

基本設計書

elsy0111*

2024年11月7日

1 はじめに

本設計書では、企業とエンジニアのジョブマッチングを円滑に行うためのシステムの基本設計について記述する。

2 総括ダイアグラム

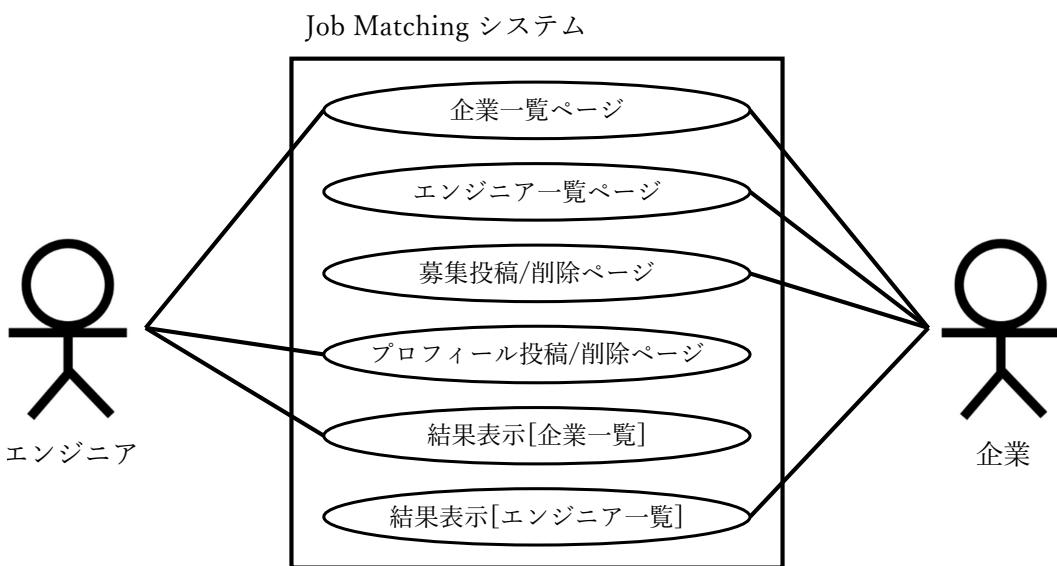


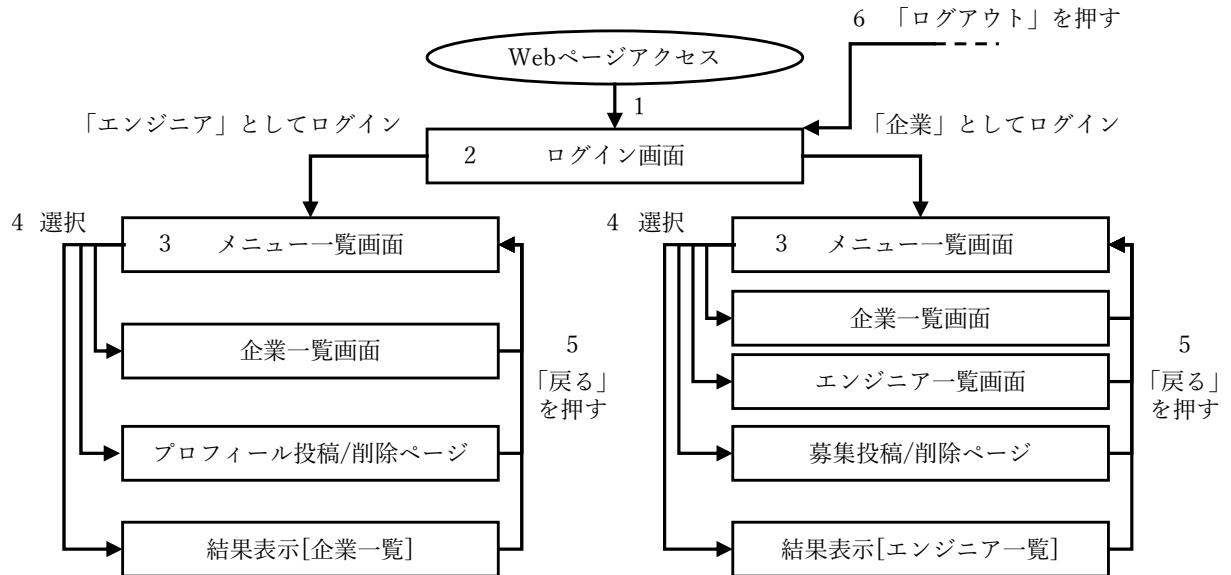
図 1: 総括ダイアグラム

総括ダイアグラムを図 1 に示す。

総括ダイアグラムは、エンジニアと企業のマッチングを行うシステムの全体像を示しており、具体的にどのページにアクセスが可能かどうかを示している。特徴として、企業側はエンジニアと企業一覧を閲覧することが可能であるが、エンジニア側は企業一覧のみ閲覧が可能であり、エンジニア一覧を閲覧することはできない。これは、一般に求人システムにおいてエンジニアは他人の情報を閲覧できないためである。

* <https://github.com/elsy0111>

3 画面遷移詳細



本システムの画面遷移図を図 2 に示す。

画面遷移図は、ユーザーがどのような操作を行った場合にどのような画面に遷移するかを示している。

以下に、具体的な遷移手順を示す。なお、手順番号は図 2 中の番号に対応している。

1. ユーザーが Web ページにアクセスする。
 - (a) 未ログインの場合、ログイン画面に遷移する。(2. へ)
 - (b) ログイン済みの場合、該当ロール^{*1*2}のメニュー一覧画面に遷移する。(3. へ)
2. ログイン画面で、Google アカウントを用いてログインする。
 - (a) 初回のログインの場合、エンジニア/企業 を選ぶことができ、いずれかのロールを選択可能。
 - (b) すでに一度 Google アカウントでログインしていた場合、以前割り当てられたロールに基づき、エンジニア/企業 いずれかのメニュー一覧画面に遷移する。
3. メニュー一覧画面が表示され、それぞれのロールに応じたメニューが表示される。
4. 各ボタンから、閲覧可能な一覧画面に遷移する。
5. 各遷移先の画面からは、「戻る」ボタンを押すことで、メニュー一覧画面に戻ることができます。
6. いずれのページにも、ログアウトボタンが設置されており、ログアウトすることができます。ログアウト後は、再度ログイン画面に遷移する。

^{*1} ここでロール (role) は、エンジニア/企業 のいずれかを指す。

^{*2} 該当ロールとは、以前ログインした際に選択したロールを指す。

4 画面レイアウト図

以下、どちらのロールからの視点かが分かるように、図のタイトルに括弧書きでロール名を記載する。記載がない場合は、共通の画面である。

4.1 ログイン

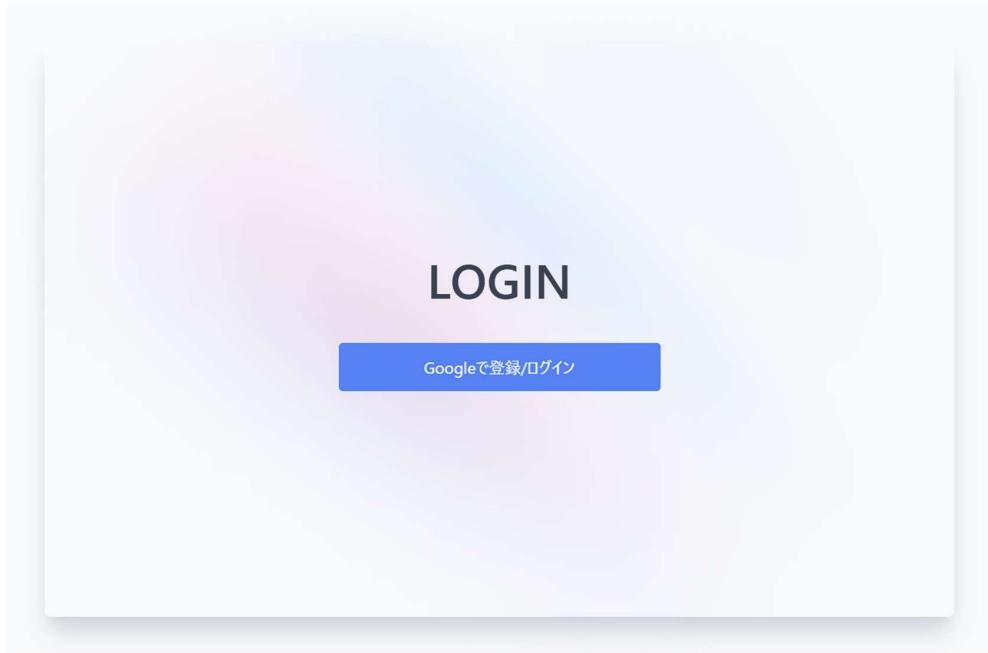


図 3: ログイン画面



(a) エンジニアとして登録

(b) 企業として登録

図 4: 初回登録画面

図 3 は、ログイン画面を示している。Google で登録/ログイン ボタンを押すと、Google アカウントでログインするフォームが表示され、ログインすることができる。

