Codebase Evaluation

Phase 1

Documentation

- Official Documentation? jsDocs, READMEs, etc. Are there ADRs?
- Is it easy to read and understand?
- Were Github issues used appropriately?
- Are there design and architecture diagrams to provide a high level and detailed project view?

Deployment/Pipeline/DevOps

- Is there a present and working CI/CD pipeline? Documentation?
- Simple deployment process? Is the deployment automatic?
- Unit Tests? Integration Tests? Are these tests comprehensive?
- Easy to update or make changes to the pipeline?

Design

- Intuitive and easy for the first time user?
- Responsive if intended for mobile and web?
- Is the UI sturdy? Does it break under user input?
- Is the UI consistent across the application?
- Does the UI allow for efficient use of the application?

Performance/Security

- How optimized is the application? Lighthouse report? Time till interaction? Mobile speeds?
- Any vulnerabilities? User inputs sanitized?
- Is the scrolling smooth, the buttons clickable and change quickly, and does the application load within 2 seconds?

Code

- Is the code mostly consistent?
  - Consistent naming conventions
- How much spaghetti code?
  - Is the code all over the place?
  - Is the code readable and easy to follow
- Modular Design or any design?
- Function sizes neither too big nor too small?
- Does the organization of the files/project consistent and make sense?

Rating System

Criteria are weighted, with 1 being the most important, and 5 being the least important.

This system is put in place such that projects missing important criteria like CI/CD or deployment pipelines are not automatically taken out of consideration.

Each of these areas will be rated out of 5 stars, with 5 being the highest and 0 being the lowest

1 **Code**

Consistency, Modularity

2 **Documentation**

Is there enough to go off of

3 **Design**

Simple enough for the first time user

4 **Performance**

Reasonably efficient

5 **Deployment**

Simple and automated