To Do to Submit Milestone:

- Create Readme in GitHub (Trevor)
- Application Description (Gio)
- Development Method (Spencer)
- Proposed Architecture Plan (Trevor)
- Use Case Diagram (Alex)

SQL Database
Define database structure (T)
Generate mock data (T)
Create the database (G)
Host the Database (G)

Website
Static Mock Up (S)
Determine Theme/Palette (S)
Basic HTML to show table for 'feed page' (A)
List/Visual Web of Pages (A)
User Account/Verification (N)
Host the Website (N)

Project Milestone 1

Team Number: 011-3
Team Name: RalphieDevs
Team Members:

- Giovanni Evans
- Trevor Liss
- Alex Scarola
- Spencer Furgerson
- Nick Cervasio

Application Name: Bufflist
Application Description: Bufflist is a multi-use website specifically catered to University of Colorado Boulder students. Bufflist provides a plethora of applications ranging from getting help with odd jobs to buying and selling clothes. This interactive website will work similarly to Craigslist or eBay in that users of the site can post on the main page about their specific needs and can also sift through a stream of other CU Boulder student posts. Additionally, Bufflist will provide a search bar where users can look for specific needs. However, the aspect of Bufflist that sets it apart from typical buying and selling websites is the fact that only CU Boulder students

can use it. Thus, every user on the website will be in the same area and users will not have to worry about shipping or driving far distances to complete a purchase.

Our goal in making Bufflist is that we can provide CU Boulder students, such as ourselves, with affordable prices for things ranging from desks to plumbing services that would not normally be affordable. One aspect of this website that helps toward this goal is that users can trade their own goods or skills for another person's goods or skills. This allows students at Boulder to make purchases without taking any money out of their bank accounts and will even help the overwhelming problem of wasted desks, couches, etc. that litter the town of Boulder.

Vision Statement: "For CU students who need a place to engage with other Buffs, the Bufflist is a common forum where students can trade items or find help from their peers. Unlike trading sites like Craigslist, our product is a platform for students only, where the primary medium of exchange is trade."

Version Control: https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-011-03

Development Method: Jira, Agile Methodology

Because of our fairly small team, our mission is to use a slightly modified, less structured form of agile development. With only five members, we find it unnecessary to use a scrum master or product owner, but do want to take advantage of agile's sprint methodology. Without a S.M. or P.O., it will be even more important for our team to communicate and plan effectively.

The primary steps of our process will include:

- Create our initial product backlog with 10 user stories

- Sprint 1 Planning
- Sprint 1 (Weeks 4-5)
- Retrospective

- Sprint 2 Planning
- Sprint 2 (Weeks 6-7)
- Retrospective

- Release
- Repeat

Based on this plan, we will release a working version of Bufflist at the least every month, which will allow for several iterations on the product.

Our Jira Board: https://csci-3308-fall21-011-03.atlassian.net/jira/software/projects/RD3/boards/1

Communication Plan: Besides Weekly Zoom Meetings, using a text group chat to keep in contact.

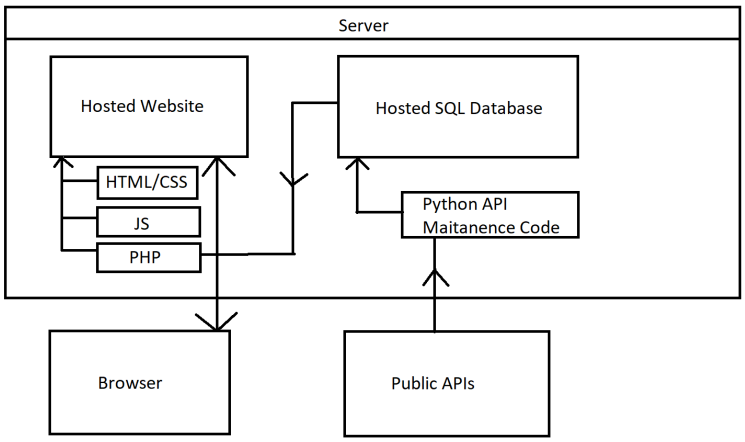
Meeting Plan:

- Weekly Group Zoom Call Mondays 11:30am-1:30pm
- Standup TA Meeting Thursdays 10:30am-10:45am

Proposed Architecture Plan:

The technologies we will be utilizing on the backend consist primarily of an SQL server to hold all of our listing and user information. Currently, we envision utilizing 2 tables, one for login and user information, and one for listing information. We will use PHP code to connect this database with the Front End of our codebase, which will be a Website that uses HTML, CSS, and some javascript to aesthetically present the contents of the database and to allow users to log in and interact with the database, creating and commenting on listings, communicating with each other, and more. We also envision connecting in a few APIs to further the user experience. For example, an OAuth connection to the Identikey service to bring users into the CU account could be a stretch goal, or using Google Maps API to represent locations of listings on campus/in Boulder and give users directions and/or hotspots. These APIs would be implemented and their OAuth maintained by either Microsoft Flows, or, if easier to implement, live javascript or python code.

Architecture Diagram:



Use Case Diagram: