システム要件定義書

【ペペ　オハナ】

TeamSakura

目次

[１：改定履歴 3](#_Toc146013682)

[２：基本設計 4](#_Toc146013683)

[開発の目的 4](#_Toc146013684)

[動作環境 4](#_Toc146013685)

[ｱﾌﾟﾘｹｰｼｮﾝ登録内容 4](#_Toc146013686)

[３：機能概要 5](#_Toc146013687)

[トップページ機能 5](#_Toc146013688)

[ログイン機能 5](#_Toc146013689)

[ユーザー登録・編集機能 5](#_Toc146013690)

[マイページ表示機能 5](#_Toc146013691)

[ネコちゃんの登録・編集機能 5](#_Toc146013692)

[ネコちゃん検索機能 6](#_Toc146013693)

[メッセージ送信・返信機能 6](#_Toc146013694)

[ログアウト機能 6](#_Toc146013695)

[４：対象ユーザー 6](#_Toc146013696)

[５：操作手順 7](#_Toc146013697)

[６：画面遷移図 8](#_Toc146013698)

[７：アーキテクチャ図 10](#_Toc146013699)

[１　ログイン機能 10](#_Toc146013700)

[２　新規ユーザー登録機能 11](#_Toc146013701)

[３　ユーザー情報編集(ログイン済み) 12](#_Toc146013702)

[４　マイページ表示機能 13](#_Toc146013703)

[５　ネコちゃん登録機能 14](#_Toc146013704)

[６　ネコちゃん登録情報編集 15](#_Toc146013705)

[７　ネコちゃん検索機能 16](#_Toc146013706)

[８　ネコ詳細表示（含むTop） 17](#_Toc146013707)

[９　メッセージ送信機能 18](#_Toc146013708)

[10　メッセージ返信機能 19](#_Toc146013709)

[11　ログアウト機能 20](#_Toc146013710)

[８：データーベース 21](#_Toc146013711)

[データーベース情報 21](#_Toc146013712)

[９：テーブル設計仕様書 21](#_Toc146013713)

[１０：クラス定義書 22](#_Toc146013714)

# １：改定履歴

|  |  |  |  |
| --- | --- | --- | --- |
| 作成日 | 作成者 | 改定内容 | 版 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# ２：基本設計

## 開発の目的

ｐｅｐｅ　ｏｈａｎａは、ハワイの言葉で

さまざまな事情で、ネコちゃんを譲渡したい方と、ネコちゃんを家族に迎えたい方をつなぎ、マッチングできることで、人にとってもネコちゃんにとっても安心して暮らせることができるようになります。

## 動作環境

|  |  |
| --- | --- |
| プログラミング言語 | Ｊａｖａ　17 |
| マークアップ言語 | HTML、ＣＳＳ、ｊａｖａｓｃｒｉｐｔ |
| データベース | ＭｙＳＱＬ 8.0.33 |
| Ｗｅｂブラウザ | Ｍｉｃｒｏｓｏｆｔ　Ｅｄｇｅ115.0.1901.183 (公式ビルド) (64 ビット) |
| 総合開発環境 | Eclipse 2023-03(4.27.0) |

## ｱﾌﾟﾘｹｰｼｮﾝ登録内容

ログイン時

* ユーザーＩＤ
* パスワード

マイページログイン時

* 猫の写真
* 性別
* 年齢
* 体重

# ３：機能概要

## トップページ機能

* 登録されているネコちゃんの新着情報が表示される
* 里親募集検索で絞り込み検索ボタンの表示
* 新規登録画面への移動
* ログインボタンの表示

## ログイン機能

* 一部機能の利用時にログインする必要がある
* ユーザーＩＤとパスワードでログインする
* ログイン中のユーザー名の表示がされる
* ユーザー認証が正しく行われたかった場合、もう一度、ログイン画面に戻る
* マッチングしたい場合には、ログインの必要がある

## ユーザー登録・編集機能

* ユーザーＩＤ、名前、パスワードの登録をする
* ユーザーＩＤは５０文字まで、名前は２０文字まで、パスワードは２０文字まで入力できる
* 正しく行われなかった場合は、もう一度、新規登録画面に戻る
* 登録されているユーザーのユーザーIDと名前の編集ができる

## マイページ表示機能

* ユーザーＩＤと名前の表示
* ユーザー情報の編集ページへ遷移できる
* ネコちゃん情報の登録するページへ遷移できる
* 登録したネコちゃんの情報がある場合、表示される
* 登録されているネコちゃんの編集機能ページへ遷移できる
* マッチング希望の方からのメッセージの受信の確認と返信ができる
* 登録されているネコちゃん情報の削除ができる

## ネコちゃんの登録・編集機能

* マイページからネコちゃん登録画面へ移動して登録できる
* ネコちゃんの名前の登録
* ネコちゃんの名前は２０文字まで入力できる
* ネコちゃんの写真の登録ができる
* 猫種を３０文字までで登録ができる
* 性別を選択して登録できる
* ネコちゃんの年齢の登録ができる
* ネコちゃんの体重の登録ができる
* ネコちゃんの性格や特徴などがコメント入力できる
* ネコちゃんのコメントは２５０文字まで入力できる
* 登録されているネコちゃんの編集ができる

