

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

2018 年 12 月 16 日

# 目 次

1	システム概要	6
2	動作環境	6
3	開発環境	7
4	内部設計書作成方針	7
4.1	コーディング規約	7
4.1.1	命名規約	7
4.1.2	コーディングスタイル	8
4.2	設計書作成環境	8
4.3	サーバ環境	8
4.4	テーブル設計	8
4.4.1	ユーザテーブル (user)	9
4.4.2	スレッドテーブル (thread)	11
4.4.3	レステーブル (res)	12
4.4.4	スレッドカテゴリテーブル (thread_category)	13
4.4.5	お気に入り掲示板テーブル (favorite.bbs)	13
4.4.6	コレクトユーザテーブル (collect_user)	14
4.4.7	不適切な単語登録テーブル (inadequacy_word)	14
4.4.8	通報テーブル (report)	15
4.4.9	凍結ユーザテーブル (suspend)	15
4.4.10	お知らせテーブル (news)	16
4.4.11	警告注意テーブル (warned_caution)	16
5	サブシステムとシーケンス図	17
5.1	アカウント登録サブシステム (ユーザ用)	17
5.1.1	新規登録機能	17
5.2	ログインサブシステム (ユーザ用)	19
5.2.1	ユーザ認証機能	19
5.2.2	ログインボーナス機能	21
5.2.3	ログアウト機能	22
5.3	検索サブシステム (ユーザ用)	22
5.3.1	検索機能	22
5.3.2	カテゴリ選択機能	23
5.4	掲示板サブシステム (ユーザ用)	24
5.4.1	スレッド作成機能	24
5.4.2	レス書き込み機能	25
5.5	お知らせサブシステム (ユーザ用)	27
5.5.1	お知らせ閲覧機能	27
5.6	通報サブシステム (ユーザ用)	27
5.6.1	スレッド・レス通報機能	27
5.7	コレクトボタンサブシステム (ユーザ用)	28

5.7.1	コレクト認定機能	28
5.7.2	ポイント獲得機能	28
5.8	マイページサブシステム (ユーザ用)	28
5.8.1	ユーザ情報設定機能	28
5.8.2	ブックマーク閲覧機能	28
5.8.3	拡張機能	29
5.8.4	通知設定機能	29
5.9	通知サブシステム (ユーザ用)	29
5.9.1	通知機能	29
5.10	ブックマークサブシステム (ユーザ用)	29
5.10.1	ブックマーク登録機能	29
5.10.2	ブックマーク解除機能	29
5.11	子管理者管理サブシステム (管理者用)	29
5.11.1	子管理者用 ID・パスワード発行機能	30
5.11.2	子管理者アカウント抹消機能	32
5.11.3	子管理者情報閲覧機能	33
5.12	お知らせ編集サブシステム (管理者用)	33
5.12.1	お知らせ編集機能	33
5.12.2	お知らせ表示機能	35
5.12.3	お知らせリンク機能	35
5.13	掲示板編集サブシステム (管理者用)	36
5.13.1	スレッド非表示化機能	36
5.13.2	レス非表示化機能	36
5.14	非表示化履歴閲覧機能	37
5.14.1	不適切な単語登録機能	38
5.14.2	不適切な単語自動置換機能	39
5.14.3	通報内容閲覧機能	41
5.15	ユーザ管理サブシステム (管理者用)	42
5.15.1	ユーザ情報機能検索機能、ユーザ情報閲覧機能機能	42
5.15.2	ユーザに対する警告通知送信機能	44
5.15.3	ユーザアカウント凍結機能	46
5.16	ユーザ登録サブシステム (管理者用)	46
5.16.1	仮 ID・仮パスワード発行機能	46
5.17	管理者ログインサブシステム (管理者用)	47
5.17.1	管理者認証機能	47
<b>6</b>	<b>ルーティング及び MVC 一覧</b>	<b>49</b>
6.1	ルーティング一覧	50
6.2	Controller	51
6.2.1	application_controller.rb	51
6.2.2	top_controller.rb	52
6.2.3	sessions_controller.rb	52
6.2.4	passwords_controller.rb	52
6.2.5	home_controller.rb	53

6.2.6	users_controller.rb . . . . .	53
6.2.7	results_controller.rb . . . . .	53
6.2.8	results_categories_controller.rb . . . . .	53
6.2.9	informations_controller.rb . . . . .	53
6.2.10	categories_controller.rb . . . . .	54
6.2.11	threads_controller.rb . . . . .	54
6.2.12	contents_controller.rb . . . . .	54
6.2.13	form_controller.rb . . . . .	55
6.2.14	bookmark.rb . . . . .	55
6.2.15	hide_controller.rb . . . . .	55
6.2.16	admins_session_controller.rb . . . . .	55
6.2.17	admins_top_controller.rb . . . . .	56
6.2.18	ng_word_controller.rb . . . . .	56
6.2.19	manage_controller.rb . . . . .	56
6.2.20	manage_user_controller.rb . . . . .	56
6.2.21	announce_controller.rb . . . . .	57
6.3	View 層 . . . . .	57
6.3.1	top/home.html.erb . . . . .	57
6.3.2	sessions/new.html.erb . . . . .	57
6.3.3	passwords/new.html.erb . . . . .	58
6.3.4	passwords/signup.html.erb . . . . .	58
6.3.5	home_pages/home.html.erb . . . . .	58
6.3.6	announce/show.html.erb . . . . .	58
6.3.7	users/show.html.erb . . . . .	58
6.3.8	results/search.html.erb . . . . .	59
6.3.9	categories/index.html.erb . . . . .	59
6.3.10	threads/new.html.erb . . . . .	60
6.3.11	threads/check.html.erb . . . . .	60
6.3.12	threads/show.html.erb . . . . .	60
6.3.13	contents/check.html.erb . . . . .	61
6.3.14	report/new.html.erb . . . . .	61
6.3.15	report/check.html.erb . . . . .	61
6.3.16	admins/login.html.erb . . . . .	61
6.3.17	admins_top/home.html.erb . . . . .	62
6.3.18	manage/home.html.erb . . . . .	62
6.3.19	manage/signup.html.erb . . . . .	62
6.3.20	manage/ok.html.erb . . . . .	63
6.3.21	manage/destroy_check.html.erb . . . . .	63
6.3.22	manage_user/home.html.erb . . . . .	63
6.3.23	manage_user/user_signup.html.erb . . . . .	64
6.3.24	manage_user/search.html.erb . . . . .	64
6.3.25	manage_user/show.html.erb . . . . .	64
6.3.26	manage_user/search.html.erb . . . . .	64
6.3.27	manage_user/warning_edit.html.erb . . . . .	65

6.3.28	manage_user/warning_check.html.erb . . . . .	65
6.3.29	announce/edit.html.erb . . . . .	65
6.3.30	announce/check.html.erb . . . . .	66
6.3.31	admins/login.html.erb . . . . .	66
6.3.32	ng_words/home.html.erb . . . . .	66
6.3.33	hide/thread_index.html.erb . . . . .	66
6.3.34	hide/res_index.html.erb . . . . .	67
6.4	Model 層 . . . . .	67
6.4.1	user.rb . . . . .	67
6.4.2	bookmark.rb . . . . .	67
6.4.3	correct_user.rb . . . . .	68
6.4.4	alert.rb . . . . .	68
6.4.5	thread.rb . . . . .	68
6.4.6	res.rb . . . . .	69
6.4.7	announcement.rb . . . . .	69
6.4.8	category.rb . . . . .	69
6.4.9	report.rb . . . . .	70
6.4.10	ng_word.rb . . . . .	70
6.4.11	freeze.rb . . . . .	70

## 1 システム概要

本システムは、学生が陥りやすいトラブルや、学生自身の学習における課題や悩みを自主的に解決するための掲示板型のウェブアプリケーションである。学生同士が問題をネット上でかつ匿名に解決できるようにするために開発するシステムが、本システムである。利用する対象は、本校の学生である。また管理者は、本校の事務員を想定している。

本システムは、メインシステムとして下記に示す「ユーザシステム」、「掲示板システム」、「管理者システム」の3つを実装しており、各システムを構成するサブシステムも併せて以下に示す。

- 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 7
- \* ubuntu version 18.04
- \* MacOS
- \* Windowss7 , 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 version 7
  - ubuntu version 18.04
  - MacOS
  - Windowss7 , 8 , 10
  - iOS 10.0
- HTML : version 5
- 使用言語
  - ruby version 2.4.2
  - Ruby on Rails version 5.1.4
  - CSS
  - JavaScript
- サーバ :
- データベース : MySQL version 8.0
- web サーバ : AmazonWebServices

### 4 内部設計書作成方針

#### 4.1 コーディング規約

##### 4.1.1 命名規約

- 変数名・メソッド名
  - 小文字始まりとする

- 基本的に意味のある単語を使用する
- 
- 定数
  - 全て大文字を使用する
- クラス名・構造体名
  - 変数名・メソッド名を同様に意味のある単語を使用する
  - 大文字始まりとする
  - 複数の単語を組み合わせる際は先頭文字を大文字で表記する

#### 4.1.2 コーディングスタイル

- インデント
  - インデントにはタブを使用する（半角スペース 4 文字）
- 括弧
  - 中括弧は改行して始める
  - 小括弧の前後にはスペースを使用しない
- 演算子
  - 演算子の前後には半角スペースを一文字使用する

## 4.2 設計書作成環境

内部設計書の作成環境は、表 1 に示します。

表 1: 内部設計書の作成環境環境

組版処理システム	LATEX,dvpdfmx
文字コード	UTF-8
改行コード	LF(0x0A)

## 4.3 サーバ環境

本システムを利用するためには Amazon Web Services(AWS) の EC2 インスタンスを用いて実現します。サーバ環境は表 2 に示します。

## 4.4 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。



表 2: サーバ側の動作環境

対応 OS	Ruby on Rails
vCPU	1
メモリ (GiB)	1
ストレージ	30GB

#### 4.4.1 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 3 に示す。

表 3: 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	varchar	20	×	-	-
パスワード	password	varchar	20	×	-	-
ハンドルネーム	write_name	varchar	15	×	-	-
保有ポイント	now_point	int	4	×	-	-
取り消しフラグ	cancel_flag	int	1	×	-	-
赤色フラグ	color_flag	int	1	×	-	-
斜体フラグ	diagonal_flag	int	1	×	-	-
太文字フラグ	bold_letters_flag	int	1	×	-	-
通知フラグ	report_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。NULL 値は含まない。値は固定 7 文字の半角英数字にて構成される。
- ユーザ ID  
ユーザを識別するための ID を示す値である。NULL 値は含まない。値は 20 文字以下の半角英数字にて構成される。
- 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。NULL 値は含まない。値は 20 文字以下の半角英数字にて構成される。
- ハンドルネーム  
ユーザがレス書き込みを行う際に表示する名前を示す値である。NULL 値は含まない。値は 15 文字以下の文字列にて構成される。
- 保有ポイント  
ユーザがログインした日数とコレクトボタンを 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。NULL 値は含まない。4 桁 (0001 から 9999) の数値にて構成される。

- 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。NULL 値は含まない。値は1桁の0(OFF)、1(ON)の数値にて構成される。
- 赤色フラグ  
拡張機能の1つであり、赤色を付与することを示す値である。NULL 値は含まない。値は1桁の0(OFF)、1(ON)の数値にて構成される。
- 斜体フラグ  
拡張機能の1つであり、斜体を付与することを示す値である。NULL 値は含まない。値は1桁の0(OFF)、1(ON)の数値にて構成される。
- 太文字フラグ  
拡張機能の1つであり、太文字を付与することを示す値である。NULL 値は含まない。値は1桁の0(OFF)、1(ON)の数値にて構成される。
- 通知フラグ  
通知設定の切り替えを示す値である。NULL 値は含まない。値は1桁の0(OFF)、1(ON)の数値にて構成される。
- 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。
- 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。NULL 値は含まない。値は1桁の0(OFF)、1(ON)の数値にて構成される。
- ログイン日数  
ユーザーがログインをした日数を記録することを示す値である。NULL 値は含まない。値は4桁(0001 から 9999)の数値にて構築される。

#### 4.4.2 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 4 に示す。

表 4: 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 ID	thread_id	int	5	×	unsigned zerofill auto_increment	PK
スレッドカテゴリ ID	thread_category_id	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	varchar	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 update current_timestamp	-

- スレッド ID  
自テーブルの主キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構成され自動追加される。
- スレッドカテゴリ ID  
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。NULL 値は含まない。値は 3 桁 (001 から 999) の数値にて構成され自動追加される。
- 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。NULL 値は含まない。固定 7 文字の半角英数字にて構成される。
- スレッドタイトル名  
スレッドタイトルを示す値である。NULL 値は含まない。値は 50 文字以下の文字列にて構成される。
- 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を示す値である。NULL 値は含まない。値は 1 桁の数値で 0(OFF)、1(ON) の数値にて構成される。  
例としてスレッド非表示フラグを 1 にした場合、ユーザ側からはスレッドが非表示になる。
- 作成日時  
レコードを作成した日付・時刻を示す値である。NULL 値は含まない。レコードが作成されるたびに自動追加を行う。
- 更新日時  
レコードを更新した日付・時刻を示す値である。NULL 値は含まない。レコードが更新されるたびに自動更新を行う。

#### 4.4.3 レステーブル (res)

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 5 に示す。

表 5: 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 ID	res_id	int	4	×	unsigned zerofill auto_increment	PK
スレッド ID	thread_id	int	5	×	unsigned zerofill auto_increment	PK(res 複合) FK:thread
学籍番号	student_id	char	7	×	-	FK:user
書き込み ID	write_id	char	8	×	-	-
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	res_hide_flag	int	1	×	-	-
作成日時	created_at	timestamp	-	×	default current_timestamp	-

- レス ID  
自テーブルの主キーである。NULL 値は含まない。値は 4 桁 (0001 から 9999) の数値にて構成され自動追加される。
- スレッド ID  
スレッドテーブルを参照する際の外部キーであり、スレ ID との複合主キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構成され自動追加される。
- 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。NULL 値は含まない。固定 7 文字の半角英数字にて構成される。
- 書き込み 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。NULL 値は含まない。値は固定 8 文字の半角英数字にて構成される。
- 書き込み内容  
書き込み内容を示す値である。NULL 値は含まない。値は 200 文字以下の文字列にて構成される。
- 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。NULL 値は含まない。値は 4 桁 (0001 から 9999) の数値にて構成される。
- レス非表示フラグ  
管理者の操作権限でレスを非表示を示す値である。NULL 値は含まない。値は 1 桁の数値で 0(OFF)、1(ON) の数値にて構成される。  
例としてレス非表示フラグを 1 にした場合、ユーザ側からはレスが非表示になる。

- 作成日時

レコードを作成した日付・時刻を示す値である。NULL 値は含まない。レコードが作成されるたびに自動追加される。

#### 4.4.4 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 6 に示す。

表 6: 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 ID	category_no	int	3	×	unsigned zerofill auto_increment	PK
カテゴリ名	category_name	varchar	15	×	-	-

- カテゴリ ID

自テーブルの主キーである。NULL 値は含まない。値は 3 桁 (001 から 999) の数値にて構築され自動追加される。

- カテゴリ名

カテゴリの名前を示す値である。NULL 値は含まない。値は 15 文字以下の文字列にて構成される。

#### 4.4.5 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 7 に示す。

表 7: 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 ID	thread_id	int	5	×	unsigned zerofill auto_increment	FK:thread

- 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。NULL 値は含まない。固定 7 文字の半角英数字にて構成される。

- スレッド ID

スレッドテーブルを参照する際の外部キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構築され自動追加される。

#### 4.4.6 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 8 に示す。

表 8: 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 ID	thread_id	int	5	×	unsigned zerofill auto_increment	FK:thread
レス ID	res_no	int	4	×	unsigned zerofill auto_increment	FK:res

- 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。NULL 値は含まない。固定 7 文字の半角英数字にて構成される。
- スレッド ID  
スレッドテーブルを参照する際の外部キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構成され自動追加される。
- レス ID  
レステーブルを参照する際の外部キーである。NULL 値は含まない。値は 4 桁 (0001 から 9999) の数値にて構成され自動追加される。

#### 4.4.7 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 9 に示す。

表 9: 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 ID	word_id	int	5	×	unsigned zerofill auto_increment	PK
単語名	word_name	varchar	200	×	-	-

- 単語 ID  
自テーブルの主キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構成され自動追加される。
- 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。NULL 値は含まない。値は 200 文字以下の文字列にて構成される。

#### 4.4.8 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 10 に示す。

表 10: 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 ID	report_id	int	5	×	unsigned zerofill auto_increment	PK
通報ユーザ ID	report_user_id	varchar	20	×	-	FK:user
スレッド ID	thread_id	int	5	×	unsigned zerofill auto_increment	FK:thread
レス ID	res_id	int	4	×	unsigned zerofill auto_increment	FK:res
通報内容	report_content	varchar	200	×	-	-
通報日時	report_at	timestamp	-	×	default current_timestamp	-

