

MyRental Technical Review

Billing Team

Jimmy Liao

**Question:
What's the core values
in Gogoro software teams?**

**Answer:
Fast Development and High Quality**

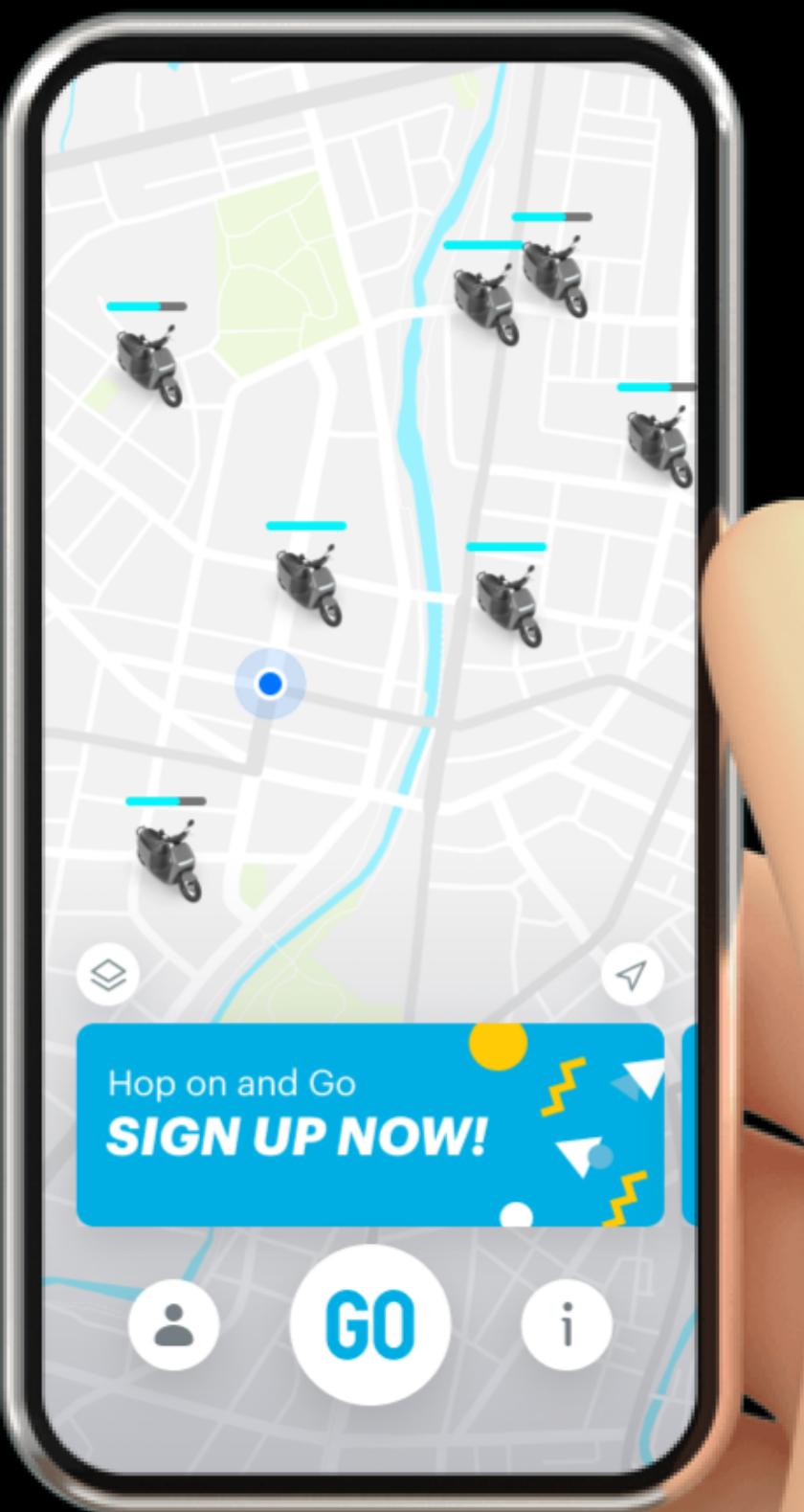
MyRental Project

We have to do something different to reach Fast Development and High Quality.

