BUZZ CHRONICLES > EDUCATION
Saved by @SteveeRogerr
See On Twitter

Twitter Thread by Kati Frantz

Kati Frantz
@bahdcoder

I get asked a lot how you can improve your skills and chances of getting a job as a developer. Best way is to work on a real-world project, deploy it, make it open-source, get feedback from others, share your knowledge, rinse, repeat.

Here are my top 7 project ideas. Thread ■

1. ■ Build an embeddable user feedback form (clone of https://t.co/xFHvT7iFEf) . Have a top notch design, fully working, minimal bugs, open-source, deploy it free on Heroku / Netlify / Vercel. If you can spare $11, buy a domain. Share with the whole world when done.

2. ■ Build a product roadmap SAAS.(https://t.co/Rq9DBeCMlh) Users can create new projects, create different stages for their projects. The community can submit project ideas, vote on existing ideas. Project owners pay a monthly fee per project.

3. ■■ Build a digital marketplace. (https://t.co/BWd1aeWMt5) Sellers can upload digital products for sale. Customers can purchase digital products and securely download. Sellers are paid out at the end of every month. Don't make it complicated, implement a great design.

4. ■■■ Build a job board software (https://t.co/EjWoMyqi9H). Companies can post jobs for a price, providing a link to the job application form. Jobs can be highlighted as urgent for an additional price.

5. ■ Build a form builder (https://t.co/RdWTSogUZm) Customers to this software can create forms with text, textareas, checkboxes, radios, selects and file uploads. Form owners can view all form submissions.

6. ■ Build a beautiful online Skribbl game (https://t.co/znLWtqOvVI). Let it be extremely pretty, and allow people to play actual games. Let it be bug free, deployed to a real server.

7. ■ For each of the projects above, I recommend getting a domain, writing articles and tutorials on how you built the functionality, making it have an amazing UI. Make the project open-source, send it to some amazing developers you know for code reviews...

... add continuous integration and deployment, automated unit / integration / end to end tests, use Typescript, Scss, add a full project documentation. If possible, collaborate with another developer, and get a more senior person supervise your work weekly.

Thank you for reading so far. Please follow for more amazing content ■