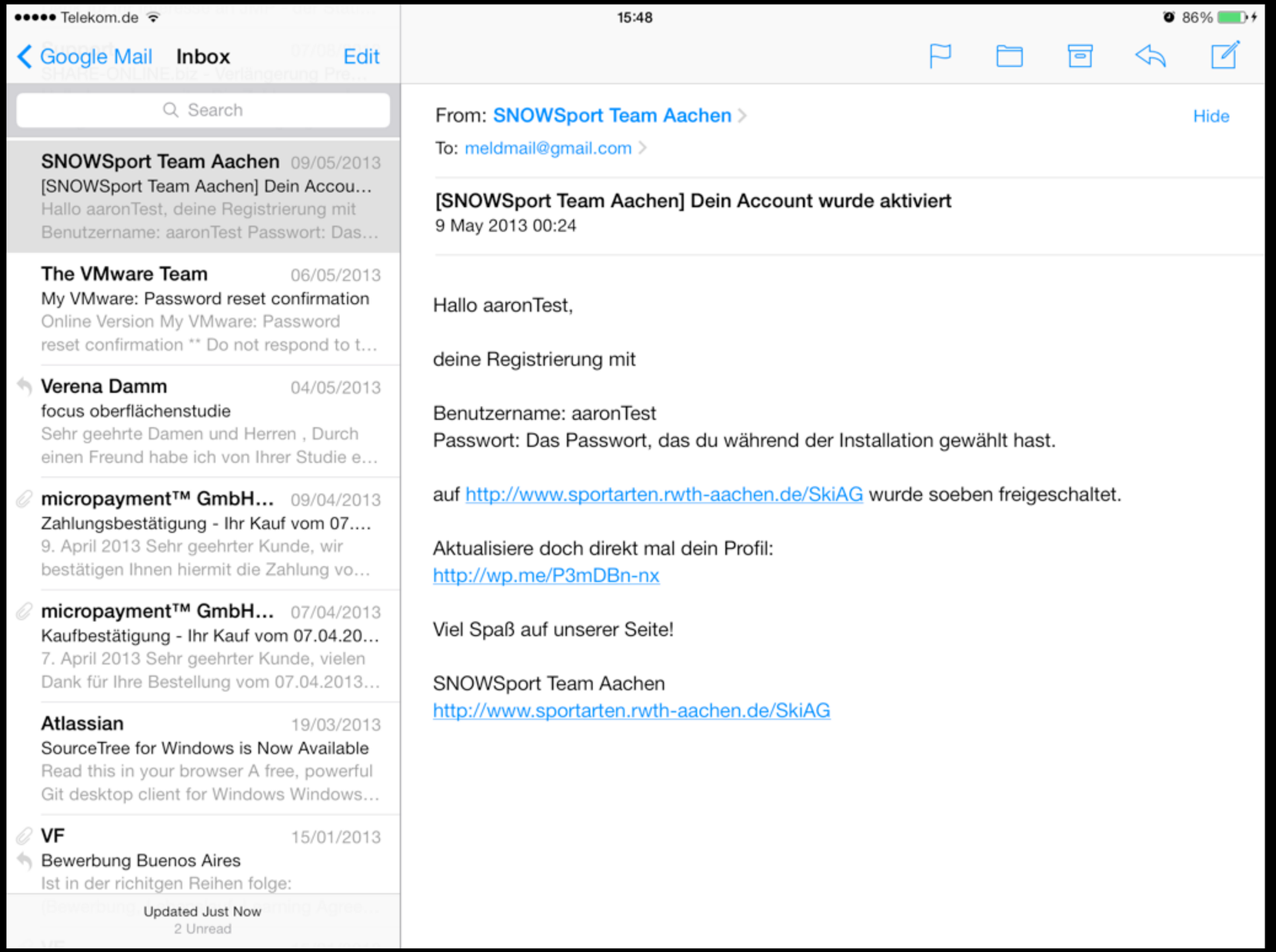
- Device orientation: portrait or landscape
- Designing for the iPad requires more than increasing the resolution

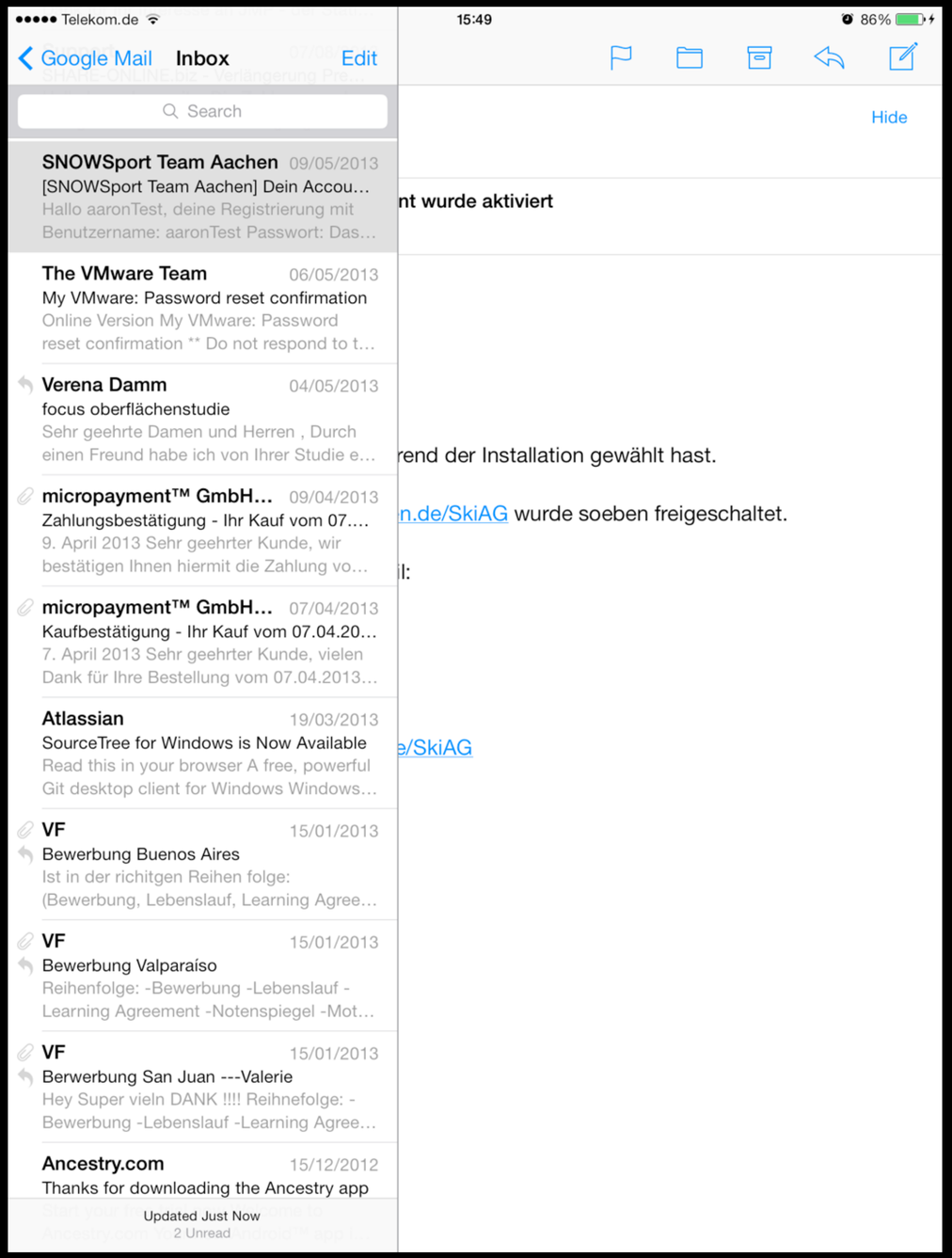










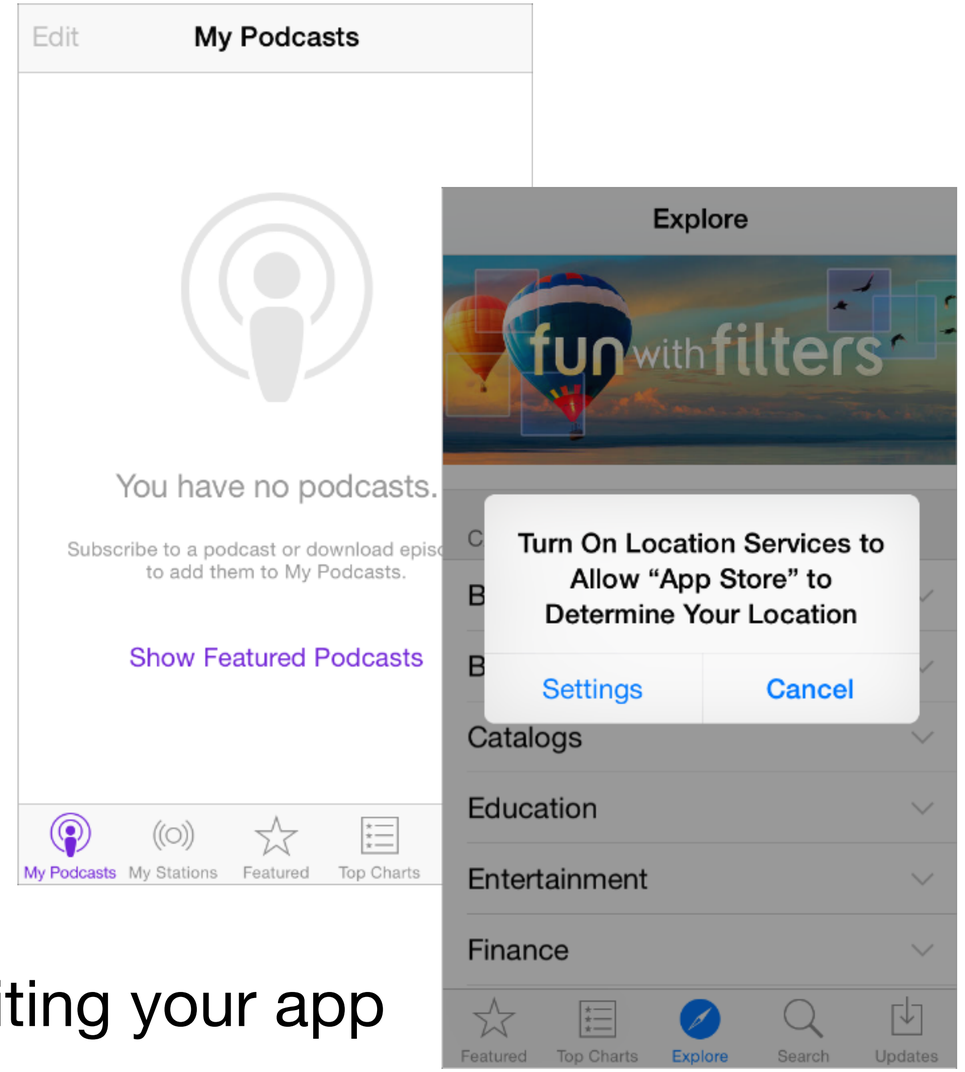


Launching

- Apps should start quickly to provide a fluid user experience
- Show a launch image that closely resembles the first screen of your app (launch file) — no splash screens
- Restore the state of last run (minimize user input)
- Delay a login requirement for as long as possible
- By default, launch in device's current orientation
- Think carefully before providing an onboarding experience
- See **Apple's Design Patterns: Launching**

Stopping

- No Quit button or menu item
- Be prepared to quit at any time
- Program flow interrupted by external events
  - Incoming phone call
- Store state when stopping
- Application moved to background
- Notify users what feature are unavailable and limiting your app