- 通報 ID  
自テーブルの主キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構成され自動追加される。
- 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。NULL 値は含まない。値は 20 文字以下の半角英数字にて構成される。
- スレッド ID  
スレッドテーブルを参照する際の外部キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構成され自動追加される。
- レス ID  
レステーブルを参照する際の外部キーである。NULL 値は含まない。値は 4 桁 (0001 から 9999) の数値にて構成され自動追加される。
- 通報内容  
通報内容を示す値である。NULL 値は含まない。値は 200 文字以下の文字列にて構成される。
- 通報日時  
レコードを作成した日付・時刻を示す値である。NULL 値は含まない。レコードが作成されるたびに自動追加を行う。

#### 4.4.9 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を凍結した情報を管理する。このテーブルの詳細は表 11 に示す。

表 11: 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 ID	suspend_id	int	4	×	unsigned zerofill auto_increment	PK
学籍番号	student_id	char	7	×	-	FK:user

- 凍結 ID

自テーブルの主キーである。NULL 値は含まない。値は 4 桁 (0001 から 9999) の数値にて構成され自動追加される。

- 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。。NULL 値は含まない。値は固定 7 文字の半角英数字にて構成される。

#### 4.4.10 お知らせテーブル (news)

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 12 に示す。

表 12: 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 ID	news_id	int	5	×	unsigned zerofill auto_increment	PK
お知らせタイトル名	news_title	varchar	50	×	-	-
お知らせ内容	news_title	varchar	400	×	-	-
作成日時	created_at	timestamp	-	×	default current_timestamp	-

- お知らせ ID

自テーブルの主キーである。NULL 値は含まない。値は 5 桁 (00001 から 99999) の数値にて構成され自動追加される。

- お知らせタイトル名

お知らせタイトルを示す値である。NULL 値は含まない。値は 50 文字以下の文字列にて構成される。

- お知らせ内容

お知らせ内容を示す値である。NULL 値は含まない。値は 400 文字以下の文字列にて構成される。

- 作成日時

レコードを作成した日付・時刻を示す値である。NULL 値は含まない。レコードが作成されるたびに自動追加を行う。

#### 4.4.11 警告注意テーブル (warned\_caution)

警告テーブルでは管理者がユーザに対しての警告・注意喚起に関する情報を管理する。このテーブルの詳細は表 13 に示す。



表 13: 警告注意テーブル

論理名	物理名	データ型	精度	NULL	オプション	PK/FK:mode
警告注意 ID	warned_caution_id	int	4	×	unsigned zerofill auto_increment	PK
学籍番号	student_id	char	7	×	-	FK:user
警告注意タイトル名	warned_caution_title	varchar	50	×	-	-
警告注意内容	warned_caution_title	varchar	400	×	-	-
作成日時	created_at	timestamp	-	×	default current_timestamp	-

- 警告注意 ID  
自テーブルの主キーである。NULL 値は含まない。値は 4 桁 (0001 から 9999) の数値にて構成され自動追加される。
- 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。NULL 値は含まない。値は固定 7 文字の半角英数字にて構成される。
- 警告注意タイトル名  
警告注意タイトルを示す値である。NULL 値は含まない。値は 50 文字以下の文字列にて構成される。
- 警告注意内容  
警告注意内容を示す値である。NULL 値は含まない。値は 400 文字以下の文字列にて構成される。
- 作成日時  
レコードを作成した日付・時刻を示す値である。NULL 値は含まない。レコードが作成されるたびに自動追加を行う。

## 5 サブシステムとシーケンス図

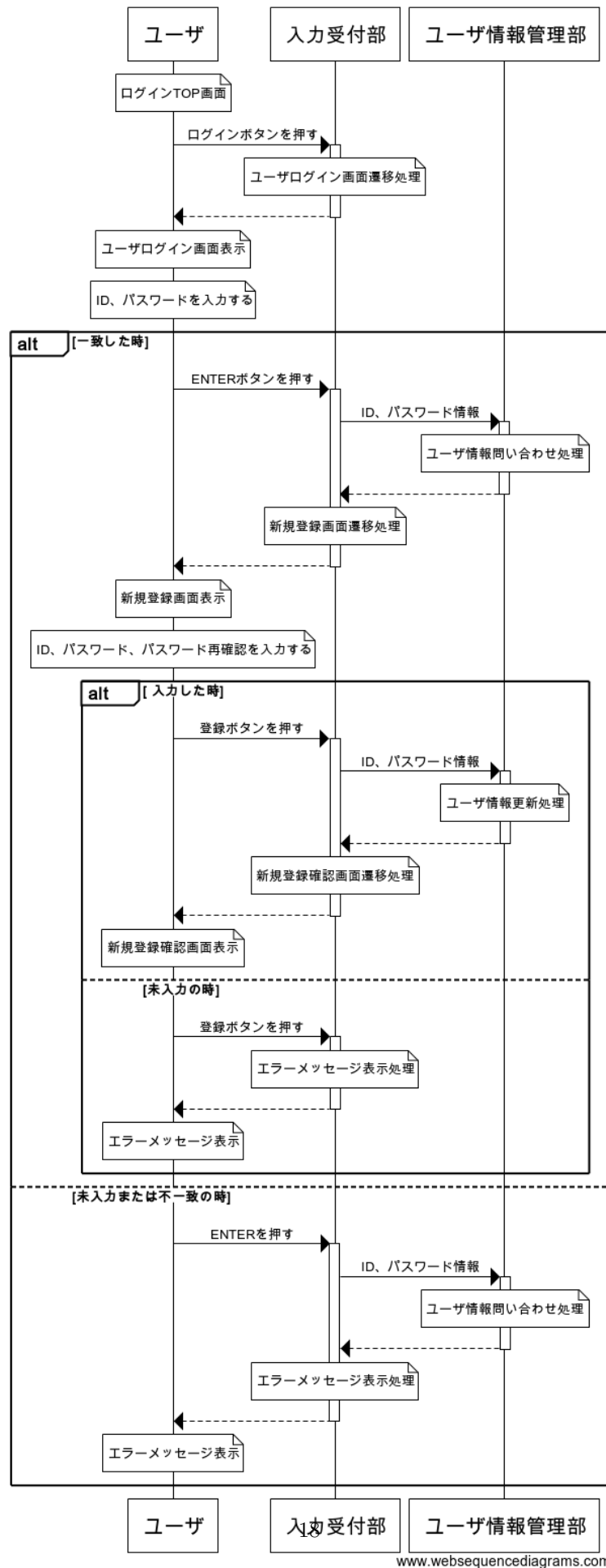
本システムが用いているサブシステムの構成を示す。また各サブシステムごとの機能についてもシーケンス図を用いて示す。

### 5.1 アカウント登録サブシステム (ユーザ用)

アカウント登録サブシステムでは仮 ID と仮パスワードからユーザ専用の ID とパスワードへ変更し、アカウント登録を行う。アカウント登録サブシステムの機能は小節で示す。

#### 5.1.1 新規登録機能

新規登録機能では新規登録時、ID・パスワード入力画面に遷移する。仮 ID と仮パスワードを入力することで、ID・パスワード変更画面に遷移することができる。その後、ユーザは各自で ID とパスワードを変更し、登録を行うことができる。なお、ID とパスワードを登録しなければ、本システムは利用することができないようになっている。新規登録機能のシーケンス図は図 1 に示す。



www.websequencediagrams.com

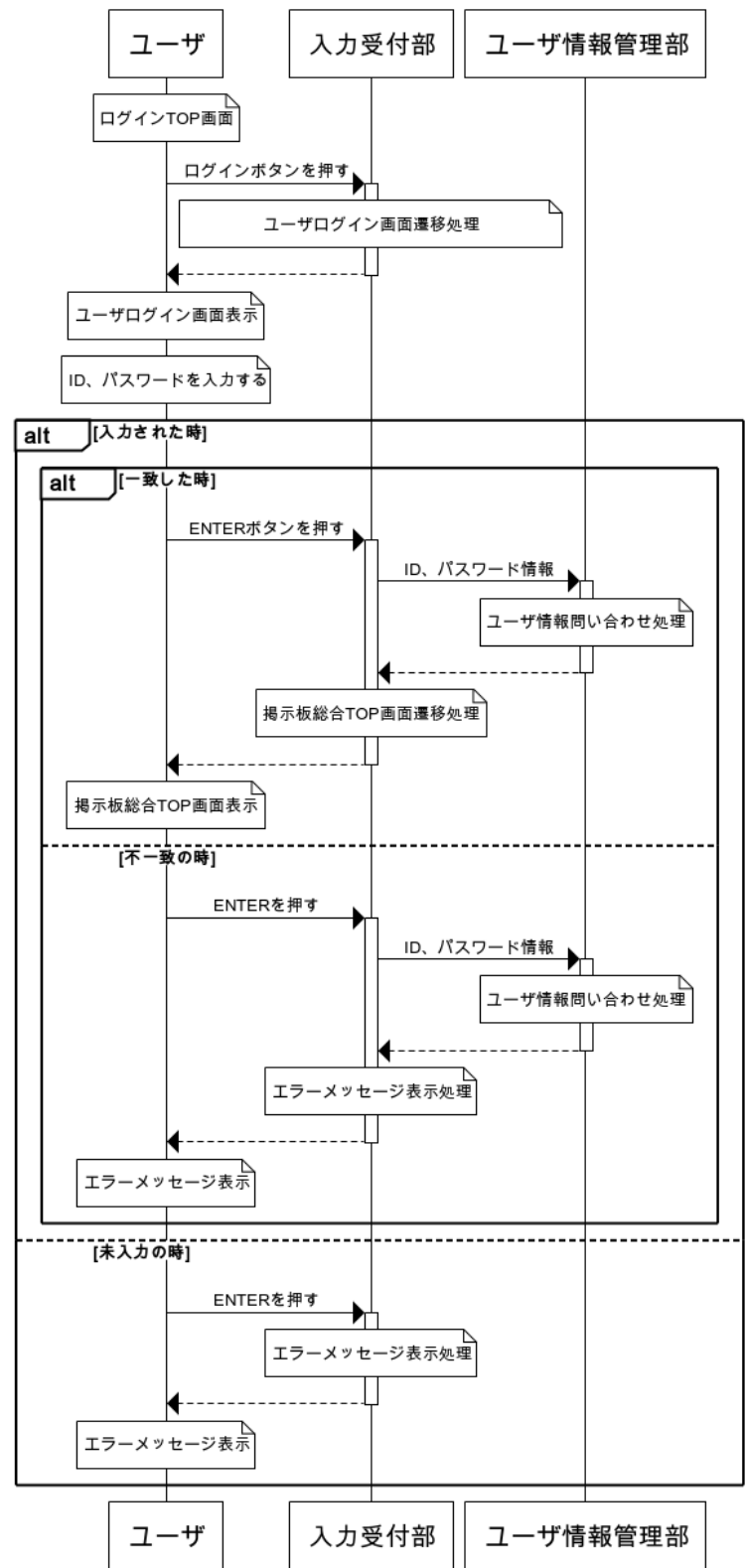
図 1: 新規登録機能のシーケンス図

## 5.2 ログインサブシステム (ユーザ用)

ログインサブシステムではユーザが設定した ID とパスワードを入力を行う。ログイン時にデータベース登録されている ID とパスワードの認証を行い一致すれば自分のアカウントにログインをすることができる。ログインサブシステムの機能は小節で示す。

### 5.2.1 ユーザ認証機能

ユーザ認証機能では本システムを利用するユーザが、本学の在学生であるかの認証を、ID とパスワードを用いて行う。認証に成功したユーザは、本システムの正当なユーザとして本システムを利用することができる。ユーザ認証機能のシーケンス図は図 2 に示す。



www.websequencediagrams.com

図 2: ユーザ認証機能のシーケンス図

### 5.2.2 ログインボーナス機能

ログインボーナス機能ではユーザが本システムにログインしたとき、拡張機能と交換できるポイントが付与される。ポイントが付与される頻度は、1日1回に設定している。1回で付与されるポイント数は10ポイントである。ログインボーナス機能のシーケンス図は図3に示す。

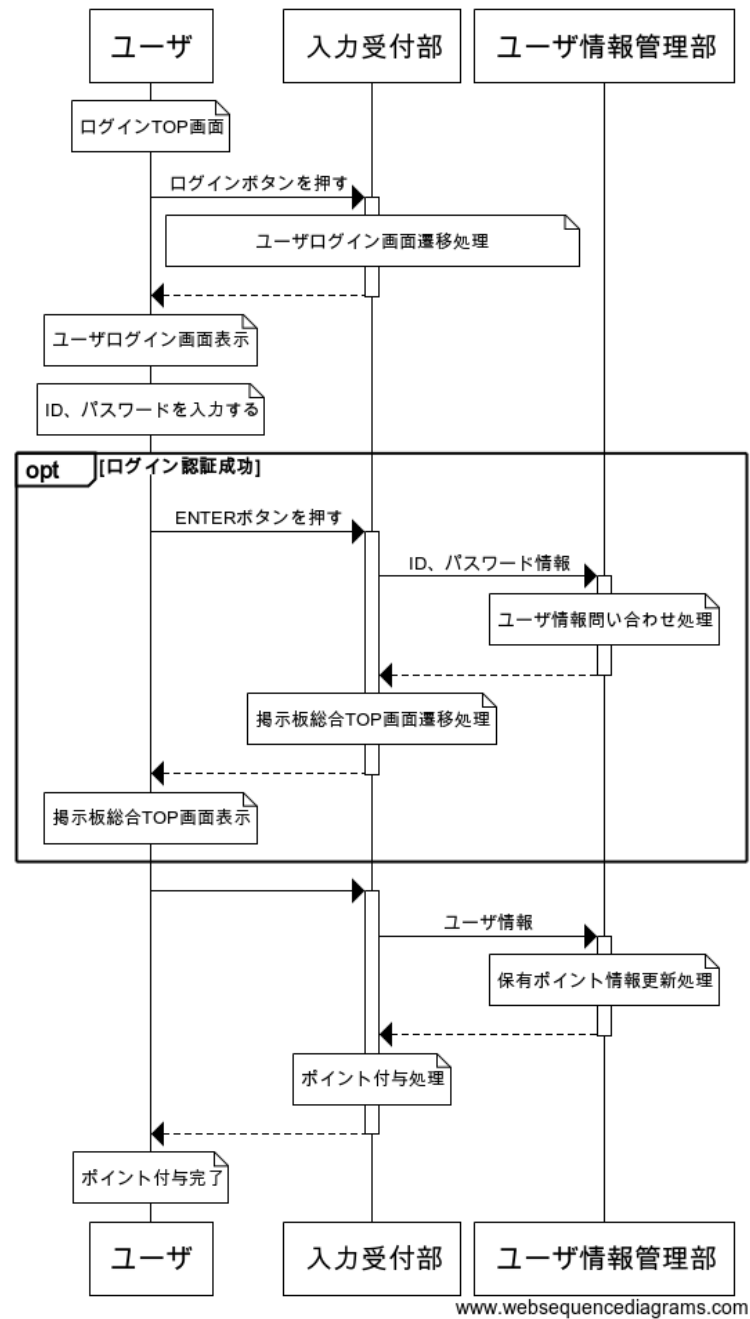


図 3: ログインボーナス機能のシーケンス図

### 5.2.3 ログアウト機能

ログアウト機能ではユーザが本システムにログインしている状態であるとき、「ログアウト」ボタンを押すことで本システムからログアウトすることができる。ログアウト機能のシーケンス図は図4に示す。

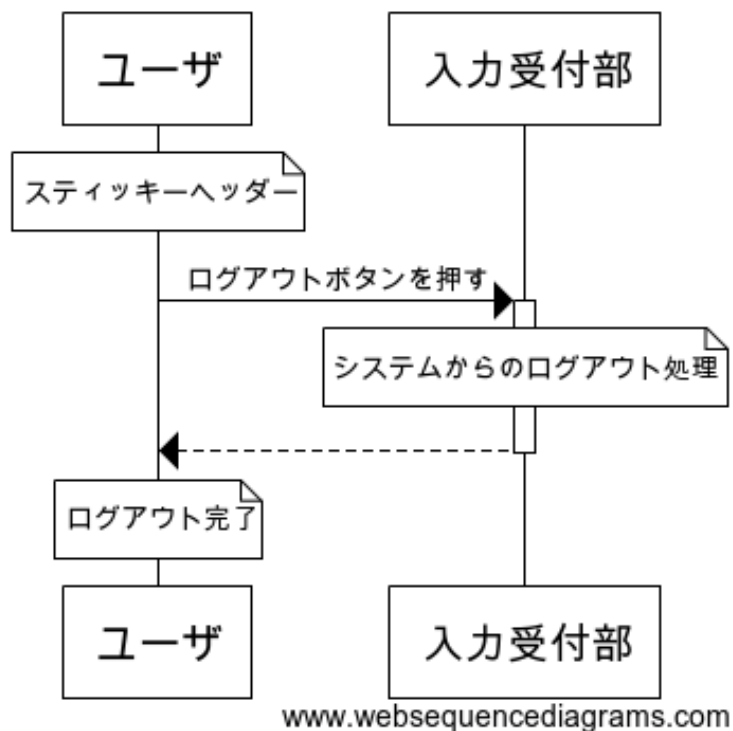


図 4: ログアウト機能のシーケンス図

## 5.3 検索サブシステム (ユーザ用)

検索サブシステムではカテゴリ検索・キーワード検索においてカテゴリを選択、もしくはテキストを入力されたキーワードに該当するスレッドの一覧を表示することができる。

検索サブシステムの機能は小節で示す。

### 5.3.1 検索機能

検索機能では所定の検索テキストボックスに任意の文字列を入力し、その文字列に関連したスレッドを一覧で表示することができる。検索方法は and 検索、or 検索が使用可能である。検索機能のシーケンス図は図5に示す。

