Department of Informatics

André van der Hoek, Department Chair
5019 Donald Bren Hall
949-824-2901
ichair@ics.uci.edu

Introduction

Our world runs on information, with more and more aspects of daily life having information technologies and digital systems at their core. Topics such as open-source software, virtual organizations, online political campaigns, digital television, social media, and computer games need to be understood and advanced from both a technical and human perspective simultaneously.

This is what Informatics does.

We seek to make a positive difference in how people live, work, and build in a digital world. To that end, we study interactions among information technologies and people, create innovative information technologies that serve the diverse needs of society, and educate our students to be leaders in these endeavors.

Values

Our work is shaped by four key values:

- **Creativity**. We create new technologies, new experiences, and new ways of understanding. We believe that information technology provides a rich platform for expression, from programming environments to digital media, and creative arts.
- **Engagement**. We focus on real-world concerns, with a strong empirical focus and a commitment to understanding and advancing technology in real life, around the world.
- **Interdisciplinarity**. We use knowledge and methods from multiple disciplines to study and improve the relationships among people, information, and technology from a holistic perspective.
- **Partnership**. We build relationships across campus and beyond, partnering with other schools and educational institutions; with corporations and technology providers; with civic agencies and nonprofits; and with consumers, advocates, and interest groups to locate novel and important contexts for conducting and applying our work.

These values help us deliver results that matter. Our research has, as just a few examples, resulted in technology that improves the early diagnosis of cerebral palsy in preterm babies; in apps that help kids with autism spectrum disorder live fuller lives; and in new tools that assist software developers in locating and fixing bugs — real results that make a difference every day.

Our values similarly define the nature of our teaching. Our students' experience is not confined to campus. Instead, they are constantly exposed to the real world, the issues at play, and the possibilities of information technology making a difference. For instance, students in our capstone design course have designed a customizable Analytics dashboard for Google; a new web portal for the Down Syndrome Foundation; an at-home energy saving recommender for Edison; a mobile application to capture statistical data related to clinical cases for the UC Irvine Medical Center; and a freelance game in which a mystical fish has to protect its aquatic environment.

- Business Information Management, B.S.
- Computer Game Science, B.S.
- Digital Information Systems, Minor
- Health Informatics, Minor
- Informatics, B.S.
- Informatics, M.S.
- Informatics, Minor
- Informatics, Ph.D.
- Master of Human Computer Interaction and Design
- Master of Software Engineering
- Software Engineering, B.S.
- Software Engineering, M.S.
- Software Engineering, Ph.D.

Faculty

Iftekhar Ahmed, Ph.D. Oregon State University, *Assistant Professor of Informatics* (data mining, software engineering, software testing and analysis, software maintenance, empirical studies)

Rebecca W. Black, Ph.D. University of Wisconsin-Madison, *Associate Professor of Informatics; Education* (digital media and learning, fan studies)

Geoffrey C. Bowker, Ph.D. University of Melbourne, *Chancellor's and Donald Bren Professor of Informatics; Visual Studies* (values in design, social studies of databases, science and technology studies)

Stacy Branham, Ph.D. Virginia Tech, *Assistant Professor of Informatics* (human-computer interaction, design, assistive technology, safe and brave space, well-being, disability, feminism, marginality, inclusion, interdependence)

Yunan Chen, Ph.D. Drexel University, *Associate Professor of Informatics; Program in Public Health* (medical informatics, human computer interaction)

Roderic N. Crooks, Ph.D. University of California, Los Angeles, *Assistant Professor of Informatics* (science and technology studies, education technology, critical data studies, data visualization, community archives)

Darren Denenberg, Ph.D. University of Maryland, *Lecturer of Informatics*

James P. Dourish, Ph.D. University College London, *Chancellor's Professor of Informatics* (human-computer interaction, computer-supported cooperative work)

Daniel Epstein, Ph.D. University of Washington, *Assistant Professor of Informatics; Computer Science* (human-computer interaction, personal informatics, ubiquitous computing, social computing, health informatics)

Joshua Garcia, Ph.D. University of Southern California, *Assistant Professor of Informatics* (software engineering, software security, software analysis and testing, software architecture, software maintenance)

Gillian Hayes, Ph.D. Georgia Institute of Technology, *Robert A. and Barbara L. Kleist Professor of Informatics; Education* (interactive and collaborative technology, human-computer interaction, computer-supported cooperative work, educational technology, ubiquitous computing)

James Jones, Ph.D. Georgia Institute of Technology, *Associate Professor of Informatics* (software engineering, software testing and analysis, debugging and fault localization, static and dynamic analysis, software visualization)

David G. Kay, J.D. Loyola Marymount University, *Professor Emeritus of Teaching of Informatics; Computer Science* (computer law, computer science education)

Alfred Kobsa, Ph.D. University of Vienna, *Professor Emeritus of Informatics; Computer Science* (user modeling, human-computer interaction, artificial intelligence, cognitive science, interdisciplinary computer science)

Cristina V. Lopes, Ph.D. Northeastern University, *Professor of Informatics* (programming languages, acoustic communications, operating systems, software engineering)

Sam Malek, Ph.D. University of Southern California, *Associate Professor of Informatics* (software engineering, software architecture, software security, software analysis and testing)

Gloria J. Mark, Ph.D. Columbia University, *Professor of Informatics* (computer-supported cooperative work, human-computer interaction)