9:41



Skip



Fast Development



Fast Development Vite

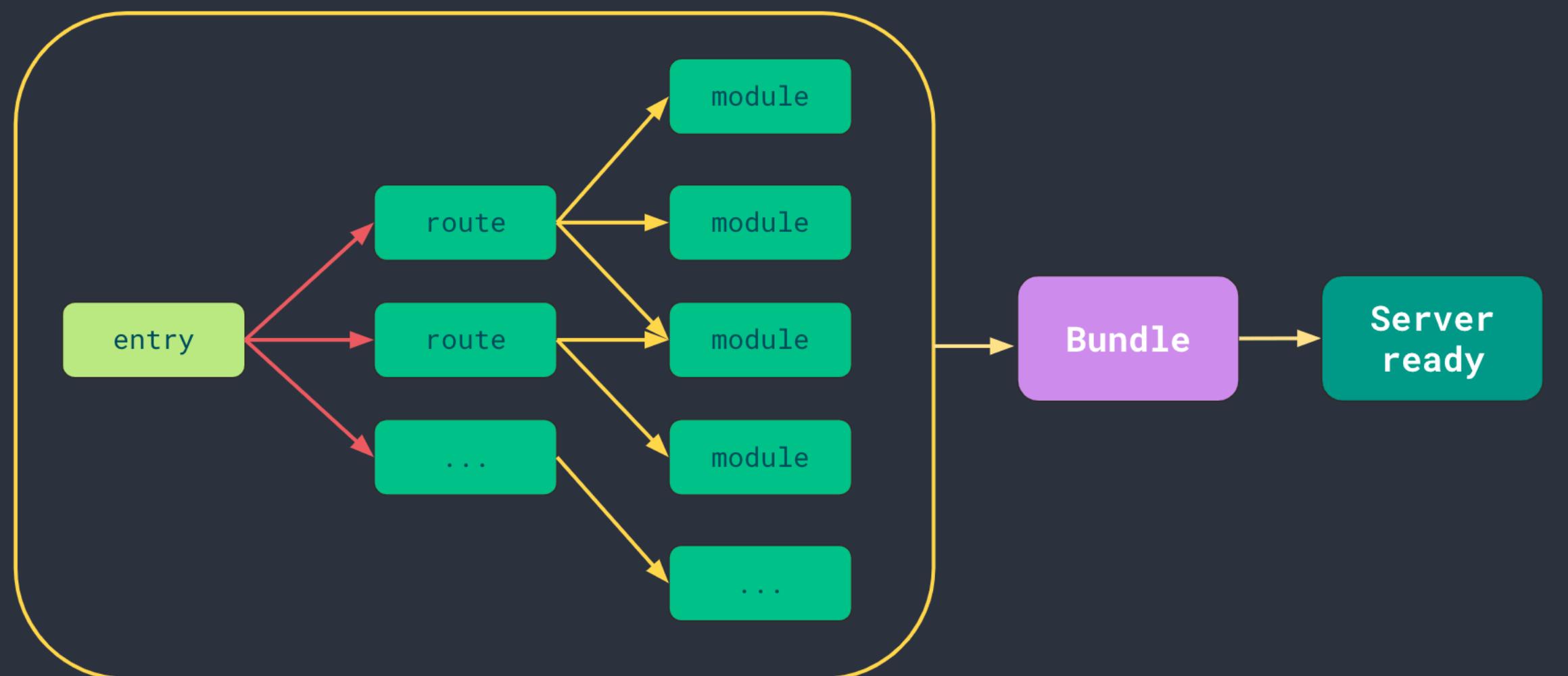
A build tool that aims to provide a faster and leaner DX

Vite

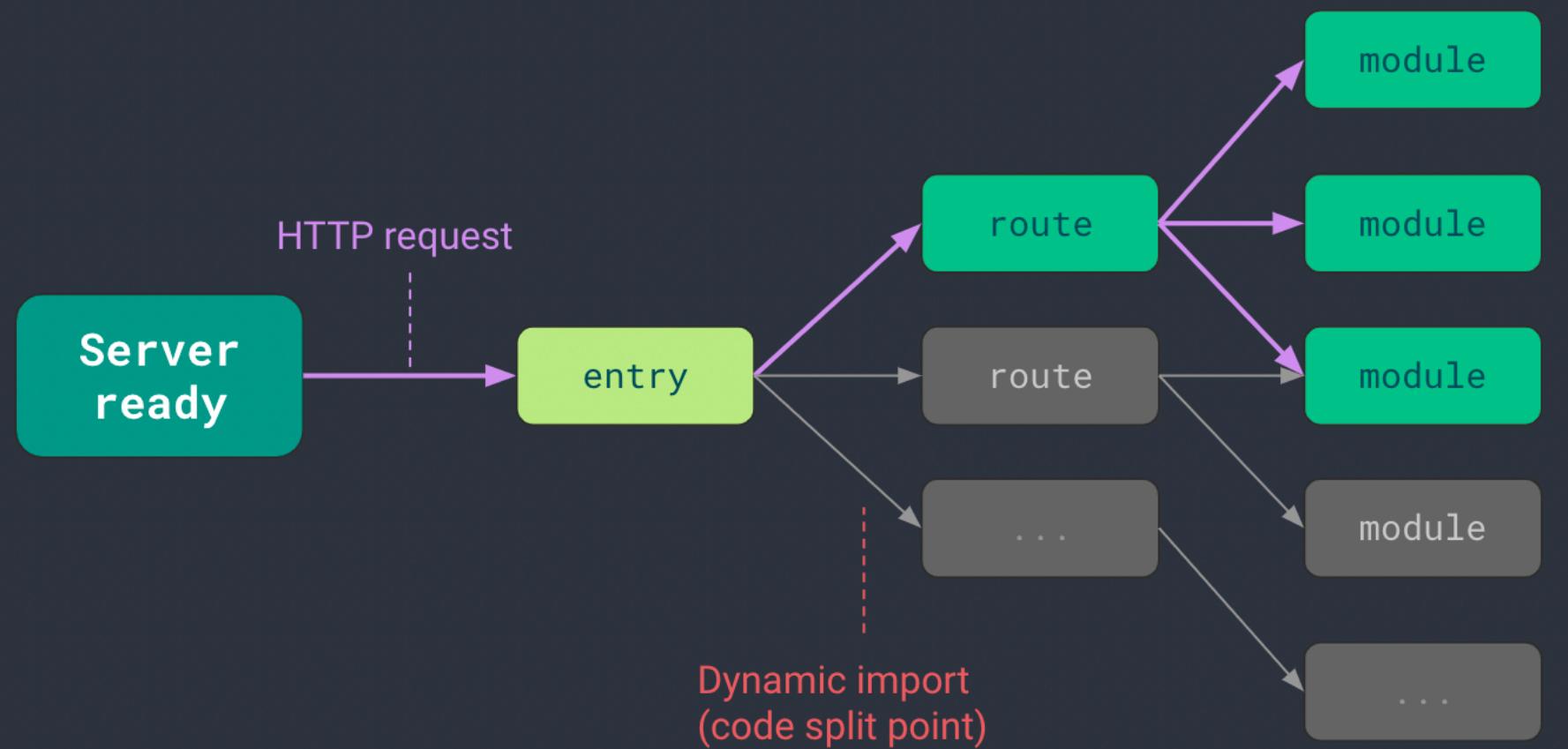
- French word for "quick", pronounced /vit/, like "veet"
- Super fast dev server start
- Super fast Hot Module Reload (HMR)



