Sprint 1 Report

Product: Recipe Retriever

Team Name: RexFetch

Members: Kim Ho, Vyankatesh Nandapurkar, Jackson Kohls,
Kevin Yosifov, Prithvi Arunshankar

Date: 04/25/2023

Note: The "team" or "group members" here refers to all team members, not just the development team.

Actions to Stop Doing:

1. The team members should stop waiting for people to assign them tasks, or tell them what to do before they start working. There were times when group members didn't have a clear idea of what they can or should work on, and they stalled until someone told them what to do. We should all be self-initiating our work, and if we need help we should ask either the Product Owner or the Scrum Master.
2. The team should stop being insular with ideas. If members have any ideas about the project or implementation, they should share it with the group and promote a group discussion. This will also make for easier integration, and give everyone an idea of where we all stand. This point goes along with the first one as well, because they are both critiquing a lack of communication.

Actions to Start Doing:

1. The team should feel more comfortable asking other team members for help or advice, and keep everyone in the loop about where they stand. This will promote more interconnectedness and prevent members from being lost or unsure about what they can work on.
2. Once someone completes a task or pushes some code to the repository, we want others to review it for correctness as well as understand what it is doing. Again, related to point 1, this will ensure everyone has a clear idea about the current state of our project.
3. The team should prioritize this project more and specifically make time to work on the project, as well as take more initiative when assigning tasks to themselves. We were not able to complete quite as many tasks as we wanted, and making this class a greater priority and taking more initiative would help with that.
4. The team should also make code testing a priority. We have some tests for the backend database functionality, but we want to integrate testing into everything we do, especially reproducible testing.
5. The team should have a better estimate of how many hours tasks might take, and a more realistic estimate of how many hours each member can put in.

Actions to Keep Doing:

1. Continue our impromptu stand-up meetings after class. It's a great way to check in and talk to our teammates and talk about things we need clarification on or ask for help. Also it's helpful to get a general sense of how far every member got in their tasks.
2. We've been fairly good in completing the tasks we self-assign. Trello has been helpful in keeping us on track for that, and we want to continue doing so.
3. Most of us did a great job of keeping everyone else in the loop about the things we are working on, and especially the problems we have. We want to continue doing that, and encourage even more of it.

Work Completed/Not Completed:

1. User Story 1 (not complete): As a user, I want to be able to send a request with the ingredients that I have, so that I can get a recipe(s) I can make.
    a. Communication Stack between frontend and backend (documentation complete)
    b. Parse and format incoming backend data (not complete)
    c. Server design (complete)
    d. Recipe caching (not complete)
    e. Parse incoming front-end requests (not complete)
    f. Parse incoming recipe data (not complete)
2. User Story 2 (complete): As a user, I want my login information and other relevant data stored in a profile, so I can keep track of my fridge content, favorite recipes, etc
    a. Set up user database and integrate with server (complete)
    b. Front-end changes (complete)
    c. Synchronize data with live front-end (not complete)
3. User Story 3 (complete): As a user, I want a list of what's in my fridge
    a. Front-end design (complete)
    b. Add ingredients data in front end (complete)
    c. Edit/remove ingredients data in front end (complete)

Note: A very important implied task was not completed, that of integrating the backend and front-end via our communication stack. Once that's done, User Story 1 would be closer to being complete. For now, we completed User Story 3 and most of User Story 2.

Work Completion Rate:

- Total number of user stories completed: 2 out of 3
- Estimated ideal work hours spent:

1. User Story 1 - 17% Complete
    a. Communication Stack between frontend and backend (documentation complete)
    b. Parse and format incoming backend data (not complete)
    c. Server design (complete)
    d. Recipe caching (not complete)

e. Parse incoming front-end requests (not complete)
f. Parse incoming recipe data (not complete)

2. User Story 2 - 70% Complete
   a. Set up user database and integrate with server (complete)
   b. Front-end changes (complete)
   c. Synchronize data with live front-end (not complete)
3. User Story 3 - 100% Complete
   a. Front-end design (complete)
   b. Add ingredients data in front end (complete)
   c. Edit/remove ingredients data in front end (complete)