Software Engineering

JD Kilgallin

jdk72@zips.uakron.edu

CPSC:480 Fall 2022

08/22/22

Course description

- Software development lifecycle as a business process with phases including requirements definition, analysis, design, implementation, testing & validation, release, maintenance, and evolution.

- Principles of software engineering and application of them in order to effectively prepare for a role in a software development organization.

- Methodologies and tools of software development.

Course Information

- Course number: CPSC:480
- Time: MW 5:15-6:30
- Location: CAS 134
- Instructor: JD Kilgallin (Jonathan Kilgallin)
- Website: github.com/kilgallin/SWEF22
- Recommended textbook: Software Engineering: A Practitioner's Approach, 9th edition, by Roger S. Pressman – bookstore or kindle.
- Recommended equipment: Bring a laptop to class
- Prerequisites: CPSC:210 Computer Science II; Basic knowledge of OOP

Topics I – Basics (August)

- Software engineering fundamentals
- History of software
- Software engineering terminology
- Software products and projects
- Software lifecycle
- Software engineering methodologies
- Software development team roles and organization
- Careers in software engineering

Topics II – Project Management (Sept)

- Tools for software engineering
- Version control
- Collaborative development
- Requirements engineering
- Software project planning
- Software product design
- Software systems modeling
- Formal specifications and verification
- *Midterm up to this point*

Topics III – Development Activities (Oct)

- Implementation considerations – security, privacy, performance, etc
- Coding style and standards
- Code reviews
- Program architecture
- Complexity and refactoring
- Bug and defect management
- Software testing
- Process automation
- Software builds, releases, and deployments

Topics IV – Business Considerations (Nov)

- Risk

- User experience

- Solution engineering

- Sales & marketing

- Research & data science

- System administration

- Engineering metrics

- Process improvements

Projects

Individual:

1. Account, résumé, and portfolio listing on GitHub.

Groups of ~5:

1. Project design and plan
2. Code review, bugs, and testing
3. Complete product iteration and report

Groups are subject to change at instructor's discretion. All group members are expected to participate in each project. Anonymous contribution surveys will be conducted as part of group project grading and members of a group may not all receive the same grade.

In-class Exercises

- Navigating Git and GitHub
- Planning development of a product feature
- Designing & modelling a product feature
- Analyzing architecture and concepts in source code
- Reviewing source code
- Implementing automated tests of a product feature

Grading

- 40% 1 individual, 3 team projects.
- 10% 6 in-class exercises (lowest score dropped).
- 10% In-class quizzes, presentations, and other assessments.
- 20% Midterm, Wednesday, Oct 5. Office hours on Monday that week.
- 20% Final, Wednesday, Dec 7, 5:15-7:15, CAS 134 (regular classroom).
  - Cumulative, focused on topics since midterm.
- Students who identify and correct significant errors in course content or suggest significant improvements, or routinely engage meaningfully in class discussions, can receive up to 1% extra credit in final grade for participation. Status of participation bonus will be recorded about once a month.
- Other extra credit opportunities will be available to all students.

Grade Cutoffs

- A 90.0%, A- 88.5%
- B+ 87.0%, B 80.0%, B- 78.5%
- C+ 77.0, C 70.0%,
- C- 65%, D+ 60%, D 55%, D- 50%,
- (i.e. 90/80/70 letter cutoffs with following round-ups):
  - Within 1.5% = letter-minus. Within the next 1.5% = next-letter-plus
  - Below a 70% C, cutoffs in 5% increments down to D- = 50%
- Grades will be rounded up to nearest 0.1%.
- No individual extra credit opportunities or additional rounding.

Course Policies

- It is up to the student to make up any missed lecture material. Late assignments require a valid excuse. Make-ups of any graded work for this class are given only with an excused absence or a documented emergency. I encourage you to contact me if an emergency arises.

- Students whose names are not on the University's official class list are not permitted to attend class.

- Any source code or other artifact created for this course is committed to a GitHub repository. For credit, the source code must appear in the proper repository. All submitted work must be your own. Submission of work that is even partly not yours results in a report to the Office of Student Conduct.

