CS 350
Software Development Principles

Dan Tappan

Department of Computer Science
and Electrical Engineering

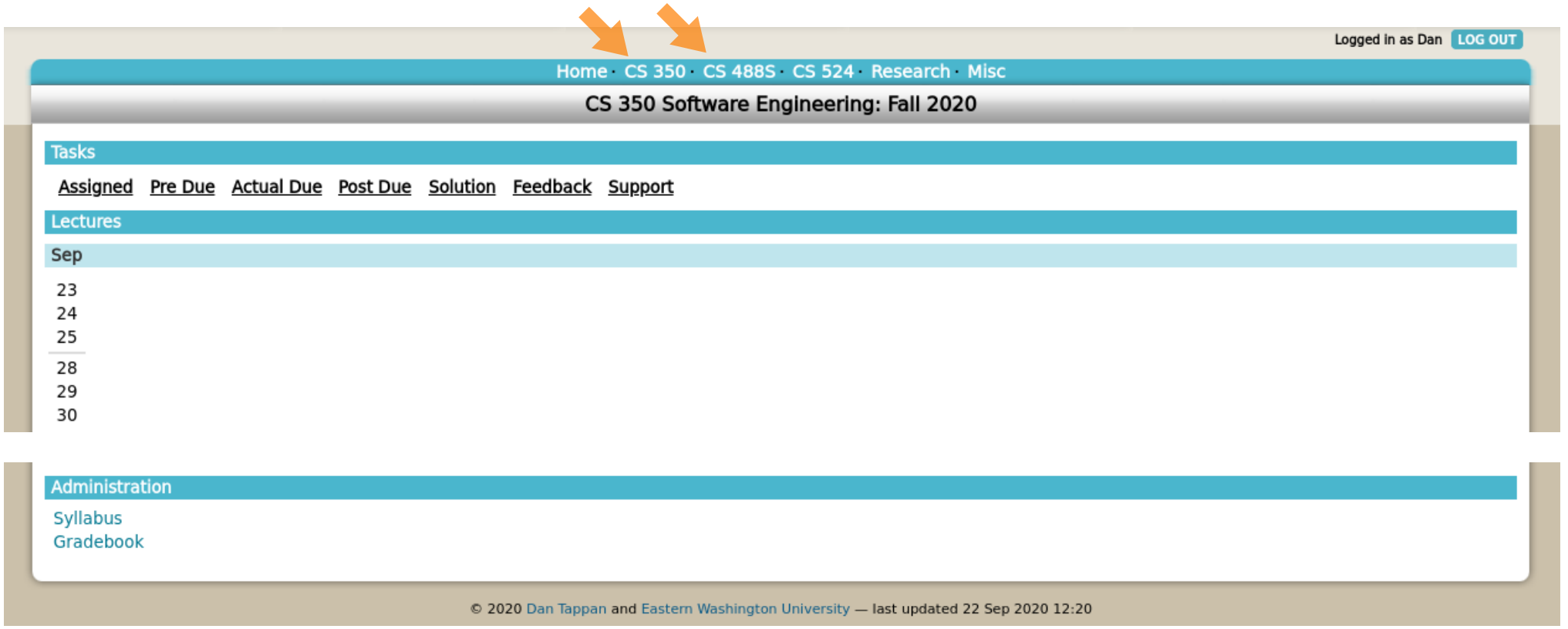
Fall 2020

Course Introduction

Logistics

- Online delivery
  - synchronous
  - 4 credit hours
  - 4 days lecture, 5 early on
  - 1 day lab later (how TBD)
  - office hours: 11-12 M-R
    - https://ewu.zoom.us/j/91996621895?pwd=Z0g5azcvUC9BVHEyT2dKYVROaldOQT09
- Resources
  - everything except lecture videos: shelby.ewu.edu
    - slides
    - audio
    - tasks (download and submit)
    - syllabus
      - read yourself; you're responsible for its contents
  - lectures: youtube.com or Panopto TBD

