

RUML Interview Summaries

Anandita Misra - Business Technology Analyst II, Deloitte USI

- Visualization is used in presentations.
- All development is agile.
- As of now, there is neither need nor any easy scope of integrating a new tool in the Requirements Engg phase of Software Cycle.
- During the Remediation Techniques phase and during onboarding a new project, a sequential diagram of how the codebase works and flow-of-control in the code will help a lot
- Knowing about the dependencies in the codebase will help in writing better code, faster.
- The sequential flow will also help in understand the Functional sets.
- Sequence Diagram is very important for integrations.
- Sequence Diagram of the functions
- Suggestions are rewarded at Deloitte but the call to purchase a tool is a long one and involves multiple levels of management hierarchy.

Aditya Mohan - SSE, Filter.AI

- The onboarding is not an issue at Filter.AI, being a total tech team of 10 people.
- Debugging in languages would be easy for developers if they could have the exact sequential calls of the programs' functions.
- If the dependencies can be visualized in both small and large codebases, code reviews would be faster and better.
- Features could be shipped a lot faster and with lesser rollbacks if they could trace the Use-Case in the source code.
- Aditya holds the power to suggest and the higher management will make the final call.

Prastut Kumar, Partner and CTO at BUIDL Labs

- The user journey could be better understood had the sequential flow of the large codebases is made available, easily visualized.
- Figuring out dependencies is an issue
- A tree like diagram of the codebase that shows dependencies and people involved at some level is needed.
- Figuring out technical debts is a challenge that involves way too many humans and way too much human dependency
- Aviral should focus on solving Software issues for inside-the-org.

Ishan Kakkar, Android Engg Lead, former Atlan and current Go-Jek

- The IDEs are too slow and doesn't help me understand the codebase being written, merged
- A visualization is a must
- Current tools provide the triggers and events of the codebase but not at the code level
- Basic modeling such as UML diagrams is needed in order to present the best way possible.
- Sub-optimal or not at all optimal collaboration and code writing can be avoided
- Ishan holds the authority to strongly suggest for purchasing a tool.