# Passion Fruits Code Review

## 

[**Passion Fruits Code Review**](#_nwukdfwkvkjy) **1**

[Positive points](#_72fbq7mbuerl) 2

[Included config files like, .dockerignore, .gitignore and others essential to keep the project lean and easy to run](#_f3romt7avsxa) 2

[Clear status updates through the readme.MD](#_ojo05rkflkb) 4

[Clear tracking of features through requirements.MD](#_z5hunifw2qpi) 5

[Use of sidekiq for background jobs makes the app more scalable and enables it to process data in an async way improving the general app performance](#_fopsyxkrz9b5) 6

[Modular Design models, jobs, services, and controllers, following Rails conventions which makes the codebase easy to navigate.](#_t5lh5d92myli) 7

[Graphql integration makes it easier to consume the app from the frontend](#_uhg8dri37zrh) 8

[Error Handling](#_dpcjsh6hrljv) 8

[Shopify App Best Practices](#_z4hhuevigwfw) 8

[Points that need improvement](#_n4u9i6n8tb6s) 9

[overall the app lacks testing, adding that will make it easier in the future to add new features without introducing regression bugs](#_t7ysc5rv9fw7) 9

[Some values are hardcoded inside the code and it would be better to read them from a config file](#_ygm9c475cwp9) 10

[Improve state management to avoid props drilling](#_nrveiubzkyo8) 12

[Lack of clear guide on how to run the app and more technical documentation on readme.md](#_x012yep721mn) 15

