

# COURSE OVERVIEW

- COMP 5517 -- Human Computer Interaction
- Open to postgraduate (MSc and research) students
- Prerequisite: Some background and knowledge of programming.
- Lectures/Labs: Thursday 18:30-21:30
  - Content for labs:
    - Learning about project-related technical issues
    - Presentations
  - Location for lecture/labs may change as needed: Keep an eye on Blackboard!

# GRADING

- Continuous Assessment: 60%
  - Qualitative Evaluation (40%)
  - Class Project (55%)
  - Class Participation (5%)

# QUALITATIVE EVALUATION

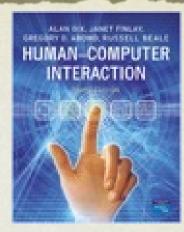
- Goal: Conduct an end-user evaluation and comparison of real interfaces
- Grade Distribution:
  - Execution (65%)
  - Report (35%)
- Group Project: Maximum 4 ppl per group.
- Details will be released in Week 3

### FINAL PROJECT

- Goal: Design and implement a prototype physical and ubiquitous computing environment.
- Design and Proposal (20%) -- Evaluated via written proposal.
- Checkpoint 1 (15%) -- First Prototype. Evaluated via demonstration.
- Final Deliverable (65%)
  - Functionality and Design (65%)
  - Report (20%)
  - Presentation (15%)
  - Class participation (5%)
- Group Project: Maximum 4 people per group.
- Details and practice session sometime after Week 4

#### TEXTBOOK

- Two recommended texts:
  - Human-Computer Interaction, Dix, Finlay,
    Abowd and Beale, Prentice Hall
  - Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition) Ben Shneiderman, Catherine Plaisant, Addison-Wesley Press





# A FINAL WORD...

- HCI is about how humans interact with computers, and how to make their job easier.
- It is very interdisciplinary, with aspects of psychology, cognitive science, and design as well as computer science.
- Some bits may even seem very obvious at first blush.
- You may even wonder why we have a course devoted to this topic.
- However, remember two things:
  - Whether a system is successful or not, usually depends more on its design than on its utility.
  - Good designs are harder to realize than you think!