- Students must follow the University Code of Student Conduct at all times. The code prohibits unauthorized video or audio recording of lectures.

University Notices

- Any student who feels she/he may need an accommodation based on the impact of a disability should contact the Office of Accessibility at 330-972-7928. The office is at 105 Simmons Hall.

- The University of Akron is committed to providing an environment free of all forms of discrimination, including sexual violence and sexual harassment. This includes instances of attempted and/or completed sexual assault, domestic and dating violence, gender-based stalking, and sexual harassment. Additional information, resources, support and the University of Akron protocols for responding to sexual violence are available online.

- Refer to University COVID-19 policy. Masks are optional in this class.

University COVID Notice

- The COVID-19 pandemic is still present and serious. Before entering class, you should have completed your daily health assessment. You should not come to class if you fail your health check or feel ill. At that time, I also ask you notify me that you will be absent. When campus policies require masks to be worn indoors, all students are required to wear a mask during in-person classes. While you are in class on campus, you are required to: always cough or sneeze into your elbow or a tissue and adhere to other public safety protocols and directives for your specific classroom/lab/ studio. Students who do not follow these health and safety requirements will be instructed to leave class immediately. Students who violate this protocol will need to leave the classroom and MAY be marked absent. Repeated violations of these health-saving protocols may lead to sanctions under the Student Code of Conduct up to and including suspension or expulsion. Current guidelines can be found at: uakron.edu/return-to-campus/.

Instructor

- JD Kilgallin (Jonathan Kilgallin)
- Office hours: Wednesday 6:45-7:45
- Office: CAS 230
- Email: jdk72@uakron.edu, jd.kilgallin@keyfactor.com
- Employer: Keyfactor (keyfactor.com, github.com/keyfactor)
- Job title: Senior Integration Engineer
- Professional networks:
  - facebook.com/jd.kilgallin/
  - linkedin.com/in/kilgallin/

Bio

- 7 years in Keyfactor engineering organization
- B.S. in Computer Science + Math + Philosophy, Carnegie Mellon, 2010
- M.S. in Computer Science, University of Akron, 2015
- Prior SWE experience: Microsoft, Facebook, Google, Lockheed Martin
- Publications:
  - Data mining associations between subreddit communities (IEEE CSCI)
  - Optimizing on-campus directions (IEEE CSCI)
  - Compromising weak RSA keys on IoT devices (IEEE TPS-ISA)
  - Counting solutions to combinatorial seating problem with constraints (OEIS)

Keyfactor

- Cryptographic asset management: digital certificates, PKI, keys
- 20+ years of security consulting
- 10+ years of enterprise software
- 2016 investment by GM Ventures
- 2018 investment by Insight
- 2019 acquisition of Redtrust
- 2021 acquisition of PrimeKey
- Solutions used at scale in automobiles, airplanes, implanted medical devices, government agencies, and fortune 500 companies

Job Referrals

- Will provide references for A/B average post-midterm.
- Will give advice on résumés and job-seeking any time. No guarantees.
- Keyfactor routinely hires interns and new grads from Akron.
- Positions include:
  - Software developer (main product suite)
  - Integration engineer (cert. issuance and management in other platforms)
  - Solutions architect (implement products and integrations for customers)
- Ideal skills (not *required*): C#, .NET, Java, Cryptography, REST, DevOps
- Will offer Keyfactor recommendations case by case.

Survey

- Preferred name
- Major
- Graduation/Career plans
- Preferred programming languages
- Existing GitHub account?
- Laptop for class available?
- Office hours conflicts?
- Slide calibration check
- Hobbies (for tailoring examples used)
- Questions and interests

References

- Course website https://github.com/kilgallin/SWEF22
- Textbook - Software Engineering: A Practitioner's Approach, 9th edition, by Roger S. Pressman
- University of Akron Code of Student Conduct
- University of Akron Coronavirus Information
- University of Akron Office of Accessibility
- University of Akron Title IX
- Instructor professional social networks: Facebook and Linkedin
- Instructor IEEE publications and OEIS contribution
- Keyfactor website and GitHub
- *Reading for next lecture: Pressman Ch 1*