## ネコちゃん検索機能

* 男の子か女の子、男の子と女の子両方の選択ができる
* 検索機能で選択した結果の表示ができる
* 検索結果一覧表示からネコちゃんの表示ボックスをクリックして詳細情報を表示できる
* ネコちゃん詳細情報画面から里親希望の場合のメッセージ送信画面へ遷移できる

## メッセージ送信・返信機能

* マッチング希望するネコちゃん詳細表示画面からメッセージ送信ができる
* マッチング希望のメッセージが受信されると、マイページから返信することができる
* メッセージは５００文字まで入力できる

## ログアウト機能

* ログアウトできるのは、ログイン中のユーザーのみ
* ログイン中の各ページからログアウトできる

# ４：対象ユーザー

* 家族になる「ねこ」を探す方と、「ねこ」の家族を探す方

# ５：操作手順

* ＵＲＬ

[http://localhost:8080/ | pepeohana](http://localhost:8080/%20|%20pepeohana)/

* トップ画面

現在登録されているネコちゃんの一覧が表示される。

男の子、女の子選択して“まっちんぐ”を押して登録されているネコちゃんを検索できる。

未ログインの場合、右上に「未ログイン▼」と表示される。

新規の場合、右上の「未ログイン▼」を押して“新規ユーザー登録”を選択して新規ユーザー登録画面へ遷移できる。

未ログインの場合、右上の「未ログイン▼」を押して“ログイン”を選択してログイン画面へ遷移できる。

* 新規ユーザー登録

「ユーザーＩＤ」　メールアドレスで５０文字まで登録できる。

「ユーザー名」　２０文字まで登録できる。

「パスワード」　２０文字まで登録できる。

* ログイン画面

登録している「ユーザーＩＤ」と「パスワード」を入力してログインできる。

ログインに成功すると右上に登録している「ユーザー名」が表示される。

ログインに失敗した場合、「ユーザーＩＤ」と「パスワード」入力画面が再度表示される。

* マイページ画面

ログイン中の「ユーザー名▼」を押して“マイページ”を選択してマイページへ遷移できる。

マイページで登録されているユーザー情報とネコ情報の編集ができる。

* ログアウト

ログイン中の「ユーザー名▼」を押して“ログアウト”を選択してログアウトできる。

ログアウト完了のメッセージ画面が表示される。

“ホームに戻る”を押してトップページへ遷移する。

# ６：画面遷移図



画面遷移図（ログイン・ユーザー情報・ペット情報登録）

# ７：アーキテクチャ図

## １　ログイン機能

index.jsp

button.login

LoginBL.java

executeSelectUserInfo

Exelogin.java  
doPost

login.jsp

　　userid

　　pass

　　inputUserId

　　inputPassWord

UsersInfoDao.java

doSelect

usersInfo

　　USER\_ID

　　USER\_NAME

PASSWORD

settion.ｓetAttribute

(LOGIN\_INFO,dto)

（dto.getUserId() != null）

　　　（成功）　　　　　　　　（失敗）

index.jsp

login.jsp

## ２　新規ユーザー登録機能

index.jsp

button.regist

registUser.jsp

　　userid

　　username

　　password

　　USER\_ID

　　USER\_NAME

PASSWORD

　　inputUserId

　　inputPassWord

ExeRegistUser..java

doPost

UsersInfoDao.java

doInsert

UsersInfo

RegistUserBL.java

executinsertUserInfo

**successInsert**

**isSuccess**

**true**

**false**

index.jsp

registusererror.html

registUser.jsp

error.html

finish.html

**isSuccess**

**successInsert**

error.html

finish.html

**isSuccess**

**successInsert**

## ３　ユーザー情報編集

mypage.jsp

button.update

EditUser.java

doGet

**dto!=null**

**false**

**true**

ｓetAttribute (user)

mypage.jsp

index.jsp

getAttribute (user)

EditUser．ｊｓｐ

dto

request.getParameter

ID

USERID

USERNAME  
　　PASSWORD

UsersInfoDao.java

doUpdateUser (Id)

**dto**

UpdateUserBL.java

executeUpdateUser（dto）

ExeEditUser.java

doPost

usersInfo

dto

USERID

USERNAME  
　　PASSWORD

**isSuccess**

**successInsert**

ExeMyPage.java

EditUserError.html

**true**

**false**

mypage.ｊｓｐ

mypage.ｊｓｐ

## ４　マイページ表示機能

index.jsp

button.login

ExeMyPage.java

doGet

**userID**

MyPageDao.java

doselectMyInfo (Id)

