


# DC2260 HCI

## Introduction to HCI

Module Tutor: Dr. Lilit Hakobyan  
[hakobyl2@aston.ac.uk](mailto:hakobyl2@aston.ac.uk)



# introduction...

- ▶ what do we mean by **human-computer interaction**, **usability**, and **user interfaces**?
- ▶ “**human-computer interaction (HCI)** is the discipline concerned with the design, evaluation, and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them” (Hewett et al., 1992)
- ▶ **user interface** is often shortened to “**UI**” – it refers to the user interface artefact (i.e., the “thing” that users interact with to engage the functionality of a software system)
- ▶ **usability** is the goal of HCI practitioners – for a UI to be usable, it has to be appropriate for the users and for the users’ tasks
  - ▶ don’t mistake with **user friendly** which is an ill-used term
    - ▶ *users don’t want to be friends with their computers!*
    - ▶ *“friendly” is a binary concept and usability is a lot more complex than that!*

Hewett, T.T., Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G., and Verplank, W., (1992,1996), *ACM SigCHI Curricula for Human-Computer Interaction, Chapter 2: Human-Computer Interaction*, retrieved from <http://www.sigchi.org/cdg/cdg2.html>

# what influences UI success?...

- ▶ success of a UI is usually determined by a broad range of issues:
  - ▶ type of user
  - ▶ type of task
  - ▶ hardware constraints
  - ▶ social and cultural limitations
- ▶ the above factors + user interface = user experience

# who should participate in HCI?...

- ▶ broad, multidisciplinary field
  - ▶ computer scientists
  - ▶ ergonomists
  - ▶ psychologists
  - ▶ social scientists – e.g., sociologists, anthropologists
  - ▶ cognitive scientists...
  - ▶ ...and let's not forget HCI practitioners (usability engineers)
- ▶ majority of the rest of your course emphasises programming...
- ▶ ...DC2260 will challenge you to widen your view of what computer science includes
  - ▶ lots of things besides programming are involved in the design of computer systems!
- ▶ *“if students don't know about user interfaces, they will not serve industry needs. It seems that only through CS does HCI research disseminate to products”* (Myers, 1998)

Myers, D.G., (1998), *Psychology* (5<sup>th</sup> Ed.), New York: Worth Publishers

# usability in more detail...

- ▶ **usability** is a quality attribute that measures how easy a user interface is to use
- ▶ **utility** is another important quality attribute which measures whether a system does what the user needs it to do
- ▶ usability and utility are equally important – one is no good without the other but don't confuse the two

# usability in more detail...

- ▶ ISO 9241-11 defines **usability** as: *“the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use”*
- ▶ 4 points underpinning **usability**:
  - ▶ need to focus on the users during design to end up with a usable system
  - ▶ users use systems to be productive
  - ▶ users are typically task-oriented, busy, and often multi-tasking
  - ▶ users (not developers) decide if a product is usable or easy to use

# usability in more detail...

- ▶ usability is generally accepted to be based on 5 components:
  - ▶ **learnability**: how easy is it for users to learn how to use the system? how easy is it for them to complete basic tasks the first time they use the system?
  - ▶ **efficiency**: once users are familiar with a system, how quickly can they accomplish their tasks?
  - ▶ **memorability**: when users return to the system after a period of not using it, how easily can they become proficient again?
  - ▶ **errors**: how many errors do users make? how severe are those errors? and how easily can they recover from the errors?
  - ▶ **satisfaction**: how pleasant is the system to use?
- ▶ aim for recognition rather than recall

# haven't we got usability sorted already?...

- ▶ in the early years of computer science, usability wasn't given much consideration...
- ▶ ...today, hard to sell computing products that are not usable!
- ▶ 3 principal factors have increased the emphasis placed on usability:
  - ▶ changes in hardware environments
  - ▶ diversification of users
  - ▶ diversification of applications





# changes in hardware environments...

- ▶ hardware has become much more powerful in terms of processing speed, storage capacity, and output characteristics
  - ▶ permitted development of more sophisticated UIs
- ▶ hardware that users interact with directly (peripherals) has become increasingly diverse
  - ▶ contributed to the development of different kinds of UI
- ▶ hardware has become increasingly accessible to a wider range of users due to decreasing costs
  - ▶ diversity of users are demanding increasing number of ways to interact with technology

# diversification of users...

- ▶ users have changed from being predominantly computing professionals to discretionary users
- ▶ novice users have sophisticated expectations about computing systems they use
  - ▶ expectation that they are easy to use and are robust
  - ▶ non-computer professional users are technologically savvy and expect sophisticated UIs with low error rates and fast response times

# diversification of applications...

- ▶ software use has moved from applications that support computation in the likes of banks, government organisations, and the military to personal computing
- ▶ when buying new computing technology, users now consider the software they want to run and select the hardware best suited to support it
- ▶ software applications that users interact *directly* with are now far more diverse than 40 years ago

usability engineering now has far more to consider – the concept of usability is much more complex now than it ever was before and so, far from being “sorted”, it is becoming increasingly difficult to achieve



# what is usability engineering?...

- ▶ usable interfaces don't just happen by accident – they are the result of **usability engineering**
- ▶ to build a good UI, you have to follow an organised development cycle
- ▶ **product** = content of the UI (based on human factors issues, design guidelines, and interaction styles)
- ▶ **process** = strategy for developing the product and involves methods, techniques, and tools for development and assessment
- ▶ HCI practitioners are concerned with both product + process

# why is usability engineering difficult?...

- ▶ UIs are hard to design because the detailed application knowledge required is often very hard to obtain
- ▶ **know your user because they are not you!**
  - ▶ really knowing users is time intensive and may require designers to change their views on issues
- ▶ good UI designs may have to deal with a number of conflicting requirements
- ▶ increasing need for UIs to guarantee accessibility to users with different skills, needs, and expectations

# why invest the effort?...

- ▶ *“There is substantial empirical evidence that attention to usability dramatically decreases costs and increases productivity” (Myers, 1994)*
- ▶ *“It’s estimated that 80% of maintenance costs are spent on unforeseen user requirements, while only 20% are due to bugs” (Chrusch, 2000)*



Myers, B.A., (1994), Challenges of HCI Design and Implementation, *Interactions*, 1(1), 73-83

Chrusch, M., (2000), The Whiteboard: Seven Great Myths of Usability, *Interactions*, 7(5), 13-16

# how do we guarantee usability?...

- ▶ no one formula works
- ▶ relying on designer's intuition is risky
- ▶ evaluation of UIs is critical
- ▶ 4 **myths** of usability:
  - ▶ good UIs are just appealing graphics ✖
  - ▶ usability is the result of common sense ✖
  - ▶ application of guidelines to UIs will lead to usability ✖
  - ▶ usability problems can be resolved with help/training/documents ✖
- ▶ **buy into these myths and you are in trouble!**

## Relevant Reading:

Leventhal, L., & Barnes, J., (2008), *Usability Engineering: Process, Products, and Examples*, Pearson-Prentice Hall:  
Chapters 2-3