Website



From shelbyemailrelay@gmail.com
Subject **[CS 488S] Access information**
To Me

DTAPPAN@EWU.EDU

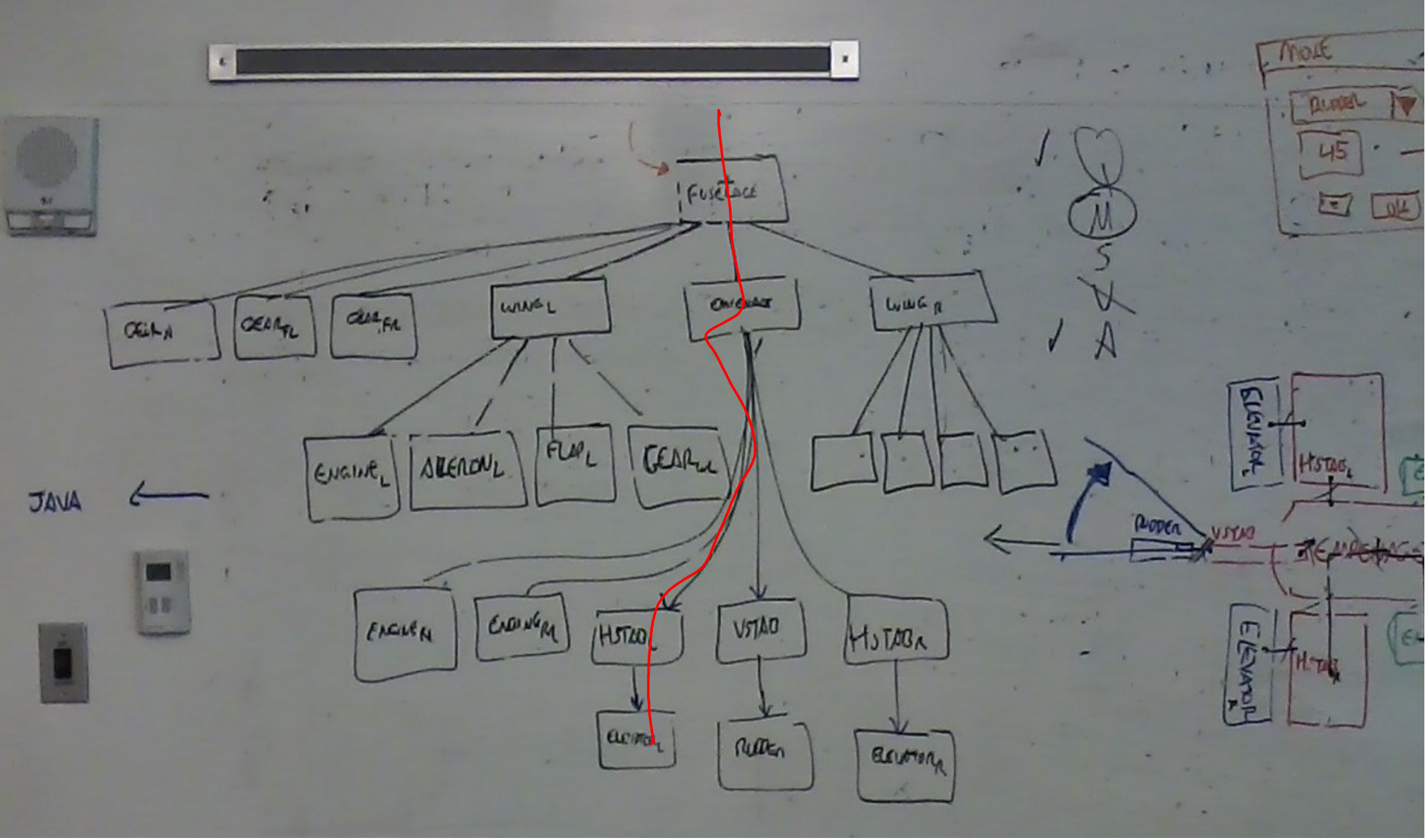
Dan,

