Milestone 3:
Project Requirement Documentation

Learning Goals

- Identify an appropriate scope for your system beginning with goals/objectives of the project.
- Synthesize requirements from various sources, including stakeholder interviews and other resources (e.g., literature review).
- Document all information sources used to acquire knowledge of your system, and justify why these sources are authoritative in a bibliography.
- Create user stories documenting the functional/quality requirements of the system.
- Create user interaction designs to clarify requirements
- Learn to balance design trade-offs and realism in your user interface designs
- Develop and apply appropriate metrics to specify and measure quality attributes.
- Learn to think critically and deliver constructive criticism.

Deliverables

1. (Group) Project Requirement Document (50pt, 2% of final mark)

Each *group* should submit a single PDF of their completed requirements document on Quercus **by Sunday, 10/4, 23:59pm EST.**

Format: Only one team member should submit a PDF file to Quercus called ECE444_[GroupNumber]_Milestone3.pdf containing all the information above. Please also save this file in your Shared OneDrive folder.

Document Outline
(Note: no need to follow the structure section by section. As long as your report is clear, well-organized, and easy to understand by others.)

Title: [Name of your system] Requirement Document (RD)
Note that you do not have to stick to 'Chef Co-Pilot'. Feel free to create your own.
(*Example: Chef Bear Café – Requirement Document (RD)*
Prepared by (format: [GroupNumber_GroupName]: author names)
(*Example: [Group21_Fork] – Grizzly, Ice bear, Panda,..)*
Date submitted
Version
(There could be multiple revisions so a number assignment helps to make sure all are on same content.)
Last updated date
Creation Date

Sec. 1. Goals (2pt)

Define the most important goals in a short paragraph.
Example:
The objectives for this system are to create a graduate student admissions system that enables productivity for all users. This will be achieved by ensuring that the system is easier to use for users of the system. The primary benefit of the system is to save stakeholder's time by ensuring accurate and useable tracking of applicant materials and decisions.

Sec. 2. User Personas (8pt)

Describe a diverse set of user viewpoints (between 2-3 personas for a single user role are expected)

Sec. 3. Functional Requirement (Use cases and user stories) (10pt)

Document the functional requirements in the form of *use cases(epics), user stories*. Provide at least one *use case* per student in your group. You don't have to capture every possible way of interacting with your system, but you should capture the important ones. One test of whether a use case is important is whether any of your stakeholders would be upset if the system did not support that use case. The user stories should follow the conventions discussed in class.

Note that you will need to prioritize the requests you get from your stakeholders and decide which of them you plan to support in your system (see also conflicts below). You should not write use cases for everything your stakeholders mention; you should choose wisely according to a set of features that is cohesive and represents a reasonable first iteration of a product.

From the transcripts and source documentation, identify a non-trivial aspect of the system (i.e., trivial aspects include problems that have routine solutions, such as user account management – exclude these from your project's focus). Instead, focus on what is novel and unique about this problem; this may be what is least understood by your team members.

There is no specific number of goals/use cases/epics that will ensure your success.

<u>To receive full credit on the functional requirements, we expect:</u>

- At least *n* use cases (for a team with *n* members) that describe important system functionality at an adequate level of detail and quality.
- Enough user stories to describe the rest of the system's functionality, following the conventions and quality expectations of user stories.

Sec. 4. Quality requirement (5pt)

Define and order the *five* most important quality attributes within the context of the system. A good description of quality attributes unambiguously defines what the requirements mean within the context of the system; why each requirement deserves the given rank; how to measure each requirement; and how those metrics should be interpreted. If two or more quality attributes are equally important, it is acceptable to give them the same rank, but be sure to justify why they are equally important.

To receive full credit on the quality attributes, we expect:

- The identification and description of at least five (5) quality attributes, including a ranking of their importance and a coherent justification of that ranking.

Sec. 5. User interaction and design (8pt)

Link the design explorations and wireframes (or other high-fidelity designs) to the page. You could also attached designs with each user story, it helps them visualize the customer journey and makes your life easy too.

Please experimenting with the high and low fidelity prototyping tools as alternatives to design your interfaces. If you use a particular tool, discuss your experiences in your reflection.

Sec. 6. Questions (5pt)

As the team solves the problems along the project progression, they inevitably have many questions arising. A good practice is to record all these questions and track them. (see example in appendix)

Sec. 7. Out of Scope (1pt)

List the things which you aren't doing now but plan on doing soon. Such a list will help you organize your teamwork and prioritize features.

Sec. 8. References (1pt)

Appendix A. Conflict Requirements and Resolution (5pt)

Briefly describe and discuss all conflicts you identify during elicitation. For conflicts that you could resolve, describe how you resolved them. For conflicts that you could not resolve within this project development lifecycle (this semester), briefly outline how you would resolve them if you were doing this for a real customer (e.g., be concrete of whom you might ask which question and how answers would inform your decision).

To receive full credit on the conflicting requirements, we expect:

- A description of all conflicts you identify during elicitation.
- For conflicts that you can resolve, describe how you resolved them. For conflicts that you cannot not resolve within this semester, briefly outline how you would resolve them if you were doing this for a real customer

Appendix B. Quality Assurance (5pt)

(This is also part of the individual reflection report)
We established in class that good requirements should be Correct, Consistent, Unambiguous, Complete, Feasible, Relevant, Testable, and Traceable and discussed the INVEST method for quality assurance of user stories. Briefly describe what process you used to assure the quality of your requirements. Afterward pick N of your user stories (*N = number of team members*) and explicitly argue why they meet the quality criteria -- we recommend an explicit list in which you argue separately for each criteria (INVEST) and why your user story fulfills it.

