

Your guide to: Preserving behavior across changes

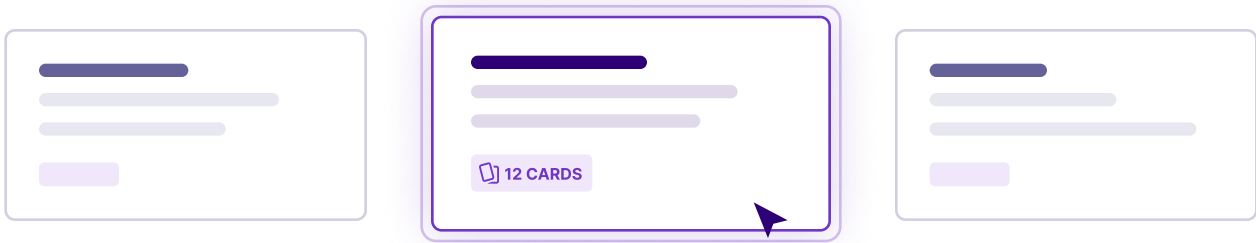
First, we discover and decide on how the code should behave.
Then, we make sure there are enough tests to enforce this behavior.

1 Scan your codebase



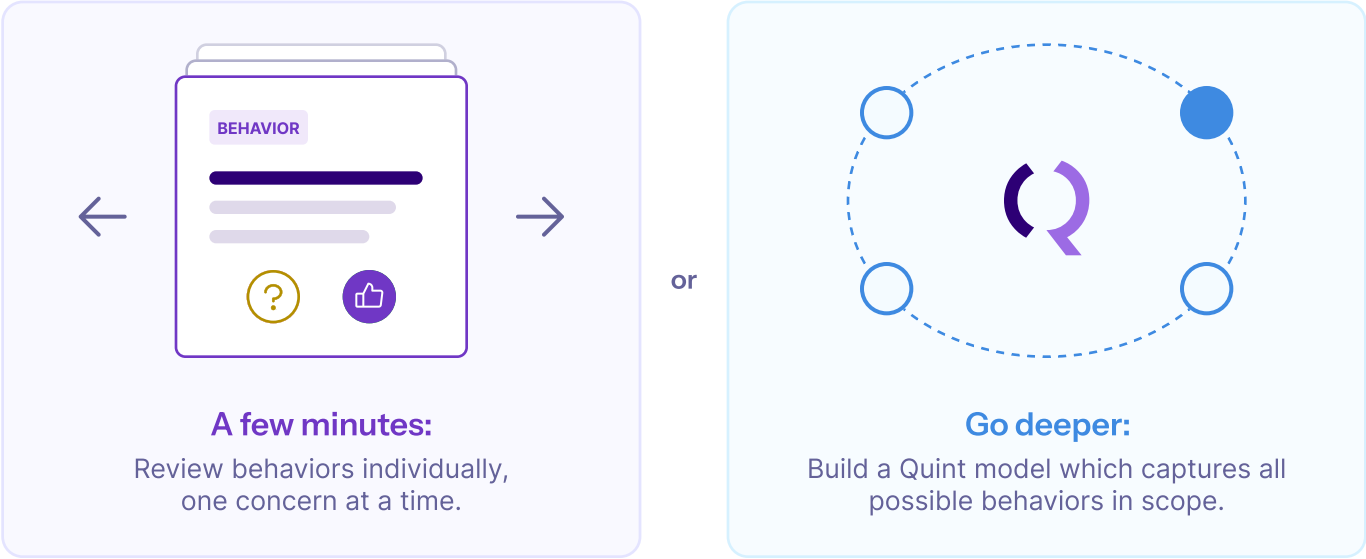
Studio maps code into components with the right scope for it to work from.

2 Pick one or more component to get started



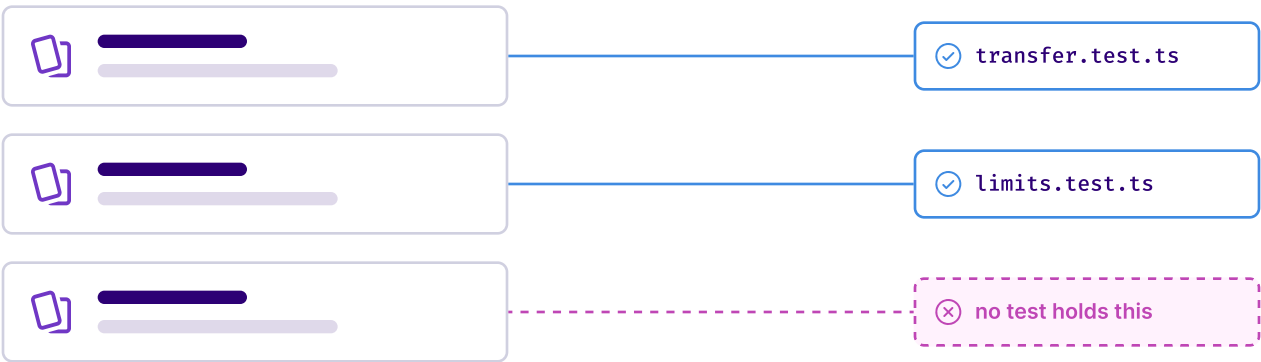
Studio recommends where to start from, but it's ultimately your choice.

3 Confirm the behavior



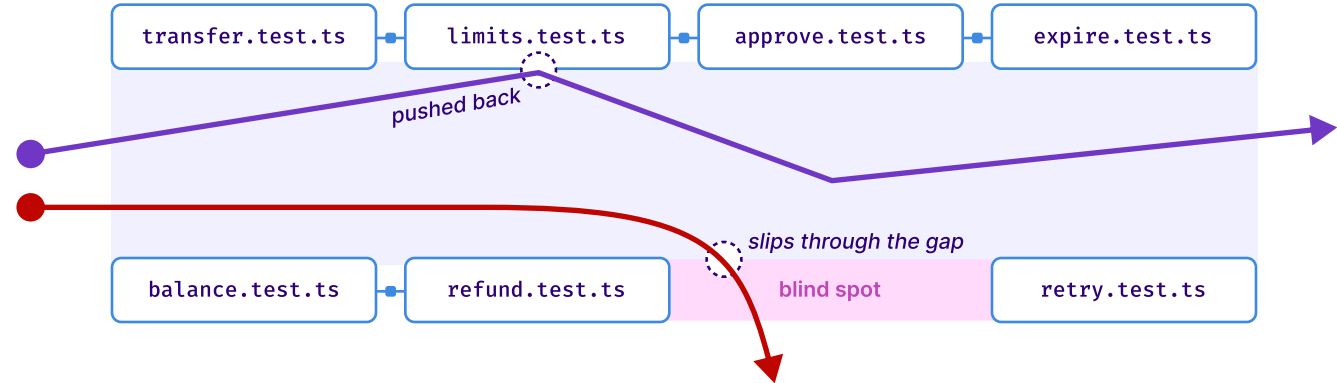
★ This step often surfaces unexpected behaviors. Studio can help you fix any of these you mark as bugs.

4 Check the tests against the behavior, and find blind spots



Every confirmed behavior or interesting trace from the Quint model is either linked to real tests or flagged as a blind spot.

5 Cover the blind spots



Every behavior-associated test is a segment of the rail.
A blind spot is a gap in the rail, until Quint Studio generates a test that closes it.

✓ Quint Studio is a local tool



Quint runs locally on your compute, driving coding agents you already have. The only things that ever leave your machine are anonymous usage telemetry and your agents' own traffic to your agent provider. Code, specs, and coverage are never shared with Quint.