Melissa Mazmanian, Ph.D. Massachusetts Institute of Technology, *Associate Professor of Informatics* (computer-mediated communication, organization studies, information and communication technologies in practice, social response to emerging technologies, work/non-work negotiations in the information age)

Bonnie A. Nardi, Ph.D. University of California, Irvine, *Professor Emeritus of Informatics* (computer-supported collaborative work, human-computer interaction, computer-mediated communication, user studies methods, activity theory, cultural responses to technology development)

Emily Navarro, Ph.D. University of California, Irvine, *Lecturer of Informatics*

Gary Olson, Ph.D. Stanford University, *Professor Emeritus of Informatics* (interactive and collaborative technology, human-computer interaction, computer-supported cooperative work)

Judith Olson, Ph.D. University of Michigan, *Professor Emeritus of Informatics; Paul Merage School of Business; Urban Planning and Public Policy* (interactive and collaborative technology, human-computer interaction, computer-supported cooperative work)

Richard Pattis, M.S. Stanford University, *Professor of Teaching of Computer Science; Informatics* (MicroWorlds for teaching programming, debugging, computational tools for non-computer scientists)

Kylie Peppler, Ph.D. University of California, Los Angeles, *Associate Professor of Informatics; Education* (learning sciences, design, maker culture, arts, game design, computer programming, wearables)

David F. Redmiles, Ph.D. University of Colorado Boulder, *Professor of Informatics* (computer-supported cooperative work, human computer interaction, software engineering, globally distributed development teams, user interfaces, software tools)

Debra J. Richardson, Ph.D. University of Massachusetts, *Professor Emeritus of Informatics* (software engineering, program testing, life-cycle validation, software environments)

Katie Salen Tekinba#, M.F.A. Rhode Island School of Design, *Professor of Informatics* (game design, connected learning design, human-computer interaction)

Kurt Squire, Ph.D. Indiana University, *Professor of Informatics; Education* (video game design, games for learning, mobile technologies, civic engagement, place-based learning)

Constance Steinkuehler, Ph.D. University of Wisconsin-Madison, *Professor of Informatics; Education* (video games for impact, game-mediated cognition and learning, online social interaction, video games and policy)

Theresa Tanenbaum, Ph.D. Simon Fraser University, *Assistant Professor of Informatics* (digital games and narrative, tangible and wearable interaction, maker and DIY culture, nonverbal communication and virtual worlds)

Richard N. Taylor, Ph.D. University of Colorado Boulder, *Professor Emeritus of Informatics* (software engineering, user interfaces, environments, team support)

William M. Tomlinson, Ph.D. Massachusetts Institute of Technology, *Professor of Informatics; Education* (environmental informatics, educational technology, computer graphics/visualization/digital arts)

André W. Van der Hoek, Ph.D. University of Colorado Boulder, *Professor of Informatics* (software engineering)

Kai Zheng, Ph.D. Carnegie Mellon University, *Associate Professor of Informatics* (health informatics, human factors and human-computer interaction, technology adoption and acceptance, outcomes and evaluation)

Hadar Ziv, Ph.D. University of California, Irvine, *Associate Professor of Teaching of Informatics* (software testing, requirements engineering, Bayesian modeling)

Affiliate Faculty

Jonathan Alexander, Ph.D. Louisiana State University, *Associate Dean, Division of Undergraduate Education and Campus Writing Coordinator and Chancellor's Professor of English; Culture and Theory; Education; Gender and Sexuality Studies; Informatics* (writing studies, sexuality studies, queer theory, new media studies)

John L. Crawford, Media Artist and Software Designer, *Professor of Dance; Informatics* (screendance, interactive media, telematic performance, motion capture, digital arts)

John Christopher Dobrian, Ph.D. University of California, San Diego, *Professor of Music; Informatics*

Shayan Doroudi, Ph.D. Carnegie Mellon University, *Assistant Professor of Education; Informatics* (learning analytics, learning sciences, educational technology)

Magda S. El Zarki, Ph.D. Columbia University, *Professor of Computer Science; Informatics* (telecommunications, networks, wireless communication, video transmission)

Vijay Gurbaxani, Ph.D. University of Rochester, *Taco Bell Chair in Information Technology Management and Professor of Paul Merage School of Business; Informatics* (economics of information systems management, impact of information technology on organization and market structure)

Jesse C. Jackson, M.A. University of Toronto, *Director of the Minor in Digital Arts and Associate Professor of Art; Informatics* (digital art, informatics, design, architecture)

Peter O. Krapp, Ph.D. University of California, Santa Barbara, *Professor of Film and Media Studies; English; European Languages and Studies; Informatics; Music; Visual Studies* (digital culture, media history, cultural memory)

Simon G. Penny, M.F.A. Sydney College of the Arts, *Professor of Art; Informatics* (informatics, robotic sculpture, interactive environments, electronic media)

Kavita S. Philip, Ph.D. Cornell University, *Professor of History; Informatics* (history of modern South Asia, science and technology, political ecology, critical theoretical studies of race, gender, colonialism, new media, and globalization)

Stephanie Reich, Ph.D. Vanderbilt University, *Associate Professor of Education; Informatics; Psychological Science* (child development, parenting, peer interactions, media, program evaluation)

Patricia Seed, Ph.D. University of Wisconsin-Madison, *Professor of History; Informatics* (mapping: history and design, game design, navigation)

Mark J. Warschauer, Ph.D. University of Hawaii at Manoa, *Professor of Education; Informatics* (language, literacy, technology, STEM)

Informatics Courses

IN4MATX 12. Barter to Bitcoin: Society, Technology and the Future of Money. 4 Units.
Digital money has captured the broad imagination of speculators, coders, regulators, criminals and the mass media. Course puts this change in context: how do we understand money as a social, political and technological phenomenon?.

