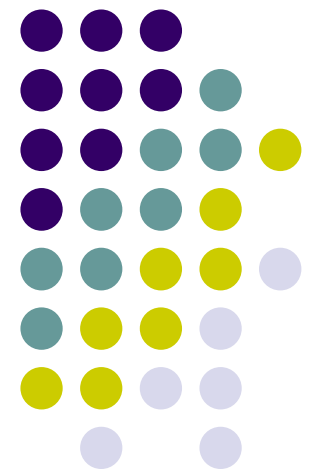# CS 528 Mobile and Ubiquitous Computing
Lecture 7: Final Projects + Smorgasbord of Stuff!!

Emmanuel Agu

smor·gas·bord

/ˈsmôrɡəsˌbôrd/

noun

a buffet offering a variety of hot and cold meats, salads, hors d'oeuvres, etc.

- a wide range of something; a variety.
  "the album is a smorgasbord of different musical styles"

Translations, word origin, and more definitions

Final Project Overview & Proposal Guidelines

Final Project

- Most projects will probably build an app

- App solves some societal problem

- App should be **mobile** or/and **ubicomp**

  - **Mobile?** Probably location-dependent, maps, deliver time-sensitive information

  - **Ubicomp?** Uses at least 1 sensor (accelerometer, microphone, camera, etc)

- Don't build app that has no mobile or ubicomp aspects
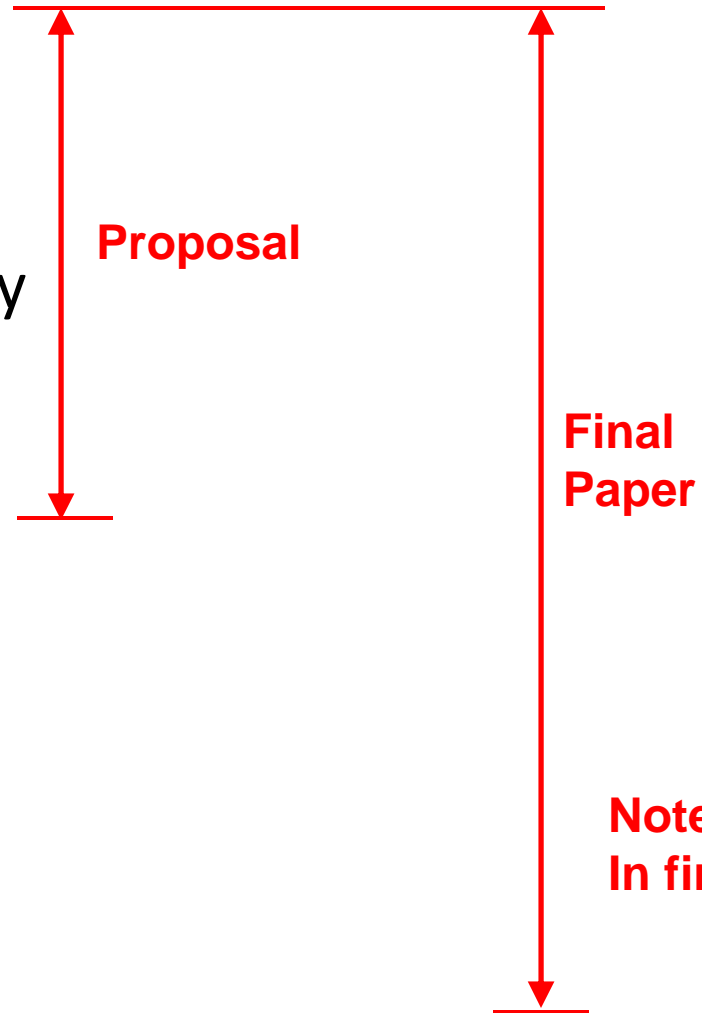
- If you have questions, talk to me

Typical Paper

- Introduction
- Related Work
- Approach/methodology
- Implementation
- **Project timeline**

- Evaluation/Results
- Discussion
- Conclusion
- Future Work

Proposal

Final Paper

Note: No timeline In final paper

Proposal

- Submit (Written 2 pages max PDF file): due March 23!!
  - Introduction
    - List team members
    - State problem app will solve. Preferably has social benefit
    - Why is problem important?
    - E.g. Find statistics: How much time, money, resources is being wasted on this problem today? How many people problem affects
    - Potential gain: how will your solution save time, money, etc?
  - Related work
    - What other research has been done to solve this problem (academic + commercial apps)
    - How is your app/approach/work different?

