Success Spectrum Exercise

1. Begin by naming your **high-level goals** (what you'd like to accomplish & why)
2. For all of the following, be as specific as possible!
   - Fill out the "Minimum" column
   - Fill out the "Target" column
   - Fill out the "Epic" column (*do this one twice, since we tend to self-censor)
   - Fill out the "Failure" column
3. Now look over the whole board. Are there discrepancies? Overlap? Are the success scenarios too hard or too soft? Do they reflect what matters most?
4. If necessary, readjust and reframe until each team member feels good about what's on the spectrum!
5. Upload a copy of your team's Success Spectrum onto MyCourses. **(Submit 1 copy per Team)**

Example

Minimum Success *Success means achieving 100% of these*	**Target Success** *Success means achieving 40-60% of these*	**Epic Success**	**Failure** *Success means avoiding 100% of these*
What has to happen in order for the project to meet basic expectations and succeed? *This should be attainable if everyone does their part*	What are you hoping will happen? *This should be challenging, and have a 40-70% likelihood of happening if the team is diligent*	What is your dream picture of success for this project? *This goes beyond your target scenario and allows you to envision what you really care about*	What does failure look like? *Brainstorm scenarios that depict a failed project and/or failed sense of teamwork. Be specific!*

References and Credit

History and Credit:
Eugene Eric Kim and Amy Wu (Duende) created this toolkit. The idea of defining success along a spectrum was inspired by Kristin Cobble. The idea of explicitly incorporating failure scenarios was inspired by Seb Paquet, who also named the “Epic” column.

Goals + Success Spectrum (Faster Than 20 Toolkits): http://fasterthan20.com/toolkit/goals-success-spectrum/

Project: Service Methods, Restful API, service Unit Testing and integration testing

Time Span: 1 week

High Level Goal Statement : Achieve a perfect harmony in the backend between classes

Minimum Success *Success means achieving 100% of these*	**Target Success** *Success means achieving 40-60% of these*	**Epic Success**	**Failure** *Success means avoiding 100% of these*
All our addons to the second deliverable should not affect our tests or our implementations of the first deliverable. Our project should build and run without any errors. All the services must be written. We should write all the DTO's needed for the assignment. We should all help in the coding. We should write all testing necessary to avoid bad surprises. We should write all the controllers.	**All the services must be written in such a way that it eases our front-end development. We must achieve high coverage in all the Mockito testing. We should write our integration testing using JS Node to become more familiar with the way it functions. We should write good controller methods that encapsulates all our needs while maintaining some level of security in our application.**	**All our service methods should be both optimized and very well written to facilitate the understanding. We should have 100% coverage in our tests. We should cover all the failure cases in both our integration and our Mockito. Finally, we implement the detailed use case diagrams in our coding to the letter.**	**We shall not work intensely in short bursts. We should be able to plan all our work sessions in such a way that we scatter our workload effectively. We should not ask our teammates to solve each of our small issues. We should avoid code smells.**

Template