

- 1) Class Retrospective
 - a) Conduct Safety Check; **everybody is good**
 - b) Learning Outcomes: **break down features into specific user stories with better details of tasks, less dependencies**
 - c) Questions on project; **Login Features break down, procedures: Social Media API completed, writing api still needed.**
 - d) Improvements: **separate user stories so they not dependent on each other; break down Epics into smaller bits: assure same page between members: be clear on which members are working on which user stories.**
 - e) Does the teacher need to be involved? **No at the moment.**
- 2) Team Project; update model diagrams/update architecture drawing: **Shared the document with team members so all can edit; splitting tables with login features**
- 3) Create a Welcome Page on Bitbucket;
 - a) Description of project, rules when writing code/database, such as naming conventions
 - b) List all team members involved
 - c) Software constructions processes
 - d) Team Rules?/Song/cheer/dance/Mascot
 - e) List all items that need to be installed by Nuget/include version numbers(its maybe a work in progress)
- 4) Backlog Refinement/grooming.
 - a) Select user stories for Sprint 1, change/add items if needed, commit to Sprint 1 and assign team members
 - b) Each member needs to perform 18 e.p. Per week. We will need at least 144 effort points for our first Sprint between all team members
 - c) Each members chooses their tasks for the week, second week choose them on-the-fly
 - d) Print out Sprint 1 backlog
 - e) Fill out Sprint review sheet, **fill it out one for each team members before the Sprint Review meeting next week and bring to the meeting.**