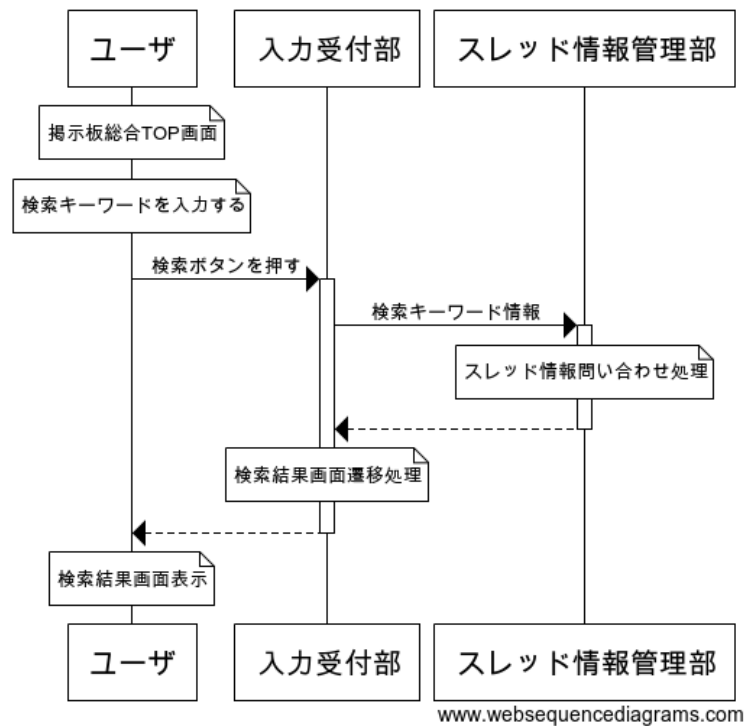


図 5: 検索機能のシーケンス図

### 5.3.2 カテゴリ選択機能

カテゴリ選択機能では所定の検索テキストボックスの横に設置されたカテゴリ選択ボタンで特定のカテゴリを選択すると、選択されたカテゴリ内に存在するスレッドの中から検索することができる。また、ステイキーヘッダーからカテゴリ検索を行うこともできる。カテゴリ選択機能のシーケンス図は図に 6 示す。

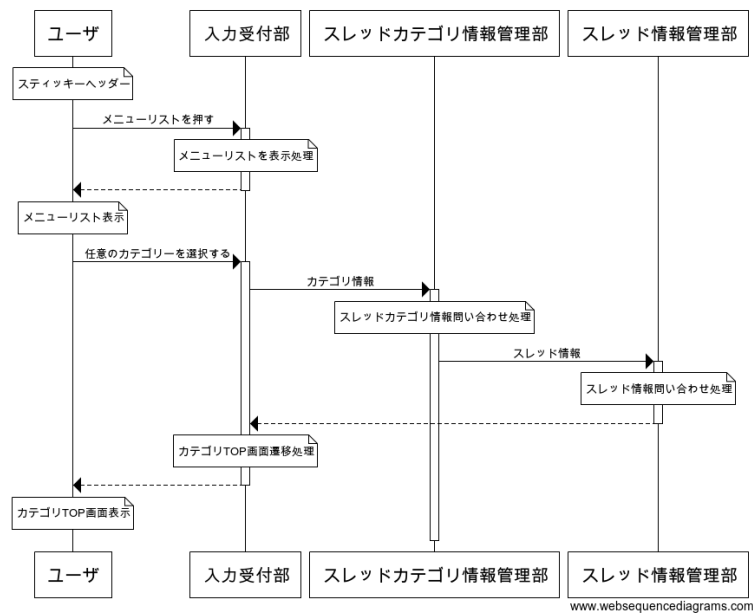


図 6: カテゴリ選択機能のシーケンス図

## 5.4 掲示板サブシステム (ユーザ用)

掲示板サブシステムではユーザがスレッドの作成・レスの書き込みを行うことができる。掲示板サブシステムの機能は小節で示す。

### 5.4.1 スレッド作成機能

スレッド作成機能ではユーザがスレッドタイトルと最初のレスを入力して、新規のスレッドを作成することができる。スレッド作成機能のシーケンス図は図 7 に示す。



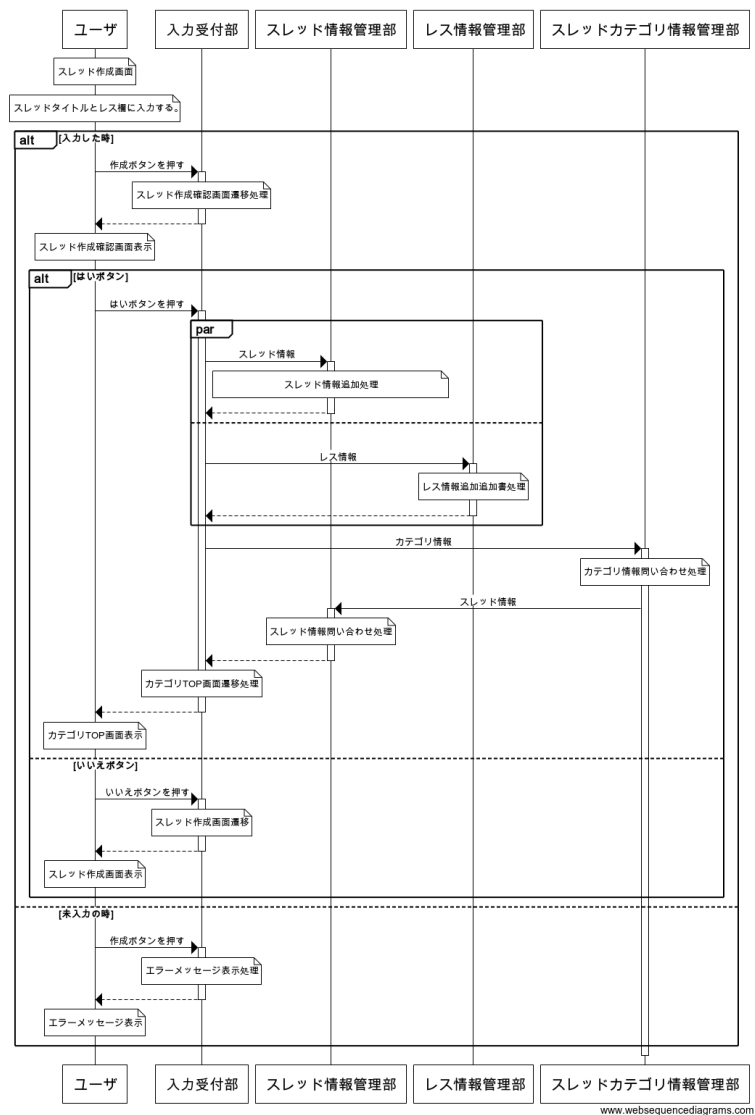
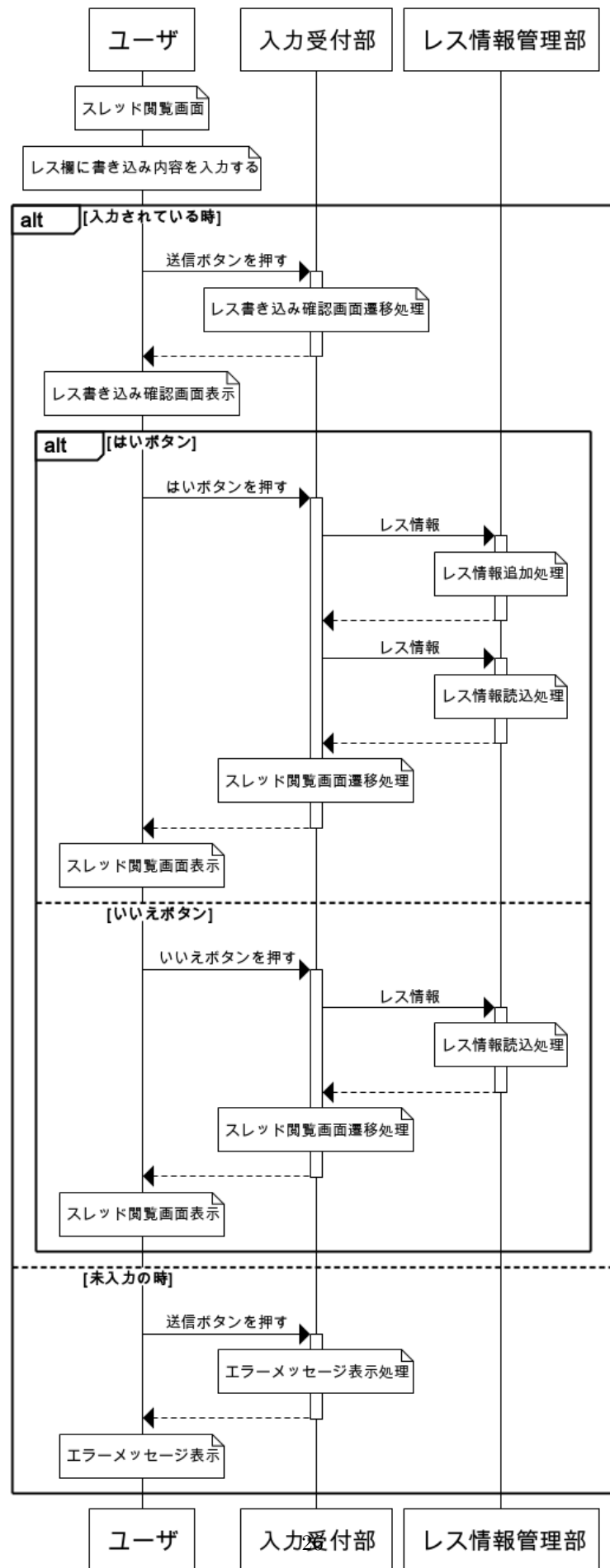


図 7: スレッド作成機能のシーケンス図

#### 5.4.2 レス書き込み機能

レス書き込み機能では既存のスレッド内に、レスを書き込むことができる。このとき、レス番号・ハンドルネーム・日付・レス ID が表示される。レスの書き込み主がハンドルネームを設定している場合、そのハンドルネームが表示される。レス書き込み機能のシーケンス図は図に 8 示す。



www.websequencediagrams.com

図 8: レス書き込み機能のシーケンス図

## 5.5 お知らせサブシステム (ユーザ用)

管理者からユーザに向けて通知されたお知らせ情報を閲覧することができる。  
お知らせサブシステムの機能は小節で示す。

### 5.5.1 お知らせ閲覧機能

お知らせ閲覧機能では管理者から通達されたお知らせを閲覧することができる。お知らせ閲覧機能のシーケンス図は図に 9 示す。

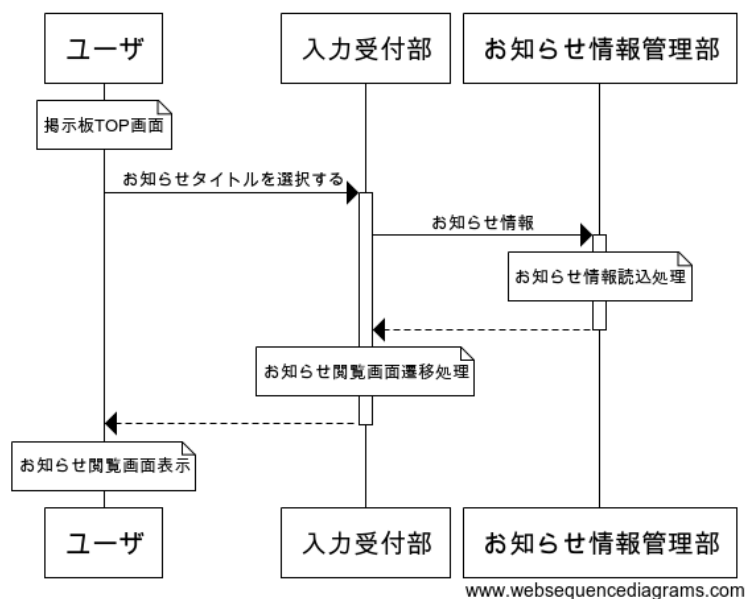


図 9: お知らせ閲覧機能のシーケンス図

## 5.6 通報サブシステム (ユーザ用)

通報サブシステムでは、誹謗中傷や公序良俗に違反してると考えれるスレッドやレスを管理者に通知することができます。

通報サブシステムの機能は小節で示す。

### 5.6.1 スレッド・レス通報機能

スレッド・レス通報機能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することができる。このとき、通報者は通報理由を明記する。スレッド・レス通報機能のシーケンス図は図 ( ) に示す。

## 5.7 コレクトボタンサブシステム (ユーザ用)

コレクトボタンサブシステムでは、ユーザがレスの内容に対して確からしいと判断した際にコレクトボタンを押すことで、コレクト認定 (信頼性の認定) を行うことができる。  
コレクトボタンサブシステムの機能は小節で示す。

### 5.7.1 コレクト認定機能

コレクト認定機能ではユーザが特定のレスに対して、その内容が確からしいと判断した場合、コレクトボタンを押してコレクト認定を行うことができる。レスの傍らにコレクト認定された回数が表示されるため、コレクト認定は、ユーザがそのレスの内容が確からしいと判断した指標となる。よって、コレクト認定された回数が多いレスは、信憑性のある情報として、ユーザの情報の取捨選択の判断材料となる。なお、1つのレスに対してコレクト認定できる回数は1回のみであり、コレクトボタンを偶数回押すと、コレクト認定を解除することができる。コレクト認定機能のシーケンス図は図 () に示す。

### 5.7.2 ポイント獲得機能

ポイント獲得機能ではユーザが書き込んだレスに他のユーザからコレクト認定された場合、レスの書き込み主は拡張機能を解放するためのポイントを獲得することができる。拡張機能とは、スレッドでレスを行う際にレスの表示が変化する機能である。ポイント獲得の相場は、1回のコレクト認定につき1ポイントである。1つのレスにおいて、1人のユーザからポイント獲得できる回数は1回のみである (2回目以降のコレクト認定はポイント獲得できない)。また、コレクト認定が解除されたとしても獲得済みのポイントが減ることはない。ポイント獲得機能のシーケンス図は図 () に示す。

## 5.8 マイページサブシステム (ユーザ用)

マイページサブシステムでは、スティッキーヘッダによって表示されているマーク部分を押すことでユーザ情報・ブックマーク閲覧・拡張機能・通知機能を設定することができる。  
マイページサブシステムの機能は小節で示す。

### 5.8.1 ユーザ情報設定機能

ユーザ情報設定機能ではユーザのハンドルネームの設定ができる。ハンドルネームが設定されていない場合は、「名無し」と表示される。ユーザ情報設定機能のシーケンス図は図 () に示す。

### 5.8.2 ブックマーク閲覧機能

ブックマーク閲覧機能では登録したブックマークの一覧を表示することができる。その中から、任意のスレッドに遷移することができる。ブックマーク閲覧機能のシーケンス図は図 () に示す。

### 5.8.3 拡張機能

拡張機能では本システムを利用して得られたポイントと引き換えに、拡張機能が使用可能となる。主な拡張機能としては、取り消し線、太文字、赤字、斜体などが存在する。1つの拡張機能につき、600ポイントを引き換えると想定する。拡張機能は、今後の仕様変更で追加する可能性がある。拡張機能のシーケンス図は図 () に示す。

### 5.8.4 通知設定機能

通知設定機能では通知機能の ON/OFF の設定をすることができる。通知設定機能のシーケンス図は図 () に示す。

## 5.9 通知サブシステム (ユーザ用)

通知サブシステムでは、各ユーザに対してスレッドの書き込み・レスへの返信があった際に処理を行う。通知サブシステムの機能は小節で示す。

### 5.9.1 通知機能

通知機能ではユーザが作成したスレッドに新たな書き込みがあった場合、またはユーザが書き込んだレスに対して返信が書き込まれた場合に、その旨を通知する。通知機能のシーケンス図は図 () に示す。