この操作の後、手順 2(a) のような初回ログインの場合、図 4(a), 図 4(b) のように、いずれかを選択するフォームが表示される。この選択により、Google アカウントに 企業/エンジニア の情報が紐づけられ、次回以降のログイン時には、選択したロールに基づいてメニュー一覧画面に遷移 (手順 2(b)) する。

4.2 メニュー一覧



図 5: メニュー一覧画面 (エンジニア)



図 6: メニュー一覧画面 (企業)

図 5 は、エンジニアとしてログインした際のメニュー一覧画面を示しており、図 6 は、企業としてログインした際のメニュー一覧画面を示している。

2. 総括ダイアグラムに示した通り、エンジニアは企業一覧画面のみ閲覧可能であり、企業はエンジニア一覧画面と企業一覧画面の両方が閲覧可能である。このボタンを押すと、図 7, 図 8 のような一覧画面に遷移する。

”募集してみる”内のボタンを押すと、図 10, 図 11 のような応募画面に遷移する。

”あなたにおすすめ？”の右側「詳しく見る»」ボタンを押すと、図 12a, 図 12b のようなマッチング画面に遷移する。

画面左下には、Google アカウントに紐づいたユーザー名・メールアドレスに加え、ロールが表示されており、現在どのアカウントでログインしているか、自身のロールが何かを一目で分かるようになっている。