Bundle based dev server

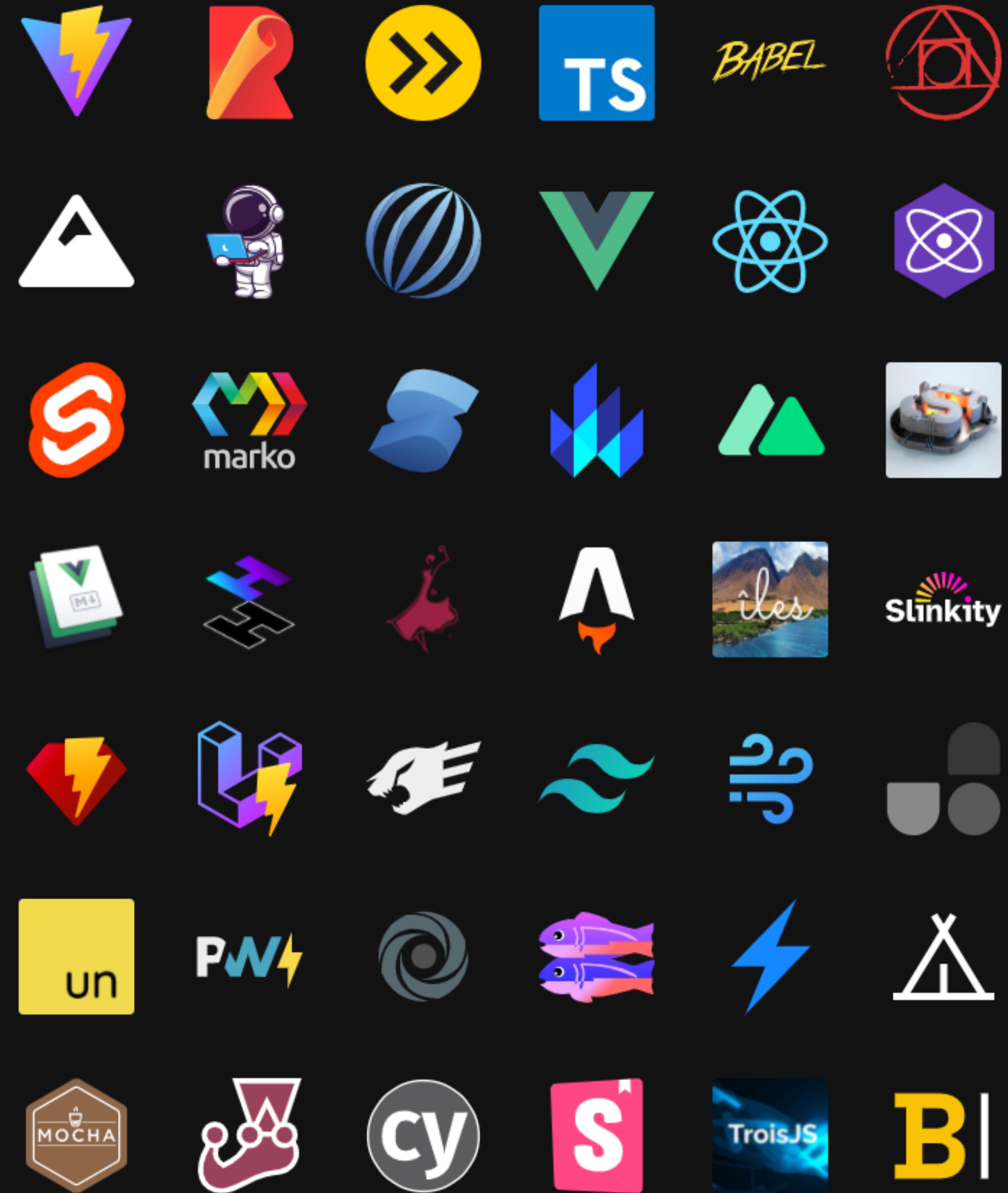


Native ESM based dev server



