Grading rubric for Requirements Specification Document

Team Name: Javascript Deniers

Max Points	Team Points	Content
10	10	Do the requirements clearly state the customers' needs?
5	5	Do the requirements avoid specifying a design (*note: customer-specified design elements are allowed; non-functional requirements may specify some major design requirements*)?
		Completeness
25	25	Are use cases written in sufficient detail to allow for design and planning?
4	4	Do use cases have acceptance tests?
20	17	Is your use case model complete? Are all major use cases included in the document?
8	8	Has the team provided an appropriate swim-lane diagram for the scenario where faculty reviews a student's application?
10	10	Are the User Interface Requirements given with some detail? Are there some sketches, mockups?
		Clarity
4	4	Is the document carefully written, without typos and grammatical errors?
2	2	Is each part of the document in agreement with all other parts?
2	2	Are all items clear and not ambiguous? (Minor document readability issues should be handled off-line, not in the review, e.g. spelling, grammar, and organization).
		GitLab Issues
10	5	Has the team setup their GitHub Issues page? Have they generated the list of user stories, use-cases, created milestones, assigned use-cases (issues) to milestones? Example GitLab repo (check the issues): - **Missing several use cases. No user stories. Only iteration 1 use cases are present.**
100	**92**	**TOTAL**

Use Cases:

1. Register user (Student/Faculty)
2. Login user (Student/Faculty)
3. Edit profile (Student/Faculty)
4. Create research position (Faculty)
5. View applications (Faculty)
6. View the qualifications of each student (Faculty)
7. Update application status (Faculty)
8. Delete research positions (Faculty)
9. View all research positions (make sure the status are also displayed) (Student)
10. View recommended research positions (Student)
11. Apply research positions (Student)
12. Withdraw pending research positions (Student)