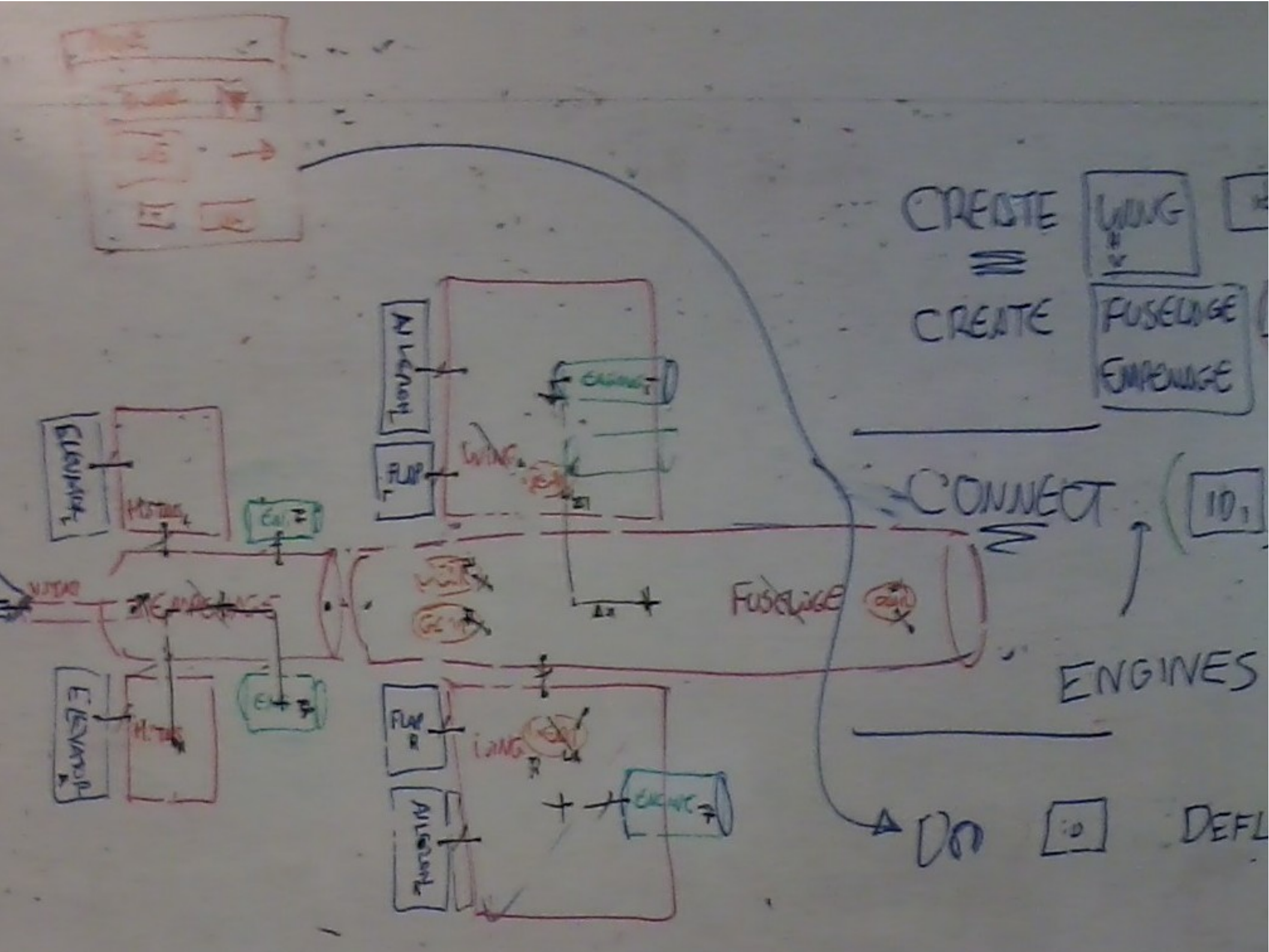
You have been granted access to the online resources for CS 488S Stu's Senior Capstone at http://shelby.ewu.edu/.

