Administrivia & Organizationals CECS 343

The essence for making this course a successful one for all of us

Organizationals

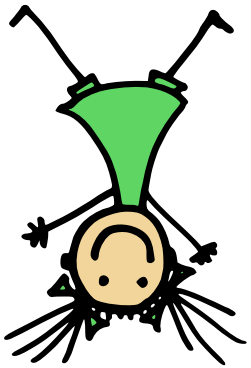
- Who?
- Where, when, what?
- Learning Goals
- Rules
- Assessment
- Assignments
- Grading
- Questions
- Exceptions
- Schedule
- Research opportunities
- What do you want out of this course?

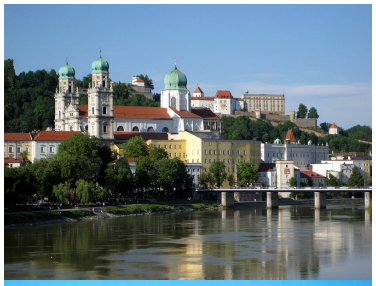
Who?

- You: 25 students
- Professor:
  - Me: Birgit Penzenstadler
    birgit.penzenstadler@csulb.edu
    office hours: Mon/Wed 4:30-5:15

Who am I?


Denmark
Baltic Sea
North Sea
Netherlands
Poland
Belgium
Germany
Czech Republic
Luxembourg
France
Switzerland
Austria
Liechtenstein
Hamburg
Schwerin
Bremen
Hannover
Berlin
Potsdam
Magdeburg
Essen
Düsseldorf
Cologne (Köln)
Bonn
Leipzig
Dresden
Weimar
Frankfurt
Wiesbaden
Mainz
Saarbrücken
Stuttgart
Munich
©1999 www.maps.com






ERDINGER
ERDINGER

When, where, what?

- Seminar: Mon/Wed 7:00-7:50pm in VEC-417
- Lab: Mon/Wed 8:00-9:15pm in ECS-413
- Material
  - Book: Roger S. Pressman
    Software Engineering, A practitioner's approach
    (7^{th} Edition, McGraw-Hill, 2010)
  - Slides
  - Exercises & discussions
  - Assignments

Learning Goals

A knowledge of and an ability to apply

- Software engineering and its place as an engineering discipline
- The principles of object orientation
- Developing clear, concise, and sufficiently formal requirements
- Use cases and user-centered design
- Applying design principles and patterns
- Making UML class diagrams which model aspects of the domain and the software architecture
- Creating UML sequence diagrams and state machines that correctly model system behavior
- Representing software behavior: Sequence diagrams, state machines, activity diagrams
- General software design principles: decomposition, decoupling, cohesion, reuse, reusability, portability, testability, flexibility
- Implementing a simple graphical user interfaces for a system
- Simple measurement techniques for software quality
- Reusable technologies as a basis for software engineering: frameworks and design patterns (singleton, observer, delegation, façade, adapter, observer, etc.)
- Demonstrate an appreciation for the breadth of software engineering
- Introduction to testing and project management

Rules

- You get out of this class what you put into it.
- Attend class & be actively involved
- Visit course website on BeachBoard
  & check e-mail regularly
- Silence mobile devices

Assessment

- 2 mid-term exams (each 20%)
- Final exam (20%)
- Project Use Cases 10%
- Behavior Specification 10%
- Design Specification 10%
- Implementation. 10%

Assignments

- Package properly
  - Submit as one file per assignment (PDF or zip)
  - With a cover page and description text
  - Listing the names and team
- Avoid inconsistencies
  - When feedback leads to changes, incorporate that into the older artifacts to keep consistent
- No handwriting, no deadline extensions

Grading

- I give grades with reasons and explanation, but sometimes I might also miss something
- Disagreements: If you believe a mistake has been made, prepare the following before meeting with me.
  - What is the mistake?
  - Why is it a mistake?
  - Support that demonstrates your arguments

Questions

- When in doubt … ask! Preferably during class. Why? There might be more with same questions.
- E-mail questions
  - Answer (generally) will be copied to everyone
  - Please put [CECS 343] at the beginning of subject line and include your full name signing the email message
- Questions will not be answered on the day before assignment is due

Exceptions

- Contact me as soon as possible,
  not at the last minute!
- Valid reasons
  - Serious illness, accident, family emergency, etc.
  - DOCUMENTED

Schedule (preliminary!)

CECS 343		Lecture	Deliverables
Week 1			
	21-Jan	Intro, Orga, Overview. 1 Software and Software Engineering	
Week 2	26-Jan	1 Software and Software Engineering	
	28-Jan	2 Process models	
Week 3	2-Feb	2 Process models	
	4-Feb	2 Process models	
Week 4	9-Feb	3 Agile development	
	11-Feb	3 Agile development	
Week 5	16-Feb	4 Principles that guide practice	
	18-Feb	Mid-term I	
Week 6	23-Feb	5 Understanding requirements	
	25-Feb	5 Understanding requirements	
Week 7	2-Mar	5 Understanding requirements	Use Cases due
	4-Mar	6 Requirements modeling: Scenarios, information, and analysis classes	
Week 8	9-Mar	6 Requirements modeling: Scenarios, information, and analysis classes	
	11-Mar	6 Requirements modeling: Scenarios, information, and analysis classes	
Week 9	16-Mar	7 Requirements modeling: flow, behavior, patterns and webapps	
	18-Mar	7 Requirements modeling: flow, behavior, patterns and webapps	Behavior Specs due
Week 10	23-Mar	Mid-term II	
	25-Mar	8 Design Concepts	
Spring break	30-Mar	Spring break	
	1-Apr	Spring break	
Week 11	6-Apr	8 Design Concepts	
	8-Apr	8 Design Concepts	
Week 12	13-Apr	9 Architectural Design	
	15-Apr	9 Architectural Design	
Week 13	20-Apr	10 Component-level Design	Design due
	22-Apr	10 Component-level Design	
Week 14	27-Apr	12 Pattern-based Design	
	29-Apr	12 Pattern-based Design	
Week 15	4-May	17 Testing	Implementation due
	6-May	29 Maintenance and Reengineering	
Finals	13-May	Final exam	

Dates for deliverables still to be specified – this is preliminary!

Research opportunities

- Would you like to do some research?
- Contact me for possible topics.
  - Sustainability in software engineering
  - Requirements engineering
  - Interview studies
  - Literature studies
  - Software development projects

What do you want out of this course?

- What are your expectations?
  - Teaching methods – how do you want me to teach?
  - Learning experience – how do you want to learn?
  - Acquired skills – what do you want to learn?
- Please take a piece of paper and write down 2-3 thoughts.
- I will do an early feedback evaluation in a few weeks to see whether we are on track.