**Reflection**

1. **Which SDLC phase was the most challenging? Why?**

The Design Phase was the most challenging because it required translating abstract requirements into a detailed and concrete system model. This involved ensuring that both the database design (backend entities, relationships) and the UI structure (frontend components, navigation, user flows) aligned perfectly with the functional needs. Achieving a balance between technical feasibility, user experience, and scalability made this phase particularly complex.

1. **Which SDLC model (Waterfall, Agile) best fits this project? Why?**

The Agile Model best fits the FinVerse project because it supports:

* **Iterative development :**allowing us to build the system in increments (signup, signals, subscriptions, etc.).
* **Continuous feedback :** from traders, users, and admins to refine features.
* **Flexibility :** making it easier to handle evolving requirements compared to the rigid Waterfall approach.

Given the fast-changing nature of trading platforms and user expectations, Agile ensures faster delivery and adaptation.

**3. How did you determine functional and non-functional requirements?**

-> Functional Requirements were determined by analyzing user needs, system objectives, and expected workflows. We gathered them through Google Form surveys, interviews with traders and potential users, and mapping them against core features such as authentication, signal posting, watchlist, subscriptions, and admin management.

-> Non-Functional Requirements were identified by focusing on system quality attributes like performance, security, usability, and scalability. These were based on industry best practices, expected user base, and ensuring the platform remained secure, responsive, and reliable under real-world trading workloads.