Rapid prototyping with AI

March 1, 2025

Traditional prototyping takes too long

Slow process

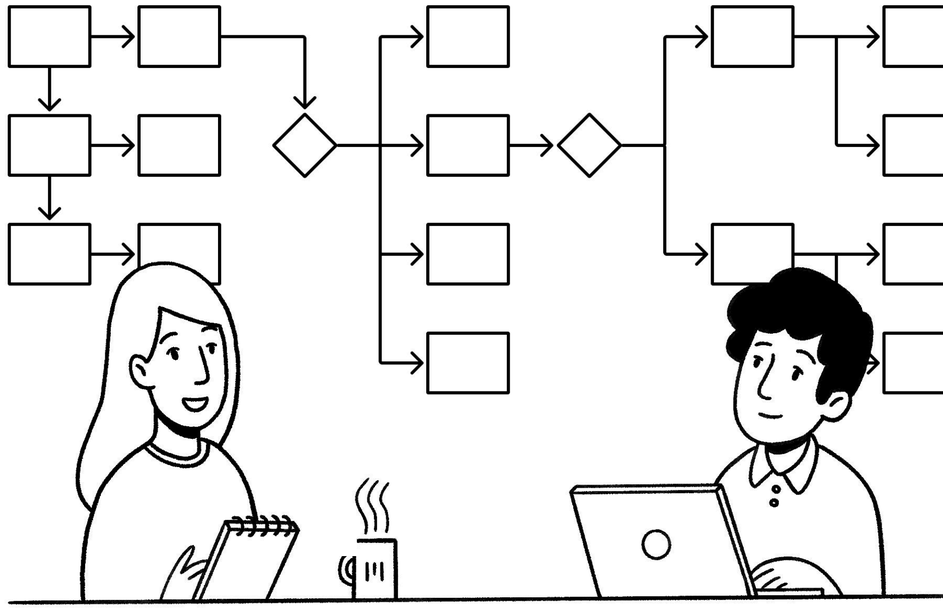
Weeks mapping flows in Figma, designing, then connecting screens.

Speed issues

Even efficient studios struggle to rapidly deliver.

Perfectionism

Striving for perfection slows down validation.



Meet AI-Powered rapid prototyping

1 Focus on validation

Prioritize validation over pixel-perfect code and design.

2 Clickable journeys

Experience full user journeys in the browser, the best interactive environment

3 Enhanced interactivity

Static Figma designs don't compare to dynamic prototypes.

4 Data-based: real or fake

Easily import real data or ask to fake data via prompting

Make ideas real, fast

1 Rapid iteration

See results in days, not weeks.

2 Skip design cycles

Focus on what truly matters for the user experience.

From rough to ready in record time

1

Rough wireframes

Define the core flows based on user stories

Figma

Collaborative design

2

Translation of UI parts

Transform user stories and rough wireframes into UI parts exportable to code

Claude + Artifacts

AI-Powered code generation

Magic Patterns

AI-Powered code generation

3

Combine

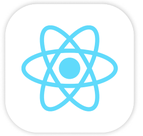
Combine code into a static site generator and link them together using a context-aware code editor

Cursor

Context-aware AI code Editor

Tailwind

Styling



React

Front-end prototypes

Who is this for?