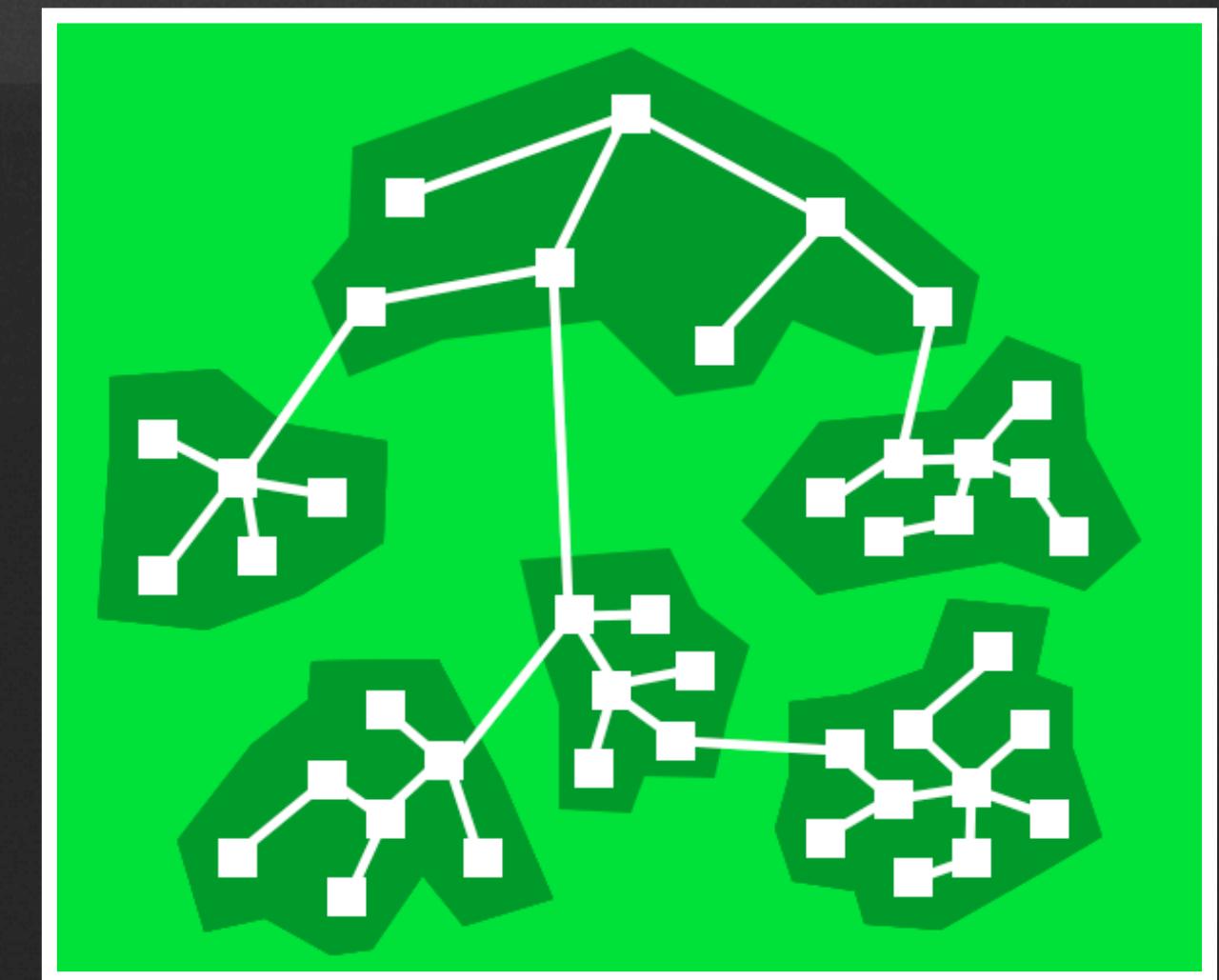
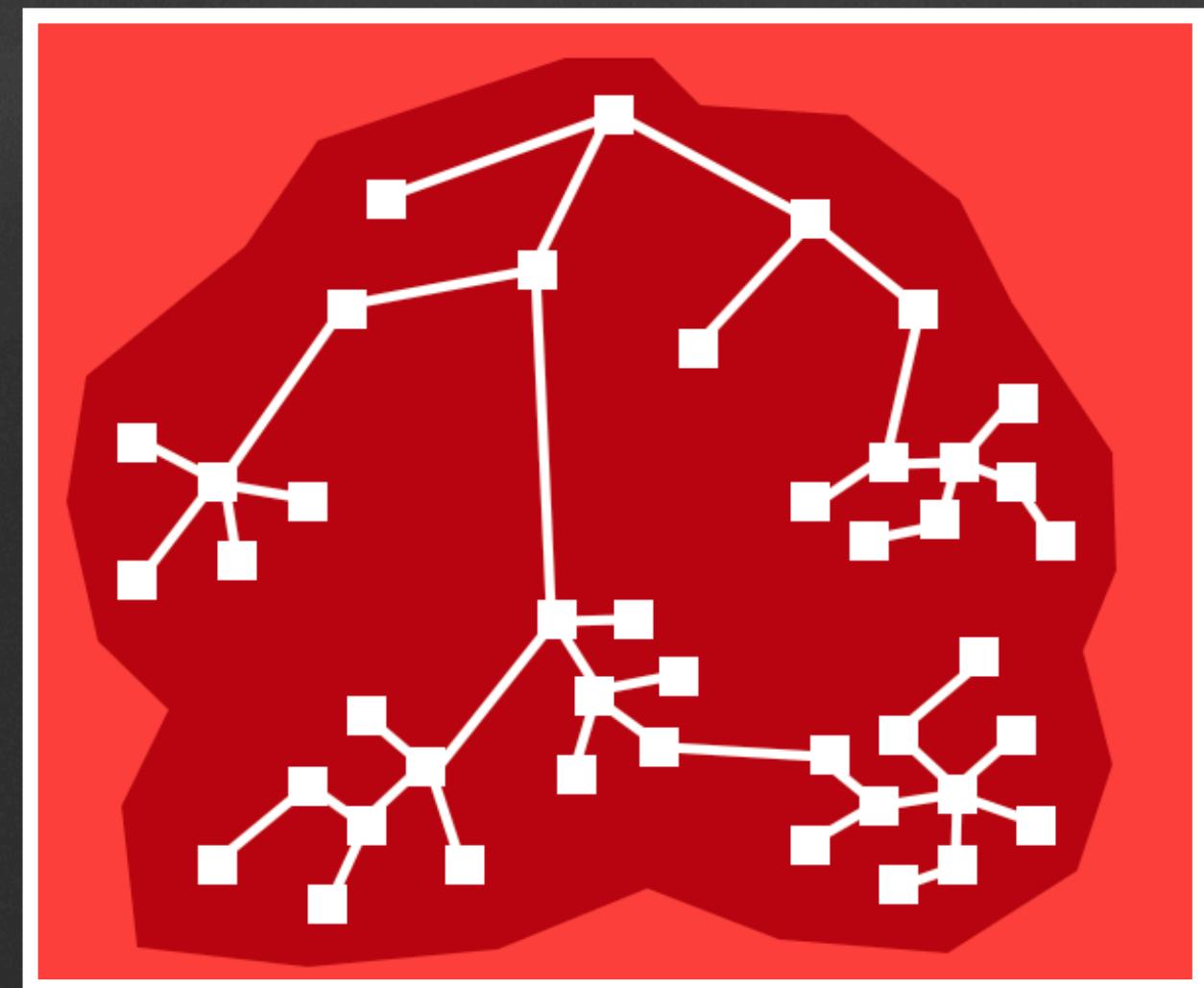
Ecosystem

