

高知工科大学様
大学掲示板 (KUTBBS)
内部設計書 v1.0

グループ 10
Pathfinder

平成 30 年 12 月 7 日

目次

1	システム概要	4
2	動作環境	4
3	開発環境	5
4	データベース設計	5
4.1	データテーブル構成	5
4.2	テーブル設計	6
4.2.1	ユーザテーブル (user)	6
4.2.2	スレッドテーブル (thread)	7
4.2.3	レステーブル (ress)	8
4.2.4	スレッドカテゴリテーブル (thread_category)	9
4.2.5	お気に入り掲示板テーブル (favorite.bbs)	9
4.2.6	コレクトユーザテーブル (collect_user)	10
4.2.7	不適切な単語登録テーブル (inadequacy_word)	10
4.2.8	通報テーブル (report)	11
4.2.9	凍結ユーザテーブル (suspend)	12
4.2.10	お知らせテーブル (notice)	12
5	モジュール設計	13
5.1	admin_top.html	13
5.2	admin_child_top.html	13
5.3	create_admin.html	13
5.4	new_admin.html	14
5.5	delete_admin.html	14
5.6	delete_admin_ok.html	14
5.7	user_admin_top.html	14
5.8	create_user.html	15
5.9	new_user.html	15
5.10	search_user.html	15
5.11	user_info.html	15
5.12	login_top.html	16
5.13	login_user.html	16
5.14	subscribe_user.html	17
5.15	subscribe_ok.html	17
5.16	read_news.html	17
5.17	login_admin.html	18
5.18	news_edit.html	18
5.19	news_edit_confirm.html	18
5.20	news_edit_ok.html	19
5.21	search.html	19
5.22	report_reason.html	19

5.23	report_confirm.html	19
5.24	report_ok.html	20
5.25	NGword.html	20

1 システム概要

本システムは本校の学生同士によって問題提起から問題解決までを匿名で行えるようにするためのシステムである。

本システムは Web ブラウザ上で「掲示板システム」「ユーザシステム」「管理者システム」によって構築される各要素の主な機能は以下に示す。

- 掲示板システム
 - － 掲示板サブシステム
 - － 検索サブシステム
 - － コレクトボタンサブシステム
 - － 通報サブシステム
 - － 通知サブシステム
- ユーザシステム
 - － アカウント登録サブシステム
 - － ログインサブシステム
 - － マイページサブシステム
 - － ブックマークサブシステム
 - － お知らせ表示サブシステム
- 管理者システム
 - － 管理者ログインサブシステム
 - － 子管理者管理サブシステム
 - － お知らせ編集サブシステム
 - － 掲示板編集サブシステム
 - － ユーザ管理サブシステム

2 動作環境

本システムの動作環境は以下の通りである。

- 動作環境
 - － CPU : ARM Cortex-A53 1GHz 以上
 - － GPU : Broadcom VideoCore IV
 - － メモリ : 2GB 以上
 - － ストレージ : 4GB eMMC / SD カード PIN
 - － OS
 - * Linux version

- * ubuntu version
- * MacOS version
- * Windows7 , 8 , 10
- * iOS 10.0
- web サーバ : AWS EC2
- App サーバ : Ruby on Rails version 5.1.4
- RDBMS : MySQL version 8.0
- 使用ブラウザ : GoogleChrome62.0 , Firefox version 57.0

3 開発環境

本システムの開発環境は以下の通りである。

- OS : Linux , ubuntu , MacOS , Windows , iOS10.0
- 使用言語
 - ruby version 2.4.2
 - Ruby on Rails version 5.1.4
 - HTML5
 - CSS
 - JavaScript
- サーバ : Raspberry Pi 3
- データベース : MySQL version 8.0
- web サーバ : AmazonWebServices

4 データベース設計

本章では本システムにおけるデータベース MySQL についてを示す。

4.1 データテーブル構成

本小節ではデータテーブルの構成を ER 図を用いて示す。

4.2 テーブル設計

本小節ではデータベース構成するテーブルについて示す。また、各カラムについても詳細も示す。

4.2.1 ユーザテーブル (user)

ユーザテーブルではユーザに関する情報を管理する。このテーブルの詳細は表 1 に示す。

表 1: ユーザテーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
学籍番号	student_id	char	7	×	-	PK
ユーザ ID	user_id	char	20	×	-	-
パスワード	password	char	20	×	-	-
保留ポイント	now_point	int	4	×	-	-
取り消しフラグ	cancel_flag	int	1	×	-	-
色フラグ	color_flag	int	1	×	-	-
斜線フラグ	diagonal_flag	int	1	×	-	-
太文字フラグ	bold_letters_flag	int	1	×	-	-
管理者フラグ	administrator_flag	int	1	○	-	-
初回ログインフラグ	first_login_flag	int	1	×	-	-
ログイン回数	count_login	int	4	×	-	-

- 学籍番号

ユーザの学籍番号を示す値であり、自テーブルの主キーである。値は 7 文字の半角数字にて構成される。
(例:1200***)

- ユーザ ID

ユーザを識別するための ID を示す値である。値は最大 20 文字の半角英字と半角数字の組み合わせにて構成される。(例:user1234)

- パスワード

ユーザを識別するためのパスワードを示す値である。値は最大 20 文字の半角英数字と半角数字の組み合わせにて構成される。(例:pass1234)

- 保留ポイント

ユーザがログインした回数、コレクトボタンを 10 回以上押されたことで獲得したポイントを示す値である。値は 4 桁の数値にて構成される。(例:9999)

- 取り消しフラグ

拡張機能の 1 つであり、取り消し線を付与することを示す値である。値は 1 桁の 0(OFF)、1(ON) の数値にて構成される。(例:1 → 取り消し線付与)

- 色フラグ

拡張機能の 1 つであり、色を付与することを示す値である。値は 1 桁の 0(OFF)、1(ON) の数値にて構成される。(例:1 → 色付与)

色フラグは 1(ON) にした場合、文字を赤色に変更することができるよう想定している。

- 斜線フラグ

拡張機能の 1 つであり、斜線を付与することを示す値である。値は 1 桁の 0(OFF)、1(ON) の数値にて構成される。(例:1 → 斜線付与)

- 太文字フラグ
拡張機能の1つであり、太文字を寄与することを示す値である。値は1桁の0(OFF)、1(ON)の数値にて構成される。(例:1 → 太文字付与)
- 管理者フラグ
親管理者と子管理者を識別することを示す値である。値は1桁の0(子管理者)、1(親管理者)の数値にて構成される。(例:1 → 親管理者と識別される)
- 初回ログインフラグ
初回ログインを識別することを示す値である。値は1桁の0(OFF)、1(ON)の数値にて構成される。(例:1 → 初回ログインと識別)
- ログイン日数
ユーザがログインをした日数の回数を記録することを示す値である。値は4桁の数値にて構築される。(例:9999)

4.2.2 スレッドテーブル (thread)

スレッドテーブルではスレッドに関する情報を管理する。このテーブルの詳細は表2に示す。

表 2: スレッドテーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
スレッド No	thread_no	int	5	×	unsigned zerofill auto_increment	PK
スレッドカテゴリ No	thread_category_no	int	3	×	unsigned zerofill auto_increment	FK:thread_category
学籍番号	student_id	char	7	×	-	FK:user
スレッドタイトル名	thread_category_name	char	50	×	-	-
スレッド非表示フラグ	thread_hide_flag	int	1	×	-	-
作成日時	created_at	timestamp	-	×	default current_timestamp	-
更新日時	updated_at	timestamp	-	×	default current_timestamp on updat current_timestamp	-

- スレッド No
自テーブルの主キーである。値は5桁の数値に構成され自動追加される。(例:00001 → 対応するスレッド No が追加される)
- スレッドカテゴリ No
スレッドカテゴリテーブルを参照する際の外部キーである。値は3桁の数値に構成され自動追加される。(例:001 → 対応するスレッドカテゴリ No が追加される)
- 学籍番号
ユーザテーブルを参照する際の外部キーである。値は7文字の半角数字にて構成される。(例:1200***)
- スレッドタイトル名
ユーザが作成したスレッドタイトルを示す値である。値は最大50文字の全角文字にて構成される。(例: 研究室どこにしようか …)