Proposal

- Methodology/Design/Tools:
  - Brain storm!
  - Summary of what you intend to do
  - How you intend to do it? Build android app, use scenario, etc
  - App screen mock-ups:
    - Hand-drawn? Android Studio? Lucid Charts?
  - Don't promise too much,
    - Some features can be future work

Methodology
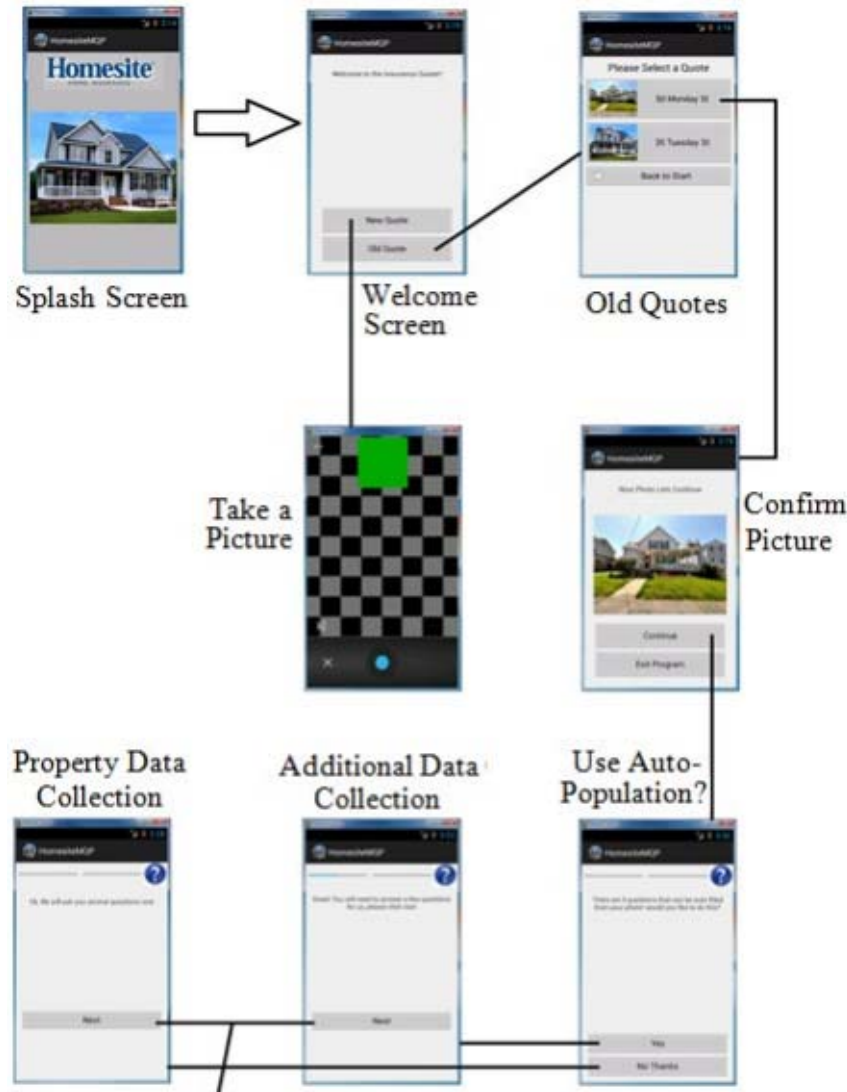
- Preliminary design from team

- Screen mock-ups + flow

- Use Android Studio Design view, lucidcharts.com, hand-drawn?

Proposal

- Implementation **plan**:
    - E.g. Android, what modules? external tools? Packages? etc
- Timeline
    - Break down tasks, mini-deadlines, allot time for each task
- Proposal due March 23 (Next week)!!

Separate Vision and Prototype

1. Big picture if funds/time not an issue (e.g. company of 200 employees over 6 years)

Vision

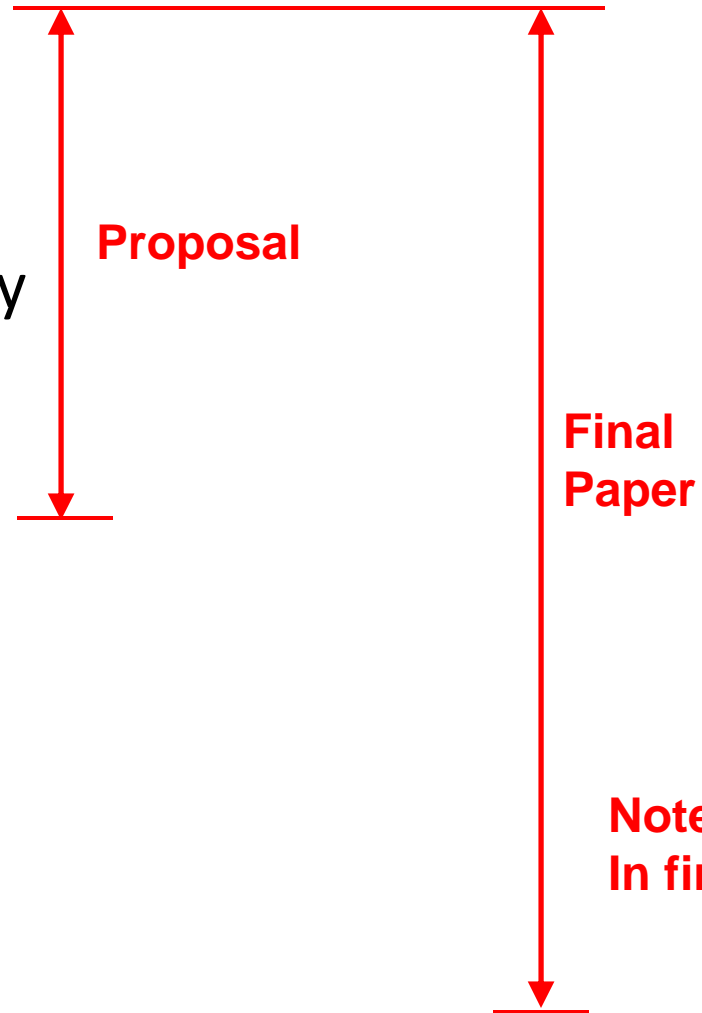**2. Which reasonable Subset of the big vision can you do in 6 weeks?**

Can make simplifying assumptions

Prototype

Typical Paper

- Introduction
- Related Work
- Approach/methodology
- Implementation
- **Project timeline**

- Evaluation/Results
- Discussion
- Conclusion
- Future Work

Proposal

Final Paper

Note: No timeline In final paper

Final Paper: Evaluation

- Depends on what your project is.

- **Basic question:** How well did your solution work?

  - User studies

  - Measure performance. E.g. energy consumption, bandwidth consumption, etc

- User Studies

  - Pre-Survey:

    - Establish problem exists, need for your app, gather/refine requirements

  - Post-Survey:

   - Get users to use/rate your app, ask about likes dislikes

Recruiting Subjects For User Studies

- 3Fs: Friends, Family and ??
- Classmates (Do a trade with another group)
- On campus: post flyers, set up table at campus center

Discussion, Conclusion, Future Work

- Discussion:
    - How was your app received? Rationalize your findings in user studies, Say why certain features worked, did not work, etc