- Vitest
- Storybook by Vite
- Vue.js, Angular and more



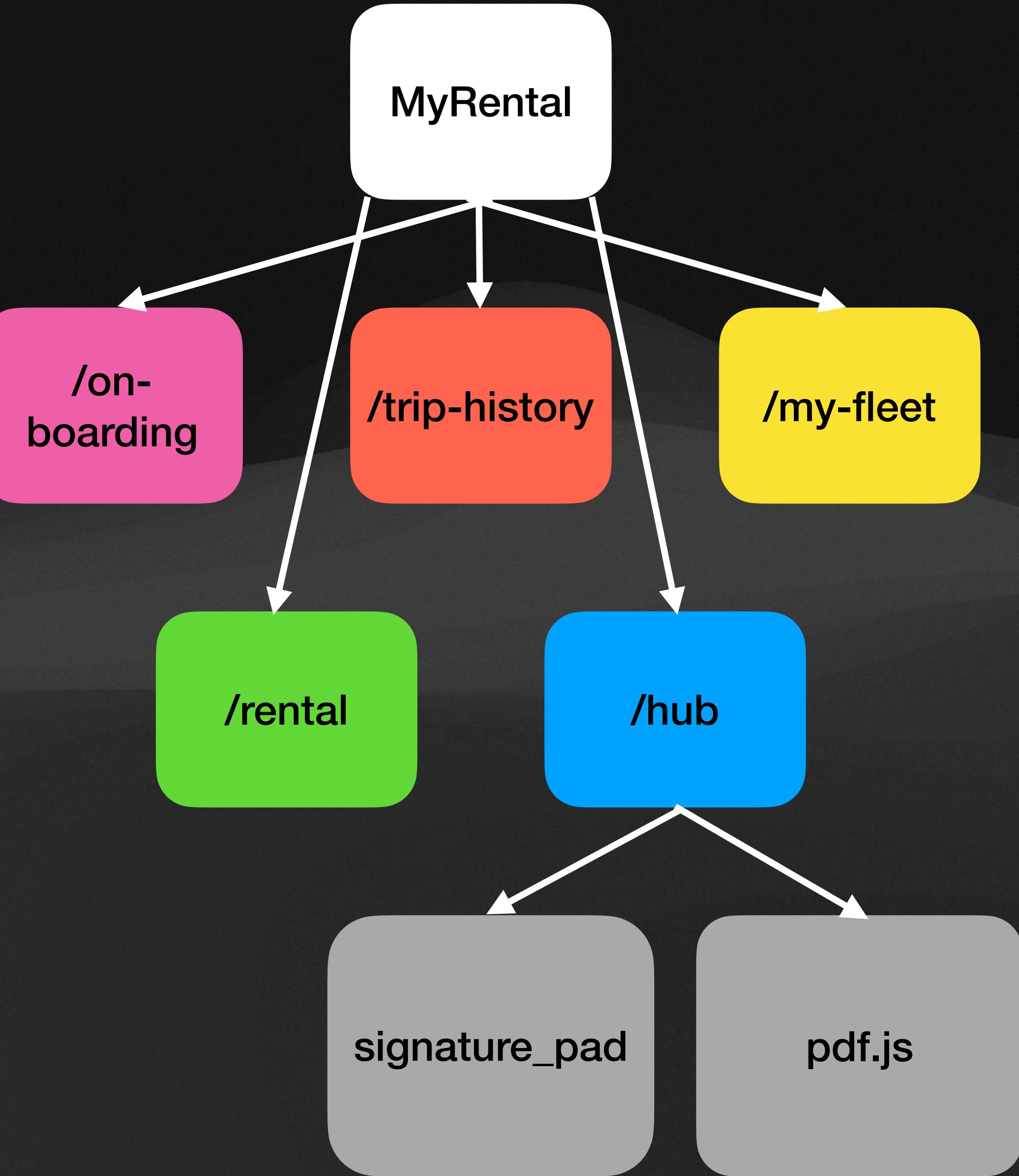
Code-Splitting

- Splitting app bundle into smaller chunks
- Only loading the code required
- Improve app's initial load



Splitting Strategy

- By endpoint
- By react-router + react.lazy
- Chunk size should be less than 500 kB



