Center for
Human-Computer Interaction
University of Salzburg

UNIVERSITÄT
SALZBURG

User Interface Engineering

Alexander Meschtscherjakov

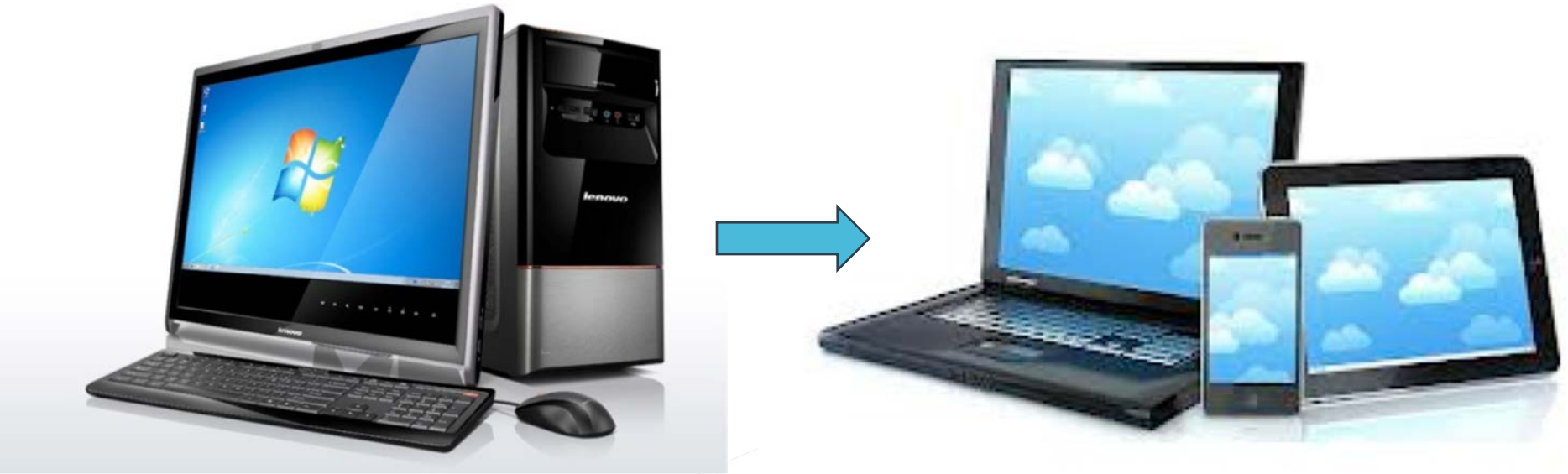
Wiederholong

- Visuelles Design (Zielgruppe, Hierarchie, Präsentationsart, Kommunikationsziel)
- Werkzeuge (Punkt, Linie, Form, Muster, Textur, Raum, Größe, Bilder)
- Typografie
- Farbe
  - http://colorschemedesigner.com/csd-3.5/
  - http://paletton.com
  - https://color.adobe.com
- Visuelle Hierarchie
- Weißraum
- Konstanz und Variation
- CRAP Design Prinzipien

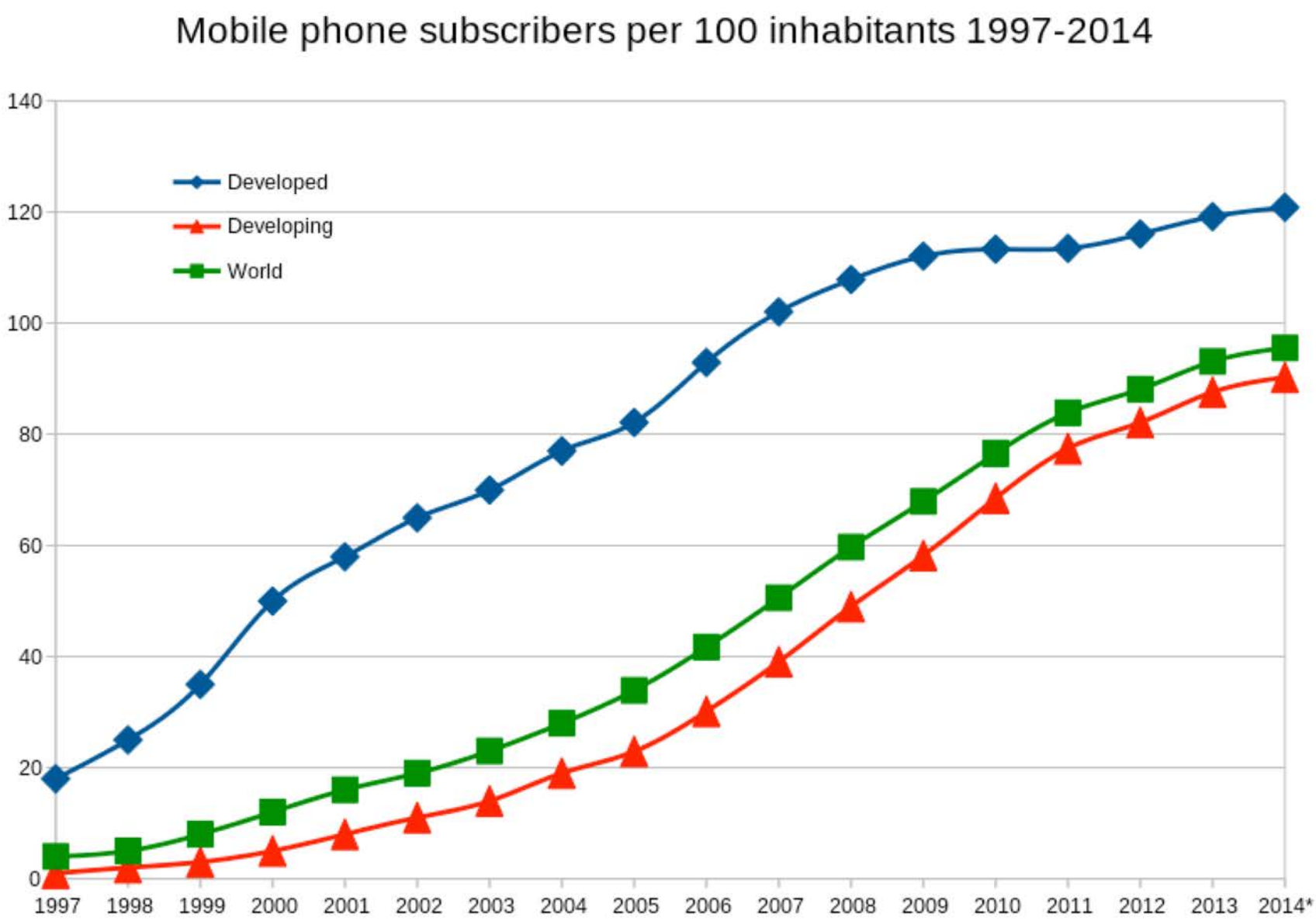
Übersicht – Mobile Interfaces

- **Geschichte des Mobile Computing**
- Interaktionsstile für mobile Geräte
- Touch Screen Interaction
- Webseiten-Design für Touch-Screens

Mobile Computing



2010 wurden weltweit zum ersten mal mehr Smartphones als Personal Computer verkauft.


Mobile phone subscribers per 100 inhabitants 1997-2014
Developed
Developing
World
140
120
100
80
60
40
20
0
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014*

Evolution des Mobiltelefons

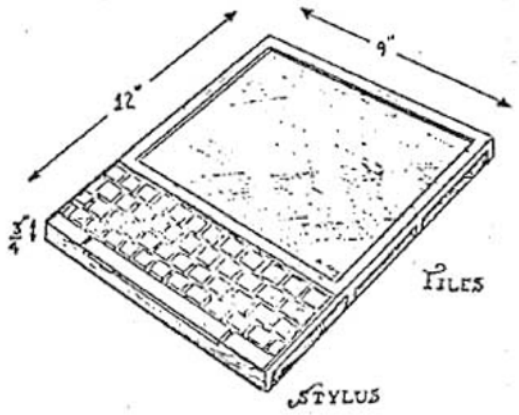
© via https://storify.com/yonathan32/the-evolution-of-cell-phones

7 Phasen des Mobile Computing [Kjeldskov2013]

1) Portability
2) Miniaturization
3) Connectivity
4) Convergence
5) Divergence
6) Apps
7) Digital Ecosystems

Phase 1 – Portability

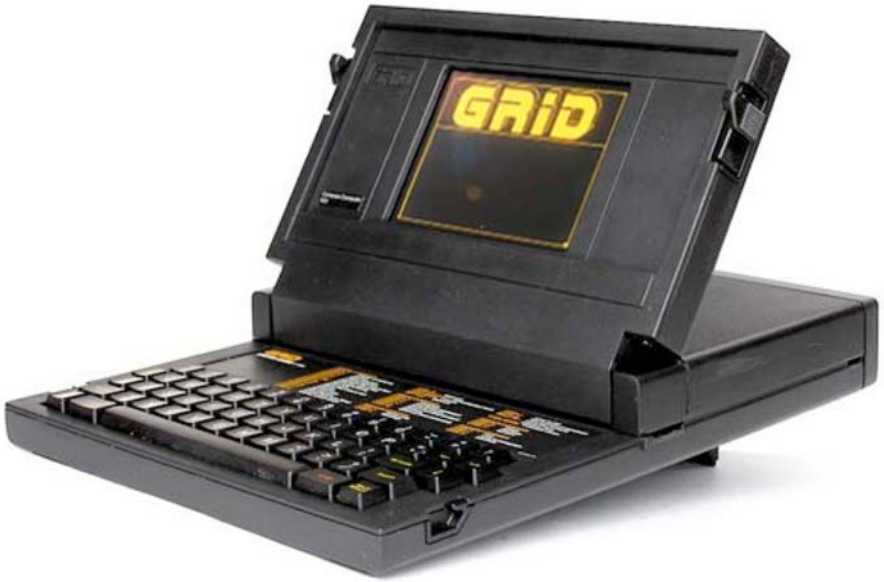
- First mobile computers, „early" laptops, were developed in the late 1970s and early 1980s inspired by the portability of Alan Kay's Dynabook concept from 1968 [Kay1972]
- The Dynabook concept was originally thought of as a machine for children



Phase 1 – Portability

- First laptop computer was the GRiD Compass 1101 designed by Bill Moggridge in 1981
- 16MHz Intel 8086 processor
- 256K DRAM
- 6-inch 320x240 pixel flat display
- 1200 bit/s modem
- 5 kg
- GRiD OS

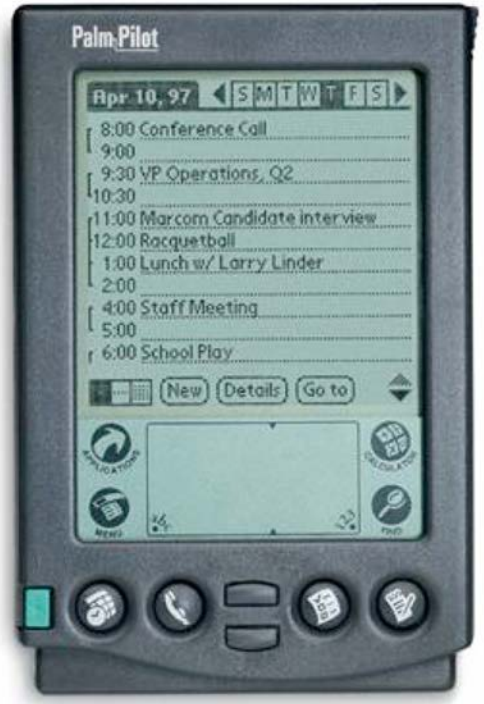
- NASA used GRID for Space Shuttle missions



Phase 2 – Miniaturization

© Apple Inc.

1992 Apple Newton



© Palm Inc.

1997 PalmPilot

contextual.interaction.design.research

Phase 2 – Miniaturization

© Psion

1997 Psion 5

© Compaq

2000 Compaq iPaq

Phase 2 – Miniaturization

- CPU miniaturization allowed for smaller computers in the early 1990s and for new and smaller form factors
- Personal Digital Assistants (PDAs)
- Users could operate while actually moving around physically
- No alternatives to desktop or laptop computers, but supplemental devices for businessmen

Phase 2 – Miniaturization

- Focus on applications and interaction styles designed specifically for mobile devices and mobile users
- Small touch-sensitive screen and a separate pen (or stylus) for user interaction
- On-screen keyboard or handwriting recognition software
- Function buttons, navigation keys, "one-click" dock for synchronizing

Phase 3 – Connectivity

© Motorolla

© Nokia

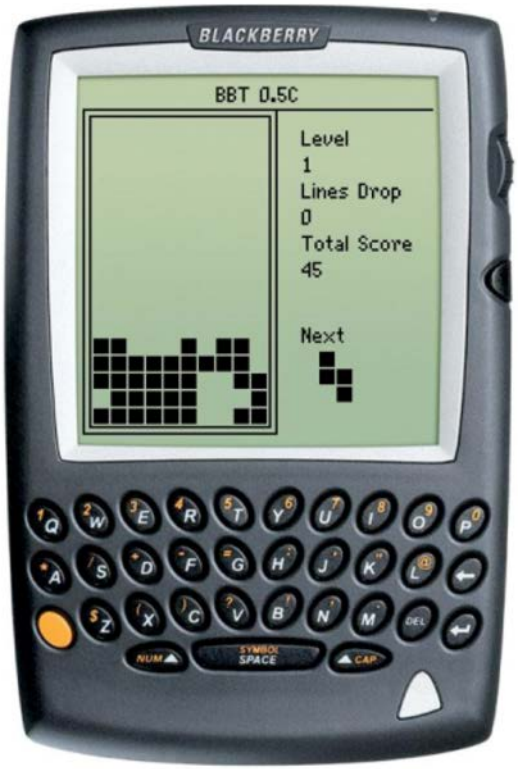
© Nokia

Phase 3 – Connectivity

- Origins in wireless telecommunication
- 1973, Motorola team led by Martin Cooper patented a handheld mobile phone concept
  → DynaTAC 8000X (1983)
- 1991 Global System for Mobile Communications (GSM) included the Short Message Service (SMS)
- Broad population worldwide
- Challenge of interaction design not only for making phone calls, but also for handling contacts, calendars, text-based messages, and browsing the Internet
- In late 1990ies, cell phone design and market was dominated by Nokia
- Tiny low-resolution displays, 12-key numeric keypad, small number of function and navigation keys
- Simple graphical menu system and the "Navi-key" concept, T9 predictive text, games, customisable ring tones, changeable covers
- WAP (Wireless Application Protocol) for internet access

Phase 4 – Convergence

© IBM
IBM Simon



© Blackberry

Phase 4 – Convergence

- Specialised mobile devices converged into new types of hybrid devices with fundamentally different form factors and interaction designs
- Emergence of "smart phones" combined the functionality of a PDA with that of a mobile phone
- IBM Simon from 1992 (no physical buttons)
- Blackberry: PDA size display and a miniature QWERTY keyboard (used for calendars, addresses, notes, e-mail, fax, and games)
- 2nd phase of convergence combined mobile phones with various rich media capabilities
  (cameras, music players, video recording, TV, radio)
- First mobile phone feature a digital camera was the Sharp J-SH04 from 2001
- 2009: 1.9 billion camera phones in existence
- Swiss army knife effect: clumsy technology with a wide range of functions, none of which are ideal

Phase 4 – Convergence

© Sharp

2001 Sharp J-SH04

© Nokia

Nokia N90

contextual.interaction.design.research

Phase 4 – Convergence

© Nokia

Nokia N-Gage

© Sony Ericcson

Sony Ericsson W600

Phase 5 – Divergence

© Apple

Apple iPod

© Sony

Sony PSP

Phase 5 – Divergence

- Contrasting the convergence approach, the trend of divergence suggested a single function/many devices or "information appliance" approach where each device is "designed to perform a specific activity, such as music, photography, or writing" [Bergman2000]
- Having a wide range of good specialised tools is better than a general one that does not perform any task particularly well
- Trend of divergence in the early 2000s was a deliberate interaction design choice and not a technological necessity

Phase 6 – Apps

© Apple

Phase 6 – Apps

© Apple

Phase 6 – Apps

- 2007, Apple launched the iPhone
- Converged mobile device (camera phone, media player, Internet client with e-mail, web browsing)
- iPhone represented a significant rethinking of the design of mobile interactions and a series of notable interaction design choices
- Large high-resolution capacitive multi-touch display with simple gesture capabilities
- Almost no physical keys and stylus
- No deep hierarchies of menus
- Embedded context sensors (e.g., portrait mode)
- GPS and digital compass → "context-awareness" capability, location-based services
- Web browser
- Dedicated applications purchasing music from the iTunes Store
- *Preferred* gateway to the Internet
- Cloud computing services

Phase 6 – Apps

- In 2008 Apple launched the online App Store
- Easy download, and pay for, third-party application content directly from mobile device
- Free iPhone OS software development kit (SDK)
- Business model: Apple handles payments and distribution while leaving App creators with 70% of the profit
- By 2012, more than 25 billion Apps had been downloaded
- Prior to the iPhone, downloading and installing software onto a mobile phone or PDA was difficult
- Improving hardware less important than improving the software

Phase 6 – Apps

- 2010, Apple launched the iPad
- iPhone OS instead of classical PC OS
- New category of mobile devices that are not just laptops without keyboards
- In March 2011 more than 65.000 applications available for iPad

Phase 6 – Apps

- Android was unveiled in 2007, along with the founding of the Open Handset Alliance – a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices
- By July 2013, Google Play store has had over one million Android applications published, and over 50 billion applications downloaded.
- A 2013 survey of mobile application developers found that 71% of developers create applications for Android, and a 2015 survey found that 40% of full-time professional developers see Android as their priority target platform, which is comparable to Apple's iOS on 37% with both platforms far above others.
- By 2016, Android was on the majority of smartphones in virtually all countries in the world, excluding United States and Canada (while including North America continent as a whole), Australia and Japan. A few countries, such as the UK, lose Android-majority if tablets are included.

Phase 7 – Digital Ecosystems

Phase 7 – Digital Ecosystems

- Technical capabilities of our mobile devices have improved significantly
- Skilled at designing for relatively small screens and for the different input capabilities
- But what are then the challenges and opportunities for the design of mobile interactions to come?
- What will the next wave of mobile computing be about?
- Shift away from desktop computing is imminent
- Mobile devices are becoming more and more important and widespread

Phase 7 – Digital Ecosystems

- Dominating point of access to the Internet in combination with the growth of cloud computing will soon dominate peoples' use of computational power
- Creation of *digital ecosystems* in which mobile computing plays a central role
- Rich digital ecosystem of interactive devices, systems and services often referred to as ubiquitous or pervasive computing

Übersicht – Mobile Interfaces

- Geschichte des Mobile Computing
- **Interaktionsstile für mobile Geräte**
- Touch Screen Interaction
- Webseiten-Design für Touch-Screens

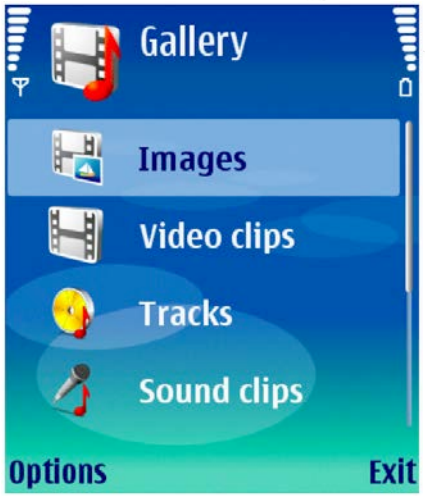
Design für Mobile Interfaces

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

- Physikalität und Einschränkungen
  - Einfache, kleine Interfaces
  - Bandbreite und Konnektivität
- Benutzung von mobilen Interfaces
  - Konstante Verfügbarkeit führt zu erhöhter Benutzung
  - Benutzung parallel zu anderen Tätigkeiten
- Einstellung des Benutzers (Studie von Foolproof)
  - Viele Benutzer haben eine emotionale Beziehung zu Smartphone
  - Viele Benutzer fühlen sich verloren, wenn Smartphone nicht in Reichweite

Interaktionsstile – Indirekte Manipulation

- Kein direktes Mapping zwischen physikalischen Steuerelementen und Funktionalität des Geräts
- Menubasierte Interaktion ist vorherrschender Interaktionsstil in klassischen Mobiltelefonen und Smartphones
- Verbindung von Menus und physikalischen Steuergeräten



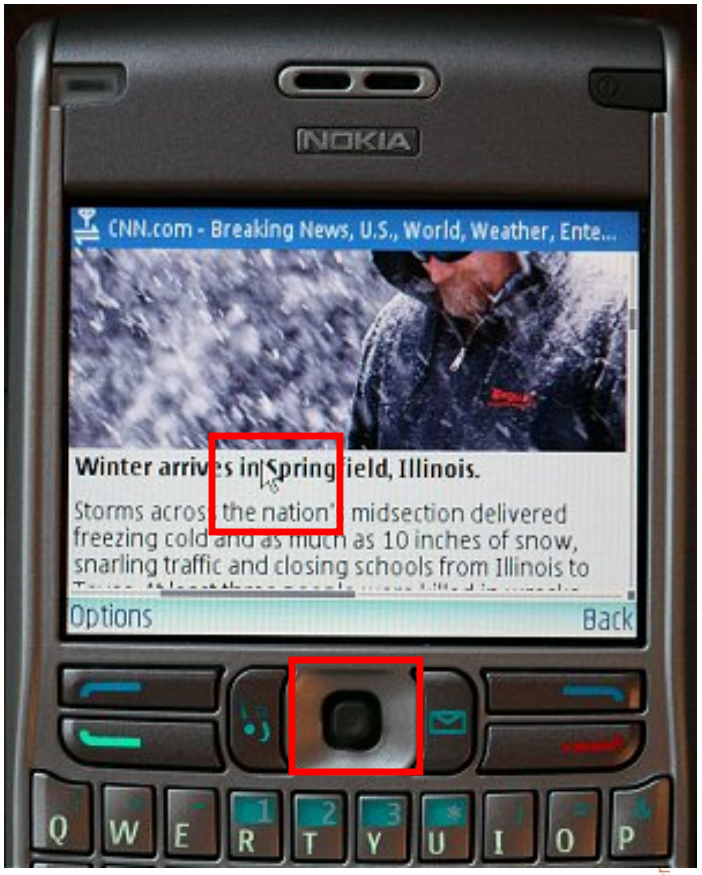
Interaktionsstile – Direkte Manipulation

- Wird für spezielle Funktionen verwendet
  - Wählen
  - Eingabe von Texten
- Sprachsteuerung
  - Wählen
  - Kalendereinträge
  - Erinnerungen
  - Alarme

contextual.interaction.design.r

Interaktionsstile – Direkte Manipulation

- Verwendung von Zeigegeräten und Joystick
- Ähnlich der Desktop-Interaktion
- Heute Eingabe mit Touch
- Verwendet um größere Datenmengen (z.B. in Web-Browsern) zu navigieren

