

Rails API

(Rails for JSON enthusiasts)

On the agenda

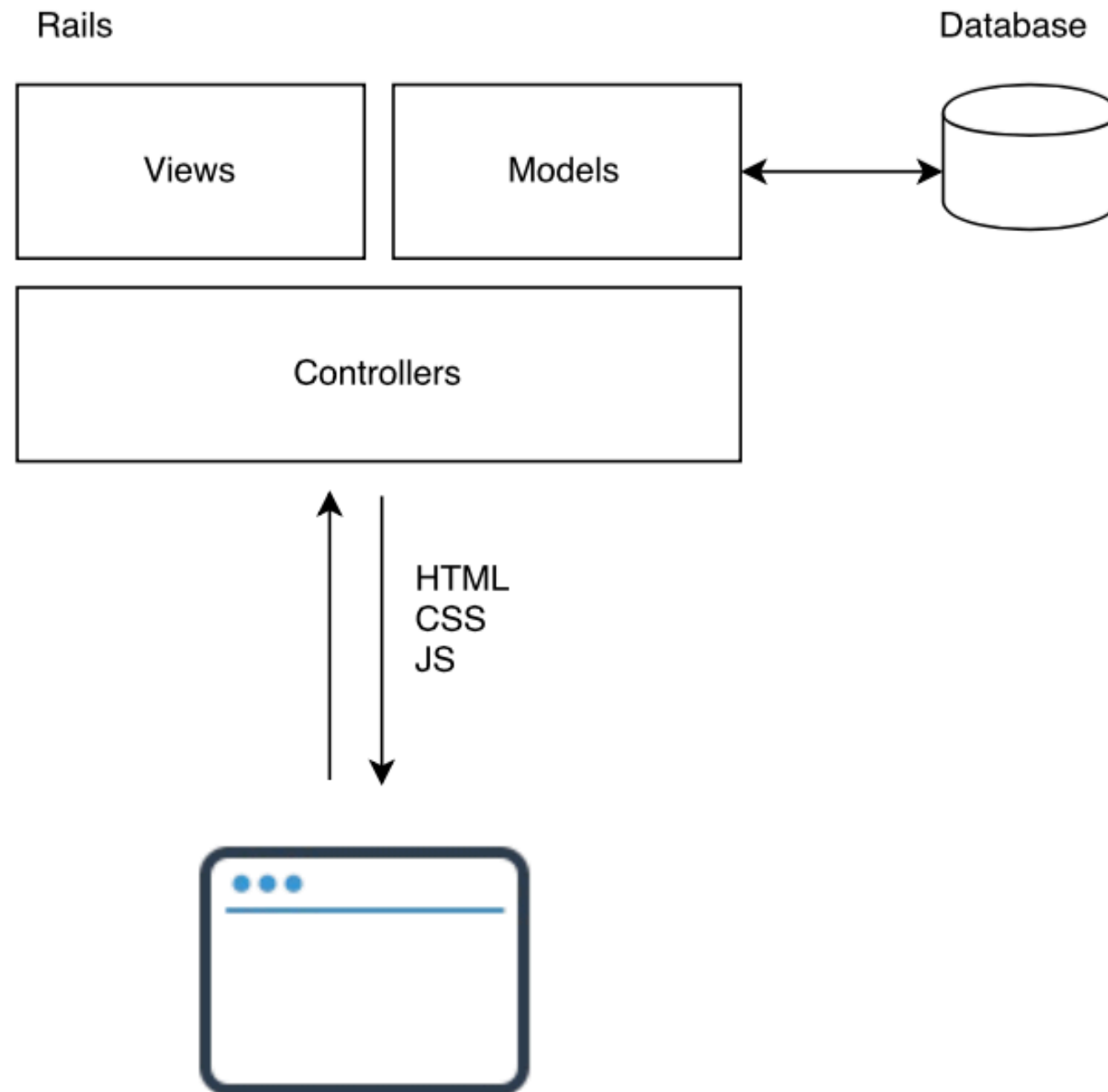
- A look at a common problem
- What is rails-api?
- Let's build something
- Caveats
- The future

Why do we need rails- api?