To receive full credit on the quality assurance discussion, we expect:

- A brief but comprehensive description of the process you adopted to assure quality of your functional requirements.
- An explicit discussion of qualities of user stories

Appendix C. Glossary (if needed)

You will create a glossary of relevant domain terminology. Include any specialized terms that appear in your use case descriptions and that deviate from standard uses (i.e., meanings that are different from what one would find in a standard English dictionary).

---

2. (Individual) Individual reflection report (25pt, 1% of final mark, <=1 page)

(Due **Sunday, 10/4, 23:59pm EST**)

1) About Use Story confirmation
Briefly describe what process you used to assure the quality of your requirements. Afterward pick one of your user stories and explicitly argue why they meet the quality criteria -- we recommend an explicit list in which you argue separately for each criteria and why your use case/user story fulfills it.

(You could directly copy the part from the group report, but make sure each team member copy different user stories.)

2) About low/high fidelity design
Please experimenting with the high and low fidelity prototyping tools as alternatives to design your interfaces. The depth of your reflection to express the advantages and disadvantages of the alternatives you considered is important.

Format:
ECE444_[GroupNumber]_[YourFirst&LastName]_Milestone3Reflection.pdf containing all the information above. For example: ECE444_Group21_ShuruiZhou_ Milestone3Reflection.pdf

To receive full credit on the individual reflection, we expect:

- An analysis beyond mere descriptions and superficial statements, including supporting evidence for claims, that reflects on the causes of identified issues or your own experience

3. (Individual) Peer review (50pt, 2% of final mark)

(Due Wednesday 10/7 11:59PM)

Provide reasonable and constructive comments in ratings with regard to evaluation criteria for the group report. Effectively critiquing the work of others and receiving such criticism is an important communication skill in any technical profession. You need to be honest, direct and always constructive.

Assume that you are the customer (or other stakeholders) of this product and is involved in the agile development team. When reading the requirement documentation, try to figure out: did the team miss any questions or overlook any ambiguities? Are there missing stakeholders (e.g., did the team overlook a critical persona)? Is there any inconsistency/ambiguity in the document?

Remember, you are trying to provide assistance to another project team. The objective is not to make yourself look smart, or to look smarter than your colleagues on the other team. The goal is to do your utmost to help them do the best job they possibly can. This includes pointing out the strengths of their work, being very clear about its weaknesses and shortcomings, and providing the most helpful, practical suggestions that you can.

Task:

1) Comment on one strength
2) Point out two weaknesses (ambiguity, inconsistency, important info missing, etc) and suggest how to improve them

Please leave comments to the group report on Quercus.
Your project critique will be evaluated based on:
• A clear, honest, direct, and accurate expression of strengths and weaknesses.
• Maintaining a constructive tone throughout.

Appendix (Requirement Document Example)

One web-page software requirements document created by using Atlassian Confluence (https://www.atlassian.com/software/confluence), the content collaboration software

Mobile Web Requirements

Created by Mitch Davis, last modified just a moment ago

Target release	1.0
Epic	MDT-18 - Mobile optimized web app TO DO
Document status	DRAFT
Document owner	@Mitch Davis
Designer	@Cassie Owens
Developers	@Harvey Jennings
QA	@Kevin Campbell

Background and strategic fit

We all know mobile is on the rise. A recent survey to customers showed that 85% of users use their mobile on a daily basis. Most of our customers also use competitor apps, so this is something we need to have.

Customer research

- Customer interview - Netflix
- Customer interview - Homeaway
- Customer interview - Bitbucket

Requirements

#	User story title	User story description	Priority	Notes
1	Facebook Integration MDT-13 TO DO	A user wants to sign up via Facebook	Must Have	• We will need to talk to Cassie Owens. • There has also been some research done on this (see Facebook integration prototype)
2	Activity Stream MDT-14 TO DO	A user wants to view the latest updates via the mobile dashboard so that they can get a better understanding of what is in place	Must Have	
3	Post Updates MDT-15 TO DO	A user wants to be able to post status updates on the go	Must Have	The key things we will need to support: • Text status updates • Mentions • Support for images • Smart embedding for YouTube vids
4	API MDT-16 TO DO	A developer wants to integrate with the mobile app so that they can embed the activity stream on their website	Should Have	• We should chat to Team Dyno as they did something similar.

User interaction and design

Description	Login screen	Activity stream
Mockup		

Questions

Below is a list of questions to be addressed as a result of this requirements document:

Question	**Outcome**
What about Google Apps	• We think this is important, but not for version one. • We can look at this at a later stage. • 💡 It might be worth someone looking into a shared notification library to do this.
Are we supporting Blackberry?	• Again, not for initial version - but we haven't had much demand for this.
Should we have an offline mode?	• We've talked about the pros and cons. In brief: ➕ Seamless experience for customers, they won't notice if there is a connection issue ➕ Most of our competitors don't have this ➖ Could be expensive to build ❓ Should we spike this at a later sprint?

Not Doing

- Google Apps Authentication - out of scope, see above for details
- Blackberry support - we won't look at doing this, if demand picks up we can look at it.
- Native app. We are starting with a mobile web view first and get back to a native app depending on feedback that we get.

Like Be the first to like this requirements