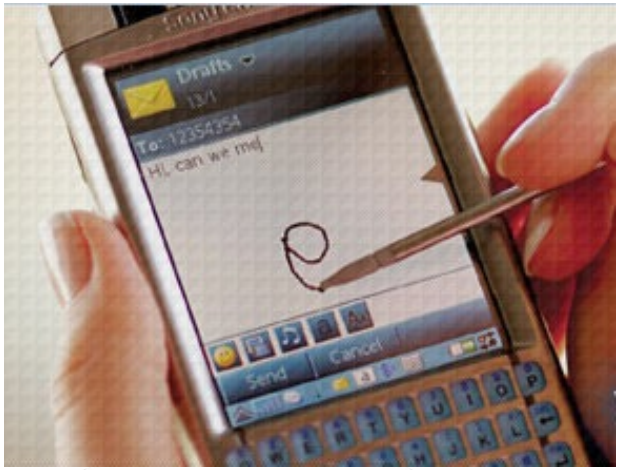


Interaktionsstile - Navigation & Selektion

- Klassisch: Navigationstasten
  - Separate Tasten, z.B. oben/unten/links/rechts
  - Integrierte Wippschalter
  - Digitaler Joystick
  - Rocker, Roller, Wheels
  - Selektionstasten
    - Hard-keys
    - Soft-keys

Interaktionsstile – Stifteingabe

- Unterschiedliche Technologien
- Touch-Screen und Stift
- Grafiktablets
- Smart Pens, Digital Pens
- Erkennung von Handschrift
- Verwendung von "Graffiti"



Interaktionsstile – Stifteingabe

Grafiktablet

Smart Pen

Interaktionsstile – Gesten

- Beschleunigungssensor detektiert Bewegungen des Telefons
- Beispiel Samsung SCH-S310: Benutzer kann Zahlen oder Buchstaben in der Luft zeichnen
- Beispiel KnockOn von LG: https://www.youtube.com/watch?v=y0ZxHRqHSGY
- Beispiel Bump App: https://www.youtube.com/watch?v=7tZPaQVBTRI

Interaktionsstile – Sensoren

- Accelerometer: Automatisches Rotieren zwischen horizontaler und vertikaler Sicht
- Abstandssensor: Display ausschalten, wenn Benutzer Telefon am Ohr hat
- Belichtung: Automatisches Anpassen der Bildschirmhelligkeit an Umgebungslicht

Übersicht – Mobile Interfaces

- Geschichte des Mobile Computing
- Interaktionsstile für mobile Geräte
- **Touch Screen Interaction**
- Webseiten-Design für Touch-Screens

Touch Interfaces

Stift, Gesten, Touch-Screen

- Flüssigere und natürlichere Interaktion
  - Schreiben, Malen, Auswählen, Bewegen
  - Fähigkeiten, die Menschen schon im Kindesalter erlernen
- Multitouch-Gesten
  - Natürlicher als Interaktion mit Maus und Tastatur
- Touch-Interfaces haben andere Benutzbarkeitsprobleme
  - Armbewegungen über weitere Strecken
  - Hand verdeckt Bildschirm

Touch-Gesten

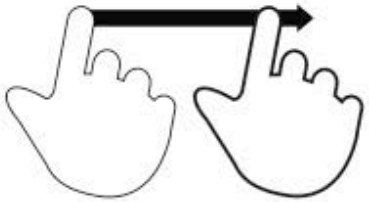
Tap

Briefly touch surface with fingertip

Double tap

Rapidly touch surface twice with fingertip

Drag

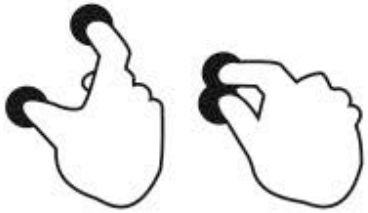


Move fingertip over surface without losing contact

Flick

Quickly brush surface with fingertip

Pinch



Touch surface with two fingers and bring them closer together

Spread

Touch surface with two fingers and move them apart

Press

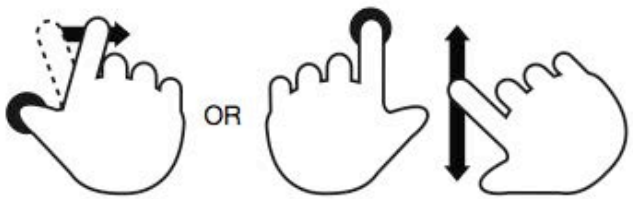
Touch surface for extended period of time

Press and tap

Press surface with one finger and briefly touch surface with second finger

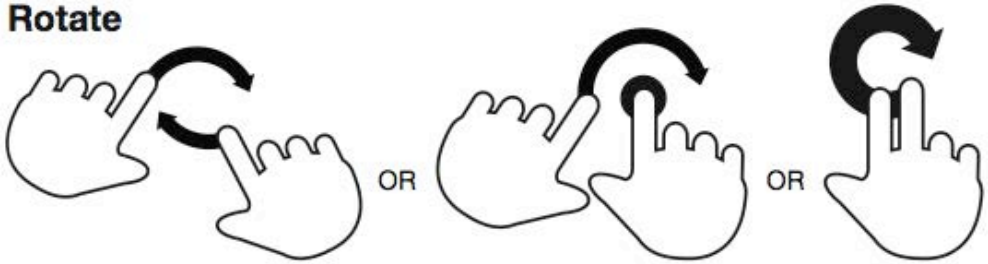
Touch-Gesten

Press and drag



Press surface with one finger and move second finger over surface without losing contact

Rotate



Touch surface with two fingers and move them in a clockwise or counterclockwise direction

Touch-Gesten und Aktionen

- Tap: Programm öffnen, Auswahl
- Double Tap: Auf Objekt zoomen, Sonderaktion ausführen
- Drag: Objekt bewegen, scrollen
- Flick: Schnelles scrollen
- Longpress: Kontext-Menu öffnen, Modus ändern
- Spread: Zoom in, Objekt vergrößern
- Pinch: Zoom out, Objekt verkleinern
- Rotate: Rotieren von Bildern

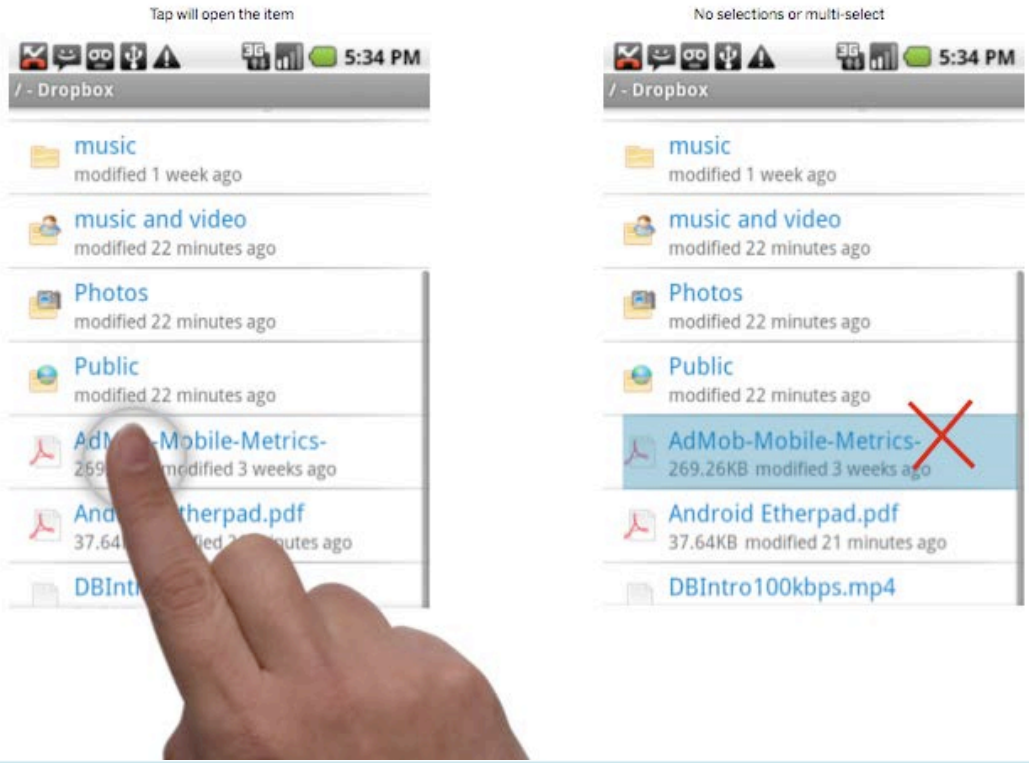
Mouse vs. Touch

	Mouse	Fingers
Precision	High	Low ("fat-finger problem")
Number of points specified	1	usually 1 2–3 with multi-touch
Number of controls	3: left/right button, scroll wheel	1
Homing time?	Yes	No
Signal states	Hover, mouse-down, mouse-up	Finger-down, finger-up
Accelerated movements	Yes	No
Suitable for use with huge screens (30-inch or more)	Yes, because of acceleration	No: arm fatigue
Visible pointer/cursor	Yes	No
Obscures view of screen	No, thus allowing for continuous visual feedback	Yes
Suitable for mobile	No	Yes: nothing extra to carry around
Ease of learning	Fairly easy	Virtually no learning time
Direct engagement with screen and "fun" to use	No: an indirect pointing device	Yes
Accessibility support	Yes	No