Same as SOC SCI 11A.

(II and III).

IN4MATX 43. Introduction to Software Engineering. 4 Units.
Concepts, methods, and current practice of software engineering. Large-scale software production, software life cycle models, principles and techniques for each stage of development.

Prerequisite: I&C SCI 32 or I&C SCI 32A

Overlaps with I&C SCI 105.

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX H81. Ethics, Technology, and Design. 4 Units.
Provides a critical framework for how and why biases of many kinds are built into everyday digital tools. Reflections on ethics, technology, and design through case studies drawn from machine learning, CS education, engineering, social media, and criminal justice.

Restriction: Campuswide Honors Collegium students only.

(III)

IN4MATX 101. Concepts in Programming Languages I. 4 Units.
In-depth study of several contemporary programming languages stressing variety in data structures, operations, notation, and control. Examination of different programming paradigms, such as logic programming, functional programming and object-oriented programming; implementation strategies, programming environments, and programming style.

Prerequisite: (I&C SCI 51 or CSE 31 or EECS 31) and (I&C SCI 46 or CSE 46). I&C SCI 51 with a grade of C or better. CSE 31 with a grade of C or better. EECS 31 with a grade of C or better. I&C SCI 46 with a grade of C or better. CSE 46 with a grade of C or better

Same as COMPSCI 141.

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 102. Concepts of Programming Language II. 4 Units.
In-depth study of major programming paradigms: imperative, functional, declarative, object-oriented, and aspect-oriented. Understanding the role of programming languages in software development and the suitability of languages in context. Domain-specific languages. Designing new languages for better software development support.

Prerequisite: IN4MATX 101 or COMPSCI 141 or CSE 141

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 113. Requirements Analysis and Engineering. 4 Units.
Equips students to develop techniques of software-intensive systems through successful requirements analysis techniques and requirements engineering. Students learn systematic process of developing requirements through cooperative problem analysis, representation, and validation.

Prerequisite: (I&C SCI 33 or CSE 43) and IN4MATX 43. I&C SCI 33 with a grade of C or better. CSE 43 with a grade of C or better. IN4MATX 43 with a grade of C or better

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 115. Software Testing, Analysis, and Quality Assurance. 4 Units.
Preparation for developing high-quality software through successful veri#cation and validation techniques. Fundamental principles of software testing, implementing software testing practices, ensuring the thoroughness of testing to gain con#dence in the correctness of the software.

Prerequisite: (I&C SCI 45J or I&C SCI 45C or I&C SCI 46 or CSE 46) and IN4MATX 43. I&C SCI 45J with a grade of C or better. I&C SCI 45C with a grade of C or better. I&C SCI 46 with a grade of C or better. CSE 46 with a grade of C or better. IN4MATX 43 with a grade of C or better

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 117. Project in Software System Design. 4 Units.
Specification, design, construction, testing, and documentation of a complete software system. Special emphasis on the need for and use of teamwork, careful planning, and other techniques for working with large systems.

Prerequisite: (IN4MATX 43 and I&C SCI 33) or CSE 43

Restriction: Upper-division students only. School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 121. Software Design: Applications. 4 Units.
Introduction to application design: designing the overall functionality of a software application. Topics include general design theory, software design theory, and software architecture. Includes practice in designing and case studies of existing designs.

Prerequisite: I&C SCI 33 or CSE 43. I&C SCI 33 with a grade of C or better. CSE 43 with a grade of C or better

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 122. Software Design: Structure and Implementation. 4 Units.
Introduction to implementation design: designing the internals of a software application. Topics include design aesthetics, design implementation, design recovery, design patterns, and component reuse. Includes practice in designing and case studies of existing designs.

Prerequisite: (I&C SCI 45J or I&C SCI 46 or IN4MATX 45) and (IN4MATX 101 or COMPSCI 141 or CSE 141)

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 124. Internet Applications Engineering. 4 Units.
Concepts in Internet applications engineering with emphasis on the Web. Peer-to-Peer and Interoperability. Topics include HTTP and REST, Remote Procedure/Method Calls, Web Services, data representations, content distribution networks, identity management, relevant W3C/IETF standards, and relevant new large-scale computing styles.

Prerequisite: (COMPSCI 132 or EECS 148) and I&C SCI 45J

Same as COMPSCI 137.
Overlaps with COMPSCI 122B.

Restriction: Upper-division students only. School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 125. Computer Game Development. 4 Units.
Introduction to the principles of interactive 2D and 3D computer game development. Concepts in computer graphics, algorithms, software engineering, art and graphics, music and sound, story analysis, and artificial intelligence are presented and are the basis for student work.

Prerequisite: COMPSCI 112 or COMPSCI 171 or IN4MATX 121 or ART 106B or I&C SCI 163 or I&C SCI 166

Same as COMPSCI 113.