Application Styles

Productivity

Photos

Utility

Weather

Immersive

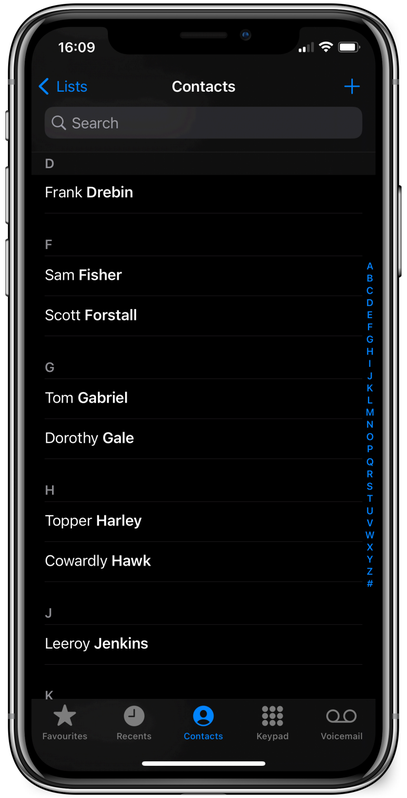
Seadragon

Productivity Applications

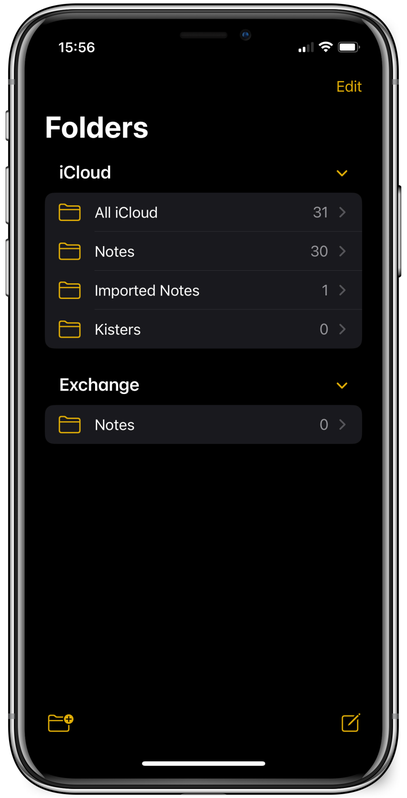
- Organizing and managing detailed information
- Often organize data hierarchically
- Organizing the list, add or remove items

Productivity Applications: Examples

Photos



Contacts

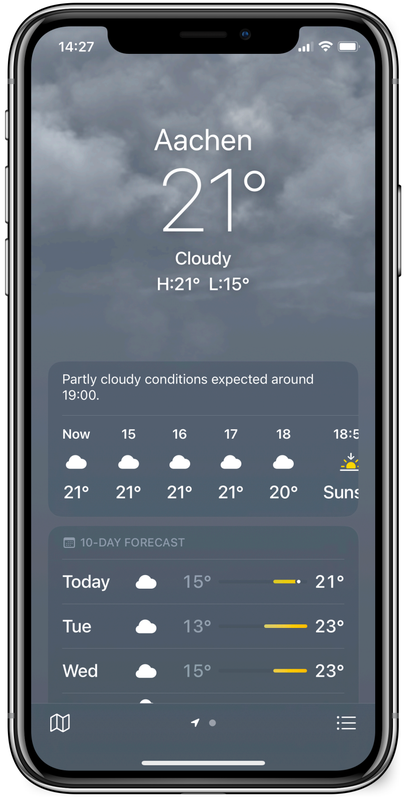


Notes

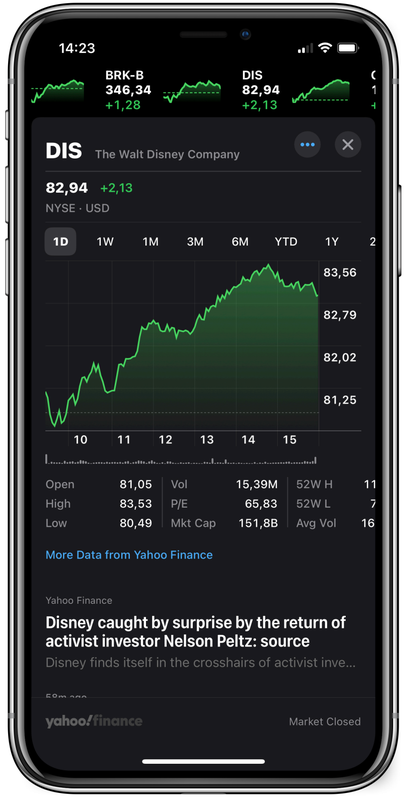
Utility Applications

- Simple task, minimum user input
- Customized, visually attractive UI that enhances the displayed information
- Data is organized in flattened list of items

