# Lecture 1: Intro & Context

## Logistics

Schedule

* Lectures – pre-recorded videos, nothing live
* Labs –
* Revision week!

## Intro

Human-Computer Interaction

* Mobile
  + Ubiquitous, pervasive, wearable
  + Def: handheld/wearable/in-vehicle/XR/smart clothes/textiles
  + ACM Mobile HCI started in Glasgow in 1998
    - Form factors and capabilities have changed, but many of the basic interaction challenges have not
* Ubiquitous Computing
  + ACM UbiComp started in Seattle in 1998
* Wearable Computing
  + MIT in 1995
* Contemporary Mobile
  + Type of devices:
    - Handheld: phones, tablets, etc
    - Wearable: watches, rings, glasses, headsets, shoes, clothes, etc
    - Embedded in environment: beacons, IoT devices, sensors, etc
    - Inside body: implants, ‘smart tattoos’, etc

## Context in General

* Challenging Environments
  + Used in many contexts
    - ex: walking, running, driving, in public transport
  + Interaction subject to many challenges and disturbances
    - ex: Connectivity problems, noise, movement, inconsistent lighting, etc
    - Users must **divide their attention** with their surroundings and other tasks
      * => “**fragmented**” interaction in “**micro-bursts**”
  + Both **I/O** are affected by environment
    - ex: difficult to type when walking, hear on subway
* Social acceptability
  + ex: Talking on train, listening to music out loud, not paying attention to others and bumping into

New Interaction Opportunities

* Cameras
* Sensors
  + Location, motion, orientation, etc
  + Project Soli (Jacquard?)
    - gestures near places on clothes etc
    - Pokémon GO
    - Google Maps
* Wireless communication

Impact on Society

* Benefits:
  + Enrich lives
* Drawbacks:
  + Difficult to disconnect – impact on mental health
  + Continuous distraction – impact on social interaction
  + Reliance on tech
  + Continuous video & audio sensing – privacy and security implications

## Context

* The situation in which a device is used
  + Highly variable for mobile devices: **location**, **environment** (light, sounds, etc), **device motion** (walking), **connectivity**, **situational impairments** (holding other items, focusing on surroundings)
* In Social Interactions
  + Humans – good at context
    - Perceive and process contextual information to determine how to act, even if we aren’t consciously aware of it
    - We have a shared understanding of situation
  + Closed-loop social interaction system
    - Human perceives **sensory information**
    - Human combines sensory input with **prior knowledge**
    - Human **infers the current context** and identifies **appropriate behaviour**
    - Human makes a decision about **how to (inter)act** in that setting
* History
  + Two broad topics of research:
    - Detecting and embedding context in info systems
      * Detection of context
      * Relevance to content
      * Usage in info retrieval
    - Using context to **improve UIs**
      * Inferring context in real-time
      * Relevance to tasks
      * Inference about current situation
      * UI response to context and change in it
* In HCI
  + Mobile devices have many sensors: ambient light, location, orientation, motion, microphones, cameras, etc
    - Can be used to infer context: to adapt UIs and present contextually appropriate interactions
  + ex: BBC Weather
    - Uses device orientation to adapt presentation, to make best use of available space
      * portrait – more descriptive info for fewer days
      * landscape – less descriptive info for more days
    - Uses time and location for relevant forecast
  + ex: Google Maps
    - Uses location to recommend nearby restaurants
    - Uses location history to highlight places you’ve been before
    - Uses time to suggest places that are currently open
  + ex: Spotify
    - Uses storage status to determine compression ratio
    - Uses connection type to determine download behaviour
  + Previous examples: **save user effort**
  + Context is:
    - multi-faceted
      * but **fusing information modalities** gives a better idea of the usage scenario (e.g., location + time + calendar appointments)
    - difficult to infer
      * having too many bad decisions **frustrates users**
    - potentially bad for **privacy**
    - what is happening **now**
  + Context-aware systems should also consider **prior context**
    - Learning from previous situations
    - ex: Google learning commute, giving time estimates
  + Senses and capabilities
    - Interaction doesn’t need to take place via a touchscreen
  + Why?
    - **Reduce interaction time** by taking actions on a user’s behalf
    - Allow users to **focus** on completing their task
    - Expose to **new opportunities**, e.g., new nearby coffee shop
    - Overcome **situational impairments**, e.g., increase screen brightness
    - Create **elegant** but **efficient** UIs, e.g., reduce irrelevant info
    - Improve **situational awareness**, e.g., when walking/cycling/driving
* Types
  + Environment
  + Location
  + Connectivity
  + Activity
  + Device status
  + …

## Context Ambiguity

* **Uncertainty** inherent in context sensing
  + Ambiguous information
* Information **quality** can vary (connectivity, GPS signal)
* **Ambiguity** in information (motion, brightness, transport type)
* UIs tend to hide inner workings
  + Hard for users to evaluate system behaviour
  + Consider how to display **context info** and **system state**
    - **Interaction resources** to help users understand
    - Reveal uncertainty through **feedback**
      * **Knowledge** of ambiguity present
      * Can try different actions and **see** **results**
      * More **forgiving** users
      * ex: Location (range)
        + Compromise

An abstraction

Hides less relevant info

Simplifies info that presents resources to help users

* + - * + System doesn’t know why GPS signal is poor

Places responsibility on user

Ambiguity becomes resource

* + “Seamless” design
    - Take info away from user
    - Alternative: **seamful** design
    - Or compromise: give enough feedback to deal with ambiguity

**Ubiquitous Computing**

* Vision – computation gets embedded in our surroundings and activities, so the computer “disappears”
* **Calm technology**: “A calm technology will move easily from the **periphery of our attention** to the **centre**, and **back**.”
  + Technology does not need to be the focus of attention at all times
  + e.g., IoT devices that learn to automate heating
* Foreground & Background interaction
  + foreground & background of attention
  + Background
    - Reducing user effort to accomplish a task
    - ex: Phone detects when it’s lifted and unlocks the screen
* Relevance to Context
  + Only know when to change focus based on context