## 5.10 ブックマークサブシステム (ユーザ用)

任意のスレッドをブックマークとして登録する際と解除する際に押すことで処理が行われる。ブックマークサブシステムの機能は小節で示す。

### 5.10.1 ブックマーク登録機能

ブックマーク登録機能では任意のスレッドをブックマークとしてマイページに登録することができる。ブックマーク登録機能のシーケンス図は図 () に示す。

### 5.10.2 ブックマーク解除機能

ブックマーク解除機能ではマイページに存在する任意のブックマークの登録を解除することができる。ブックマーク解除機能のシーケンス図は図 () に示す。

## 5.11 子管理者管理サブシステム (管理者用)

子管理者サブシステムでは親管理者が子管理者のアカウントを管理するサブシステムである。親管理者がIDとパスワードを発行することで、本システムを管理する子管理者を登録することができる。また、不要になった子管理者のアカウントを抹消することができる。

子管理者管理サブシステムの機能は小節で示す。

#### 5.11.1 子管理者用 ID・パスワード発行機能

子管理者用 ID・パスワード発行機能では親管理者は手動で子管理者の ID とパスワードを登録し、発行することができる。子管理者にはこの機能は存在しない。子管理者用 ID・パスワード発行機能のシーケンス図は図 10 に示す。

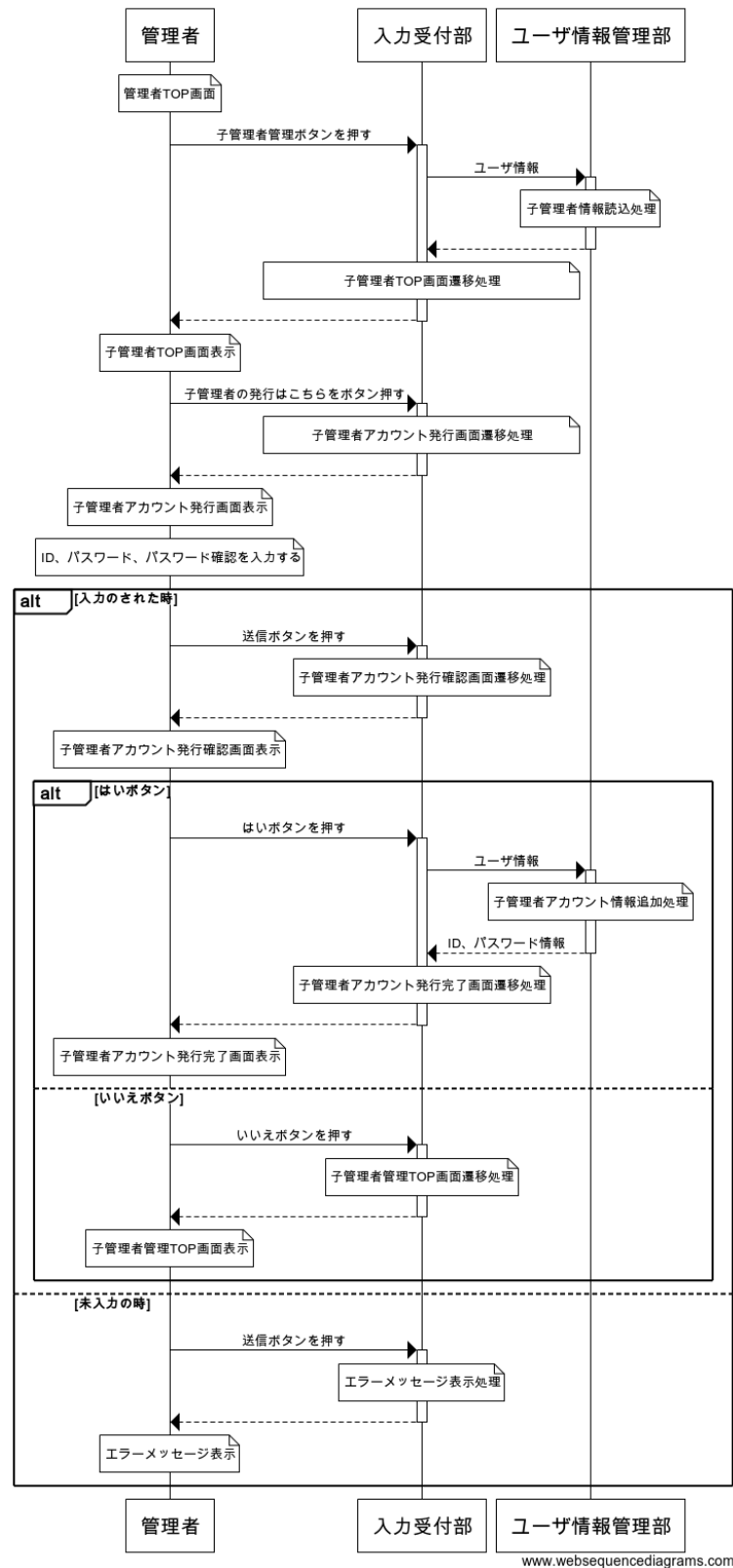


図 10: 子管理者用 ID・パスワード発行機能のシーケンス図

### 5.11.2 子管理者アカウント抹消機能

子管理者アカウント抹消機能では親管理者は手動で不要になった子管理者のアカウントを抹消することができる。子管理者アカウント抹消機能のシーケンス図は図 11 に示す。

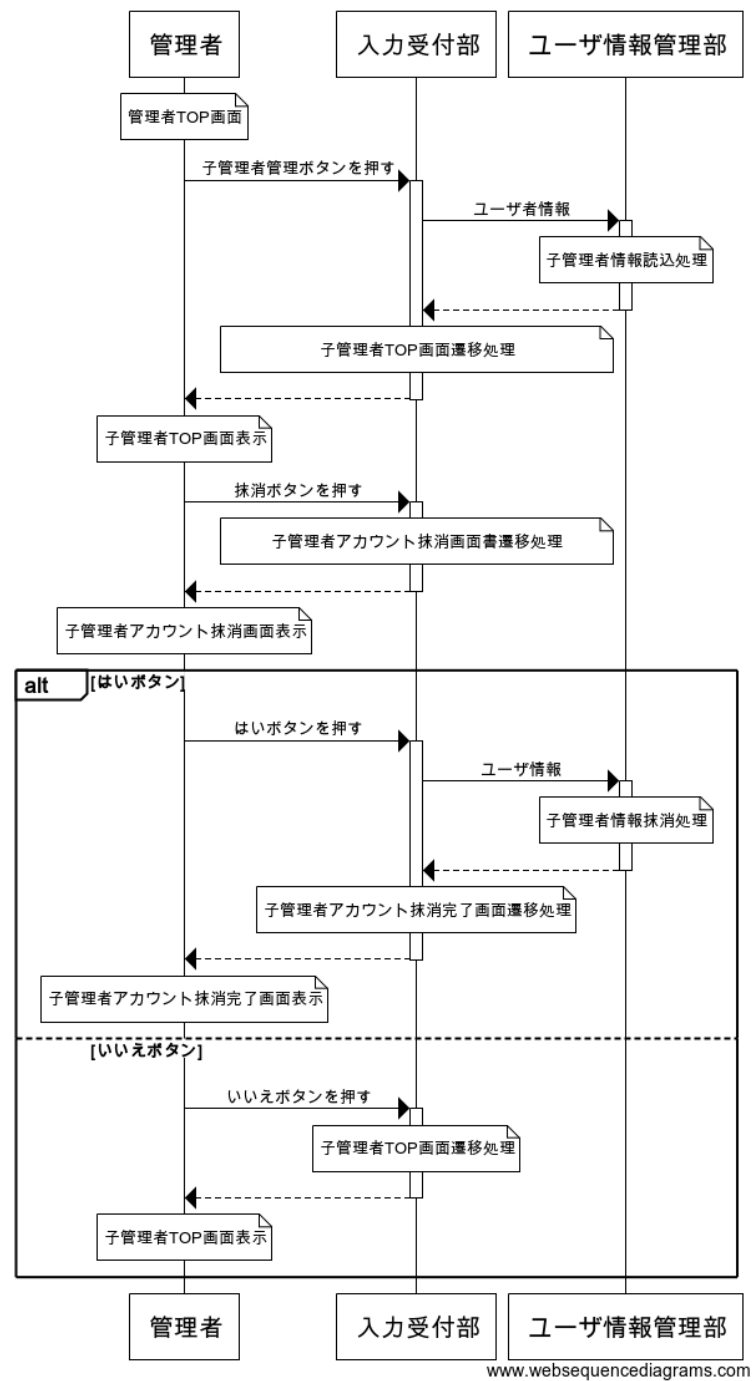


図 11: 子管理者アカウント抹消機能のシーケンス図



### 5.11.3 子管理者情報閲覧機能

子管理者情報閲覧機能では親管理者は子管理者の ID とパスワードを閲覧することができる。子管理者情報閲覧機能のシーケンス図は図 12 に示す。

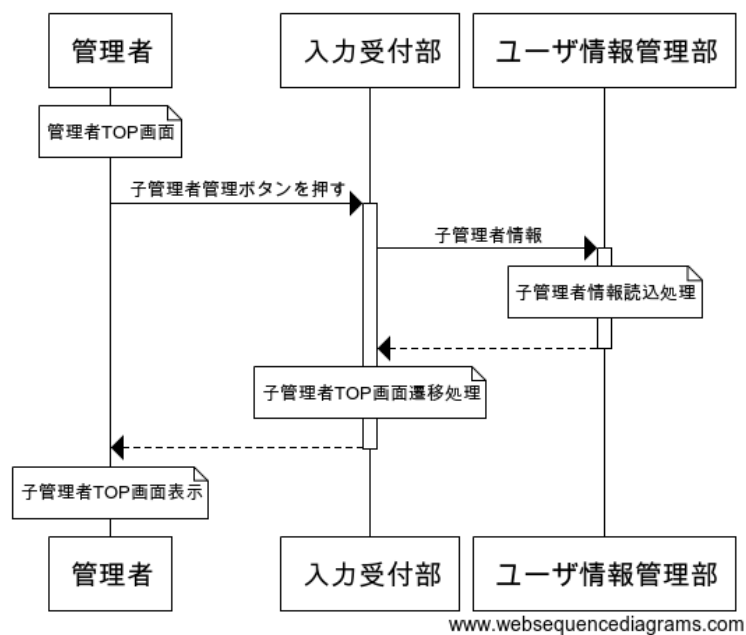


図 12: 子管理者情報閲覧機能のシーケンス図

## 5.12 お知らせ編集サブシステム (管理者用)

お知らせ編集サブシステムでは管理者からユーザ全体に対して任意のお知らせを傳達することができる。お知らせ編集サブシステムの機能は小節で示す。

### 5.12.1 お知らせ編集機能

お知らせ編集機能では管理者は、ユーザに対して通達するお知らせタイトルとその内容を編集することができる。お知らせ編集機能のシーケンス図は図 13 に示す。

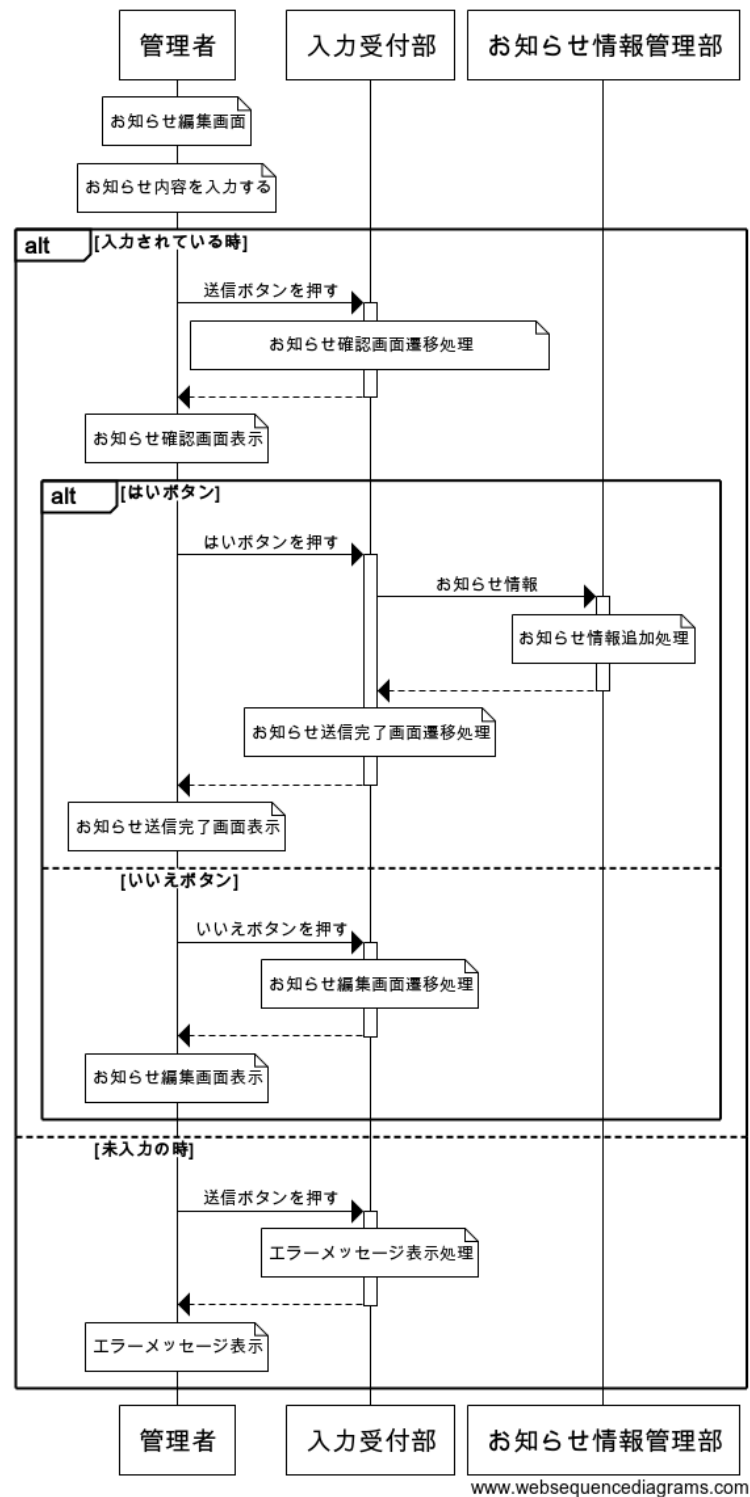


図 13: お知らせ編集機能のシーケンス図

### 5.12.2 お知らせ表示機能

お知らせ表示機能本システムのトップページに、お知らせ編集機能で編集した日付とタイトルを表示することができる。お知らせ表示機能のシーケンス図は図 14 に示す。

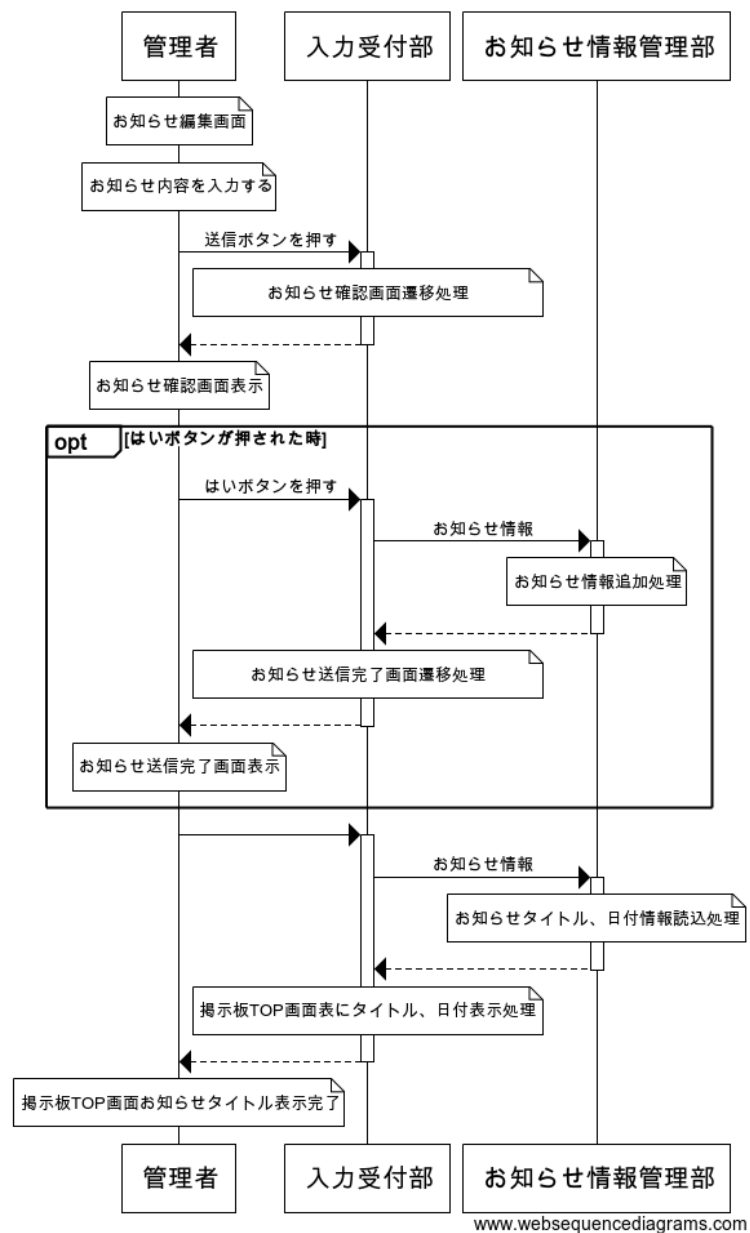


図 14: お知らせ表示機能のシーケンス図

### 5.12.3 お知らせリンク機能

お知らせ編集機能で編集したお知らせ内容を本システムトップページに表示されているお知らせタイトルにリンクを追加することができる。

## 5.13 掲示板編集サブシステム (管理者用)

掲示板編集サブシステムでは管理者は不適切であると判断したスレッドやレスを非表示することができる。また、不適切な単語を登録することで自動置換する。ユーザから送られてきたスレッドやレスに対しての通報内容を閲覧することもできる。掲示板編集サブシステムの機能は小節で示す。

