

Web 開発の特徴

中村 誠司, 大黒 智貴

Masashi NAKAMURA, Tomoki OKURO

1 はじめに

近年, スマートフォン保有率の増加に伴い, インターネットが急速に普及している¹⁾. 従来, Web サイトは静的なものであり, 文字や画像の閲覧が主な目的であった. しかし, 最近ではインターネットの回線速度の向上に伴い, 動的な Web サイトが増加している. また, Web ブラウザを介して利用するアプリケーションである Web アプリケーションが登場した.

Web サイトや Web アプリケーションの高機能化に伴い, プログラムの実装が複雑になる. しかし, 開発期間が限定されているため, より効率の良い開発手法が求められている. 現在, Web 開発速度を向上させる仕組みとしてフレームワークを用いる手法が一般的である. フレームワークとは, 開発を行うためにあらかじめ指定している枠組みのことである. 目的に応じた Web 開発を行うために, 様々なフレームワークが開発されている.

2 Web 開発

2.1 概要

Web 開発とは, Web サイトや Web アプリケーションを開発することである. 一般的に, Web 開発における開発領域は, フロントエンドとバックエンドに分割できる. Fig. 1 にフロントエンドとバックエンドの関係を示す.

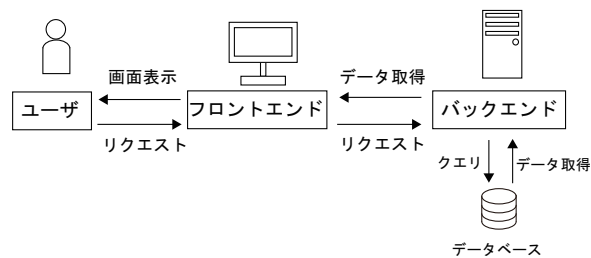


Fig.1 フロントエンドとバックエンドの関係

フロントエンドとは, ユーザからのリクエストを受け, ユーザが閲覧する表示画面の実装を行う部分である. また, バックエンドとは, フロントエンドからリクエストを受けて処理を行い, 結果を返す部分のことである. バックエンドは必要に応じて, データベースや他の Web サーバとの通信を行う.

2.2 フロントエンド開発

フロントエンド開発とは, ユーザが閲覧する表示画面の開発を行うことである. フロントエンド開発において, 使用する言語は HTML, CSS, JavaScript の 3 つである.

HTML は, Web ブラウザが表示する内容や構造を記述する言語である. CSS は, Web ページの配置や配色に使用する言語である. JavaScript は HTML, CSS の制御に用いられ, Web ページ上で動作の表現や計算を行う言語である. ユーザのデバイスのみで, タスクを処理できるため, 素早いレスポンスが可能となる.

2.3 バックエンド開発

バックエンド開発とは, データベースやサーバ上での演算処理, 他のサーバへのアクセスを行うプログラムを開発することである. サーバやアプリケーションを相互連携する際には, Ruby や PHP, Python などの言語を使用する. プログラミング言語にはそれぞれ異なる利点や特徴が存在するため, 用途に応じたプログラミング言語を選択する必要がある.

2.4 MVC アーキテクチャ

従来の Web 開発手法では, すべてのソースコードを同一の場所に記述する必要があった. そのため, システムが複雑になるにつれてコードが膨大になり, プログラムのテストや開発における役割分担が容易ではなくなった. そこで, これらの問題を解決するために考案されたアーキテクチャが MVC アーキテクチャである. Fig. 2 に MVC アーキテクチャの相関図を示す.

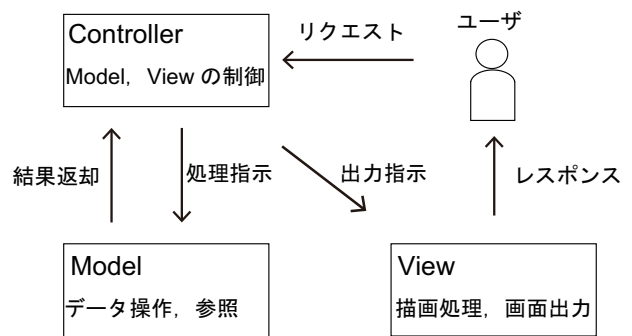


Fig.2 MVC アーキテクチャの相関図

MVC アーキテクチャは, 処理を Model, View, Controller の 3 つの役割に分割して実装する方法である. Model はシステムの中で処理の中心となるものである. データベースに格納されたデータを抽象化し, API を通じて, Controller からの処理指示に応答する. Controller は, ユーザからのリクエストを受け, Model に処理指示や View に出力指示を行う. View は返却された処理結果をもとにユーザが閲覧する画面を出力する. MVC アーキテクチャにしたがってプログラムを実装することで, 開発