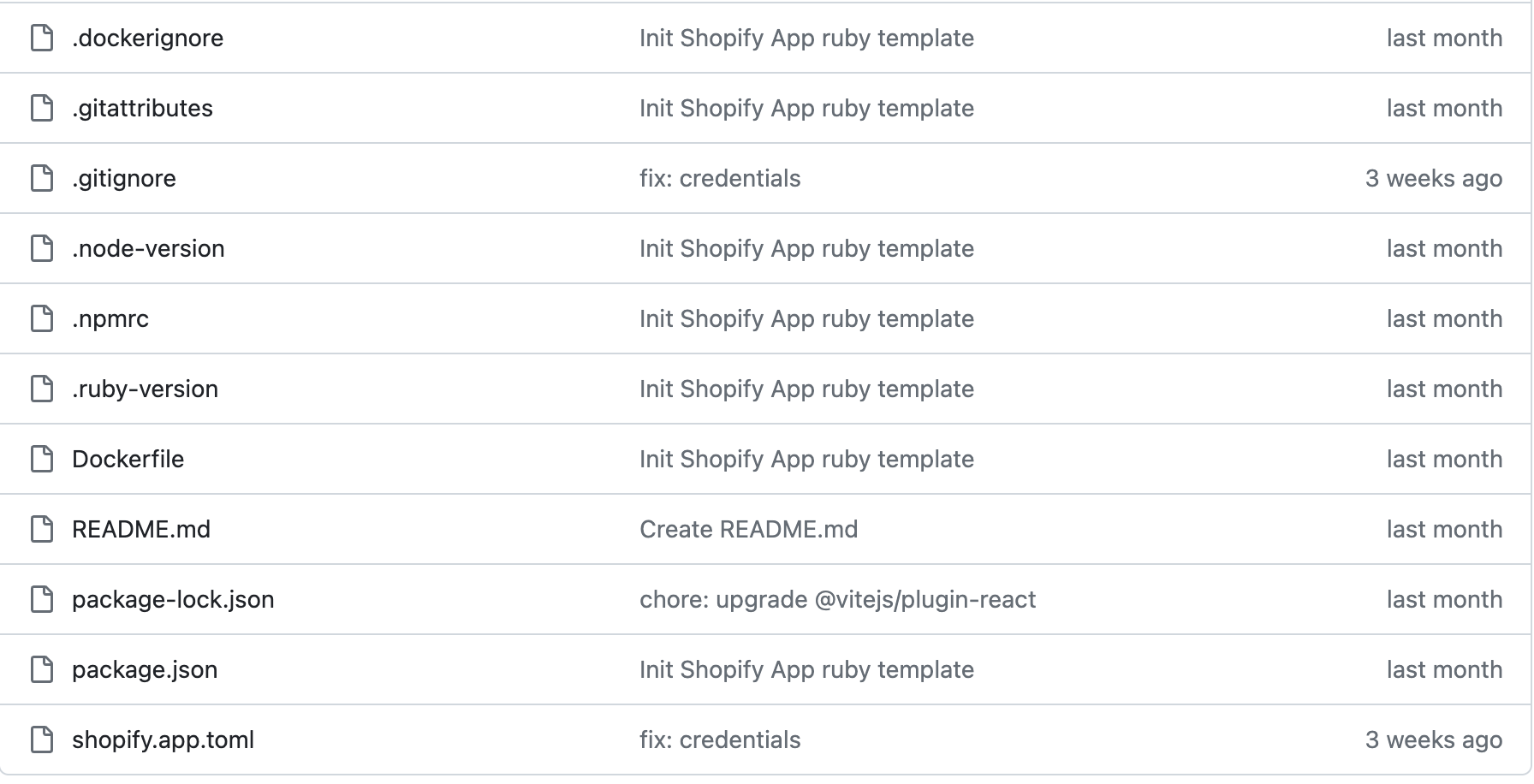
## 

## 

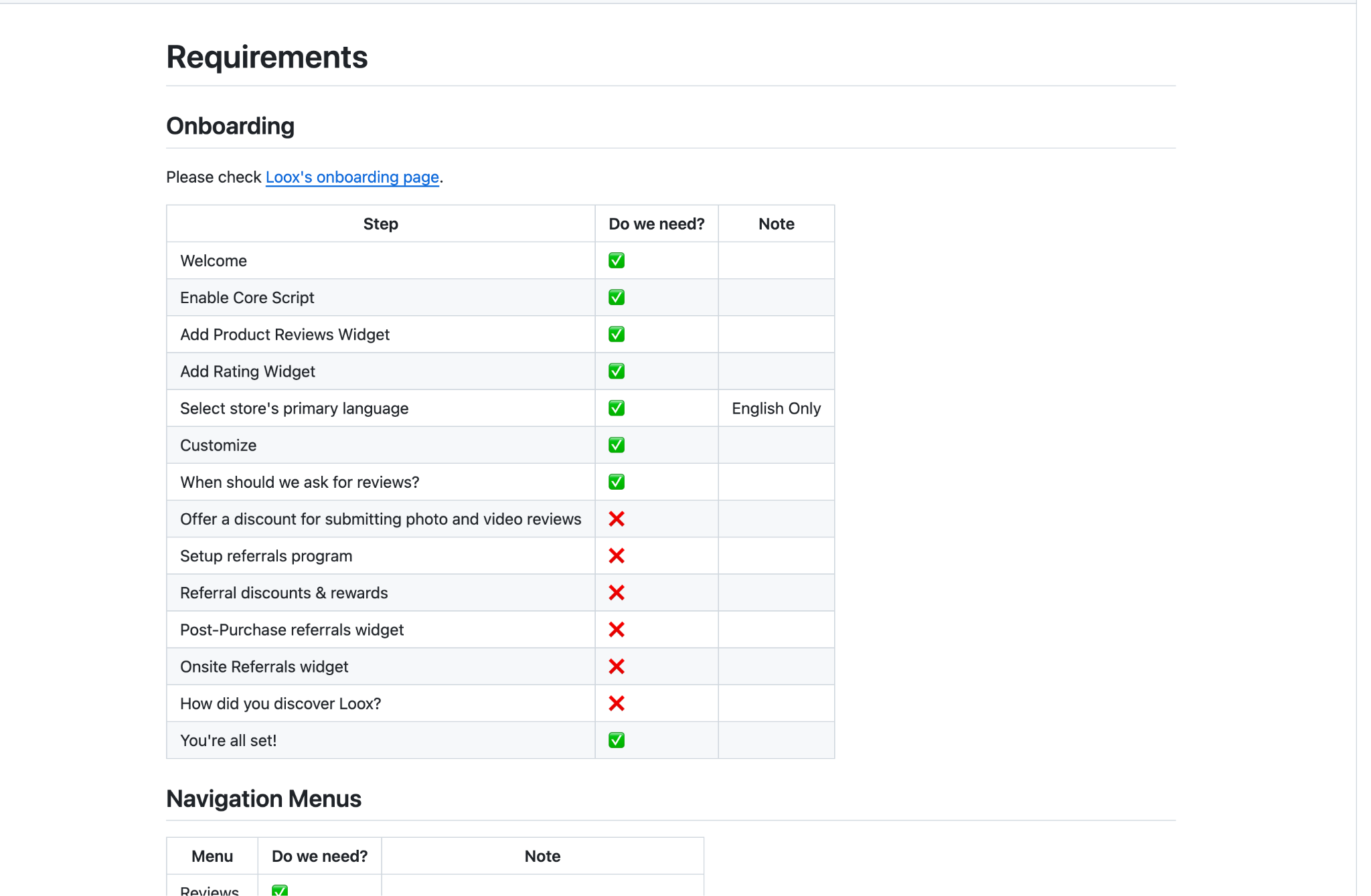
## Positive points

### 

### Included config files like, .dockerignore, .gitignore and others essential to keep the project lean and easy to run

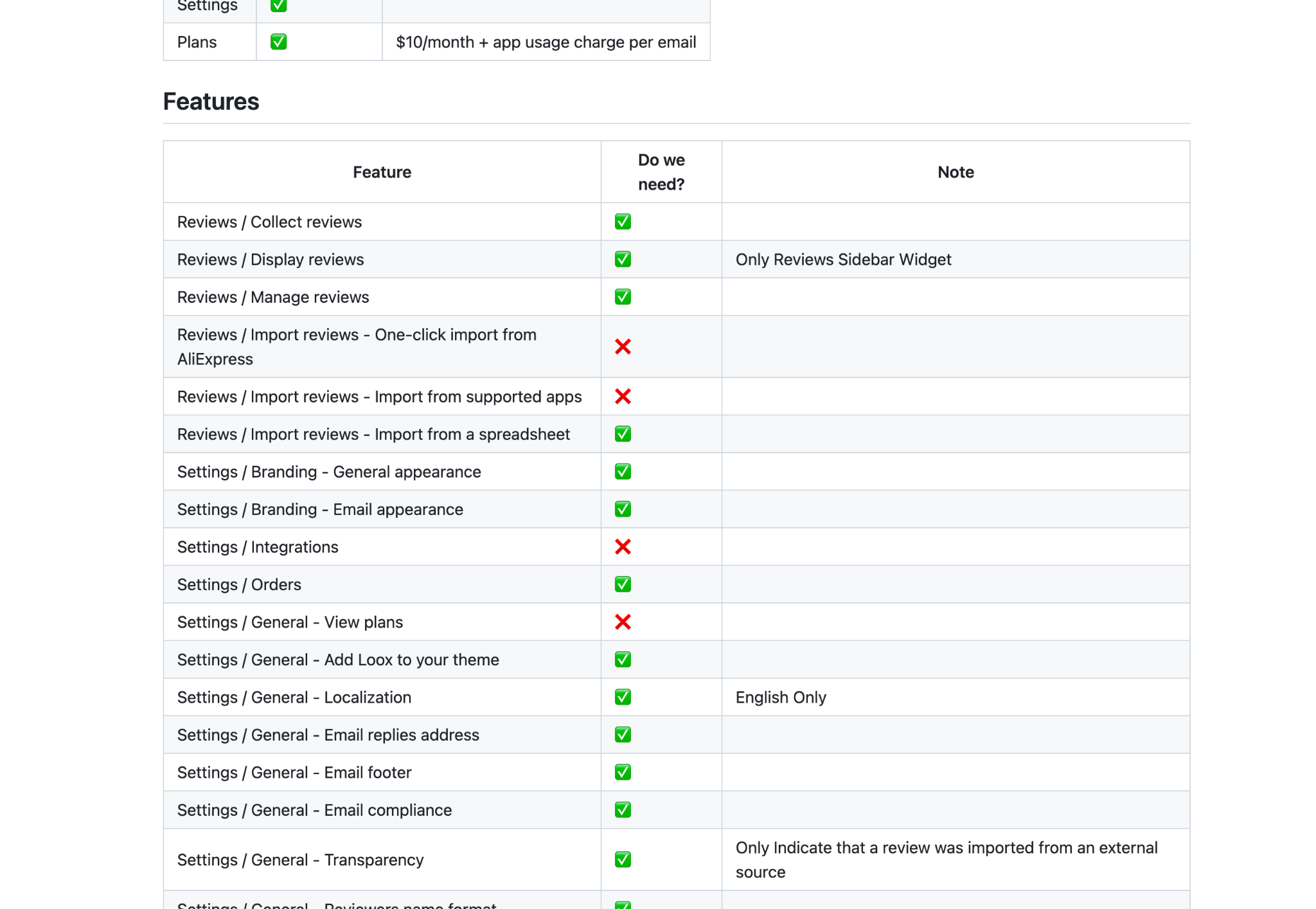


### Clear status updates through the readme.MD

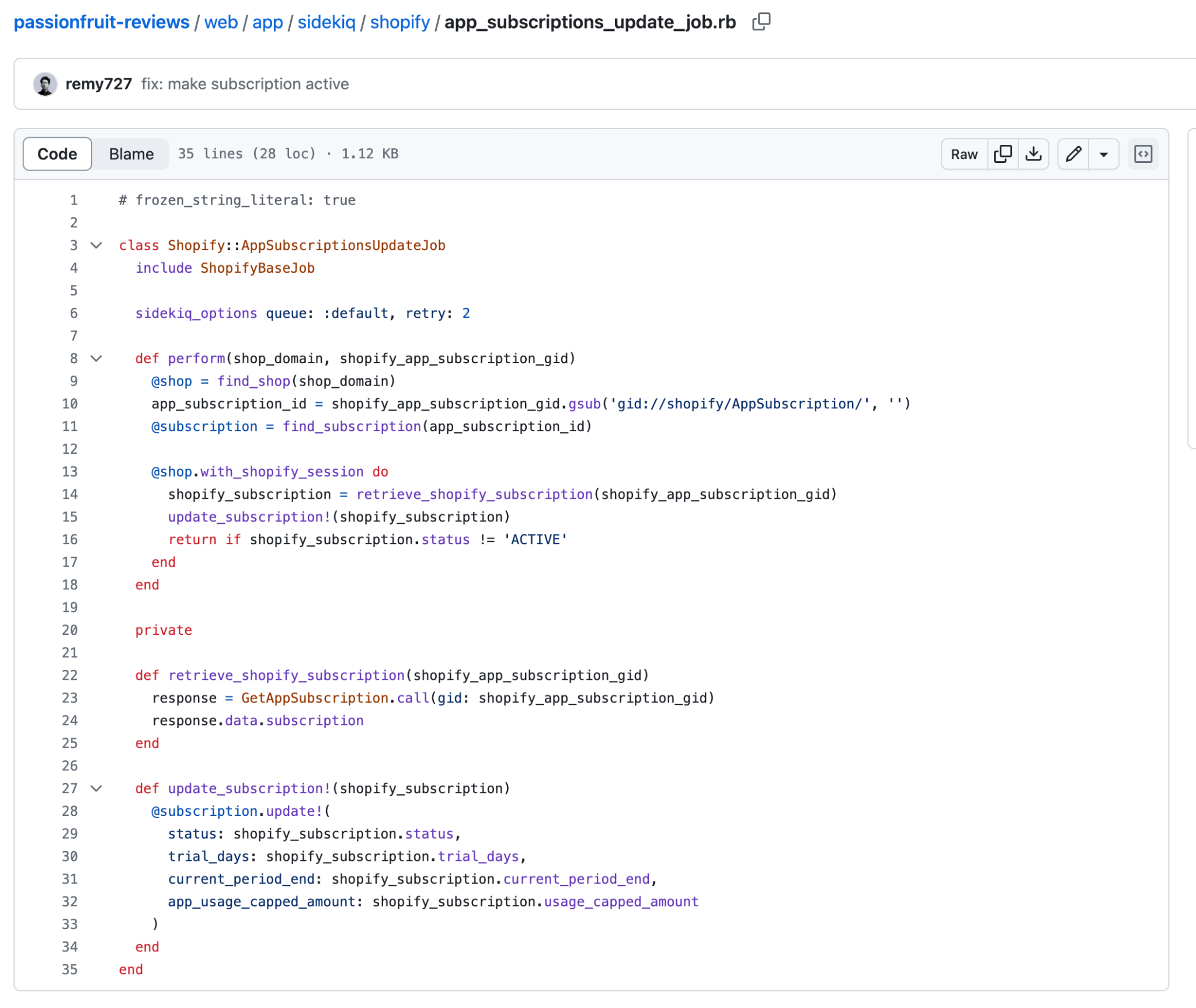


### 

### Clear tracking of features through requirements.MD

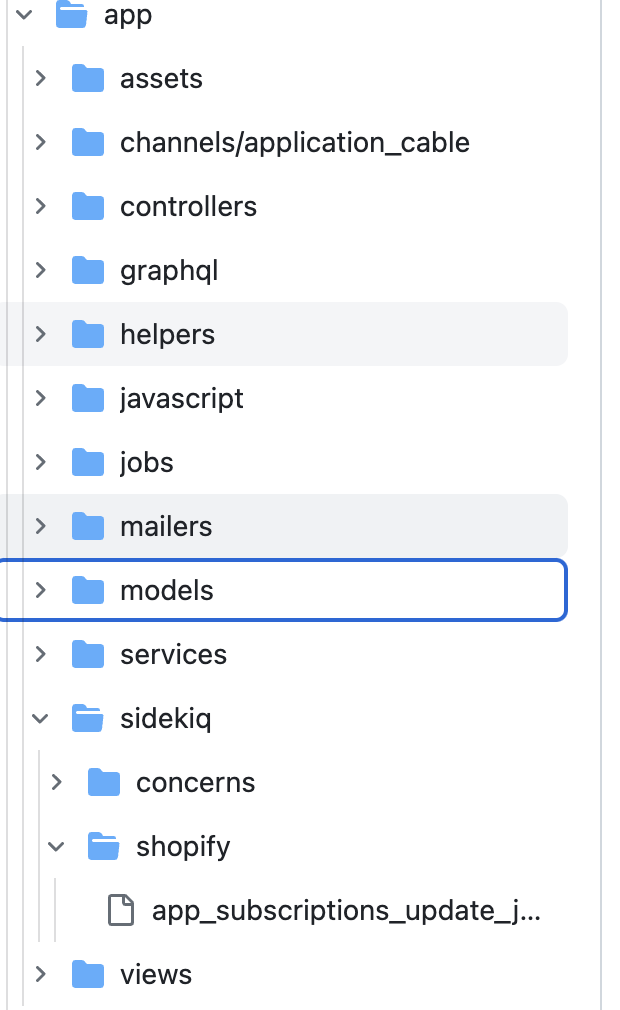


### Use of sidekiq for background jobs makes the app more scalable and enables it to process data in an async way improving the general app performance

