# Jonathon Hull

| Category | Reviewers Comment | Action taken |
|---|---|---|
| **Build** | I was able to clone but the npm install failed with a wierd error. It seams to be my OS not liking something<br><br>From what I heard it went fine if you don't get that error | This is due to an error with versioning of Node and NPM. Our project uses 10.15.3 and 6.4.1 respectively; more recent versions conflict with version 7.17.1 of pg, the node-postgres package we use. Those running MacOS were more likely to have Node 14, which did not work; not everyone on MacOS had Node 14 and those with earlier versions succeeded. We decided not to update all of these packages because support for them extends past the life of this project and there are no vulnerabilities associated with them. We added the versions to the README to avoid confusion. |
| **Legibility** | Variable names and methods were easy to follow<br><br>There were some variable naming discrepancies. Like some are using SW some are using ServiceWorker | These discrepancies were caused in files generated by create-react-app; we decided to leave them because we did not write them in the first place. |
| **Implementation** | No | No action necessary |
| **Maintainability** | Unit tests were there. | No action necessary |
| **Requirements** | Yes | No action necessary |
| **Other** | Build failed for many people on different platforms.<br>Build failed on Windows and on Mac | See above comment for build section. |

## Zhidong Zhang

| Category | Reviewers Comment | Action taken |
|---|---|---|
| **Build** | It can clone from Git, but it can't build. | See the response for the "Build" section of Jonathon's feedback. |
| **Legibility** | Yes, it look cleanly for variable name. | No action necessary |
| **Implementation** | It is cleaner to read the code. It seem well | No action necessary |
| **Maintainability** | No, it doesn't have unit test. | It does have unit tests, in both the front end and the back end. |
| **Requirements** | Yes, It does | No action necessary |
| **Other** | Seem well. Nothing can improved | No action necessary |

## Ulises Zaragoza

| Category | Reviewers Comment | Action taken |
|---|---|---|
| **Build** | Yes, I was able to easily clone the Repo and build using the README. It was straightforward and easy to follow. | No action necessary |
| **Legibility** | Yes, the code has a clear flow and easy to follow with clear variable and method names. The code adheres to general guidelines and is a clear style that shows a solidly implemented project. | No action necessary |

| Implementation | I do not think it would be any cleaner/faster/easier to write functionally equivalent code. I think the code as it stands is clear and I don't see any clear refactoring that is needed. | No action necessary |
|---|---|---|
| Maintainability | There are unit tests, and they seem to be covering interesting connection and API endpoint cases. They are indeed readable and clear as to what they are trying to accomplish. | No action necessary |
| Requirements | Yes, the requirements called for exactly what was produced in the project. It is clearly a complete project that has accomplished what it sought to complete. | No action necessary |
| Other | The only thing I can think can stand out is the website UI. It is a little plain, but does allow for full utilization of the project components. | As software engineers, we designed our UX for efficiency and ease. As a result it may look a little plain, but we don't feel we can do anything to help that. |

## Nic Davies

| Category | Reviewers Comment | Action taken |
|---|---|---|
| Build | Could not the build. I think it might be a node version or OS issue, but I had an error when I was trying to run "npm install". Seems as if this project is not very Mac OS friendly. It is a local problem that doesn't need to have action for the group. | See the response for the "Build" section of Jonathon's feedback. |
| Legibility | Code looks clean. Variable names are good and there is | No action necessary |

| | | |
|---|---|---|
| | plenty of commenting and they included headers for functions. | |
| **Implementation** | Not that I can tell, I am not a JS expert and would be out of my area of expertise recommending anything here. I cannot see any useful abstractions and would actually need to be able to run the code in order to understand all of the moving parts. | No action necessary |
| **Maintainability** | There are unit tests. They should exist and the client apparently required them to be written as well. They are readable, they are testing the database and the data inside it. | No action necessary |
| **Requirements** | From what I read from the doc, the code fills the requirements. | No action necessary |
| **Other** | N/A | No action necessary |

## Shihao Song

| Category | Reviewers Comment | Action taken |
|---|---|---|
| **Build** | Yes, clone from Git is fine, but cannot build using the README file | See the response for the "Build" section of Jonathon's feedback. |
| **Legibility** | Code is clean and readable, have clear comment and var name. | No action necessary |
| **Implementation** | I think they did API very well. | No action necessary |
| **Maintainability** | There are tests for the client and API. | No action necessary |

| Requirements | Yes, the code is fill the requirement. | No action necessary |
| :---: | :--- | :--- |
| **Other** | N/A | No action necessary |

# Shane Clancy

| *Category* | *Reviewers Comment* | *Action taken* |
| :---: | :--- | :--- |
| **Build** | Was able to build the code just fine. Other team members seemed to have issues though. | See the response for the "Build" section of Jonathon's feedback. |
| **Legibility** | API and client were clear to understand. API could have been abstracted a bit more into different files but integrates with database just fine. | The API routes are all in a single file because there is a single workflow in the front end. Typically, APIs are abstracted by functionality in the same way. Should our project have included other interfaces (for instance, an administrative one for uploading new videos) we would have made another file for API routes. Therefore, we don't believe any action is necessary. |
| **Implementation** | I think the API can be abstracted a bit more to interact with database, but it only has a few endpoints so doesn't necessarily need to be changed. The abstractions made in the web client are very good and make lots of sense. | See the previous comment for API abstraction. Otherwise, no action is necessary. |
| **Maintainability** | There are tests for the client and API. I think only the API needs tests, and it has those. They are easy to execute and test the endpoints. | No action necessary |
| **Requirements** | Yes, the code is able to | No action necessary |

|  | crowdsource the video classification of short clips of video from users. |  |
|---|---|---|
| **Other** | Compatibility on all machines would be what to aim for, since some of my team members were unable to run the project due to npm errors. | See the response for the "Build" section of Jonathon's feedback. |