djangostars

THE COMPREHENSIVE LIST OF QUESTIONS

That Will Help to Prepare to Technical Due Diligence

Contents:

1. TECHNICAL QUESTIONS
2. QUESTIONS ABOUT THE CODE
3. PRODUCT AND CULTURE QUESTIONS
4. CONTACT US

1 Technical Questions:

1. Key metrics:

- What is the lead time for changes (the average amount of time from registering the timecode with version control until it is deployed in a production environment)?
  — Note: good option is less than an hour.
- What is the deployment frequency (number of deployments in production overtime period)?
  — Note: the good indicator is fixing changes multiple times per day.
- What is the meantime to restore from a failure (period to resolve or roll back an error)?
  — Note: good is less than an hour.
- What is the percentage of the change failures (in software releases and configuration changes)?
  — Note: preferable is less than 15%.

2. What is the automated test coverage?

Testing pyramid — Unit, integration, functional.

TABLE OF CONTENT

Technical Questions

Questions about the Code

Product and Culture Questions

Contact us

1 Technical Questions:

3. Code reviews.

How often code reviews are performed.

4. Implementing continuous integration and automated delivery pipelines.

Tests, code coverage, code quality metrics, complexity metrics included.

5. How is the infrastructure built?

Who is a cloud provider? E.g., AWS, Google Cloud.

6. Monitoring and Alerting. Infrastructure and exceptions.

Infrastructure: e.g., Sentry, NewRelic. Alerting: e.g., OpsGenie.

7. How's logging performed?

E.g., Papertrail.

8. What are the code infrastructure features?

E.g., Terraform, Docker, Kubernetes.

TABLE OF CONTENT

Technical Questions

Questions about the Code

Product and Culture Questions

Contact us

2 Questions about the Code:

1. Is the stack hirable?

Is it a niche technology or plenty of talent? Onboarding speed?

2. What's the type of architecture?

Monolith, microservices, serverless? Apps — native or not?

3. Clean code?

Is it in accrodance with SOLID priniciples?

4. Using common patterns and known technologies.

If the solution is not custom-made, what common patterns or known technologies are used?

5. Are abstractions in the right places, e.g., around external integrations?

Modularization.

6. Are APIs well-structured?

E.g., RESTful API.

TABLE OF CONTENT

Technical Questions

Questions about the Code

Product and Culture Questions

Contact us

3 Product and Culture Questions:

(asked at CEO call, mostly)

1. Does the product support a business idea of lean thinking?

Do they create MVPs?

2. Setting customer-oriented goals.

Are the goals set by the prodcut customer-centered?

3. Do they perform user testing?

Note: Preferably regularly, e.g., have 3—5 users coming in twice a week.

4. Do they view the problem and solution domains as separate?

5. Do they look for the best interface solutions using user feedback?

TABLE OF CONTENT

Technical Questions

Questions about the Code

Product and Culture Questions

Contact us

3 Product and Culture Questions:

(asked at CEO call, mostly)

6. Do they take competent people to the team and rely on them?

The competency of the team members is a vital factor for the code qaulity and development consistency.

7. Do they have a meeting schedule?

E.g., daily, weekly, monthly, quarterly sessions or reports. Are they working proactively or reactively?
This shows the company's approach to processes management and handling crisis situations.

TABLE OF CONTENT

Technical Questions

Questions about the Code

Product and Culture Questions

Contact us

djangostars

BOOK A MEETING IF YOU NEED FURTHER HELP WITH UPCOMING TECH DUE DILIGENCE

BOOK A CONSULTATION

OUR DESIGN PORTFOLIO

behance.com

CLIENTS' REVIEWS

clutch.co

Testimonials

FOLLOW US

linkedin.com

Django Stars Blog