

# Databaseを用いた生徒会選挙システム

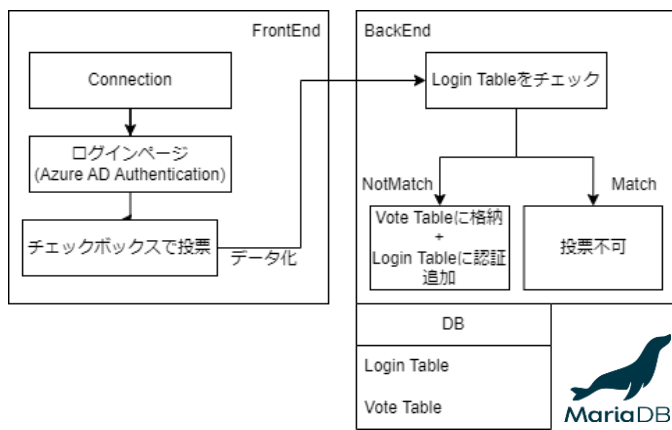
茗溪学園高等学校 1年 沼木碧泉

## 【背景】

様々なものがデジタル化している現代社会において、これまで紙での投票で行っていた生徒会選挙が、オンラインで投票できるようになれば投票も集計も楽になるのではと思い作成した。Google Formだと回答を一回のみにしてしまうとメールアドレスと選挙の内容が一緒に保存されてしまい、誰が誰に投票したかわかってしまうため自作システムとして開発した。

## 【方針】

Front: Vite + React Back: Express.jsでの開発



## 【実践結果】

9月25日に生徒会選挙が行われ、上記のようなデータがDB上に記録された。

以下のデータの他にメールアドレスを保存するDBも存在し、そちらのデータも記録することができた。

chairperson	accountingauditor	accountant	secretary	vicepresident	president	submit_date
●	●	●	●	●	●	2023-09-25 13:22:34
●	●	●	●	●	●	2023-09-25 13:22:34
●	●	●	●	●	●	2023-09-25 13:22:40
●	●	●	●	●	●	2023-09-25 13:22:54
●	●	●	●	●	●	2023-09-25 13:23:02
●	●	●	●	●	●	2023-09-25 13:23:08
●	●	●	●	●	●	2023-09-25 13:23:10
●	●	●	●	●	●	2023-09-25 13:23:12
●	●	●	●	●	●	2023-09-25 13:23:14
●	●	●	●	●	●	2023-09-25 13:23:17
●	●	●	●	●	●	2023-09-25 13:23:19
●	●	●	●	●	●	2023-09-25 13:23:19
●	●	●	●	●	●	2023-09-25 13:23:24
●	●	●	●	●	●	2023-09-25 13:23:26
●	●	●	●	●	●	2023-09-25 13:23:27
●	●	●	●	●	●	2023-09-25 13:23:27
●	●	●	●	●	●	2023-09-25 13:23:27
●	●	●	●	●	●	2023-09-25 13:23:27
●	●	●	●	●	●	2023-09-25 13:23:27

※個人情報を含むためモザイク加工済

## 【ソフトウェア構成】

### ①ログイン

Azure AD Authenticationを使用し、学校内のメールアドレスのみに制限した。

### ②投票

信任と投票及び複数選択が存在する。

このフォームは、ReactのReact-hook-formを使用し作成した。

また、このページでBackendとの通信も行っている。



### ③DBとの通信

Axiosを使用してPOST通信とGET通信を確立させた。GET通信で、DB上のLogin Tableにログインされているメールアドレスが存在するか調べる。

存在しない場合は、POST通信で選挙の選択データのJSオブジェクトとメールアドレスをBackendに送信し、SQL文に変換して格納している。

サーバー情報:

DB: MariaDB(Docker)

サーバー: 自宅サーバー (メイン)

GCP、AWS (HTTPS通信のためのStream先)

## 【課題と今後の展望】

実践の結果、投票システムとしては機能することがわかった。しかし、アクセス集中でのTimeOutなどのサーバーサイドの問題などが目立った。

来年度は、今回の不具合の解消及び開票専用ページなどの作成を行っていきたい。

## 【参考文献】

<https://qiita.com/ryuichi-f/items/392fae9ed74eeba5c397>

Azure ADの認証をReactアプリケーションに追加してみた。

<https://qiita.com/nemutya/items/b4c606168aa5be610e1e>

React+Node.js(Express)でMySQL連携する