Course Catalog:

Scrum for software development. Scrum process, user stories, acceptance criteria, and test programming. Students work in Scrum teams.

Credits: 1

Prerequisites: CS221

Course Learning Objectives:

Student teams apply the Scrum methodology to software development. More specifically, at the end of the class students should be able to:

- Define what a software development life cycle is
- Describe why teams need a software development life cycle like Scrum
- Define Scrum and identify when Scrum is an appropriate methodology to use and when it is not
- Enumerate, explain and demonstrate being in various Scrum team roles
- Enumerate Scrum artifacts generated during Scrum activities and differentiate the purpose of these activities
- Elicit requirements in order to create a Product Backlog of prioritized and estimated User Stories and relate User Stories to Acceptance Criteria
- Estimate User Stories
- Demonstrate proper application of the Sprint planning activity to create the Sprint Backlog
- Convert User Stories into tasks and distribute the work among team members
- Translate requirements into design, implementation and testing code
- Recognize when a user story is done by applying the:
  - Acceptance testing
  - Definition of Done (DOD)

Grading:

- Team Project: 30%
- Individual Assignments: 20%
- Quizzes: 30%
- Final Exam: 20%
Topics:

- Teams, Software Development Methodologies, Agile Development, Scrum, Sprints and Roles
- Product Backlog, User Stories, Planning Poker, Acceptance Criteria
- Sprint Planning, Sprint Backlog, Tasks, Daily Standup Meeting, Sprint Review and Sprint Retrospective
- Definition of Done, Burndown Charts and Scrum Tools

Textbook: