Take-Home Challenge:

Reap Take-Home Full Stack Coding Challenge

For this take-home challenge, you will be creating a simple image-sharing web app, InstaPic. You will implement both a backend and a frontend as indicated below.

1. Backend Challenge

- You will create a mock image-sharing website, InstaPic.
- Create a simple REST API where a user can upload a picture and associated data model. Feel free to use any language or framework that's most comfortable for you.

Requirements
- [ ] Implement a RESTful backend API in any stack of your choice (Python, Node, Ruby, Java, etc.)
- [ ] Users can register by username, password
- [ ] Logged In users can submit a post that includes an image and short text description
- [ ] Logged In users can get a list of uploaded posts
- [ ] Validation and error handling
- [ ] Automated tests

Optional
- [ ] Can sort all posts by time created, or filter posts by a specific user.
- [ ] Performance optimization
- [ ] Pagination

Deliverable
Please share your repo with us and publish on a platform of your choice (Heroku, AWS, etc).
Have a README that documents how to run your source.
- [ ] Production URL (Heroku, Netlify, etc.)
- [ ] Source code (Github, Bitbucket, etc.)
- [ ] Documentation on how to run the source code
- [ ] Documentation on your API

Grading and Submission Requirements
- Graded based on code quality and test coverage

2. Frontend Challenge

- Make a simple app that uses the backend API from the backend challenge to provide a frontend for InstaPic. You can use any frontend framework of your choice (React, Angular, Vue, etc.)
- Feel free to implement the UI your own way. However we'll pay attention to code quality, and how neatly you implement it.

Requirements
- [ ] Implement frontend
- [ ] Logged In users can submit post that includes an image and short text description
- [ ] Validation and error handling
- [ ] Automated tests
- [ ] Logged In users can view all posts

Optional
- [ ] Posts sorted by time
- [ ] Logged In users can view posts from a specific user
- [ ] Pagination
- [ ] Performance optimization
- [ ] Integrate with Redux and React Router

Submission
Please share your repo with us and publish on a platform of your choice (Heroku, AWS, etc).
Have a README that documents how to run your source.
- [ ] Production URL (Heroku, Netlify, etc.)
- [ ] Source code (Github, Bitbucket, etc.)
- [ ] Documentation on how to run the source code
- [ ] Documentation on your app, optional GIF or video

Grading and Submission Requirements
- Graded based on code quality and test coverage

Both challenges must be completed.