

要件定義書

☰ タグ	設計
📅 日付	@2024年8月14日

目次

1. プロジェクト概要

1-1. 背景

1-2. システム構成

1-3. 用語定義

2. 業務要件

2-1. 業務フロー

1. 要件定義

2. 設計

3. 実装

4. テスト

5. デプロイ

2-2. 期間

3. 機能要件

3-1. 機能

3-2. 画面

共通画面

User画面

FP画面

1. プロジェクト概要

ユーザーがフィナンシャルプランナー（以下FP）への相談時間を予約するシステム。予約は1枠30分単位で行えるようにする。FP側は予約を受け付ける枠を選択できるようにする。ユーザー側はFPの空いている枠を選択できるようにする。ユーザーが予約の申請を行い、申請を行なった時点で予約完了とする。ユーザーとFPは、メールアドレスとパスワードによるユーザー登録・認証を必要とする。

平日の予約は10:00 ~ 18:00

土曜日の予約は11:00 ~ 15:00

日曜日は休業日

1-1. 背景

ユーザーがFPに相談できる新サービスを立ち上げることになった。予約を簡単にするために、ユーザーがFPとの相談を予約できるようにするwebシステムを作成する。

1-2. システム構成

- ユーザーはwebブラウザから予約可能。
- FPもwebブラウザからのみ予約可能
- ユーザー情報、FP情報はデータベースに格納
- バックエンドはrails
- フロントエンドは簡略的にhtml.erbとCSSで
- インフラはAWSで構築
- データベースはMysql(ver = 8.0)で管理
- 環境構築はDockerを使用
- コード、バージョン管理はGitを使用

1-3. 用語定義

FP：フィナンシャルプランナー

2. 業務要件

2-1. 業務フロー

以下に業務の流れを示す。また今回は個人開発のため業務フロー図は割愛する。

1. 要件定義

1-1. 要件定義書を完成させる

2. 設計

2-1. 必要なデータベースモデルを設計する

2-2. ユーザーインターフェースの大まかなレイアウトをワイヤーフレームとして作成（やらなくても可）

2-3. APIエンドポイントを設計する

2-4. インフラ設計(Renderに変更も検討)

3. 実装

3-1. Dockerを使って開発環境の構築を行う

3-2. railsを使ってバックエンドを実装

Mysqlを使ってデータベースを実装

3-3. HTML/CSSを使ってフロントエンドを実装

3-4. コードでインフラ実装するならここで実装

4. テスト

4-1. 単体テストを随時行う（実装と同時に進める）

4-2. ユーザビリティテストを行い、実際のユーザーの使用感をテストする（しなくても可）

5. デプロイ

5-1. AWSを使ってデプロイ環境を整える

5-2. アプリを実際にデプロイする

5-3. SSL証明書をLet's Encryptを使って取得し、安全な通信を担保する

2-2. 期間

業務の時間軸を以下に示す。

業務	期間
要件定義	8/14 ~ 8/16

業務	期間
設計	8/17 ~ 8/31
実装・テスト	9/1 ~ 9/30
デプロイ	10/1 ~ 10/15

3. 機能要件

3-1. 機能

▼ ユーザー管理

FP、ユーザーの登録・ログイン機能をメールアドレスとパスワードを使って、認証システムを作る。メールアドレスを使ってのメールの有効化などは今回行わない。

▼ FPのスケジュール登録機能

FPが空いている時間を指定できる機能。またユーザーもFPのスケジュールを閲覧できる。スケジュールは平日は

▼ 予約機能

FPのスケジュールに合わせてユーザーが予約画面から時間を指定して予約を行う。予約はボタン形式で行えるようにする。FPのスケジュールが空いていない時間帯は予約不可にする。

