

The forest

A **forest** is a group of trees — the full scope of knowledge for a single **role**. "Front End Developer" is a forest. Its trees (Testing, Frameworks, JavaScript/TypeScript, Styling/CSS) are the logical slices; the forest is the whole territory.

The default shape: **one forest per role, covering every rank**. One "Front End Developer" forest holds everything from the junior's first skills to the senior's specialisations, rather than a separate forest per seniority level. (A forest per rank also works — but one forest per role is usually more convenient, because a person's whole journey stays on one map.)

Two properties fall out of the structure you already know:

Forests have roots too. The same derivation that gives a tree its roots applies at the forest boundary: prerequisites that sit outside the *whole forest* are what you need before entering the role at all. For a front-end forest, that might be general computer literacy or basic HTML — whatever the forest's lowest skills genuinely require but don't contain. Forest roots are the entry requirement to the role, derived, not guessed — which makes them unusually useful for hiring.

Forests are honest about size. A role is a lot of knowledge, and the forest shows all of it. That can feel intimidating on first view — until you notice that nobody is expected to hold the whole forest. You hold *your* skills at *your* levels, and the next chapter's ranks tell you which part of the forest your current chapter of the journey is about.

The forest is the method's unit of "a role, mapped." Everything after this — ranks, coverage, hiring, reviews — is a way of reading one.



Revision #3

Created 2026-07-08 16:08:50 UTC by Admin

Updated 2026-07-08 16:37:54 UTC by Admin