Sprint 1 Review Report – GrinSync

During this sprint, we accomplished things that were more on the experimentation and setup side of our project. We did not do much on the product development end since we are still in the early stages of software development (we are just getting to the design and architecture stages). As such, we do not have much progress from a customer perspective.

Customer-visible progress:

- We created a Django project with basic API functionality.
    - We have the framework for a Django project set up, with basic models for both Users and Events. These are also now accessible, with URLs that allow API calls to get model information.
        - For the non-technical user:
            - We have set up the framework to store the basic information that we're going to need about the users of our site and the events that they'll plan
            - We have also set it up so that that information is available for use automatically, without needing to go in and look at the database
            - Basically, each person (and event) has a unique number, so if we give the website that number, it will package up all the info it has about the person whose number that is and send it back to us
            - We will use this in the app to pass info about users and events to the app from the database, which will be running on a server

Customer-invisible progress: (Can look at Trello Board for more details)

- We created a Trello board to track our issues.
    - We added issues for this sprint (see below) as well as more high-level, epic issues in the backlog for the future, which will be broken down into smaller issues as we encounter them.
- We divided our team into two sub-teams: one for backend development and one for frontend development.
    - Backend: Brain & Livia
    - Frontend: Bradley, Sam, Kevin, and Nam
- We discussed technologies to use for the backend development and decided on using **Django** since our team has experience with it.
- We researched the differences between two popular cross-platform mobile development frameworks, React Native and Flutter.
    - We researched which would be easier to learn given the skillset or our group members, which had accessible learning courses, and which would allow us to get our app up and running in a short amount of time.
- We discussed our findings from above and decided to use **Flutter** to develop our mobile app (for frontend development).
- We created a GitHub organization.

    - We decided to create an organization so that we could have separate repositories for frontend and backend development in one place.
- We organized our GitHub organization by adding proper README.md files and creating the repositories and folders we need (see GitHub organization for more details; it is best to start with our general README.md file on our overview organization page).