- Future work
    - Talk about features that would extend prototype
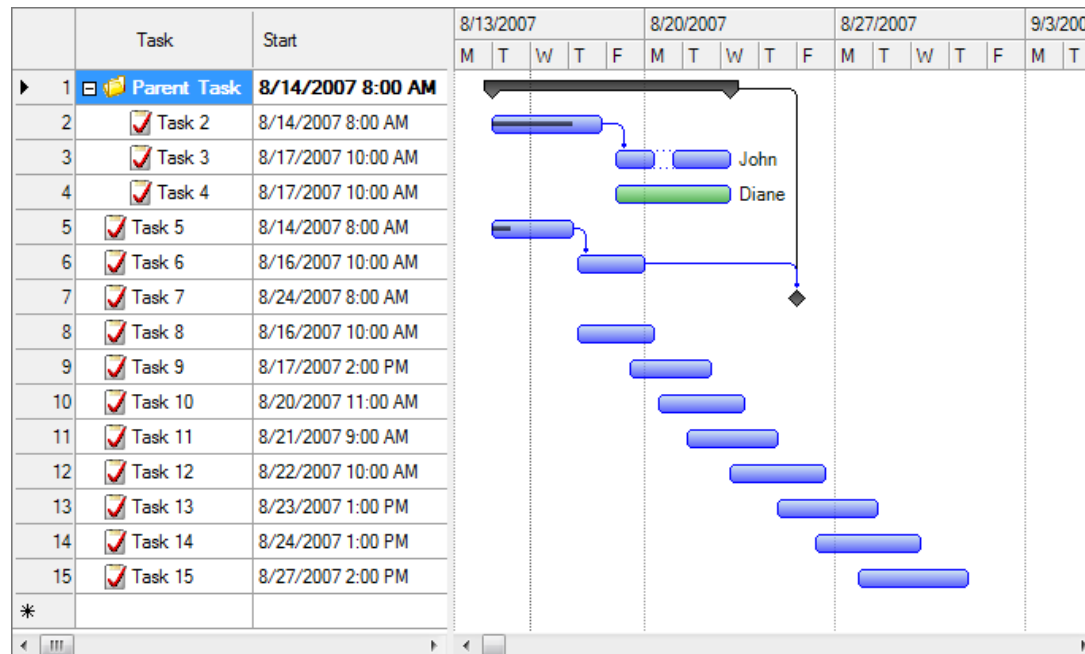   - Revisit big vision
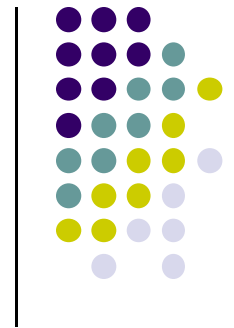
Your Team

Some Team Tips

- You already have a team!

- Everyone (team members) doesn't have to do everything equally

- Team members can work on project aspects they are good at

- Example: Who is good at:

  - Android UI design (Android Studio design view, XML file, widgets, nice look)

  - Android programming (database, sensors, maps, backend)

  - Experimental evaluation/user studies

  - Machine learning

  - Writing, making presentations

Some Team Tips

- Team should have an honest conversation
- Doing something different doesn't mean chilling
- Consider team online management tools, gantt charts, etc
- Assign tasks, mini-deadlines (every few days)
- Integrate features every few days => new version
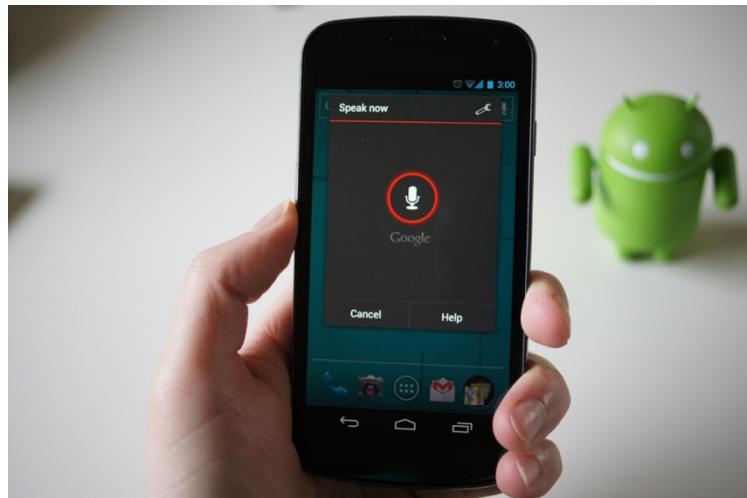- **Mantra:** Always have a working prototype, improve