また、画面右下には、ログアウトボタンが設置されており、ログアウトすることができる。このボタンは、ログインページ以外の全てのページに設置されている。

4.3 応募/求人一覧画面

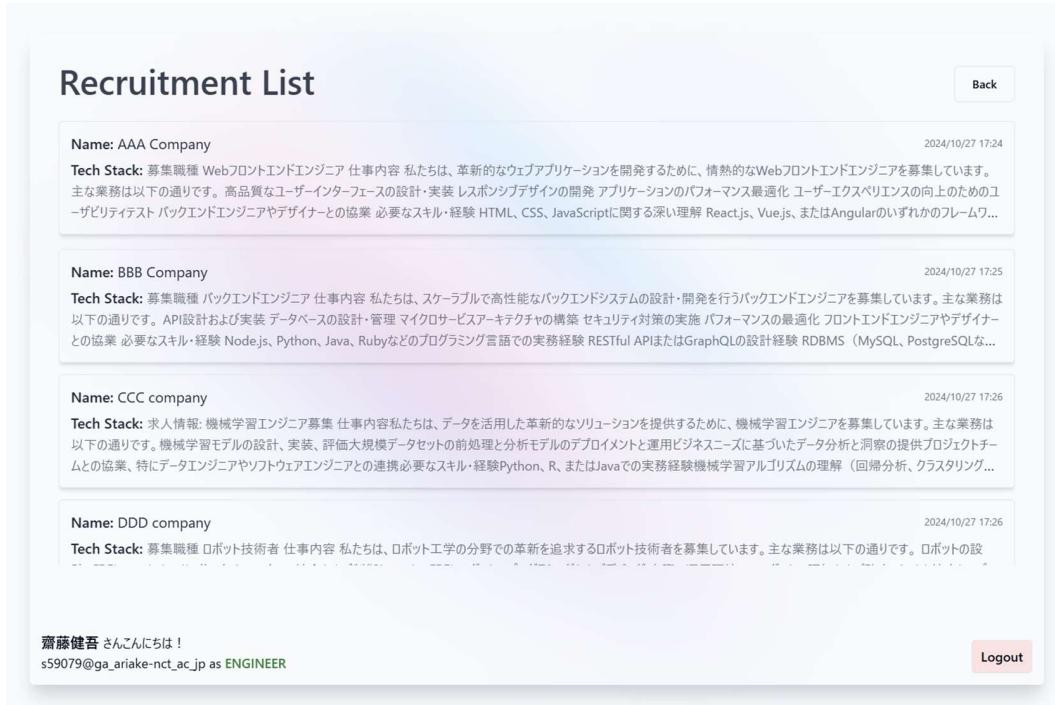


図 7: 企業一覧画面（エンジニア）

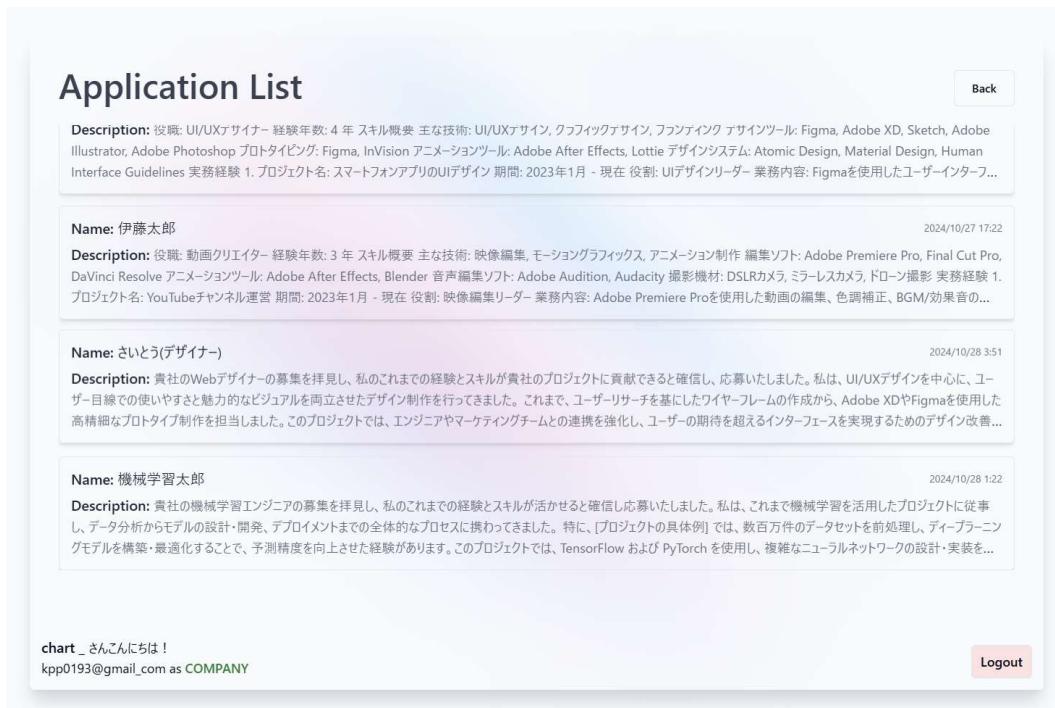


図 8: エンジニア一覧画面 (企業)

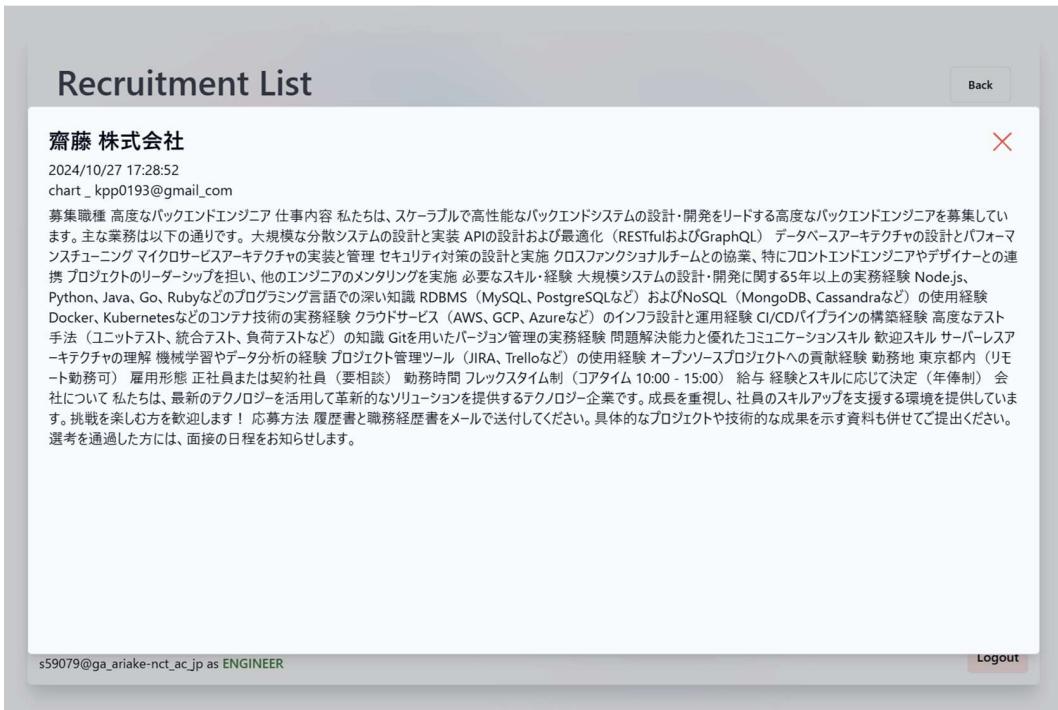


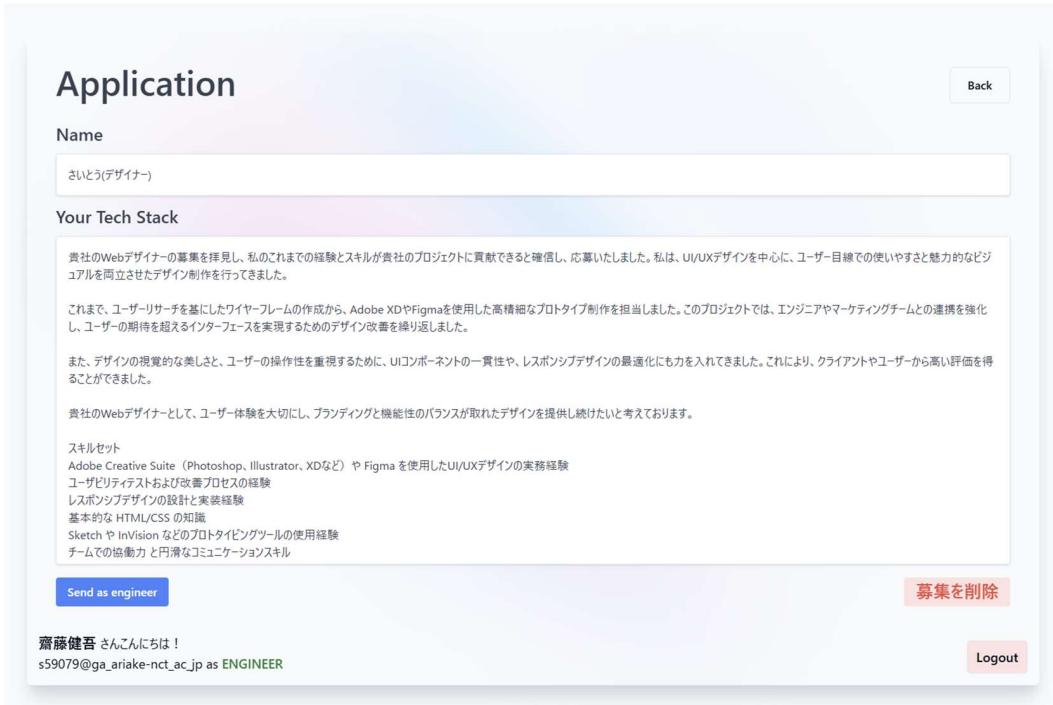
図 9: 一覧詳細 (共通)

図 7 は、エンジニアが企業一覧画面を表示した際の画面を示しており、図 8 は、企業がエンジニア一覧画面を表示した際の画面を示している。

これらのページは、後に示す応募/募集画面から投稿された情報を表示するためのページである。また、それぞれのデータには、投稿者の名前、投稿日時、本文が表示され、全て表示されるのではなく、3行で省略され、大きさが統一されている。データの詳細を見るためには、そのデータをクリックすることで、図9のような詳細画面に遷移し、先ほど表示されていた情報に加え、Gmailの名前、Gmailアドレス、全ての本文が表示される。

右上の × ボタンを押すと、このポップアップは消え、先ほどの画面に戻る。

4.4 応募/募集画面



Application

Name
さいとう(デザイナー)

Your Tech Stack

貴社のWebデザイナーの募集を拝見し、私のこれまでの経験とスキルが貴社のプロジェクトに貢献できると確信し、応募いたしました。私は、UI/UXデザインを中心に、ユーザー目線での使いやすさと魅力的なビジュアルを両立させたデザイン制作を行ってきました。

これまで、ユーザーリサーチを基にしたワイヤーフームの作成から、Adobe XDやFigmaを使用した高精細なプロトタイプ制作を担当しました。このプロジェクトでは、エンジニアやマーケティングチームとの連携を強化し、ユーザーの期待を超えるインターフェースを実現するためのデザイン改善を繰り返しました。

また、デザインの視覚的な美しさと、ユーザーの操作性を重視するために、UIコンポーネントの一貫性や、レスポンシブデザインの最適化にも力を入れてきました。これにより、クライアントやユーザーから高い評価を得ることができました。

