

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Kaiwen Tsou

Name of team member whose repo you were assigned \*

James

Link to the repo \*

<https://github.com/dnanjunda/cse110-w21-group11>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

Pomodoro timer using basic web tech (vanilla JS, html/css)

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the

project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

- Yes
- No
- Other: Unknown where the Kanban/Project board is, as there is no GitHub Projects board

Briefly describe the file structure of the repo in terms of organization and simplicity \*

Organized. All source files are in source, and specs are in specs. Nothing unusual in terms of file structure.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

None

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code strucutre, the CI/CD pipeline, testing, etc)? \*

Lack of onboarding documents. The wiki has rough function documentation, and roughly formatted documentation for the build pipeline. There is another .md file on the pipeline in /admin/documentation.

Lack of info in ADRs. No designated architecture diagram.

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

Jest, ESLint, Prettify, JSDoc,

How many ADRs do they have? \*

3

Do they have C4 Diagrams or something similar? \*

Yes

No

Other: \_\_\_\_\_

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

2 open issues, 25 closed issues. Only uses default labels. No templates

Do they have a SCRUM board or something similar? \*

Yes

No

Other: \_\_\_\_\_

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

.....

### User Centered Thinking

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

8

.....

Briefly describe the quality of the user stories \*

Organization is not very consistent, hard to read due to formatting. Overall, these don't seem too helpful in establishing a potential user base or telling us the potential audience.

How many personas are there? \*

8

Briefly describe the quality of the personas \*

Same as user stories.

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other:

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other: Not many ADRs, but the few that exist briefly mention user experience.

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other:

There are some diagrams in the pitch, but they are not very high quality. The interface design mockup closely resembles the final product, however..

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other: There is an ADR detailing the agreed upon core features of the product.

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

There is no future roadmap/features to be added.

Overall, how would rate the user centered thinking of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

---

## Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

There are 3 ADRs, one for core components, one for how the early pipeline will work, and one for which visual design is chosen. They are well formatted, however there are no other ADRs for anything, so many architectural decisions are not documented at all.

Is there a roadmap detailing what the original team wanted to implement? \*

- Yes
- No
- Other: The core-components ADR includes a single item that may be included in the future.

Are known bugs and issues documented properly, or are they obfuscated? \*

Yes

No

Other:

No bugs or major issues are noted, however it seems the program is sparsely tested so something will likely show up.

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

Not as far as I can tell.

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other:

There might be something in the status video, but besides the 25 closed issues, there is no written documentation on the history of the project.

Overall, how would rate the transparency of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the transparency of the project?

---

### Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: While it's not a well documented system, it is still a relatively simple project.

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

James Larsen

Name of team member whose repo you were assigned \*

Zhuoliang Pu

Link to the repo \*

<https://github.com/cse110-sp21-group34/cse110-sp21-group34>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

Web app (Vanilla JS), bullet journal

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

Yes

No

Other: \_\_\_\_\_

Briefly describe the file structure of the repo in terms of organization and simplicity \*

First the positives. The root of the repo is very clean. There's a clear delineation between CICD in the github workflows directory, the admin directory, the spec directory, and the source directory. So the root directory gets an A+.

Once you go a level or two down it gets much more confusing though. The delineation between what goes in admin vs what goes in spec is unclear. Why the specifications of the CICD pipeline are in the admin folder is weird.

The source folder is where it gets really hard to understand. There are multiple index.html files at different levels buried in the source. It's also hard to trace what files are inheriting from other files. Also things like audio and image dependencies are buried in editor rather than utils. CSS is scattered around between different directories. There are entire dependencies buried under several layers that are entire repos dropped in. It's hard to know where to look for stuff unless you already have gotten a sense of the repo and dug around in it.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

Uses babel, editorjs, Simple Image Tool, clipboardy, parcel, puppeteer, and possibly others that I missed

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code structure, the CI/CD pipeline, testing, etc)? \*

There are instructions on how to launch the app from a repo and test the build pipeline. There are minimal JSDocs, but the docs do give at least a sense of what's going on. The code has variable amounts of comments, not usually enough to know what's going on.

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

ESLint, Jest, and JSDocs

How many ADRs do they have? \*

10

Do they have C4 Diagrams or something similar? \*

Yes

No

Other: Yes, but they look out of date and don't reflect the current state of the app

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

37 issues, most are untagged. There seems to be minimal organization of the Issues.