https://www.nngroup.com/articles/mouse-vs-fingers-input-device/

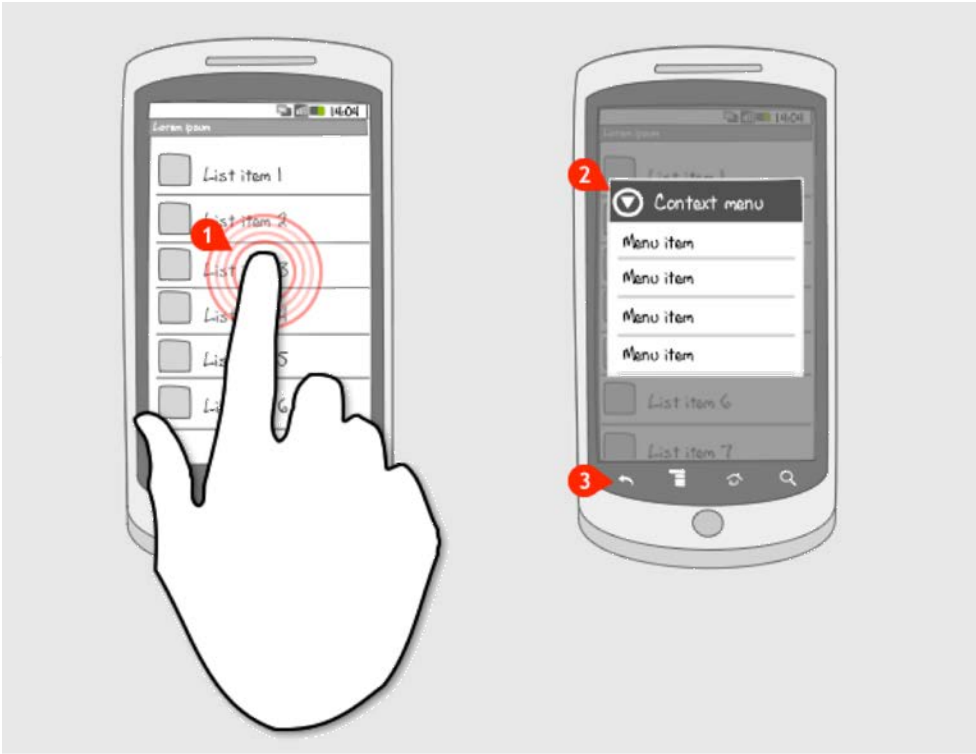
Zeiger

- Auf Touch-Screens gibt es keinen Zeiger
- Man kann Objekte nicht anwählen



Zeiger

- Lösung: Checkboxes oder gedrückt halten verwenden



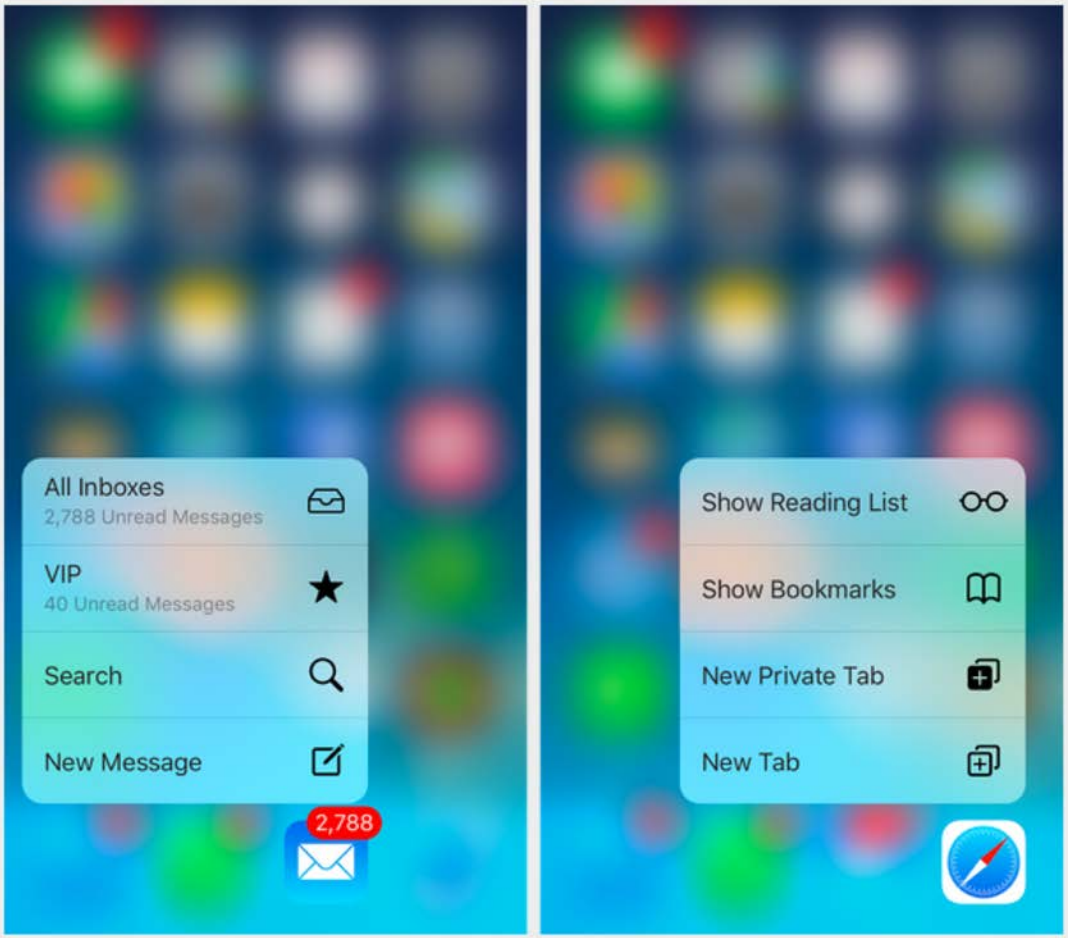
Direkte Manipulation

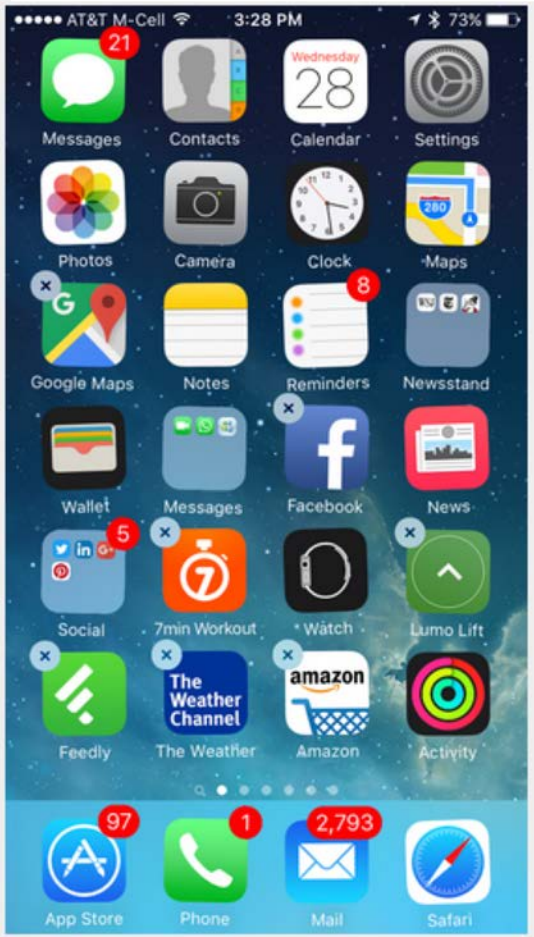
- Auf Touch-Screens kann Benutzer Objekte direkt manipulieren

Direkte Manipulation

- Vorteil:
  - Direkter Zugang zu Funktionalität
  - Natürlichere Interaktion
- Nachteil
  - Benutzer muss sich Gesten / Interaktion merken, keine Affordances wie z.B. bei Knöpfen
  - Besonders anfällig für Verzögerungen im System

Rich/Force/3D Touch / Long Press





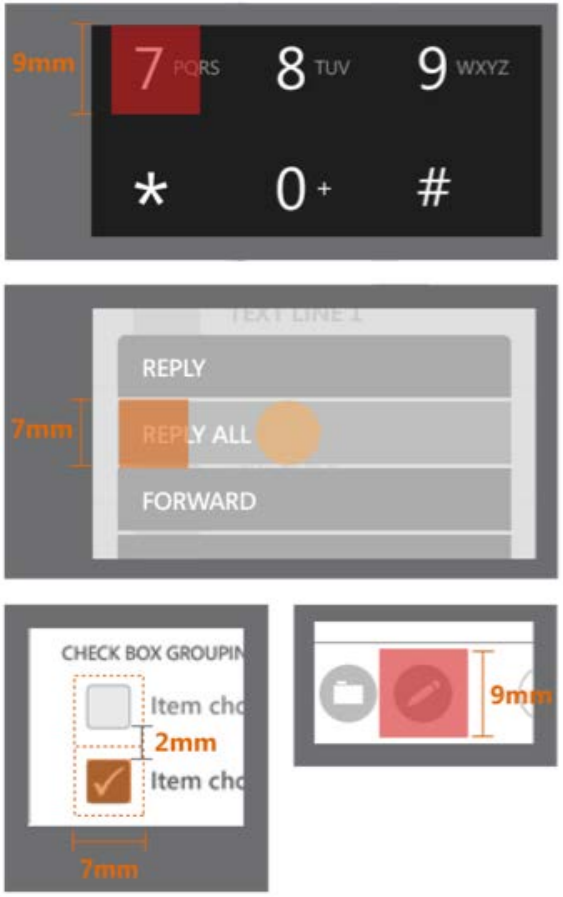
Unlike Force Touch, the 3D Touch depends on where the screen is touched. On the iPhone 6S's homescreen, the 3D Touch applied to the Mail app triggers the list of quick actions associated with this app (left). In the right screenshot you can see the quick actions for Safari.

https://www.nngroup.com/articles/3d-touch/

contextual.interaction.design.research

Empfohlene Kachelgrößen

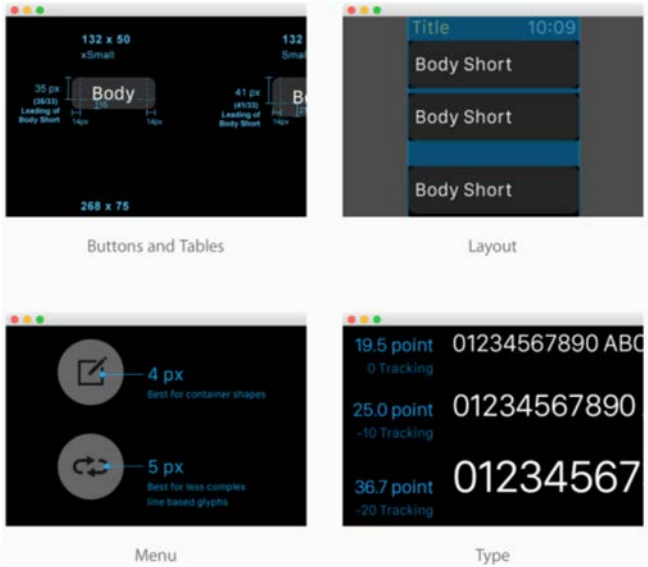
- Größe der Objekte, die anklickbar sein sollen
- Größe hängt von Gerät ab
- Empfehlungen
  - Apple: 44 x 44 Pixel / Points (abhängig von Pixeldichte)
  - Microsoft: Empfohlen 9mm/34px, Minimum 7mm/26px, Abstand zwischen Elementen 2mm/8px
  - [Colle2004]: Elemente für größere Touch-Screens 20 x 20mm