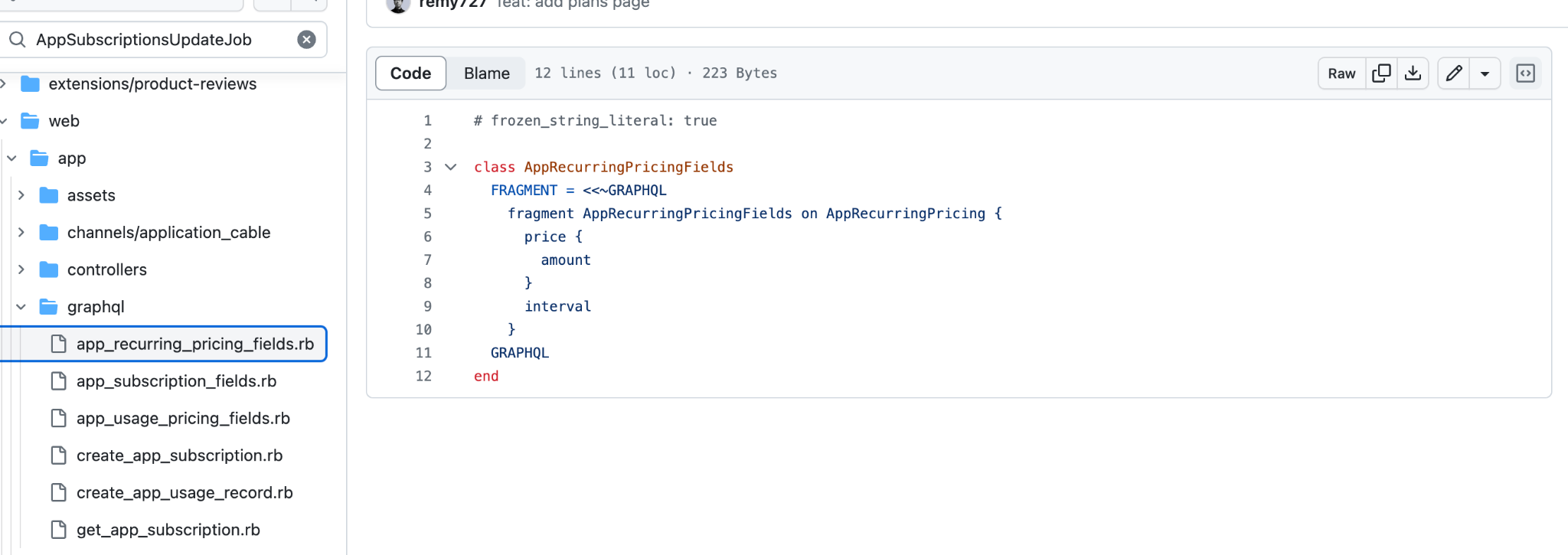


### 

### Modular Design models, jobs, services, and controllers, following Rails conventions which makes the codebase easy to navigate.



### Graphql integration makes it easier to consume the app from the frontend



### Error Handling

* The application includes error handling in several places, such as in ShopifyBaseJob, where it raises exceptions if necessary records are not found.

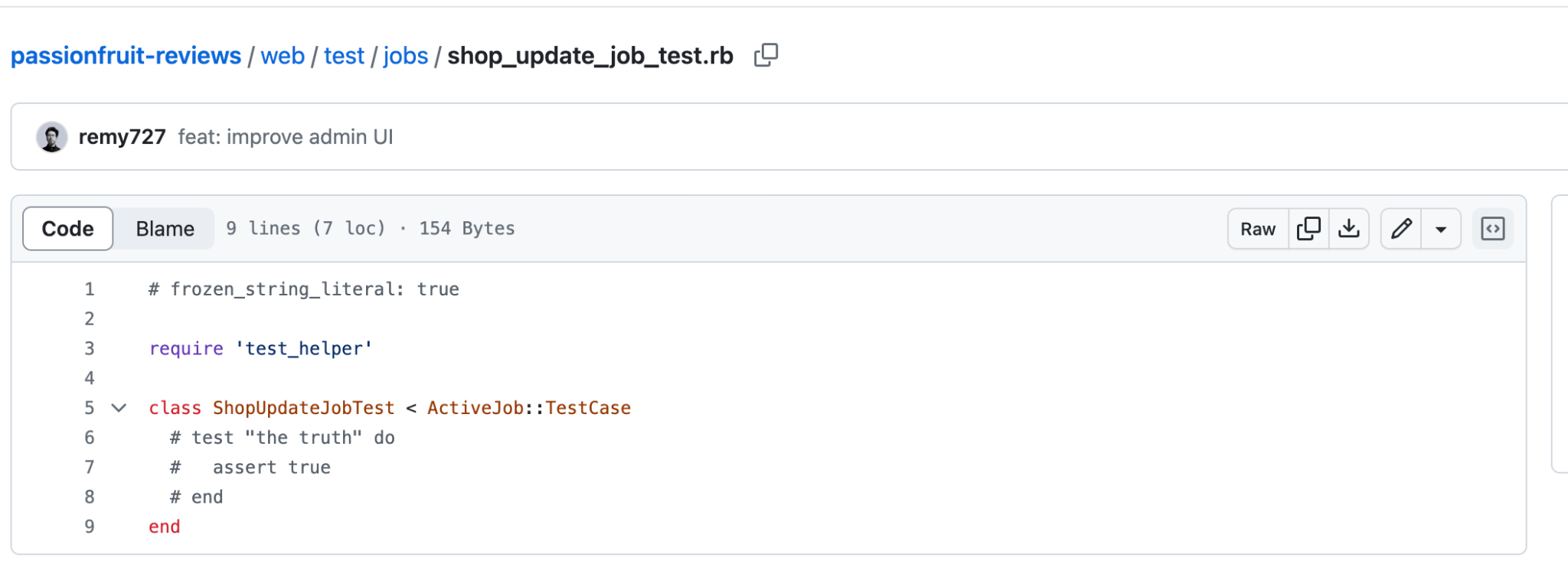
### Shopify App Best Practices

* The application follows Shopify app development best practices, such as using webhooks (config/initializers/shopify\_app.rb) and storing Shopify session tokens securely.

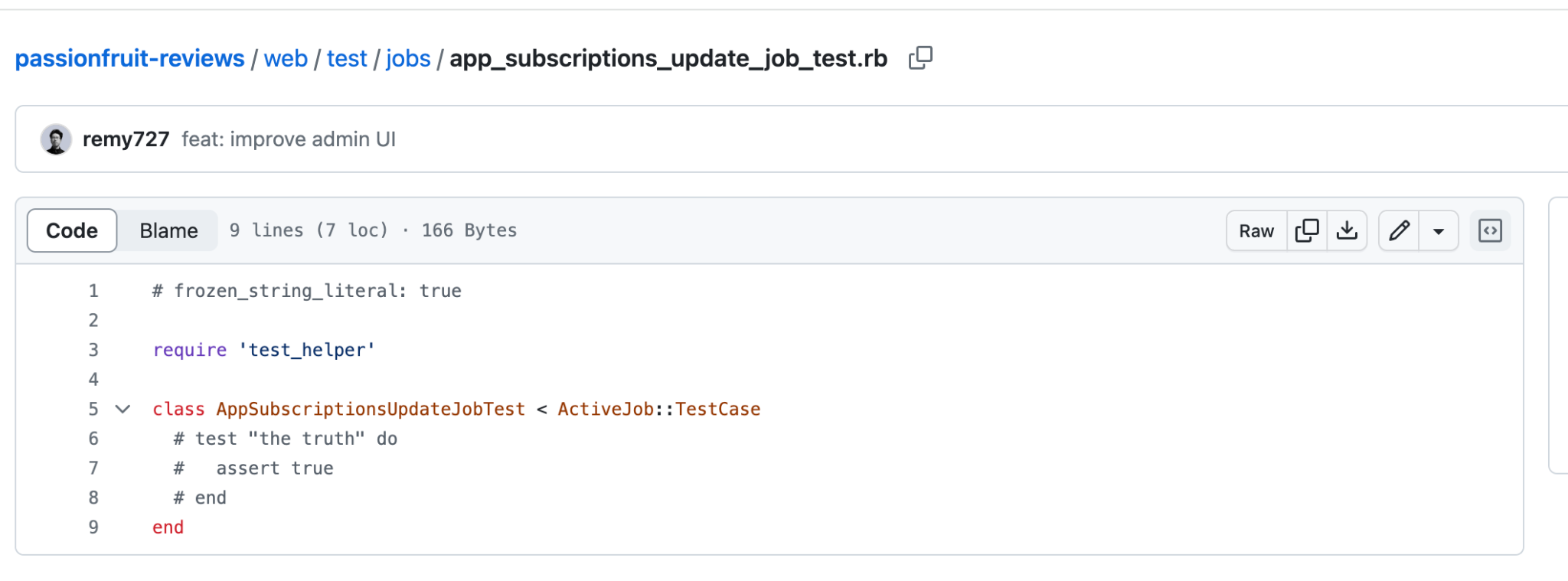
## 

## Points that need improvement

### overall the app lacks testing, adding that will make it easier in the future to add new features without introducing regression bugs