Let's look at a common scenario

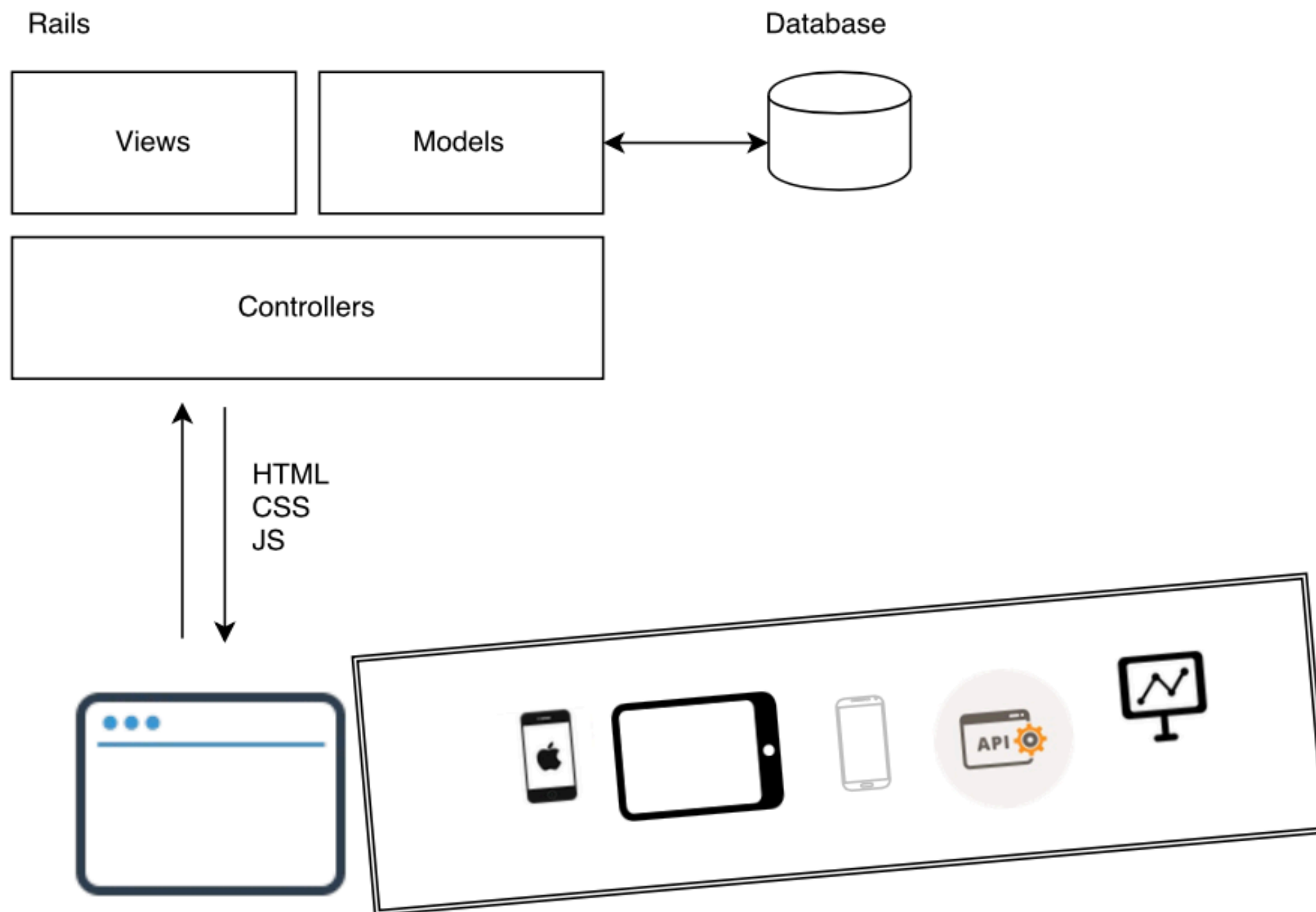