# Unit 2: Mobile Interaction

## History

* Motorola DynaTAC 8000x
  + Released in 1983
  + First ‘mobile’ phone
    - 30min of battery life (after 10 hours of charging)
    - Started the ‘brick’ era (named for size and lack of reliability)
* IBM Simon
  + Released in 1994
  + First smartphone?
    - Had ‘apps’ for email, calendar, notes, etc
    - Touchscreen input (with stylus)
* Nokia Communicator 9000
  + Released in 1996
  + First smartphone?
    - First to render web graphics
      * IBM Simon didn’t have a web browser
    - Had 2 screens
      * When closed, it worked like a normal phone
* Nokia 3210
  + Released in 1999
  + First ‘fun’ phone, not a ‘business’ phone
    - Mobile games (Snake)
    - Compose own ringtones
    - Had a precursor to emojis
    - Colourful replaceable plastic covers
* Motorola Razr V3
  + Released in 2004
  + First ‘stylish’ phone
    - Emphasised sleek design and slim profile
    - Started the trend of slimmer, lighter phones
* Apple iPhone
  + Released in 2007
  + First ‘touch’ phone?
    - Not really (LG had touchscreens first)
    - Apple made better use: swiping, scrolling, tapping with multiple fingers
* HTC Dream/GI
  + Released in 2008
  + First Android phone
    - An ‘open’ attempt to take on Apple’s closed platform
    - Form factor mostly unrecognisable now
      * Integrated wheel for 2D pointing
      * Several physical buttons
      * Slide-out keyboard
* Afterwards: boring
  + Technical improvements
    - Many-core processors, dedicated GPUs
    - Co-processors, e.g., for activity tracking
  + Camera improvements
    - Multi-lens cameras, better sensors, dynamic focus
* Recent changes
  + New **sensors**
    - Ex:
      * Soli radar in Google Pixel 4
      * Grip pressure in Pixel 3
      * 3D Touch in iPhone 6
    - New interaction techniques
    - Also new design challenges (discovering, learning, efficiency of use)
  + New **form factor** - folding
    - Ex
      * Samsung Galaxy Fold (or flop? :P)
      * Huawei Mate X
    - New interactions, new challenges (how to design apps)
* In short:
  + Forms factors
    - Bricks to flip touchscreens
  + Processing power
    - 1990s: 8 MHz processors; 2020s: 8 core processors
    - 1990s: first web images; 2020s: 3D rendered graphics in UHD

## Current Constraints

* **Processing Power** vs **Battery Life** vs **Device Size**
  + Currently the most important factor constraining performance
  + Complex relationship
* Screen size
  + Resolution, clarity, colour constantly improving
  + Size limited by need for portability and efficiency
  + Need most use of limited display space
* New **interaction techniques** need more processing power
  + ex: Speech assistants always listening
  + ex: in-air gesture sensors always sensing
  + ex: machine learning for interactions
* New **applications** and **use-cases** require more processing power
  + ex: **VR** requires HD, high-FOV, low-latency graphics
  + ex: **AR** requires object recognition, etc
* Processors need to be small and efficient, but do complex tasks
  + => mobile coprocessors
    - Dedicated processing units that do 1 task efficiently
    - ex: Apple M-Series and Snapdragon Sensor Core
      * for collecting, processing sensor data (accelerometer, gyroscope, etc)
    - ex: dedicated mobile GPUs
      * for high frame-rate but energy efficient graphics rendering
    - ex: Apple Neural Engine
      * optimised for ML operations
* Batteries need to last **long** but be **small**
  + Other components shrink
    - 4000 to 6000 mAh devices increasingly common
  + Platforms are becoming proactive about efficiency
    - ex: limiting background execution, inefficient operations
* Memory (less of an issue now)
  + Garbage collection is much faster than it was
  + Android design patterns used to exist to reduce memory use
    - ex: “view recycling” for scrolling lists in Android
    - Still used for **efficiency**, but not vital
  + Apple implements memory management at compile-time (doesn’t have garbage collection)
* Connectivity (less of an issue now)
  + Big gap between developed and developing nations

## Sensing New Interactions

* Capabilities
  + ex:
    - Accelerometer, gyroscope, magnetometer, temperature, humidity, etc
    - Microphone, cameras, touchscreen
    - NFC, Bluetooth
  + **Virtual sensors**
    - Step counter (accelerometer + gyroscope)
    - Face detectors
  + Benefits
    - New and improved **UXs**
    - Improved **security** and **privacy**
    - New **interaction techniques**
* recent ex:
  + Speech input
  + Mid-air gesture
    - Hand movements/poses in air
    - Why?
      * Quick low-effort interaction (skipping songs – Google Motion Sense)
      * Eyes-free input without touching (while driving)
      * When you can’t touch (cooking/eating)
      * Hover actions (previewing images)
    - Sensors
      * Depth
        + Many phones have it
        + Accurate
        + Not efficient
      * Cameras
        + Ubiquitous
        + Not efficient
        + Only 2D sensing
      * Proximity
        + Infrared ‘1 pixel’ sensors (mostly used to lock when holding to head)
        + Efficient
        + Low resolution
      * Ultrasonic
        + Emit ultrasound and listen for reflections
        + Efficient sensing
        + Inefficient reliance on ML
    - ex: SideSight: multi-’touch’ interaction around small devices
      * Used 1D array of infrared sensors to detect finger position beside phone
      * Users could control a ‘pointer’ on screen without needing to touch it
    - ex: HoverFlow: expanding the design space of around-device interaction
      * Used 2D array of infrared sensors to detect gestures above the screen
      * Users could swipe to skip songs
    - ex: Abracadabra: Wireless, High-Precision, and Unpowered Finger Input for Very Small Mobile Devices
      * Used watch **magnetometer** to sense movements from a magnetic ring worn on the finger
    - ex: Motion Sense in Pixel 4
      * Project Soli
      * radar
    - Challenges of user knowledge:
      * what the gestures are
      * where
      * correctly
      * how to solve problems
      * how to improve?
    - ex: Samsung AirView, Samsung Air Gesture and Google Motion Sense
      * require manual turn on, come with instructions
    - Advice:
      * Give as much **feedback** as possible
        + Visual, auditory, haptic
        + Confirmation to actions – “**system attention**”
        + Visualisations
      * Don’t repeat Xbox Kinect mistakes
  + 3D depth sensor
  + Grasp pressure input and Touchscreen pressure input
    - Why?
      * Adds extra dimension (new applications)
      * Passive grip to active input (ex: launch assistant)
      * One-handed input
      * Eyes-free input
    - ex: Pressure-based menu selection
      * Used force sensitive resistor on edge
      * Apply pressure to move cursor down a list of items
    - ex: Towards utilising one-handed multi-digit pressure input
    - ex: “Active Edge” in Pixel phones
    - ex: Apple 3D Touch in iPhone 6
      * 3 levels of pressure
    - Research shows that users can skilfully modulate pressure
    - Current implementations use far fewer levels of pressure
      * But more precise uses need use cases/evaluation of usability
  + Capabilities expand because of better technology and average user’s expanding knowledge

## Design

* Focus on mobile use
  + Screen size, limited connectivity, limited, resources, etc
  + Used in different places, contexts
  + Mobility affects interaction (e.g., walking)
  + Attention is divided
  + Used in public spaces
* **Streamline** UI and interaction design
  + **Can’t fit everything in 1 screen**
  + Break tasks down into **simpler steps**
  + Reduce clutter to emphasise **content**
    - e.g., sidebar
    - e.g., menus/popups
* **Mobile-focused interactions**
  + Tap, drag, swipe, pinch, expand, rotate, multi-touch
  + Direct manipulation
    - Engages people, facilitates understanding
    - Immediate, visible results of actions (rotation, gestures)
  + Interaction metaphors
    - e.g., flick pages in books
* Make use of context info
  + Create local experiences
    - engaging experiences: Pokémon GO, Tinder, Just Eat
* Attention Spans
  + Short bursts and fragmented interaction
    - Short bursts – attention divided
    - Fragmented – switching attention
  + Implications/recommendations:
    - Simplify UI: easier search
    - Anticipate lack of attention
    - Anticipate slowness
  + Usually in 4-second bursts
* Checking Habits
  + “brief, repetitive inspection of dynamic content quickly accessible on the device”
    - Lots of new content and notifications
    - Possible to convert into richer interactions
  + Smartphones more pervasive