▼ 予約履歴

過去の予約と現在の予約を分けて管理できるようにする。

▼ 通知

FP、ユーザーには予約確定の通知を予約をした時点で登録しているメールアドレスに送るようにする。（マイページに通知もあり）

【利用シナリオ】

1. FPもユーザーも最初の利用でそれぞれFP・ユーザー登録を行う
2. FPはアプリにFPとしてログインし、空いている時間をアプリに登録する
3. ユーザーはアプリにユーザーとしてログインし、FPのスケジュールに合わせて予約を確定させる。

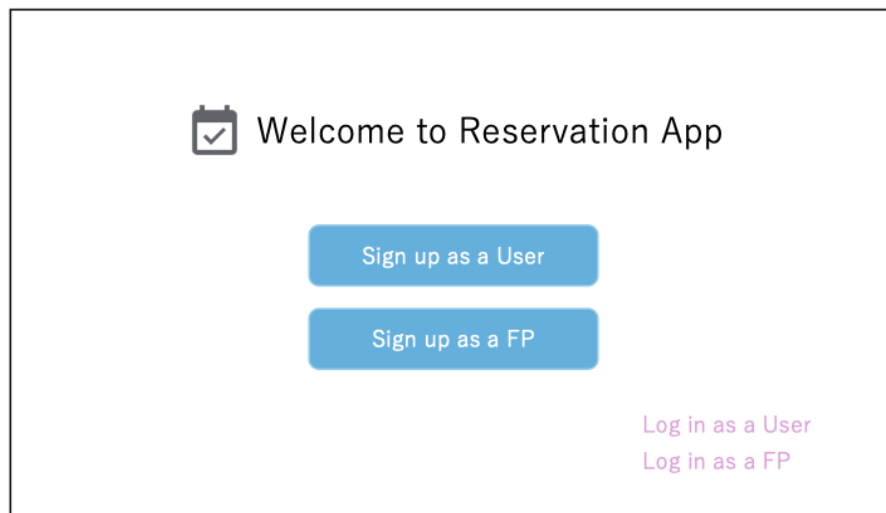
4. 予約が確定したら、FP、ユーザー共に予約完了メールを送信する。

3-2. 画面

今回のアプリで作成する画面のモックアップを以下に示す。

共通画面

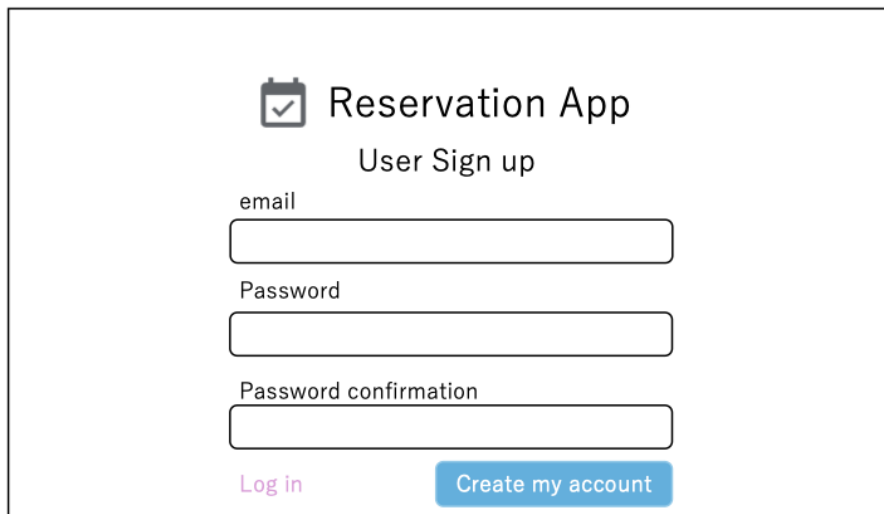
1. ホーム画面



- ホーム画面で「FP」として登録・ログインするか、「ユーザー」として登録・ログインするか選ぶ
- 選んだ後はそれぞれの登録・ログインページに遷移する。

User画面

1. 新規登録画面



The image shows a user sign-up form for a 'Reservation App'. At the top, there is a calendar icon with a checkmark, followed by the text 'Reservation App' and 'User Sign up'. Below this, there are three input fields labeled 'email', 'Password', and 'Password confirmation'. At the bottom, there are two buttons: 'Log in' (text link) and 'Create my account' (blue button).