	Task	Start	8/13/2007					8/20/2007					8/27/2007					9/3/2007	
			M	T	W	T	F	M	T	W	T	F	M	T	W	T	F	M	T
1	Parent Task	8/14/2007 8:00 AM																	
2	Task 2	8/14/2007 8:00 AM																	
3	Task 3	8/17/2007 10:00 AM						John											
4	Task 4	8/17/2007 10:00 AM						Diane											
5	Task 5	8/14/2007 8:00 AM																	
6	Task 6	8/16/2007 10:00 AM																	
7	Task 7	8/24/2007 8:00 AM																	
8	Task 8	8/16/2007 10:00 AM																	
9	Task 9	8/17/2007 2:00 PM																	
10	Task 10	8/20/2007 11:00 AM																	
11	Task 11	8/21/2007 9:00 AM																	
12	Task 12	8/22/2007 10:00 AM																	
13	Task 13	8/23/2007 1:00 PM																	
14	Task 14	8/24/2007 1:00 PM																	
15	Task 15	8/27/2007 2:00 PM																	

What other Android APIs may be useful for ubicomp?

Speaking to Android
Ref: Professional Android 4 Development, Meier, Ch 11, pg 437

- Speech recognition:
  - Accept inputs as speech (instead of typing) e.g. dragon dictate app?
  - Note: Google (remote) service Requires internet access
- Speech-to-text
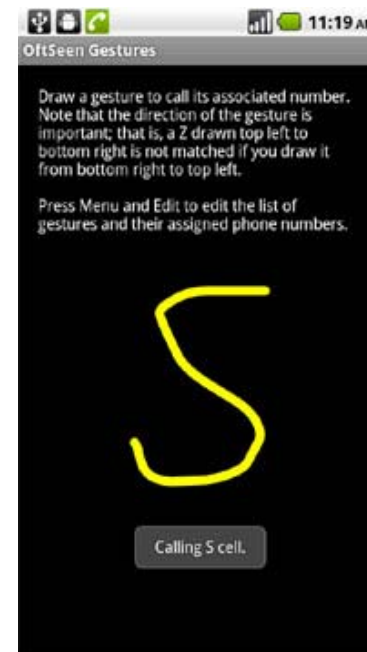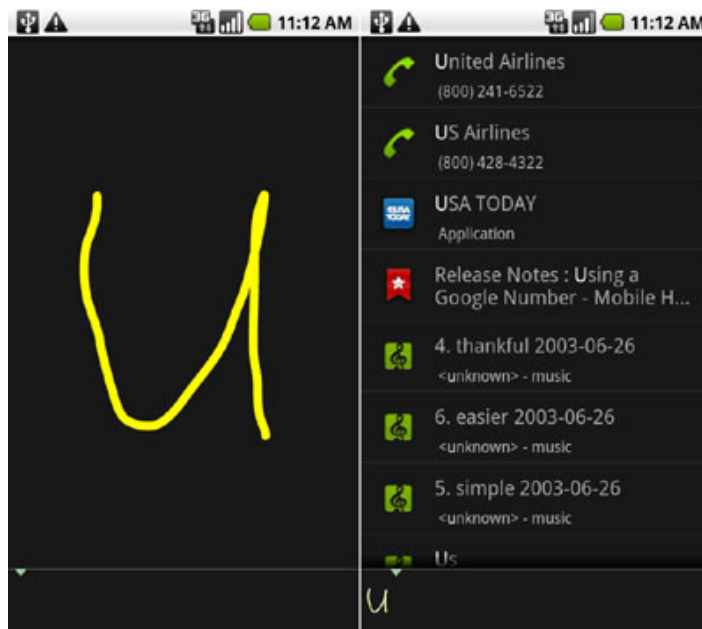 - Convert user's speech to text. E.g. display voicemails in text