Do they have a SCRUM board or something similar? \*

Yes

No

Other: .....

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

.....

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

0

Briefly describe the quality of the user stories \*

N/a

How many personas are there? \*

3

Briefly describe the quality of the personas \*

The personas certainly have a nice presentation. They go into detail about all three of their users. They go into details about what their frustrations are and what their core needs are.

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other:

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other: .....

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other: .....

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other: Yes, there is a feature list but it doesn't have much to say about the user experience

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

They have an initial project proposal which seems to have a general roadmap to how they wanted to approach the project. The features don't have any particular ranking of importance from what I could see.

Overall, how would rate the user centered thinking of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

I feel like there wasn't much recorded user centered thinking during the development of this project. I feel like it bleeds into the actual UI/UX as well, where it is confusing how to do simple tasks like make the text bold.

### Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

The ADRs are over pretty minor decisions. The ADRs are also fairly minimal in terms of the discussion surrounding them. They feel like they were backfilled. No followup was recorded.

Is there a roadmap detailing what the original team wanted to implement? \*

Yes

No

Other:

Are known bugs and issues documented properly, or are they obfuscated? \*

Yes

No

Other:

There are documented Issues, but there seem to be a number of bugs that aren't documented.

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

They cut a lot of features, which makes the system diagrams very out of date. JSDoc is up to date of course and accurately documents what functions they've commented.

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other:

Overall, how would rate the transparency of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the transparency of the project?

I don't feel like I can list this as adequate when there are unreported bugs I run into when using the bullet journal (text options disappearing/not following the cursor or text). It's also unclear how and when features were cut as well as the reasoning for how they limited the scope of the project. The fact that there are buttons which don't do anything and I can't find documentation on them is what makes it hard for me to give this an adequate.

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: \_\_\_\_\_

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other:

Unconfident, I imagine we'd spend a lot of time struggling with the different dependencies of the project as well as the structure of the current repo

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Chad Wall

Name of team member whose repo you were assigned \*

Kaiwen Tsou

Link to the repo \*

<https://github.com/cse110-fa21-group2/cse110-fa21-group2>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

The project is a recipe management web app.

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

Yes

No

Other: \_\_\_\_\_

Briefly describe the file structure of the repo in terms of organization and simplicity \*

\_\_tests\_\_ holds the testing functionality, admin holds, misc. admin work, out stores JSDocs output, source stores source files, and specs store ADRs. Most files were arranged adequately in sub-directories, but there were some miscellaneous files in directories which had no sub-directory to further define them. Additionally, the structure of directories was a bit off, as it would sometimes be based on the semantics of the directory (i.e. meetings directory) and other times it would be based on what type of files were in it (ie videos).

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

spoonacular API, Bootstrap, and Netlify

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code structure, the CI/CD pipeline, testing, etc)? \*

There seems to be good documentation on GitHub etiquette (branches, issues, pull requests, project board etc) and some documentation on how to get started as a programmer, but not much in-depth documentation on the pipeline and how to create unit tests/e2e tests.

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

Github, Github Actions, Babel, Jest, Puppeteer, ESLint and JSDocs

How many ADRs do they have? \*

7

Do they have C4 Diagrams or something similar? \*

Yes

No

Other: \_\_\_\_\_

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

126 (9 open); Some here and there had a structure to them but for the most part they were not organizaed

Do they have a SCRUM board or something similar? \*

Yes

No

Other: \_\_\_\_\_

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

Most of the important documents were there, but the wiki was lacking in information and it was hard to get a sense of what was and was not being used. Fair amount of documentation on the meta info of the project (issues, project board, branches, stories, etc) but not so much on the actual project (System diagrams, pipeline info, testing info, dependency/tool usage was a bit of a scavenger hunt to fully figure out).

## User Centered Thinking

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

1

Briefly describe the quality of the user stories \*

It isn't so much a user story more so it is an emporium of possible use cases and circumstances which play into what the user would do on the website.

How many personas are there? \*

12

Briefly describe the quality of the personas \*

The user stories are well made as they not only create a persona, but they talk in detail about how those persona

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other: \_\_\_\_\_

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other:

Mostly no, there are some instances where they reference their user story when talking about why they did something but generally outcomes are based on implementation feasibility.

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other: .....

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other:

There are some documents on extra features, mostly in the project pitch, but for the most part the extra features aren't mentioned too much and are not well documented. When they are documented, there is no elaboration on them.

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

Roadmap is briefly mentioned in their project pitch, but it does not have a whole lot of structure.

Overall, how would rate the user centered thinking of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

The user stories and persona was really fleshed out, but feature lists, roadmaps, and integration of the user needs into the project is not well documented.

## Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

Some major decisions are indeed documented in ADR (bootstrap, storage, design, timer, login, etc), and they do go into some detail on the problem, solution, pro and cons, and decision, but they are rather undetailed. Additionally there is no followup with the consequences of their actions, In short, the ADRs are not in depth.

Is there a roadmap detailing what the original team wanted to implement? \*

Yes

No

Other:

The only real mention of a road map seems to be in the project pitch, and even then it is sparse and not fleshed out.

Are known bugs and issues documented properly, or are they obfuscated? \*

Yes

No

Other:

There are some glaring bugs on the website, mainly the connection between it and the spoonacular api, however this was most likely not an issue while they were working on the project. That said, there seems to be little to no documentation on the system so it would be hard to look into this issue.

Though as a plus, they do have 9 open issues which can act as starting points for improvement

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

Not exactly. The repo's info is not deceptive/performative as what it states its pretty accurate to what is on the repo. Especially the issues seem to have a large breadth of documentation. The issue is more so that there isn't much depth in the documentation of the project.

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other: .....

Overall, how would rate the transparency of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the transparency of the project?

An issue with transparency, though more so the maintainability, is that there isn't much documentation on how the website interacts with the spoonacular api. This is an issue because the spoonacular api is no longer working with the website which may cause major issues with revitalizing the project.

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Lars

Name of team member whose repo you were assigned \*

Dzhangir

Link to the repo \*

<https://github.com/Dzhang0/RecipeHunter>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

This is a recipe manager that focuses on fast and easy recipes. It allows users to find recipes based on searches, or by ingredients they have on hand, and allows them to save recipes they find through the app as well as recipes they find elsewhere on the internet. It is a web application.

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

- Yes
- No
- Other: I don't know if there is another place that

Briefly describe the file structure of the repo in terms of organization and simplicity \*

The repo has 3 main folders: admin, source and specs. The admin folder contains branding materials, meeting notes, team rules and videos. The source folder contains assets, as well as folders for each of the four pages of the application. Each page folder contains the markup, scripts, styles, and tests for each page. The organization within each page folder is a little confusing and inconsistent as some pages are not stored within sub-folders, and two page folders contain coverage reports while the other two don't. The specs folder contains adrs, brainstorming, and ci pipeline documentation. There is a folder for documentation of the code itself, but it is empty. If there is documentation anywhere else, I can't find it.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

Bootstrap, Spoonacular API, IndexDB API

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code structure, the CI/CD pipeline, testing, etc)? \*

The main README gives instructions for how to install the repo and push a change. The CI/CD pipeline seems well-documented in the specs folder, but I can't find any documentation on code structure.

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

SuperLinter, CodeFactor, Jest, JSDocs

How many ADRs do they have? \*

6

Do they have C4 Diagrams or something similar? \*

Yes

No

Other:

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

0 open. 44 closed. Issues do not use templates. Many don't have any description at all. They do use labels, but there do not appear to be branches or pull requests associated with closing the issues.

Do they have a SCRUM board or something similar? \*

Yes

No

Other: .....

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

I don't think I had trouble finding anything, but there is definitely a lack of information. I was not able to tell if there were additional dependencies used.

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

0

Briefly describe the quality of the user stories \*

N/A

How many personas are there? \*

0

Briefly describe the quality of the personas \*

N/A

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other: \_\_\_\_\_

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other:

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other:

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other:

There is a user manual that outlines all of the features of the application and goes through how to use each one in depth. In the brainstorming, there is a board of features ranked by importance, though there is no indication of which features have been implemented.

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

Two features are on the roadmap. There is not much explanation for either one.

Overall, how would rate the user centered thinking of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

There is thought given to who the target audience of this product is, and how this application would address their particular user-base. Evidence of this can be found in the brainstorming section of the repo. However, it is hard to find formalized artifacts of user centered design (things like user stories and personas), which is why I gave it an "unsatisfactory."

## Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

ADRs document major decisions. They do include problem, solutions, pros and cons, and eventual decision. Not all ADRs have followups of the consequences, but some do.

Is there a roadmap detailing what the original team wanted to implement? \*

- Yes
- No
- Other: There are two features listed as the roadmap. Not much information is provided.

Are known bugs and issues documented properly, or are they obfuscated? \*

- Yes
- No
- Other: .....

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

There are definitely bugs on the application itself that are not reflected in the GitHub repo. However, it is hard to tell which bugs existed at the time, and which have developed since the team stopped working on the project. It is hard to tell how issues were addressed, because they were not associated with particular branches, and many were opened and closed without comment.

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other: .....

Overall, how would rate the transparency of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the transparency of the project?

I think the main issue is that there is a lack of information in regards to the issues, as well as a lack of documentation. While the ADRs provide some nice high level documentation of how problems were addressed, it would be very difficult to figure out what is currently broken and how the details of problems were addressed.

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Fenghao Yang

Name of team member whose repo you were assigned \*

Calvin Chen

Link to the repo \*

[https://github.com/xdsherman/cse110\\_Time2Meet](https://github.com/xdsherman/cse110_Time2Meet)

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

This repo is a web application that is similar to the whentomeet app. User is able to find the aviliable time to meet in a group using this application and arrange the meeting accordingly. This repo uses reactJs as the frontend and firebase as the backend.

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

- Yes
- No
- Other: It misses documentation but everthing else is in the repo

Briefly describe the file structure of the repo in terms of organization and simplicity \*

This repo has two major folders. One is src which contains all the frontend and UI of this web application. The other one is .firebase which contains the backend stuff and database for this app. There are also jsons file in the home directory that set up the enviornment for both front and back end.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

React as frontend, firebase as backend

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code strucutre, the CI/CD pipeline, testing, etc)? \*

I dont see any documents in this repo

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

N/A

How many ADRs do they have? \*

0

Do they have C4 Diagrams or something similar? \*

Yes

No

Other: \_\_\_\_\_

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

The lack of documentation makes it hard for me to track the process of the development. Other than that, the organization of the file looks good to me.

Do they have a SCRUM board or something similar? \*

Yes

No

Other: \_\_\_\_\_

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

.....

### User Centered Thinking

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

8

.....

Briefly describe the quality of the user stories \*

Most of the user stories in this repo is completed with only minor issues

How many personas are there? \*

1

Briefly describe the quality of the personas \*

The quality is quite good for personas

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other:

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other:

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other: .....

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other: .....

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

They do not have a documentation to describe the roadmap

Overall, how would rate the user centered thinking of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

It is hard to judge the user centered thinking of this project without the documentation. But from the repo and the application itself, I think the user centered thinking aspect is satisfactory and we can make some adjustments in the future to make it even better.

## Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

There are no documents in the repo

Is there a roadmap detailing what the original team wanted to implement? \*

Yes

No

Other:

Are known bugs and issues documented properly, or are they obfuscated? \*

Yes

No

Other:

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

There are no documents in the repo, but the readme shows the connection between the repo and the application itself.

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other: \_\_\_\_\_

Overall, how would rate the transparency of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the transparency of the project?

.....

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

Confident

Iffy

Unconfident

Other: \_\_\_\_\_

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

Confident

Iffy

Unconfident

Other: \_\_\_\_\_

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Calvin Chen

Name of team member whose repo you were assigned \*

Jaime Izabal

Link to the repo \*

<https://github.com/JIzabal/tag.it>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

Piazza-like bulletin board web app

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

- Yes
- No
- Other: Artifacts aren't accessible from the repo. They might exist on Jaime's Google drive

Briefly describe the file structure of the repo in terms of organization and simplicity \*

Code is cleanly separated into two folders, client and backend. Backend folder is further separated into model, controller subfolders. Firebase related code files are also separated. Client folder follows the React default directory hierarchy. Very simple and clean structure.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

Firebase (probably for managing and verifying user accounts), React is used on the front end. Other dependencies can be found in the yarn file, including Babel, Wrappy(?), Popper js(?), etc...

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code structure, the CI/CD pipeline, testing, etc)? \*

Documents not included in the repo but they exist as "artifacts" as Gary called them. Jaime should have them somewhere... I see the repo includes Travis which is a form of CI.

