

知能情報基礎演習 II  
要件定義と基本設計家書の作成  
グループ番号 7    Develoop

245429H    末吉 良多  
245745J    知念 拓弥  
245704B    武嶋 優海

2025 年 7 月 17 日

# 1 開発するアプリ名、内容、開発の背景等

## 1.1 開発するアプリ名

Shelfie

## 1.2 内容

自分が読んだ本、買った本の写真を撮り、デジタル本棚を作る

## 1.3 開発背景

多くの人が、読書を通じて得た感動や知識の証として、読んだ本をコレクションしたいという思いを持っています。

しかし、特に一人暮らしの学生など、限られたスペースや予算で生活する人々にとって、物理的な本棚を充実させていくことは簡単なことではありません。

「本は好きだけれど、置き場所がないから買えない」

「思い出深い本でも、泣く泣く手放さなければならない」

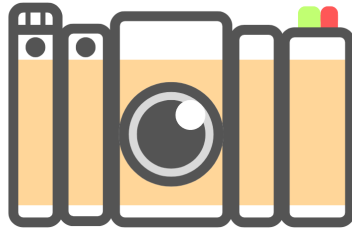
こうした悩みを解決し、誰もが気軽に自分だけの本棚を持てるようにしたいと思い開発を始めました。

## 1.4 利用者のペルソナ

田中美咲

- **基本情報：** 20 歳、大学 3 年生（文学部）。東京都内のワンルームで一人暮らし。
- **性格：** 感受性が豊かで、一人の時間を大切にする。自分の好きな作家や世界観には強いこだわりを持つ。
- **ライフスタイル：** 古本屋巡りが趣味。情報収集は主に SNS の読書アカウントから。
- **読書スタイル：** 装丁の美しい本を「モノ」として所有したい「紙の本派」。純文学や海外小説を好む。
- **IT リテラシー：** アプリの利用に抵抗はなく、直感的でおしゃれなデザインを好む。
- **課題：** 部屋が狭く本の置き場所に困っており、愛着のある本も手放さざるを得ない。
- **インサイト：** 手放した本も含め、読んだ証を「自分だけの本棚」として視覚的にコレクションしたい。

## 1.5 ロゴ



## 2 業務要件

今回の授業で作成するアプリの要件定義表を以下に示します。本棚として使用するために最低限必要な、書籍情報の登録、検索、編集、削除、一覧表示の業務を実装します。

また、今回は開発期間が短い都合上、ユーザー機能を実装しないこととしました。本棚の情報はブラウザのキャッシュに保存することとします。

識別子	業務名	概要	備考
BM01	書籍登録	書籍の情報を登録する	表紙画像、ISBN、JAN コード、 タイトル、著者名、追加日
BM02	書籍検索	書籍の情報を基に検索する	ISBN、JAN コード、タイトル、 著者名、タグ
BM03	書籍編集	感想・メモの登録タグ付け 機能	タグ作成
BM04	書籍削除	書籍の情報を削除する	一括削除
BM05	書籍一覧表示	登録書籍を一覧表示する	本棚表示

表 1 要件定義表

## 3 機能要件

基本設計では、先程の要件定義表をもとに、各機能の詳細を設計しました。優先度が高い機能から順に実装していきます。次ページに、基本設計表を示します。

識別子	機能ID	機能名	機能要件	入力	出力	前提条件	事後条件	優先度
BM01	FN01	書籍登録	書籍情報登録、識別ID割り振り	表紙画像、ISBN、JANコード、タイトル、著者名、追加日	表紙画像、完了メッセージ		表紙画像が参照できる	高
BM02	FN02	書籍検索	書籍の情報を基に検索する	キーワード、著者名、タイトル、タグ、識別ID	検索結果	書籍情報が登録されている	検索結果が表示される	中
BM03	FN03	書籍編集	感想・メモの登録、タグ付け機能	識別ID、自由記述、タグ	更新完了メッセージ	識別IDが存在	変更内容が反映される	高
BM04	FN04	書籍削除	書籍情報を削除する	識別ID	削除完了メッセージ	識別IDが存在	本棚から本の削除、登録情報が削除されている	高
BM05	FN05	書籍一覧表示	登録書籍を一覧表示する		書籍一覧	書籍情報が登録されている	本棚に登録されている書籍が一覧表示される	高
BM06	FN06	お気に入りページ表示	感想文の横でお気に入りページを表示	ページ画像	ページ画像	識別IDが存在	感想欄にページ画像が表示される	中

表2 要件定義表

### 3.1 ワイヤフレーム

Figma で作成。そのスクリーンショットを添付する。



図1 トップページ



Shelfie  
~デジタル本棚~



検索



登録



感想



ホーム

## キングダム75巻



作者名 : ~~ ~~

出版社名 : ~~~~~~

登録日 : ~~~~~

タグ :

歴史

マンガ

ヤング  
ジャンプ

## 感想

~~~~~

~~~~~

~~~~~

~~~~~

~~~~~

~~~~~

~~~~~

~~~~~

~~~~~

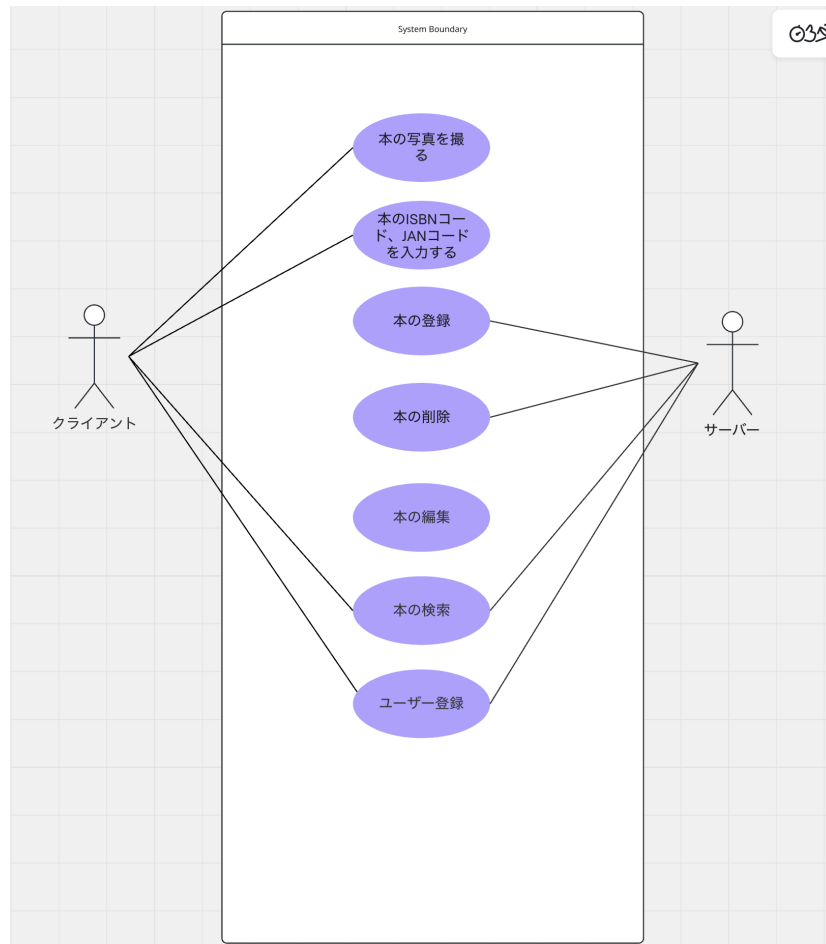
~~~



図2 本の詳細ページ

## 3.2 ユースケース

MIRO で作成。そのスクリーンショットを添付する



## 4 詳細設計書

### 4.1 開発環境・ツール

- 使用する言語
  - ー フロントエンド  
HTML(Jinja2), CSS
  - ー バックエンド  
Python3
- フレームワーク  
Flask



