Milestone Four

Primary Objectives:

1. One full (1 week) iteration of class project
2. Team project inception continues

Overall Requirements:

- Class Project 1st Full Iteration:
  - All iteration artifacts updated in Pivotal Tracker, Git repository and features deployed
- Team Project Inception (Part 2 of 3), for example
  - Lists of needs and features
  - List of requirements, NFR's, outside data
  - Overall architecture design, API's needed
  - Initial modeling (initial database design, use cases, UI design ...)
  - Timeline and release plan
  - Epics/User stories
  - Vision Statement
  - Identification of Risks

Grade sheets (ods)

In this milestone we complete the first full (but one week) iteration/sprint of the class project. For the team project you'll work hard on inception activities to get a good handle on what your project will entail.

Tasks:

1. **Complete a full one week Sprint for the class project. This is nearly the whole thing: Backlog Refinement and Grooming, Standups each class day and Sprint Review with your advisor (Retrospective will be next week).**
   - Backlog Grooming and Refinement — You need to prepare the backlog to be ready for planning the next sprint. Look at your icebox and prioritize it. Talk about these items and their feasibility. Examine the top one. Is it small enough? If not, break it apart, define, describe and re-prioritize. Each user story here **must** be a complete feature representing a thin vertical slice, and meeting as much of the INVEST qualities as you can. *It's really important to try hard to get these right. User stories that don't meet these criteria are difficult to complete and you're likely to have a hard time with them and end the sprint with fewer points.* Take the top item and talk about it with your team. Do some Just-In-Time (JIT) modeling and design. Re-estimate effort. Write descriptions and something about what it means to be "done". Do all the things you've seen in the video series and the DAD book. Got the first one into shape? OK, go on to the next one. Do this until you have 8-10 effort points worth of stories ready to go in your icebox. This represents a little more than you'll likely be able to put into the next sprint, which is also an abbreviated one like Sprint 1.

- Sprint Planning — Allocate 5 (3 person teams) or 7 (4 person teams) effort points, +/- 1 for Sprint 2, corresponding to 50% and 70% capacity respectively for a 1 week sprint. It's OK to have 2 point stories in this sprint but nothing bigger (no 4 or 8). Work on the descriptions for each item as you put it in the sprint. Do some more JIT modeling and design, write some tasks, do a sanity check. If it's all good, move on to the next one and so on until you have the sprint full. Again, do what you've read about in the DAD book and seen in the Scrum series videos.
- Construction — Commit to and fulfill the sprint. Allocate stories to team members in an equitable manner so everyone has roughly the same workload. During the sprint, keep all your team members updated. Write notes in the Activity section for your story in Pivotal Tracker. Put in the name of your feature branch. Update the description when things change. When your story is done and done, merge the latest `dev` branch into your feature branch to make sure there aren't any merge conflicts, then create a Pull Request. Move on to your next story if you have one.

Be ready to demonstrate your application (on Azure) at the review meeting. Deployment should be off your dev branch.

After your review meeting, all accepted stories get merged into master. Use the integration branch to do this.

2. **Team Project Inception phase II** — Spend quality time working on your team project. Go through the inception phase activities as we did for the class project and produce appropriate artifacts (see above and use the template from before). You're not designing or specifying the entire project, but you should be getting some good definitions of what the project will do and what high-level architecture and design is needed. Every team member should be on the same page and top features should be clear.

   In terms of the “Level of Detail” expected, if you don't end up with 3-4 well defined high level core features, translating into 6-10 Epics, with the highest priority epic broken down into some good user stories, then you've not thought of it enough. Quality matters here much more than quantity.

   We will review each Team's Vision statement during class so all teams know what all the projects are.

   You're putting everything in your Git repo right? In the markdown file is fine for most text; we'll create a new Pivotal Tracker project for you next week to start entering these.