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

Travis CI

How many ADRs do they have? \*

0

Do they have C4 Diagrams or something similar? \*

Yes

No

Other:

He should have something called a screen sequence diagram which may be Gary's equivalent of C4 diagrams

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

Just 1 default master branch

Do they have a SCRUM board or something similar? \*

- Yes
- No
- Other: Jaime should still have access to the project management website at cse110.com

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

- Yes
- No
- Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

0

Briefly describe the quality of the user stories \*

N/A, Jaime should have a User Stories artifact with this info

How many personas are there? \*

0

Briefly describe the quality of the personas \*

N/A

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other: Jaime should have a User Cases artifact with this info

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

- Yes
- No
- Other: No ADRs

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

- Yes
- No
- Other: Jaime should have a Sample Screens artifact with this info

Is there a feature list which focuses on the usability and accessibility of the product? \*

- Yes
- No
- Other: Jaime should have a Design Use Cases artifact with this info

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

No roadmap

Overall, how would rate the user centered thinking of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

Can't answer most of these questions without Jaime's artifacts

### Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

N/A

Is there a roadmap detailing what the original team wanted to implement? \*

- Yes
- No
- Other:

Are known bugs and issues documented properly, or are they obfuscated? \*

- Yes
- No
- Other: On cse110.com, Jaime should have a record of bugs/issues

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

N/A

Are there documents which highlight the challenges and struggles members faced along the way? \*

- Yes
- No
- Other:

Overall, how would rate the transparency of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the transparency of the project?

Can't answer most of these without Jaime's artifacts

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: \_\_\_\_\_

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: \_\_\_\_\_

This form was created inside of UC San Diego.

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Jaime Izabal

Name of team member whose repo you were assigned \*

Calvin Chen

Link to the repo \*

[https://github.com/xdsherman/cse110\\_Time2Meet](https://github.com/xdsherman/cse110_Time2Meet)

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

Firebase, React, HTML, CSS

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

- Yes
- No
- Other: There are artifacts associated with the repo, but I don't think they cover everything

Briefly describe the file structure of the repo in terms of organization and simplicity \*

The program is broken down into auth, components, and routes. I'm not sure if I see any tests.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

React dependencies (e.g., react-day-picker, react-dom, react-multi-email, react-scripts, web-vitals),  
Firebase

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code strucutre, the CI/CD pipeline, testing, etc)? \*

No onboarding document

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

@testing-library/jest-dom, @testing-library/react, @testing-library/user-event, ESLint, npm

How many ADRs do they have? \*

0

Do they have C4 Diagrams or something similar? \*

Yes

No

Other: .....

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

No use of GitHub Issues

Do they have a SCRUM board or something similar? \*

Yes

No

Other: .....

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

.....

### User Centered Thinking

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

9

.....

Briefly describe the quality of the user stories \*

They're good but they all don't give a specific role (e.g., All use "As a user")

How many personas are there? \*

0

Briefly describe the quality of the personas \*

No personas

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other:

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other:

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other: .....

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other: .....

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

There's a list of features with priorities and they consider the user experience, however there are only two features that aren't done.

Overall, how would rate the user centered thinking of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

---

## Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

No ADRs

---

Is there a roadmap detailing what the original team wanted to implement? \*

- Yes
  - No
  - Other:
- 

Are known bugs and issues documented properly, or are they obfuscated? \*

- Yes
  - No
  - Other:
-

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

No

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other:

Overall, how would rate the transparency of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the transparency of the project?

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Zhuoliang Pu

Name of team member whose repo you were assigned \*

Chad

Link to the repo \*

<https://github.com/Zihaokong/cse110-w21-group27>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

A web based Pomodoro timer

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

Yes

No

Other: .....

Briefly describe the file structure of the repo in terms of organization and simplicity \*

Generally very well structured repo. Design and decision documents are well stored in "specs". "Admin" contains meeting notes and slides that help keep track of their progress. Source code were stored in "source" and everything were well divided into components that are easy to locate. Tests seem to be divided into 2 portions "\_test\_" and "cypress", not sure if it is some existing standard but I feel it may be better to combine the two. But still, overall speaking very well organized and pretty simple to locate things.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

Bootstrap .....