- 開発ツール  
VSC
- バージョン管理システム  
Git, GitHub

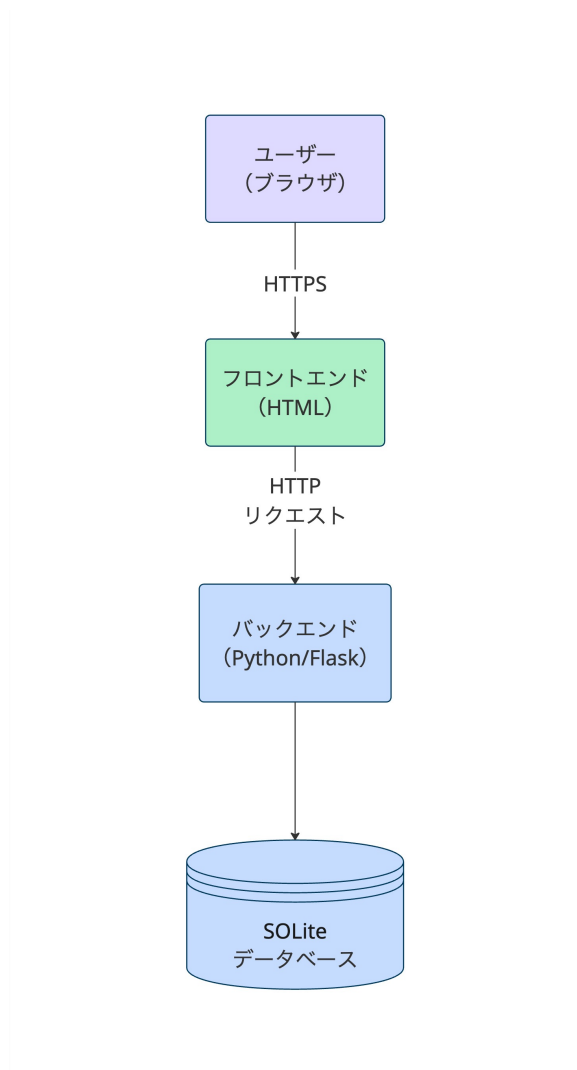
## 4.2 システム構成

ハードウェア構成は、ユーザー PC とサーバー PC の 2 つで構成されます。

ソフトウェア構成は、ユーザー PC の Web ブラウザ、サーバー PC の Web サーバーで構成されます。データベースは SQLite を使用するため、データベースサーバーは用意しません。

ネットワーク構成は、ユーザー PC がネットワークを介してサーバー PC にアクセスする形になります。

以下は、フロントエンドとバックエンドの構成図です。



## 4.3 データベース設計

### 4.3.1 ブックテーブル (book)

| カラム名     | データ型         | 制約                                  | 説明             |
|----------|--------------|-------------------------------------|----------------|
| book_id  | BIGINT       | PRIMARY KEY, AUTO_INCREMENT         | 本の識別子          |
| img      | BLOB         | NOT NULL                            | 本の表紙、ページ       |
| name     | VARCHAR(255) | NOT NULL                            | 本のタイトル         |
| author   | VARCHAR(50)  | NOT NULL                            | 本の著者           |
| add_date | TIMESTAMP    | NOT NULL, DEFAULT CURRENT_TIMESTAMP | 本の追加日          |
| code     | VARCHAR(14)  | NOT NULL                            | ISBNコード        |
| memo     | VARCHAR(511) | NOT NULL                            | 本の感想等自由記述 (任意) |
| tag      | VARCHAR(10)  | NOT NULL                            | 本のタグ (任意)      |
| message  | VARCHAR(255) | NOT NULL                            | 完了メッセージ等の管理    |

ブックテーブルの主キーは'book\_id' で、本の一意な識別子です。ID の被りを防ぐため、AUTO\_INCREMENT も制約として設定しています。画像は'img' カラムに保存され、データ型は BLOB を使用し、バイナリデータで画像を保存します。'add\_date' は本の追加日時を記録するための DATETIME 型で、デフォルト値は現在日時に設定しています。その他のカラムは、書籍のタイトル、著者、出版社、ISBN、ジャンル、評価、感想を保存するための VARCHAR 型で、それぞれ適切な長さを設定し、NOT NULL 制約を付けています。