## Lab 1

<https://docs.google.com/document/d/19jIJLuuULlq6GKY26A0QVFQEaAiBmicw_q3iTFChgog/edit>

# Unit 3: Interaction Design

## Interaction Design

Usability

* “extent to which a product can be used by specified users to ahiceve specified goals with **effectiveness, efficiency** and **satisfaction** in a specified context of use”
* “a design is not usable or unusable per se; its **features**, together with the **context** of the user (what the user wants to do with it and the user’s environment) determine its level of usability”
* Not functionality
* Combo of functionality, user interface and context

Interaction Design

* “design of interactive products and services in which a designer’s focus goes beyond the item in development to include **the way users will interact with it**”
* Prioritises **usability** during the design process
  + Good features are important, but nobody will see them if UI is bad
* Iterative process
  + Identify needs of users and/or company, establish requirements
  + Develop initial candidate designs
  + Create **interactive prototypes** of the designs
    - Paper, sketches, layout mock-ups
    - Interactive wireframe layouts, HTML layout prototypes
    - Low-cost and fast (needed to be discarded, improved, replaced)
  + **Evaluate** design using their prototypes
    - Usability and UX
    - Compare and contrast alternatives
    - Identify areas for improvement
  + Repeat!
* Interaction
  + How does the user accomplish tasks?
  + What design patterns will info and task hierarchy use?
  + How to provide input?
  + More important than the aesthetics of UI layout
* Users involved through process
  + User-centred Design and Participatory Design
  + Evaluation – core activity
  + Help refine designs at all stages

‘Design Funnel’

* Icon

  Description automatically generated
* Conceptual process for product design:
  + **Elaboration**: generating new ideas, expanding design space
  + **Reduction**: selecting best ideas, reducing design space
* In practice, funnels overlap
  + A picture containing line chart

    Description automatically generated
* As design process progresses, prototypes become higher fidelity
  + From paper-based to interactive digital

## Prototyping

Useful – [www.uxknowledgebase.com](http://www.uxknowledgebase.com)

App Definition Statement

* A concise, clear statement about the purpose of an app:
  + Identifies **intended audience**
  + Identifies precisely what app will **allow users to do**
* First introduced in iOS interface guidelines around 2008
* Useful point of reference throughout design process
  + List of **core requirements**
  + **Brand** identity
  + Refer back to this: are decisions cohesive with statement?
* ‘**Live’** piece of writing: changes during design
* Writing:

1. Identify users (who, values, motivations)
2. Identify features and requirements (brainstorm!)
3. Start filtering feature list (most important, help accomplish user goals)
4. Create (concrete, clear) (maximum of a few sentences)

Storyboarding

* Interaction design technique based around **user stories**
  + Based around **users** – needs, context, emotions, etc
  + High level: **lifestyle** and **feelings**/**actions**
  + Scenario that illustrates aspects of the **app definition statement**
* Help designers **empathise** with users, identify **interaction needs**

Sketches & Paper Prototyping

* Goal – generate many designs
  + Designs get refined/discarded
* Low-cost prototyping for refinement/discarding
* Benefits:
  + Versatile
  + Quick
  + Users more generous in giving **honest feedback** (because it’s **easy to change**)
  + Reduces documentation
* Uses:
  + Identify missing parts
  + Testing structure
  + Designing main user flows
  + Determining key interactions
  + Think about edge cases, different UI states
* Limitations:
  + Interactions hard to represent (gestures, etc)
  + Can’t simulate animations
* Robust **evaluation** method:
  + **Think-aloud** evaluation: user asked to interact with prototype and speak their thought process aloud
  + **Heuristic** evaluation: design experts evaluate interaction against heuristics (rules of thumb) for good design
* **Creativity** not limited by technology
  + Allows prototyping **complex interactions**
* Getting started:
  + Paper, card, post-it notes
  + Pencils/pens – colours for emphasis, shading, etc
  + opt: Glue, tape, blue tac, staples, adhesives, etc
  + Scissors, ruler
  + Stencils
* Example workflow:

1. **Sketch**
   1. Low-fi for app layouts, focus on info structure
2. **Evaluate** with users
   1. Aim – select best candidates for further evaluation
3. Construct **paper prototypes** for best designs
   1. More detailed
   2. Think about **interaction** as well as layout
4. Evaluate interaction
   1. Aim – understand how users interact (mistakes, clarity, understanding, identify better ways of accomplishing tasks, simplify interaction)
5. **Modify** good designs, **discard** poor designs
   1. Don’t just discard – take notes
6. **Repeat** (gradually shift to digital prototypes)

## Navigation Patterns

Navigation

* Act of **moving between screens** to **complete tasks**
* Supported by:
  + Navigation components
  + Embedding behaviour into content
  + Platform affordances
* Patterns
  + **Lateral**
    - Moving between screens at the **same level** of the hierarchy
    - Types:
      * **Distinct top-level entities**, e.g., News categories
      * **Related lower-level entities**, e.g., News stories within a category
    - UI components: Navigation drawer, Tabs, Bottom bar
      * Navigation drawer
        + ‘Hidden’ drawer on the left, **expands** to reveal destinations in app
        + Typically for access to the **top level** of the hierarchy
      * Tabs
        + Organise content at the **same level** of the hierarchy
        + Content is **related**, part of a **group**/set
        + Each tab should contain content that is **distinct** from other tabs in the set
        + Appear at top of activity, clearly labelled, **self-revealing**
        + Types:

**fixed** – all items shown at same time with fixed size (max – 4)

**scrollable** – no fixed width, can be wider than the screen

* + - * + Horizontal Swiping useful

Efficient navigation

Jumping between screens – pogo-sticking

* + - * Bottom bar
  + **Forward**
    - Moving between screens at **consecutive** levels of the hierarchy, **steps in a flow**, or **across** an app
    - Consecutive levels – go deeper for more detail
    - Steps in a flow – move to the next step in a process (ex: recipe)
    - Across an app – move directly to a deeper part of app (ex: search)
    - UI components: Content cards, Next button, Search
  + **Reverse**
    - Moving **backwards** through screens (chronologically/hierarchically)
    - Hierarchical: moving up the info hierarchy
    - Chronological: moving back through previous views
    - UI components: back button

Collections: Lists and Image Lists (Grids)

* Lists: linear set of logically grouped items from a collection
  + Best suited to text-heavy entities (news stories, emails, contacts, etc)
  + Support forward navigation
  + Good design improves **list comprehension**:
    - Thumbnails anchored at left-edge
    - Font colour, weight used to emphasise important info
    - Distinct info anchored at right-edge (ex: price)
    - Spacing between list items helps visual system ‘chunk’ related info together
    - If more than 2 lines per item – use **list dividers**
    - Eyes are drawn to ‘heavy’ content, ex: images
* Grids/Image Lists: 1D/2D grid of items from a collection
  + Best suited to **image**-heavy entities (photos, recipes, etc)
  + Collection of ‘**cards’** in an organised grid
  + Scrollable in **1** direction
  + Supports forward navigation
    - Expand cards to move deeper into the hierarchy

