University of Macau
Faculty of Science and Technology
Department of Computer and Information Science
CISB250 Human-Computer Interaction
Syllabus
1st Semester 2014/2015
Part A – Course Outline

Elective course in Computer Science

Course description:
(2-2) 3.0 credits. This junior compulsory major course introduces fundamentals of interaction design based on established learning from human-computer interactions (HCI) in relation to contextual design of interactive systems. HCI is an important area of computing knowledge, and the construction of useful and usable interfaces that ease the man-machine interaction has become required skills for all computer science students. Coverage includes: problem formulation, user requirements study, usability analysis, prototyping, and evaluation. Pedagogy includes a mixture of dialogic teaching, classroom discussions of cases, and group-based projects.

Course type:
Theoretical with substantial laboratory/practice content

Prerequisites:
- none

Textbook(s) and other required material:
- Rogers, Y., Sharp, H., and Preece, J. (2011) *Interaction Design beyond human-computer interaction*. 3rd ed., John Wiley & Sons Ltd.

References:
- Dix, A., Finlay J., Abowd, G. D., Beale, R. (2004). *Human-Computer Interaction.* 3rd ed., Prentice Hall.
- Hartson, R., Pyla, P. S. (2012). The UX Book – Process and guidelines for ensuring quality user experience, Morgan Kaufmann.

Major prerequisites by topic:
- None

Course objectives:
- Learn the fundamentals of interface design (establish requirements, user requirement analysis, prototyping, and evaluation) [a, c]
- Understand human factors in the design and during the design process [a, c]
- Practice the process of interaction design [a, c, d]
- Aware of the contemporary issues and technologies for interaction design [f, h]

Topics covered:
- **Basic Concepts (4 hours)**: Introduce the concepts of interaction design. Compare good and bad designs by examples. Discuss what to be involved to make good design: People, activities, context, and technology (PACT analysis), usability goals, and user experience goals.
- **Interaction Design Process (2 hours)**: Overview of the user-centered design and the process of interaction design. The four basic activities of the design process, i.e. requirement establishment, design alternatives, prototyping the design, and evaluation.
- **Establishing Requirements (4 hours)**: Identify requirements. Learn how to plan a data collection activity. Discuss different techniques including interviews, questionnaires, direct and indirect observations, and contextual inquiry, etc. Learn how to develop flow models, requirement statements using Volere shells, personas, scenarios, use cases, essential use cases, and to perform hierarchical task analysis (HTA) on task description.

- **Conceptualizing Interactions (4 hours)**: Discuss about the mental model and conceptual model, user interface metaphors, and core interaction types.
- **Cognitive Aspects (4 hours)**: Study the human factors in design especially the cognitive aspects, including attention, perception, memory, learning, problem solving. Introduce state-of-the-arts cognitive frameworks which are used to explain and predict user behavior, including mental models, gulfs of execution and evaluation, distributed cognition, and other external cognition.
- **Prototyping (4 hours)**: Explain prototyping and its goals. Discuss different types of prototyping activities such as storyboarding, sketching, wizard of Oz, and high-fidelity prototyping. Learn how to transform requirements into conceptual models and finally to generate the physical design. Discuss the use of scenarios in prototyping.
- **Evaluation (4 hours)**: Explain the concepts and terms used in evaluation. Introduce different evaluation methods including the DECIDE framework, usability testing, heuristic evaluation, and cognitive walkthroughs. Predictive models such as GOMS, KLM, Fitts' law will be addressed.
- **Interfaces (2 hours)**: Overview of the different kinds of interfaces, highlight the main design and research issues for each of the interfaces. Discuss the difference between graphical (GUIs) and natural user interfaces (NUIs). Learn by examples the suitability of an interface for a given application.

Class/laboratory schedule:

Timetabled work in hours per week			**No of teaching weeks**	**Total hours**	**Total credits**	**No/Duration of exam papers**
Lecture	**Tutorial**	**Practice**				
2	2	Nil	14	56	3	2 / 2+2 hours

Student study effort required:

Class contact:	
Lecture	28 hours
Tutorial	28 hours
Other study effort	
Self-study	21 hours
Homework assignment	6 hours
Project	15 hours
Total student study effort	98 hours

Student assessment:

Final assessment will be determined on the basis of:

Homework	10%	Project	30%
Mid-term	20%	Final exam	40%

Course assessment:

The assessment of course objectives will be determined on the basis of:

- Homework, project and exams
- Course evaluation

Course outline:

Weeks	Topic	Course work
1-2	**Introduction** Course introduction, evolution of interfaces, interaction design (ID), good and poor design examples, PACT analysis, usability and user experience goals	Project#T1
3	**The Process of Interaction Design** Interaction design lifecycle, design principles	Assignment#1
4-5	**Establishing Requirements** Data gathering techniques, data analysis and presentation, task description and analysis	Project#T2
6-7	**Conceptualizing Interactions** Mental model, conceptual model, interface metaphors, interaction types: instructing, conversing, manipulating, exploring	Project#T3
8-9	**Cognitive Aspects** Human cognition processes (attention, perception, memory, learning, reading, speaking, listening, problem solving, planning, reasoning, decision making), cognitive frameworks (internal and external)	Assignment#2 Mid-term
10-11	**Design and Prototyping** Conceptual design, interface metaphor, using scenarios, sketching, storyboarding, low-fidelity prototypes, high-fidelity prototypes	Project#T4
12-13	**Evaluation** Usability testing, Nielsen's 10 heuristics, heuristic evaluation, cognitive walkthroughs, predictive models (GOMS, KLM, Fitts' law)	Assignment#3
14	**Interfaces and Future of HCI** Different kinds of interfaces: command-based, WIMP/GUI, multimedia, virtual reality, web, mobile, sharable, wearable, tangible, and more	Project#T5

Contribution of course to meet the professional component:
This course prepares students with fundamental knowledge and experiences to design interactive systems.

Relationship to CS program objectives and outcomes:
This course primarily contributes to the Computer Science program outcomes that develop student abilities to:

(a) An ability to apply knowledge of computing and mathematics appropriate to the programme outcomes and to the discipline;
(c) An ability to analyse a problem, and identify and define the computing requirements appropriate to its solution;
(d) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs with appropriate consideration for public health and safety, social and environmental considerations.

The course secondarily contributes to the Computer Science program outcomes that develop student abilities to:
(f) An understanding of professional, ethical, legal, security and social issues and responsibilities;
(h) An ability to analyse the local and global impact of computing on individuals, organisations, and society.

Relationship to CS program criteria:

Criterion	DS	PF	AL	AR	OS	NC	PL	HC	GV	IS	IM	SP	SE	CN
Scale: 1 (highest) to 4 (lowest)								1				3	2	

Discrete Structures (DS), Programming Fundamentals (PF), Algorithms and Complexity (AL), Architecture and Organization (AR), Operating Systems (OS), Net-Centric Computing (NC), Programming Languages (PL), Human-Computer Interaction (HC), Graphics and Visual Computing (GV), Intelligent Systems (IS), Information Management (IM), Social and Professional Issues (SP), Software Engineering (SE), Computational Science (CN).

Course content distribution:

Percentage content for			
Mathematics	Science and engineering subjects	Complementary electives	**Total**
0%	100%	0%	100%

Coordinator:
Prof. Z. G. Gong

Persons who prepared this description:
Dr. Weng In Siu

Part B – General Course Information and Policies

1st Semester 2014/2015

Instructor:	Dr. Weng In Siu	Office:	E11-4025
Office hour:	Tue 2-3:45 pm & Fri 5:30-7:15pm	Phone:	8822 4452
Email:	shirleysiu@umac.mo		

Time/Venue: **Lecture** Tue 2:00pm - 3:45 pm E12-G021
Tutorial Fri 5:30pm - 7:15 pm E11-1026

Grading distribution:

Percentage Grade	Final Grade	Percentage Grade	Final Grade
100 - 93	A	92 - 88	A–
87 - 83	B+	82 - 78	B
77 - 73	B–	72 - 68	C+
67 - 63	C	62 - 58	C–
57 - 53	D+	52 - 50	D
below 50	F		

Comment:
The objectives of the lectures are to explain and to supplement the text material. Students are responsible for the assigned material whether or not it is covered in the lecture. Students who wish to succeed in this course should read the textbook prior to the lecture and should work all homework and project assignments. They are encouraged to look at other sources (other texts, etc.) to complement the lectures and text.

Homework policy:
Assignments are lightweight exercises to help students to revise and practice the lecture content.

- There will be 3 homework assignments.
- Homework is due one week after assignment unless otherwise noted.
- The course grade will be based on the average of the homework grades.

Course project:
Through this group project, students will investigate, design, and prototype an interactive computer system. Project requirements will be announced in the class. Grading is based on the quality and the efforts of the work, as well as the final presentation. The entire project is consisted of five subtasks: task 1 to choose the target application system, task 2 to collect user requirements, task 3 to perform requirement analysis, task 4 to design, prototype and evaluate, task 5 to present the work.