Gestures
Ref: 3 cool ways to control your phone
http://www.computerworld.com/article/2469024/web-apps/android-gestures--3-cool-ways-to-control-your-phone.html
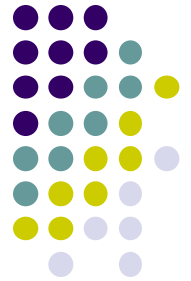
- Search your phone, contacts, etc by handwriting onto screen

- Speed dial by handwriting first letters of contact's name

- Also multi-touch, pinching

Doing More with Locations: Geocoding
Ref: Professional Android 4 Development, Meier, Ch 13, pg 513

- Maps, GPS discussed so far use longitude/latitude to pinpoint geographic addresses

- Users more likely to think in terms of street addresses

```
Latitude: 37.422005 Longitude: -122.084095

Address:
1600 Amphitheatre Pkwy
Mountain View, CA 94043
Mountain View
94043
United States
```

- **Geocoder** converts between longitude/latitude and street address
  - **Forward geocoding:** Finds latitude and longitude of an address
  - **Reverse geocoding:** Finds street address for given longitude/latitude

- Can also set proximity alerts
  - Intent delivered to your app when you are within a pre-set distance from a given location

More on Audio, Video and Camera
Ref: Professional Android 4 Development, Meier, Ch 13, pg 513
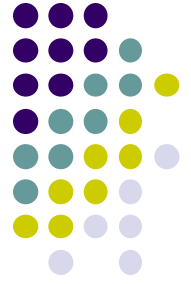
- Android MediaPlayer previously used to play audio
- Media Player can also:
  - Play videos (e.g. MPEG 4)
  - Record audio and video
  - Preview video
  - Manipulate raw audio from microphone/audio hardware, PCM buffers
    - E.g. if you want to do audio signal processing, speaker recognition, etc

More on Audio, Video and Camera
Ref: Professional Android 4 Development, Meier, Ch 13, pg 513

- Can control Camera parameter settings
  - Flash mode, scene mode, white balance
- Camera can also do face detection and feature recognition
  - Detects face up to a max number of faces + accuracy

RenderScript

- High level language for GPGPU
- Use Phone's GPU for computational tasks
- Very few lines of code = run GPU code

Wireless Communication

Ref: Professional Android 4 Development, Meier, Ch 16, pg 665

- Bluetooth
    - Discover nearby bluetooth devices
    - Control your smartphone's (device's) discoverability
    - Communicating over bluetooth

- WiFi
    - Scan for WiFi hotspots
    - Monitor WiFi connectivity, Signal Strength (RSSI)
    - Do peer-to-peer (mobile device to mobile device) data transfers