- スレッド非表示フラグ

管理者の操作権限でスレッドを非表示するを示す値である。値は1桁の数値で0(OFF)、1(ON)の数値にて構成される。(例:1→対象のスレッドを非表示にする)

- 作成日時

レコードを作成した日付・時刻を示す値である。レコードが作成されるたびに自動追加を行う。(例:2015-05-12 17:30:36)

- 更新日時

レコードを更新した日付・時刻を示す値である。レコードが更新されるたびに自動更新を行う。(例:2015-05-12 17:31:07)

4.2.3 レステーブル (ress)

レステーブルではレスに関する情報を管理する。このテーブルの詳細は表3に示す。

表 3: レステーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
レス No	ress_no	int	5	×	unsigned zerofill auto_increment	PK
スレッド No	thread_no	int	5	×	unsigned zerofill auto_increment	FK:thread
学籍番号	student_id	char	7	×	-	FK:user
レス ID	ress_id	char	8	×	-	-
書き込み名	write_name	varchar	15	×	-	-
書き込み内容	write_content	varchar	200	×	-	-
コレクトプッシュ数	collect_push_count	int	4	×	-	-
レス非表示フラグ	ress_hide_flag	int	1	×	-	-
作成日時	created_at	timestamp	-	×	default current_timestamp	-

- レス No

自テーブルの主キーである。値は5桁の数値で構成され自動追加される。(例:00001→対応するレス Noが追加される)

- スレッド No

スレッドテーブルを参照する際の外部キーである。値は5桁の数値に構成され自動追加される。(例:00001→対応するレス Noが追加される)

- 学籍番号

ユーザテーブルを参照する際の外部キーである。値は7文字の半角数字にて構成される。(例:1200***)

- レス ID

ユーザが書き込みを行う際に表示するIDを示す値である。値は最大8文字の半角英数か半角数字にて構成される(例:ero1o23u)

- 書き込み名

書き込みを行う際に表示する名前を示す値である。値は15文字以下の全角文字にて構成される。(例:名無しさん)

- 書き込み内容
書き込み内容を示す値である。値は 200 文字以下の全角文字にて構成される。(例:自分のやりたいことができる所でええんやで)
- コレクトボタンプッシュ数
レスに対してコレクトボタンが押された回数を示す値である。値は 4 桁の数値にて構成される。(例:10[コレクトボタンが押された回数])
- レス非表示フラグ
管理者の操作権限でレスを非表示するを示す値である。値は 1 桁の数値で 0(OFF)、1(ON) の数値にて構成される。(例:1 →対象のレスを非表示にする)
- 作成日時
レコードを作成した日付・時刻を示す値である。レコードが作成されるたびに自動追加を行う (例:2015-05-12 17:30:36)

4.2.4 スレッドカテゴリテーブル (thread_category)

スレッドカテゴリテーブルは各カテゴリごとに分けられたスレッドの情報を管理する。このテーブルの詳細は表 4 に示す。

表 4: スレッドカテゴリテーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
カテゴリ No	category_no	int	3	×	unsigned zerofill auto_increment	PK
カテゴリ名	category_name	varchar	15	×	-	-

- カテゴリ No
自テーブルの主キーである。値は値は 3 桁の数値に構築され自動追加される。(例:001 →対応するカテゴリ No が追加される)
- カテゴリ名カテゴリの名前を示す値である。値は 15 文字以下の全角文字にて構成される。(例:情報学群)

4.2.5 お気に入り掲示板テーブル (favorite_bbs)

お気に入り掲示板テーブルではブックマークとして登録したスレッド情報を管理する。このテーブルの詳細は表 5 に示す。

表 5: お気に入り掲示板テーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
学籍番号	student_id	char	7	×	-	FK:user
スレッド No	thread_no	int	5	×	unsigned zerofill auto_increment	FK:thread

