Course Overview

CEN 5035
Software Engineering

Stephen M. Thebaut, Ph.D.
University of Florida
Contact Info

- **Instructor**: Steve Thebaut, E314-A
  - Office Hours: Wed/Fri 9:15-10:15 or by appt.
  - e-mail: smt AT cise DOT ufl DOT edu
  - Phone: (352) 505-1564

- **TA**: TBD
Description

- An introductory survey of fundamental concepts and principles underlying current and emerging methods, tools, and techniques for the cost-effective engineering of high-quality software systems.
- NOT a “programming” course.
- Focuses on surveying some of the critical aspects of SE that may be less familiar to students of computer science. E.g.: 
identifying a development process appropriate to the circumstances,

― eliciting and documenting requirements,

― indentifying appropriate design techniques,

― employing effective verification and validation strategies (e.g., reviews & inspections, formal methods) throughout the software development lifecycle,

― software maintenance, and

― software project management.
Things this course is NOT about:
  – Furthering software engineering dogma (beliefs that are not to be doubted or questioned)
  – Software engineering evangelizing – the “one true way”

Many SE principles and methodological guidelines seem, on the surface, to be in conflict with one another. We will rationally evaluate these principles and guidelines in the context of the heterogeneous, multidimensional problem space in which SE resides today.
Prerequisites

- Familiarity with programming using a high-level language (C, C++, Java, etc.)
- Basic knowledge of algorithms, data structures (linear lists, etc.), and discrete math (symbolic logic)
- Taking CEN 5035 is generally *inadvisable* for students who have previously completed an undergraduate software engineering survey course... Please discuss with the instructor *before* the end of drop/add.
Policy for students who have already taken CEN 4072/6070 at UF

- Students who have already completed CEN 4072/6070, Software Testing & Verification, may take CEN 5035, but in modified form.
- Instead of being tested on "Intro to Proofs of Correctness" and some other related subject matter, students will be tested on additional content NOT covered in class.
- Please discuss with the instructor if you have any questions or concerns about this before the end of drop/add.
Meeting Times and Location (for on-campus students)

- Monday, Wednesday, Friday: 4th (10:40-11:30)
- Room: CSE 122
- All in-class lectures will be recorded by UF EDGE and posted shortly thereafter for viewing by registered students.
Course Web Site

www.cise.ufl.edu/class/cen5035/fa14.html

- Syllabus
- Lecture Notes
- Practice Exams
- Exam Schedule
- Exercises
- Reading assignments
- Announcements
- Contact Information

Recorded lectures and the course grade book are available to registered students via UF's “e-Learning in Sakai” at

https://lss.at.ufl.edu
Getting Help

- **E-Learning access assistance** – contact:
  E-Learning Support Services:
  learning-support At ufl DOT edu
  phone: 352-392-4357 option 3

- **EDGE registration assistance** – contact:
  UF EDGE, 352-392-9670
Getting Help (cont’d)

- *course content–related help:*

  Steve Thebaut:  smt  AT cise DOT ufl DOT edu  
  phone: 352-505-1564
Textbook and Outside Readings

- Software Engineering, 9th Ed., by Ian Sommerville, Addison-Wesley, 2011...
  - See “Readings” at website for assigned parts of Sommerville Chapters.
  - One or more copies will be placed on reserve in Marston Science Library.
  - Note that access to the 9th Edition is required.

- Recommended outside readings are also listed on-line.
Textbook and Outside Readings

- *SOFTWARE ENGINEERING, 9th Ed.*, by Ian Sommerville, Addison-Wesley, 2011...
  - See “Readings” at website for assigned parts of Sommerville Chapters.
  - One or more copies will be placed on reserve in Marston Science Library.
  - Note that access to the 9th Edition is required.

- Recommended outside readings are also listed on-line.
Lecture Topics

- **Course Overview and Introduction to SE:** professional software development, FAQs about SE, SE ethics, case studies
- **Software Processes:** process models (waterfall model, incremental development, reuse-oriented SE), process activities, coping with change (prototyping, incremental delivery, Boehm’s spiral model), the Rational Unified Process (RUP)
- **Agile Software Development:** agile methods, plan-driven development, XP, agile project management, scaling agile methods
Lecture Topics (cont’d)

● **Software Requirements Engineering:** functional vs. non-functional requirements, software requirements document, requirements specification, RE process, elicitation and analysis, validation, requirements management

● **Architectural Design:** architectural design decisions, views and patterns, application architectures

● **Design and Implementation:** OO design, design patterns, implementation issues, open source development
Lecture Topics (cont’d)

- **Software Testing**: development testing, test-driven development, release testing, user testing
- **Software Evolution**: evolution processes, program evolution dynamics, software maintenance, legacy system management
- **Formal Specification**: formal specification in the software process, sub-system interface (algebraic) specification, behavioral (model-based) specification
- **Introduction to Proofs of Correctness**: axiomatic verification, predicate transforms, functional verification
Lecture Topics (cont’d)

- **Distributed Software Engineering**: distributed systems issues, client-server computing, architectural patterns for distributed systems, software as a service

- **Aspect-Oriented Software Engineering**: separation of concerns, aspects, join points and pointcuts, V&V

- **Software Project Management**: management activities, project planning and scheduling, risk management

- **Process Improvement**: process and product quality, CMMI process improvement framework
Examinations and Grades

- Your course grade will be based SOLELY on four equally weighted 45-minute exams. (Please do NOT ask if there are additional things you can do to improve your grade.)

- The exams are designed to be comprehensive and challenging; students are not necessarily expected to complete all the exam problems.

- A histogram of numeric scores will be provided with solution notes for each exam; your course letter grade will be determined at the end of the semester.
Examinations and Grades (cont’d)

- Since exams vary in difficulty, the grading scale is not fixed in advance.
- Typical course grade distribution:
  - A: 10-20%
  - A-: 25-35%
  - B+: 25-35%
  - B: 10-20%
  - lower than a B: 5-15%.
Exam Schedule (tentative)

- Exam 1: September 22 (September 22-23 for non-Gainesville area EDGE students), covers topics 1-4 (i.e., through *Requirements Engineering*, Chapters 1-4).

- Exam 2: October 15 (October 15-16 for non-Gainesville area EDGE students), covers topics 5-8 (i.e., *Architectural Design* through *Software Evolution*, Chapters 6-9).
Exam Schedule (cont’d)

- Exam 3: November 10 (November 10-11 for non-Gainesville area EDGE students), covers topics 9-10 (i.e., Formal Specification and Intro to Proofs of Correctness, Chapter 27).

- Exam 4: December 8 (December 8-9 for non-Gainesville area EDGE students), covers topics 11-14 (i.e., Distributed Software Engineering through Process Improvement, Chapters 18, 21, 22, and 26).

Note: there is no (comprehensive) “final exam” for this course.
Exam Procedures for EDGE Students

- Proctors will be instructed to schedule a SINGLE EXAM TIME for all students at each site. If this is not possible for any reason, students must contact the instructor well in advance to discuss other arrangements.

- Proctors should return electronic copies of completed exams directly to the instructor via e-mail after administration.
Grading Errors

- General exam re-grade (fishing expedition) requests are NOT accepted.
- Suspected grading errors should be brought to my attention ASAP (but no later than two weeks after graded exams are made available for review) for appropriate consideration.
- Your original, UNALTERED, exam must be returned to me (together with a correction request form to be made available) by a specified deadline.
- Note that partial credit policies are not subject to debate.
Make-Up Exam Policy

● Students are expected to be available at scheduled exam times. Do **NOT** schedule conflicting elective activities (family gatherings, interview trips, vacations, etc.).

● If missing an exam is unavoidable (e.g., due to sickness, accident, or other reasons beyond your control), contact the instructor as far in advance as possible.

(cont’d)
Make-Up Exam Policy (cont’d)

- Make-up exams, when permitted, may be administered orally.
- If it is not practical to administer a make-up exam before the end of the term, a course grade of "I" (incomplete) may be assigned.
Exercises

● Optional, self-check exercises (together with solution notes) are available on the course website.
  – Pre- and Post-Condition Specification Exercises
  – Axiomatic Verification Exercises
  – Predicate Transforms Exercises
  – Functional Verification Exercises

† Not intended for students who have already completed CEN 4072/CEN 6070, Software Testing & Verification.
Class Attendance Policy (on-campus students)

- Students are expected to view all recorded lectures and are responsible for any recorded announcement made in class.
- On-campus students are NOT required to attend live lectures.
Academic Integrity

- You will be asked to sign the following statement on the exams in this course:

  *On my honor, I have neither given nor received unauthorized aid on this exam and I pledge not to divulge information regarding its contents to those who have not yet taken it.*

- Note that a copy will be retained of all graded exams returned to students for review.
Other Items

For info regarding:

— Accommodation for Students with Disabilities,
— UF Counseling Services,
— UF Software Use Policies, and
— Instructor background,

please see the course syllabus.
Questions?
Course Overview

CEN 5035
Software Engineering

Stephen M. Thebaut, Ph.D.
University of Florida