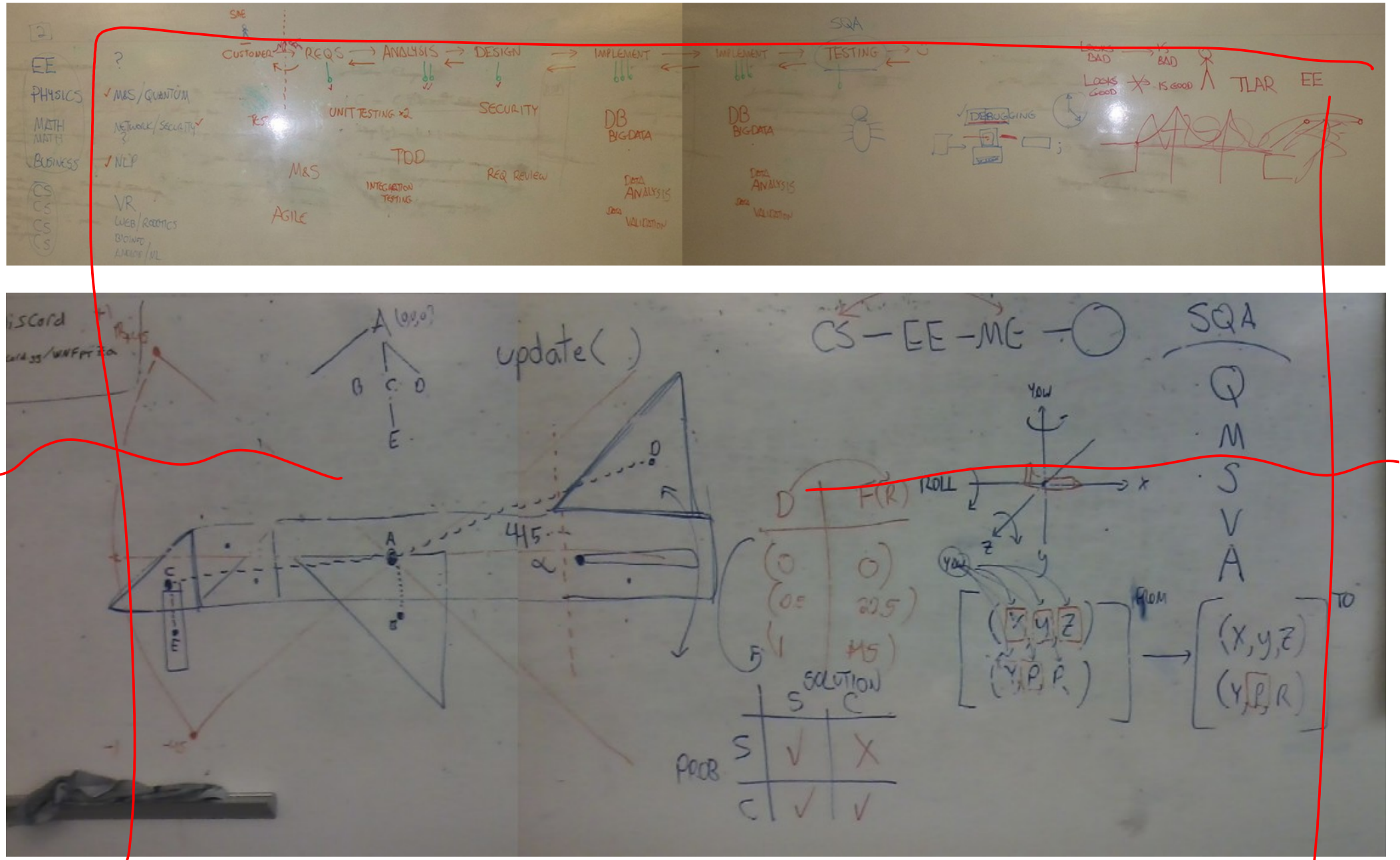
Your account information is:

Username: [redacted]
Password:

CAUTION This email originated from outside your organization. Please exercise caution when clicking any links or opening attachments.


FUSELAGE
WING L
WING R
ENGINE L
AILERON L
FLAP L
GEAR L
JAVA
HSTAB L
VSTAB
HSTAB R
RUDDER
ELEVATOR L
ELEVATOR R
45
M
S
V
A