Search

* Supports forward navigation
* Provide a search interface to **Content Provider**
  + Android abstraction for providing interactions with a content hierarchy

## UI Components

Button

* Communicate **actions** the user can take
  + Perform the action by tapping the button
* **Appearance** tells a lot:
  + Often **subconsciously** – users notice **style** before text/icons
  + importance, connotations (pos/neg), action
* In Material:
  + Low/medium/high emphasis and toggle
  + Colour shows emphasis
* Floating Action Button
  + Perform **primary action** of an activity
  + Floats **above** content, always visible
  + Don’t overuse:
    - Most effective when an activity has 1 action that’s more important than others

Text Field

* Allow users to enter text, typically in forms/dialogs
* Single line/many lines
* Field **type** determines which **keyboard** layout is used: numbers, email, text, etc
* **Hints** – in-situ labels

Text Selection:

* **Contextual actions**: display context-sensitive actions
* **Selection handles**: can be dragged to precisely change selection boundaries

Seek Bars & Sliders

* Sliders allow users to select from a range of values
* Can be **continuous**/**discrete**
* Interactive => well suited to **real-time feedback** (ex: volume, screen brightness)

Selection Controls

* Allow users to make selections:
  + Checkboxes: many items
  + Radio buttons: one item
  + Switches: binary

Dialogs

* **Modal** window that appears in front of Activity content
  + To provide **critical info**
  + To prompt for **decision**
  + All app functionality **disabled** until the dialog has been responded to
* Content:
  + Title
  + Middle content
  + Selection at bottom

Pickers

* Simple way to select single value from set
* Ex: clock time, calendar

Spinners:

* Dropdown menu for making **simple selection**
* Used in **forms** for data input
* Can be combined with other UI elements:
  + ex: text field for email address with spinner for search results, autocompletion

Menus

* Like Spinners, but not linked to particular text field
* ex: context menu

Advice – Giving Feedback

* Informs user that something is happening/has happened
* UI Component Ex:
  + progress indicators,
    - Linear, Circular, Determinate (known), Indeterminate (progress not known/end point unpredictable)
  + Snackbars,
    - Provide **brief messages** about app operations at **bottom** of screen
    - Can have embedded **action buttons**
    - Can disappear automatically or wait until dismissed by user
    - Use: feedback related to an operation in the activity
  + Toasts
    - Provide lightweight feedback about an operation
    - Disappear automatically after short amount of time
    - Part of Android, but not Material
    - Use: system-level feedback about an operation
* Hardware Ex: sound, vibration, notification LEDs, etc

## Lab: Prototyping

* [same link]

# Unit 4: Usability Evaluation

## Usability Evaluation

Evaluation is a critical part of good interaction design:

* Usability
* Improvement
* Not just about testing if app works

Happens throughout design

* Designs from sketches
* Wireframe layout
* Implemented product

Context

* Can’t predict/recreate
  + ex: settings, etc
* Context-based interactions – challenging
  + difficult to control (test location/motion?)
  + Application domains are challenging
    - ex: driving, tourists

How to Evaluate?

* Who to evaluate?
* Where to perform evaluation?
* What to evaluate?

## Evaluation Methods

Evaluate prototypes to:

* identify usability problems
* understand how users interact
* identify areas for improvement

Heuristic Evaluation

* Evaluate prototype with usability experts
* Usability heuristics
  + Jakob Nielsen’s
    - #1 Visibility of system status
    - #3 User control and freedom
* Look for potential design flaws and try to identify issues before getting real users to evaluate prototypes
* One evaluation – no
  + More evaluators – increased coverage
  + Balance cost with coverage
* Desired outcome: list of potential usability issues to fix
* Advantages
  + Can use early
  + No need for users as participants
  + Issues easily fixed
* Disadvantages
  + Can’t find all issues
  + Users act in unpredictable ways
  + Difficult to consider context

Cognitive Walkthrough

* Evaluate prototype by reasoning **how** user will interact
* For each task:
  + break into process of steps
  + reason what user will do next
  + does design support user or work against them?
* Desired outcome: list of potential usability issues to address
* Questions to ask:
  + Will user try to get the right outcome?

Think-Aloud Evaluation

Focus Groups

* Higher level (requirements rather than interaction)

Interviews

Surveys / Questionnaires

Analytics

## Usability Tests

Users perform tasks representative of typical use

* Varied, results generalised
* Under controlled conditions

How to assess usability

* Capture metrics
* Evaluators observe interaction

Usability Metrics

* Efficiency
  + Performance
    - Shows efficiency and effectiveness
    - Task time, no. of errors, success rate, accuracy
  + Domain-specific performance
    - wpm, score in game, etc
  + Cognitive load
    - Important for mobile: demanding usage environment
  + NASA-TLX: Task-Load Index
    - Method for measuring cognitive workload
  + Walking:
    - Percentage Preferred Walking Speed (PPWS)
    - Total distance walked by user
    - Inspect route walked by user
      * Good for location-based interactions
* Satisfaction
  + Metrics aim to quantify attitudes and perceptions
  + Surveys
  + AttrakDiff
    - Evaluates attractiveness of a product
      * pragmatic quality
      * hedonic quality identity
      * hedonic quality stimulation
      * attractiveness
  + Product Reaction Cards
    - Choose adjectives to describe product
  + System Usability Scale (SUS)
    - Likert scale with 10 questions
  + Questionnaire for User Interface Satisfaction (QUIS)
  + Usability Metric for User Experience (UMUX)
    - 4 Likert scale questions
  + UMUX-Lite
    - 2 questions

Designing Mobile Usability Test

* Where to evaluate?
  + Intended use
    - Location and context-based interactions best tested in situ
    - Anticipate demands of context, replicate them
  + Indoors
    - Controlled
    - Consistent
    - Safe
  + Outdoors
    - More realistic
    - Demanding
    - Ideal for location and context info
    - Typical challenges (GPS ambiguity, connection loss, noise, etc)
* How much does the user move?
  + Treadmill
    - Evaluate physical effects
    - Not cognitive effects
    - Safe for early stage
  + Walking around a controlled route
    - Physical effects
    - Cognitive effects
  + Walking “in the wild”
    - Most realistic
    - Hardest to control
    - Evaluate effects, distractions, attention demands
    - Perfect for evaluating social acceptability
* Be conscious of safety
  + Hazards
  + Weather
* Ethical approval necessary
* Minimise risk
  + ex: walking speed, safe location (park), daylight, driving simulator (not on road)

# Unit 6: Interaction on the Move

## Wearable Devices

\* - devices worn on the body

Attributes:

* **Unrestrictive**
  + can do other things
* Unmonopolising
  + no demand for attention
* **Observable**
  + **output**
* **Controllable**
  + **input**
* Attentive
* Communicative

Wearable – always available (observable and controllable)

Can enhance other activities

Input Challenges

* No traditional input
* Limited input
* Moving body parts – accuracy

Output challenges:

* Limited display
* May not always be visible
* Moving body parts – difficult to perceive

History

* First
  + 1960s
  + special shoes with buttons as input
  + output: earpiece
  + cheating at roulette in casinos
  + challenges: electric shocks
  + stopped after $10 000 profit
* Early
  + Big
  + Performance and storage changed much
* Modern
  + Smart Watches
  + Bracelet
    - Myo
      * Muscle activity
  + Rings
    - Logbar, Orii, Oura, Tap Strap 2
    - Depend on smartphone for computational power
  + Headsets/Glasses
    - Google Glass
    - Bose Frames
      * Audio only
      * Bluetooth to smartphone for more power
    - Microsoft Hololens

Social Acceptability

* No attention attracted
  + Weird input actions
  + Obtrusive output
    - ex: audio everyone can hear
* Privacy concerns
  + ex: Google Glass
* Improvement – making system **subtle**
  + Subtle form factors:
    - Appearance of product
    - Location of product
    - Interactions similar to typical behaviours

## Wearable Input

Challenges

* Limited input capabilities
* Mobility
  + Unstable

Input modalities:

* Buttons
  + Simple to use
  + **Eyes-free input**
  + **Mode-switch**
    - ex: start speech recognition, start gesture detection
  + Hard to map functionality to **small no.** of buttons
  + Tradeoff between no. of buttons and **mobile** usability
    - More buttons:
      * Need to focus more
      * Larger
      * Heavier
      * Less comfortable
* Touchscreens
  + Familiar
    - Similar interactions to phones
    - Directly manipulate content
    - Visual feedback
  + Challenges:
    - Small
      * **Fat finger problem**
        + Targets are smaller than fingertip
      * **Screen occlusion problem**
        + Content occluded (obstructed) by fingers
* Touch panels
  + Can be integrated in small devices
    - Glasses (ex: Minuum in Google Glass)
    - Wristbands, rings
  + Lacks direct visual feedback
* Hand Gestures
  + Gesture – motion for communication
  + Eyes-free input (kinaesthetic sense)
  + Can be sensed by wearables
  + Easy to perform:
    - Good at skilfully moving our bodies
    - Good at fine-grained motor tasks
    - Good at memorising movements
  + Challenging:
    - Accurate recognition is difficult
      * Poor recognition -> frustration
      * Larger/slower movements – less subtle
      * Can be improved by simplifying
* Gloves
  + Directly measure finger motions
    - Many degrees of freedom
    - Large gesture vocab
  + Limitations:
    - Social acceptability
    - Uncomfortable
    - Need to put on/off
  + Trade-off: subtlety vs input capability
    - Knuckles?
  + Use cases:
    - VR?
      * Already encumbered by headset
      * Private
* Voice input
  + Can be easy to perform (language precise, expressive)
  + Social acceptability
  + Speech recognition
    - false positives, noise
* Device Movement
  + Can be subtle, discreet, eyes-free
    - One-handed for rings
    - Hand in pocket
    - Watch under a sleeve
    - Hands by side of body
  + Socially acceptable?
    - ex:
      * fidgeting
      * head movement – looking around
      * eyes – gaze

## Wearable Output

Challenges:

* Limited display
* Can be offloaded to other modalities
  + How to represent info using sounds/vibrations?

Modalities:

* Speech and Non-Speech Audio
  + Easy to deliver
  + Social acceptability?
    - Only if obtrusive to other people
    - No concerns if silent to others
      * Bose Frames, Echo Frames (directional speakers towards ears)
* Haptics
  + Touch
  + Vibration – common
    - Notifications
    - Typically simple: on/off
    - Richer haptic (ex: Apple Taptic Engine)
  + Good privacy
  + Very socially acceptable (only noticeable by wearer)
* Projected Screens
  + Novel
  + Expand available display space
    - Larger targets
    - Easier to perceive
  + Can be integrated with other wearables
    - Glasses onto arm
    - watch onto arm
  + Social acceptability?
    - Highly visible to others
    - Potentially weird?

In the Future?

* Implants in body
* (Smart) Tattoos

## Very Mobile HCI

Challenges:

* Divided attention
  + ex: walking, looking around, driving, riding

In general:

* Emerging use cases: physical activity, driving, etc
* Physically, mentally demanding activities – challenging
* Amplifies challenges
* Negative consequences from distraction
  + ex: harm

Cycling

* Challenges:
  + Physically demanding
  + Mentally demanding
    - focus
  + Visual attention focused elsewhere
  + Limited ability to provide input
    - ex: hands on bars
  + Situational impairments
    - ex: changes in lighting (perception of visual output), noise, sweaty hands
* Design within constraints
  + Make it easy to **glance**
    - minimise info, maximise size
  + Make it easy to **provide input**
    - minimise actions, maximise target size
    - focus on **eyes-free**
  + Eyes-free output
    - Tactile feedback
    - Or: visual cues in periphery
  + Customisation
    - Different users, different uses

Running

* Wearables
  + Frees up hands
  + Specialist purpose
    - Mapping, routing, real-time tracking and analysis
  + Physical buttons
    - eyes-free input with tactile feedback
    - more info in screen
    - easier input with moving arms
  + Minimal info: **glance**
  + Customisable data fields

Driving

* Challenges:
  + Safety-critical (risks)
  + Mentally demanding
    - Focus on a lot
    - All sensory channels busy
  + Visual attention should be focused on road
  + Limited input
  + Situational impairments
    - ex: lighting, noise, car vibrations, in-car info (people, music, satnav)
* Reduce ‘**eyes off the road time’** (EORT)
  + Metric in car interaction research
  + Find alts to demanding visual interfaces
    - ex: audio, novel ideas (windshield display)
  + Minimise info/UI
* Make it easy to provide input
  + Support eyes-free
  + Alternative modalities:
    - speech,
    - contactless gestures,
    - etc

# Unit 7: Mixed Reality

## Mixed Reality Intro

Mixed reality

* Space between reality and virtual reality
* Alt: virtual objects integrated in reality
* XR (extended reality) – catch-all

Interface and device no longer necessarily physically coupled

Orientation / Position Tracking for MR

* Why
  + Perspective dependent rendering
    - Essential for depth perception
  + Tracking controllers, hands, etc
* Degrees of freedom
  + Ability of an object to move around in space
  + Translational
    - Heave (up down)
    - Sway (away, incoming)
    - Surge (left right)
  + Rotational
    - Roll (wheel)
    - Pitch (rolodex)
    - Yaw (globe)
  + Together = 6 degrees of freedom
  + ex:
    - Viewfinder: 0 DoF
    - Smartphone: 3 DoF
    - “roomscale” for VR: 6 DoF
      * Inside-Out Positional Tracking
        + SLAM
        + Scan room from headset
        + Pros:

Markerless setups are self-contained, easy to setup, more mobile-friendly

Lower cost

No need for cables/other devices

* + - * + Cons:

Cameras need to be able to fix onto objects (/markers) in environment to operate

Positioning is relative, not absolute (a problem for multi-user experiences)

* + - * Outside-In Positional Tracking
        + External devices determine position of headset
        + ex: VR cameras in corners of room
        + Pros:

High accuracy, absolute positioning

Multiple devices can use the same sensing/tracking

* + - * + Cons:

External hardware required

Often costly (Optitrack – £10 000+)

