Course Overview: CS189A

• Week 1
  – Class: Introduction to SWE, team management, and processes
  – Friday in lieu of discussion section
    • Company representatives present the challenge problems (Pitch Night!)
    • Sep 30 4-8pm in HFH 1104 (attendance mandatory)
• **Week 1**
  – Class: Introduction to SWE, team management, and processes
  – Friday in lieu of discussion section
    • Company representatives present the challenge problems (Pitch Night!)
    • Sep 30 4-8pm in HFH 1104 (attendance mandatory)

• **Week 2**
  – Oct 3: Form teams (**4-5 members**)
    • Identify **group leader** and **scribe**
      – **Lead**: motivator, picks up all loose ends, settles debates/makes decisions
      – **Scribe**: records scrums, retrospectives, sprint planning, mentor/TA meetings
    • Choose team/project – Select by 5pm (see **Claiming** process)
Course Overview: CS189A

- **Claiming a project**
  - Only **complete** emails dated after **5pm** on Monday Oct 28 considered
  - Email to Chandra (ckrintz@gmail.com) with
    - Subject: 189a project selection
    - Sent by group leader (or his/her representative)
    - List of group member names, **identify lead and scribe**
    - A picture of each group member for public posting
    - List all company participants in order of preference
  - FCFS assignments; you are **not** likely to get your top preferences
    - One team per project/company!
Course Overview: CS189A

• Week 2
  – Oct 3: Form teams (4-5 members)
    • Identify group leader and scribe
      – Lead: motivator, picks up all loose ends, settles debates/makes decisions
      – Scribe: records scrums, retrospectives, sprint planning, mentor/TA meetings
    • Choose team/project – Select by 5pm (see Claiming process)

  – Oct 5
    • Turn in team info to TA by end of class
    • (1) Team Google group; (2) Daily scrum schedule, (3) Weekly Mentor meetings; (4) 1-sentence description to TA via email; (5) Setup Github repo/email TA (add TA/Chandra to team if private)
    • Work on vision statements
      – Send to mentors after class requesting feedback

  – Oct 7: Turn in vision statements via email to TA by end of discussion
    • Plan Sprint I (which starts Monday)
Four 2-week sprints:
- Oct 10-28 (PRD v1 – tools, technologies, design);
- Oct 31-Nov10 (design and prototyping, PRD v2);
- Nov14-28 (design, prototyping, testing, PRD v2);
- Nov 28-Dec9 (prototype presentations, prototyping and testing)

Specify what the product will do
- Vision statement
- Product Requirements Document (PRD)
- Design tools, brainstorming, coding (tests and implementation)

Build and test an initial prototype
- Typically teams iterate on these activities until they converge to a working prototype!

189A last two weeks of class + discussion
- Prototype Demonstration
- Recorded, order determined randomly
Your Grade

- Attendance at classes and discussion section
  - One miss for class/discussion is allowed with excuse, followed by letter grade decrease per miss
- Participation in class discussion, questions for speakers
- Weekly substantial contributions to code repository
- Completion of all of the project requirements (next slides)
- Demo performance
  - Judging criteria, amount of work put in, robustness, extensiveness
- Group participation: peer grade to reflect the level of effort and contributions to the project throughout quarter
Capstone Project Requirements (1/2)

• Use of agile development process with per-sprint task tracking (recommended: Trello or PivotalTracker)
• Daily scrums recorded by scribe in shared Google Doc
  – Class/discussion days: last 15mins of class
  – Shared with Instructor, Mentor, TA, and team
• Weekly meetings (virtual is ok) with mentor (scribe logs)
• Weekly meeting with TA (discussion/class is OK)
• Class/discussion attendance and participation in team activities
  – Bring laptop to class, email Chandra if you don’t have one
• Vision statement turned in by deadline (& approved by mentor)
• Draft 1 & 2 of PRD turned in by deadline
  – Evolve as you design and prototype; approved by mentor
• Working prototype for base functionality demonstrated in the last week of the quarter
Capstone Project Requirements (2/2)

• Use of a code repository (recommended: GitHub)
  – **Ongoing** contributions by all members **each week**
    • Identify a workflow that works best for your team
  – Can include preparation of requirements documents
• Use of an issue tracker (recommended: Waffle.io or github)
• Documented code
• Automated unit tests and integration and/or functional tests
  – Code defensively!
• Use of user stories and/or use cases for requirements & design
• Use of UML for system requirements modeling and design
• Wireframes for user interfaces if any
• Complete 4 2-week sprints, record retrospectives and planning for each
F16 Deadlines

- Oct 3 by 5pm
  - Project choices (using format specified) to instructor via email
- Oct 5 in class: complete setup and start on project planning
  - Github repo, project description sentence, google doc & group setup and sent to TA via email (invites sent to TA, instructor, team, mentors)
- Oct 7 by end of discussion section
  - **Vision statement** (email PDF to TA)
  - Scrum intro, Plan Sprint 1 (starts Oct 10)
- Oct 28 by end of discussion section
  - Demo and retrospective (retro) for Sprint 1, plan Sprint 2 (starts Oct 31)
- Oct 28 by end of discussion
  - Project requirements doc (PRD) (using format specified) Draft 1 as PDF to TA
- Nov 9 by end of class
  - Demo and retro for Sprint 2, plan Sprint 3 (starts Nov 14)
- Nov 28 by end of class
  - Demo and retro for Sprint 3, plan Sprint 4 (starts Nov 28)
- Nov 28 by **start** of class
  - Project requirements doc (PRD) Draft 2 as PDF to TA
- Nov 28, 30, Dec 2: Project Demos to class/discussion
- Dec 9 by 5pm Demo and retro for Sprint 4; plan development over break
2-Page Vision Statement

- PDF via email to TA by end of class (class time allocated) on date specified
  - Project Title / Name (can change)
  - Team name, members names/emails
  - Team lead
  - what the project is about
    - What problem the project is solving (what is innovation, the science, and new core technical advance)?
    - Why the problem is important
    - How the problem is solved today (if it is)
  - Identify the outcome of the project
  - Define initial project milestones: specification, design, prototyping
  - How do you plan to articulate and design a solution
    - List the implementation platform and technologies will plan to use to develop the solution
    - Overview the process model you will employ to achieve the milestones
Getting Started – see Link on Web Page

• Outline project investigations, **divide up work** across members
  • **Teach each other**

• Identify software systems you want to use to track progress/files
  – Trello, github, google docs, pivotal, sphynx, issue trackers

• Identify programming languages, frameworks
  – Work on **tutorials** if new to you
  – **Testing frameworks and mock tools** (balsamiq), language/web frameworks, UML
  – Mobile development platforms: Coronalabs.com, iOS, Android
  – System configuration: Ansible, Puppet, Chef, Saltstack/Saltcloud
  – Cloud technologies
    • Infrastructure (servers): Amazon EC2/S3
    • Platforms: Google App Engine, Heroku
    • Mobile Backends: Backendless, Google Endpoints
    • Services: MongoLab, Instacluster, Amazon RDS, Hadoop/ElasticMapReduce
    • APIs: Twitter, Facebook, Google technologies (maps/earth/drive)
    • Development environments: Apache JClouds