

Mission_6-1 企画&要件定義書 （9月4日時点）

1 企画名

へなちょこちゃっとぷらすあるいあ

2 問題意識

掲示板をつくる。

「ぷらすあるいあ」の要素として、

■予定表

■アニメスタッフデータベースのレプリカサイトも考えたい。

2.1 うんちく

手段が豊富であるということは、より多様な角度から世界に出会うことを意味しており、問題発見の機会に恵まれることを示唆する。（手段が増えたところで、内実が伴っていなければ元も子もないというもっともな指摘をしてはならない。）

自分が何かを匿名で発信する。誰かがイイネをするかもしれない、リツイートをするかもしれない。ディスって来るかもしれない。世界が豊かになるかもしれない。ならないかもしれない。誰にもわからない。

■ネーミングセンスは亀井さんや逸見さん(xyzzy 界限)をリスペクト

■アニメスタッフデータベースのレプリカサイトを作るのは、演出オタクなる人が少ないため。（作画オタク界限は結構いる印象）。にちゃんねるの演出板は老人会的な雰囲気が入りにくい。そこで、新気鋭としての当サイトです。おはぎさん（<http://nextsociety.blog102.fc2.com/blog-entry-1451.html>）などのように知名度のある人であれば、こちらをメインに、ブログの方を置き場所にしてもらえると面白いのかもしれない。（絵師が Twitter メインで活躍して、pixiv で定期的に過去ログを保持するような感じ）

2.2 対象：不特定多数。（ニックネームをつけるが、おそらく匿名に等しい）

2.3 いつ：任意の時間。

2.4 誰が：ログインユーザと一般ユーザで、書き込みができるのはログインユーザのみ。

2.5 何のために：何かを書いたり、写真を貼れたりできるようにする。アニメの演出を語りたい。

2.6 想定するユーザ数：20 人を上限で試運用。（データをどのくらい食うかを見計らって上限を決定。）

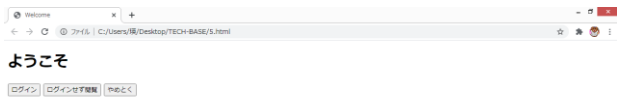
3 用意する画面

3.0 ホーム画面（入口）(構成要素 ボタン 3 個)（画面 1）

[1] 「ログイン」を押すと、画面 2 に遷移。

[2] 「ログインせずに閲覧」を押すと、画面 5 に遷移する。

[3] 「やめとく」を押すと、人格を否定するような言葉をアラートで発する。



3.1 ログイン画面（画面 2）（構成要素 入力欄 4 個 ボタン 2 個）

必要とするもの

①ユーザ id , ②ユーザパスワード, ③ニックネーム, ④email アドレス。

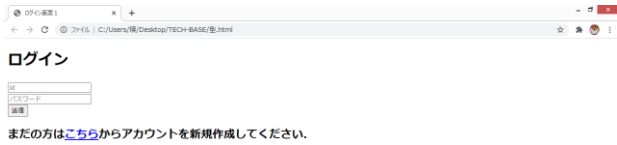
考え方

✓ ①～④はいずれも重複を許さない。

✓ ②は 8 文字以上。平文で保持せず毎回ハッシュにする。

✓ ④のメールアドレスはユーザ認証時に使う。

✓ ログインごとに、メールでお知らせを送ることも可能にする。（ユーザページで設定できるようにする）



3.2 新規登録画面

3.2.0 新規登録までのながれ

イ) 情報入力（画面 3）

ロ) 入力情報の確認（画面 4）

ハ) パスコード認証（画面 5）

ニ) 認証登録完了（画面 6）

3.2.1 画面 3（構成要素 入力欄 4 個+ボタン 2 個）

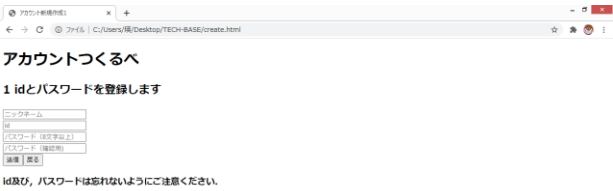
[1] ①~④を任意に登録してもらう。ここで、②は 2 つ欄を設けて 2 回入力させる。