Needs to be securely mounted in environment

* Technologies
  + Active sensors (emission of signal)
  + Passive sensors (retrieve measurements without interacting with environment)
  + Hybrid / Sensor fusion
* Content positioning
  + Reference frames
    - Location of virtual objects in space
    - **Egocentric**: location based on user
      * “Display locked or View/Body/Hand-fixed content”
      * Why?
        + Mobile
        + Reachable
        + Consistent
        + Low cost, but could cause occlusion
      * Why not?
        + Can be bad for nausea and discomfort
        + Bad for collaboration
        + Breaks implicit association between context and content
        + Social acceptability concerns
        + Reduced accuracy
    - Exocentric: location independent of user
      * Grounded, world-fixed
      * Diegetic Exocentric – fully integrated with fictional setting
      * Why?
        + Memory (good for remembering)
        + Augmentation of environment
        + Reachable physical surfaces
        + Reachable when overlaid on body thanks to proprioception
        + More immersive
        + Collaboration
        + Rendering to a known surface allows better rendering
        + Delineation of spaces
      * Why not?
        + Need SLAM (lots of cameras)
        + Reachability impaired by context
        + Discoverability required
        + Content layout and transitions between spaces

Challenging to design

Saliency vs memorability cs comfort

* + - * + Requires appropriate surfaces

## Augmented Reality

In general

* Part of the MR continuum
  + More towards reality than VR
* Combined real and virtual imagery
* Virtual objects are registered in 3D

History

* 1800s
  + Pepper’s Ghost

Technology

* Mobile/Handheld
  + ex: Ikea furniture try out
  + ex: restaurant menu trying out foods
  + Pros:
    - Low requirements
    - Easy to develop fore
    - Pragmatic
    - Can judge virtual elements against real backdrops
  + Cons:
    - Limited view window
    - Limited immersion / presence
    - Limited/no 3D interaction (interact via phone)
    - Typically “weak” AR – limited knowledge of environment, imprecise tracking (but this is changing)
* Spatial AR / Projection Mapping
  + ex: holograms
  + ex: smart home with walls
  + Pros:
    - Accessible to all in space
    - High resolution/fidelity
    - Limited sensing requirements
    - Immersive
  + Cons:
    - Needs digital twin of the physical object to augment
    - Often perspective-dependent
    - Need multiple projectors to cover 3D volumes/large surfaces
    - Requires installation
* Heads-Up Display (HUD)
  + Smart Glasses
  + ex: Focals by North
  + Pros:
    - Low requirements
    - Easy to develop for
    - Gateway to AR
  + Cons:
    - Limited view window
    - Egocentric only, limited sensing of world
    - Limited/no 3D interaction (interact with wearables/peripherals)
    - Typically, monocular displays (cheaper, lower power usage)
* Headset / Head-Mounted Display
  + ex first: Sword of Damocles (1968)
  + ex: HoloLens
  + Pros:
    - Personal and private augmentations of the world
    - Augmentations rendered correct to your perspective
    - Powerful tool
    - Binocular, depth perception
  + Cons:
    - Higher processing requirements
    - Higher hardware requirements (FoV, rendering technology)
    - Form factors not yet consumer-friendly/fashionable/wearable all day
    - Expensive
    - Typically referred to as “strong” AR – very accurate tracking, seamless integration with reality
* Retinal / contact les
  + Only potential in the future
  + Pros:
    - Discreet
    - Wearable,
    - High field of view
  + Cons:
    - Powering of the displays
    - Communications
    - Fidelity
    - Comfort
* Multi-sensory / Non-visual Displays
  + ex: Bose Frames with audio integration

Future of consumer AR

* Near-term
  + Fashionable HUD-type visual AR glasses
    - Limited fov
  + Auditory AR as standard
  + Connected eyewear, smart glasses
  + Smartphone providing bulk of rendering power
  + Limited camera integration
* Long-term
  + True 6DoF AR headsets
  + Full SLAM tracking

Possibilities

* Extended Reality
  + ex: exocentric signpost
* Diminished Reality
  + ex: Black Mirror White Christmas blocking people from reality
* Altered Reality
  + ex: spatial AR building
* Hybrid Reality
  + ex: AR shoe design
* Extended Perception
  + ex: radiation/ultrasound/UV light

## Virtual Reality

“Virtual reality entails presenting our senses with a computer-generated virtual environment that we can explore in some fashion.”

Headset / Head-Mounted Display VR

* Completely supplants your visual (and mostly supplants your auditory) perception of reality
* Technical properties
  + Display: Binocular 3D stereoscopic, HD
  + Lens: wide FoV
  + Tracking: Low latency head tracking, usually 6DoF
  + Variable ergonomics (size, weight, comfort)

Smartphone-based

* Mobile VR enclosure where the smartphone provides the guts of the hardware

Inside-out Mobile VR

* ex: Oculus Quest
  + Cheapest
  + Best tracking
* Rendering handled by mobile chipset (weaker than PC, but still good thanks to optimisation)
* Notable platform limitations (Oculus Quest requiring FB account)

Outside-In PC BR

* ex: Valve Index
* Significantly higher fidelity
* Tracking handles via 1+ lighthouses
* Con – mobility

Multi-sensory VR

* ex: Meta Cookie+ - illusion based gustatory dusokay
* ex: Virtual Sweet
* ex: Olfactory VR

Non-HMD VR

* Projection/Large Display
  + Similar to spatial/projective AR
  + Room-scale projection
    - CAVE, multi-wall environments
  + Dome projection
    - Hemisphere/spherical displays
    - ex: museums, art exhibitions
    - ex: film locations
  + Vehicle simulators
    - Physical props used to enhance immersion
    - Pros:
      * Physical/haptic realism
      * Motion simulation

Key Characteristics

* Immersion
* Interactivity, Plausibility, Autonomy
* Presence
  + How AR differs from VR
  + with Psychological Realism
  + Factors
    - Social Presence
      * others are there
    - Spatial/Environmental presence
      * environmental reaction to yourself
    - Personal presence / Body Ownership
      * ex: limbs shown, body visible in mirror
    - Systems factors
      * ex: FoV, frame rate, sensory modalities utilised
    - Fidelity/Realism
      * lighting, polygon counts, HD
    - Usability and UX
  + ex: Birdly, the plank
  + Measuring
    - Active area of research
    - Subjective
      * self-report questionnaires (IPQ, PQ)
      * Breaks in Presence (BIPs) – dramatic changes in presence
    - Objective
      * Behaviour (ex: startle response)
      * Physiological – heart rate, skin conductance, temperature
      * Hard to capture

VR and AR

* VR depends on real-world sensing and inputs (VR – part of MR/XR)
  + ex: hand tracking
* Because every sense is not entirely supplanted by simulated/virtual inputs, there is some inherent intermixing of real and virtual

# Unit 8: XR Interaction and Prototyping

## XR Interactions and 3D UIs

XR Interaction Technique Fundamentals

* Interaction Modalities on HMDs
  + Trackable objects external to HMD
    - Active tracking
      * 3DoF using IMU
    - Passing tracking
      * CV object recognition
  + Gaze
  + Speech
  + Physical HMD controls
  + Non-tracked peripherals
* Building blocks:
  + Collisions
    - Direct touch / grabbing
    - 3D objects have a mesh (collection of vertices defining their shape)
    - Collider – can be different geometry for performance
    - 3D engines usually provide primitive types
    - Collision – trigger events
      * Colliders overlapping/interacting
  + Raycasting
    - Cursor/pointer-based
    - Core building block of mid-air controller, pen, gaze interactions in MR
    - Ray – controller

