Agenda

- Don Norman's Design of Everyday Things
- Hick's Law
- Android Design Principles
Design Concepts from Don Norman

- **Visibility**
  - **Discoverability**
    - A product should **afford** its use, or
    - Use external **signifiers**
  - **Understandability**
    - Mapping: what controls what?
    - Direct vs. indirect mapping
    - **Support the motive**
  - **Feedback**
- **Expect errors**
  - **Constraints**
  - **Undo**
- **Make common tasks easy**
Feedback

- What has been done?
- What did I do?
- What did you do, computer?

- Visual, audio, or tactile feedback
  - Password entry on mobile
  - 'Click' of seat belt
Feedback needs to make sense

• Immediacy
• What makes sense can be culturally modulated
• Don't give too much feedback
  - Some safe driving systems
Constraints

- Make errors impossible
  - Example: grayed out menu options
  - Email client: don't enable Send until a To: address is specified
  - Calculator: don't allow entry of bad expressions
Lack of Constraints leads to errors

- Possible to put a battery in the wrong way
- Possible solutions:
  - Design battery holders so only one way works
  - Design batteries so that orientation doesn't matter
- Also: credit card stripe readers
Activity-Centered Controls

- Activity-centered controls **reduce errors and make common tasks easy**
- They must be well researched to cover all common cases
  - Offer an advanced mode, extra choices menu, and/or customization to help users with uncommon needs
  - Don't be too constrained
Android Principle: Launch into Action

When you know the common case

- Don't present a menu
- Just launch into the most likely activity.
- Then user can change it.
- Don't take it too far!
  - Netflix doesn't launch a movie on start-up
  - But it gives you one prominent suggestion and rows of smaller ones, not a menu where you first must choose Comedy, Drama, etc.
Hick's Law: Avoid too many choices

- 1952: Measure response time given a stimulus
  - \( RT = a + b \log N \)
    - \( RT \): time to choose a response
    - \( a \): base processing time
    - \( n \): number of choices
- Jam study (Iyengar and Lepper, 2000): People buy more jam when given 6 options vs 24.
Microwave Smackdown!

- Compare each microwave on
  - Discoverability
  - Understandability
  - Opportunity for Feedback
  - Too Many Choices?
  - Too Few Choices?
Microwave 5

Microwave 6

Assume LED number panel on top

Microwave 7
Undo: Software's super power!

- Physical objects don't support Undo well. You can't undo:
  - coffee brewing
  - cutting physical paper with scissors
- Software's power comes from undo:
  - you can undo a 'crop image' command
Undo: you have to plan for it

- Adding Undo to software later is tough
  - It has to be designed in from the start
- Retain a history of commands to support multiple undo
- Undo is always better than an “Are you sure?” popup.
Android Design Principles

• Ideas from activity theory
  – Support activity towards the motive object
    • Launch into Action
    • Only interrupt me if it's important
  – Support the automatic mind
    • Pictures > words
    • If it looks the same, it should act the same
    • Give me tricks that work everywhere
Android Design Principles

- Ideas from classic HCI (Norman, etc.)
  - Visibility
    - I should always know where I am
    - Only show me what I need when I need it
  - Understandability
    - Keep it brief
  - Constraints and Undo
    - Decide for me but let me have the final say
  - Make common tasks easy
    - Get to know me / Never lose my stuff
    - Do the heavy lifting for me
    - Make important things fast
Android Design Principles

- Positive reinforcement is good!
  - Delight me in surprising ways
  - Real objects are more fun than words
  - Let me make it mine
  - It's not my fault
  - Sprinkle encouragement