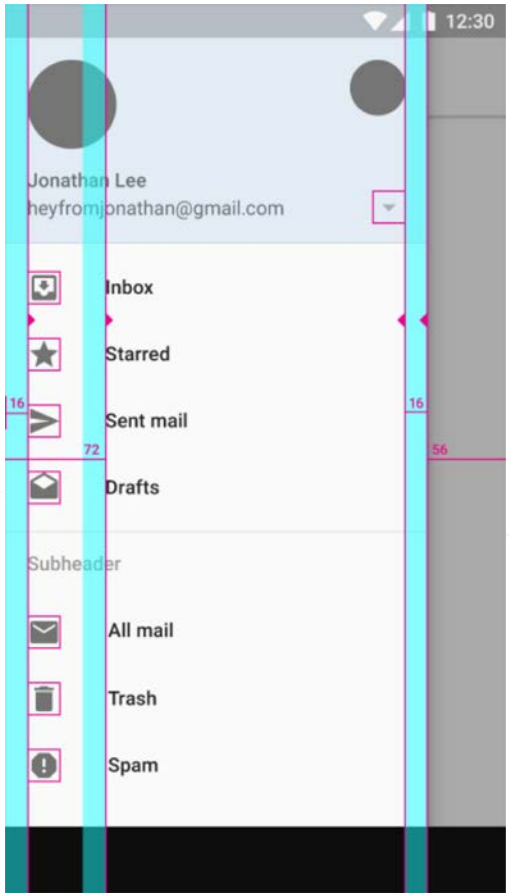


http://www.lukew.com/ff/entry.asp?1085

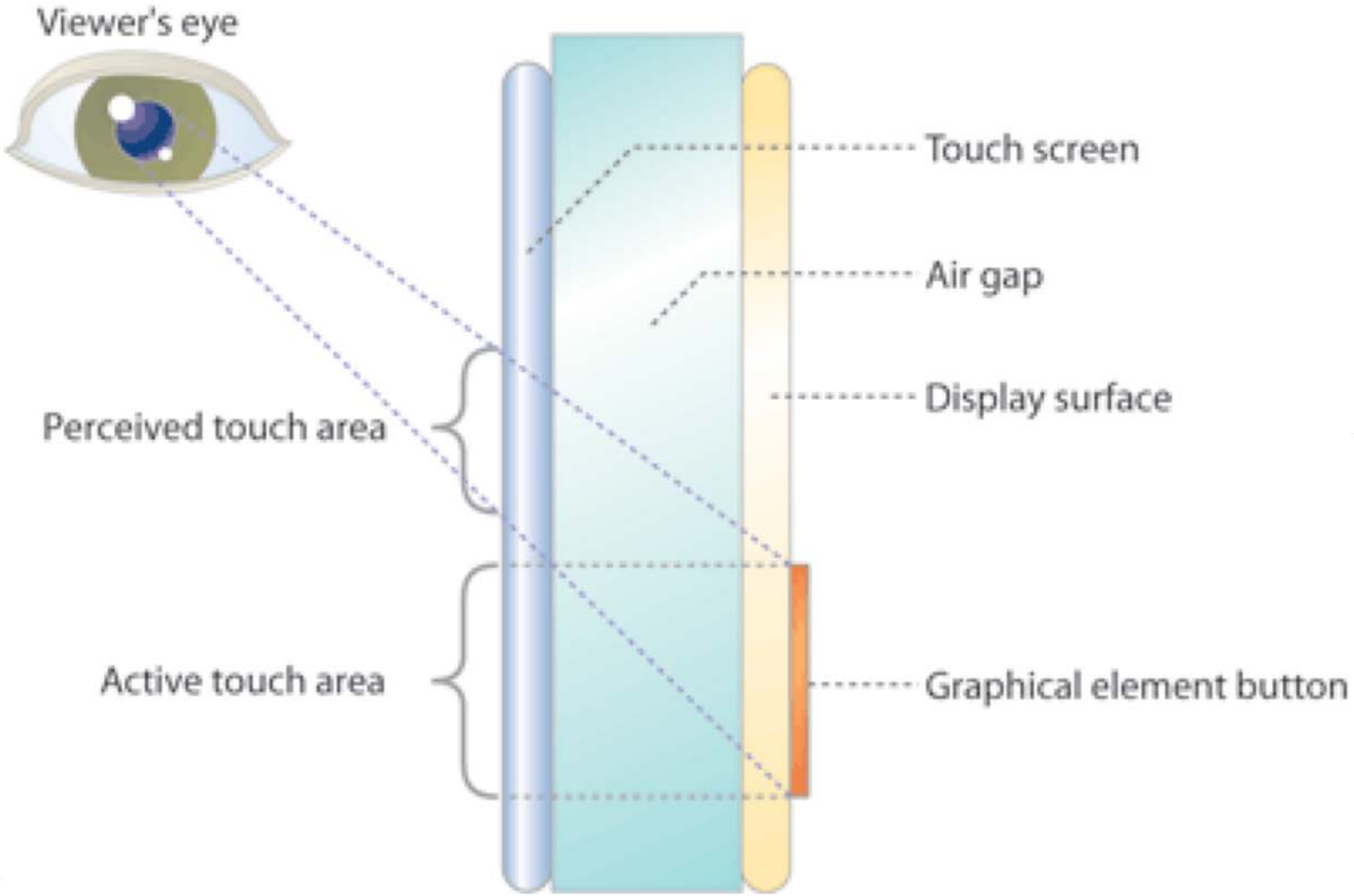
contextual.ir

Hersteller-Richtlinien



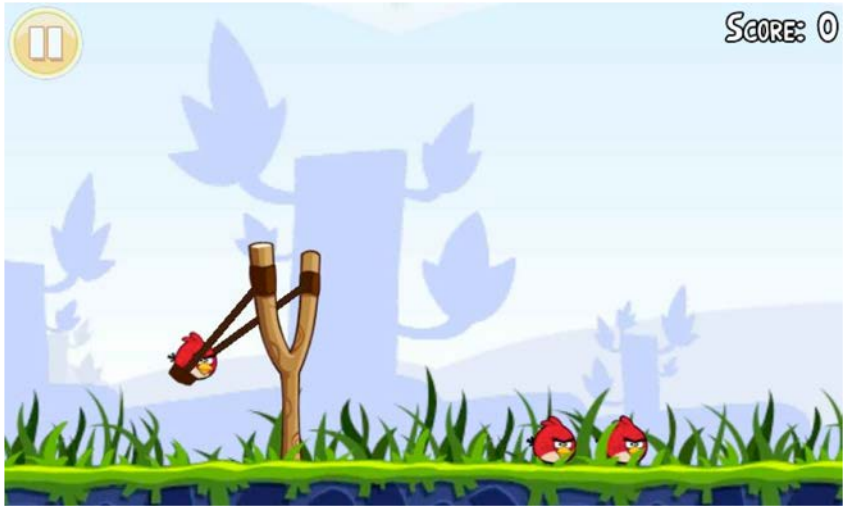


Parallax


Viewer's eye
Touch screen
Air gap
Display surface
Perceived touch area
Active touch area
Graphical element button

Parallax

Target Accuracy



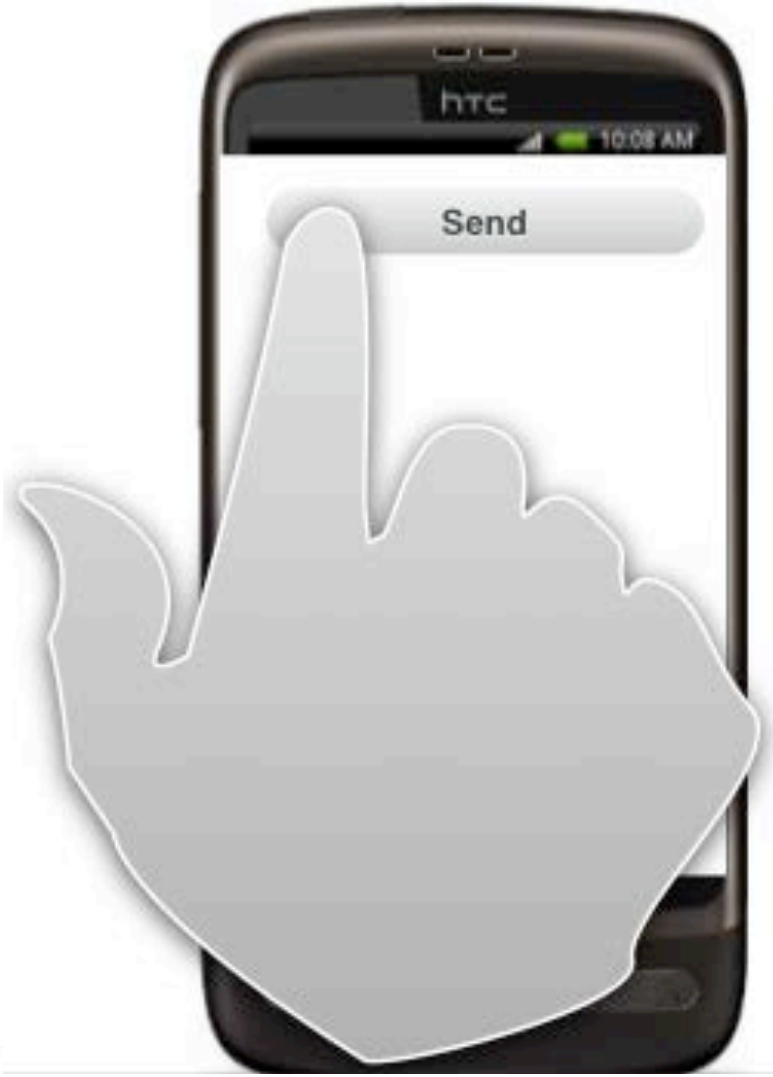
Finger-Reichweite

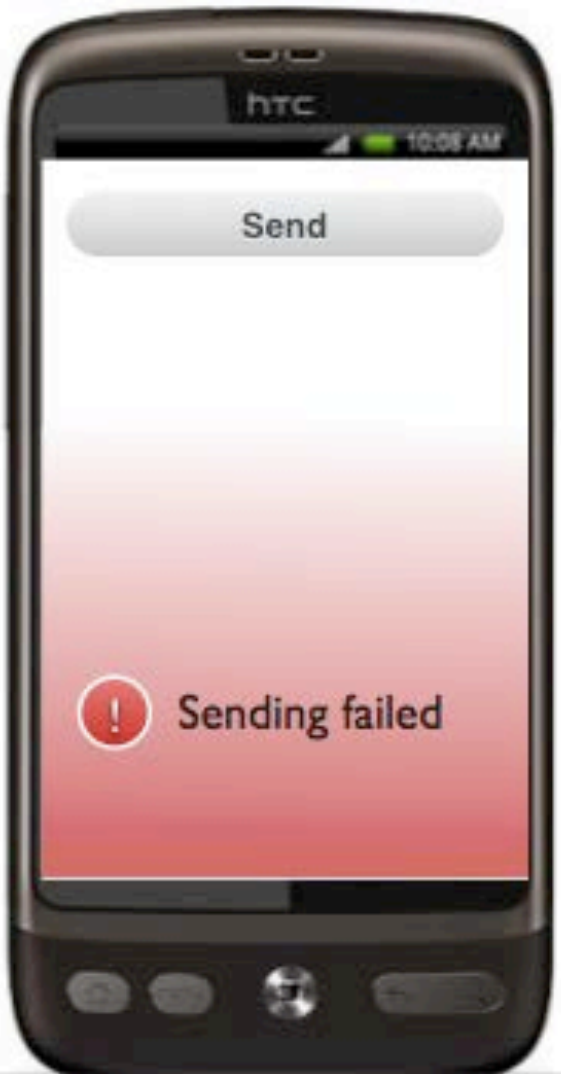
Mobile (right hand)

Mobile (left hand)

Mobile (two hands)

Verdeckung (Fat Finger Problem)





Feedback

- Haptisches Feedback
  - Vibration
  - Piezo-eletktrisches Feedback
  - https://www.youtube.com/watch?v=FiCqlYKRlAA
- Feedback durch andere Kanäle
  - Visuell
  - Akustisch

Affordances

- Klar herausstellen, welche Elemente berürbar sind und zu einer Aktion führen
- Versteckte / zusätzliche Funktionalität muss klar kommuniziert werden
- Veränderungen müssen dort passieren wo die Berührung stattgefunden hat, speziell auf großen Touch-Screens

Touch-Screens Zusammenfassung

- Nicht klassische Desktop GUIs replizieren
- Wenn möglich, direkte Manipulation verwenden
- Konventionen der Hersteller / Betriebssysteme beachten

- Herausforderungen
  - Handschuhe
  - Feuchtigkeit
  - Helles Licht, Blendungen

Übersicht – Mobile Interfaces

- Geschichte des Mobile Computing
- Interaktionsstile für mobile Geräte
- Touch Screen Interaction
- **Webseiten-Design für Touch-Screens**

Design für mobile Interfaces

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

- Mobile Delivery Methods
  - **Mobile**-specific: z.B. Webseite, speziell für Smartphones
  - **Responsive**: z.B. Webseite wird an Größe des Bildschirms angepasst
  - **Native** App: Eigenständige App, alle Inhalte sind schon vorher bekannt
  - **Hybrid** App: Flexible Apps, die z.B. zusätzliche Inhalte aus dem Internet dazu laden.

Design für mobile Interfaces

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

Consideration	Mobile	Responsive	Native	Hybrid	Comments
Tailored to user priorities	★★★	★★	★★★	★★★	A Mobile first approach can improve responsive design's rating to three stars.
Content delivery	★★	★★★	★★	★★★	Responsive sites (and hybrid apps) are more easily indexed in search engines.
Functionality	★★	★★	★★★	★★★	Native apps provide access to device features (e.g. GPS, Camera), allowing more engaging experiences.
Compatibility	★★	★★★	★	★	Responsive design is easily viewed on any screen. Mobile sites and applications are dependant on the device for which they're designed.
Development costs	★★	★★★	★	★★	Dependent on if you are developing a whole site from scratch. Responsive design incurs extra build time, but not as much mobile + conventional design.
Maintenance costs	★★	★★★	★	★★	Individual native apps require individual maintenance. However, responsive design results in a single site that needs to be maintained.

Information Architecture Patterns

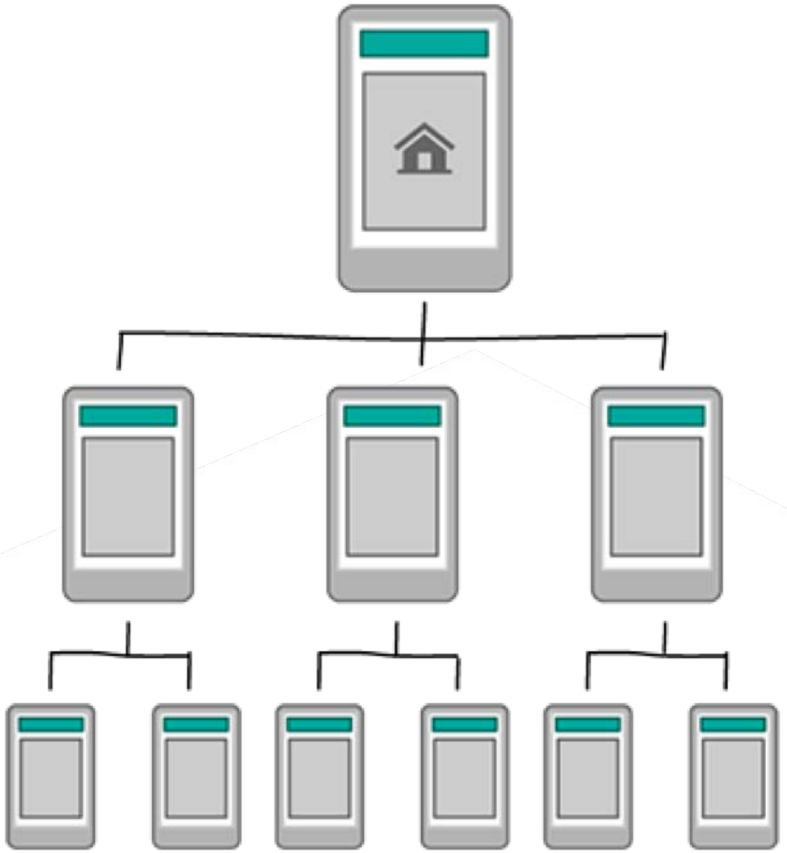
http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

- Hierarchy
- Hub & spoke
- Nested doll
- Tabbed view
- Bento box / Dashboard
- Filtered view

Hierarchy

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

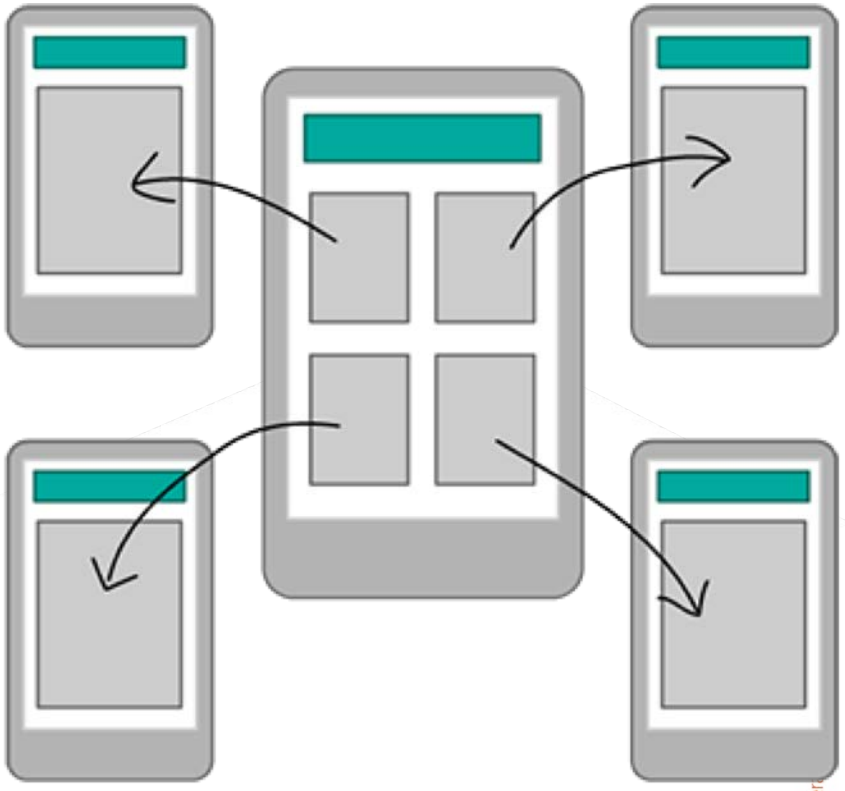
- Index-Seite mit Unterseiten
- Gut für umfangreiche Seiten
- Kann umständlich bei Navigation sein



Hub & Spoke

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

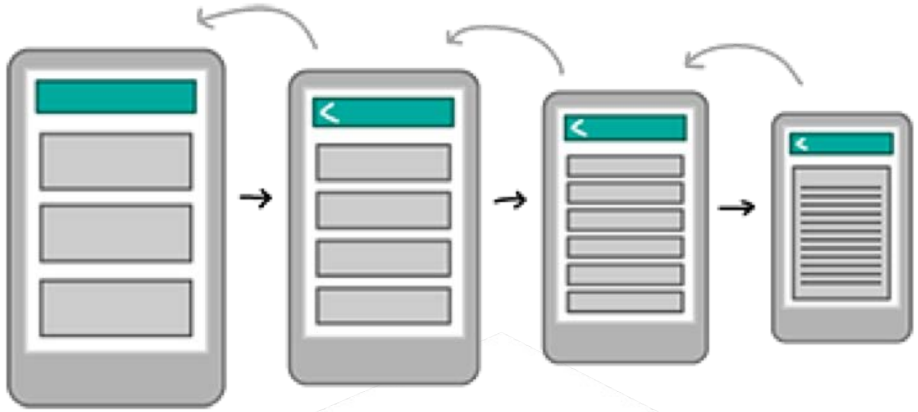
- Zentrale Index-Seite aus der Benutzer heraus- und hereinnavigieren
- Standard Pattern bei iPhone
- Gut für Apps, die mehrere klar abgetrennte Funktionen anbieten
- Ungünstig für Multi-Tasking



Nested Doll

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

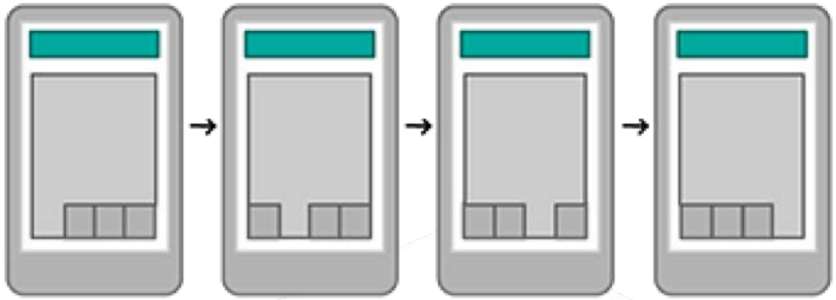
- Führt Benutzer linear zu detaillierteren Informationen
- Gut für Inhalt zu einem Thema
- Kann auch als Pattern für Untersektionen verwendet werden
- Schlecht: Benutzer kann nicht schnell zwischen Sektionen springen



Tabbed View

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

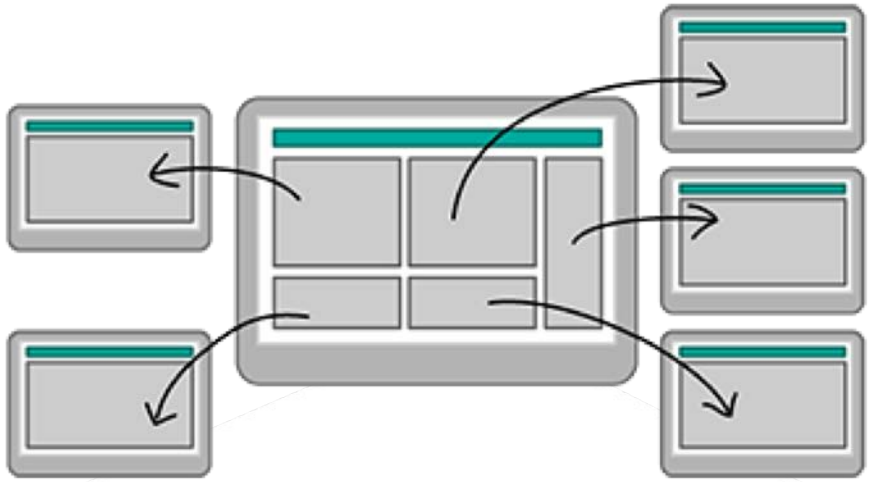
- Sammlung von Untersektionen, die durch eine Toolbar zusammengehalten werden
- Gut für Benutzer um schnell alle Funktionen einer App kennenzulernen
- Gut für Multi-Tasking
- Schlecht für komplexe Seiten



Bento Box / Dashboard

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

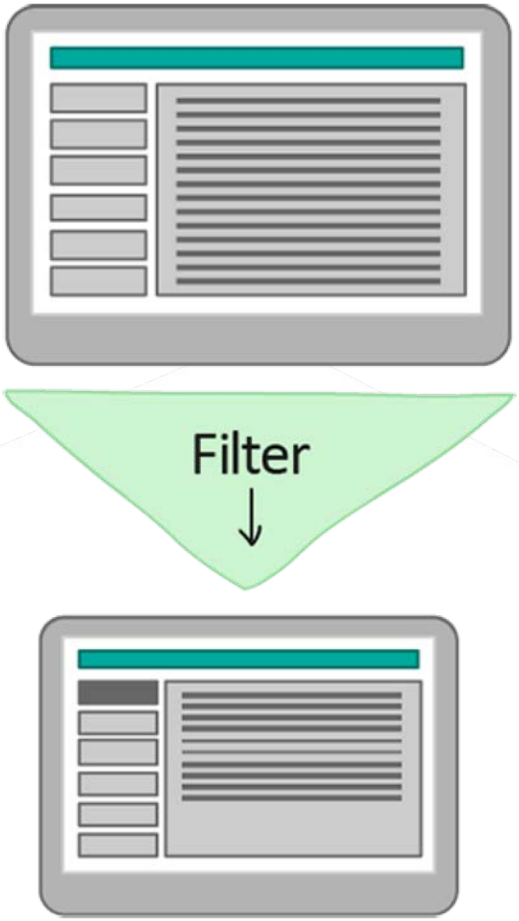
- Stellt detaillierte Inhalte durch Repräsentationen auf Hauptseite dar
- Gut geeignet für Tablets
- Benutzer bekommt wichtigste Informationen auf einen Blick
- Gut für Apps mit vielen Inhalten
- Schlecht: Vorsicht mit zu viel Inhalt!



Filtered View

http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

- Erlaubt Benutzer durch Filter alternative Ansichten von Inhalten zu erstellen
- Benutzer kann Inhalt auf seine Bedürfnisse einstellen
- Gut für Seiten mit vielen Inhalten (Artikel, Bilder, Videos)
- Schlecht: Bei kleinen Screens schlecht darstellbar



Ausblick nächster Termin

In der nächsten Einheit machen wir einen gemeinsamen Einstieg in App Inventor.

