

Soprano.ai / Mintdropz and Why we're the future:

Soprano.ai is essentially building the next generation of technology that is disturbing the entertainment space and is bringing power back to the creators.

We provide the biggest creators across Hollywood, Bollywood, Youtube, Gaming, E-Sports, Music (Hip Hop - Pop), K- Pop, Cricket, and Formula 1 Racing to release Direct to consumer applications with paywalls, NFT's, and even their own crypto currency!

Learn more at: [Soprano.ai](https://soprano.ai) & [Mintdrops.com](https://mintdrops.com)

Your Task:

Your Goal is to build a small application where a user can create a post on their own social media type platform.

Why?

We're a social media platform that is disrupting the space of Cryptocurrency and NFT's. The small scale of these application functions will teach us more about your technical ability and how you think about the usage of tools like MongoDB, Bcrypt, And authorization.

What we're looking for:

- Clean Code
- Logic
- Use of Tools
- Creativity (can you add something to this like an additional feature? Maybe some blockchain or AI implementation? Video streaming? Show us what you got!

Next Steps?

Once completed, upload to a public repo on github and email abhishek@soprano.ai and ryan@soprano.ai

Once we receive the email, please allow 24-36 hours for us to review. The sooner you submit, the sooner we can get back to you!

BACKEND TASK DESCRIPTION

- Step1- Create an expressjs server
- Step2- Connect mongodb atlas with this server
- Step3- Create a user model with name email password image
- Step4- Create an api get the data (name email password , image) and save it using mongodb
- Step5- Password should be hashed using bcrypt js
- Step6- Validate the email is not already exist and not invalid
- Step7- Send the response with created user without password
- Step8-Create Post model with image title and description
- Step9-Create post functionality
- Step10- Get all post functionality
- Step11-Get post by id functionality
- Step12- User can delete his own post
- Step13- User can update his own post
- Step14- Only signed in user can see all post
- Step15- Implement Authorization flow make apis private