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 131. Human Computer Interaction. 4 Units.
Basic principles of human-computer interaction (HCI). Introduces students to user interface design techniques, design guidelines, and usability testing. Students gain the ability to design and evaluate user interfaces and become familiar with some of the outstanding research problems in HCI.

Prerequisite: I&C SCI 10 or I&C SCI 31 or I&C SCI 32A or CSE 41 or ENGR 10 or ENGRMAE 10 or EECS 10. I&C SCI 10 with a grade of C or better. I&C SCI 31 with a grade of C or better. I&C SCI 32A with a grade of C or better. CSE 41 with a grade of C or better. ENGR 10 with a grade of C or better. ENGRMAE 10 with a grade of C or better. EECS 10 with a grade of C or better

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 132. Project in Human-Computer Interaction Requirements and Evaluation. 4 Units.
Students undertake significant projects in the elicitation and specification of HCI requirements and the thorough evaluation of user interfaces.

Prerequisite: IN4MATX 131

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 133. User Interaction Software. 4 Units.
Introduction to human-computer interaction programming. Emphasis on current tools, standards, methodologies for implementing effective interaction designs. Widget toolkits, Web interface programming, geo-spatial and map interfaces, mobile phone interfaces.

Prerequisite: I&C SCI 45J or I&C SCI 45C. I&C SCI 45J with a grade of C or better. I&C SCI 45C with a grade of C or better

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 134. Project in User Interaction Software. 4 Units.
Students complete an end-to-end user interface programming project based on an iterative design paradigm. Topics may include requirements brainstorming, paper prototyping, iterative development, cognitive walk-through, quantitative evaluation, and acceptance testing. Materials fee.

Prerequisite: IN4MATX 131 and IN4MATX 133

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Computer Science Engineering Majors have first consideration for enrollment.

IN4MATX 141. Information Retrieval. 4 Units.
An introduction to information retrieval including indexing, retrieval, classifying, and clustering text and multimedia documents.

Prerequisite: (I&C SCI 45C or I&C SCI 45J) and (STATS 7 or STATS 67). I&C SCI 45C with a grade of C or better. I&C SCI 45J with a grade of C or better

Same as COMPSCI 121.

Restriction: School of Info & Computer Sci students have first consideration for enrollment.

IN4MATX 143. Information Visualization. 4 Units.
Introduction to interactive visual interfaces for large datasets, and to principles of human visual perception and human computer interaction that inform their design. Various applications for data analysis and monitoring are discussed.

Prerequisite: (IN4MATX 131 or IN4MATX 43) and (I&C SCI 31 or I&C SCI 32A or CSE 41). IN4MATX 131 with a grade of C or better. IN4MATX 43 with a grade of C or better. I&C SCI 31 with a grade of C or better. I&C SCI 32A with a grade of C or better. CSE 41 with a grade of C or better

Restriction: School of Info & Computer Sci students have first consideration for enrollment.

IN4MATX 148. Project in Ubiquitous Computing. 4 Units.
Introduction to ubiquitous computing research methods, tools, and techniques. Prototyping, design, and evaluation of physical computing applications, smart environments, embedded systems, and future computing scenarios. Includes hands-on in-class laboratory exercises. Materials fee.

Prerequisite: I&C SCI 10 or I&C SCI 31 or CSE 41 or I&C SCI 32A. I&C SCI 10 with a grade of C or better. I&C SCI 31 with a grade of C or better. CSE 41 with a grade of C or better. I&C SCI 32A with a grade of C or better

Restriction: Upper-division students only. School of Info & Computer Sci students have first consideration for enrollment.

IN4MATX 151. Project Management. 4 Units.
Introduces theoretical and practical aspects of project management. Topics include organizational theory, group behavior, project management skills, case studies, personal and group productivity tools, management of distributed work, stakeholders, consultants, and knowledge management. Students do a project exercise.

Prerequisite: IN4MATX 43

Restriction: School of Info & Computer Sci students have first consideration for enrollment.

IN4MATX 153. Computer Supported Cooperative Work. 4 Units.
Introduces concepts and principles of collaborative systems. Topics may include shared workspaces, group interaction, workflow, architectures, interaction between social and technical features of group work, and examples of collaborative systems used in real-world settings. Students develop a simple collaborative application.

Prerequisite: (IN4MATX 161 or IN4MATX 43) and (I&C SCI 31 or I&C SCI 32A or CSE 41). I&C SCI 31 with a grade of C or better. I&C SCI 32A with a grade of C or better. CSE 41 with a grade of C or better

Restriction: School of Info & Computer Sci students have first consideration for enrollment.

IN4MATX 161. Social Analysis of Computing. 4 Units.
Introduction of computing as a social process. Examines the social opportunities and problems raised by new information technologies, and the consequences of different ways of organizing. Topics include computing and work life, privacy, virtual communities, productivity paradox, systems risks.

Prerequisite: I&C SCI 10 or I&C SCI 31 or I&C SCI 32A or CSE 41 or ENGR 10 or EECS 10 or ENGRMAE 10. I&C SCI 10 with a grade of C or better. I&C SCI 31 with a grade of C or better. I&C SCI 32A with a grade of C or better. CSE 41 with a grade of C or better. ENGR 10 with a grade of C or better. EECS 10 with a grade of C or better. ENGRMAE 10 with a grade of C or better. Satisfactory completion of the Lower-Division Writing requirement.

Restriction: School of Info & Computer Sci students have first consideration for enrollment.

