

Node.jsを用いた出欠管理システムの設計

315156 續橋 涼

1. はじめに

現在, 研究室での学生の卒業研究の出欠状況を用紙に記録して管理している. 出欠状況を確認する際, 用紙に書かれたデータを読み取り, 集計する必要があり, 負担が増えるため効率的にデータを管理することが好ましい.

出席管理を支援するサービスは多く存在する. 何かサービスを設計, 移行する際, まずクラウドサービス¹⁾を第一に検討するクラウドファーストという方針が普及してきている. クラウドサービスを利用することで簡単に出席管理システムを利用することが可能になる. しかし今後, 研究室で運用していく中で卒業に伴う管理者の交代や研究グループの変更などに対応するためには, 高度な知識を必要とするため好ましくない. 以上の背景から比較的学習と管理が容易なNode.js²⁾を利用して出席管理システムの設計を行った.

2. システム構成

本研究では, 毎年管理者が変わるという研究室の特色上, システム構成には以下の3点を考慮し設計した.

- (1) 学習コストの低さ
- (2) 理解しやすい構造
- (3) 運用のし易さ

これらを満たすためNode.jsを利用しサーバーを構築する. Node.jsはJavaScriptのランタイム環境であり, サーバーサイドとフロントエンドをJavaScriptで言語を統一できるため新規の学習コストを低くすることができる. また言語自身がサーバーの機能を持っているためNginxやApacheなどのWebサーバーを必要としないため運用がしやすくなる. さらにnpm(Node Package Manager)と呼ばれるNode.jsを管理するシステムが存在し, これらを利用することにより生産性を高めることができる. Webアプリケーションを効率的に開発するためにExpressというMVC(Model View Controller)フレームワークを用い, RDB(リレーショナルデータベース)としてSQLite3, DB(データベース)をJavaScriptで管理するためのモジュールとしてBookshelf.jsをJavaScriptのテンプレートエンジンとしてEJSを用いて図1に図示したシステムの設計を行う.

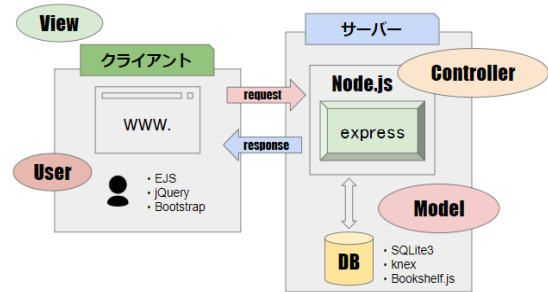


図1 システム構成

3. データベース

本研究でデータベースはユーザーテーブル, 予定テーブル, ユーザーの予定テーブル, コメントテーブル, ユーザーデータテーブル, 1日ごとのデータテーブルから構成される. 図2に各テーブルに定義された主なフィールドおよびテーブル間の関係を示す. これらのテーブルはSQLite3にてGUIツールであるDB Browser for SQLiteを用いて定義し, クエリービルダーと呼ばれるknex.jsを用いてSQLのクエリーを作成し, Bookshelf.jsを用いてJavaScriptで実行されたメソッドをSQLに変換しデータベースに問い合わせを行う.

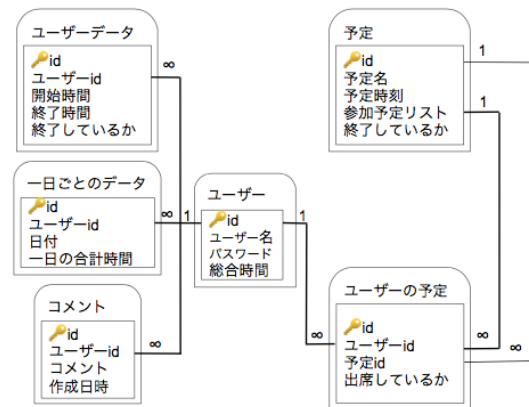


図2 テーブル間の関係

4. ユーザーインターフェース

本システムはJavaScriptのテンプレートエンジンであるEJSを用いてHTMLのページを動的に生成し, HTMLのFORMを利用してサーバーにリクエストを送っている.

5. 結果

5.1. 出席登録

開発したシステムを学生が利用する例について説明する。

- (1) DBに登録されているユーザー名とパスワードを入力してログインする。
- (2) ログインに成功すると、TOPページに移動する。TOPページにあるマイページを選択する。
- (3) マイページから予定を確認することができる(図3)。
- (4) 予定が始まってから30分以内までは出席することができる(図4)。
- (5) 出席ボタンをクリックすると、出席になる(図5)。



図3 マイページ(学生用)



図4 予定参加



図5 予定参加後

5.2. 予定の作成と出欠データの閲覧

管理者権限をもつユーザーが利用する例について説明する。

- (1) DBに登録されているユーザー名とパスワードを入力してログインする。
- (2) 一般ユーザーと異なり予定作成のリンクがTOPページに表示されるので予定作成のボタンをクリックする。
- (3) 予定作成ページにてタイトルと日付を入力し参加するメンバーを選択し作成ボタンをクリックすることで予定が作成される(図6)。
- (4) 再びTOPページから予定一覧を選択することで予定一覧を閲覧することができる(図7)。
- (5) 閲覧したい予定名をクリックすることでその予定の出欠状態を閲覧することができる(図8)。

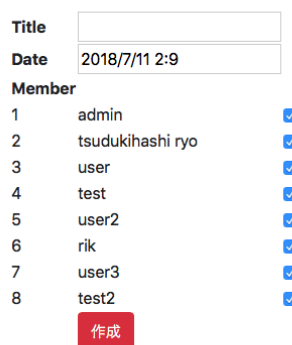


図6 予定作成



図7 予定一覧



図8 出欠状態

6. 考察

全てJavaScriptを利用してコードを記述することができたため、シンタックスを気にすることなくシステムの構成やデータベースのモデル、ルーティング、ミドルウェアに集中して取り組むことができたためNode.jsのメリットが活かされた。

7. おわりに

本研究ではNode.jsを利用して研究室での出欠管理システムの開発を行った。今後の課題として毎週行うスケジュールは自動生成させ、予定作成の負担を減らす必要がある。また作成した出欠管理システムをスマートスピーカーを用いて音声のみで動作させることで、利便性を高めることができる。

8. 参考文献

- 1) 天野滉久, 平石広典, "行動履歴に基づく居室者状況予測システムの設計", 情報処理学会第76回全国大会, Vol. 3, pp. 195-196, 2014. 3.
- 2) 掌田津耶乃, "Node.js 超入門", 秀和システム, 2017.8.