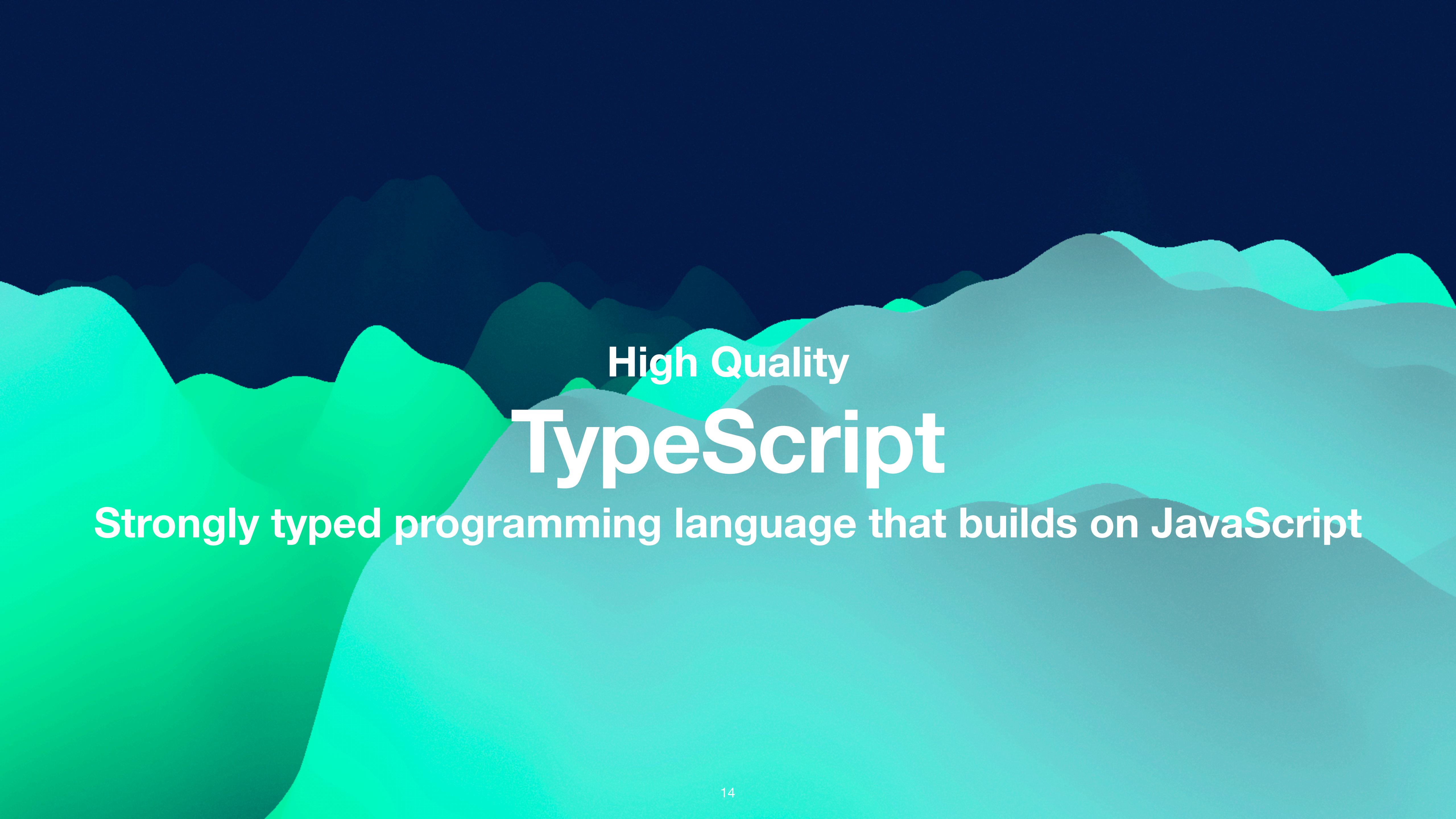
Demo

**Once you experience how fast Vite is,
you'd not be willing to put up with bundled development again.**

Vite Q&A



High Quality



High Quality TypeScript

Strongly typed programming language that builds on JavaScript

TypeScript

- Type checking
- IDE support, better DX
- Safety at scale and confident



TypeScript Cons

- 3rd party and framework support (~90%)
- Some learning curve
- Slow checking while project grows up



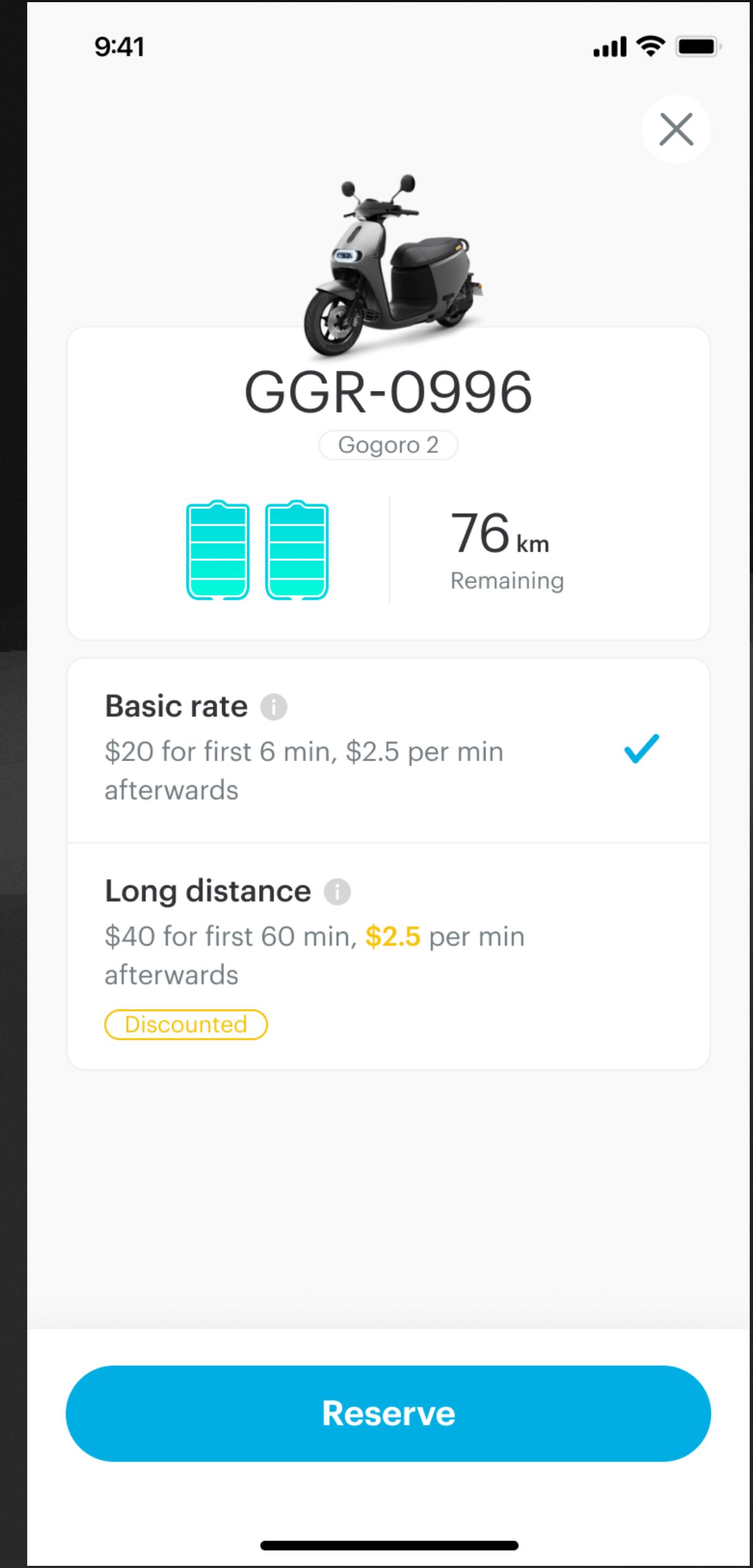
Who use TypeScript

- TypeScript was used by **78%** of the 2020 State of JS respondents, with 93% saying they would use it again.
- Google, Slack, Airbnb...
- Dcard, Line, KKBox...