We need a **website**

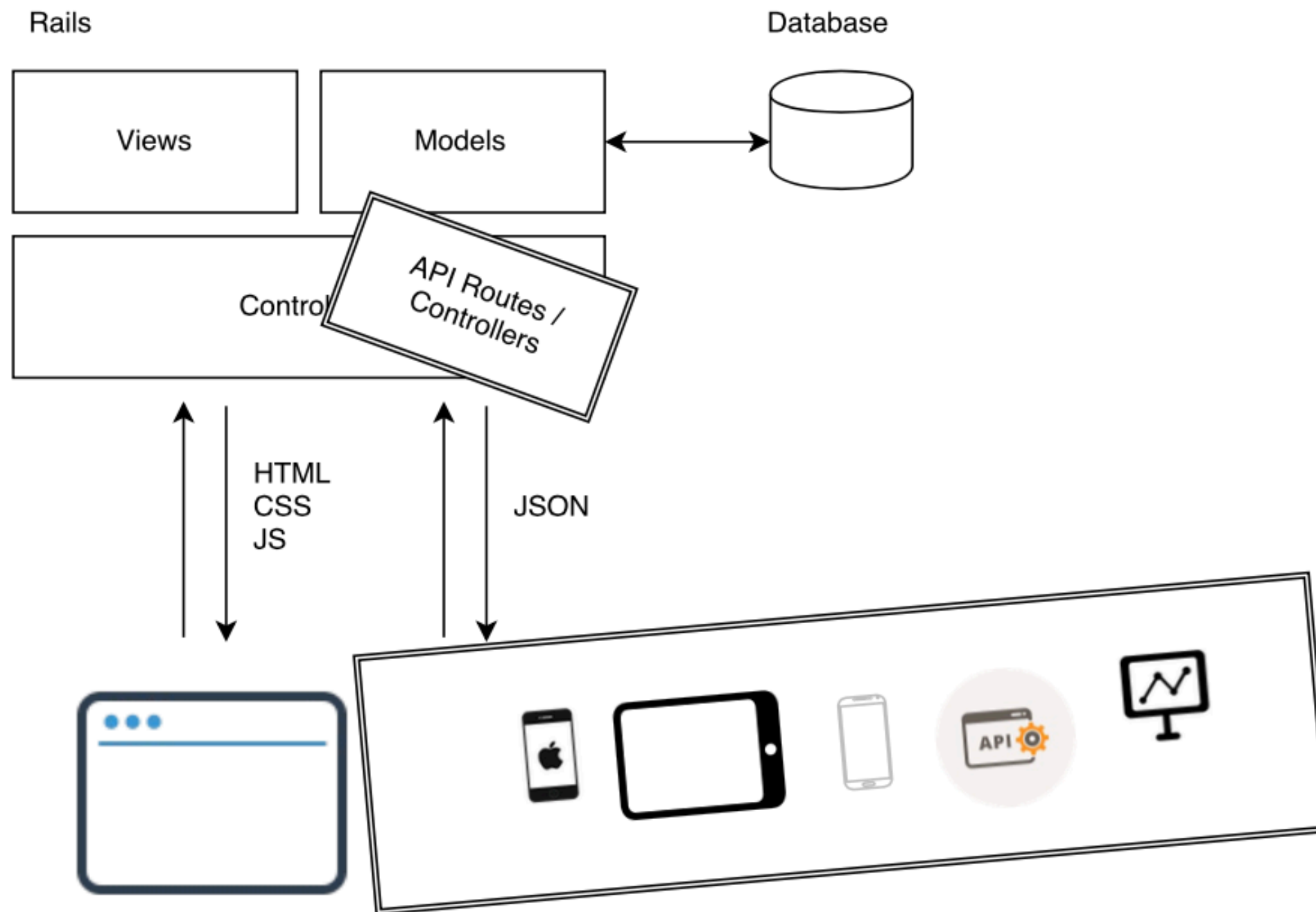
– **Day 1**