→ Bitte vorab schon einmal die vier Schritte unter http://appinventor.mit.edu/explore/get-started.html ansehen

→ Bitte in der nächsten Einheit einen Computer und (falls vorhanden) ein Android Device mitbringen

Weiterführende Links

- The evolution of cell phones
  https://storify.com/yonathan32/the-evolution-of-cell-phones
- Mobile Computing in the Encyclopedia of Human-Computer Interaction
  https://www.interaction-design.org/encyclopedia/mobile_computing.html
- Offizielle Interface Guidelines von Herstellern wie Apple und Blackberry
  http://www.mobilexweb.com/blog/ui-guidelines-mobile-tablet-design
- Designing for mobile
  http://www.uxbooth.com/articles/designing-for-mobile-part-1-information-architecture/

Referenzen

[Colle2004] Colle, Herbert A., and Keith J. Hiszem. "Standing at a kiosk: Effects of key size and spacing on touch screen numeric keypad performance and user preference." *Ergonomics* 47.13 (2004): 1406-1423.

[Kay1972] Alan C. Kay, A Personal Computer for Children of All Ages. In Proceedings of the ACM National Conference, Boston Aug. 1972.

[Kjeldskov2013] Jesper Kjeldskov, Mobile Computing. In: Soegaard, Mads and Dam, Rikke Friis (eds.). "The Encyclopedia of Human-Computer Interaction, 2nd Ed.„, Aarhus, Denmark, 2013, The Interaction Design Foundation. Available online at https://www.interaction-design.org/encyclopedia/mobile_computing.html

App Inventor Links

- App Inventor
  http://ai2.appinventor.mit.edu/
  Anmeldung mit Google Account
- App Inventor Website
  http://appinventor.mit.edu/explore/
- Einfacher Einstieg (Verbindung zu Android Device, erste App)
  http://appinventor.mit.edu/explore/get-started.html
- Anleitungen
  http://www.appinventor.org/course-in-a-box-intro

Setting Up App Inventor

http://appinventor.mit.edu/explore/ai2/setup

System Requirements:

- **Computer** (Mac OS X 10.5 or higher, Windows XP, Windows Vista, Windows 7 or higher, Ubuntu 8 or higher, Debian 5 or higher)
- **Browser** (Mozilla Firefox 3.6 or higher, Apple Safari 5.0 or higher, Google Chrome 4.0 or higher, Microsoft Internet Explorer is not supported)
- **Android Phone** (Operating System 2.3 ("Gingerbread") or higher)
- **Wifi**

Build your project on your computer

Test it in real-time on your device

Setting Up App Inventor

Option Two:

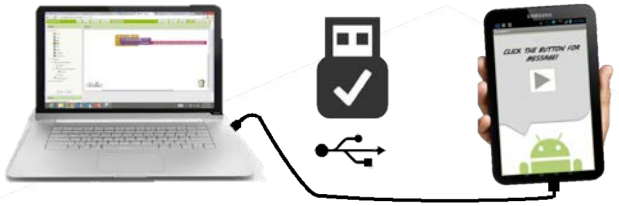
- No Android device?
- Use the Emulator

Build your project on your computer

Test it in real-time on your computer with the onscreen emulator

Option Three:

- No WiFi?
- Use USB Cable



Build your project on your computer

Test it in real-time on your device

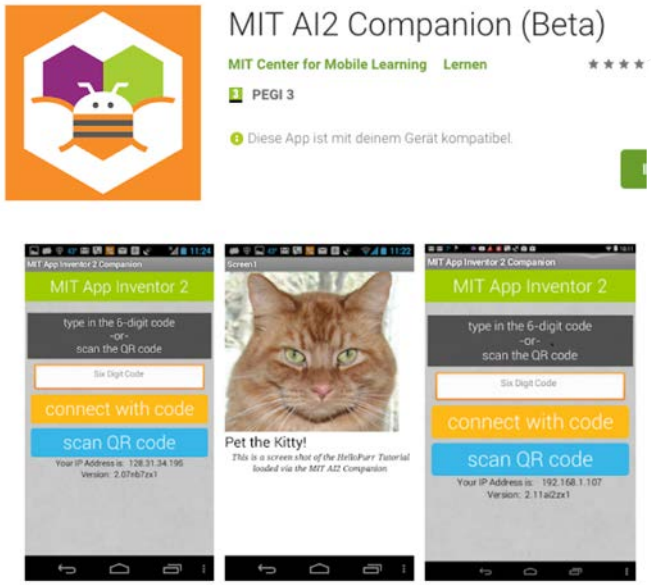
http://appinventor.mit.edu/explore/ai2/setup

Option 1

- No download to your computer
- Develop apps on website: ai2.appinventor.mit.edu
- Live testing on Android device install the MIT App Inventor Companion app

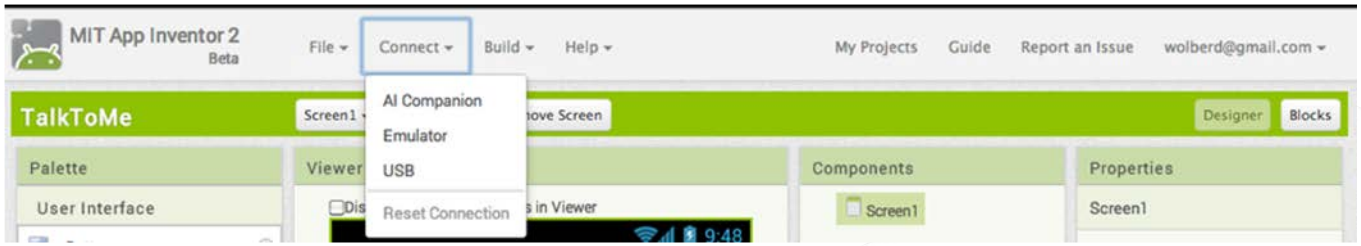
Install

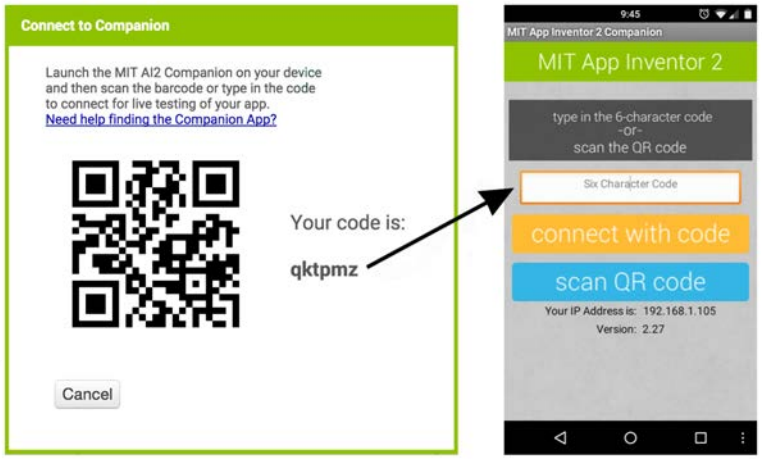
- Step 1:
  Download and install the „MIT AI2 Companion" App on your Android phone
- Goolge Play Store:
- https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3



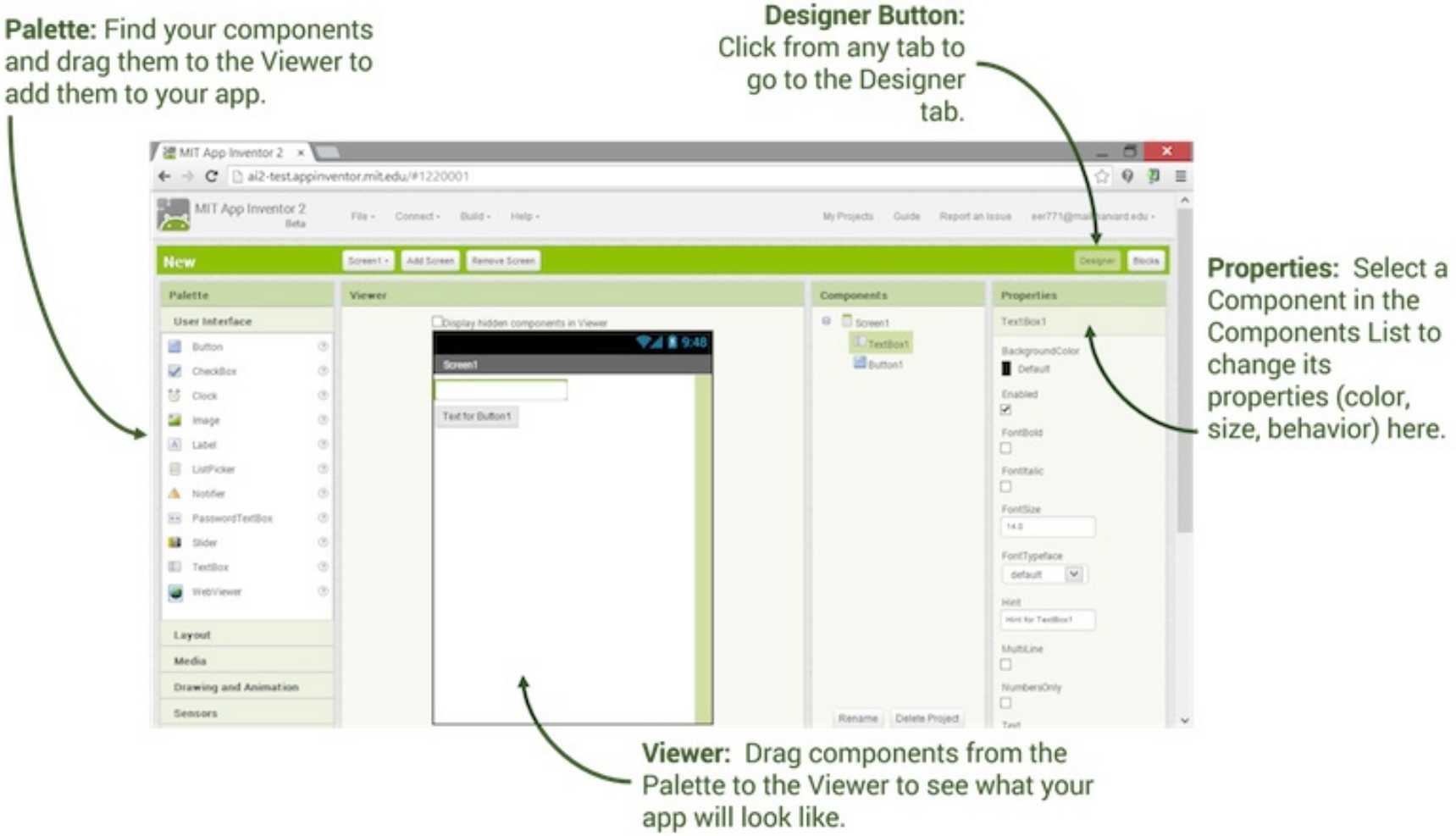
Install

- Step 2:
  Connect both your computer and your device to the SAME WiFi Network
- Step 3:
  Open an App Inventor project (ai2.appinventor.mit.edu) and connect it to your device