TypeScript in MyRental

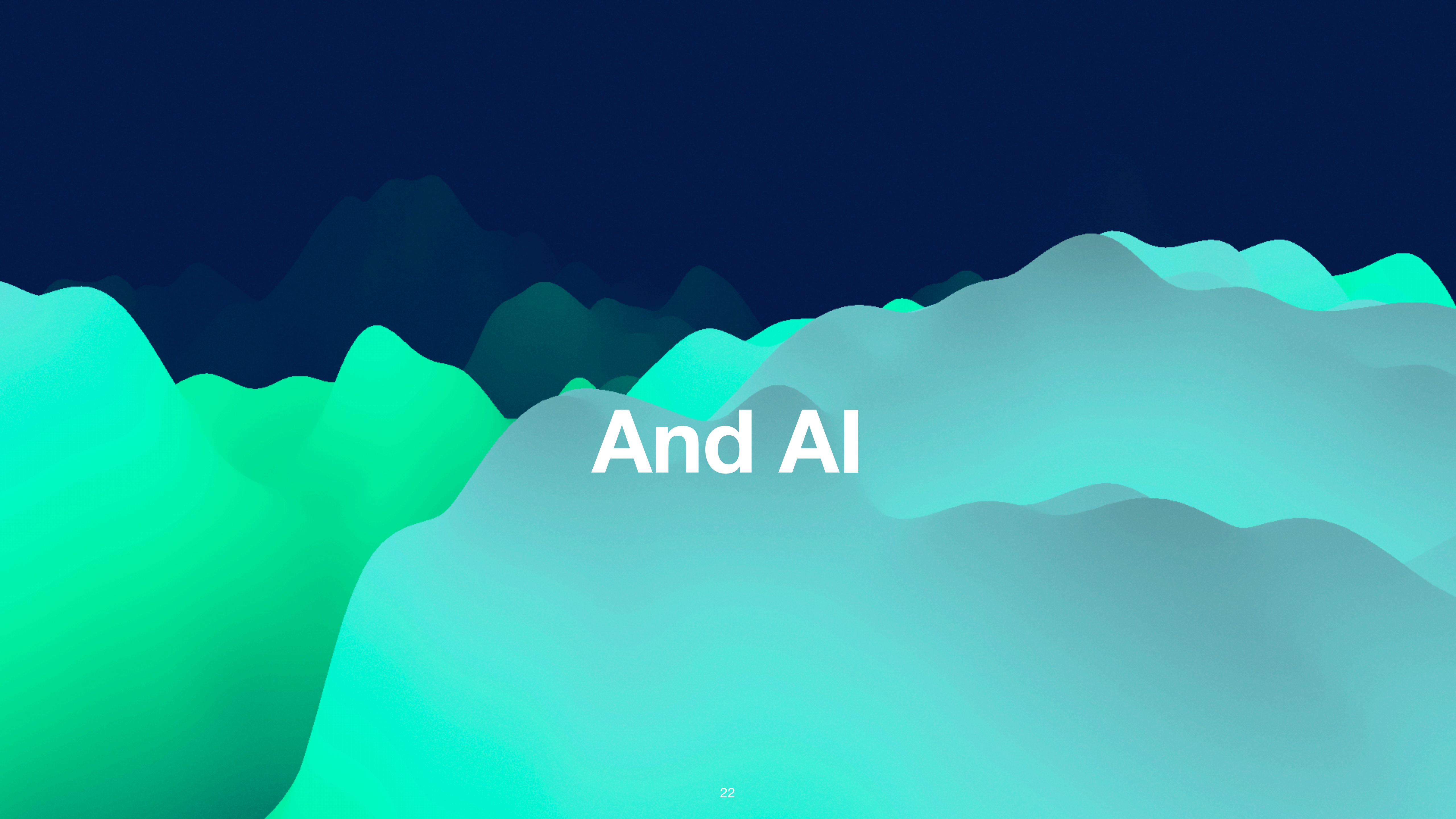
- Vite supports TypeScript files out of the box.
- Type checking before building project
- 100% TS and TSX