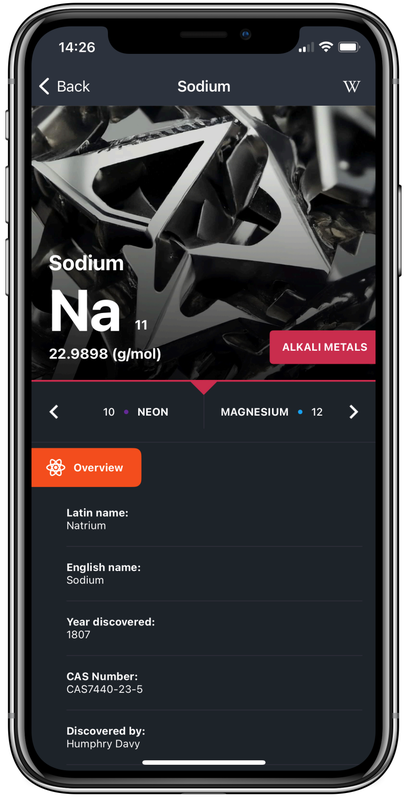
Utility Applications: Examples



Weather



Stocks



Periodic Table

Immersive Applications

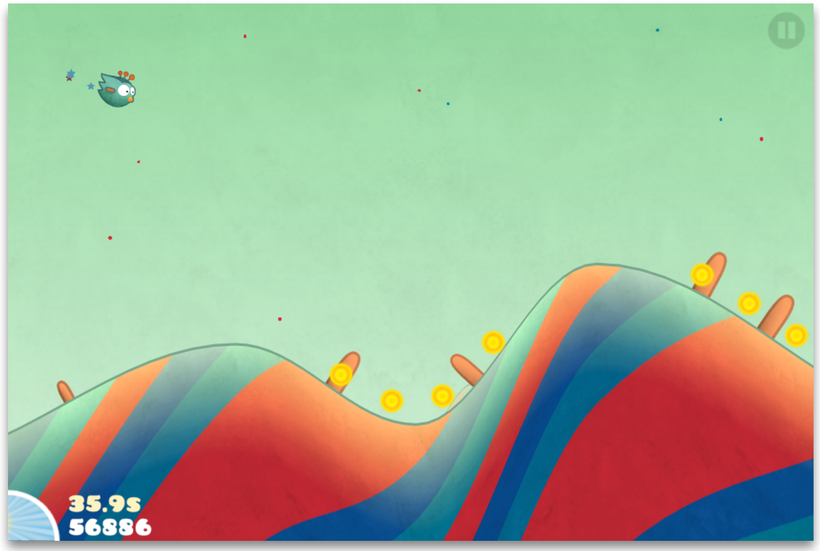
- Full-screen, visually rich UI
- Focussed on content and user experience
- Tends to hide much of the device’s user interface
- Custom navigational methods

Immersive Applications: Examples

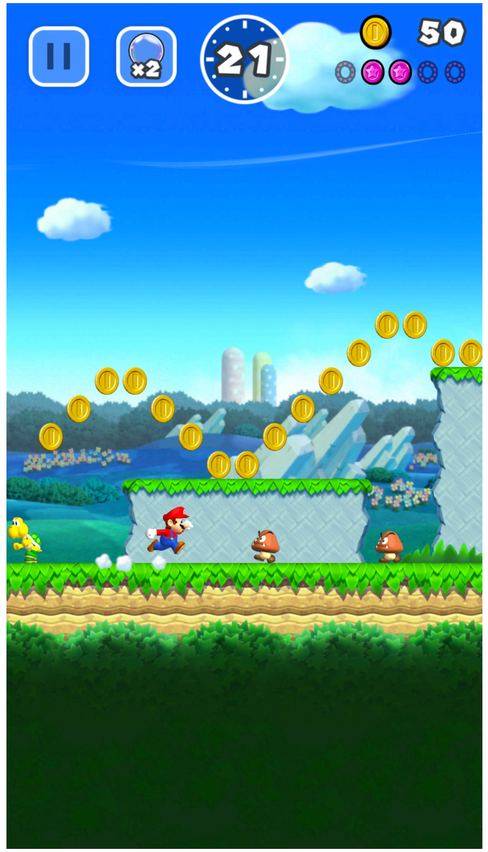
Weather Now

Tape Measure

Immersive Applications: Examples



Tiny Wings



Super Mario Run

The Machines

Summary

- Mobile vs. desktop apps: user, **task**, **context**
- Keep hardware restrictions in mind
- App-centric vs. document-centric
- Application styles: productivity, utility, immersive
- Further reading:
  - iOS Human Interface Guidelines: https://developer.apple.com/design/human-interface-guidelines

What's Next?

- Register, and sign the Declaration of Compliance and upload it to the sciebo folder.
- Next lecture: Tomorrow, 08.10.2024 10:30–12:00
  - Presentation of seminar topics
  - Introducing Swift
- Remember the class page: hci.rwth-aachen.de/ios