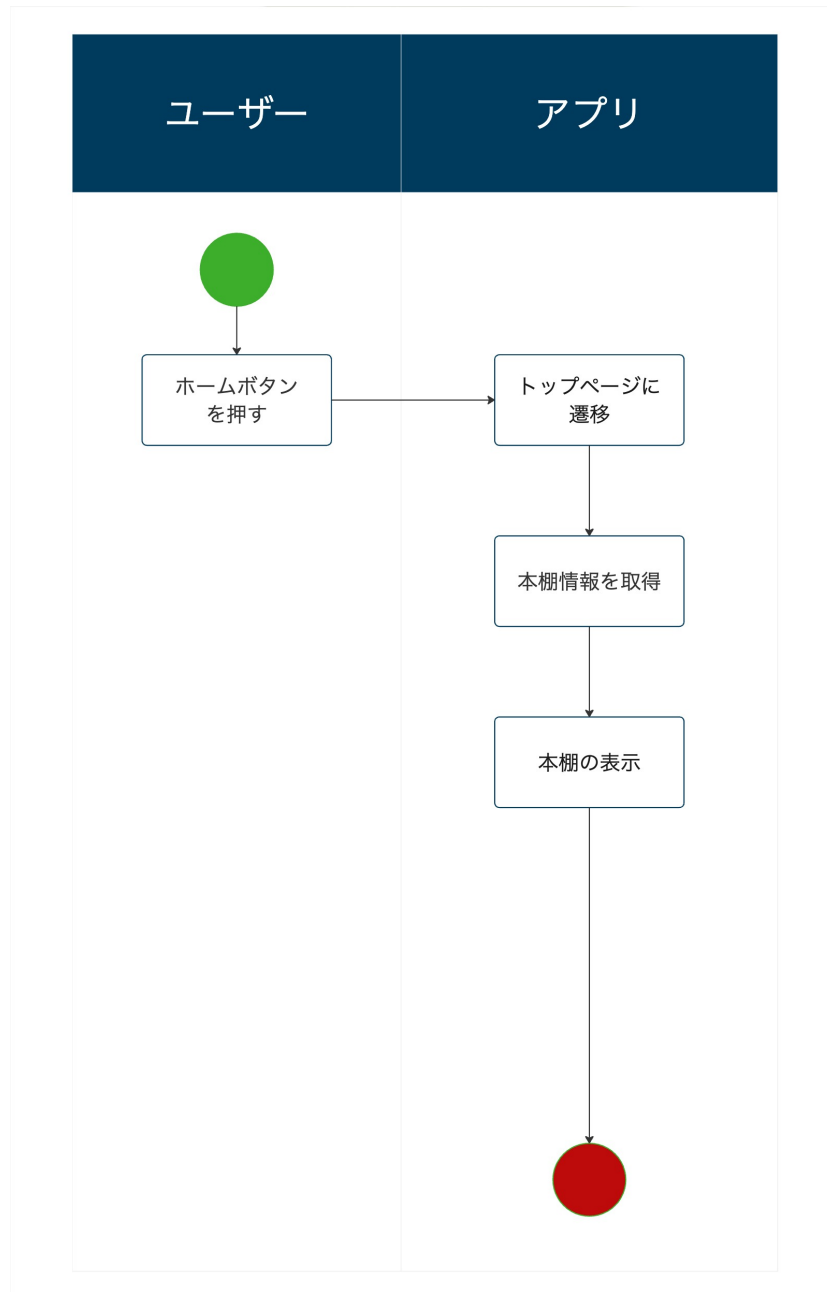
### 4.3.2 本棚テーブル (shelf)

| カラム名     | データ型   | 制約       | 説明     |
|----------|--------|----------|--------|
| shelf_id | BIGINT | NOT NULL | 本棚の識別子 |
| book_id  | BIGINT | NOT NULL | 本の識別子  |

本棚テーブルの主キーは'shelf\_id' で、本棚の一意な識別子です。'shelf\_name' は本棚の名前を保存するための VARCHAR 型で、NOT NULL 制約を付けています。外部キーとして book\_id を設定し、ブックテーブルの book\_id と関連付けています。これにより、本棚に登録されている本の情報を参照できます。

