Exceptional Null Pointers Development Plan
Sponsor: AppFolio

Group Members:

- Wade Varesio (Team Lead) - wadevaresio@ucsb.edu
- Kevin Pham (Scribe) - k.pham@ucsb.edu
- Kobe Shavolian - kobeshavolian@ucsb.edu
- Eric Yuan - zichen.yuan@ucsb.edu
- Noah Pang - noahpang@ucsb.edu

Development Plan

➢ Official AppFolio Property Listing Implementation: Implement AppFolio’s given bank of listing information to our website for up-to-date housing/property information. This should allow us to add more features to the table so the user can know more about the property (a description, consistent pricing, etc.).

➢ Clean Up Social Features: Provide a more well-rounded social functionality with more strict access to who has admin control of a group (i.e. who can add/remove other ‘roommates’), etc.

➢ AI recommendation system: Recommend property listings based on personal preferences and roommate group preferences

➢ Add Social Features: Add the ability for group members to send property listings via email and/or text message to the rest of their roommate group.

➢ Rating Functionality: Determine, using some factors, a rating for any one property that compares said factors and provides a result.

Sprint 5:

- Modify search functionality to use AppFolio listing data instead of 3rd party external API’s.
- Build a model for an AI based recommendation system.
- Improve roommate group functionality.
○ Add a “Leave Group” button to easily remove someone from the group.
○ Modify it such that only the group admin (creator) can remove individuals from the group.
○ Add an invitation system such that users are not automatically enrolled into a group.

● Research feasibility of implementing a common application form allowing users to submit leasing applications from the application.

**Sprint 6:**

● Improve user experience in the search results page.
● Add the ability for users to email property listings to their roommate group.
● Add the ability for users to text property listings to their roommate group.
● Optimize the application to run faster via lazy loading and caching.
● Implement an AI based recommendation system.
● Perform second user study for feedback.
● Begin gathering application metrics.
  ○ User engagement/event sequence
  ○ Uptime/average latency

**Sprint 7:**

● Implement feedback from user study in sprint 6.
● Implement bug fixes.
● Bring QA environment up to production head for a backup presentation environment.
● Develop a contingency plan if Amazon AWS goes down for presentation.
● Determine whose computer will be used during the presentation.
  ○ Designate a different individual as a backup.
● Collect any additional application metrics.

**Sprint 8:**

● Stress test application.
● Patch any last minute bugs.
● Plan presentation and rehearse presentation.