



March 15, 2018

From: Team 47

To: The Spoiled Tomatillos Board

Re: Assessing project's needs against the team's experiences and skills

Date: March 15th, 2018

Project's needs:

1. UI/UX Design
2. Spring/Spring Boot Experience
3. AWS, Jenkins, Sonarqube
4. Database Design
5. Project Management

In terms of the frontend, designing the UI has not proven to be much of an issue as our team is experienced in this area; keep in mind though that it is easy to design a basic working layout, but is very time consuming to make pages appealing for users. One challenge we are dealing with is our lack of experience and skill with Thymeleaf. This has proven to be very time consuming to learn as the syntax is unfamiliar and the documentation is not very clear. For example, using Thymeleaf to pass information from the frontend to the backend took half a day to figure out. Unfortunately, the only solution we have to avoid this problem in the future is to simply learn as we go; there simply isn't enough time to spend learning Thymeleaf first as we have lots of other aspects of the project to work on.

For the backend we are using Spring Boot to build our application. Besides one member on our team we do not have experience with the framework. We have to figure out how to build

our application almost from scratch. This includes learning how to configure our application, how to interact with our database, how to interact with our frontend, etc.. For instance we are currently figuring out how to maintain a user session but none of us have experience on the topic. Our plan to address this issue is simply to learn as we go and tackle each problem as team by bouncing information we have learned individually as well as utilizing our one member who has experience with Spring. Our lack of experience with the framework is certainly a gap that affects the progress of our project. However, as most of the tasks require some form of interaction with Spring, we are all learning rapidly and are able to fulfill the tasks we have been assigned to.

One major issue we have run into during sprint 3 is the cost of Amazon Web Services. Going into sprint 3, we had two separate instances, one for development and one for the final product. However, this became a major issue due to the unexpected costs to have two instances running along with two databases. Our AWS configuration accumulated a large bill and we were forced to close all instances to prevent further charges.

The solution to the AWS problem comes in two parts. The first part of the solution was to contact AWS support in hopes of a refund as well as clarification of what is covered under the free tier. The second part of the solution will be to get the old instances with databases up and running on free tier options. This most likely will restrict our databases to the smallest size option on AWS as well as setting budget restrictions on the instances. Fortunately, we were able to take snapshots of all of the instances and databases before terminating which will make the process of creating new free-tier instances easier. We have already taken action on the first part of the solution and are waiting for a response from AWS support, a response expected this week, before we move forward with part two of the solution.

Database Design was one of the aspects that most of us were unfamiliar with. As a result, whoever was in charge of managing the database had to do a lot of self teaching in order to get things up and running properly. However, since we are planning on rotating tasks frequently, the skills required would hopefully be transferred from teammate to teammate as each person experiences working with Sql. So far, in the database end, sending friend requests, receive and handle friend request, storing user relationship information in local database are finished. Creating movie list, adding movie into movie list functionalities are also finished in the backend local database. We will fulfill the functionalities after properly modifications.

Overall, project management has been relatively smooth. Each member has been consistently delegated manageable tasks by the team. We are rapidly working towards building a minimum viable product, in which we will show to the clients as to get a quick confirmation on our process. Although we have not made much explicit contact with clients regarding requirement changes as such, we plan to do so much more frequently in the future so that we may be more agile in the way we develop Rotten Tomatillos. For now, we are just burning

through the highest priority backlog items so that our product has the core functionalities it needs, such as search, or ratings. We are coming up with improvements to our use cases as we develop, but those are filed under low priority, and will be run by the clients as well before implementing.