IN4MATX 162W. Organizational Information Systems. 4 Units.
Introduction to role of information systems in organizations, components and structure of organizational information systems, and techniques used in information systems analysis, design, and implementation.

Prerequisite: IN4MATX 161. Satisfactory completion of the Lower-Division Writing requirement.

(Ib)

IN4MATX 163. Project in the Social and Organizational Impacts of Computing . 4 Units.
Students undertake projects intended to gather and analyze data from situations in which computers are used, organize and conduct experiments intended to test hypotheses about impacts, and explore the application of concepts learned in previous courses.

Prerequisite: IN4MATX 162W

IN4MATX 164. Children's Learning and Media. 4 Units.
Examines how popular media may impact how young people learn, develop, and communicate by looking at research related to the impacts of a wide range of popular media including television, video games, digital environments, mobile devices, and other multimedia.

Same as EDUC 130.

Restriction: Education Sciences Majors only. Informatics Majors only. Informatics Minors only.

IN4MATX 171. Introduction to Medical Informatics. 4 Units.
Broad overview of medical informatics for students with varied backgrounds. Electronic medical records, online resources, mobile technologies, patient safety, and computational design. Legal, ethical, and public policy issues. Health systems management. Evaluation and fieldwork for health systems.

Prerequisite: WRITING 39C or WRITING 30 or WRITING 31 or HUMAN 1C or HUMAN H1C. WRITING 39C with a grade of C or better. WRITING 30 with a grade of C or better. WRITING 31 with a grade of C or better. HUMAN 1C with a grade of C or better. HUMAN H1C with a grade of C or better

Same as PUBHLTH 105.

Restriction: Upper-division students only. Satisfactory completion of the Lower-Division Writing requirement.

IN4MATX 172. Project in Health Informatics. 4 Units.
Students undertake significant quarter-long projects related to health informatics. Topics may include field evaluations of health care technologies, prototypes, iterative design, and system implementations.

Prerequisite: PUBHLTH 105 or IN4MATX 171

Same as PUBHLTH 106.

IN4MATX 190. Special Topics in Informatics. 4 Units.
Studies in selected areas of informatics. Topics addressed vary each quarter.

Prerequisite: Prerequisites vary.

Repeatability: Unlimited as topics vary.

IN4MATX 191A. Senior Design Project. 4 Units.
Group supervised project in which students analyze, specify, design, construct, evaluate, and adapt a significant information processing system. Topics include team management, professional ethics, and systems analysis.

Prerequisite: IN4MATX 113 and IN4MATX 121 and IN4MATX 131 and IN4MATX 151

Grading Option: In Progress (Letter Grade with P/NP).

Restriction: Seniors only. Software Engineering Majors have first consideration for enrollment. Informatics Majors have first consideration for enrollment.

IN4MATX 191B. Senior Design Project. 4 Units.
Group supervised project in which students analyze, specify, design, construct, evaluate, and adapt a significant information processing system. Topics include team management, professional ethics, and systems analysis.

Prerequisite: IN4MATX 191A. In Progress (IP) grade for IN4MATX 191A is also accepted.

Restriction: Seniors only.

IN4MATX H198. Honors Research. 4 Units.
Directed independent research in Informatics for honors students.

Prerequisite: Satisfactory completion of the Lower-Division Writing requirement.

Repeatability: May be repeated for credit unlimited times.

Restriction: Bren School of ICS Honors students only. Campuswide Honors Collegium students only.

IN4MATX 199. Individual Study. 2-5 Units.
Individual research or investigation under the direction of an individual faculty member.

Repeatability: May be repeated for credit unlimited times.

IN4MATX 201. Research Methodology for Informatics. 4 Units.
Introduction to strategies and idioms of research in Informatics. Includes examination of issues in scientific inquiry, qualitative and quantitative methods, and research design. Both classic texts and contemporary research literature are read and analyzed.

IN4MATX 203. Qualitative Research Methods in Information Systems. 4 Units.
Introduction to qualitative research methods used to study computerization and information systems, such as open-ended interviewing, participant observation, and ethnography. Studies of the methods in practice through examination of research literature.

Prerequisite: IN4MATX 261 or IN4MATX 251

IN4MATX 205. Quantitative Research Methods in Information Systems. 4 Units.
Quantitative research methods used to study computerization and information systems. Design of instruments, sampling, sample sizes, and data analysis. Validity and reliability. Longitudinal versus cross-sectional designs. Analysis of secondary data. Studies of the methods through examination of research literature.

Prerequisite: IN4MATX 251 or IN4MATX 261. Basic knowledge of elementary statistics is also required.

IN4MATX 207S. Doctoral Seminar on Research and Writing. 2 Units.
Doctoral seminar centered on original research and writing. Provides a chance for doctoral students at all levels to present original work, brainstorm ongoing issues, and learn to provide and receive critical feedback from peers.

Grading Option: Satisfactory/unsatisfactory only.

Repeatability: May be repeated for credit unlimited times.

IN4MATX 209S. Seminar in Informatics. 2 Units.
Current research and research trends in informatics. Forum for presentation and criticism by students of research work in progress.

Repeatability: Unlimited as topics vary.

IN4MATX 231. User Interface Design and Evaluation. 4 Units.
Introduction to the design and evaluation of user interfaces, with an emphasis on methodology. Cognitive principles, design life cycle, on-line and off-line prototyping techniques. Toolkits and architectures for interactive systems. Evaluation techniques, including heuristic and laboratory methods.