### 5.13.1 スレッド非表示化機能

スレッド非表示化機能は管理者が、データベースに存在するスレッド及びスレッドに格納された全てのレスを非表示化することができる。スレッド非表示化機能のシーケンス図は図 15 に示す。

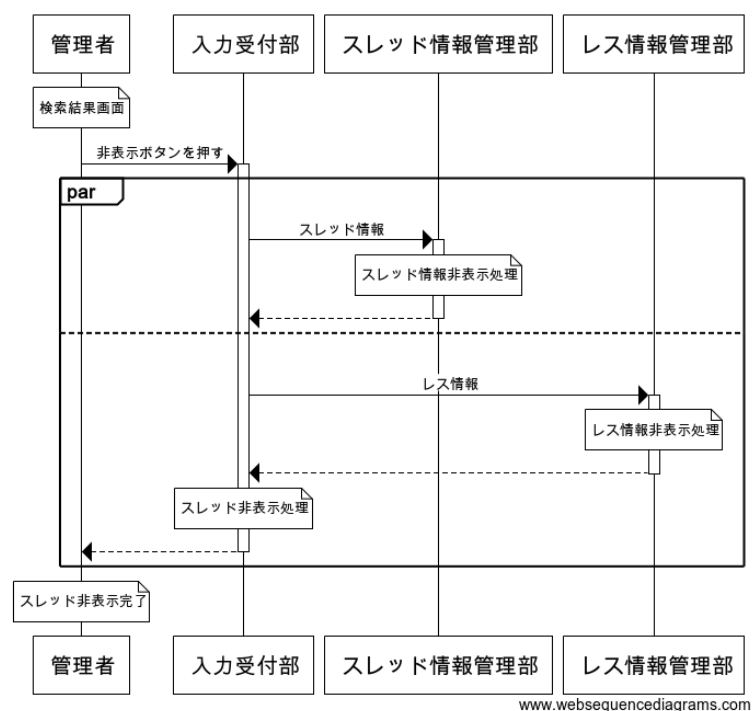


図 15: スレッド非表示化機能のシーケンス図

### 5.13.2 レス非表示化機能

レス非表示化機能では管理者が、データベースに存在するスレッド内のレスを非表示化することができる。レス非表示化機能のシーケンス図は図 16 に示す。

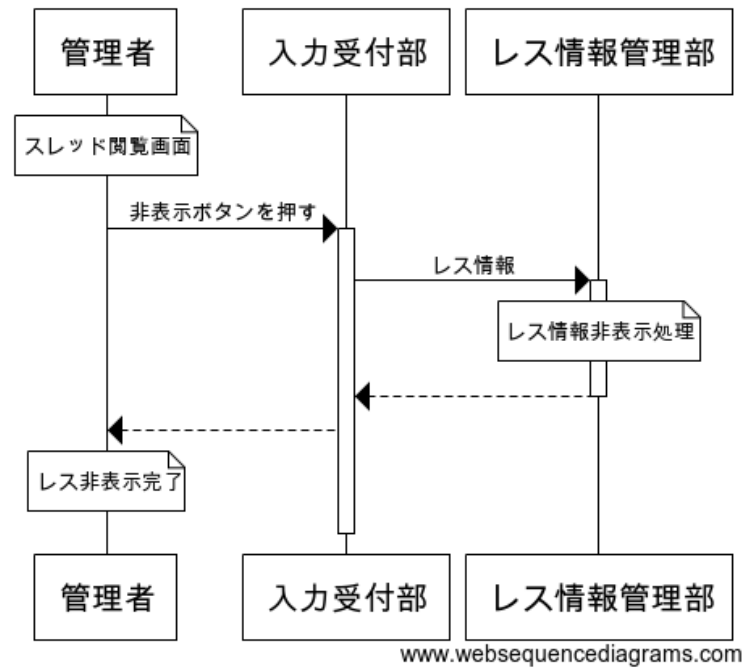


図 16: レス非表示化機能のシーケンス図

#### 5.14 非表示化履歴閲覧機能

非表示化履歴閲覧機能では管理者は、非表示化したスレッドまたはレスを一覧で閲覧することができる。非表示化履歴閲覧機能のシーケンス図は図 17 に示す。

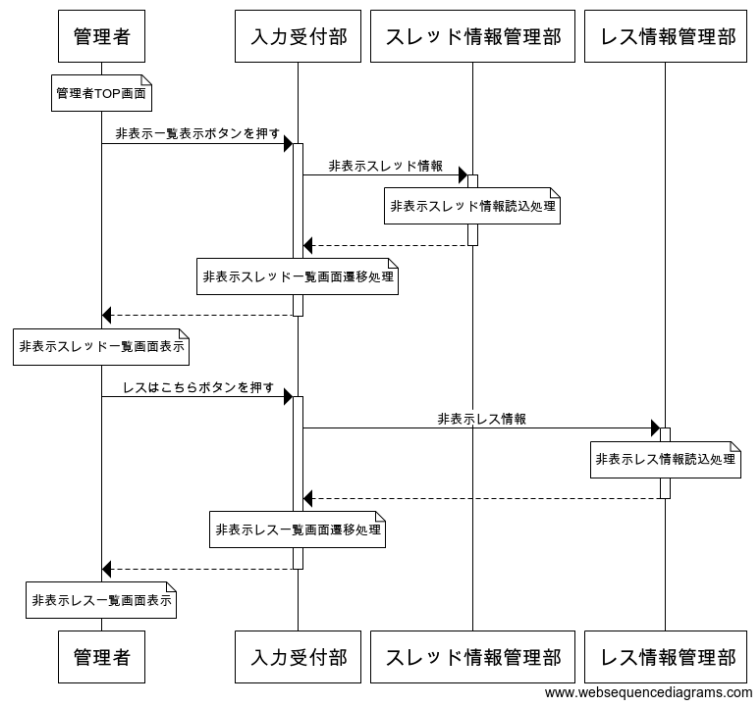


図 17: 非表示化履歴閲覧機能のシーケンス図

#### 5.14.1 不適切な単語登録機能

不適切な単語登録機能では管理者が、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することができる。不適切な単語登録機能のシーケンス図は図 18 に示す。

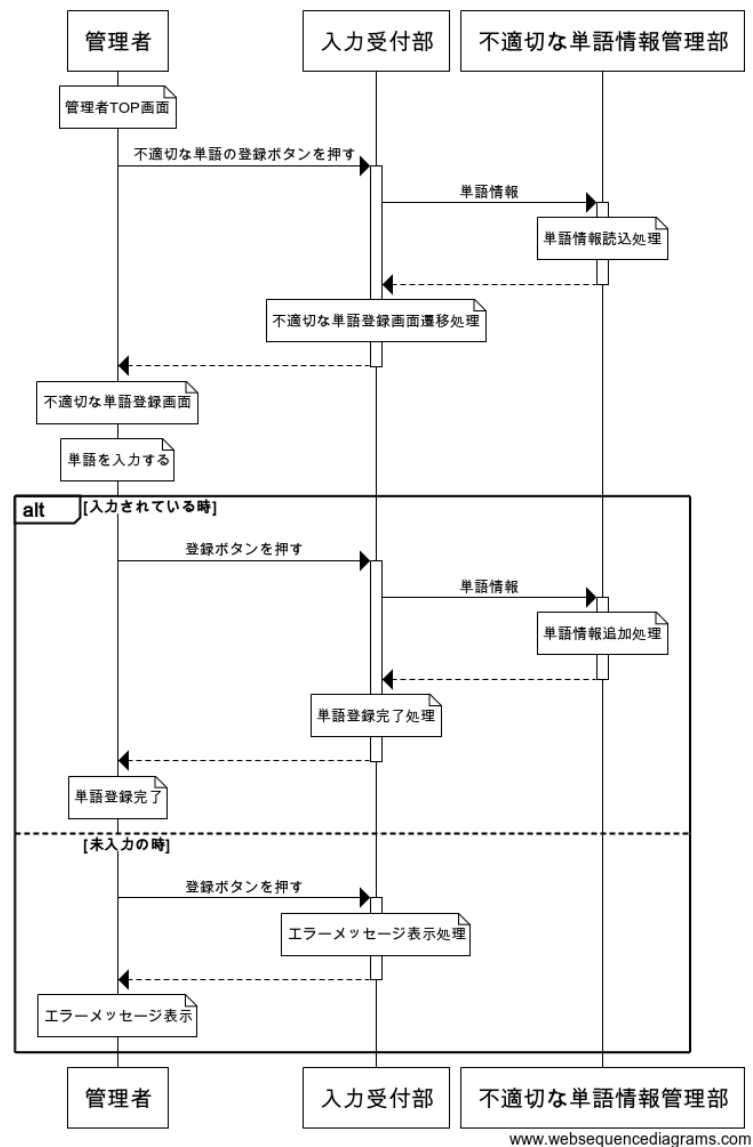


図 18: 不適切な単語登録機能のシーケンス図

#### 5.14.2 不適切な単語自動置換機能

不適切な単語自動置換機能では管理者が設定した不適切な単語を検出し次第、自動で伏せ字に置換することができる。不適切な単語自動置換機能のシーケンス図は、図 19 に不適切な単語登録機能で単語登録された時に不適切な単語を検出した場合について示し、図 20 にユーザがレス書き込みを行った場合に不適切な単語を含んでいた場合について示す。

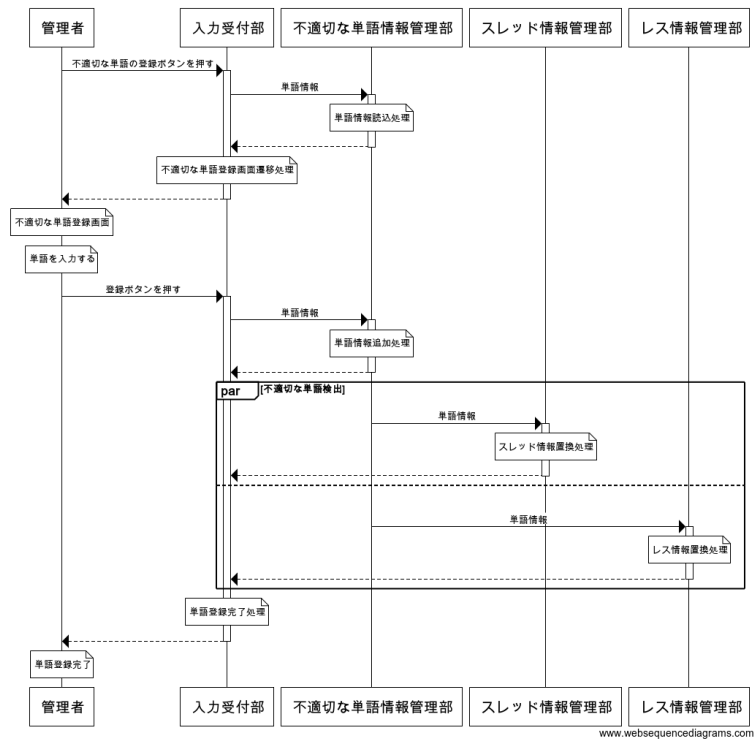


図 19: 不適切な単語自動置換機能のシーケンス図 (不適切な単語を検出した時)



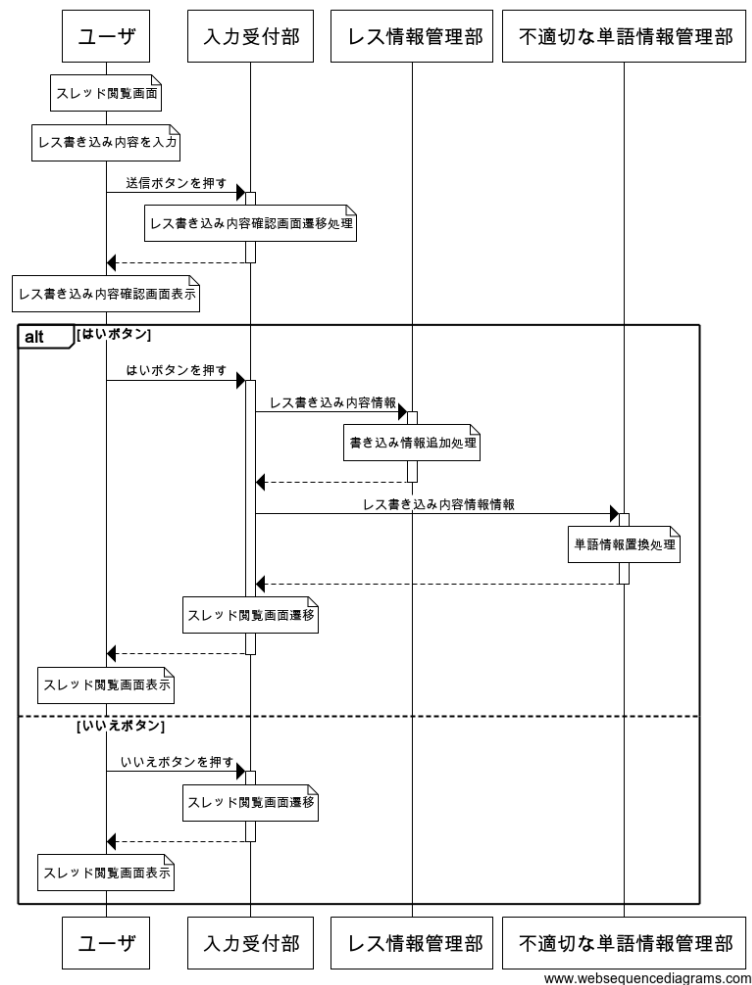


図 20: 不適切な単語自動置換機能のシーケンス図 (レス書き込みの時)

### 5.14.3 通報内容閲覧機能

通報内容閲覧機能ユーザによって通報されたスレッドまたはレスの内容の一覧を、通報理由と共に閲覧することができる。このとき、通報者のユーザ情報、通報されたスレッド主またはレスの書き込み主のユーザ情報も同時に表示される。通報内容閲覧機能のシーケンス図は図 21 に示す。

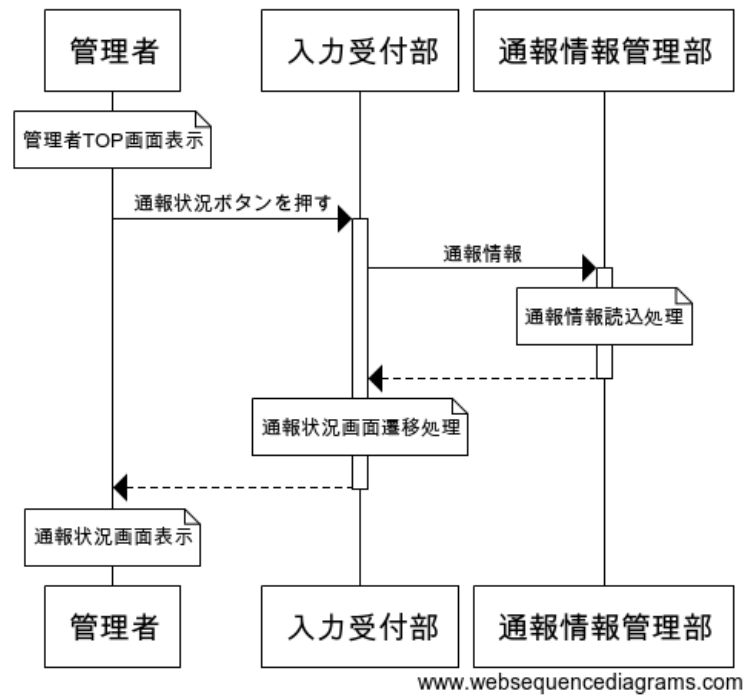


図 21: 通報内容閲覧機能のシーケンス図

## 5.15 ユーザ管理サブシステム (管理者用)

管理者はユーザの情報を検索し、閲覧することができる。また、不適切なスレッドやレスを書き込んだユーザに対して警告やアカウント凍結などの処置を行うことができる。  
ユーザ管理サブシステムの機能は小節で示す。

### 5.15.1 ユーザ情報機能検索機能、ユーザ情報閲覧機能機能

ユーザ情報機能検索機能とユーザ情報閲覧機能機能では管理者が、データベースに登録されているユーザ情報の中から特定のユーザ情報を検索しユーザの情報を閲覧することができる。ユーザ情報検索機能とユーザ情報閲覧機能のシーケンス図は図 22 に示す。