In general most of the test files are empty

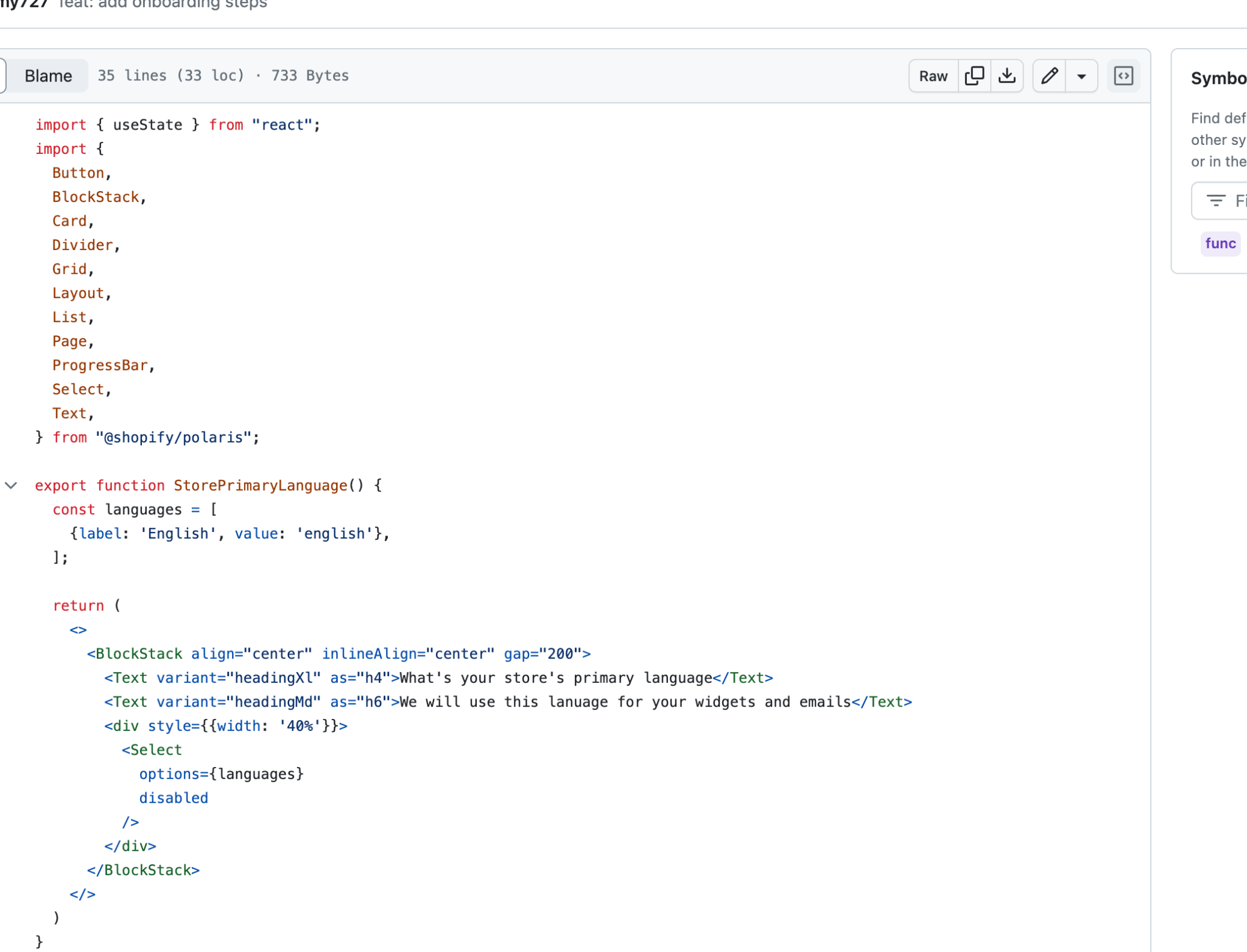


The same is happening in the frontend

### 

### Some values are hardcoded inside the code and it would be better to read them from a config file





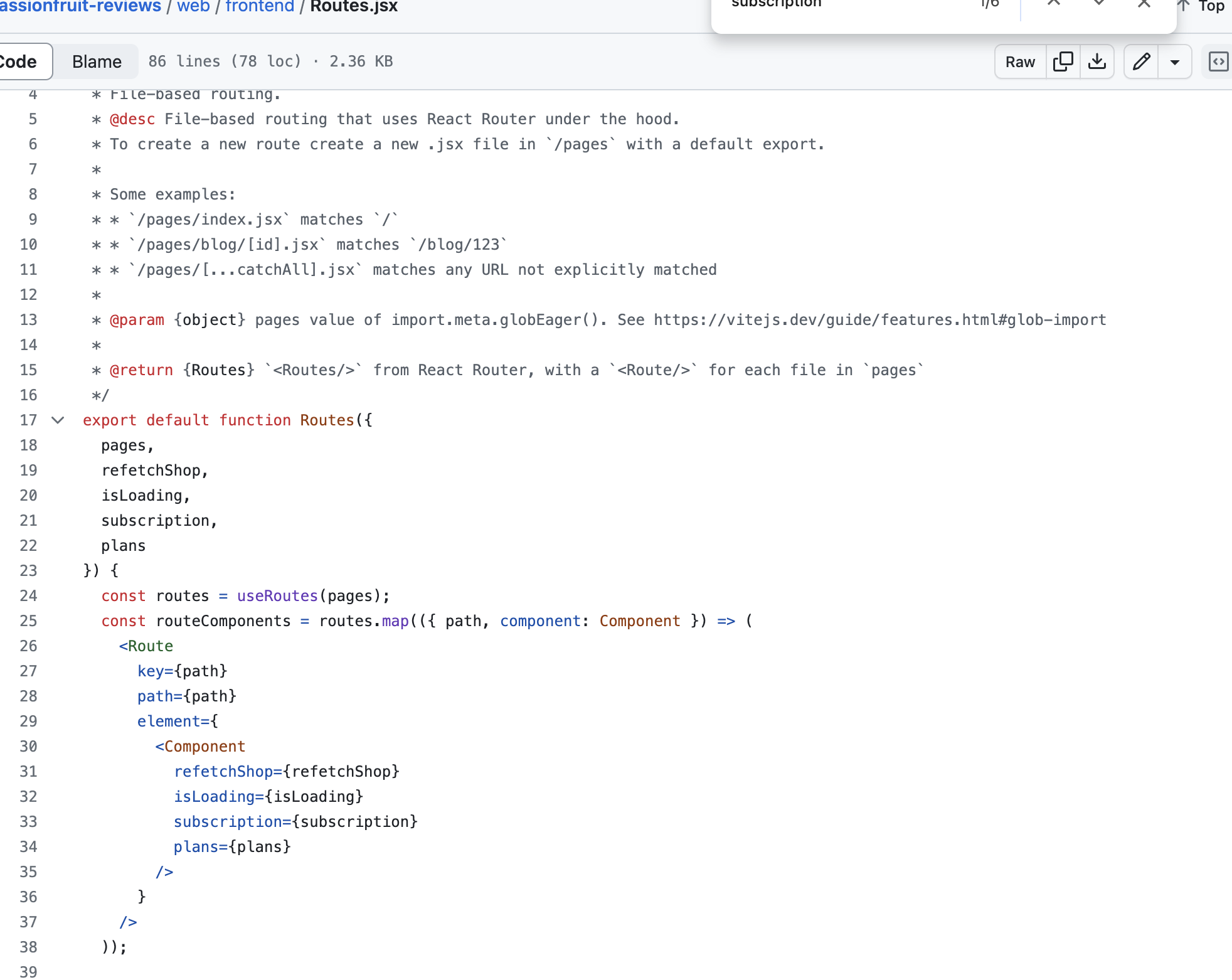
### 

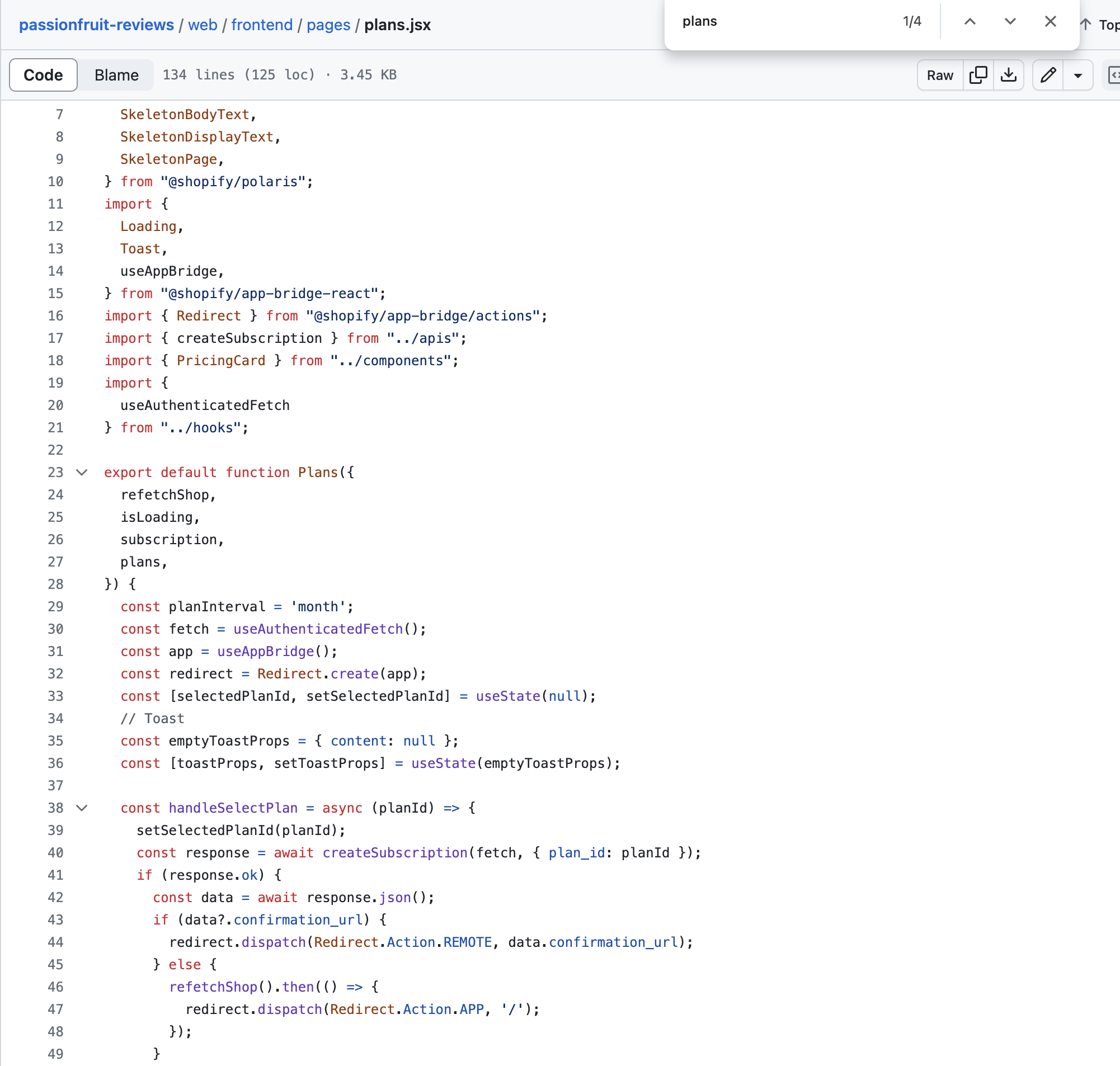
### Improve state management to avoid props drilling

Probably it would be a good idea to use a more robust state management solution, either through context, redux, zustand or other to avoid props drilling(props being passed down multiple levels to the components)

Example of props drilling happening







### 

### Lack of clear guide on how to run the app and more technical documentation on readme.md

