As we discussed in our first meeting, the department grading criteria can be found here.

<https://www.seattleu.edu/scieng/computer-science/student-projects/mse/>

# MSE Project Grading Criteria

I have been asked to put together a very specific set of objective grading criteria. I want this document to serve as a contract. I would like you to review it and respond that you understand it. If you DO NOT respond, the assumption will be made that you have read and understand it.

Grading will be: A 90%+, B 80-90%, C 70-80% etc.

The overall grade is based on the actual project work done and your use of good software engineering practices. The department web site lists out the areas we feel are important for software engineering practices and their relative weights. Each project advisor can modify this list to reflect the reality of their team’s project.

Since we began a project that had already undergone significant requirements identification, we are focusing more on the deliverables for the second quarter. Here is what I expect from you to meet the department SWE goals:

## Q1

Project Planning – 15%

Requirements/Business Rules – 10%

Design - 25%

Presentation – 20%

Test Plan – 10%

Peer and Self Eval – 5%

Advisor Eval – 15%

## Q2

Project Planning – 15%

Requirements/Business Rules – 10%

Design - 20%

Presentation – 20%

Test Plan – 15%

Peer and Self Eval – 5%

Advisor Eval – 15%

## Q3

Project Planning – 15%

Requirements/Business Rules – 5%

Design - 5%

Presentation – 20%

Test Plan – 15%

Peer and Self Eval – 5%

Experience Report – 10%

Delivery Documentation – 10%

Advisor Eval – 15%

## Q1 Grading

**Project Planning (15%)** – by the end of the quarter you should have a backlog that contains all the high level tasks (user stories, etc.) that make up an MVP product. You will likely have other non-MVP tasks in your backlog as well. The high level MVP tasks should have a story point ranking associated with them. Large, Medium, Small is sufficient and your goal is to simply try and size these tasks relative to each other. A goal would be to break these high level tasks into user stories such that a Small could be accomplished by one person in a sprint, Medium could be accomplished by two people in a sprint, and a Large might require two sprints. Your backlog should be updated at the end of each sprint as needed. Adding, deleting, merging tasks, etc. (5%)

You should have the high level tasks that you expect to accomplish in the next sprint broken down into low level tasks with hour estimates assigned to them. These low level tasks should also be assigned to an individual. Sprint burn down charts should then reflect the tasks being accomplished. (5%)

Based on the total story points for your MVP tasks, you have a total value of the project. I would like to see a simple burn up or accumulated value chart that shows your progress to the goal. For example, if your backlog story points add up to 100 and you finish 8 story points in a sprint, you would indicate that you are at 8 out of 100. I only want to see this at the end of quarters. (2%)

The last (3%) of project planning will be based on effective meetings. I have been disappointed with the disorganized meeting schedule this quarter.  *By end of first week in January, I want 5 end of sprint demo meetings with Elizabeth scheduled. I do not want this schedule to change during the quarter. This should be like clockwork. At least one of you needs to be at this meeting on time and ready to run it. The schedule should be, Demo work done, get feedback, readjust backlog, determine high level tasks for next sprint. If this doesn’t happen EVERY time, you will all lose this 3%.*

**Requirements/Business Rules (10%)** – Create a business rule document that logically organizes and documents the business rules as we encounter them.

**Design/Architecture (25%) –** Look at the recommendations on the department web site. Basically you should create a document that identifies key system quality requirements. In our case they would be maintainability, extensibility, scalability, usability and security. You then need to describe what you are doing in terms of both architecture and process to meet these goals.

For example, maintainability - using an MVC arch with separate code pages, data access layer abstraction, etc.. Scalability – cloud arch. Usability – iterative dev with client reviews and eventually perhaps usability studies. Security – describe Su and non-SU accounts.

In addition you should have an ERD model and scripts to recreate the database and seed it with test data.

Also, for maintainability, I would like to see any non-trivial code routines or stored procedures include a brief description of their purpose. I would also like to see comments in the data model scripts that clearly define any data fields that are not obvious.

**Test Plan (10%) –** You need to create a document that identifies how you are approaching testing with an appendix to hold all your test plans as they are developed. I want you to do test development as you proceed, not put it off to the end. So, to say that a user story is completed and its story points can be added to the cumulative value chart, you will need to write whatever test plan you feel is important for that story.

**Presentation (20%)** – your presentation should be professional, clearly organized and rehearsed and speak to all the topics listed above as well as lessons learned.

**Peer and Self Eval (5%) –** The form is on the department web site. I will not grade off based on a negative evaluation you receive from another student. However, if you do not turn this in you will lose 5%. If there are consistent concerns being voiced, we will address them.

**Advisor Evaluation (15%) –** this is where I will look at individual level of effort and effectiveness. Professionalism will also come in here.

Grading for Q2 and Q3 will be similar, but note that in Q3 you are expected to write an experience report. You should read these guidelines now and plan accordingly.

For all the grading issues except the Advisor Evaluation, the grading will be done by the department and I will be just one of the graders. Follow the guidelines and you will be fine.

I want to break down the *Advisor Evaluation* grade portion so we are absolutely clear on my expectations. This section focuses on getting the work done properly. My expectations are:

End of Q1 –

* Development architecture in place
* CRUD for Partner Opportunity with scripts outlining steps for acceptance testing
* Student dashboard page created
* Student list of current sign ups
* Database schema 95% complete
* Database views, stored procedures, triggers created as necessary to support the above functionality.

By the end of Q3 I would like all the functionality for an MVP. So, to be clear about what is expected,

* Partner CRUD for organization, opportunity, organization people
* Partner write evaluations and approve time
* Student signup for opportunities, drop opportunity, view/edit profile, view/export opportunity work history, view/enter work hours, view/enter evaluation and reflection
* Professor (current qtr only) view opportunities for their sections, view students in their section and the opportunity they are signed up for, view hours, reflection, evaluations for a student for an opportunity
* Admin – basically all the above functionality plus Assign opportunities and slots to sections, update student CRC, add community partner accounts
* Security – we may not be integrated with SU security but you should have a document that describes how your system should be hooked into SU security
* Data Load – ideally we should have access to SU data downloads and have scripts written to refresh the data in our system. If this dependency is problematic, you should at least have a document explaining how the data load process should work.

Given this definition of an MVP, after this quarter you have 10 sprints left. Given this backlog I would expect roughly half of the remaining work done by the end of Q2. I will let you determine what you view as half the backlog and get that to me first week in January. If, at the end of the quarter, you only accomplish 50% of the work, you will get 50% of the points for this section (15 points total). This will be looked at on an individual basis.

Going forward, I will only be attending the end of sprint meetings with the client. If you have questions, send them to me and I will address them. If you have requirements from SU IT department I will send you the contact name and email but it is up to you to schedule any meetings and get your questions answered.

Your action items:

Prepare for end of quarter deliverables

Schedule all 5 end of sprint meetings for Q2 by the end of first week in January or lose 3% of your quarter grad.

Present me with the portion of the backlog you expect to accomplish in Q2 by end of first week in January.