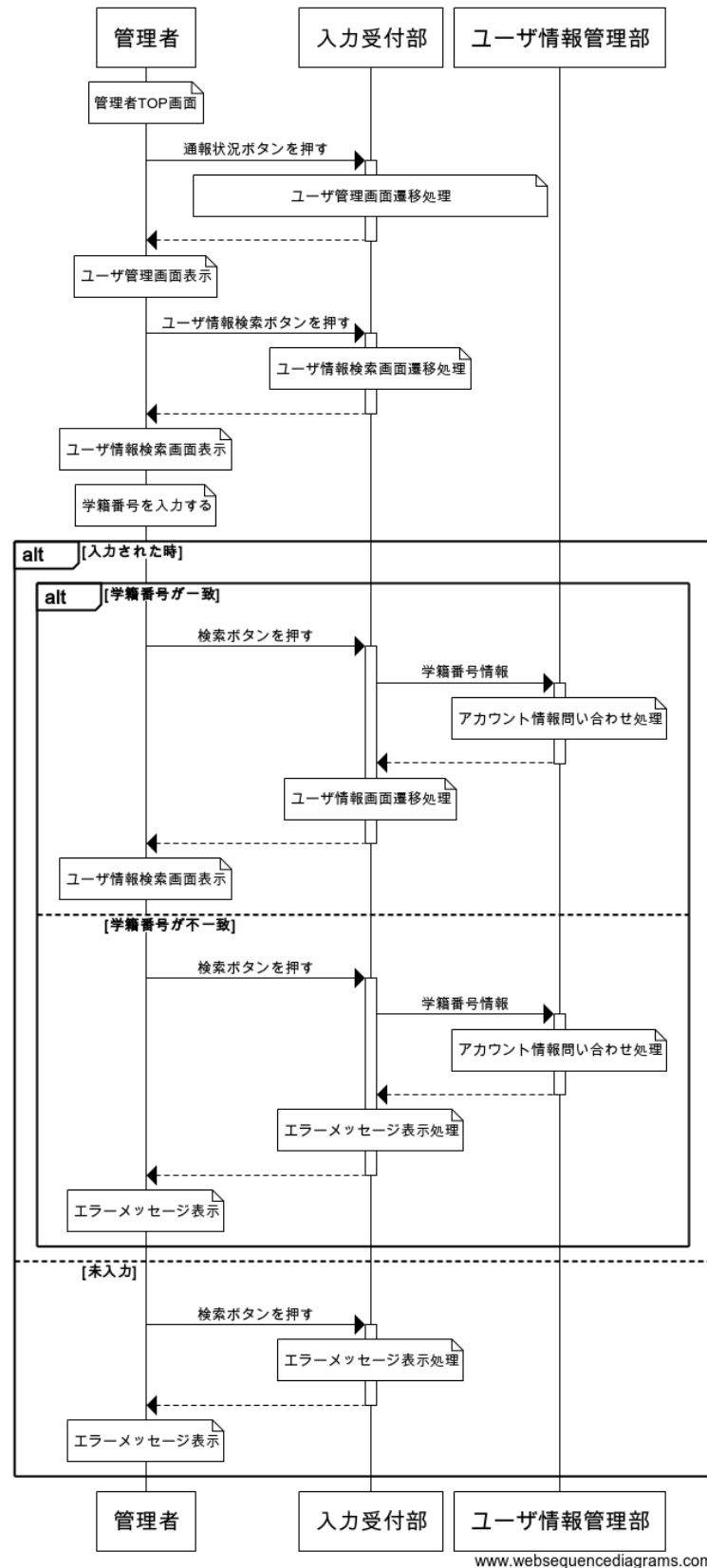


図 22: ユーザ情報機能検索機能とユーザ情報閲覧機能機能のシーケンス図

### 5.15.2 ユーザに対する警告通知送信機能

ユーザに対する警告通知送信機能では管理者が、不適切な内容のスレッド作成またはレスの書き込みを度々行ったユーザに対して、注意喚起の旨をユーザのマイページに通知することができる。ユーザに対する警告通知送信機能のシーケンス図は図 23 に示す。

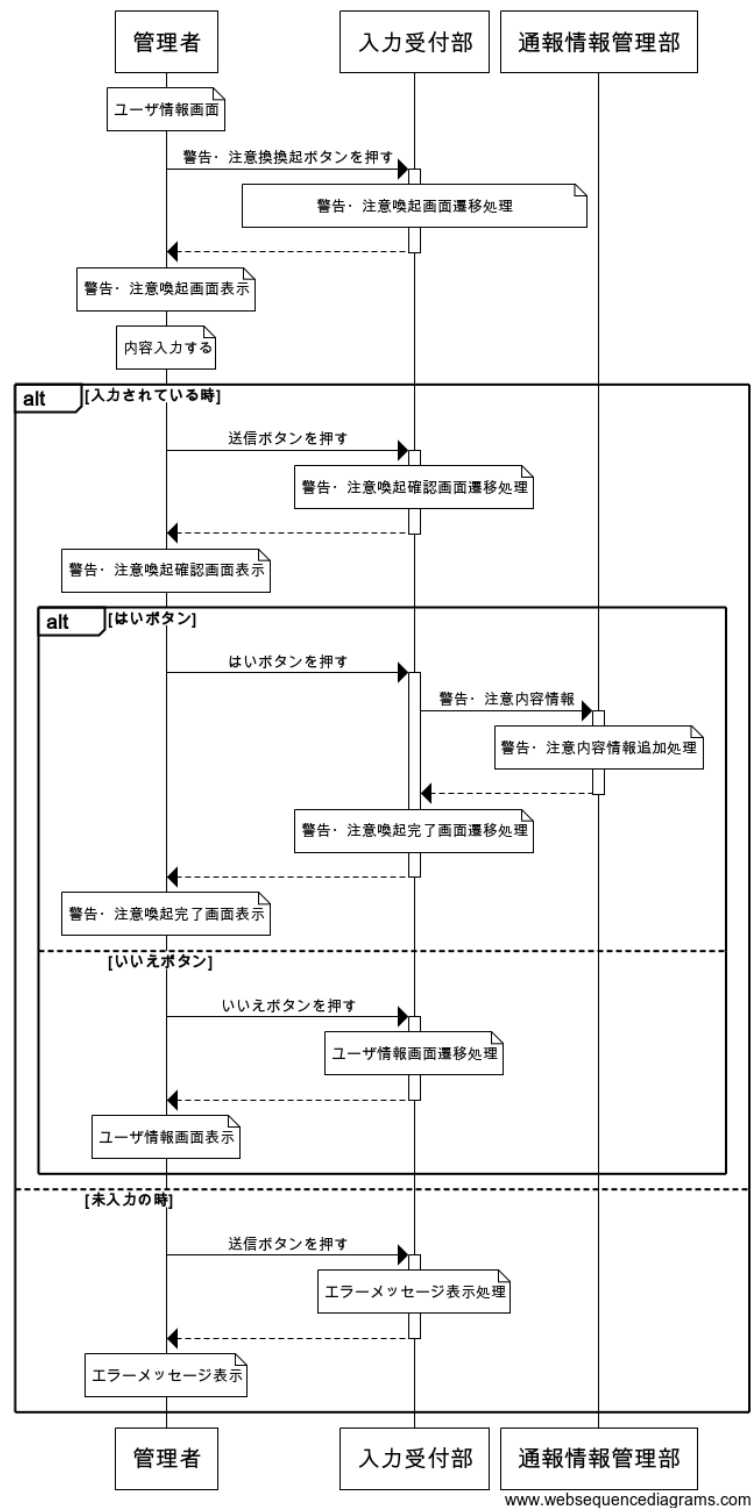


図 23: ユーザに対する警告通知送信機能のシーケンス図

### 5.15.3 ユーザアカウント凍結機能

ユーザアカウント凍結機能管理者が、度重なる注意を通達したにも関わらず、迷惑行為が改善されないユーザのアカウントに対して、書き込み不可能にすることができる。ユーザアカウント凍結機能のシーケンス図は図 24 に示す。

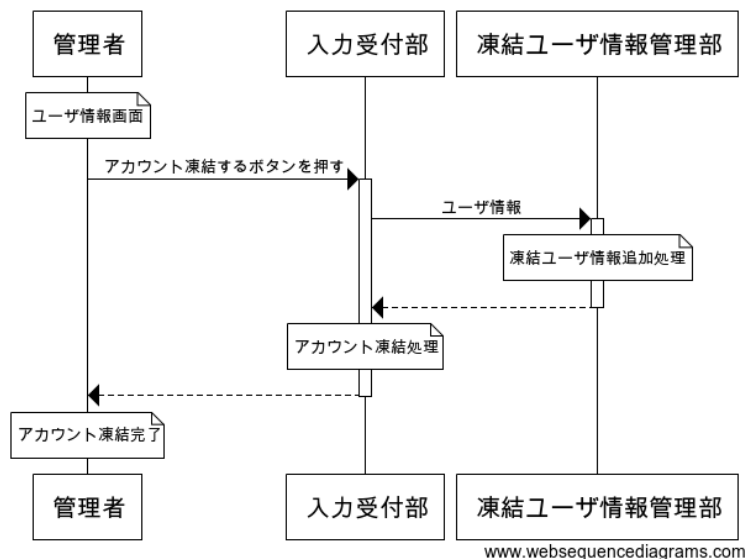


図 24: ユーザアカウント凍結機能のシーケンス図

## 5.16 ユーザ登録サブシステム (管理者用)

管理者はユーザの仮 ID と仮パスワードの発行を行うことができる。仮 ID と仮パスワードを用いて、ユーザは本システムに新規登録することができる。ユーザ登録サブシステムの機能は小節で示す。

### 5.16.1 仮 ID ・ 仮パスワード発行機能

仮 ID ・ 仮パスワード発行機能では管理者が仮 ID と仮パスワードを発行することができる。仮 ID と仮パスワードの生成には、乱数を用いる。 仮 ID ・ 仮パスワード発行機能のシーケンス図は図 25 に示す。

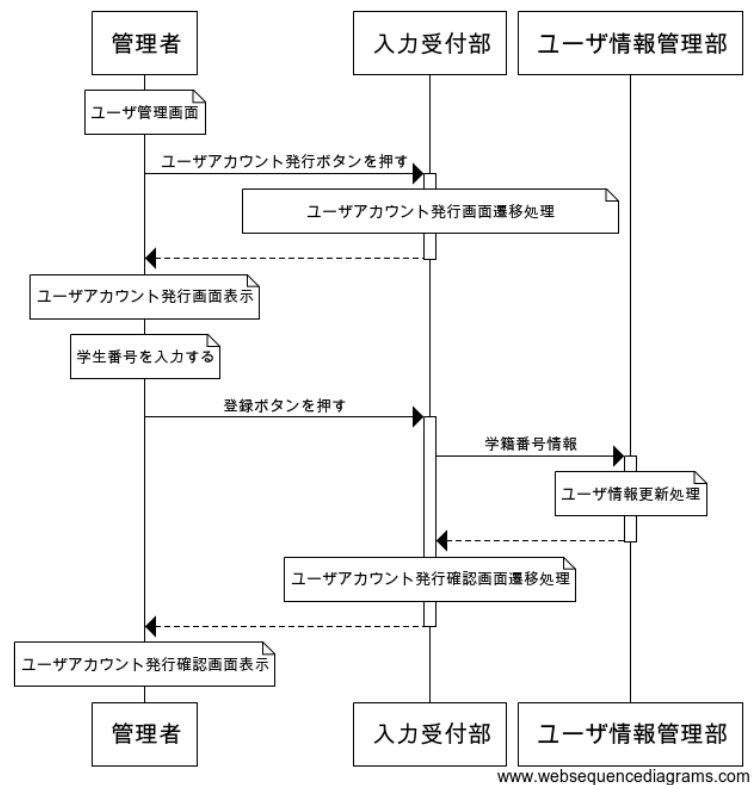


図 25: 仮 ID ・ 仮パスワード発行機能のシーケンス図

## 5.17 管理者ログインサブシステム (管理者用)

管理者は学内 LAN に接続された電子デバイスからのみ、管理者の ID とパスワードを入力することで、管理者として本システムにログインすることができる。  
管理者ログインサブシステムの機能は小節で示す。

### 5.17.1 管理者認証機能

管理者認証では本システムの管理作業を行うユーザが、管理者であるかの認証を、ID とパスワードを用いて行う。認証に成功したユーザは管理者として本システムの管理を行うことができる。管理者認証機能のシーケンス図は図 26 に示す。

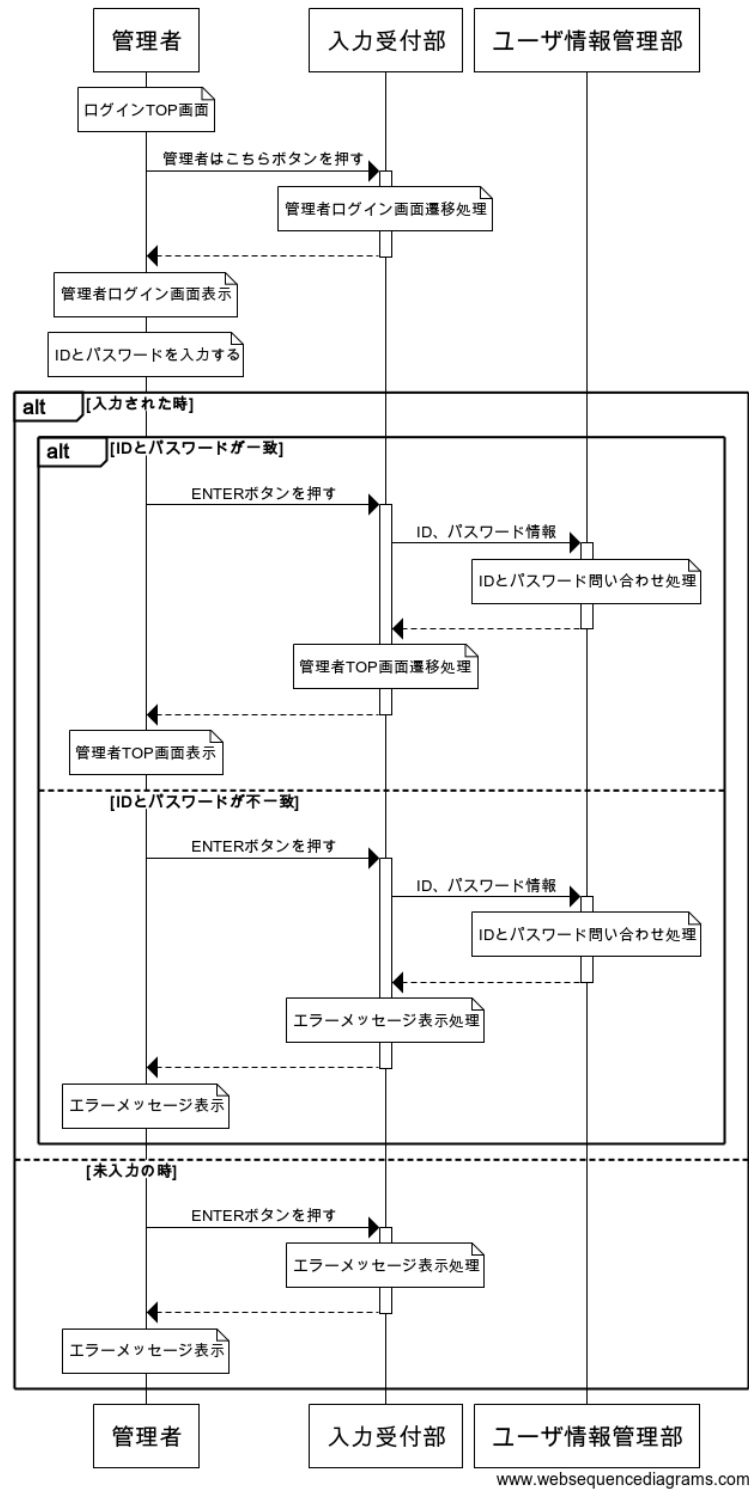


図 26: 管理者認証機能のシーケンス図



## 6 ルーティング及びMVC一覧

この章では、Rails の規約に従った URL 規則を示す。また、HTTP メソッドと URL によって呼び出される Controller と Action を示す。さらに、View 及び Model も示す。

