CSIS 672 – Human Computer Interaction
Course Syllabus – Fall 2006

Professor: Dr. Bill Manaris

Office: Room: 223 J.C. Long Building
        Phone: (95)3-8159
        E-mail: manaris@cs.cofc.edu
        Web: http://www.cs.cofc.edu/~manaris/

Office Hours: MTWR 11 am – noon, and by appointment.

Course Description: Introduction to human computer interaction and user interface development. Topics include definitions of Human-Computer Interaction, importance of good interfaces, psychological foundations, user-interface design examples, interaction models and dialog types for interfaces, user interface life-cycle, user-centered design and task-analysis, prototyping and the iterative design cycle, prototyping tools and environments, user interface implementation, and interface quality and methods of evaluation.

This course stresses the importance of good interfaces and the relationship of user interface design to human-computer interaction. It is intended to provide an adequate basis in software design and implementation for user interfaces. There will be content on both the issues and engineering process for user interface development.

Prerequisites: Each student must have completed CSCI 230 (Data Structures and Algorithms) or an equivalent or higher course, or have permission of the instructor. Minimally, each student should have strong background in software development, data structures, and algorithms; also strong background in a high-level programming language such as C/C++, Java, Python, or MS Visual Basic.


Additional materials will be made available via handouts and the class webpage at http://www.cs.cofc.edu/~manaris/ (follow the CSIS 672 link).

Learning Goals:
• To understand how HCI relates to other aspects of software engineering
• To understand basic human and machine factors that influence the development of interactive computing systems
• To gain basic skills and knowledge for user interface design
• To acquire skills in integrating HCI into the system development life-cycle (analysis, design, implementation, evaluation)
• To develop an appreciation for user-centered design
• To learn at least one development methodology and one toolkit for prototyping/implementing user interfaces
• To gain awareness of other tools and methods available
• To develop at least one user interface
• To gain additional experience with team work and collaborative development efforts
Grading: Scale: A: 90-100; B: 80-89; C: 70-79; D: 60-69; F: <60. The grades of B+/−, C+/−, and D+/− may be given at the professor's discretion.

Final Grade Computation: Assignments (4-6) 30%, Tests (2) 40%, Comprehensive Final Exam 20%, and Class Participation 10%.

Collaboration Policy: • You must do your assignments alone (or with your teammates, for group assignments). You are not allowed to discuss assignments and possible solutions with anyone other than the instructor, lab instructor, and department-assigned tutors. You are not allowed to look at someone else’s solution (including books and the Internet), or show your solution to someone else other than the instructor (unless otherwise specified). Any violation of the above rules is an honor offense. See The Honor System of the College of Charleston and the Student Code of Conduct (www.cofc.edu/student-life/handbook/), especially sections on Cheating, Plagiarism (pp. 10-11), and Computer Use (p. 13).
• On assignments you will be asked to identify the person(s) you received help from, if any.
• In-class exercises, when identified as collaborative, are excluded from the above.

Course Policies: • Tests:
  • Attendance at tests is mandatory. Students must complete tests with no discussion or sharing of information with other students.
  • Calculators, computers, cell phones, etc. may not be used during a test.

• Classroom:
  • You should turn off all electronic devices (e.g., cell-phones, pagers, etc.) during class.
  • You are expected to attend all classes. Regardless of actual attendance, you are responsible for announcements made in class, assignment due dates, etc.
  • You are expected to participate in class with questions and invited discussion. However, you should respect your classmates right to learn; see Student Handbook section on Classroom Code of Conduct (pp. 49-50).

• Assignments:
  • Programming assignment grades will be based on design and style as well as correctness of result.
  • Reading feedback is essential in learning. Upon return of graded work, you have one week to ask questions about your grade.
  • Assignments are to be submitted to the instructor by the date and time they are due. If the instructor is unavailable, they may be turned in at the CSCI department office (Long 216).
  • Do not submit programs with syntax errors. They are not eligible for credit.

• Late Policy:
  • You have four “late” days for the whole semester. You may use these days as you wish for assignment submission. If you use them up, no late assignments will be accepted.
  • If you submit everything on time (use no late days), 2.5 bonus points will be added to your course grade.