IN4MATX 232. Research in Human-Centered Computing. 4 Units.
Introduction to contemporary topics in human-computer interaction, including methods, technologies, design, and evaluation. Emerging application domains and their challenges to traditional research methods. Advanced architectures and technologies. Critical issues.

Prerequisite: Some familiarity with HCI principles.

IN4MATX 237. Usable Security and Privacy. 4 Units.
Introduces usability problems in security and privacy methods, tools, and software. Overviews prominent examples of both failures and successes in usable security and privacy. Surveys state-of- the-art techniques and evaluation methodologies.

Same as COMPSCI 204.
Overlaps with IN4MATX 231, COMPSCI 203.

Restriction: Informatics Majors have first consideration for enrollment. Computer Science Majors have first consideration for enrollment. Undergraduate degree in Compsci or Informatics is strongly recommended.

IN4MATX 241. Introduction to Ubiquitous Computing. 4 Units.
The "disappearing computer" paradigm. Differences to the desktop computing model: applications, interaction in augmented environments, security, alternate media, small operating systems, sensors, and embedded systems design. Evaluation by project work and class participation.

Same as COMPSCI 248A.

IN4MATX 244. Introduction to Embedded and Ubiquitous Systems. 4 Units.
Embedded and ubiquitous system technologies including processors, DSP, memory, and software. System interfacing basics; communication strategies; sensors and actuators, mobile and wireless technology. Using pre-designed hardware and software components. Design case studies in wireless, multimedia, and/or networking domains.

Prerequisite: I&C SCI 51 and COMPSCI 152 and COMPSCI 161 and (I&C SCI 6N or MATH 3A or MATH 6G or I&C SCI 6D). B.S. degree in Computer Science is also accepted.

Same as COMPSCI 244.

IN4MATX 251. Computer-Supported Cooperative Work. 4 Units.
The role of information systems in supporting work in groups and organizations. Examines various technologies designed to support communication, information sharing, and coordination. Focuses on behavioral and social aspects of designing and using group support technologies.

IN4MATX 261. Social Analysis of Computing. 4 Units.
The social and economic impacts of computing and information technologies on groups, organizations, and society. Topics include computerization and changes in the character of work, social control and privacy, electronic communities, and risks of safety-critical systems to people.

IN4MATX 263. Computerization, Work, and Organizations. 4 Units.
Selected topics in the influence of computerization and information systems in transforming work and organizations. Theories of organization and organizational change. Processes by which diverse information technologies influence changes in work and organizations over short and long time periods.

Prerequisite: IN4MATX 251 or IN4MATX 261

IN4MATX 265. Theories of Information Society . 4 Units.
Social and economic conceptions of information technology. Macrosocial and economic conditions that foster changes in information technologies. Social construction of information and computer technology in professional worlds. Theories of information technology and large-scale social change.

Prerequisite: IN4MATX 251 or IN4MATX 261

IN4MATX 267. Digital Media and Society. 4 Units.
Selected topics in the technological and social aspects of online interactions, and policy including online games, social media, electronic activism, e-commerce, and digital libraries. Media-theoretic approaches to digital technology. Architectures, infrastructure considerations, and their consequences.

Prerequisite: IN4MATX 251 or IN4MATX 261

IN4MATX 273. Information Technology in Global Sustainability. 4 Units.
Explores the relationship between recent developments in information technology and the global transition to sustainability. Topics include the role of IT systems in the provision of human needs and wants (e.g., smart grids, food systems, and other IT-enabled infrastructure).

Restriction: Graduate students only.

IN4MATX 280. Overview of Human-Computer Interaction and Design . 4 Units.
Introduction to human-computer interaction and user-centered design. The material is focused on laying the groundwork for understanding the history, importance, and methods of human-computer interaction and design.

IN4MATX 281. User Needs Analysis . 4 Units.
Understanding the user's context, needs, and preferences. Topics include interviews and observations, modeling the context, flow, culture, space and artifacts involved in an endeavor, ways of aggregating what is found, and presenting these findings to others.

Prerequisite: IN4MATX 280

IN4MATX 282. Design and Prototyping . 4 Units.
Introduction to user-centered design and prototyping. Focused on practical methods for interaction design. Topics include the nature of design and the challenges to creating and evaluating good designs, as well specific skills for designing interactive systems.

Prerequisite: IN4MATX 280

IN4MATX 283. User Experience Evaluation . 4 Units.
Evaluating prototypes and completed systems. Topics include comparative analysis, laboratory experiments, heuristic evaluation, cognitive walkthroughs, surveys, clickstreams, and help-desk.

Prerequisite: IN4MATX 280

IN4MATX 284. Advanced Design and Prototyping . 4 Units.
Develop and communicate interactive technology design prototypes. Moving concepts from brainstorming and paper prototypes to wireframe and limited functionality mock-ups.

Prerequisite: IN4MATX 282

IN4MATX 285. Interactive Technology Studio . 4 Units.
Technologies, languages, and skills required for creating prototypes to communicate interactive technology concepts. Topics include HTTP, CSS, CSS scripting, AJAX, Design Patterns, Javascript, Javascript libraries such as jQuery, SQL, MVC, and cloud architectures.