貴社のWebデザイナーとして、ユーザー体験を大切にし、ブランディングと機能性のバランスが取れたデザインを提供し続けたいと考えております。

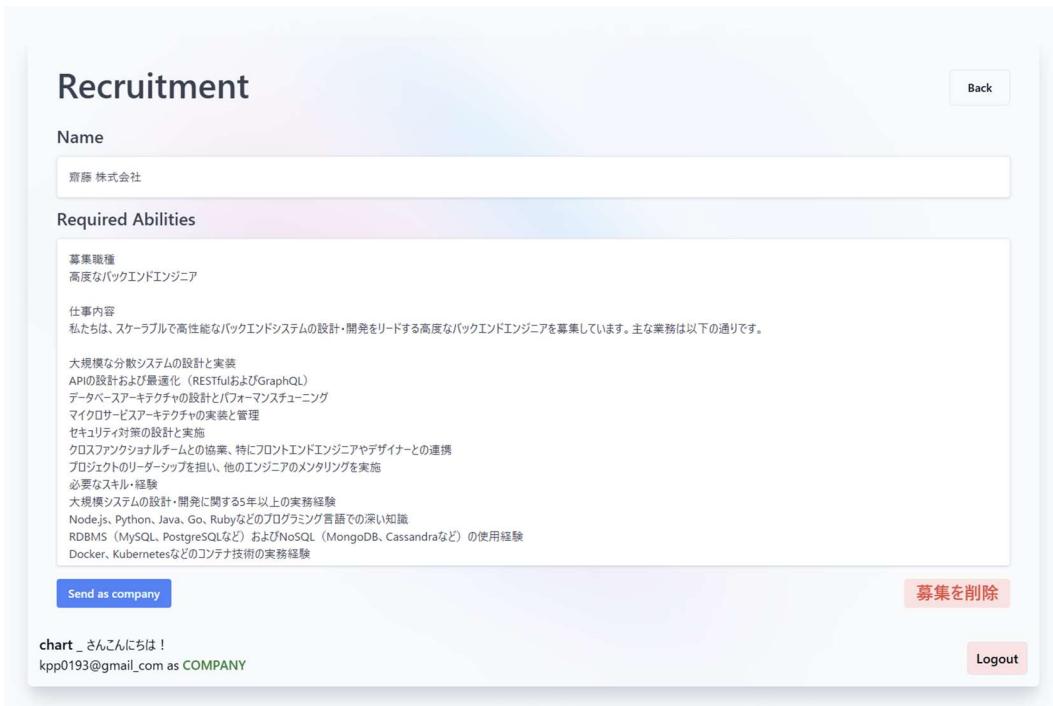
スキルセット
Adobe Creative Suite (Photoshop, Illustrator, XDなど) や Figma を使用したUI/UXデザインの実務経験
ユーザビリティテストおよび改善プロセスの経験
レスポンシブデザインの設計と実装経験
基礎的な HTML/CSS の知識
Sketch や InVision などのプロトotypingツールの使用経験
チームでの協働力と円滑なコミュニケーションスキル

Send as engineer **募集を削除**

齋藤健吾 さんこんにちは！
s59079@ga_ariake-nct_ac.jp as ENGINEER

Logout

図 10: 応募画面 (エンジニア)



Recruitment

Name
齋藤 株式会社

Required Abilities

募集職種
高度なバックエンドエンジニア

仕事内容
私たちは、スケーラブルで高性能なバックエンドシステムの設計・開発をリードする高度なバックエンドエンジニアを募集しています。主な業務は以下の通りです。

大規模な分散システムの設計と実装
APIの設計および最適化 (RESTfulおよびGraphQL)
データベーススキーマの設計とパフォーマンスチューニング
マイクロサービスアーキテクチャの実装と管理
セキュリティ対策の設計と実施
クロスフックショナルチームとの協業、特にフロントエンドエンジニアやデザイナーとの連携
プロジェクトのリーダーシップを担い、他のエンジニアのメンタリングを実施
必要なスキル・経験
大規模システムの設計・開発に関する5年以上の実務経験
Node.js, Python, Java, Go, Rubyなどのプログラミング言語での深い知識
RDBMS (MySQL, PostgreSQLなど) およびNoSQL (MongoDB, Cassandraなど) の使用経験
Docker, Kubernetesなどのコンテナ技術の実務経験

Send as company **募集を削除**

chart_ さんこんにちは！
kpp0193@gmail.com as COMPANY

Logout

図 11: 募集画面 (企業)

図 10 は、エンジニアが応募画面を表示した際の画面を示しており、図 11 は、企業が募集画面を表示した際の画面を示している。

一度投稿した情報は、企業一覧画面、エンジニア一覧画面に表示され、既に自分のアカウントから投稿さ

れていれば、フォームに一から入力することなく、編集することができる。

現在、自分のアカウントから投稿されている情報があれば、右下の「募集を削除」ボタンを押すことができ、その情報を削除することができる。

エンジニアと企業でのUIの違いは、左下の情報と表示する文字の違い(ヘッダのApplication/RecruitmentやボタンSend as _など)であり、本質的な機能に違いはない。