3D Interaction / UIs

* Limitless/infinite
* May be:
  + 2/2.5D
    - planar/curved flat display
  + 3D
    - manipulating objects
    - buttons
    - drawing
    - data visualisation
    - etc
* Designing techniques
  + Non-trivial problem
  + Challenges:
    - Dizziness
  + Reusable components in frameworks
* Challenges
  + Precision
    - ex: hand-tracking
  + Speed
    - UI position can vary relative to user
    - scaling
  + Lack of Constraints / Haptic Feedback
    - Harder to learn
  + Context & Mobility
    - Can have different expectations
  + Social Acceptability / Physical Bounds
* Ergonomics
  + Depth of content
    - Conflicting distance info on where to focus
      * Vergence-accommodation conflict can lead to eye strain
    - Use of Depth in UIs
      * Can be detrimental
    - UI placement Legibility
      * Screens with different intended distances but same legibility
  + Content size in relation to FoV
    - Each virtual display / piece of content should fit within the comfortable field of view of the headset
  + Ergonomics of Head/Neck/Eyes
    - Place UI where it’s easy to work with and read
    - ex: Shoulder movement, visual centre, neck movement
    - Angled – scrolling more difficult
  + Ergonomics of Upper Body & Fatigue
    - Fatigue when arm moves higher
    - Larger muscle groups are more tiring and less accurate
    - Rested elbow/arm interactions preferable

Simulator Sickness / Cybersickness

* Causes
  + Sensory mismatch
    - ex: accelerations, rotations
    - Vection – illusion of self-motion
  + Low framerates
  + Unstable tracking / mismatched motion
  + Motion parallax / depth cues
* Avoiding
  + Maintain recommended framerates
  + Avoid vection
  + Avoid abrupt movements of content
  + Be careful with use of egocentric head-locked content
  + Move content slowly and smoothly
  + Short animations, slow down timing
  + Animate from down/left/right/fade instead of Z
  + Allow plenty of world to be seen during animation

(Lack of) Standards / Layout Managers

* Lots of competing frameworks

Accessibility

* Currently woeful
* ex: SeeingVr
* At least – try to allow for some customisation/variation
* Assess available accessibility tools as you go

## Prototyping XR Experiences

Why?

* Jumping to dev is costly
  + Takes time to develop, choose right features
  + Harder to refactor
  + Harder to explore divergent concepts
* Why prototype in XR:
  + Simulate sense of scale/immersion
  + Get user impression on a realistic, high fidelity prototype
  + Can make quicker edits to the scene
  + Spatial apps need spatial creation tools

Spatial Prototyping

* ex: Microsoft Maquette
  + Windows only
* ex: Tvori
  + Support for animations
  + Can animate user workflows
* 360 video / Panoramas
  + Provides basis for seated/standing VR env
* 360 sketches
  + ex: 360proto
* Shared/Multi-User Environments
  + Mozilla Hubs, VR Chat
  + Multiplayer

Models & Modelling

* DIY, PC-based
  + Hardest
  + Blender, Cinema4D, Paint3D
* DIY, headset-based
  + Tilt Brush by Google, Google Blocks, Gravity Sketch
* Scavenging existing assets
  + Unity asset store
  + Licensing concerns
* Reality Capture / Photogrammetry
  + Matterport, 3DF Zephyr, etc
  + Typically need LIDAR/ Time of Flight cam er sensor
  + Film/Scan objects which get translated to virtual objects
  + Caveats
    - Meshes need simplifying/tidying/texturing
    - Animation difficult
  + Great for singular objects without moving parts/ with static env

# Unit 9: People

## Social Acceptability

Perception of UI by user and other people

Concern in mobile interaction because of varying context

Multiple perspectives:

* **User**: how comfortable does the user feel?
* **Spectator**: does interacting appear normal?
  + Not part of interaction
  + Meeting expectations, social rules

Interaction Visibility

* Input vs output **visibility**
  + Input actions may be **highly visible** (gestures, speech, etc)
  + Output effects may be **invisible** (small screen, haptics)
* **Input** may draw **unwanted attention**
  + Unusual actions (tapping on face, waving hands around, etc)
  + Disruptive actions (speech input, clicking fingers, tapping on surfaces)
    - Making spectators get involved

Situation

* Depends on **situation** in which an interaction occurs
  + **Audience**: alone, family, friends, colleagues, strangers?
    - Users’ **willingness** to make interactions
  + **Location**: home, office, train, street, pub?
    - Busy locations – avoid unwanted attention

Interaction Design

* **More** acceptable: hands near torso, small movements, short duration
* **Less** acceptable: hands away from body, large movements, long duration

Device Location (Design)

* **More** acceptable: wrist, forearm, hands
* **Less** acceptable: neck, waist, torso

Spectator Familiarity

* Smartphones: more established, more **easily explainable** user behaviour
* Smart glasses: less familiar, fear of the unknown, assume the worst
* Attitudes can and do change

Design for Social Acceptability

* Avoid **unusual** input/output
  + Is there a clear explanation for it?
  + ex: tapping nose, using elephant noises for button-press feedback
* Avoid I/O that **may draw attention**
  + ex: clapping hands, having watch that flashes bright colours
* Avoid actions that don’t have a **clear target**
  + Is it obvious that an action is directed at a device?
  + ex: “Is he waving at me or is he shaking his wrist?”
  + Consider **subtler** alternatives
    - Can you minimise action visibility?
    - ex: twisting ring vs tapping on a watch
    - ex: haptic feedback vs audio feedback

Evaluating Social Acceptability

* Very subjective
* Typically evaluated using surveys, e.g.:
  + What would you think if you saw someone else performing this gesture (e.g., when walking down the street)?
  + In which locations would you use this gesture? Tick all
    - Home, Pavement, Driving, Bus/Train, Pub/Restaurant, Workplace
  + Who would you perform this gesture in front of? Tick all
  + When would it be appropriate to use this gesture? Tick all
    - Home alone, home with family, at work alone, at work with colleagues, in public around strangers, in public with friends
  + Write a number from1 to 5 in each of the circles to indicate how comfortable you felt performing a gesture at that particular position?