- 学生番号
ユーザテーブルを参照する際の外部キーである。値は 7 文字の半角数字にて構成される。(例:1200***)
- スレッド No
スレッドテーブルを参照する際の外部キーである。値は 5 桁の数値に構築され自動追加される。(例:00001
→対応するレスド No が追加される)

4.2.6 コレクトユーザテーブル (collect_user)

コレクトユーザテーブルではコレクトボタンを押されたことに関する情報を管理する。このテーブルの詳細は表 6 に示す。

表 6: コレクトユーザテーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
学籍番号	student_id	char	7	×	-	FK:user
スレッド No	thread_no	int	5	×	unsigned zerofill auto_increment	FK:thread
レス No	ress_no	int	5	×	unsigned zerofill auto_increment	FK:ress

- 学生番号
ユーザテーブルを参照する際の外部キーである。値は 7 文字の半角数字にて構成される。(例:1200***)
- スレッド No
スレッドテーブルを参照する際の外部キーである。値は 5 桁の数値にて構築され自動追加される。(例:00001
→対応するレスド No が追加される)
- レス No
レステーブルを参照する際の外部キーである。値は 5 桁の数値にて構築され自動追加される。(例:00001
→対応するレス No が追加される)

4.2.7 不適切な単語登録テーブル (inadequacy_word)

不適切な単語登録テーブルでは管理者が誹謗中傷や公序良俗に違反していると考えられる単語を登録した情報を管理する。このテーブルの詳細は表 7 に示す。

表 7: 不適切な単語登録テーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
単語 No	word_no	int	5	×	unsigned zerofill auto_increment	PK
単語名	word_name	varchar	200	×	-	-

- 単語 No
自テーブルの主キーである。値は 5 桁の数値に構成され自動追加される。(例 00001 →対応する単語 No
が追加される)

- 単語名

管理者の操作権限で登録した単語を示す値である。値は 200 文字以下の全角文字にて構成される。(例: バカ)

4.2.8 通報テーブル (report)

通報テーブルではユーザが誹謗中傷や公序良俗に違反するなどの不適切な内容であると判断したスレッドまたはレスを管理者に通報した時の情報を管理する。また、通報した内容の情報も管理する。このテーブルの詳細は表 8 に示す。

表 8: 通報テーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
通報 No	report_no	int	5	×	unsigned zerofill auto_increment	PK
通報ユーザ ID	report_user_id	char	20	×	-	FK:user
スレッド No	thread_no	int	5	×	unsigned zerofill auto_increment	FK:thread
レス No	ress_no	int	5	×	unsigned zerofill auto_increment t	FK:ress
通報内容	report_content	varchar	200	○	-	-
通報日時	report_at	timestamp	-	×	default current_timestamp	-

- 通報 No

自テーブルの主キーである。値は 5 桁の数値に構成され自動追加される。(例 00001 → 対応する通報 No が追加される)

- 通報ユーザ ID

ユーザテーブルを参照する際の外部キーである。値は最大 20 文字の半角英字と半角数字の組み合わせにて構成される。(例:user1234)

- スレッド No

スレッドテーブルを参照する際の外部キーである。値は 5 桁の数値に構築され自動追加される。(例:00001 → 対応するスレッド No が追加される)

- レス No

レステーブルを参照する際の外部キーである。値は 5 桁の数値に構築され自動追加される。(例:00001 → 対応するレス No が追加される)

- 通報内容

通報内容を示す値である。値は 200 文字以下の全角文字にて構成される。(例:このユーザ○○は誹謗中傷に反している思ったから)

- 通報日時

レコードを作成した日付・時刻を示す値である。レコードが作成されるたびに自動追加を行う。(例:2015-05-12 17:30:36)

4.2.9 凍結ユーザテーブル (suspend)

凍結ユーザテーブルでは迷惑行為が改善されないユーザのアカウントに対しての情報を管理する。このテーブルの詳細は表 9 に示す。

表 9: 凍結ユーザテーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
凍結 No	suspend_no	int	4	×	lunsigned zerofill auto_increment	PK
学籍番号	student_id	char	7	×	-	FK:user