Prerequisite: IN4MATX 280

IN4MATX 286. Innovations in HCI and Design . 4 Units.
Recent social and technological developments in human-computer interaction and design. Topics will vary as the field progresses but include novel input techniques, novel platforms, and innovations in theory and methods of design.

Prerequisite: IN4MATX 280

IN4MATX 287. Capstone Project in HCI and Design . 4 Units.
Group project that reinforces all concepts learned in this program, including knowing where user experience work is most appropriate and essential, and executing the appropriate steps.

Prerequisite: IN4MATX 283 and IN4MATX 284

IN4MATX 288. Capstone Project and Portfolio . 4 Units.
Completion of capstone projects and development of portfolios. Ideation, critique, development, and critique.

Prerequisite: IN4MATX 287

IN4MATX 290. Research Seminar. 2 Units.
Forum for presentation and criticism by students of research work in progress. Presentation of problem areas and related work. Specific goals and progress of research.

Grading Option: Satisfactory/unsatisfactory only.

Repeatability: May be repeated for credit unlimited times.

Restriction: Graduate students only.

IN4MATX 291S. Literature Survey in Software Engineering. 2 Units.
Reading and analysis of relevant literature in Software Engineering under the direction of a faculty member.

Repeatability: May be repeated for credit unlimited times.

IN4MATX 295. Special Topics in Informatics. 4 Units.
Studies in selected areas of informatics. Topics addressed vary each quarter.

Repeatability: Unlimited as topics vary.

Restriction: Graduate students only.

IN4MATX 298. Thesis Supervision. 2-12 Units.
Individual research or investigation conducted in preparation for the M.S. thesis option or the dissertation requirements for the Ph.D. program.

Repeatability: May be repeated for credit unlimited times.

Restriction: Graduate students only.

IN4MATX 299. Individual Study. 1-12 Units.
Individual research or investigation under the direction of an individual faculty member.

Repeatability: May be repeated for credit unlimited times.

Restriction: Graduate students only.

Software Engineering Courses

SWE 211. Software Engineering. 4 Units.
Study of the concepts, methods, and tools for the analysis, design, construction, and measurement of complex software-intensive systems. Underlying principles emphasized. State-of-the-art software engineering and promising research areas covered, including project management.

SWE 212. Analysis of Programming Languages. 4 Units.
Concepts in modern programming languages, their interaction, and the relationship between programming languages and methods for large-scale, extensible software development. Empirical analysis of programming language usage.

Same as CS 253.

SWE 213. Requirements Engineering and Specification. 4 Units.
Rigorous techniques in requirements engineering - the requirements definition phase of software development - with a focus on modeling and specification. Topics include notations and models for requirements specification; and methods, tools, and processes for software requirements elicitation, representation, analysis.

Restriction: Graduate students only.

SWE 214. Program Analysis. 4 Units.
Covers principles and concepts of automated program analysis. Topics include program representations; intra-procedural data-flow analysis; call-graph construction; pointer analysis; alias analysis; inter-procedural analysis; inter-procedural finite distributive subset problems; inter-procedural distributive environment problems; symbolic execution; and program-analysis applications.

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Graduate students only.

SWE 215. Software Analysis and Testing. 4 Units.
Studies techniques for developing confidence in software from traditional testing schemes to integrated, multitechnique analytic approaches. Considers strengths and weaknesses and explores opportunities for synergistic technique application. Emphasis is on approaches integrated into the software process.

Restriction: Graduate students only.

SWE 221. Software Architecture. 4 Units.
Study of the concepts, representation techniques, development methods, and tools for architecture-centric software engineering. Topics include domain-specific software architectures, architectural styles, architecture description languages, software connectors, and dynamism in architectures.

Restriction: Graduate students only.

SWE 225. Information Retrieval, Filtering, and Classification. 4 Units.
Algorithms for the storage, retrieval, filtering, and classification of textual and multimedia data. The vector space model, Boolean and probabilistic queries, and relevance feedback. Latent semantic indexing; collaborative filtering; and relationship to machine learning methods.

Prerequisite: CS 161 and CS 171 and (ICS 6N or MATH 3A or MATH 6G)

Same as CS 221.

Restriction: Graduate students only.

SWE 233. Intelligent User Interfaces. 4 Units.
Explores example software systems and their underlying concepts that leverage computing to empower and augment human individuals in their activities. Topics span the fields of user interface design, human-computer interaction, software engineering, and cognitive computing.

Prerequisite: CS 171

Restriction: Graduate students only.

SWE 234. Human Aspects of Software Engineering. 4 Units.
Selected topics about the cooperative and human aspects of software engineering from the perspective that software engineering is inherently an endeavor involving human stakeholders. Topics may include but are not limited to collaboration, trust, emotion, language, gender, and software tools.

Restriction: School of Info & Computer Sci students have first consideration for enrollment. Graduate students only.

SWE 241P. Applied Data Structures and Algorithms. 2 Units.
Exploration of strategies to tackle computational problems whose solutions include well-known algorithms and data structures. Topics include sorting, searching, indexing, among others.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only.

SWE 242P. Network Programming. 2 Units.
Exploration of networking principles and concepts for the development of distributed software. Topics include programming against well-known network protocols, ports and sockets, and network APIs.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only.

SWE 243P. Database Programming . 2 Units.
Exploration of software development with substantial reliance on a database for storage and retrieval of data. Topics include relational databases, structured query language, relational database management systems, APIs and libraries for database programming, among others.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only.