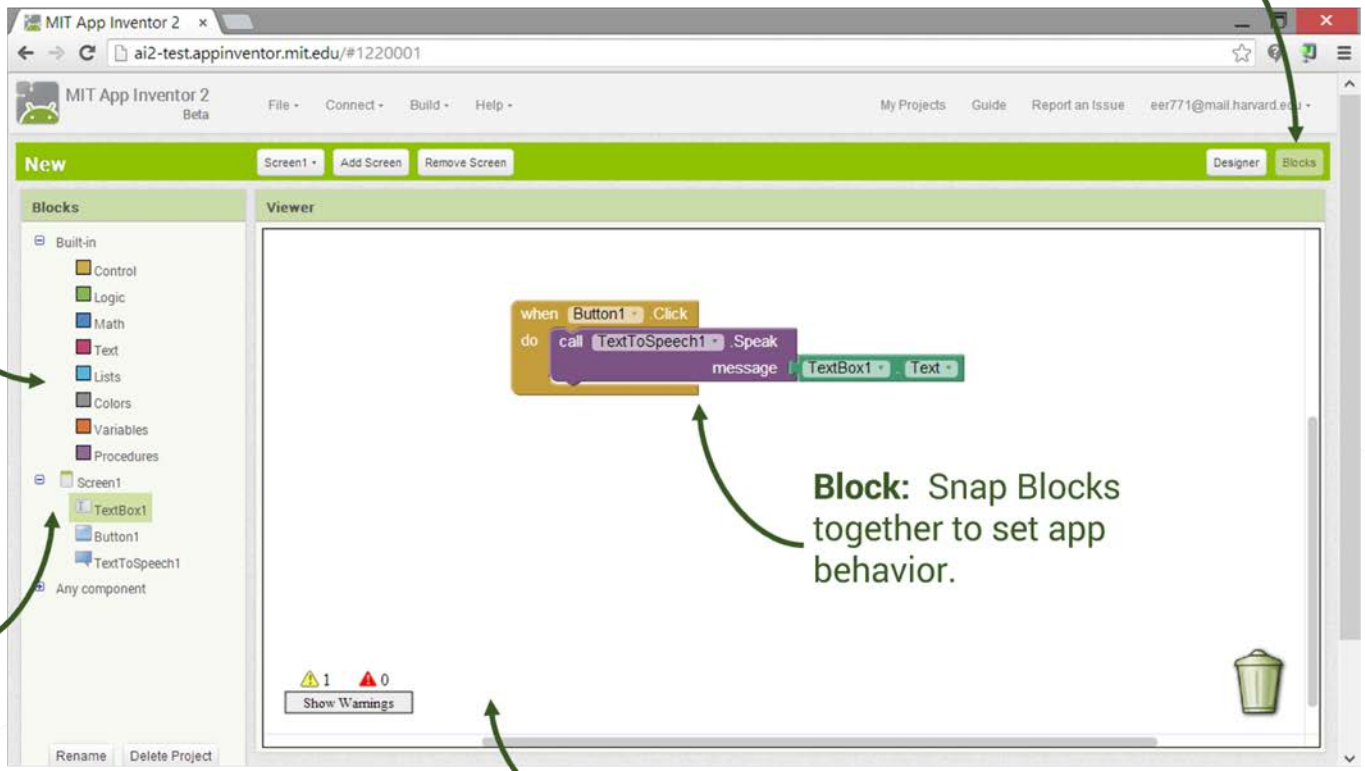
App Inventor Designer



App Inventor Blocks Editor

Built-In Drawers: Find Blocks for general behaviors you may want to add to your app and drag them to the Blocks Viewer.

Blocks Button: Click from any tab to go to the Blocks tab.



Component-Specific Drawers: Find Blocks for behaviors for specific Components and drag them to the Blocks Viewer.

Block: Snap Blocks together to set app behavior.

Viewer: Drag Blocks from the Drawers to the Blocks Viewer to build relationships and behavior.

1st App: Talk To Speak App

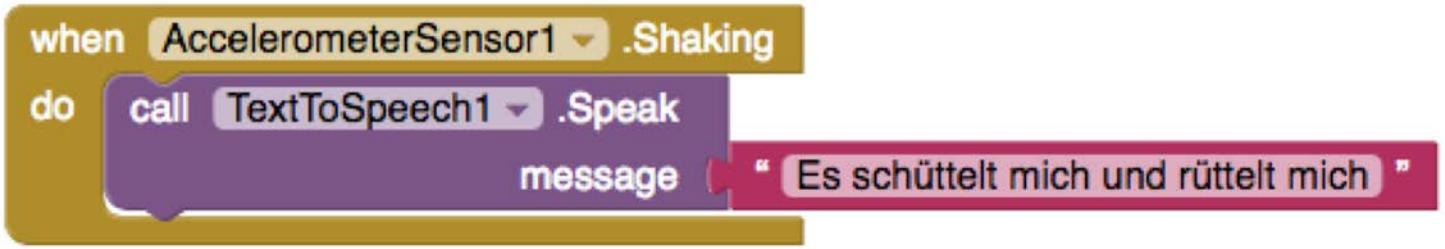
Goal:
When button is pushed phone should speak „Hello World"

1. Create Button
   - Rename
   - Appearance
   - Allignment
2. Create Text2Speech Component
3. Build Blocks

2nd App: Talk To Speak App 2

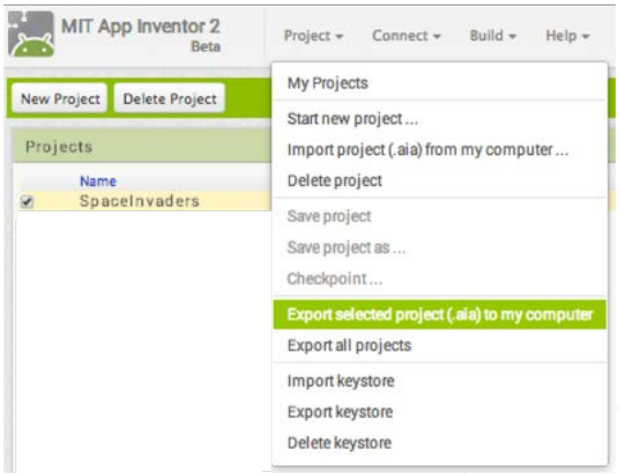
Goal:
When smartphone is shaked phone should speak
„Es schüttelt mich und rüttelt mich"

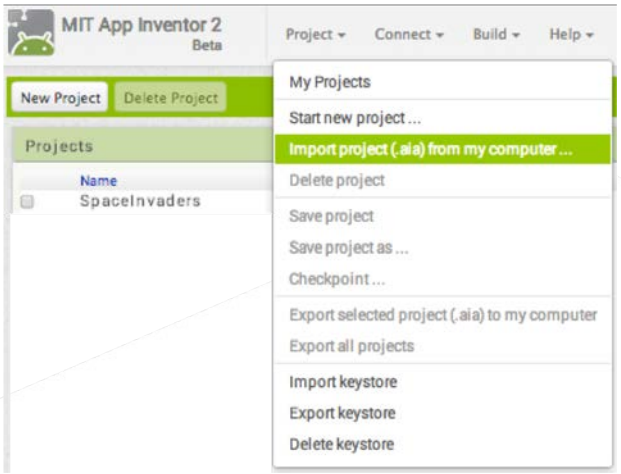
1. Create Accelometer Event Handler
2. Build Blocks

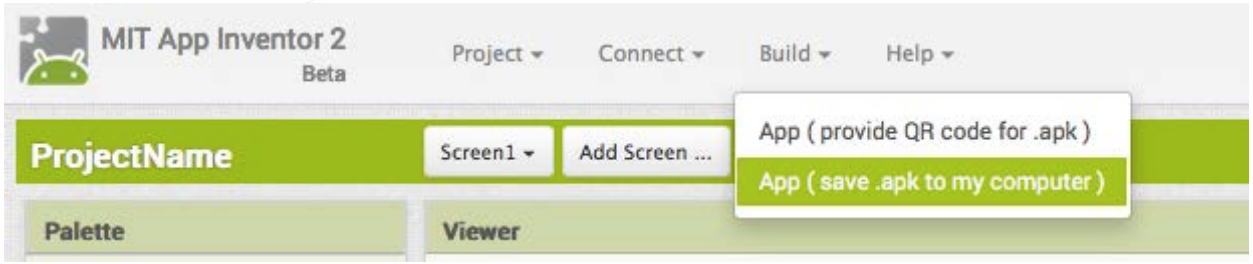


Sharing and Packaging Apps

- Share your app in
  - *source* code form (.aia) - can be loaded into App Inventor
  - *executable* form (.apk) - can be installed on a device







Ball Bounce App

Goal:

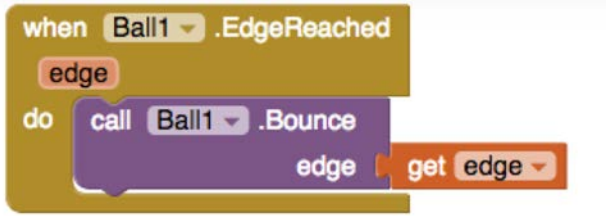
Stelle einen Ball dar, der wenn eine Flick-Geste gemacht wurde in Flicker Richtung mit entsprechender Geschwindigkeit weg fliegt

Canvas: Bereich in dem Grafik Elemente angezeigt werden (Container)

Sprite: Grafik Element (Ball)

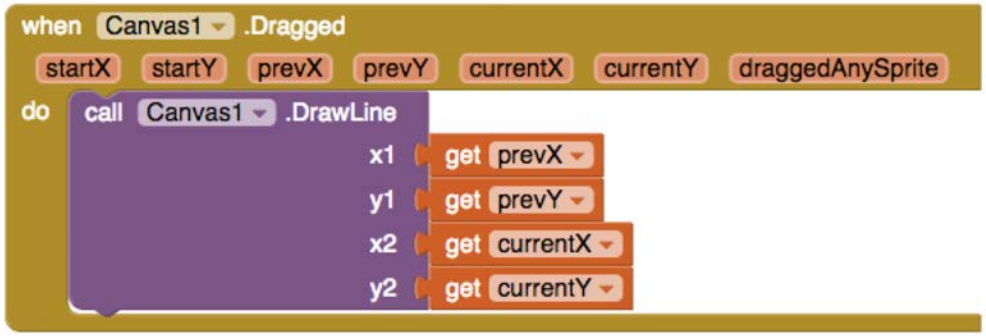
Event Handler: Daten die aus System kommen

1. Screen property: uncheck scrollable
2. Drawing & Animation → Canvas & Ball
3. Block

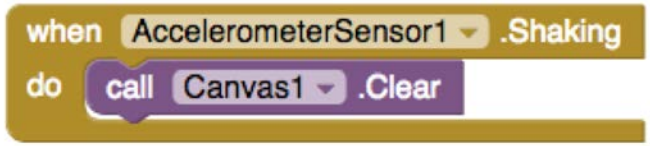


Draw App

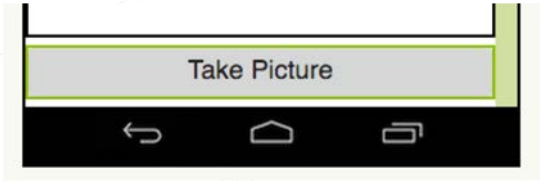
Zeichne Linie mit Finger:

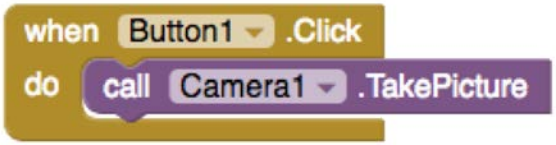


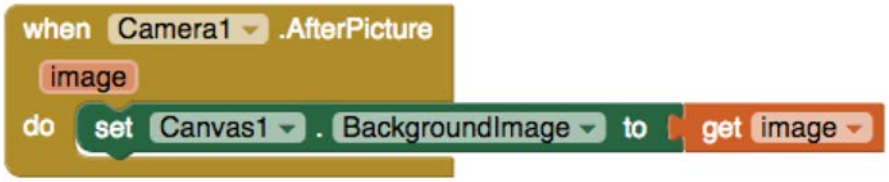
Lösche Linie durch Schütteln:



Hintergrundbild von Kamera:







Noch Fragen...