TypeScript User Experience

TypeScript Demo

TypeScript Q&A



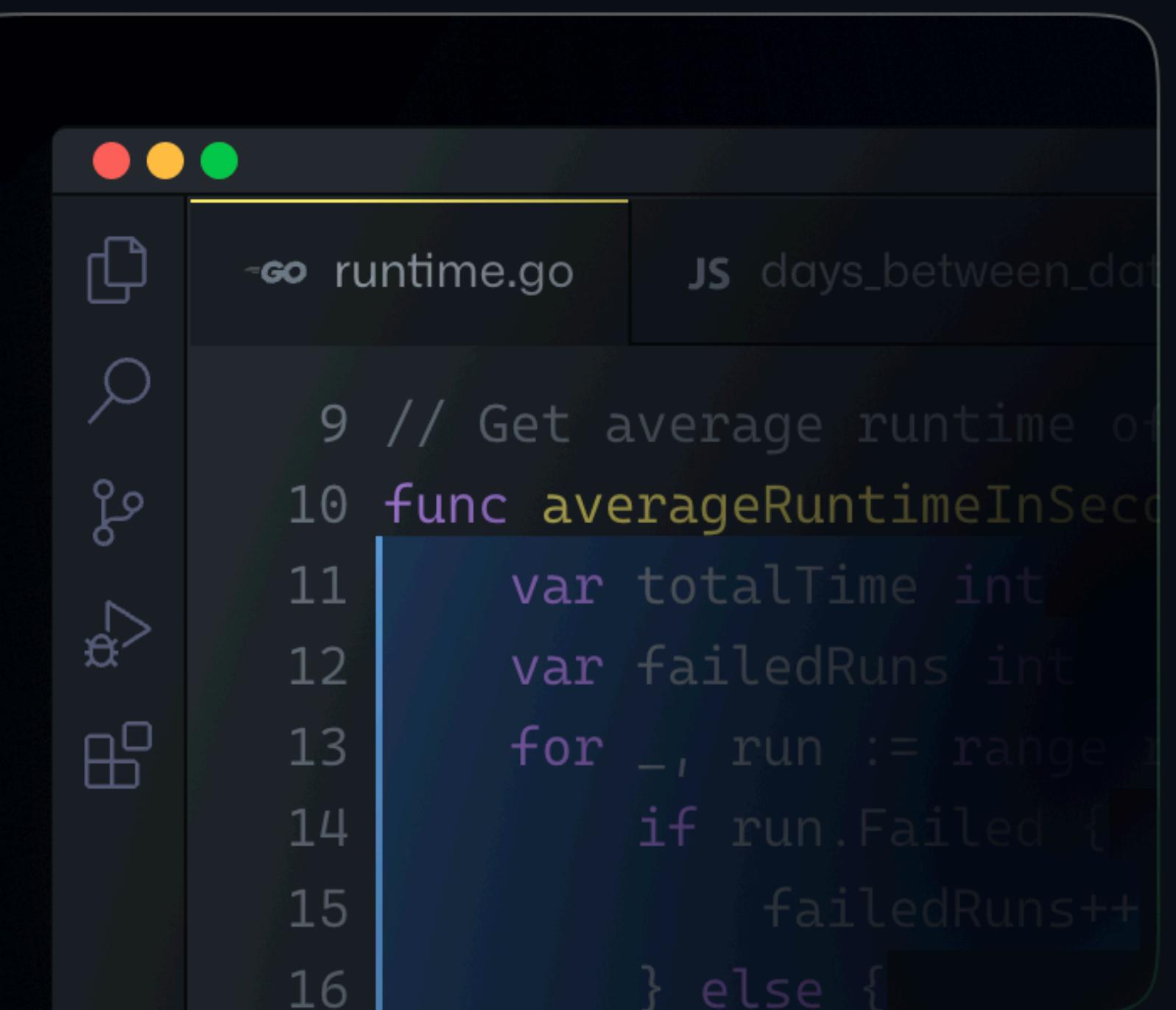
And AI



AI pair programming is here.

75% more fulfilled

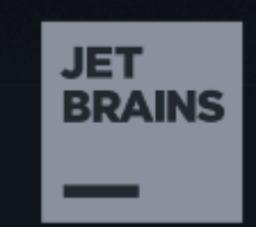
55%
faster coding



A screenshot of a code editor interface. On the left, there's a sidebar with icons for file, search, navigation, and others. The main area shows two tabs: 'GO runtime.go' and 'JS days_between_dat'. The 'runtime.go' tab is active, displaying the following Go code:

```
9 // Get average runtime of all runs
10 func averageRuntimeInSeconds() int {
11     var totalTime int
12     var failedRuns int
13     for _, run := range runs {
14         if run.Failed {
15             failedRuns++
16         } else {
```

Keep flying with your favorite editor



46%
code written



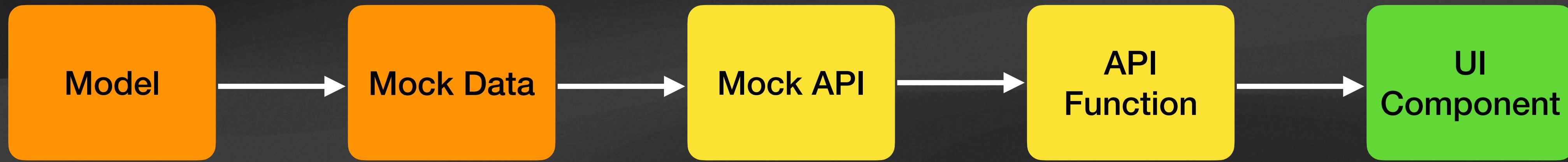
Now available for
all businesses

GitHub Copilot

- Integrate with VSCode
- Generate TypeScript code
- Generate unit testing



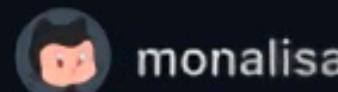
Copilot Working Flow



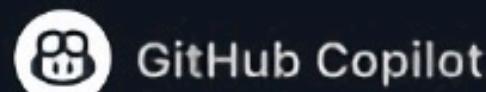


GitHub Copilot Chat

CHAT: GITHUB COPILOT



Write a set of unit test functions for the selected code



```
import unittest
import datetime

class TestParseExpenses(unittest.TestCase):
    def test_parse_expenses_with_valid_input(self):
        expenses_string = '2023-01-02 -34.01 USD\n'
        expected_output = [(datetime.datetime(2023, 1, 2), -34.01, 'USD')]
        self.assertEqual(parse_expenses(expenses_string), expected_output)

    def test_parse_expenses_with_empty_input(self):
        expenses_string = ''
        expected_output = []
        self.assertEqual(parse_expenses(expenses_string), expected_output)
```

module.py X

```
1 import datetime
2
3 def parse_expenses(expenses_string):
4     """Parse the list of expenses and return the list of triples (date, amount, currency).
5     Ignore lines starting with #.
6     Parse the date using datetime.
7     Example expenses_string:
8         2023-01-02 -34.01 USD
9         2023-01-03 2.59 DKK
10        2023-01-03 -2.72 EUR
11
12        """
13        expenses = []
14
15        for line in expenses_string.splitlines():
16            if line.startswith("#"):
17                continue
18            date, value, currency = line.split(" ")
19            expenses.append((datetime.datetime.strptime(date, "%Y-%m-%d"),
20                            float(value),
21                            currency))
22
23        return expenses
```

GitHub Copilot Demo

Q&A



**Fast Development
High Quality
AI**

And Our Gogoro Teams



Thank you for listening