4.5 マッチング画面

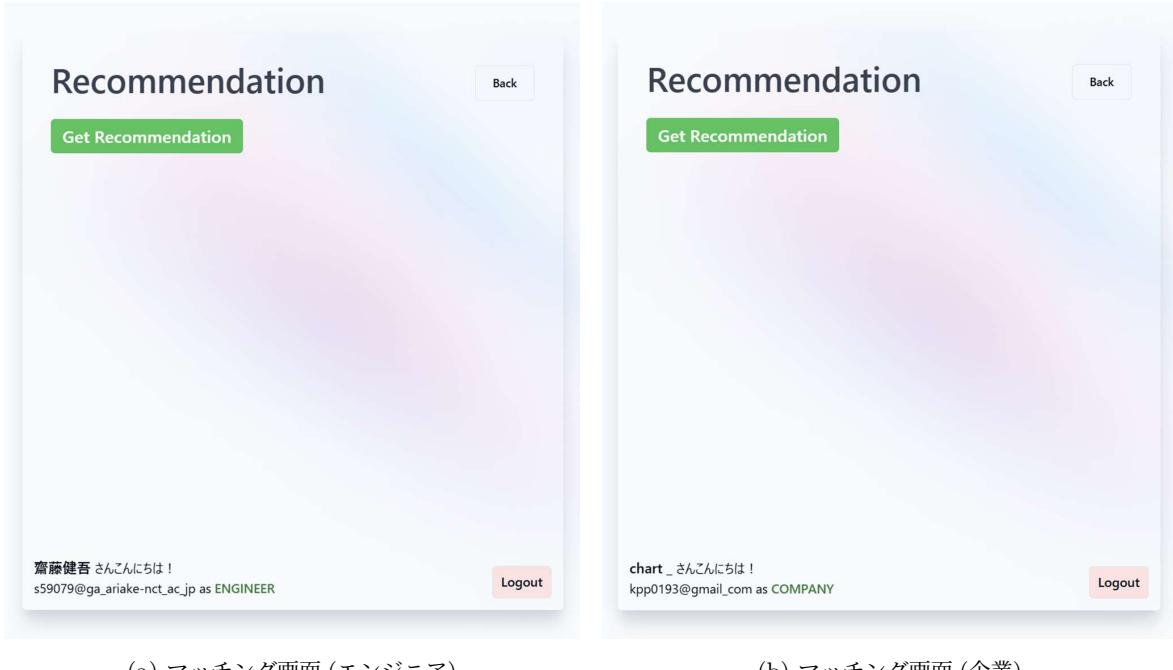


図 12: マッチング画面

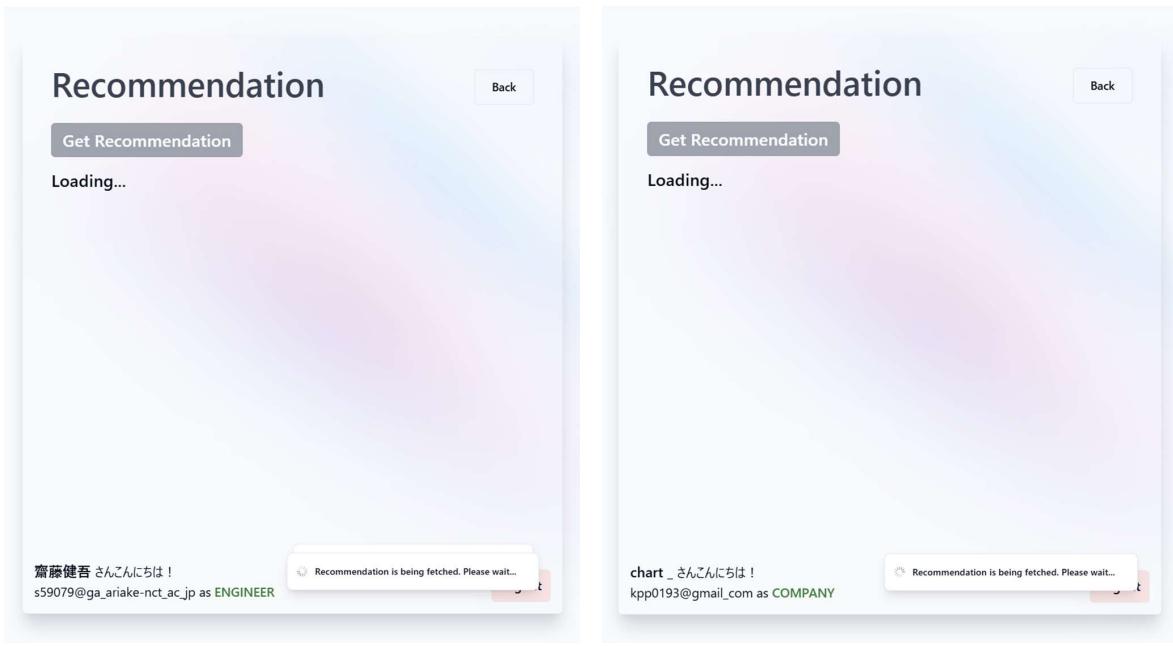


図 13: マッチング度計算中画面

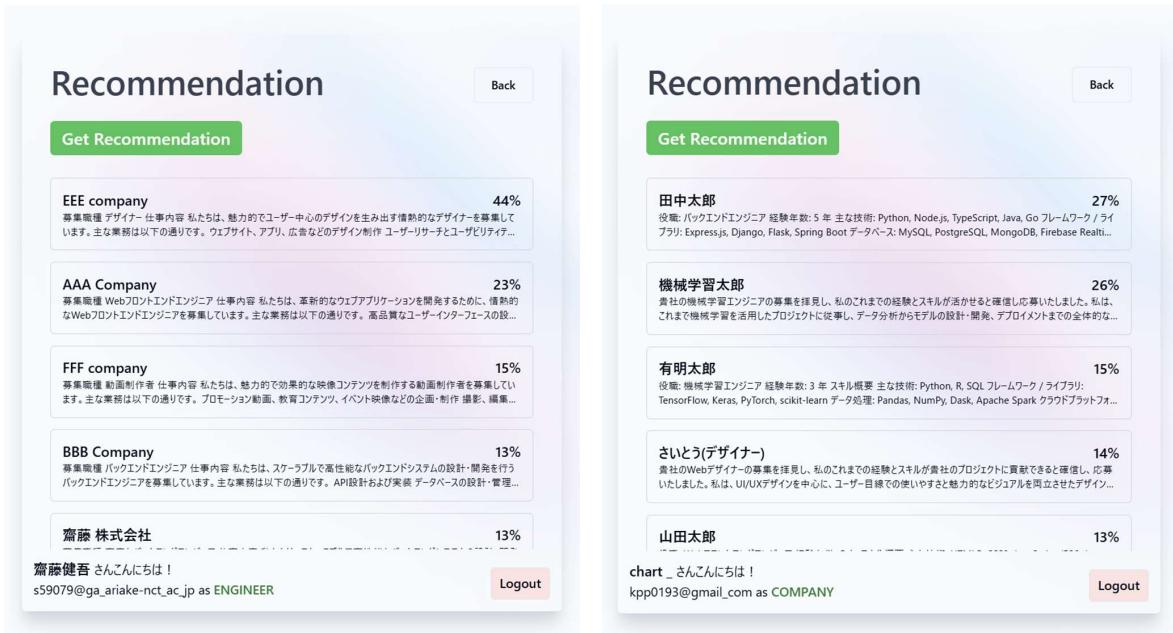


図 14: マッチング結果画面

図 12a は、エンジニアがマッチング画面を表示した際の画面を示しており、図 12b は、企業がマッチング画面を表示した際の画面を示している。

Get Recommendation ボタンを押すと、図 13a、図 13b のように、マッチング度を計算中の画面に遷移し、その間は Loading... と表示され、ボタンは押せなくなる。

図 14a、図 14b は、マッチング結果画面を示しており、エンジニアと企業の双方が、自身のアカウントに登録されている情報をもとに、マッチング度を計算し、表示する画面である。

より具体的には、自身のアカウントから投稿された 1 つの情報と、異なるロールのアカウントから投稿された多数の情報を Tf-idf によりベクトル化し、Cosine 類似度を計算、ソートすることで、マッチング度を算出する。

ゆえに、降順で表示される情報は、マッチング度が高い順になっており、各情報の右側に表示されている値(%)は、Cosine 類似度を 100 倍したものになっている。

また、図 14 は、図 10、図 11 で投稿された情報に対する結果を示しており、このような出力が期待される。

参考文献

- [1] Next.js Documentation <<https://nextjs.org/docs>>
- [2] Firebase Documentation <<https://firebase.google.com/docs?hl=ja>>
- [3] [Next.js] App Router × NextAuth.js で Google ログイン機能を実装したい <<https://qiita.com/k-sukesakuma/items/4b56b9e81c1788d38440>>
- [4] はじめに: 最初の関数の記述、テスト、デプロイ - Firebase <<https://firebase.google.com/docs/functions/get-started?hl=ja>>
- [5] Cloud Functions for Firebase を Python で使ってみた <<https://dev.classmethod.jp/articles/firebase-functions-python/>>
- [6] [python] Firebase Realtime Database のはじめ方 <<https://qiita.com/sai-san/items/24dbe74c5744033c330>>