

Oral Exam Questions SWTII 2019/20

- What is a RESTful architecture?
- What is the DRY principle?
- What is CRUD functionality? And which part of a Ruby on Rails software has CRUD functionality?
- What is a “definition of done” and who decides what “done” means?
- What are the principles of behavior-driven development?
- What differs a model testing, view testing and controller testing? What should developers pay attention to when writing tests in each category?
- How does automated testing from differ manual testing and which of them is the preferred choice?
- Illustrate and explain the behavior-driven development cycle
- Illustrate and explain what happens during a test run
- How can test data be provided to test cases? What are pros and cons of the methods?
- What are test doubles used for?
- Which types of review do you know? What are they for?
- What are the pros and cons of post-commit code reviews?
- What is the difference between a SCRUM-Board and a Kanban-Board?
- What is the difference between a git merge and a rebase?
- Give an example of when frontloading a software development process makes sense. Why?
- Give an example of when employing an agile development process makes sense outside of software development. Why?
- Explain the role of a Scrum Master.
- Given the scenario that a developer does not show up to daily stand ups, how would you go about resolving this if you were the Scrum Master?
- Does the PO have to attend daily stand ups? Why/Why not?
- Given the scenario that a PO does not show up to team meetings, how would you go about resolving this if you were the Scrum Master?
- Explain S.M.A.R.T. tasks.
- Explain the I.N.V.E.S.T. properties of requirements.
- How long should a daily scrum be?
- Should a PO attend retro meetings? Why/why not?
- Does a Scrum Master play planning poker?
- Does a PO play planning poker?
- How should a manager ideally be involved in technical reviews?
- How should a manager ideally be involved in management reviews?
- Explain Scrum of Scrums.
- Differentiate blob, commit, tree, tag.
- Explain technical debt. How would you prevent it?
- What is a detached HEAD?
- What is the difference between pull and fetch?

- Can you combine Scrum and Kanban? How/Why not?
- Explain the GitHub flow.
- Give an example process from customer requirements to deployment.
- What is a Release? What is its purpose?
- Differentiate Release and Build.
- Differentiate PaaS and IaaS.
- Given a legacy code base with no test coverage, how would you, as a new developer, go about testing?
- Where is Lean Software Development represented in the modern world?
- Why should you not estimate with real time but an own currency
- What are the main differences between test-driven development and behavior-driven development?
- Why/When should you do test-driven development?
- What is the opposite of the DRY principle?
- In Kanban and in Scrum there each is an element that limits the work in progress for the team. Which element is that and how does it work?
- What is the MoSCoW-prioritization and what problems does it come with?
- What does “Convention over Configuration” mean?
- What are Acceptance Tests and why should you use them?
- How do you handle a user story which you cannot really estimate?
- Agile Techniques are very focused on delivering value to the customer. How can developers work on necessary things that do not explicitly add value to the product with a good conscience and how can they defend the necessity in front of the customer?
- What is one of the core differences between Kanban and Scrum? And is it still possible to use elements from both Process Patterns?
- What should one focus on when trying to do a good code review? What should one try to avoid?
- Why should one automate deployments? Isn't it better to have full control over the process and be there in case errors happen?
- What are the major goals of code reviews? How do they help in achieving them?
- Ab wann lohnt sich der Overhead von Prozesstechniken wie Scrum?
- Angenommen, Sie sind CTO eines Startup mit acht festangestellten Entwicklern ohne nennenswerter Erfahrung in DevOps. Ihre Anwendung verarbeitet sensible Patientendaten (Name, Geburtsdatum, Arzt, Diagnose) und speichert sie in einer Datenbank. Ein stadtweiter Testlauf auf einem selbst verwalteten Server lief erfolgreich, und nun möchten Sie Ihr Produkt deutschlandweit anbieten. Ihr Marketing-Team rechnet mit mehreren tausend Kunden in den nächsten Wochen, und Sie rechnen damit in den Wochen nach Release je nach Kundenfeedback schnell Fixes und ggf. auch kleinere Features in Produktion bringen zu müssen. Für welche Hosting-Lösung entscheiden Sie sich (IaaS, PaaS, Dedicated Hosting) und wieso?