## Accessibility

Accessible Design

* Interaction design is about improving UX – for **all** people
  + For all users
  + Some users may find it difficult to interact
  + Many challenges to accessibility: physical, cognitive, motor, etc
* Important to design with accessibility in mind at all times
  + ex: aging population, situational impairments

Screen Readers

* Visual impairments
  + People with blindness may be unable to see the screen at all
  + Low vision – small text
  + Cataracts – screen too bright
* Alternative interactions
  + **Screen readers** – mobile devices **‘speak aloud’** their screen contents
    - Only useful with effective implementation
* Interactions
  + Modes:
    - **Explore by touch**: tapping, dragging, swiping to find items on screen
      * screen reader speaks aloud what the user touches
    - **Linear navigation**: moving through UI elements in linear order
  + More **exploratory** in nature
    - User and device take turns
    - Every action has an **extra step**
      * Tap a target on screen to find out about it, *then* open/interact with it
* ex: iOS Voice Over
  + Different interactions to
* Platform Integration
  + Built in to Android (TalkBack) and iOS (Voice Over)
  + Compatible with all UI elements at the platform level
    - Not compatible with all content by default (images, videos)
* Supporting
  + Build UIs that help the screen reader give quality output
    - Don’t necessarily need to change appearance
    - Implementation-level meta-data
  + Consider: item order, item groups, descriptive text

Item Order

* Same appearance, different **linear navigation** behaviour
* Order determined by **layout hierarchy** in code
* Consider **focus order**
  + Widget attributes (ex: nextFocusRight on Android)

Item Groupings

* Linear navigation goes through items individually
* Simplify screen reader output by grouping info together
  + group label \_ content into a single screen reader ‘phrase’
* How:
  + Android: related UI elements in a single focusable layout group
  + iOS: define UIAccessibilityElement

Descriptive Text & Hints

* Should read **content descriptions** and **text hints**
  + Content description: spoken representation of content
    - For images:
      * Concise, descriptive
      * If it has caption, alt text describes **what it looks like**
  + Text hints: suggestive text, like ‘tooltip’

Evaluating

* Following guidelines helps improve accessibility
* Perform automated accessibility audit
* Evaluate apps with Accessibility Scanner (Android)
  + Simple checks on everything on screen
* Test your apps with the accessibility tools users might use
  + Are your UI and app content understandable?
  + Are touch to explore and linear navigation modes both usable?
* Test apps with accessibility experts and real users?
  + People experiences with accessibility tools who *don’t know the app*

## Privacy

Case Study: Strava

* Social network for joggers
* Users upload activity data for cycling, running, etc
  + location x time x physiological data
* Potential privacy concerns?
* Strava heatmap
  + Combines all activity data from last 2 years
  + Shows where people are riding/running
  + Useful for finding popular cycling routes
  + Useful for intelligent navigation
    - Choose routes cyclists already use
  + Concerns:
    - Individual users/households identifiable
      * (even though data is aggregated)
    - Revealing users’ homes
      * … to bike thieves?
    - Accidentally reveals US Army base layout
      * Users overlooked implications of sharing
      * Strava mostly used by westerners
        + Revealing sensitive areas in places without local users
* “Fly-By”
  + Users you “flew by” on your activity
  + See full route of people whose activities would otherwise not be discoverable to you
* Whose responsibility?
  + Users?
    - Have not created ‘**privacy zones**’
    - But: probably didn’t expect their data to be publicly shared
  + Strava?
    - All users were **opted-in** to sharing by default
    - Allegedly, ignored **privacy settings** in their aggregated data
    - Not pro-active about enabling privacy controls

Summary

* Privacy violations may occur in **future**, not just ‘now’
  + Data created **opportunities** for future privacy violation
* Partly due to ambiguity over data **scope**, **use**, and **access**
  + What data is collected?
  + What else is it used for?
  + Who can access?
  + What can users do to control access to data?

# Summary & Revision

Unit 1: Context

* Context – situation in which a device is being used
* Motivation:
  + Reduce **interaction time**
  + Allow users to **focus on completing their task**
  + Expose users to **new opportunities** (new nearby coffee shop, etc)
  + Overcome **situational impairments** (increase screen brightness, etc)
  + Create **elegant** but **efficient** UIs (reduce irrelevant info, etc)
  + Improve **situational awareness** (when walking/cycling/driving/etc)
* Context Ambiguity
  + Uncertainty inherent in context sensing
  + Reveal uncertainty through **feedback**
    - Users then know ambiguity is present and might affect system behaviour
    - Users then know if their **corrective actions** are having a **positive effect**
    - Users are often forgiving of systems that are honest

Unit 2:

* Device Constraints
  + New **interaction techniques** require more processing power
    - ex: always-on sensing, machine learning, recognition algorithms
  + New **applications** and **use-cases** require more processing power
    - ex: **VR** requires high-res, high-FOV, low-latency graphics
    - ex: **AR** requires object recognition
  + **Screen size** vs **battery** vs **processing power**?
* Sensing New Interactions
  + Sensors
  + Recent ex:
    - Speech input
    - Mid-air gesture
    - 3D depth sensor
    - Grasp pressure input
    - Touchscreen pressure input
  + There are interaction challenges inherent in these
* Design for Mobile Use
  + Design for **short bursts** and **fragmented interaction**
    - Attention is divided between interaction and other tasks
    - Attention constantly switches between concurrent tasks
  + Implications/recommendations:
    - **Simplify** interface: make it easier
    - Anticipate **lack of attention**
    - Anticipate **slowness**
  + Think about challenges of mobility
    - device constraints
    - connectivity
    - environment
    - impact of moving
    - input & output challenges
    - social acceptability

Unit 3: Interaction Design

* **Design funnel**
  + **Elaboration**: generating new ideas, expanding design space
  + **Reduction**: selecting best ideas, reducing design space

Unit 4: Evaluation

* Happens at all stages with all prototypes
* Motivation:
  + Identify usability problems
  + Understand how users interact
  + Find places of improvement

(Unit 5: Android (optional))

Unit 6: Wearables

* Interaction is always available
  + Provide input when needed
  + Perceive output when needed
  + Interaction is “constant”
* Input challenges:
  + **Fat finger problem**
    - Targets are smaller than fingertip
  + **Screen occlusion problem**
    - Content occluded by fingers
* Other challenges:
  + Social **acceptability**
  + Usable **interactions**
    - Not many buttons, small touch area, gesture and speech inaccuracy
  + Usable **output**
    - Limited display space, divided attention
* Very Mobile HCI
  + ex: cycling, driving, running
  + Challenges are amplified
  + Consider context
    - I/O, situational impairments, not demanding too much attention

Unit 7: Mixed Reality

* Virtuality continuum: Reality – AR – Augmented Virtuality – VR
* Device capabilities
* Device **tracking**
  + **Outside-in** vs **inside-out** tracking, degrees of freedom
* Content **positioning**
  + **Egocentric** vs **exocentric** frames of reference

Unit 8: MR Interaction

* How to interact
  + What’s available
* Challenges
  + Sensing and technology limitations
  + Usability issues
  + Ergonomics
  + Social acceptability
  + Nausea

Unit 9: Social Acceptability

* \* - how a user interface is perceived by the user and by other people
* Varying usage context
  + Different places, people, social norms, settings
* Affects I and O
  + Input: acceptable/weird thing to do
  + Output: attracts attention/annoys
* Privacy
  + Data creates opportunities for future privacy violation
  + Think about data scope, use, and access
    - What data is collected?
    - Who can be identified from the data?
    - What is it being used for?
    - Who can access that data?
    - What can users do to control access to data?

Exam Guidance

* Similar format to last year’s exam
  + Use past exam papers and Moodle quizzes to practice
* Think about how everything fits together
  + Context, device limitations, interaction challenges, mobility, multi-tasking
* Some questions have multiple valid answers – **justify** yours
  + Analyse, argue, critique, compare, contrast, evaluate, explain, justify, etc