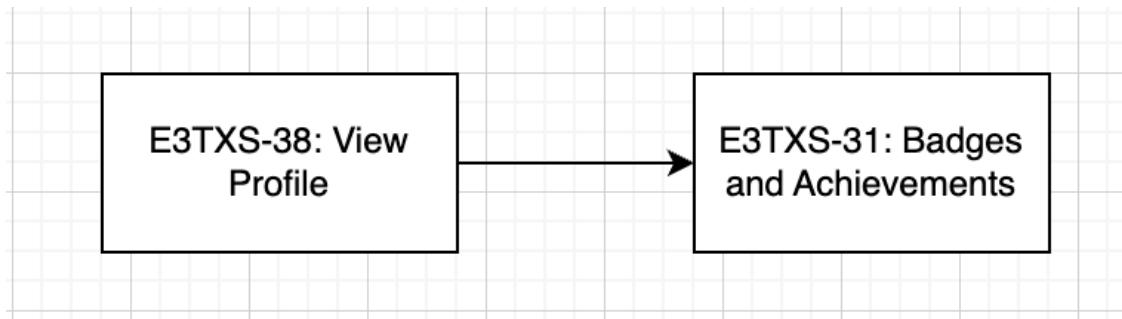


Network Flow Diagram



In Sprint 3, most of the tasks were independent of each other, except for 2, which had a clear dependency chain:

E3TXS-38 – View Profile

E3TXS-31 – Badges and Achievements

The badges and achievements are displayed inside the profile, so the profile UI structure and backend retrieval needed to exist first. Without the profile page available, there is no place to show the badges.

All other tasks (View History of Budgets, Dynamic Debt Summary, Alert Notifications, Sign Out, and Generate Monthly Report) were separate features and did not rely on one another.

Therefore, these tasks were able to be completed at any time during the sprint in parallel with other tasks. Since this is the only dependency chain, this is also the **critical path** of this sprint. Any delay in finishing the profile page would automatically delay the badges feature, because the badges rely on the profile UI being done.

How We Worked Together

- We identified early that Badges required the Profile page to exist first, so we scheduled E3TXS-38 at the start of the sprint to remove blockers.
- Once the profile was completed, the Badges task could continue without delay.
- All the other tasks were independent, so team members could work in parallel without waiting for each other.
- We used regular standups to confirm dependencies were clear, and no one was stuck. If there were any blockers, we worked together to resolve them as quickly as possible to prevent any delay in tasks.
- We learned that identifying even small dependencies early makes the sprint run more smoothly. Most tasks were independent, but the Profile → Badges relationship showed how missing one prerequisite can delay another task. Planning the dependency first allowed us to keep everything else on track.