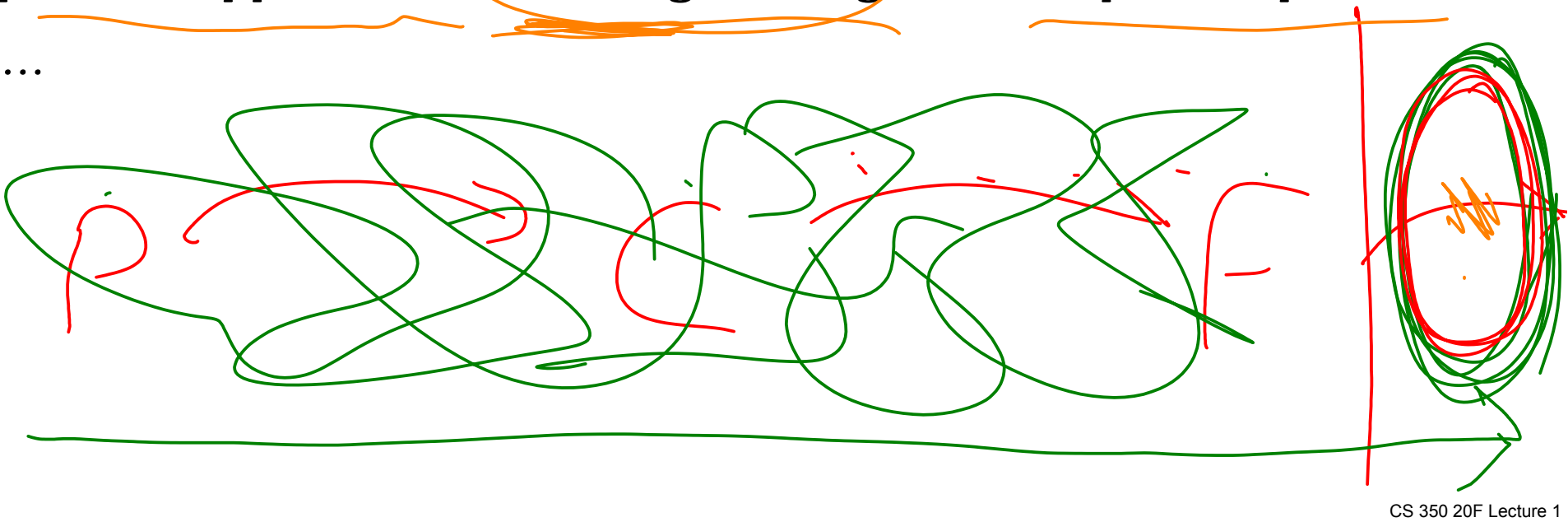

CREATE
CREATE
FUSELAGE
EMPENNAGE
CONNECT
ENGINES
FUSELAGE
DEFL


SAE
CUSTOMER
REQS
ANALYSIS
DESIGN
IMPLEMENT
TESTING
SQA
EE
PHYSICS
MATH
BUSINESS
CS
M&S / QUANTUM
NETWORK / SECURITY
NLP
VR
WEB / ROBOTICS
UNIT TESTING x2
SECURITY
DB
BIGDATA
TDD
INTEGRATION TESTING
REQ REVIEW
AGILE
DEBUGGING
LOOKS BAD
LOOKS GOOD
IS GOOD
TLAR
update( )
A (0,0,0)
B C D
E
45
CS - EE - ME
SQA
Q
M
S
V
A
ROLL
YAW
PITCH
F(R)
SOLUTION
S
C
POOR
TO


XP-PEN
AUTO
MENU
POWER

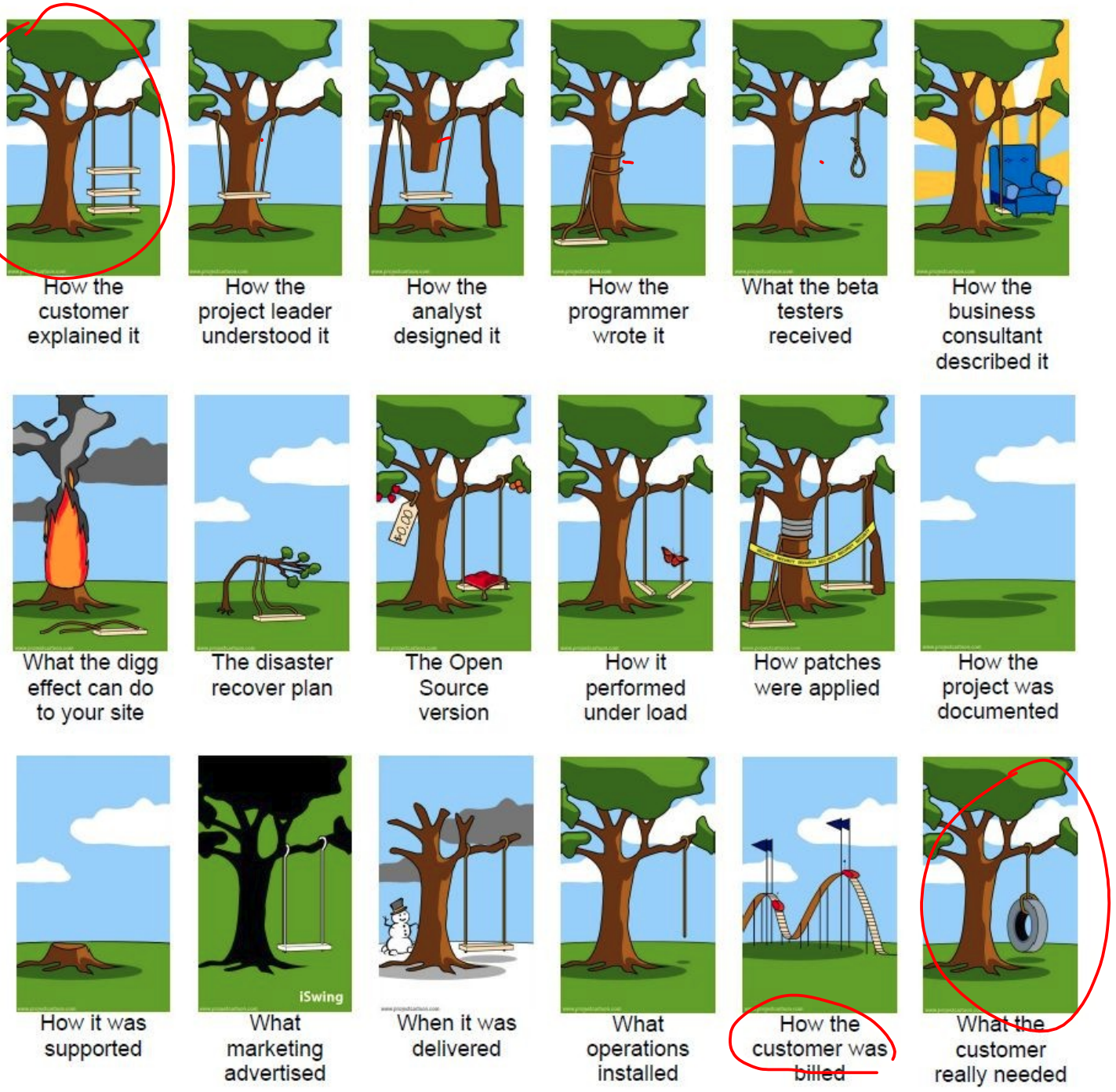
Objectives

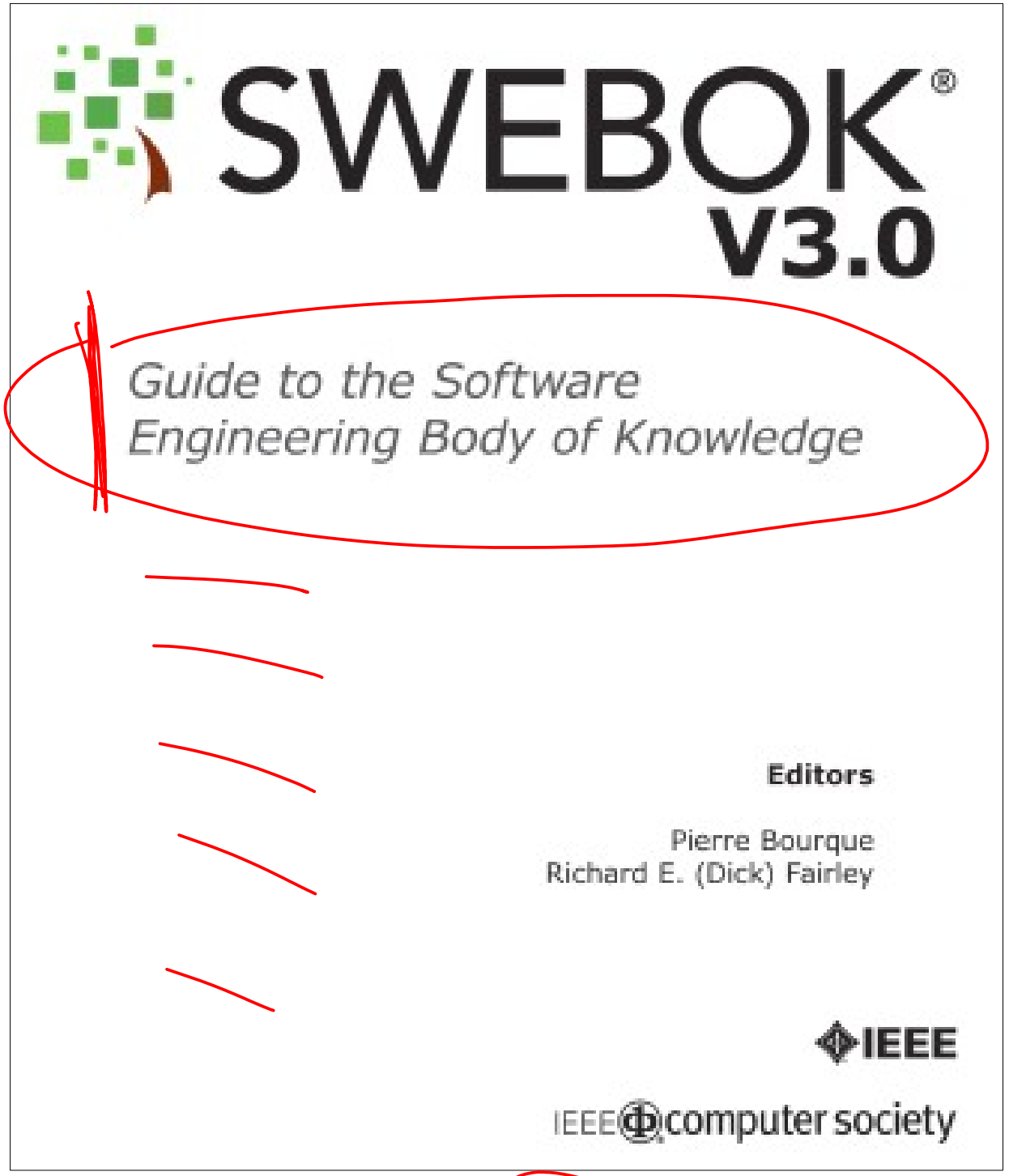
- To understand the fundamentals of software engineering
- To learn practical development tools and techniques
- To extend your skills from previous courses and prepare you for others
- To learn to think and act critically as software engineers
- To apply these elements in a realistic but reasonable context
- To function effectively on teams
- To improve your skills in written and verbal technical communication
- To promote appreciation of lifelong learning and workplace expectations
- and…



Objectives

- To mitigate:





on lecture link (335 pages)

Computer Science Curricula 2013

Curriculum Guidelines for Undergraduate Degree Programs in Computer Science

December 20, 2013

The Joint Task Force on Computing Curricula
Association for Computing Machinery (ACM)
IEEE Computer Society

on lecture link (518 pages)

IAS/Secure Software Engineering

[Elective]

Fundamentals of secure coding practices covered in other knowledge areas, including SDF and SE. For example, see SE/Software Construction; Software Verification and Validation.

Topics:

- Building security into the software development lifecycle (cross-reference SE/Software Processes)
- Secure design principles and patterns
- Secure software specifications and requirements
- Secure software development practices (cross-reference SE/Software Construction)
- Secure testing - the process of testing that security requirements are met (including static and dynamic analysis).
- Software quality assurance and benchmarking measurements

Learning outcomes:

1. Describe the requirements for integrating security into the software development lifecycle. [Familiarity]
2. Apply the concepts of the Design Principles for Protection Mechanisms, the Principles for Software Security [2], and the Principles for Secure Design [1] on a software development project. [Usage]
3. Develop specifications for a software development effort that fully specify functional requirements and identifies the expected execution paths. [Usage]
4. Describe software development best practices for minimizing vulnerabilities in programming code. [Familiarity]
5. Conduct a security verification and assessment (static and dynamic) of a software application. [Usage]

on lecture link (175+ pages)