- 凍結 No
自テーブルの主キーである。値は 3 桁の数値に構成され自動追加される。(例:00001 →対応する凍結 No が追加される)
- 学籍番号
ユーザテーブルを参照する際の外部キーである。値は 7 文字の半角数字にて構成される。(例:1200***)

4.2.10 お知らせテーブル (notice)

お知らせテーブルではお知らせに関する情報を管理する。このテーブルの詳細は表 10 に示す。

表 10: お知らせテーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
お知らせ No	notice_no	int	5	×	lunsigned zerofill auto_increment	PK
お知らせタイトル	notice_title	varchar	50	×	-	-
お知らせ内容	notice_title	varchar	400	×	-	-
作成日時	created_at	timestamp	-	×	default current_timestamp	-

- お知らせ No
自テーブルの主キーである。値は 4 桁の数値に構成され自動追加される。(例:00001 →対応するお知らせ No が追加される)
- お知らせタイトル名
管理者が作成したお知らせタイトルを示す値である。値は 50 文字以下の全角文字にて構成される。(例:現在確認している不具合について)
- お知らせ内容
お知らせ内容を示す値である。値は 400 文字以下の全角文字にて構成される。(例:KUTBBS 運営事務局です。…)
- 作成日時
レコードを作成した日付・時刻を示す値である。レコードが作成されるたびに自動追加を行う。(例:2015-05-12 17:30:36)

5 モジュール設計

5.1 admin_top.html

【名称】

管理者 TOP 画面

【概要】

管理者がログインを済ませると最初に現れる画面である。ここでは管理者に与えられた権限の選択をすることができる。

【処理フロー】

- 「掲示板はこちらボタン」を押すと、が呼び出される。
- 「子管理者管理ボタン」を押すと、admin_child_top.html が呼び出される。
- 「ユーザ管理ボタン」を押すと、が呼び出される。
- 「お知らせ編集ボタン」を押すと、が呼び出される。
- 「通報状況確認ボタン」を押すと、が呼び出される。
- 「不適切な単語登録ボタン」を押すと、が呼び出される。

5.2 admin_child_top.html

【名称】

子管理者管理 TOP

【概要】

各子管理者について発行・抹消を行うための画面である。

【処理フロー】

- 「子管理者アカウント発行ボタン」を押すと、create_admin.html が呼び出される。
- 「子管理者抹消ボタン」を押すと、delete_admin.html が呼び出される。
- 「管理者 TOP ボタン」を押すと、admin_top.html が呼び出される。

5.3 create_admin.html

【名称】

子管理者アカウント発行画面

【概要】

子管理者アカウントの発行を行うかどうかを決定する画面である。

【処理フロー】

- 「はいボタン」を押すと、.html.erb で処理が行われる。
- .html.erb で新たなアカウントの発行が完了すると new_admin.html が呼び出される。
- 「いいえボタン」を押すと、admin_child_top.html が呼び出される。

5.4 new_admin.html

【名称】

子管理者アカウント発行完了画面

【概要】

子管理者アカウント発行完了画面が表示される

【処理フロー】

- 「確認ボタン」を押すと、admin_child_top.html が呼び出される。
- 「管理者 TOP ボタン」を押すと、admin_top.html が呼び出される。

5.5 delete_admin.html

【名称】

子管理者アカウント抹消画面

【概要】

特定の子管理者アカウントについて抹消するかどうかを決定する画面である。

【処理フロー】

- 「はいボタン」を押すと、.html.erb で処理が行われる
- .html.erb で子管理者アカウントの抹消が完了すると delete_admin_ok.html が呼び出される。
- 「いいえボタン」を押すと、admin_child_top.html が呼び出される。

5.6 delete_admin_ok.html

【名称】

子管理者アカウント抹消完了画面

【概要】

子管理者アカウント抹消完了画面が表示される

【処理フロー】

- 「戻るボタン」を押すと、admin_child_top.html が呼び出される。

5.7 user_admin_top.html

【名称】

ユーザ管理画面

【概要】

ユーザのアカウント発行とユーザ情報検索を行うことができる画面である。

【処理フロー】

- 「ユーザアカウント発行ボタン」を押すと、create_user.html が呼び出される。
- 「ユーザ情報検索ボタン」を押すと、search_user.html が呼び出される。