## 6.1 ルーティング一覧

以下の表は、Rails の規則に従った URL 規則の表である。また、Controller とその Action についても示す。

表 14: ルーティング一覧 No.1

No.	URL	METHOD	Controller#Action
1	/top	GET	top#home
2	/login	GET	sessions#new
3	/login	POST	sessions#create
4	/logout	DELETE	sessions#destroy
5	/passwords/:id	GET	passwords#new
6	/passwords/:id	PATCH	passwords#update
7	/passwords/:id/signup	GET	passwords#signup
8	/home	GET	home_pages#home
9	/users/:id	GET	users#show
10	/users/:id/name_update	PATCH	users#name_update
11	/users/:id/notification	PATCH	users#notification
12	/users/:id/warning	GET	users#warning
13	/users/:id/extension	PATCH	users#extension
14	/results	GET	results#search
15	/categories/:id	GET	categories#index
16	/threads/new	GET	threads#new
17	/threads/check	GET	threads#check
18	/threads	POST	threads#create
19	/threads/:id	GET	threads#show
20	/threads/:id/contents/check	GET	contents#check
21	/threads/:id/contents/create	POST	contents#create
22	/threads/:id/contents/correct	PATCH	contents#correct
23	/threads/:id/report_form	GET	report#new
24	/threads/:id/contents/:id/report_form	GET	report#new
25	/threads/:id/report_check	GET	report#check
26	/threads/:id/contents/:id/report_check	GET	report#check
27	/threads/:id/report	POST	report#craete
28	/threads/:id/contents/:id/report	POST	report#create
29	/admins/report	GET	report#index
30	/threads/:id/bookmark_create	POST	bookmark#create
31	/threads/:id/bookmark_destroy	DELETE	bookmark#destroy

表 15: ルーティング一覧 No.2

No.	URL	METHOD	Controller#Action
32	/threads/:id/hide	PATCH	hide#thread_hide
33	/threads/:id/contents/:id/hide	PATCH	hide#res_hide
34	/admins/threads/:id/hide	GET	hide#thread_index
35	/admins/threads/:id/contents/:id/hide	GET	hide#res_index
36	/admins/login	GET	admins_session#new
37	/admins/login	POST	admins_session#login
38	/admins/logout	DELETE	admins_session#logout
39	/admins/top	GET	admins_top#home
40	/admins/ng_word	GET	ng_word#home
41	/admins/ng_word/create	POST	ng_word#create
42	/admins/ng_word/destroy	DELETE	ng_word#destroy
43	/admins/manage	GET	manage#home
44	/admins/manage/signup	GET	manage#signup
45	/admins/manage/signup	POST	manage#create
46	/admins/manage/ok	GET	manage#ok
47	/admins/manage/destroy	GET	manage#destroy_check
48	/admins/manage/destroy	DELETE	manage#destroy
49	/admins/manage_user	GET	manage_user#home
50	/admins/manege_user/signup	GET	manage_user#user_signup
51	/admins/manage_user/signup	POST	manage_user#user_create
52	/admins/manage_user/search	GET	manage_user#search
53	/admins/manage_user/:id	GET	manage_user#show
54	/admins/manage_user/edit	GET	manage_user#warning_edit
55	/admins/manage_user/check	GET	manage_user#warning_check
56	/admins/manage_user/create	POST	manage_user#warning_create
57	/admins/manage_user/id/ban_create	PATCH	manage_user#ban_create
58	/admins/manage_user/id/ban_destroy	PATCH	manage_user#ban_destroy
59	/admins/announce	GET	announce#edit
60	/admins/announce_check	GET	announce#check
61	/admins/announce_create	POST	announce#create
62	/home/announce/:id	GET	announce#show

## 6.2 Controller

### 6.2.1 application\_controller.rb

処理:

-set\_current\_user:現在ログインしているユーザ ID を取得する。

-authenticate\_user:ログインしていないユーザに対してアクセス制限をかける。ログインしていないユーザがアクセス制限がかかっている URL にリクエストした場合、flash メッセージで「ログインが必要です」と表示して/login ヘリダイレクトする。

-forbid\_login\_user:既にログインしているユーザに対してアクセス制限をかける。ログインしているユーザがアクセス制限がかかっている URL にリクエストした場合、flash メッセージで「既にログインしています」と表示して /home へリダイレクトする。

### 6.2.2 top\_controller.rb

名称:セッション情報処理

処理:

-home:トップ画面である top.html.erb を表示させる。ログインボタンを押すと、/session にリダイレクトを行う。画面右下にあるリンクは管理者用ログインフォームで、これを押すと /admin にリダイレクトを行う。

### 6.2.3 sessions\_controller.rb

名称:セッション情報処理

処理:

-new:ログインフォームである sessions.new.html.erb を表示させる。

-create:ユーザの ID とパスワードで認証を行い、認証されるとそのユーザのセッションを作成する。必要なパラメータ params ハッシュから取り出し、取り出した情報を User.find\_by メソッドと authenticate メソッドで用いてデータベースとの照合を行う。セッションの作成に成功した場合、セッションを作成したユーザによって以下にリダイレクトを行う。

ユーザが管理者:/admin に対してリダイレクトを行う。ユーザが一般ユーザ:/home に対してリダイレクトを行う。また、そのユーザが初回ログインの場合は、/passwords にリダイレクトを行う。また、一般ユーザの場合ログインボーナスの処理を行う。具体的には、(create アクション中で) そのユーザの updated\_at カラム（最終ログイン日時）の情報を取得した後、Time オブジェクトを作成し、そのオブジェクトのインスタンス変数から日付を取得する。それらの情報を比較して、一致しないならばログインボーナスの処理（保有ポイントの追加）を update メソッドで行う。情報の変更成功した場合、フラッシュメッセージで、「ログインに成功しました。」と表示する。

なお、認証が失敗した場合は再度/Login に対してリダイレクトを行う。

-destroy:作成したユーザのセッションを delete メソッドで破棄する。

### 6.2.4 passwords\_controller.rb

名称:ID・パスワード変更処理

処理:

-new:初回ログインを行なった（初回ログインフラグが1）のユーザに対して、ID・パスワード変更用フォームである passwords.new.html.erb を表示させる。

-change:あらかじめ登録されていたユーザのパスワード情報を、変更用フォームから入力された新しいパスワードに更新する処理を User.update\_attribute メソッドで行う。正しく更新が行われた場合、「更新を完了しました。」というフラッシュメッセージを表示する。更新が正常に完了したら、/signup にリダイレクトする。

-signup:ID とパスワードを更新した後、更新完了画面である signup.html.erb を表示する。

### 6.2.5 home\_controller.rb

名称：ホーム画面処理

処理:

-home:ホーム画面である home.html.erb を表示する。スレッドタイトル入力テキストボックスに検索したいスレッドタイトルを入力し、検索ボタンを押すと、/results/:id にリダイレクトを行う。お知らせを押すと、/informations/:id にリダイレクトを行う。マイページボタンを押すと、/user/:id にリダイレクトを行う。

-threads\_hide:不適切なスレッドを非表示化する。この処理は、管理者以外行うことはできない (非表示化ボタン自体が存在しない)。このアクションを行うと、/home に対してリダイレクトをする。

### 6.2.6 users\_controller.rb

名称：マイページ画面処理

処理:

-show:マイページ画面である show.html.erb を表示する。

-update:拡張機能を開放する (User テーブルにおけるユーザの各種拡張フラグを 1 に更新する) アクションである。解放後、/user/:id に PATCH メソッドでルーティングにリクエストする。

### 6.2.7 results\_controller.rb

名称：スレッドタイトル検索処理

処理:

-search:スレッドタイトル検索機能の処理を行うアクションである。検索を行い、該当するスレッドを取得した後は、そのスレッドを一覧として表示する (これが search.html.erb となる)。表示されたスレッドタイトルを押すと、/threads/:id/contents に GET メソッドでルーティングをリクエストする。

### 6.2.8 results\_categories\_controller.rb

名称：カテゴリ別スレッドタイトル検索処理

処理:

-search:スレッドタイトル検索機能の処理を行うアクションである。先ほど述べたスレッドタイトル検索処理とほとんど同じであるが、スレッドタイトル以外にも、該当するカテゴリであるかどうかを条件として検索を行う処理である。検索条件以外の処理に違いは存在しない。

### 6.2.9 informations\_controller.rb

名称：お知らせ画面処理

処理:

-show:お知らせ詳細画面である show.html.erb を表示する。

### 6.2.10 categories\_controller.rb

名称：カテゴリ別トップ画面処理

処理:

-new:カテゴリ別トップ画面である categories.html.erb を表示する。この画面もトップ画面同様、スレッドタイトル入力テキストボックスが存在しているがこちらから検索を行うと、/results/:title/categories/:id に GET メソッドでルーティングをリクエストする。マイページに遷移するボタンの処理は、トップ画面と同様である。また、スレッドタイトルを押した場合は、/threads/:id/contents に GET メソッドでルーティングをリクエストする。スレッドタイトルの通報ボタンを押した場合は、/thread/:id/report\_form に GET メソッドでルーティングを行う。

### 6.2.11 threads\_controller.rb

名称：スレッド作成処理

処理:

-new::スレッドを新規作成するフォームである new.html.erb を表示させる。作成ボタンを押した後は /threads/check に GET メソッドでルーティングをリクエストする。  
-check:スレッド作成確認画面である check.html.erb を表示させる。確認ボタンを押すと、/threads に POST メソッドでルーティングをリクエストする。  
-create:new アクションで入力されたスレッドタイトルと最初の書き込み内容を反映したスレッドを作成するアクションである。スレッドを作成した後は、カテゴリ別トップページである /categories/:id に GET メソッドでルーティングをリクエストする。

### 6.2.12 contents\_controller.rb

名称：スレッド閲覧処理

処理:

-bbs::スレッドを閲覧するページを表示するアクションである。通報ボタンを押すと、/thread/:id/contents/:id/report に GET メソッドでルーティングをリクエストする。書き込みボタンを押すと、/thread/:id/contents/check に GET メソッドでルーティングをリクエストする。コレクトボタンを押すと、/correct に PATCH メソッドでルーティングをリクエストする。また、ブックマークボタンをおすと、/threads/:id/bookmark\_create に POST メソッドでルーティングをリクエストする。  
-check:書き込み確認ページである contents.check.html.erb を表示するアクションである。  
-write:thread に書き込みを行うアクションである。書き込み確認ページに存在する書き込みボタンを押すところのアクションが実行され、その後 /thread/:id/contents/ に POST メソッドでルーティングをリクエストする。  
-responses\_hide:このアクションを実行できるのは管理者だけである。このアクションを実行すると、PATCH メソッドで /thread/:id/contents にルーティングをリクエストする。  
-report:このアクションは、通報用フォームである report\_form.html.erb を表示させるアクションである。通報ボタンを押すと、通報確認ページである /thread/:id/contents/:id/report\_form に GET メソッドでルーティングをリクエストする。  
-check:通報確認ページである check.html.erb を表示させるアクションである。  
-correct:コレクトボタンを押したレスに対する処理を行う。

### 6.2.13 form\_controller.rb

名称：通報画面表示・通報処理

処理:

-new:thread・レスの通報画面である new.html.erb を表示させるアクションである。-cheak:通報確認画面である cheak.html.erb を表示するアクションである。create:通報内容をデータベースに反映する処理を行うアクションである。通報後、スレッドを通報していれば通報前の categories/:id に get メソッドでルーティングをリクエストし、レスを通報していれば thread/:id/contents に GET メソッドでルーティングを行う。

### 6.2.14 bookmark.rb

名称：通報画面表示・通報処理

処理:

create:ブックマーク処理を行うアクションである。処理としては、データベースの bookmark テーブルにスレッドの ID 及び登録したユーザの ID を挿入する。処理後、/threads/:id/contents に

### 6.2.15 hide\_controller.rb

処理:

-thread\_hide: 特定のスレッドを非表示にする。スレッド ID を取得し、該当するスレッド非表示フラグを 1 に変更する。

-res\_hide: 特定のレスを非表示にする。レス ID を取得し、該当するレス非表示するフラグを 1 に変更する。

-thread\_index: /admins/threads/:id/hide.html.erb を表示する。また、スレッド ID から非表示フラグが 1 であるスレッドを検出する。

-res\_index: /admins/threads/:id/contents/:id/hide.html.erb を表示する。また、レス ID から非表示フラグが 1 であるレスを検出する。

### 6.2.16 admins\_session\_controller.rb

処理:

- new: /admins/login.html.erb を表示する。- login: 入力された ID とパスワードを取得する。if 文でデータベース上に存在するか判別する。

存在する場合の処理

- session 変数にデータベース上のユーザ id を代入する。
- flash メッセージで「ログインしました」と表示する。
- /admins/top へリダイレクトする。

存在しない場合の処理

- エラーメッセージ用の変数にエラーメッセージを代入する。

- 入力された ID とパスワードを初期値として設定する。
- /admins/login へリダイレクトする。

-logout: session 変数に nil を代入する。flash メッセージで「ログアウトしました」と表示する。/admins/login へリダイレクトする。

### 6.2.17 admins\_top\_controller.rb

処理:

-home: /admins/top.html を表示する。

### 6.2.18 ng\_word\_controller.rb

処理:

-home: admins/ng\_word.html.erb を表示する。

-create: 入力された文字列を受け取り、受け取った文字列をデータベースに保存する。保存に成功した場合、flash メッセージで「単語を登録しました」と表示し、/admins/ng\_word へリダイレクトする。

-destroy: 特定の文字列を受け取り、削除処理をする。削除後、flash メッセージで「単語を削除しました」と表示して /admins/ng\_word へリダイレクトする。

### 6.2.19 manage\_controller.rb

処理:

-home: /admins/manage.html.erb を表示する。

-signup: /admins/manage/signup.html.erb を表示する。

-create: 入力された ID とパスワードを取得し、データベースに保存する。保存後、/admins/manage/ok へリダイレクトする。

-ok: /admins/manage/ok.html を表示する。-destroy\_check: /admins/manage/destroy.html.erb を表示する。

-destroy: 特定の子管理者アカウント情報を受け取り、削除処理をする。flash メッセージで「子管理者アカウントを削除しました」と表示し、/admins/magnage へリダイレクトする。

### 6.2.20 manage\_user\_controller.rb

処理:

-home: /admins/manage\_user.html.erb を表示する。

-signup: /admins/manage\_user/signup.html.erb を表示する。

-create: 受け取った整数の範囲のユーザの ID とパスワードを乱数で生成し、データベースに保存する。保存に成功した場合、flash メッセージで「ユーザアカウントを作成しました」と表示し、/admins/manage\_user へリダイレクトする。保存に失敗した場合、エラーメッセージを表示し、さらに入力された文字列を初期値として /admins/manage\_user/signup へリダイレクトする。

-search: 入力された学籍番号を受け取り、/admins/manage\_user/search.html.erb を表示する。

-show: 特定のユーザの id を受け取り、/admins/manage\_user/:id.html.erb を表示する。

-warning\_edit: admins/manage\_user/:id/warning\_edit.html.erb を表示する。



-warning\_check: admins/manage\_user/:id/warning\_check.html.erb を表示する。  
-warning\_create: 入力された文字列を受け取り、データベースに保存する。保存に成功した場合、flash メッセージで「警告・注意喚起を作成しました」と表示し、admins/manage\_user/:id へリダイレクトする。  
-ban\_create: 特定のユーザ ID を凍結ユーザテーブルに保存する。処理後は admins/manage\_user/:id へリダイレクトする。  
-ban\_destroy: 特定のユーザ ID を凍結ユーザテーブルから削除する。処理後は admins/manage\_user/:id へリダイレクトする。

### 6.2.21 announce\_controller.rb

処理:

-edit: /admins/announce.html.erb を表示する。  
-check: /admins/announce\_check.html.erb を表示する。  
-create: 入力されたお知らせ内容をデータベースに保存する。保存後は admins/announce へリダイレクトする。  
-show: /home/announce/:id.html.erb を表示する。

## 6.3 View 層

(現時点では掲示板の画面詳細のみ記述しています。v2 で管理者側の画面詳細の追記とこの章全体の修正をします。)

### 6.3.1 top/home.html.erb

【名称】

ログイン TOP 画面

【概要】

ユーザ・管理者のログイン画面に遷移するためのボタンを表示する画面である。

【処理フロー】

- ログインボタンを押すと、/login に GET メソッドでルーティングにリクエストする。
- 管理者はこちらボタンを押すと、/admins/login に GET メソッドでルーティングにリクエストする。

### 6.3.2 sessions/new.html.erb

【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】

- ENTER ボタンを押すと、入力された ID とパスワードを情報として/login に POST メソッドでルーティングにリクエストする。
- 初回ログインである場合に限り、ENTER ボタンを押すと、/passwords/:id に GET メソッドでルーティングにリクエストする。

### 6.3.3 passwords/new.html.erb

【名称】

ID・パスワード変更画面

【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】

- 登録ボタンを押すと、入力された ID とパスワードを情報として /passwords/:id に PATCH メソッドでルーティングにリクエストする。

### 6.3.4 passwords/signup.html.erb

【名称】

ID・パスワード変更完了画面

【概要】

初回ログインをしたユーザが ID とパスワードの変更が完了したことを確認する画面である。

【処理フロー】

- 利用開始ボタンを押すと、/home に GET メソッドでルーティングにリクエストする。

### 6.3.5 home\_pages/home.html.erb

【名称】

掲示板 TOP 画面 【概要】

掲示板 TOP 画面を表示する画面である。

【処理フロー】

- 検索ボタンを押すと、/results に GET メソッドでルーティングにリクエストする。
- お知らせタイトルを押すと、/home/announce/:id に GET メソッドでルーティングにリクエストする。