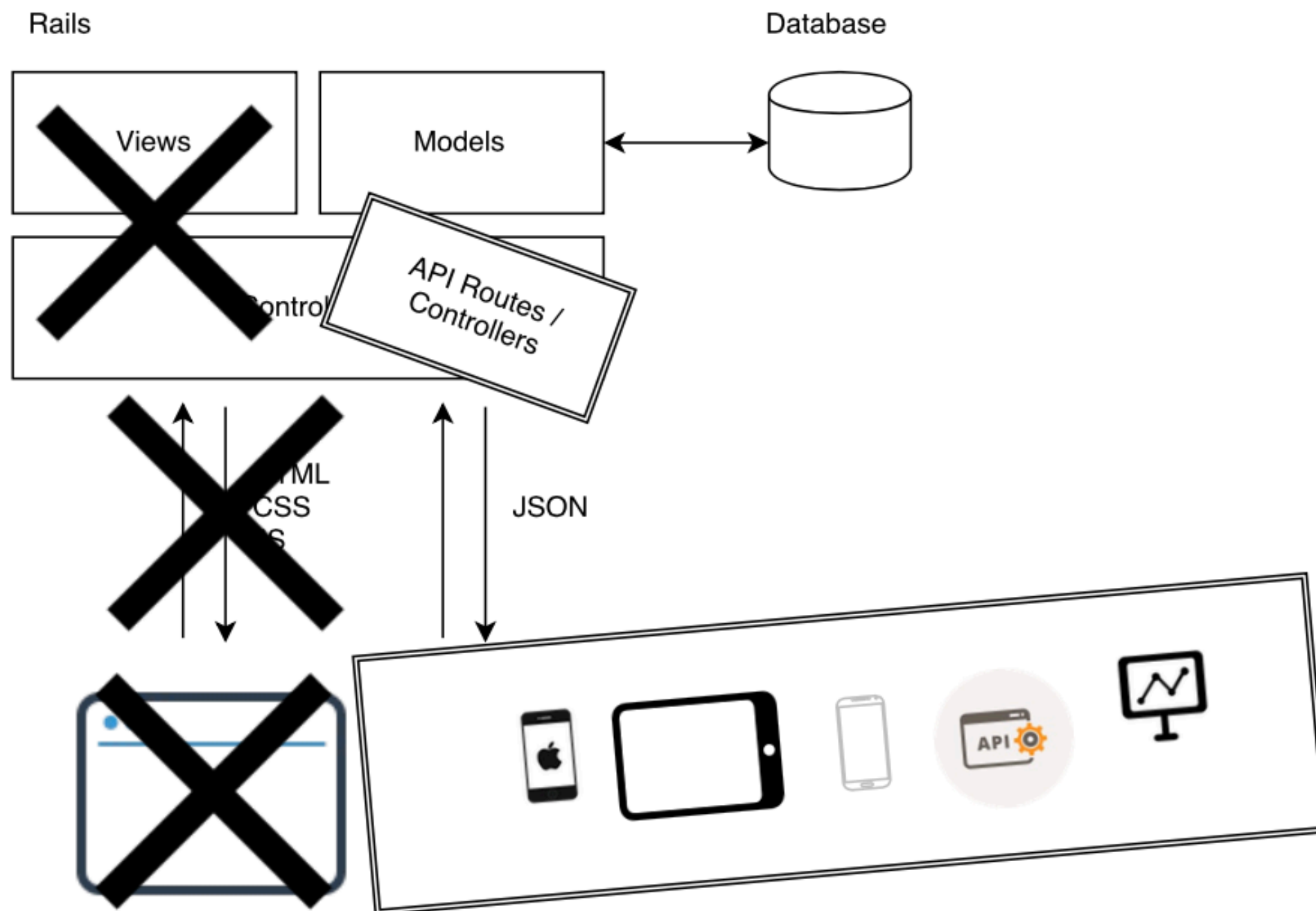


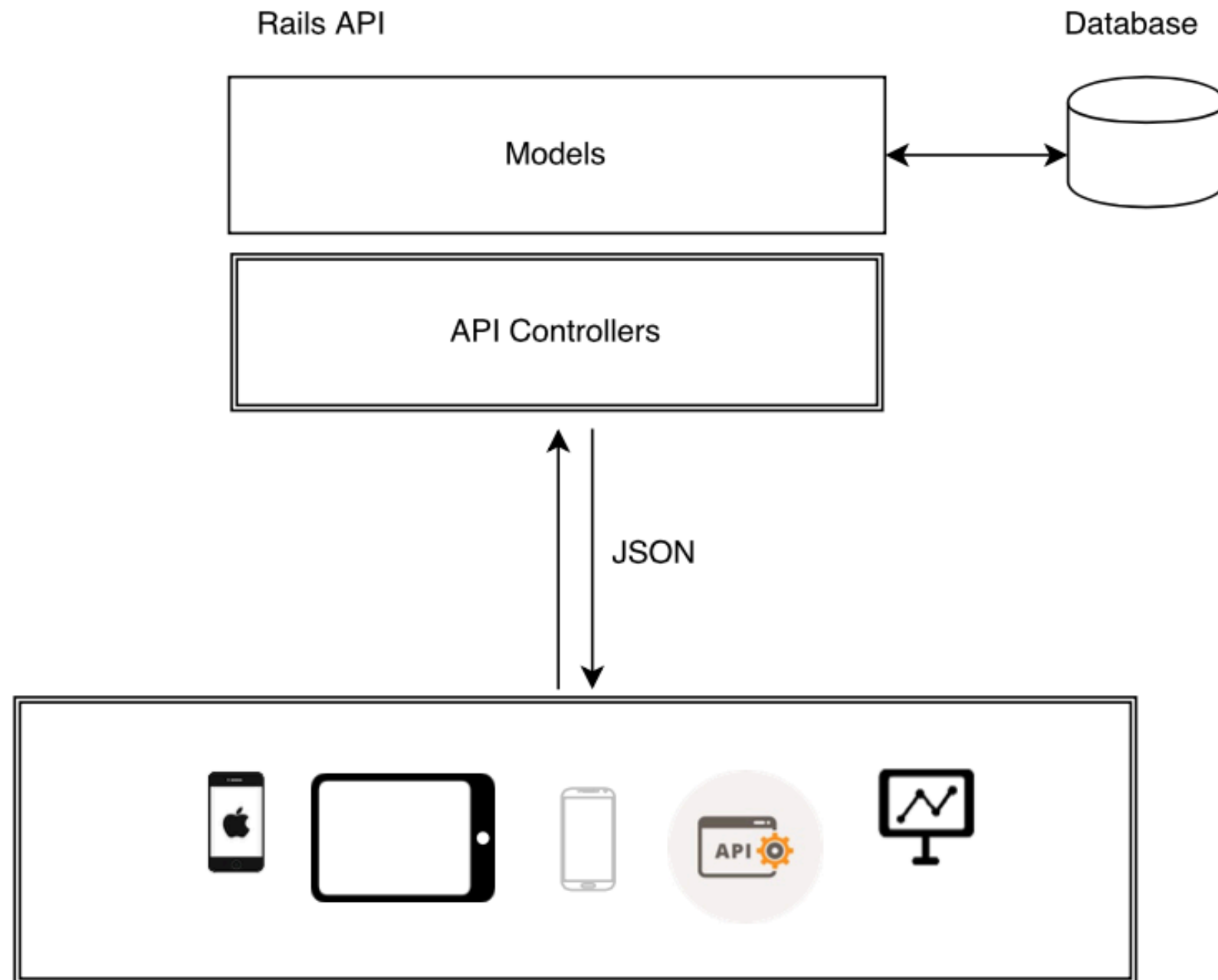
We need native **Android**
+ **iOS** apps and a public
API for **3rd party**
integrations