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code structure, the CI/CD pipeline, testing, etc)? \*

They have detailed documentations on code structure, pipeline, testing and design on their wiki page. Have documentations on how to run test locally, but doesn't seem to have one that simply tells you how to deploy the project locally. Also I couldn't find easily accessible documentations on the features and usage of the actual product.

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

"cypress" & "jest" for testing, "husky" for git commit, "jsdoc" for auto code documentation, "eslint" and "prettier" for code analysis and format.

How many ADRs do they have? \*

9

Do they have C4 Diagrams or something similar? \*

Yes

No

Other:

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

3 open (which is for future improvements) and 29 resolved. They look well structured and there are labels indicating priority and estimated difficulties.

Do they have a SCRUM board or something similar? \*

Yes

No

Other: .....

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

.....

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

4

Briefly describe the quality of the user stories \*

Have good points and very related to the project.

How many personas are there? \*

8

Briefly describe the quality of the personas \*

Well thought and covers wide range of users.

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other:

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other: .....

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other: .....

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other: .....

Not documented but there are 2 brought up on the issue lists, and seems to have one aiming for accessibility in the actual product (question mark icon)

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

Design-> task list CRUD-> add working timer-> add statistics tracking They have a feature list during brainstorm stage, but other than that I couldn't find anything else.

Overall, how would rate the user centered thinking of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

#### Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

Yes.

Is there a roadmap detailing what the original team wanted to implement? \*

Yes

No

Other: .....

Are known bugs and issues documented properly, or are they obfuscated? \*

Yes

No

Other: .....

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

Not really. ....

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other: .....

Overall, how would rate the transparency of this repo? \*

- Satisfactory
- Adequate
- Unsatisfactory

Any additional comments in regards to the transparency of the project?

---

### Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: \_\_\_\_\_

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

Confident

Iffy

Unconfident

Other: .....

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Sohum Modha

Name of team member whose repo you were assigned \*

James

Link to the repo \*

<https://github.com/dnanjunda/cse110-w21-group11>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

Web app. It's a Pomodoro timer coded mostly in JS.

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

Yes

No

Other: .....

Briefly describe the file structure of the repo in terms of organization and simplicity \*

The majority of the code is in "node\_modules," and it's well organized. The majority of team information is in "admin," and the development process is in "specs"

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

Express (source was James, not the repo. Was not sure where to get this info)

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code structure, the CI/CD pipeline, testing, etc)? \*

No .....

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

ESLint .....

How many ADRs do they have? \*

3

Do they have C4 Diagrams or something similar? \*

Yes

No

Other:

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

I don't see any issues. The files are well organized into folders.

Do they have a SCRUM board or something similar? \*

Yes

No

Other:

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

No .....

### User Centered Thinking

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

0 .....

Briefly describe the quality of the user stories \*

I think that there are personas and no user stories. In my CSE 110 class, a user story was of the form "As a user, I want -- so that --" and I didn't see anything like that in the repo.

How many personas are there? \*

8

Briefly describe the quality of the personas \*

There are 8 personas and each of them are very detailed. There are sticky notes for different view points and details for each persona.

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other: \_\_\_\_\_

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other: \_\_\_\_\_

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other: .....

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other: .....

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

No ranking of features, but there was a list of accepted and rejected app components

Overall, how would rate the user centered thinking of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

No

## Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

Yes. There are 3 ADRs and they are very detailed.

Is there a roadmap detailing what the original team wanted to implement? \*

Yes

No

Other:

Are known bugs and issues documented properly, or are they obfuscated? \*

Yes

No

Other:

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

No

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other:

Overall, how would rate the transparency of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the transparency of the project?

No

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

This form was created inside of UC San Diego.

Google Forms

# CSE 112 Team Cre8tors Project Evaluation Form

This form will be used to evaluate the project you were assigned to evaluate. This evaluation should take somewhere between 60 to 90 minutes so be sure to set some time aside to work on the evaluation! The evaluations are technically due Tuesday (4/19) at 11:59PM, so these responses should be finished by 04:00PM on Tuesday (4/19) so that we can congregate and submit the responses.

Your Name \*

Dzhangir Bayandarov

Name of team member whose repo you were assigned \*

Lars

Link to the repo \*

<https://github.com/cse110-sp21-group13/cse110-sp21-group13>

Briefly describe what the project is and what technologies are used (i.e. is it a web app, android app, something else). \*

Bullet Journal Web App that mimics pen&paper journaling style

Simplicity & Organization

A repo should be simplistic and organized because it allows us to leverage more time into developing and improving the project instead of having to spend large amounts of time getting acquainted with the system

Does the GitHub act as the single source of truth (meaning is all the information pertaining to the project created found in the repository)? \*

Yes

No

Other: .....

Briefly describe the file structure of the repo in terms of organization and simplicity \*

admin, source, specs folders. Very simple folder structure.

Briefly list what dependencies are used (by dependencies we mean things which are directly connected with the product like Bootstrap, React, etc). \*

PouchDB, NodeJS(Express), JSDocs

Are there on boarding documents? If so, are they simple and easy to understand? How much details do they go over in these documents (i.e. do they mention the code strucutre, the CI/CD pipeline, testing, etc)? \*

There are some descriptions of the pipeline and overall structure but no clear onboarding documentation. The diagrams are rather convoluted and the descriptions are a wall of text rather than concise instructions

Briefly list what tools are used in the project (by tools we mean things which are used in support of the product like ESLint, Jest, etc). \*

ESList, Jest, Pupeteer, Lighthouse reports, Codacy.

How many ADRs do they have? \*

11

Do they have C4 Diagrams or something similar? \*

- Yes
- No
- Other: Only for the CICD pipeline

How many issues do they have? Are they well structured and organized (ie templates, labels, associated branches, etc)? \*

0 open issues. 90 closed issues neatly organized.

Do they have a SCRUM board or something similar? \*

- Yes
- No
- Other:

Does the code have an enforced standard (i.e. a linter or some documentation on coding standards)? \*

Yes

No

Other: .....

Overall, how would rate the simplicity and organization of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the simplicity and organization of the project?

.....

### User Centered Thinking

There is evidence of the previous developers designing this product with users in mind (e.g., User stories, personas). General user experience is fundamental to the goals that the previous developers set up.

How many user stories are there? \*

3

.....

Briefly describe the quality of the user stories \*

They are sophisticated and cover the broad range of usability case.

How many personas are there? \*

3

Briefly describe the quality of the personas \*

The personas cover different age groups and distinguish between the core usability.

Do they have flow charts detailing the use cases from the perspective of the user? \*

Yes

No

Other:

Do the ADRs consider and/or account for the user experience (i.e. accessibility features, performance, etc)? \*

Yes

No

Other:

Are there user focused diagrams like wire-frame and high fidelity mock-ups? \*

Yes

No

Other: .....

Is there a feature list which focuses on the usability and accessibility of the product? \*

Yes

No

Other: .....

Briefly describe the roadmap of the project and if there is documentation on a ranking of features in regards to the user experience. \*

There is a pitch document that describes the core functionality and how it would affect the user experience.

Overall, how would rate the user centered thinking of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the user centered thinking of the project?

.....

## Transparency

Documentation of both the code as well as the documentation process. This would capture the pitfalls past developers fell into. This would help us prevent wasting time discovering issues that could have been documented for us.

Do the ADRs document major decisions? Do these ADRs include the problem, solution, pros and cons, and an eventual decision and a followup of consequences? \*

Yes, moderately, the ADRs do not go into depth but prompt a strong foundation for the functionality of the app.

.....

Is there a roadmap detailing what the original team wanted to implement? \*

- Yes
  - No
  - Other:
- .....

Are known bugs and issues documented properly, or are they obfuscated? \*

- Yes
  - No
  - Other:
- .....

After looking at both the product as well as the GitHub repo, is there a disconnect between what is documented and what is presented? \*

I don't think there is disconnect but I do think the project appears more simple than its implementation.

Are there documents which highlight the challenges and struggles members faced along the way? \*

Yes

No

Other: .....

Overall, how would rate the transparency of this repo? \*

Satisfactory

Adequate

Unsatisfactory

Any additional comments in regards to the transparency of the project?

.....

## Constraints

For each value, we will also be keeping in mind the constraints that will inevitably show up as we work on the project.

Given the skills needed for the repo, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

Given the time we calculated in the evaluation slides, how confident are you that we can get this project into a "1.0" level by the end of the quarter \*

- Confident
- Iffy
- Unconfident
- Other: .....

This form was created inside of UC San Diego.

Google Forms