5.8 create_user.html

【名称】

ユーザアカウント発行画面

【概要】

ユーザアカウントの発行を行うことができる画面である。学籍番号の開始番号と末尾番号を入力することで範囲を指定し、同時に複数のアカウントの発行を行う。

【処理フロー】

- 「開始番号テキストボックス」に新しく発行するユーザの開始番号を入力する。
- 「末尾番号テキストボックス」に新しく発行するユーザの末尾番号を入力する。
- 「登録ボタン」を押すと、.html.erb で処理が行われる。
- .html.erb でユーザのアカウント発行が完了すると new_user.html が呼び出される。
- 「管理者 TOP ボタン」を押すと、admin_top.html が呼び出される。

5.9 new_user.html

【名称】

ユーザアカウント発行確認画面

【概要】

ユーザアカウント発行確認画面が表示される

【処理フロー】

- 「確認ボタン」を押すと、user_admin_top.html が呼び出される。

5.10 search_user.html

【名称】

ユーザ情報検索画面

【概要】

【処理フロー】

- 「学籍番号検索テキストボックス」にはユーザの情報を入手する際に学籍番号を入力する。
- 「検索ボタン」を押すと、.html.erb で処理が行われる。
- .html.erb で該当するユーザがあれば user_info.html が呼び出される。

5.11 user_info.html

【名称】

ユーザ情報画面

【概要】

ユーザ情報の検索結果を表示する画面である。【処理フロー】

- 「警告・注意喚起ボタン」を押すと、が呼び出される。
- 「アカウント凍結ボタン」を押すと、が呼び出される。
- 「パスワード表示ボタン」を押すと、が呼び出される。
- 「管理者 TOP ボタン」を押すと、admin_top.html が呼び出される。

5.12 login_top.html

【名称】

ログイン TOP 画面

【概要】

利用者が本システムを利用する際の初期画面である。ここでは、ユーザとしてログイン、管理者としてログインのいずれかを選択することができる。

【処理フロー】

- ログインボタンを押すと login_user.html が呼び出される。
- 管理者はこちらボタンを押すと login_admin.html が呼び出される。

5.13 login_user.html

【名称】

ユーザログイン画面

【概要】

ユーザが ID とパスワードを入力して本システムにログインするための画面である。

【処理フロー】

- 「ID 入力テキストボックス」にユーザの ID を入力する。
- 「パスワード入力テキストボックス」にユーザのパスワードを入力する。
- 初回のログインでは、ID とパスワードが入力された状態で「ENTER ボタン」を押すと subscribe_user.html が呼び出される。
- 2回目以降のログインでは、ID とパスワードが入力された状態で「ENTER ボタン」を押すと BBS_top.html が呼び出される。
- データベース上に存在しない ID またはパスワードが入力された状態で「ENTER ボタン」を押すと、.html.erb が呼び出される。
- いずれかのテキストボックスに何も入力されていない状態で「ENTER ボタン」を押すと、.html.erb が呼び出される。

5.14 subscribe_user.html

【名称】

新規登録画面

【概要】

初回ログインをしたユーザが ID とパスワードを変更し、本システムに本登録する画面である。

【処理フロー】

- 「ID 入力テキストボックス」に変更後のユーザ ID を入力する。
- 「パスワード入力テキストボックス」に変更後のユーザのパスワードを入力する。
- 「パスワード再確認入力テキストボックス」に「パスワード入力テキストボックス」に入力したものと
同じパスワードを入力する。
- 「ID」「パスワード」「パスワード再確認」のテキストボックスに文字列が入力されており、かつ「パスワード」と「パスワード再確認」のテキストボックスに入力された文字列が一致した場合において、「登録ボタン」を押すと subscribe_ok.html が呼び出される。
- いずれかのテキストボックスに何も入力されていない状態で「登録ボタン」を押すと、.html.erb が呼び出される。
- 「登録ボタン」を押したときに「パスワード」と「パスワード再確認」に入力された文字列が合致しなかった場合、.html.erb が呼び出される。