### 6.3.6 announce/show.html.erb

【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。

【処理フロー】 -

### 6.3.7 users/show.html.erb

【名称】

マイページ画面

【概要】

ユーザのマイページを表示する画面である。

【処理フロー】

- ハンドルネームの変更ボタンを押すと、入力された名前を情報として、`/users/:id/name_update` に PATCH メソッドでルーティングにリクエストする。
- 通知設定のラジオボタンを押すと、ラジオボタンの入力を情報として、`/users/:id/notification` に PATCH メソッドでルーティングにリクエストする。
- 警告・注意喚起確認ボタンを押すと、`/users/:id/warning` に GET メソッドでルーティングにリクエストする。
- 拡張機能の開放ボタンを押すと、保有ポイントを情報として、`/users/:id` に PATCH メソッドでルーティングにリクエストする。

### 6.3.8 results/search.html.erb

【名称】

検索結果画面

【概要】

スレッドの検索結果を表示する画面である。

【処理フロー】

- 検索ボタンを押すと、`/results` に GET メソッドでルーティングにリクエストする。
- スレッドタイトルを押すと、`/threads/:id` に GET メソッドでルーティングにリクエストする。
- 通報ボタンを押すと、`/threads/:id/report_form` に GET メソッドでルーティングにリクエストする。
- 非表示ボタンを押すと、スレッド ID を情報として、`/threads/:id/hide` に対して PATCH メソッドでルーティングにリクエストする。

### 6.3.9 categories/index.html.erb

【名称】

カテゴリ TOP 画面

【概要】

各カテゴリのカテゴリ ID に該当するスレッドを表示する画面である。

【処理フロー】

- スレッド新規作成ボタンを押すと、`/thread/new` に GET メソッドでルーティングにリクエストする。
- 検索ボタンを押すと、`/result` に GET メソッドでルーティングにリクエストする。
- スレッドタイトルを押すと、`/thread/:id/` に GET メソッドでルーティングにリクエストする。
- 通報ボタンを押すと、`/threads/:id/report_form` に GET メソッドでルーティングにリクエストする。
- 非表示ボタンを押すと、スレッド ID を情報として、`/threads/:id/hide` に対して PATCH メソッドでルーティングにリクエストする。

### 6.3.10 threads/new.html.erb

【名称】

スレッド作成画面

【概要】

スレッドの新規作成フォームを表示する画面である。

【処理フロー】

- 作成ボタンを押すと、/threads/check に GET メソッドでルーティングにリクエストする。

### 6.3.11 threads/check.html.erb

【名称】

スレッド作成確認画面

【概要】

新規作成するスレッドの内容確認を表示する画面である。

【処理フロー】

- はいボタンを押すと、スレッドタイトル名と書き込み内容を情報として、/threads に POST メソッドでルーティングにリクエストする。
- いいえボタンを押すと、/threads/new に GET メソッドでルーティングにリクエストする。

### 6.3.12 threads/show.html.erb

【名称】

スレッド閲覧画面

【概要】

スレッドの書き込み及び書き込みの入力フォームを表示する画面である。

【処理フロー】

- レスの書き込みフォームは、凍結ユーザには表示されない。
- ブックマークボタンを押すと、スレッド ID を情報として、/bookmark/create に POST メソッドでルーティングにリクエストする。
- ブックマークボタンを押すと、スレッド ID を情報として、/bookmark/create に POST メソッドでルーティングにリクエストする。
- 通報ボタンを押すと、/threads/:id/contents/report に GET メソッドでルーティングにリクエストする。
- 非表示ボタンを押すと、スレッド ID とレス ID を情報として、/threads/:id/contents/ に PATCH メソッドでルーティングにリクエストする。
- 送信ボタンを押すと、/threads/:id/contents/check に GET メソッドでルーティングにリクエストする。

### 6.3.13 contents/check.html.erb

【名称】

レス書き込み内容確認画面

【概要】

スレッドに書き込むレスの内容確認を表示する画面である。

【処理フロー】

- はいボタンを押すと、書き込み内容を情報として、/contents/write に POST メソッドでルーティングにリクエストする。
- いいえボタンを押すと、/threads/show に GET メソッドでルーティングにリクエストする。

### 6.3.14 report/new.html.erb

【名称】

通報画面

【概要】

不適切な書き込みを通報する入力フォームを表示する画面である。

【処理フロー】

- 通報対象がスレッドの場合、スレッドタイトルのみを表示する。
- 通報対象がレスの場合、スレッドタイトルとレスを表示する。
- 送信ボタンを押すと、/threads/:id/contents/:id/check に GET メソッドでルーティングにリクエストする。

### 6.3.15 report/check.html.erb

【名称】

通報内容確認画面

【概要】

通報の内容確認を表示する画面である。

【処理フロー】

- はいボタンを押すと、通報内容を情報として、/threads/:id/contents に POST メソッドでルーティングにリクエストする。
- いいえボタンを押すと、/threads/:id/contents/:id/report に GET メソッドでルーティングにリクエストする。

### 6.3.16 admins/login.html.erb

【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】

- ENTER ボタンを押すと、入力された ID とパスワードを情報として /admins/login に POST メソッドでルーティングにリクエストする。

### 6.3.17 admins\_top/home.html.erb

【名称】

管理者 TOP 画面

【概要】

管理者が各管理画面に遷移するための画面である。

【処理フロー】

- 掲示板はこちらボタンを押すと、/home に GET メソッドでルーティングにリクエストする。
- 子管理者管理ボタンを押すと、/admins/manage に GET メソッドでルーティングにリクエストする。なお、管理者フラグが親管理者の状態であるもののみ、このボタンが表示される。
- ユーザ管理ボタンを押すと、/admins/manage\_user に GET メソッドでルーティングにリクエストする。
- お知らせ編集ボタンを押すと、/admins/announce に GET メソッドでルーティングにリクエストする。
- 通報状況確認ボタンを押すと、/admins/report に GET メソッドでルーティングにリクエストする。
- 不適切な単語登録ボタンを押すと、/admins/ng\_words に GET メソッドでルーティングにリクエストする。
- 非表示一覧表示ボタンを押すと、/admins/threads/:id/hide に GET メソッドでルーティングにリクエストする。

### 6.3.18 manage/home.html.erb

【名称】

子管理者管理画面

【概要】

親管理者が子管理者を管理するための画面である。

【処理フロー】

- 子管理者アカウント発行ボタンを押すと、/admins/manage/signup に GET メソッドでルーティングにリクエストする。
- 抹消ボタンを押すと、/admins/manage/destoroy に GET メソッドでルーティングにリクエストする。

### 6.3.19 manage/signup.html.erb

【名称】

子管理者アカウント発行画面

【概要】

親管理者が子管理者のアカウントを発行するための画面である。

【処理フロー】

- はいボタンを押すと、乱数により子管理者の ID とパスワードを生成し、それらを情報として、/admins/manage/signup に POST メソッドでルーティングにリクエストする。
- いいえボタンを押すと、/admins/manage に GET メソッドでルーティングにリクエストする。

#### 6.3.20 manage/ok.html.erb

【名称】

子管理者アカウント発行完了画面

【概要】

親管理者が子管理者のアカウントの発行完了を確認するための画面である。また、この画面にて、発行した子管理者の ID とパスワードを表示する。

【処理フロー】

- 確認ボタンを押すと、/admins/manage に POST メソッドでルーティングにリクエストする。
- 管理者 TOP ボタンを押すと、/admins\_top/home に GET メソッドでルーティングにリクエストする。

#### 6.3.21 manage/destroy\_check.html.erb

【名称】

子管理者アカウント抹消画面

【概要】

親管理者が子管理者のアカウントの抹消を行うための画面である。

【処理フロー】

- はいボタンを押すと、/admins/manage/destroy に DELETE メソッドでルーティングにリクエストする。
- いいえボタンを押すと、/admins/manage に GET メソッドでルーティングにリクエストする。

#### 6.3.22 manage\_user/home.html.erb

【名称】

ユーザ管理画面

【概要】

管理者がユーザの各管理をするための画面である。

【処理フロー】

- ユーザアカウント発行ボタンを押すと、/admins/manage\_user/signup に GET メソッドでルーティングにリクエストする。
- ユーザ情報検索ボタンを押すと、/admins/manage\_user/search に GET メソッドでルーティングにリクエストする。

### 6.3.23 manage\_user/user\_signup.html.erb

【名称】

ユーザアカウント発行画面

【概要】

管理者がユーザの各管理をするための画面である。

【処理フロー】

- 発行ボタンを押すと、入力された学籍番号（開始番号・末尾番号）を情報として、/admins/manage/signup に POST メソッドでルーティングにリクエストする。
- 管理者 TOP ボタンを押すと、/admins\_top/home に GET メソッドでルーティングにリクエストする。

### 6.3.24 manage\_user/search.html.erb

【名称】

ユーザ情報検索画面

【概要】

管理者がユーザの情報を検索するための画面である。

【処理フロー】

- 検索ボタンを押すと、/admins/manage/:id に GET メソッドでルーティングにリクエストする。
- 管理者 TOP ボタンを押すと、/admins\_top/home に GET メソッドでルーティングにリクエストする。

### 6.3.25 manage\_user/show.html.erb

【名称】

ユーザ情報画面

【概要】

管理者がユーザの情報を閲覧するための画面である。

【処理フロー】

- 警告・注意喚起ボタンを押すと、/admins/manage\_user/edit に GET メソッドでルーティングにリクエストする。
- ユーザが凍結ユーザでない場合にアカウント凍結ボタンを押すと、ID と学籍番号を情報として、/manage\_user/:id/ban に POST メソッドでルーティングにリクエストする。
- ユーザが凍結ユーザである場合にアカウント凍結解除ボタンを押すと、/admins\_top/:id/ban に DELETE メソッドでルーティングにリクエストする。

### 6.3.26 manage\_user/search.html.erb

【名称】

ユーザ情報検索画面

【概要】

管理者がユーザの情報を検索するための画面である。

【処理フロー】



- 検索ボタンを押すと、/admins/manage/:id に GET メソッドでルーティングにリクエストする。
- 管理者 TOP ボタンを押すと、/admins\_top/home に GET メソッドでルーティングにリクエストする。

### 6.3.27 manage\_user/warning\_edit.html.erb

【名称】

警告・注意喚起画面

【概要】

管理者がユーザに送信する警告・注意喚起を記述するための画面である。

【処理フロー】

- 送信ボタンを押すと、/admins/manage\_user/check に GET メソッドでルーティングにリクエストする。
- ユーザ情報ページに戻るボタンを押すと、/admins/manage\_user に GET メソッドでルーティングにリクエストする。

### 6.3.28 manage\_user/warning\_check.html.erb

【名称】

警告・注意喚起確認画面

【概要】

管理者がユーザに送信する警告・注意喚起の内容確認をするための画面である。

【処理フロー】

- はいボタンを押すと、入力された警告・注意喚起内容を情報として、/admins/manage\_user/create に POST メソッドでルーティングにリクエストする。
- いいえボタンを押すと、/admins/manage\_user に GET メソッドでルーティングにリクエストする。

### 6.3.29 announce/edit.html.erb

【名称】

お知らせ編集画面

【概要】

管理者がお知らせを編集するための画面である。

【処理フロー】

- 送信ボタンを押すと、/admins/announce\_check/ に GET メソッドでルーティングにリクエストする。
- 管理者 TOP ボタンを押すと、/admins\_top/home に GET メソッドでルーティングにリクエストする。

### 6.3.30 announce/check.html.erb

【名称】

お知らせ編集確認画面

【概要】

管理者がお知らせを編集するための画面である。

【処理フロー】

- はいボタンを押すと、入力されたお知らせタイトルと内容を情報として、/admins/ng\_words/create に POST メソッドでルーティングにリクエストする。
- いいえボタンを押すと、/admins/announce に GET メソッドでルーティングにリクエストする。

### 6.3.31 admins/login.html.erb

【名称】

通報状況画面

【概要】

通報の一覧を表示する画面である。

【処理フロー】

- 非表示ボタンを押すと、スレッド ID とレス ID を情報として、/threads/:id/contents/ に PATCH メソッドでルーティングにリクエストする。
- 非表示ボタンを押すと、入力された ID とパスワードを情報として /admins/login に POST メソッドでルーティングにリクエストする。

### 6.3.32 ng\_words/home.html.erb

【名称】

不適切な単語登録画面

【概要】

管理者が不適切であると判断した単語を登録するための画面である。

【処理フロー】

- 登録ボタンを押すと、入力された単語を情報として、/admins/ng\_words/create に POST メソッドでルーティングにリクエストする。
- 削除ボタンを押すと、/admins/ng\_words/destroy に DELETE メソッドでルーティングにリクエストする。

### 6.3.33 hide/thread\_index.html.erb

【名称】

非表示スレッド一覧画面

【概要】

管理者が非表示にしたスレッドの一覧を表示する画面である。

【処理フロー】

- レスはこちらボタンを押すと、/amins/thread/:id/contents/:id/hide に GET メソッドでルーティングにリクエストする。
- 管理者 TOP ボタンを押すと、/admins\_top/home に GET メソッドでルーティングにリクエストする。

#### 6.3.34 hide/res\_index.html.erb

【名称】

非表示レス一覧画面

【概要】

管理者が非表示にしたレスの一覧を表示する画面である。

【処理フロー】

- スレッドはこちらボタンを押すと、/admins/thread/:id/hide に GET メソッドでルーティングにリクエストする。
- 管理者 TOP ボタンを押すと、/admins\_top/home に GET メソッドでルーティングにリクエストする。

## 6.4 Model 層

### 6.4.1 user.rb

処理:バリデーションを設定する。

- null 値なし。
- ID は一意性制約を持つ。
- 文字数制限:ID・パスワードは最高 20 文字かつ最低 8 文字、ハンドルネームは最大 15 文字。

関係:

表 16: user.rb と他のモデルとの関係

	関係	他モデル
user.rb	1:N	bookmark.rb
user.rb	1:N	correct_user.rb
user.rb	1:N	alert.rb
user.rb	1:1	freeze.rb
user.rb	1:N	thread.rb
user.rb	1:N	res.rb
user.rb	1:N	report.rb

### 6.4.2 bookmark.rb

処理:バリデーションを設定する。

- null 値なし。

関係:

表 17: bookmark.rb と他のモデルとの関係

	関係	他モデル
bookmark.rb	1:N	thread.rb
bookmark.rb	N:1	user.rb

#### 6.4.3 correct\_user.rb

処理:バリデーションを設定する。

- null 値なし。

関係:

表 18: correct\_user.rb と他のモデルとの関係

	関係	他モデル
correct_user.rb	N:1	res.rb
correct_user.rb	N:1	user.rb

#### 6.4.4 alert.rb

処理:バリデーションを設定する。

- null 値なし。
- 文字数制限:タイトルは最高 50 文字かつ最低 5 文字、内容は最高 400 文字かつ最低 40 文字。

関係:

表 19: alert.rb と他のモデルとの関係

	関係	他モデル
alert.rb	N:1	user.rb

#### 6.4.5 thread.rb

処理:バリデーションを設定する。

- null 値なし。
- 文字数制限:スレッドタイトルは最高 50 文字かつ最低 5 文字。

関係:

表 20: thread.rb と他のモデルとの関係

	関係	他モデル
thread.rb	N:1	category.rb
thread.rb	1:N	report.rb
thread.rb	1:N	res.rb
thread.rb	N:1	user.rb
thread.rb	1:1	bookmark.rb

#### 6.4.6 res.rb

処理:バリデーションを設定する。

- null 値なし。
- 文字数制限:最高 200 文字。

関係:

表 21: res.rb と他のモデルとの関係

	関係	他モデル
res.rb	1:N	report.rb
res.rb	M:N	ng_word.rb
res.rb	N:1	user.rb
res.rb	1:N	correct_user.rb
res.rb	N:1	thread.rb

#### 6.4.7 announcement.rb

処理:バリデーションを設定する。

- null 値なし。
- 文字数制限:お知らせタイトルは最大 50 文字かつ最低 5 文字、お知らせ内容は最大 400 文字かつ最低 40 文字。

#### 6.4.8 category.rb

処理:バリデーションを設定する。

- null 値なし。

関係:

表 22: category.rb と他のモデルとの関係

	関係	他モデル
category.rb	1:N	thread.rb

#### 6.4.9 report.rb

処理:バリデーションを設定する。

- null 値なし。
- 文字数制限:最高 200 文字かつ最低 5 文字。

関係:

表 23: report.rb と他のモデルとの関係

	関係	他モデル
report.rb	N:1	user.rb
report.rb	N:1	thread.rb
report.rb	N:1	res.rb

#### 6.4.10 ng\_word.rb

処理:バリデーションを設定する。

- null 値なし。
- 一意性制約 (ユニーク) を持つ。
- 文字数制限:最高 200 文字。

関係:

表 24: ng\_word.rb と他のモデルとの関係

	関係	他モデル
ng_word.rb	N:1	res.rb

#### 6.4.11 freeze.rb

処理:バリデーションを設定する。

- null 値なし。

関係:

表 25: freeze.rb と他のモデルとの関係

	関係	他モデル
freeze.rb	1:1	user.rb