[2] 「送信」（下図参照）が押され、②の 2 個の入力が一致し、かつ上記の考え方を満たすとき、新規登録画面 2 に遷移。

ダメな時は、何がいけないか可能な限りアラートを出す。

[3] 「戻る」が押された時、画面 2 に遷移。

[4] 「ホームへ」が押された時、画面 1 に遷移。



3.2.2 画面 4（構成要素 テキスト＋ボタン 2 個）

- [1] 新規登録画面 1 で入力された値を表示する。
- [2] 「ok」を押すと新規登録画面 3 に遷移。
- [3] 「訂正する」を押すと、新規登録画面 1 に遷移。
- [4] 「ホームへ」を押すとホーム画面に移動。



3.2.3 画面 5（構成要素 入力欄 1 個＋ボタン 2 個）

- [1] 新規登録画面 1 で登録したメールアドレスに宛てて、パスコード（任意の 5 桁の数字）を送る。
- [2] 入力欄（下図参照）に正しい値が入力されて、「送信」が押された時、画面 6 に遷移。
- [3] 連続で 4 回間違えたときは、強制的に画面 1 に飛ばす。（画面 3 で入力された値は破棄）
- [4] 「ホームへ」ボタンを押すと入力画面に戻る。（画面 3 で入力された値は破棄）



3.2.4 画面 6（構成要素ボタン 3 個）

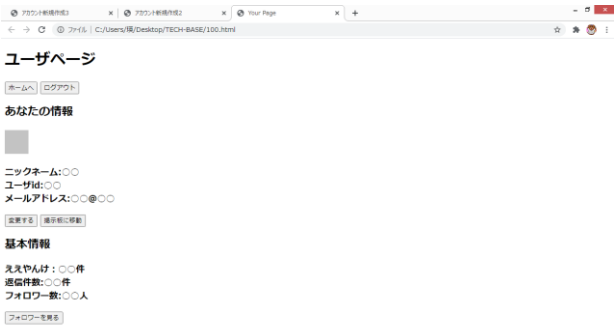
- [1] 「写真を選択」を押すと、ウィンドウが開き、写真を登録できるようになる。
- [2] 「写真を登録」を押すと、[1]で登録した写真が保存され、画面 7 に遷移。
- [3] 「しない」を押すと、画面 7 に遷移。



3.3 ユーザページ

3.3.1 画面 7（構成要素 ボタン 5 個）

- [1] 「ホームへ」を押すと、画面 1 に遷移。（ログイン情報は保持）
- [2] 「ログアウト」を押すと、ログアウトし画面 10 に遷移。
- [3] 「変更する」を押すと、画面 8 に遷移。
- [4] 「掲示板に移動する」を押すと画面 12 に遷移。
- [5] 「フォロワーを見る」を押すと、画面 1 に遷移。



3.3.2 画面 8（構成要素 入力欄 3 個＋ボタン 2 個）

- [1] 入力欄 3 つに正しく入力され、「送信」が押された時、画面 9 に遷移。このとき、画面 9 では、画面 8 で入力された値を保持している。
- [2] 「戻る」を押した時、画面 7 に遷移。このとき、画面 8 の入力欄の値は全部破棄されている。



更新します

あなたのid
あなたのパスワード
メールアドレス
<input type="button" value="送信"/> <input type="button" value="戻る"/>

3.3.3 画面 9（構成要素 入力欄 1 個＋ボタン 1 個）

[1] パスコードが正しく入力され、「送信」が押されるとき、画面 7（変更済み）に遷移。

[2] 「ユーザページへ」が押されると、画面 7 に遷移。このとき、画面 8,9 で入力した値は破棄される。



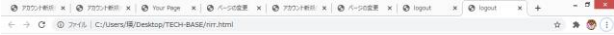
3.3.4 画面 10（構成要素 ボタン 1 個）

[1] 「ホームへ」が押されると、画面 1 に遷移。（ログイン情報の一切を保持していない状態）



3.3.5 画面 11（構成要素 ボタン 1 個）

[1] 「戻る」を押すと、画面 7 に遷移。



3.4 掲示板

3.4.1 画面 12

[0] 掲示板一覧が並んでいる。

[1] 「ログアウト」を押すと、画面 11 に遷移。

[2] いずれかの掲示板のリンクを踏むと、そこに遷移。

[3] 新規作成を押すと、スレッドのタイトルが聞かれるので、入力（図を参照）

管理者以外は掲示板を一度作ると消すことはできない。掲示板の作者は匿名。

[4] 「検索」のボックスに値を入力して「検索」を押すと、その掲示板の中の書き込みを検索して、該当するものを黄色で示す。存在しないときは、アラートで知らせる。



3.4.2 画面 13

[1] 「コメント」「削除」「編集」のいずれかに入力されているとき、送信を押すと書き込まれる。

[2] それぞれの投稿に、「いいね」や返信ができる。（実行時のイメージは図を参考）

[3] 「削除」「編集」は自分のアカウントで行った投稿のみ。但し、管理者はすべて操作可能。

[4] 「掲示板一覧へ」を押すと画面 11 に遷移。

[5] 「検索」のボックスに値を入力して「検索」を押すと、その掲示板の中の書き込みを検索して、該当するものを黄色で示す。存在しないときは、アラートで知らせる。

雑談 1

3.4.3 画面 14

[0] 画面 12 の新規作成が存在しないもの.

3.4.4 画面 15

[0] 画面 13 の 3 つの入力欄が存在しないもの.

3.4.5 画面 16

[0] 予定表（無駄機能）

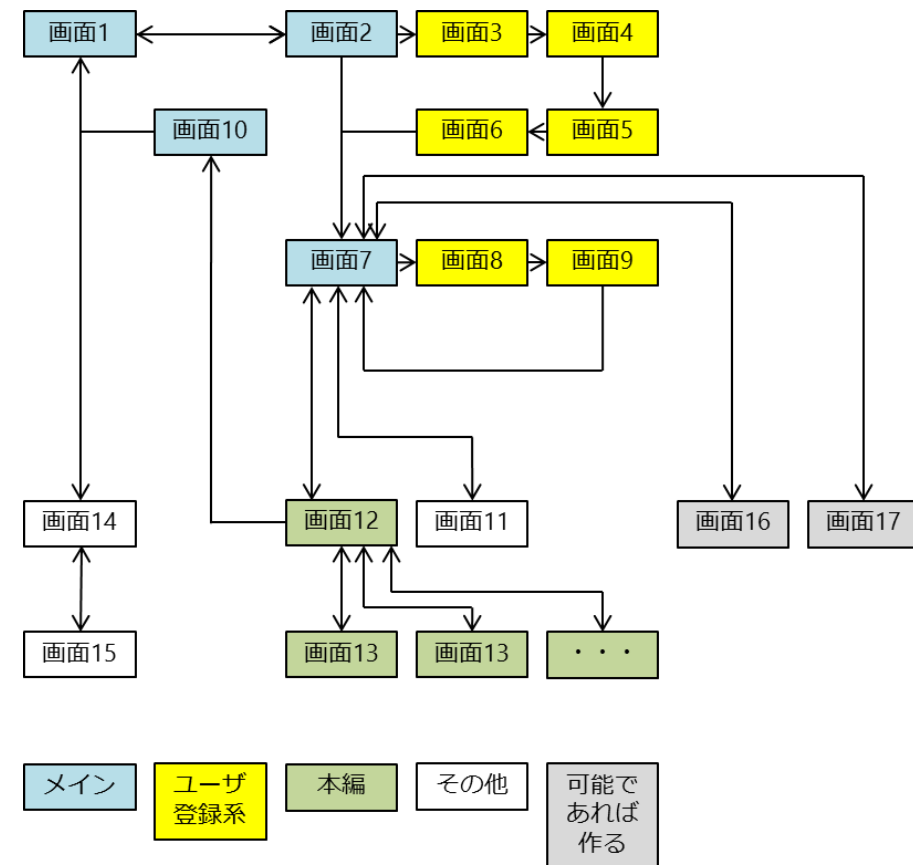
アイデア

- ✓ 日にち選択をセレクトタブで行う。月ごと表示編集。（おそらく入力フォームに書いて、送信するとページに反映させることはできるはず）
- ✓ 入力したものを、ボタンを押すとテキストファイルまたは PDF で保存する。

3.4.5 画面 17

アニメスタッフデータベース(https://seesaawiki.jp/w/radioi_34/)のレプリカ。何らかの形で、チャットと連関させることを想定しているが、明確になっていない。

3.5 関係性



必ず画面 1 に帰ることができる.

4 データベース設計の目安

4.1 想定するデータベース

個人情報保持(db1)

フィールド名	型	意味
id	INT	会員番号(auto increment)
name	TEXT	ニックネーム
userid	TEXT	ユーザ任意の id
email	TEXT	メールアドレス
password	TEXT	パスワード
picture	TEXT	プロフィール写真のパス
created	TEXT	作成日
modified	TEXT	変更日

チャット一覧 (db2)

フィールド名	型	意味
id	INT	掲示板番号(auto increment)
chatname	TEXT	チャット名
chat_author	INT	作者(db1 の id で紐づけ)
created	TEXT	作成日
modified	TEXT	変更日

チャットの中身 (db3×n) (db2 と連関)

フィールド名	型	意味
id	INT	メッセージ id(auto increment)
message	TEXT	メッセージ
memeberid	INT	投稿者の id
replyid	INT	返信先のメッセージ id
goodness	INT	いいねの個数
created	TEXT	作成日
modified	TEXT	変更日

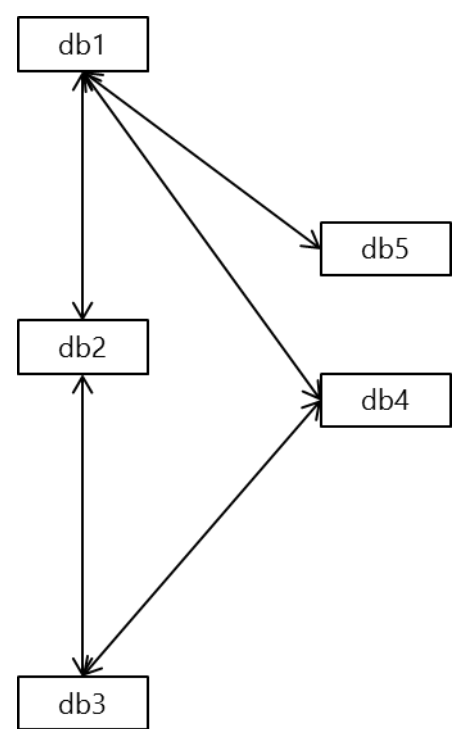
フォロワーおよびいいね(db4)(db3 および, db1 と連関)

フィールド名	型	意味
id	INT	会員番号
userid	INT	ユーザ任意 id
followers	INT	フォロワーid
goods	INT	いいね数の集計
created	TEXT	作成日

modified	TEXT	変更日
----------	------	-----

スケジュール(db5)(db1 と連関)

フィールド名	型	意味
id	INT	id
username	INT	ユーザ名
date	TEXT	日にち
schedule	TEXT	スケジュール
created	TEXT	作成日
modified	TEXT	変更日



この図は目安であり，集計用にデータベースを作るかもしれない。

5 作成手順

- [1] データベース 1~5 まで作成. (一瞬で終わる)
- [2] 画面を順番に作成.
- [3] PHP を実装 ([2][3]は比較的交互に行う) (メイン) (かなりかかる)
- [4] デザインを css でそれなりにする. (こだわらない)
- [5] sessawiki の仕組みを解体する. アニメスタッフデータベースについては, 仕組みをお勉強しながら行う. (時間の許す限り)