SWE 244P. Concurrent Programming. 2 Units.
Exploration of concepts and mechanisms for the development of concurrent software. Topics include threads, locks, race conditions, and deadlocks, among others.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 245P. GUI Programming . 2 Units.
Exploration of interactive software with substantial graphical user interface elements. Topics include libraries and frameworks for GUI programming, layout design and alternatives, event-driven programming, among others.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 246P. Mobile Programming. 2 Units.
Exploration of contemporary libraries and frameworks for construction of mobile applications. Topics include emulators, mobile development standards and patterns, energy consumption issues, screen layout, among others.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 247P. Applied Information Retrieval. 2 Units.
Exploration of principles and concepts for textual information retrieval. Topics include tokenization, inverted indexes, scored retrieval, and precision and recall.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 248P. Neural Network Programming . 2 Units.
Exploration of the concepts, terminology, and processes for training and using deep neural networks for classification problems. Topics include tensors and tensor operations, gradient-based optimization, feature engineering and learning, and workflow of learning systems.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 249P. Applied Data Analytics. 2 Units.
Exploration of the concepts, terminology, and processes used in supervised and unsupervised learning for classification and clustering, from a software engineering perspective. Topics include K-Nearest Neighbors, Logistic Regression, Random Forests, Support Vector Machines, K-Means, and model assessment.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only.

SWE 250P. Web Programming. 2 Units.
Exploration of the concepts, terminology, and popular frameworks for developing full-stack web applications. Students develop simple applications using multiple development stacks, and deploy them on the cloud.

Repeatability: May be taken for credit 2 times.

Restriction: Master of Software Engineering Degree students only.

SWE 261P. Software Testing and Debugging. 4 Units.
Designed to teach students how to ensure high-quality software by means of testing, debugging, and other quality assurance activities. Students learn a combination of both theoretical and practical skills, including hands-on experience with modern tools and approaches.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 262P. Programming Styles. 4 Units.
Designed to teach students the various ways software can be decomposed and put back together. Students are exposed to a variety of different programming styles and composition mechanisms.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 263P. User Experience and Interaction. 4 Units.
Provides an introduction to the basic principles of human-computer interaction (HCI) and the pragmatic aspects of usability engineering. Topics include the fundamentals of interaction, user experience, design for usability, and evaluation of products for their usability.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 264P. Distributed Software Architecture. 4 Units.
Study of software system architectures and architectural styles for large-scale distributed applications, and contemporary technologies and standards for their construction. Topics include client-server, peer-to-peer, publish-subscribe, REST, cloud computing, content distribution networks, scalability, latency, caching, and security, among others.

Restriction: Master of Software Engineering Degree students only.

SWE 265P. Reverse Engineering and Modeling. 4 Units.
Introduces theories, concepts, representations, techniques, and case studies in understanding large-scale, complex software systems. Topics include static and dynamic modeling notations, manual and (semi-)automated reverse engineering techniques, APIs, patterns, and styles. A significant, hands-on project is included.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 266P. Software Security and Dependability . 4 Units.
Principles and concepts for the design and construction of secure software. Topics include common types of software security vulnerabilities, methods for detecting vulnerabilities, design and process methodologies to improve security of software, and techniques for assessing security properties of software.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 271P. Career and Entrepreneurship. 4 Units.
Teaches practical skills for spoken, written, and electronic communication in a range of business and technical contexts, including promoting project ideas and portfolio development. Students practice their skills in classroom presentations and written exercises.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 272P. Project Management. 4 Units.
Provides an introduction to project management in software engineering from several perspectives. Topics include team behavior; globally distributed work; resource estimation, scheduling, and budgeting. Students apply their knowledge in an ongoing class project.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 275P. Curricular Practical Training. 1 Unit.
Mandatory internship in which students individually work at an outside organization to gain experience with the challenges involved in the practice of software engineering.

Grading Option: Satisfactory/unsatisfactory only.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 276P. Capstone Project in Software Engineering. 4 Units.
Quarter-long software-intensive project focusing on the design and implementation of a novel software system. Students are expected to bring to bear the concepts acquired during the program.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 290. Research Seminar. 2 Units.
Forum for presentation and criticism by students of research work in progress. Presentation of problem areas and related work. Specific goals and progress of research.

Grading Option: Satisfactory/unsatisfactory only.

Repeatability: May be repeated for credit unlimited times.

Restriction: Graduate students only.

SWE 290P. Advisement. 0.3 Units.
Mandatory weekly meetings with a faculty advisor.

Grading Option: Satisfactory/unsatisfactory only.

Repeatability: May be taken for credit 3 times.

Restriction: Master of Software Engineering Degree students only. Graduate students only.

SWE 295. Special Topics in Software Engineering. 4 Units.
Studies in selected areas of software engineering. Topics addressed vary each quarter.

Repeatability: May be repeated for credit unlimited times.

Restriction: Graduate students only.

SWE 298. Thesis Supervision. 2-12 Units.
Individual research or investigation conducted in preparation for the M.S. thesis option or the dissertation requirements for the Ph.D. program.

Repeatability: May be repeated for credit unlimited times.

Restriction: Graduate students only.

SWE 299. Individual Study. 1-12 Units.
Individual research or investigation under the direction of an individual faculty member.

Repeatability: May be repeated for credit unlimited times.

Restriction: Graduate students only.