作業の役割分担が可能となる。そのため、それぞれの専門分野のみを担当することや開発作業の並行処理が可能となる。また、仕様変更の際には、対象となる部分のプログラム以外は変更せずに利用できるため柔軟に対応できる。これらのことから、MVC アーキテクチャを使用することで、Web 開発における開発速度を向上させることができる。

3 Web 開発の動向

3.1 LAMP

Web アプリケーション開発では、従来 LAMP 環境を使用していた。LAMP とは、OS の Linux、Web サーバの Apache HTTP Server、データベースの MySQL、プログラミング言語の PHP/Perl/Python の頭文字をとった開発環境を指す言葉である。Linux は無料で改変可能なオペレーティングシステムである。Apache HTTP Server は Web サーバソフトウェアである。豊富な機能や高い安定性が特徴であり、世界でもっとも多く利用されている。MySQL は、リレーショナルデータベースマネジメントシステムの 1 つである。LAMP では、すべてのソフトウェアがオープンソースであるため、調達コストを抑えることができ、改善が容易である。しかし、保証やサポートがなく、運用管理にそれぞれの専門的な知識が必要となる。そのため、運用コストが大きくなる可能性がある。

3.2 MEAN

MEAN とは Web アプリケーションを構成する MongoDB, Express, AngularJS, Node.js の頭文字をとった開発環境を示す語である。従来、データベースへのアクセスやバックエンドでの処理はそれぞれ異なる言語で実装していた。そのため、専門的な知識を持った人がそれぞれの部分の実装を行う必要があった。MEAN では、フロントエンド、バックエンドの両方を JavaScript で記述できる。そのため、開発における役割分担が容易となった。Fig. 3 に MEAN スタックの相関図を示す。

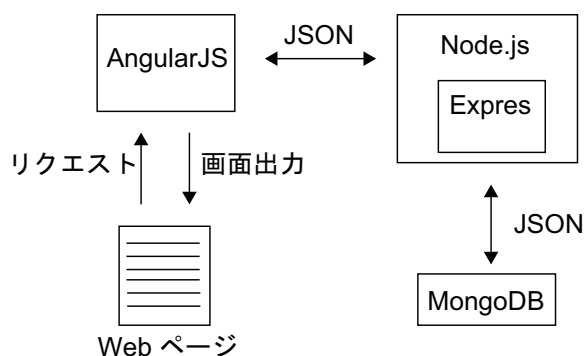


Fig.3 MEAN スタックの相関図

Node.js はバックエンドで用いる JavaScript 環境である。Node.js を利用するとフロントエンドの言語のみを実装できる人であったとしてもアプリケーション全般を実装することができるので、適切な人材配置ができ、開発速度

が向上する。また、非同期 I/O を扱うため、1 つのスレッドで大量の処理を行うことができる。Node.js はシングルスレッドであるため、1 つの処理が原因で全体の実行速度が遅れる可能性がある。そのため、ループを止めないようにプログラムを書く必要があるため、CPU 負荷の高い処理には向いていない。

MongoDB は、ドキュメント指向のデータベースである。MongoDB は、一般的なリレーショナルデータベースとは異なり、事前にテーブルの構造を決定する必要がない。そのため、データ構造は自由であり、データを追加する度、データ構造を変更できる。しかし、MongoDB では、SQL の使用が不可であり、関係性を利用して、データをまとめて処理することは容易ではない。また、整合性はリレーショナルデータベースの方が良い。

AngularJS はフロントエンドで用いる JavaScript のフレームワークであり、View を構成するために使用する。また、MEAN におけるすべてのデータ通信は軽量のデータ交換フォーマットである JSON(JavaScript Object Notation) を用いて行われる。JSON は、言語から独立したテキスト形式であるため、開発環境を問わない。そのため、保守性の高いシステムを作成することが可能である。

Express は、Node.js 上で動作する MVC フレームワークである。Express は、フロントエンドの AngularJS からリクエストを受け取り、必要に応じて MongoDB にアクセスする。

MEAN の実用例として、オンライン決済サービスを提供している PayPal は Node.js を利用して、Java で実装していたコードを JavaScript で実装した。使用変更に伴い、以前に比べページの平均応答時間が減少している²⁾。

4 今後の展望

今後もモバイル端末は普及していくと考えられる。そのため、モバイル機器に特化した Web サイトの記述方法が普及すると予想される。また、インターネット環境の発展に伴い、Web サイトや Web アプリケーションの高機能化が想定される。したがって、Web 開発において用いられるプログラムはより複雑になる。そのため、Web 開発を短期間で効率的に行うための技術が発展すると考えられる。また、現在ソフトウェアの振る舞いが想定した通りであるかの検証を自動化するようなフレームワークが広がりつつある。よって、今後テストの高速化が実現されると予想される。

参考文献

- 1) 総務省, <http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h28/html/nc252110.html> 参照 Apr.8, 2017
- 2) PayPal, <https://www.paypal-engineering.com/2013/11/22/node-js-at-paypal/> 参照 Apr.25, 2017