Reservation App

User Sign up

email

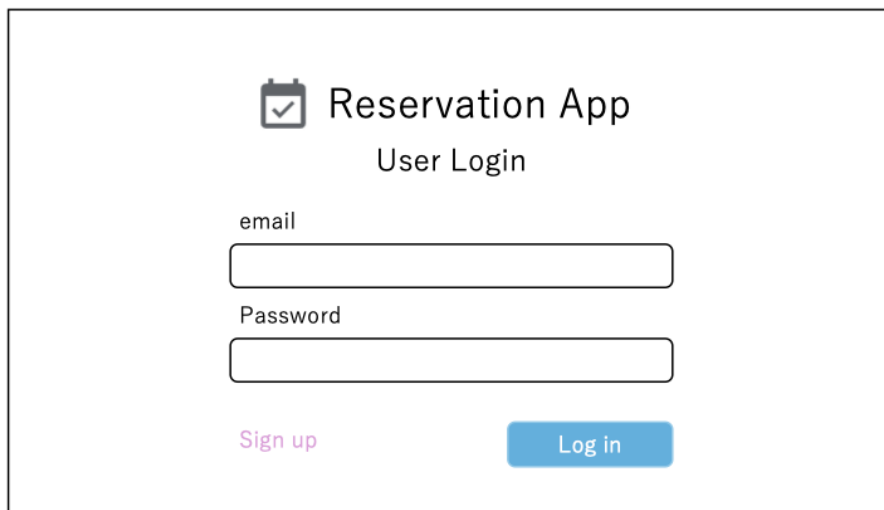
Password

Password confirmation

Log in Create my account

- ユーザーの新規登録ページ
- ホーム画面から遷移するようにする
- 「メールアドレス」と「パスワード」を要求する
- 登録が完了したら「My page」に遷移するようにする

2. ログイン画面



The image shows a user login form for a 'Reservation App'. At the top, there is a calendar icon with a checkmark, followed by the text 'Reservation App' and 'User Login'. Below this, there are two input fields labeled 'email' and 'Password'. At the bottom, there are two buttons: 'Sign up' (text link) and 'Log in' (blue button).