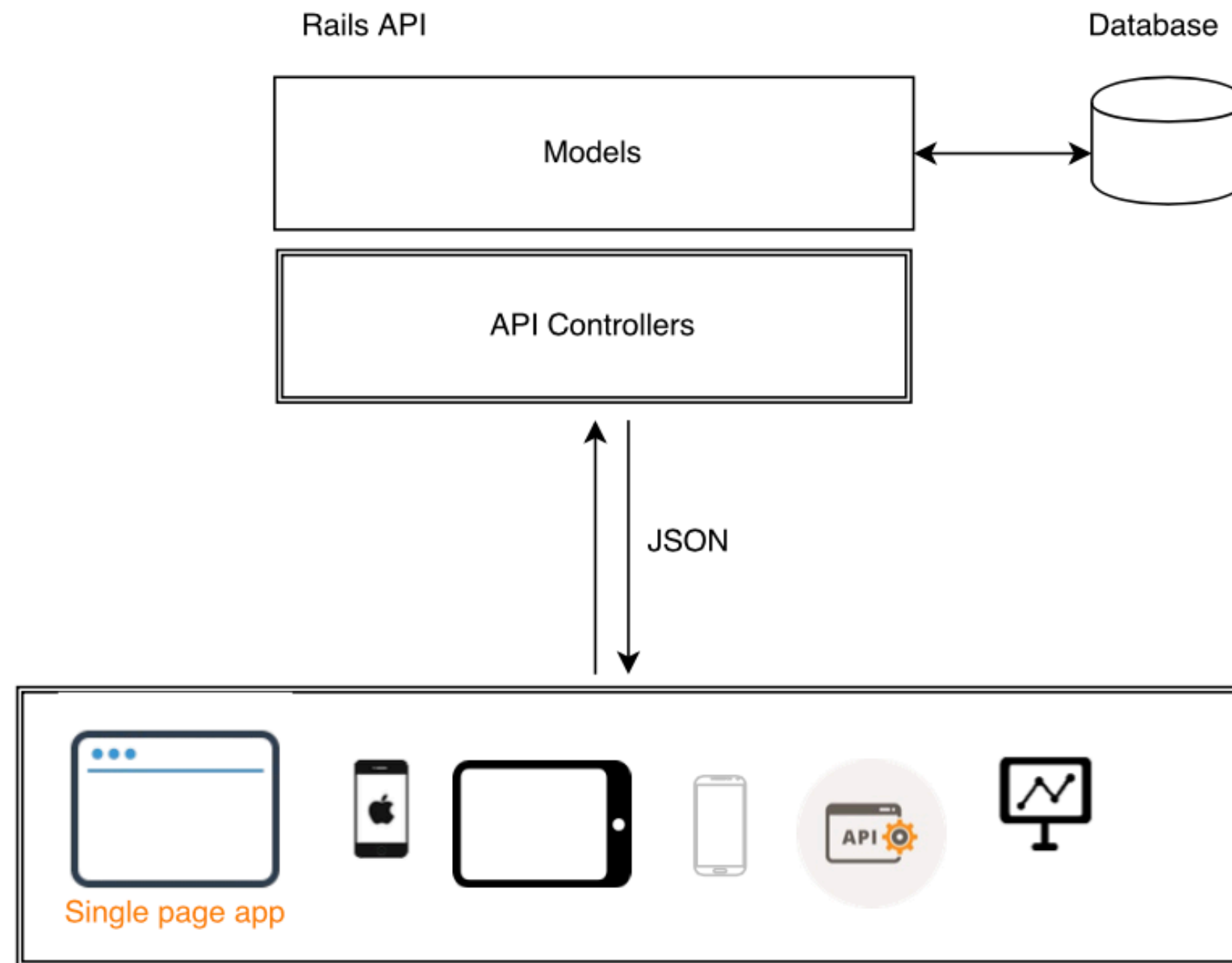
– Year 1

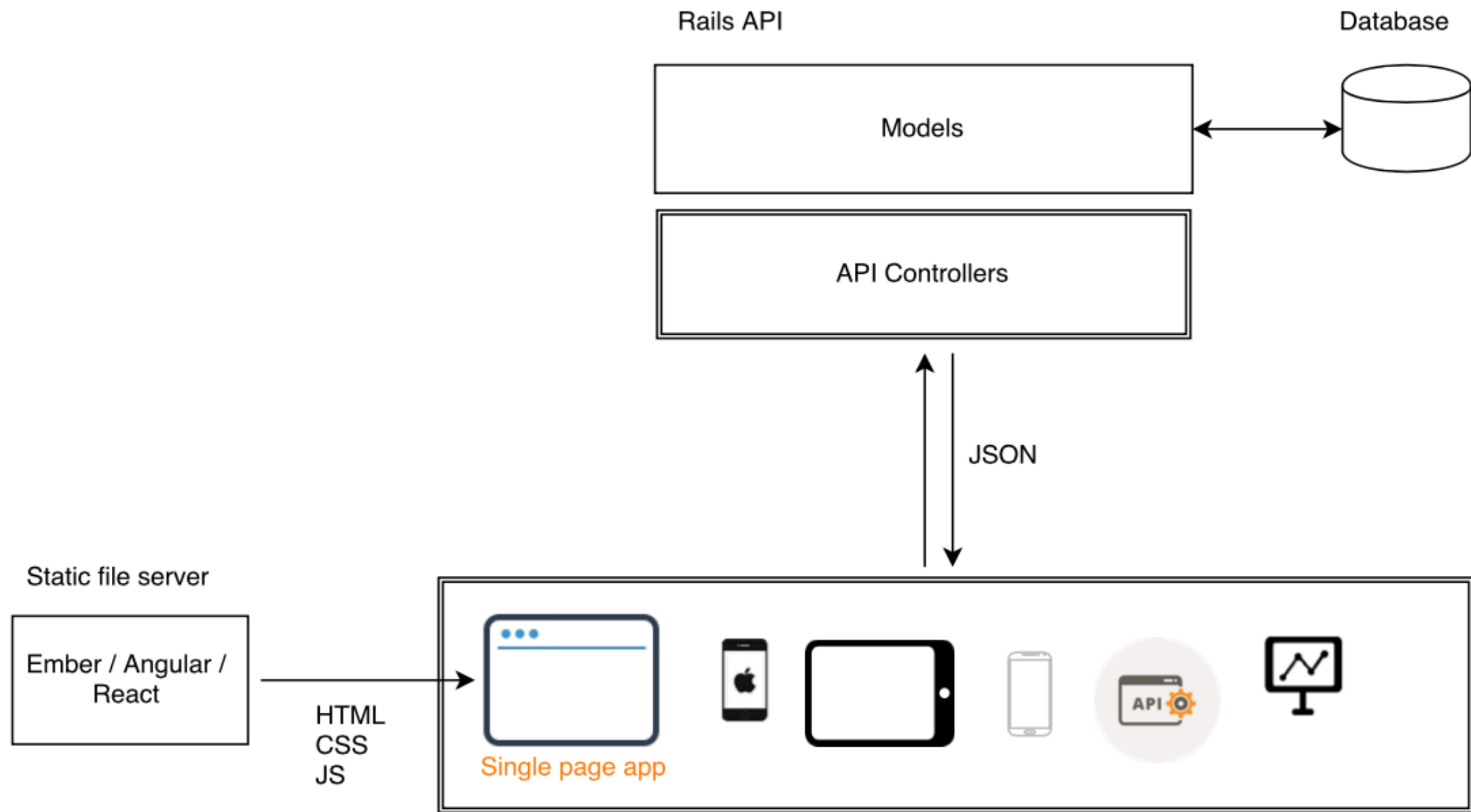












Key Benefits

- Separation of concerns
- Support multiple clients
- Scale dev teams
- API releases independent of view updates
- Performance*

What is rails-api?

Rails goodness but..

- No views
- No assets
- Fewer middleware
- JSON focused scaffolding

Let's build something

*Build an API that lets **users** manage **teams** and **players***

Install the gem

```
gem install rails-api
```

Create your app

```
rails-api new project_name
```

What do production APIs need?

- Serialization
- Pagination
- Error handling
- Authentication
- Authorization
- Versioning

Serialization

What does the payload look like?

```
{  
  "team": {  
    "name": "Manchester United",  
    "type": "team"  
  }  
}
```

Query Params

How do I filter collections?

Filters

GET /players?team_id=1

Options

GET /players?include_team=true

Pagination

How do I traverse large amounts of data?

```
GET /players?page=2
```

```
{
  "players": [
  ],
  "meta": {
    "total_pages": 3,
    "current_page": 2,
    "page_size": 10,
    "total_entries": 35
  }
}
```

Error handling

How do I know something went wrong?

```
{  
  "error": {  
    "message": "Something went wrong!"  
  }  
}
```

Authentication

Who's making the request?

HTTP Headers

Authorization: Token 137a9ba584314e0aaed7a7a64a2ac323

Authorization

Who's allowed to make the request?

HTTP Headers

Authorization: Token 137a9ba584314e0aaed7a7a64a2ac323

Versioning

How can we remain backwards compatible?

Legacy API, but still works :)

GET /v1/players

New and improved API

GET /v2/players

Wohoo!

Caveats

- Some gems don't work, especially with web views or cookies
- Sometimes monoliths are good!
- Single page apps are new (SEO, Load time)
- Rails 5 is coming

The future

- Merged into Rails 5!
- JSON API 1.0
- Single page web apps are going strong
- The internet of things

Thanks!

Bill Curtis, @billcurtis_

ps. We're hiring at Sweet Tooth!