5.15 subscribe_ok.html

【名称】

新規登録完了画面

【概要】

新規登録が完了したことを表示する画面である。

【処理フロー】

- 「利用を開始するボタン」を押すと、BBS_top.html が呼び出される。

5.16 read_news.html

【名称】

お知らせ画面【概要】

管理者からユーザに対するお知らせの詳細を表示する画面である。

5.17 login_admin.html

【名称】

管理者ログイン画面

【概要】

管理者が ID とパスワードを入力して本システムにログインするための画面である。

【処理フロー】

- 「管理者 ID 入力テキストボックス」に管理者の ID を入力する。
- 「管理者パスワード入力テキストボックス」に管理者のパスワードを入力する。
- 入力された ID とパスワードがともにデータベースに登録されたものと合致している場合、「ENTER ボタン」を押すと admin_top.html が呼び出される。
- 入力された ID とパスワードのいずれかがデータベースに登録されているものと合致しない場合、「ENTER ボタン」を押すと、.html.erb が呼び出される。
- ID とパスワードのいずれかが入力されていない場合、「ENTER ボタン」を押すと、.html.erb が呼び出される。

5.18 news_edit.html

【名称】

お知らせ編集画面

【概要】

管理者がユーザにお知らせを通知する内容を編集する画面である。

【処理フロー】

- お知らせタイトル、内容に文字列が入力されている状態で「送信ボタン」を押すと、news_edit_confirm.html が呼び出される。
- お知らせタイトル、内容のいずれかに文字列が入力されていない状態で「送信ボタン」を押すと、.html.erb が呼び出される。
- 「管理者 TOP ボタン」を押すと、admin_top.html が呼び出される。

5.19 news_edit_confirm.html

【名称】

お知らせ内容確認画面

【概要】

お知らせ編集画面で編集したお知らせ内容を確認する画面である。内容を確認し、送信するか否かを選択する。

【処理フロー】

- 「はいボタン」を押すと、.html.erb が呼び出される。また、news_edit_ok.html が呼び出される。
- 「いいえボタン」を押すと、news_edit.html が呼び出される。

5.20 news_edit_ok.html

【名称】

お知らせ作成完了画面

【概要】

お知らせの作成が完了したことを示す画面である。

【処理フロー】

- 「戻るボタン」を押すと admin_top.html が呼び出される。

5.21 search.html

【名称】

検索結果画面

【概要】

利用者が検索を行った際に表示される画面。スレッドタイトルを押すとそのスレッドの閲覧画面が表示される。

【処理フロー】

- 「スレッドタイトル」を押すと、read_thread.html を呼び出す。

5.22 report_reason.html

【名称】

通報画面

【概要】

利用者が通報ボタンを押した際に表示される画面。

【処理フロー】

- 「理由テキストボックス」に通報する理由を入力する。
- 「送信ボタン」を押すと、report_confirm.html を呼び出す。

5.23 report_confirm.html

【名称】

通報確認画面

【概要】

利用者が通報画面にて、「送信ボタン」を押した際に表示される。

【処理フロー】

- 「はいボタン」を押すと、report_ok.html を呼び出します。
- 「いいえボタン」を押すと、report_reason.html を呼び出す。

5.24 report_ok.html

【名称】

通報完了画面

【概要】

利用者が通報確認画面にて、「はいボタン」を押した際に表示される画面。

【処理フロー】

- 「戻るボタン」を押すと category_top.html を呼び出す。

5.25 NGword.html

【名称】

不適切な単語の登録画面

【概要】

管理者が掲示板において不適切な単語を登録する画面。「不適切な単語入力テキストボックス」に入力した単語を「不適切な単語登録ボタン」を押すことで、不適切な単語として登録ができる。表示されている単語の「削除ボタン」を押すことで、その単語を不適切な単語から外す。

【処理フロー】

- 「不適切な単語入力テキストボックス」に単語を入力し、「不適切な単語登録ボタン」押すことで表示されている単語の一覧に入力された単語を追加し、(不適切な単語のデータベース) に追加する。
- 単語の「削除ボタン」を押すことで、不適切な単語の一覧からその単語を外し、(不適切な単語のデータベース) から外す。