MyPageBL.java

executeUpdateUser（dto）

**dto**

**dto**

usersInfo

messages

catsinfo

mypage.jsp

dto

USERID

USERNAME  
　　PASSWORD

　　CATID

　　CATNAME

BIRTH

GENDER

WEIGHT

　　IMAGE

　　COMMENT

　　messages

## ５　ネコちゃん登録機能

　mypage.jsp

button.toroku

**null**

index.jsp

RegistCat.java

settion.ｇetAttribute

(LOGIN\_INFO,dto)

**name**

registcat．ｊｓｐ

CatsInfo

　getParameter

CATID

　　CATNAME

　　KIND

BIRTH

GENDER

WEIGHT

　　IMAGE

　　COMMENT

　　USERID

　　REG\_DATE

　　UP\_DATE

,

RegistCatBL.java

executeInsertCat（dto）

CatsInfoDao.java

doInsertcatinfo

catsinfo

ExeRegistCat

.java

dopost

**isSuccess**

**SuccessInsert**

**true**

**false**

Mypage.java

registcaterror.html

registcat．ｊｓｐ

RegistCat.java

　mypage.jsp

## ６　ネコちゃん登録情報編集

**dto**

Update

CATID

　　CATNAME

　　KIND

BIRTH

GENDER

WEIGHT

　　IMAGE

　　COMMENT

　　USERID

　　REG\_DATE

　　UP\_DATE

ID

**ID**

CatsInfoDao.java

doSelectCatInfo()

Id)

**successInsert**

**userId**

**dto**

catsinfo

dto

CATID

　　CATNAME

　　KIND

BIRTH

GENDER

WEIGHT

　　IMAGE

　　COMMENT

　　USERID

　　REG\_DATE

　　UP\_DATE

　　DEL

WHERE ID=?

ExeEditCat.java

doGet

ShowCatInfoBL.java

executeSelectSurvey（userId）

**dto**

ｓetAttribute (caｔ)

editCat．ｊｓｐ

getAttribute (caｔ)

ExeEditCat.java

doPost

CatsInfoDao.java

doUpdateCat (Id)

catsinfo

UpdateCatBL.java

executeUpdateSurvey（dto）

**isSuccess**

**true**

**false**

mypage.ｊｓｐ

editCatError.jsp

mypage.jsp

button.update

CATID

　　CATNAME

　　KIND

BIRTH

GENDER

WEIGHT

　　IMAGE

　　COMMENT

　　USERID

　　REG\_DATE

　　UP\_DATE

　　DEL

WHERE ID=?

WHERE ID=?

削除機能も含める

editCat．ｊｓｐ

## ７　ネコちゃん検索機能

index.ｊｓｐ

ExeSearchCat.java

doGet

CatsInfoDao.java

doSelectCatLists()

SearchCatBL.java

executeSelectSurvey（userId）

**dto**

**dto**

catsinfo

dto

　　CATID

　　CATNAME

BIRTH

KIND

GENDER

　　IMAGE

ｃatLists.ｊｓｐ

## ８　ネコ詳細表示（含むTop）

catLists.ｊｓｐ

CatsInfoDao.java

doSelectCatInfo()

ShowCatInfoBL.java

executeSelectSurvey（userId）

showCatInfo.ｊｓｐ

ExeShowCatInfo.java

doGet

**dto**

**dto**

catsinfo

dto

　　CATID

　　CATNAME

BIRTH

GENDER

WEIGHT

　　IMAGE

　　COMMENT

　　REG\_DATE

　　UP\_DATE

## ９　メッセージ送信機能

**catid・catname**

showCatInfo.ｊｓｐ

Message.java

message.ｊｓｐ

SendMessageBL.java

ExeSendMessage.java

doPost

MessagesDao.java

doinsertMessage()

**false**

**true**

**isSuccess**

**sendSuccessrt**

messages

dto

　　ID

　　CATID

　　RECEIVERID

　　M\_DATE

　　MESSAGED

　　SENDERID

senderror.html

ExeMyPage.java

Message.java

mypage.ｊｓｐ

message.ｊｓｐ

## 10　メッセージ返信機能

mypage.ｊｓｐ

ExeReplyMessage.java

doGet

**catid**

**id**

MessagesDao.java

doselectmessage

()

ReplyMessageBL.java

**dto**

**dto**

catinfo

message.ｊｓｐ

ReplyMessageBL.java

MessagesDao.java

doinsertreplyms()

ExeReplayMessage.java

doPost

**id・message**

**id**

**false**

**true**

**isSuccess**

**sendSuccessrt**

messages

dto

　　ID

　　CATID

　　RECEIVERID

　　M\_DATE

　　MESSAGED

　　SENDERID

replyerror.html

Message.java

ExeMyPage.java

mypage.ｊｓｐ

mypage.ｊｓｐ

## 11　ログアウト機能

ログイン後のすべてのｊｓｐ

ExeLogout.java

doGet

（フォワード）

settion.ｇetAttribute

(LOGIN\_INFO,dto)

　　　　　　　session.invalidate();

compLogout．ｊｓｐ

# ８：データーベース

## データーベース情報

# ９：テーブル設計仕様書

# １０：クラス定義書