Reservation App

User Login

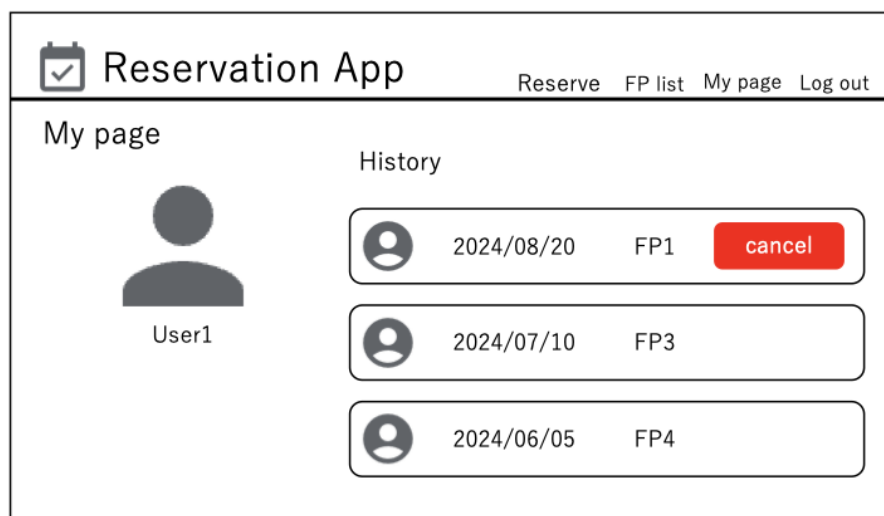
email

Password

Sign up Log in

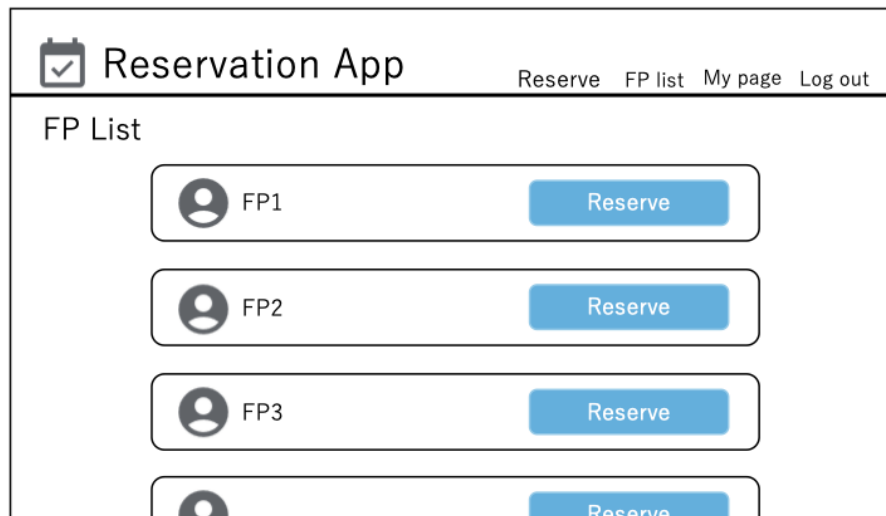
- ユーザーのログイン画面
- ホーム画面から遷移するようにする
- 「メールアドレス」と「パスワード」を要求する
- ログインが完了したら「My page」に遷移するようにする

3. My page



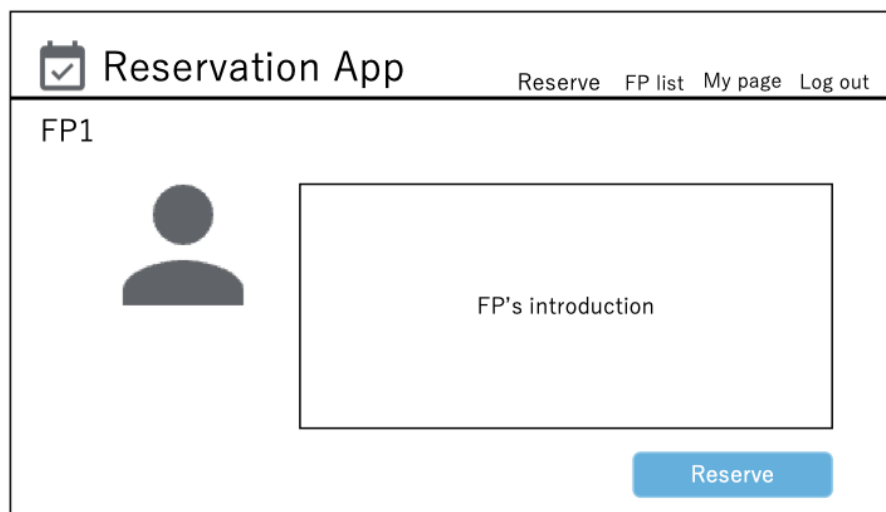
- ユーザーのマイページ画面
- 写真と名前はここから編集できるようにする
- 履歴もこのページで閲覧可能にする
- まだ行われていない予約履歴だけキャンセルできるようにする

4. FP一覧ページ



- FP一覧を表示するページ
- FPをクリックすると、そのFPの紹介ページに遷移するようにする
- このページから「Reserve」ボタンを押すと、Reserve画面に遷移し、FPを選択した状態にする

5. 特定のFPの紹介ページ



- FPの紹介ページ
- ここからも予約できるようにする。その場合、FPを選択した状態にする。

6. 予約ページ

Reservation App

[Reserve](#)
[FP list](#)
[My page](#)
[Log out](#)

Reservation

FP1

FP2

FP3

FP4

FP5

Last week	2024/08							Next week
	01(Mo)	02(Tu)	03(We)	04(Th)	05(Fr)	06(Sa)	07(Su)	
10:00	○	○	○	○	○	×	×	10:00
10:30	○	○	○	○	×	×	×	10:30
11:00	×	○	×	×	○	○	×	11:00
11:30	○	×	○	×	×	○	×	11:30
12:00	×	○	○	×	×	×	×	12:00

Reservation App

[Reserve](#)
[FP list](#)
[My page](#)
[Log out](#)

Reservation

14:00	×	○	×	×	○	○	×	14:00
14:30	○	○	○	○	○	×	×	14:30
15:00	○	○	○	○	×	×	×	15:00
15:30	×	○	×	×	○	○	×	15:30
16:00	○	×	○	×	×	○	×	16:00
16:30	×	○	○	×	×	×	×	16:30
17:00	×	×	○	○	×	○	×	17:00
17:30	○	○	○	○	×	×	×	17:30