Wireless Communication
Ref: Professional Android 4 Development, Meier, Ch 16, pg 665
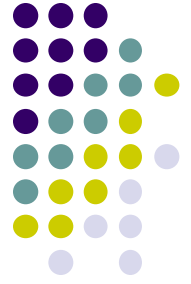
- NFC:
  - Contactless technology
  - Transfer small amounts of data over short distances
  - **Applications:** Share spotify playlists, Google wallet
  - **Google wallet?**
    - Store debit, credit card on phone
    - Pay by tapping terminal
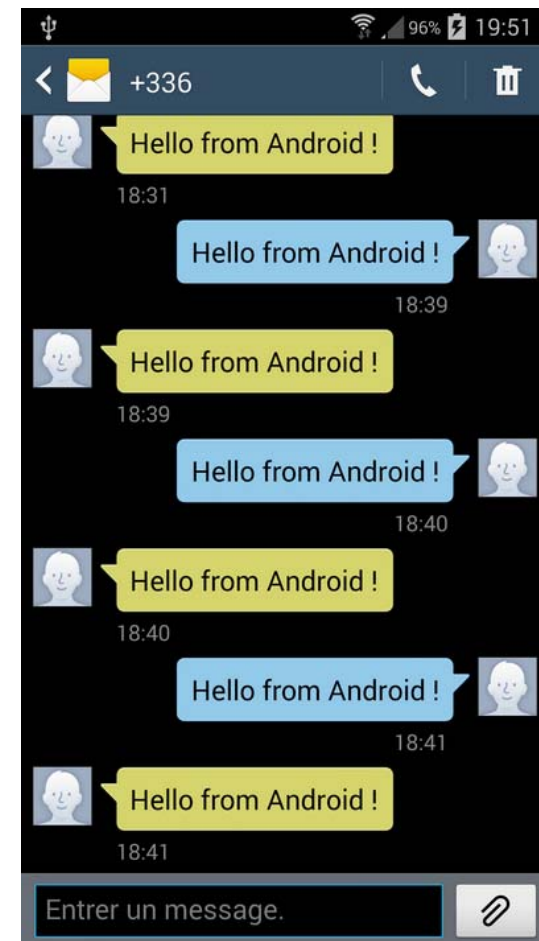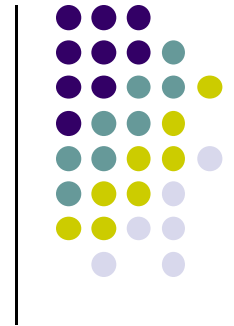   - Fly through checkout?

Telephony and SMS

Ref: Professional Android 4 Development, Meier, Ch 17, pg 701

- **Telephony:**
  - Initiate phone calls from within app
  - Access dialer, etc
- **SMS:**
  - Send/Receive SMS/MMS from app
  - Handle incoming SMS/MMS in app

Finding Idea to Work on

Pick an Idea to Work on

- Examples of previous projects from grad class:
  - Hearing aid
  - WiFi vulnerability
  - Mobile tweeter mining (mobile computing, ubicomp stuff),
  - weather prediction along user's path

- Projects from Andrew Campbell class
  https://docs.google.com/document/d/1hg44pm9PPPnIxBfNthAktUD9XoHBLmkMdmq6BmJiWaI/pub

- What else is detected in ubicomp (5W's, 1H), examples ideas, how to do it in Android

Coming up with a Project

1. Click on papers,
    i. What areas you like?
    ii. What are your strengths? Machine learning? Signal processing?
2. Find papers you like within area or search Google Scholar, ACM digital library or IEEE Xplore
3. Can each paper be extended?
    a. Look at future work
    b. Repeat experiments + other things they didn't try. E.g.
        i. Re-implement a simple idea: E.g. Bewell
        ii. Implement PART(S) OF complex idea (e.g. place sense paper)
        iii. Propose new idea based on your prior knowledge/experience (GREAT!!! Maybe publishable?)
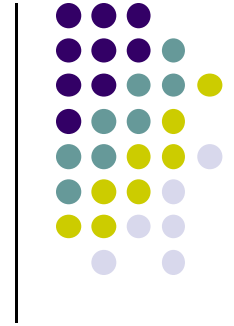
Other Random Project Ideas?

Some Project Ideas

- **Machine learning:**
  - Detect personality type from detecting/analyzing daily interactions.
  - E.g. number of friends seen per day, number of people talked to per day, activity levels/type, etc.

- **Signal/processing:**
  - Detect speaker, extract conversations, convert speech to text, record
  - Detect emotion/stress levels from speech
  - Detect sleep duration, quality detection from accelerometer, microphone (iSleep paper)

Some Project Ideas

- **Image/Video Analysis:**
  - Detect a person's emotion/mood from an image video of their face
  - Detect if a person/student watching a youtube video is engaged/not engaged

- **Mobile Twitter**
  - Search Twitter messages, analyze how much important mobile topics are being discussed (e.g. security, malware, health)

References

- Busy Coder's guide to Android version 4.4
- CS 65/165 slides, Dartmouth College, Spring 2014
- CS 371M slides, U of Texas Austin, Spring 2014