Keyhook tech test

Keyhook is an all-in-one property management software. One of our big features is the ability for property managers to advertise their rental property through our platform and manage things like enquiries, private viewings, applications and more.

However, whilst property managers can schedule a private viewing with potential tenants, there is no way for tenants to currently see available times and request a viewing directly.

Can you design and build a proof of concept system that would allow a property manager to specify times of the week they are available to conduct viewings, and then allow tenants to book out those times.

You are free to use the languages and stack of your choice. But we ask that you provide documented interactions to run in order to have your solution up and running. When it comes time to submitting your solution, please upload it to a cloud git provider of your choice (eg: Github, Bitbucket, Gitlab etc) and provide the link to the recruiter you are working with.

What we're looking for:

- We are not expecting production ready code. However, we would like to see comments regarding design decisions and what you would expect to do to have this production ready.
- An understanding of patterns and idiomatic code in the tech you’ ve chosen (don’ t pick a new one)
- Are you solving the customer's problem.
    - How easy is it for both property managers and tenants to accomplish their end goals using your proof of concept.
    - Does the property manager have good oversight over what bookings are made and who they are meeting?
- How you implement the logic for bookings and the level of attention to detail you apply to this process.
- If you are a frontend leaning developer, we will weigh the UI/UX of your implementation quite highly.

Things you don't need to worry about.

- Authentication / registration / login etc. It can be stubbed out, faked or skipped.
- Sending communications (such as emails or SMS)
- Deployment, we just care that it runs locally
- Choosing production ready data stores eg: Postgresql, things like SQLite are fine for this.
- Feel free to use 3rd party components / design frameworks when building the frontend.