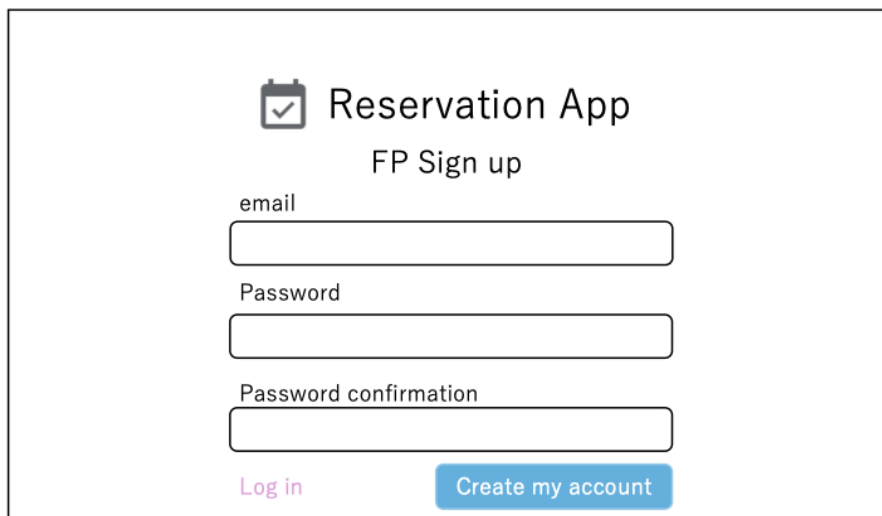
Reserve

- ユーザーの予約ページ

- 1週間ごとにスケジュール表を表示するようにする
- 「時間」と「FP」を両方選んでいる状態以外で「Reserve」ボタンを押せないようにする
- 予約が完了したらmy pageに自動的に遷移し、完了のフラッシュを出力するようにする

FP画面

1. 新規登録画面



Reservation App

FP Sign up

email

Password


Password confirmation

Log in

Create my account

- FPの新規登録ページ
- ホーム画面から遷移するようににする
- 「メールアドレス」と「パスワード」を要求する
- 登録が完了したら「My page」に遷移するようにする

2. ログイン画面

 Reservation App

FP Login

email


Password

Sign up

Log in


- FPのログインページ
- ホーム画面から遷移するようににする
- 「メールアドレス」と「パスワード」を要求する
- 登録が完了したら「My page」に遷移するようににする

3. My page

 Reservation App

ScheduleMy pageLog out


My page





FP1

Self Introduction

History


 2024/08/15User7

 2024/07/10User5

 2024/06/05User8


- FPのマイページ画面
- 写真と名前、自己紹介文はここから編集できるようにする
- 履歴もこのページで閲覧可能にする
- FPの履歴にはキャンセル機能はつけない

4. スケジュール登録画面


Reservation App
Schedule My page Log out

Schedule

Last week	2024/08							Next week
	01(Mo)	02(Tu)	03(We)	04(Th)	05(Fr)	06(Sa)	07(Su)	
10:00	○	○	○	○	○	×	×	10:00
10:30	○	○	○	○	×	×	×	10:30
11:00	×	○	×	×	○	○	×	11:00
11:30	○	×	○	×	×	○	×	11:30
12:00	×	○	○	×	×	×	×	12:00
12:30	×	×	○	○	×	○	×	12:30
13:00	○	×	○	×	○	×	×	13:00


Reservation App
Schedule My page Log out

Reservation

14:00	×	○	×	×	○	○	×	14:00
14:30	○	○	○	○	○	×	×	14:30
15:00	○	○	○	○	×	×	×	15:00
15:30	×	○	×	×	○	○	×	15:30
16:00	○	×	○	×	×	○	×	16:00
16:30	×	○	○	×	×	×	×	16:30
17:00	×	×	○	○	×	○	×	17:00
17:30	○	○	○	○	×	×	×	17:30

Save

- FPのスケジュール登録ページ
- 最初は全てのスケジュール（土曜日の特定時間・日曜日以外）を○にしておく
- クリックだけで「○」、「✕」を入れ替え夏の運みする
- 「Save」ボタンを押すとMy pageに自動的に遷移し、「スケジュール登録完了」のフラッシュを出力する