Early adopter founders

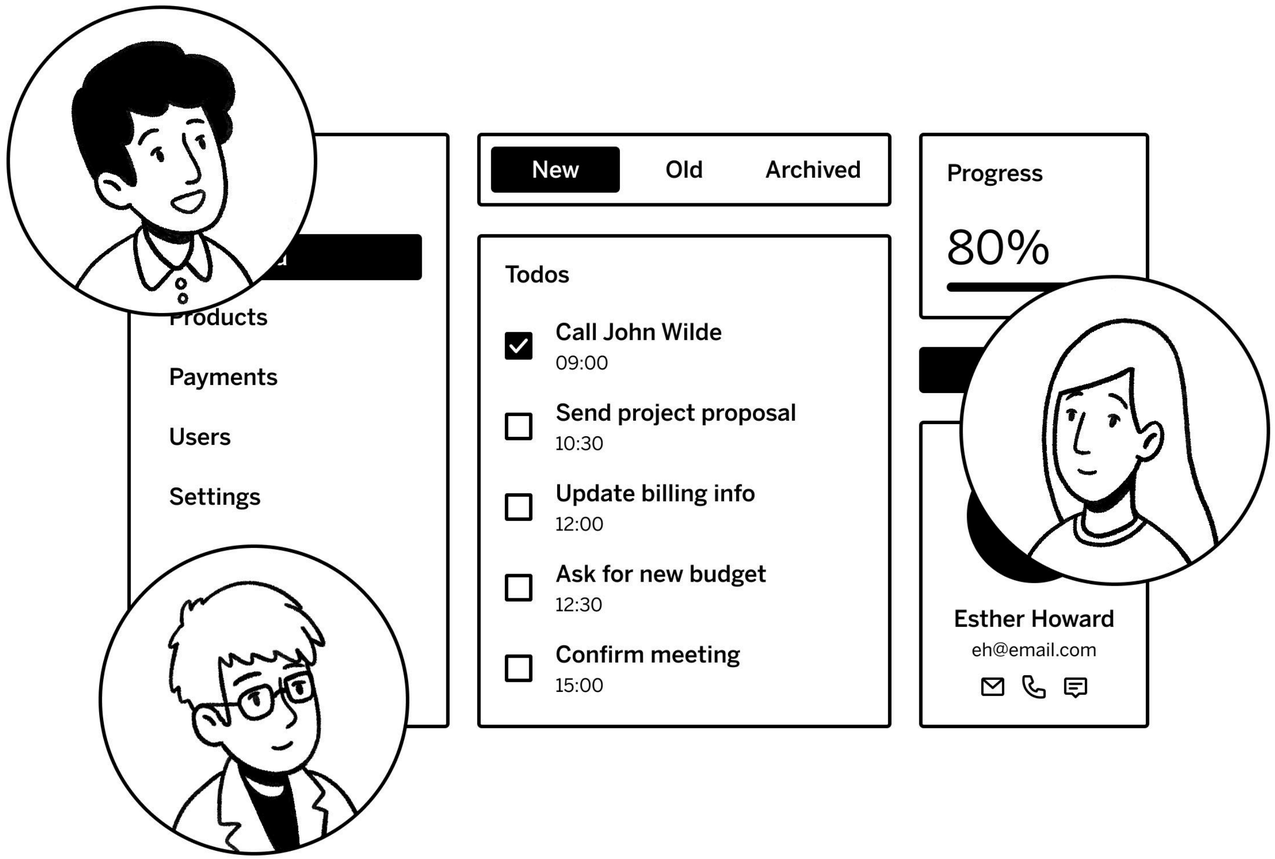
The technology exists, so why not use it?

Budget-friendly

Experiment without the big costs

Validating product ideas

Ideal to validate with actual users



Who is it not for

Complex business logic

Not for custom software with complex business logic

Custom looks

Not for projects with heavy branding requirements: projects will have a “bootstrap” look.

Code quality and testability

The code is essentialy to throw away, not to reuse. Don’t expect quality and working tests.

Complex animation

Not for projects relying on complex animation

FAQ

Can you reuse the code?

The default set-up uses React (Next.js) + Tailwind. Best results come when you intend to reuse this stack.

AI-generated code is a good starting point but it's definitetly not the final thing you want to ship. In almost all cases, the code should be adapted to higher quality code.

Once the prototype reaches a certain scale, the rate of development slows down tremendously and it's better to take a step back and think things through.

Who will be working on these projects?

Within Obra we have a team of frontend developers and designers that are using these new techniques.

Curious?
E-mail us or book a meeting!

Johan Ronsse

johan@obra.studio

cal.com/wolfr/30min