Exam:
There will be one 2-hour mid-term exam during the semester and one 2-hour final exam. Both exams are closed book.

Note:

- Check UMMoodle (https://ummoodle.umac.mo/) for announcements, homeworks and lectures.
- No make-up exam is given except for CLEAR medical proof.
- Cheating is absolutely prohibited by the university.

Appendix - Measurement Dimensions and Rubric for Program Outcomes (a), (c), (d), (f), and (h)

(a) An ability to apply knowledge of computing and mathematics appropriate to the programme outcomes and to the discipline

Measurement Dimension	Excellent (80-100%)	Average (60-79%)	Poor (<60%)
1. An ability to apply knowledge of computing to the solution of complex computing problems.	Students understand the computing principles, and their limitations in the respective applications. Use the computing principles to formulate and solve complex computing problems.	Students understand the computing principles, and their limitations in the respective applications. But they have trouble in applying these computing principles to formulate and solve complex computing problems.	Students do not understand the computing principles, and their limitations in the respective applications. Do not know how to apply the appropriate computing principles to formulate and solve complex computing problems.
2. An ability to apply knowledge of mathematics to the solution of complex computing problems.	Students understand the mathematical principles, e.g., calculus, linear algebra, probability and statistics, relevant to computer science, and their limitations in the respective applications. Use mathematical principles to formulate and solve complex computing problems.	Students understand the theoretical background and know how to choose mathematical principles relevant to computer science. But they have trouble in applying these mathematical principles to formulate and solve complex computing problems.	Students do not understand the mathematical principles and do not know how to formulate and solve complex computing problems.

(c) An ability to analyse a problem, and identify and define the computing requirements appropriate to its solution

Measurement Dimension	Excellent (80-100%)	Average (60-79%)	Poor (<60%)
1. An ability to understand problem and identify the fundamental formulation	Students understand problem correctly and can identify the fundamental formulation	Student understand problem correctly, but have trouble in identifying the fundamental formulation	Students cannot understand problem correctly, and they do not know how to identify the fundamental formulation
2. An ability to choose and properly apply the correct techniques	Students know how to choose and properly apply the correct techniques to solve problem.	Students can choose correct techniques but have trouble in applying these techniques to solve problem.	Students have trouble in choosing the correct techniques to solve problem.

(d) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs with appropriate consideration for public health and safety, social and environmental considerations

Measurement Dimension	Excellent (80-100%)	Average (60-79%)	Poor (<60%)
1. An ability to	Students understand how to	Students understand how to	Students do not know how

design, implement, and evaluate a computer-based system, process, component, or program	properly design, implement and evaluate a computer-based system, process, component, or program	design and implement a computer-based system, process, component, or program, but have trouble in evaluating it.	to design, implement, and evaluate a computer-based system, process, component, or program
2. An ability to understand what needs to be designed and the realistic design constraints regarding public health and safety, social and environmental considerations.	Students understand the design goals and the realistic design constraints regarding public health and safety, social and environmental considerations.	Students understand the design goals; but they are not clear about the realistic design constraints regarding public health and safety, social and environmental considerations.	Students have trouble in understanding what needs to be designed and the realistic design constraints regarding public health and safety, social and environmental considerations.

(f) An understanding of professional, ethical, legal, security and social issues and responsibilities

Measurement Dimension	Excellent (80-100%)	Average (60-79%)	Poor (<60%)
1. An ability to understand how to critique and analyse design trade-offs and constraints with respect to safety, liability and integrity of data, and context of use.	Students understand the design trade-offs and constraints with respect to safety, liability and integrity of data, and context of use. They also know how to appropriately critique and analyze these tradeoffs.	Students understand the design trade-offs and constraints with respect to safety, liability and integrity of data, and context of use. But they have trouble in appropriately critiquing and analyzing these tradeoffs.	Students cannot understand the design trade-offs and constraints with respect to safety, liability and integrity of data, and context of use. They do not know how to critique and analyze these tradeoffs.

(h) An ability to analyse the local and global impact of computing on individuals, organisations, and society

Measurement Dimension	Excellent (80-100%)	Average (60-79%)	Poor (<60%)
1. An ability to analyse the local and global impact of computing on individuals and organizations	Students understand the local and global impact of computing on individuals and society, and can analyze such impact in terms of scope and depth.	Students understand the local and global impact of computing on individuals and society, but have trouble in analyzing such impact in terms of scope and depth.	Students cannot understand the local and global impact of computing on individuals and society