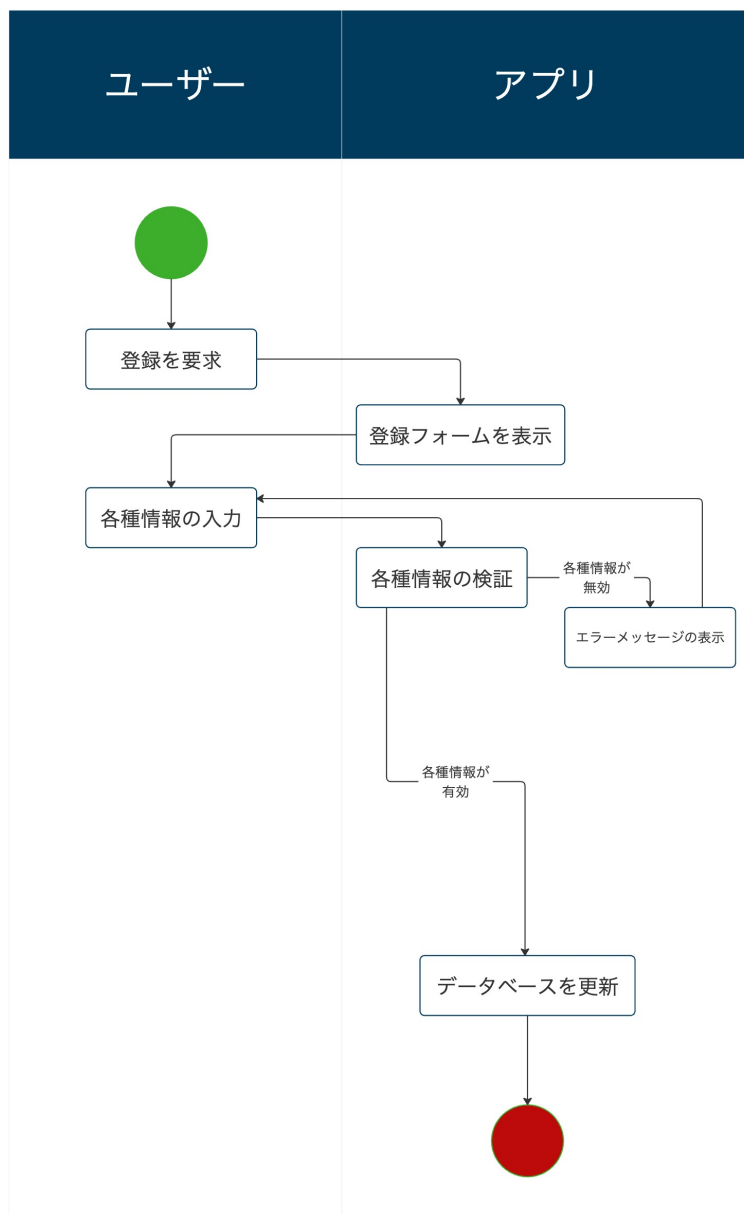
## 4.4 各機能の処理フロー

### 4.4.1 書籍一覧表示



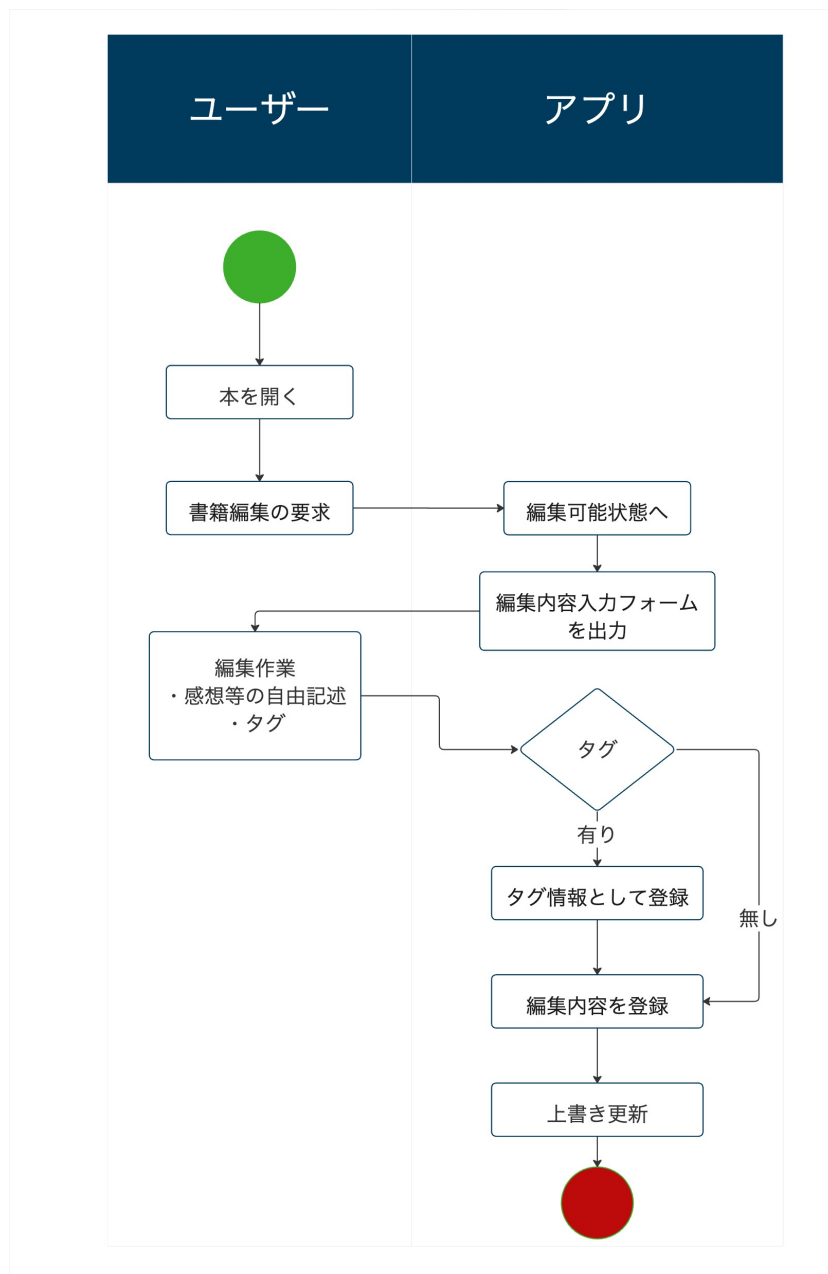
ユーザーがはじめに行う操作は、ホームボタンを押すことです。それにより、アプリ側はトップページに遷移し、本棚情報をデータベースから取得します。取得した本棚情報から、本棚の画像や名前をレンダリングします。

#### 4.4.2 書籍登録



ユーザーが書籍の登録を要求する、すなわち登録ボタンを押すと、アプリ側で登録フォームを表示し、ユーザーに各種情報の入力を求めます。ユーザーが必要な情報を入力し、登録ボタンを押すと、アプリ側はその情報がフォームの入力規則に従っているかを検証します。検証が成功すると、アプリ側はデータベースに書籍情報を登録し、検証が失敗すると、エラーメッセージを表示します。登録が成功すると、ユーザーは登録完了のメッセージを受け取り、トップページにリダイレクトされます。

#### 4.4.3 書籍編集



ユーザーが本のページを開き、書籍編集を要求、すなわち編集ボタンを押すと、アプリは本の詳細を編集可能な状態にし、各種内容を編集するフォームを出力します。ユーザーは感想等の自由記述、タグの設定が可能で、アプリ側でタグがすでに存在するなら、書籍情報がタグに紐づけられます。タグが存在しない場合は、編集内容を登録し、アプリ側は新しいタグを作成します。ユーザーが編集内容を保存すると、アプリ側はデータベースに変更を上書き更新し、更新完了のメッセージを表示します。これにより、ユーザーは本の詳細情報を自由に編集できるようになります。

## 5 グループでの分担内容と貢献度、MVP

### 5.1 詳細設計書におけるグループでの分担内容と貢献度

- 245429H 末吉 良多 (30%) 開発環境・ツールと書籍一覧表示フローを作成。
- 245745J 知念 拓弥 (40%) システム構成図と書籍登録フローを作成。とりまとめ。
- 245704B 武嶋 優海 (30%) データベース設計と書籍編集フローを作成。

MVP : 245745J 知念拓弥