

# Webアプリ開発の基本

## 本学習の目的

2021年時点でのWebアプリ開発のトレンドを踏まえて、その基本的な知識を学びます。

## 対象者

- Webアプリ開発未経験者
- モダンなWebアプリ開発の経験が無い（下記キーワードを明確に説明できない、経験がない）開発者。
  - Grunt/Gulp/webpack, typescript, npm/yarn, AngularJS/React/Vue, Redux, node.js, ES20xx/ESNext, jest, eslint, prettier
- 脱 jQuery をしたいと思っている人、する必要はないと思っている人

## トレンド

トレンドとは、単なる流行りはなく、世界中のWeb開発者が必要に応じて"良い"と考えていることから、多くの人が利用し、情報を発信することでトレンドとなるものです。

もちろん、プロジェクトごとの状況に応じてトレンドのものが合わない場合もあるので、検討は必要です。

しかし特に理由がない場合は、多くの人が"良い"といっているものを採用するべきであり、それを利用しない場合はその理由を明確にすべきです。

開発者のスキルが伴っていない、という理由は理由足り得ますか、通常は低い優先度と考えます。（期限などの重要な他の理由がない限り、です）それを優先的な理由に持ってくると常にトレンドが取り入れられず、他社がトレンドを取り入れ高効率な開発を行う一方、その組織は低効率のまま、更に問題なのは低効率であることを知らないままの高コストな開発を続けてしまうことになります。

また、マイナーなものを選択してしまうと、技術的に詰まってしまった場合やトラブルが発生したときの情報が少なく、多くの工数を使ってしまうことにもなります。

開発者は、常にトレンドに注目し、見定めて、必要なものをキャッチアップすることが求められます。

## 動作環境と開発環境

Webアプリ開発では、様々なツールやフレームワーク、ライブラリを組合わせます。

条件や環境によって細かく入れ替えできるメリットがある反面、選択や準備が大変にもなります。組み合わせは無限です。それぞれ相性が良かったり悪かったりもあります。

トレンドを中止しつつ、すべてのものに対して造詣を深めるのは難しいので、いくつか試して良いと感じたものから習得してください。

本学習では、それぞれ類似性の高いものに関して、メジャーなものをそれぞれ比較することとします。

## 本学習の進め方

本学習は、フレームワークやライブラリを1つずつ、その概要を説明し簡単なワークショップによってその機能を確認します。

それぞれのさわりの部分（重要なコアな部分）のみの解説をします。それぞれ深堀りしたい場合は、Web等で情報を収集してください。

必要によっては、別途勉強会を開きます。

## 目次

1. アプリケーションのアーキテクチャ
2. パッケージマネージャ: npm / yarn
3. 言語: ECMAScript, TypeScript

4. タスクランナー/モジュールバンドラ: Grant, Gulp, Webpack
5. Webフレームワーク: AngularJS/React/Vue.js
6. リンター／コードフォーマッタ: eslint/prettier
7. テストツール: Jest