

# 内部設計書

グループ: T10

システム名:「自動テスト採点システム」

## 第 10 チーム

メンバ:	クラス: <u>IS1</u>	番号: <u>10</u>	名前: <u>ダン・ヴァン・ルアン</u>
	クラス: <u>IS1</u>	番号: <u>11</u>	名前: <u>リー・トゥアン・ナム</u>
	クラス: <u>IS2</u>	番号: <u>01</u>	名前: <u>リュウ・テー・アイン</u>
	クラス: <u>IS2</u>	番号: <u>02</u>	名前: <u>リー・ティ・ボン</u>
	クラス: <u>IS2</u>	番号: <u>15</u>	名前: <u>トゥオン・ギーア</u>
	クラス: <u>IS3</u>	番号: <u>16</u>	名前: <u>ゴー・アイン・トゥアン</u>

## 目次

I. テーブル設計 .....	5
1. ER 図 .....	5
2. テーブルの一覧表 .....	5
2.1 User テーブル .....	5
2.2 Organization テーブル .....	6
2.3 Booklet テーブル .....	6
2.4 Exam テーブル .....	6
II. 処理設計 .....	7
1. ログインモジュール .....	7
1.1. ログイン .....	7
1.2. 個人情報変更 .....	8
1.3. パスワードリセット .....	9
2. テストモジュール .....	10
2.1 解答者にテストの情報を送る .....	10
2.2 テスト問題を作成する .....	11
2.3 テストの選択 .....	12
2.4 テスト画面 .....	13
2.5 テスト結果の送信 .....	14
2.6 テスト問題を採点 .....	15
3. ユーザ管理モジュール .....	16
3.1. ユーザ検索 .....	16
3.2. ユーザ追加 .....	17
3.3. ユーザ削除 .....	18
4. 団体者管理モジュール .....	19
4.1. 一覧団体者を表示 .....	19
4.2. 団体者検索 .....	20
4.3. 団体者追加 .....	21
4.4. 団体者削除 .....	22

5. お知らせ管理モジュール .....	23
5.1 お知らせ変更.....	23
5.2 お知らせ追加 .....	24
5.3 お知らせ削除.....	25
III. メッセージ設計 .....	26
IV. ログ設計 .....	30
1. 作成法 .....	30
1.1. システムが各ログタイプ.....	30
1.2. ログ作成法.....	30
1.3. ログを保存する時間.....	31
2. ログレコードの一覧表 .....	31
2.1. ログインモジュールのログ.....	31
2.2. テストモジュールのログ.....	33
2.3. ユーザ管理モジュールのログ.....	38
2.4. 団体者管理モジュールログ.....	40
2.5. お知らせ管理ログ.....	41
2.6. デバッグログ.....	43
V. 例外設計 .....	43
1. 二つのユーザシステムにログインの例外 .....	43
2. テストを解答するとき、インターネットをコネクションできない例外 ...	43
3. 採点者がテストを採点するとき、インターネットをコネクションできない例外 .....	43
VI. 定数・コード設計 .....	44
1. 定数設計 .....	44
2. コード設計 .....	44

## 図の目次

図 1 : システムの ER 図 .....	5
図 2 : ユーザログインアクティビティ .....	7
図 3 : 個人情報変更アクティビティ .....	8
図 4 : パスワードリセットアクティビティ .....	9
図 5 : 解答者にテストの情報を送るアクティビティ .....	10
図 6 : テスト問題を作成するアクティビティ .....	11
図 7 : テストの選択アクティビティ .....	12
図 8 : テスト画面アクティビティ .....	13
図 9 : テスト結果の送信アクティビティ .....	14
図 10 : テスト問題を採点アクティビティ .....	15
図 11 : ユーザ検索アクティビティ .....	16
図 12 : ユーザ追加アクティビティ .....	17
図 13 : ユーザ削除アクティビティ .....	18
図 14 : 一覧団体者を表示アクティビティ .....	19
図 15 : 団体者を検索アクティビティ .....	20
図 16 : 団体者追加アクティビティ .....	21
図 17 : 団体者削除アクティビティ .....	22
図 18 : 予約参照と状態変更アクティビティ (管理者) .....	23
図 19 : 収入と予約数を統計アクティビティ .....	24
図 20 : 使用実績と課長参照アクティビティ .....	25
図 21 : ログディレクトリ .....	30

# I. テーブル設計

## 1. ER 図

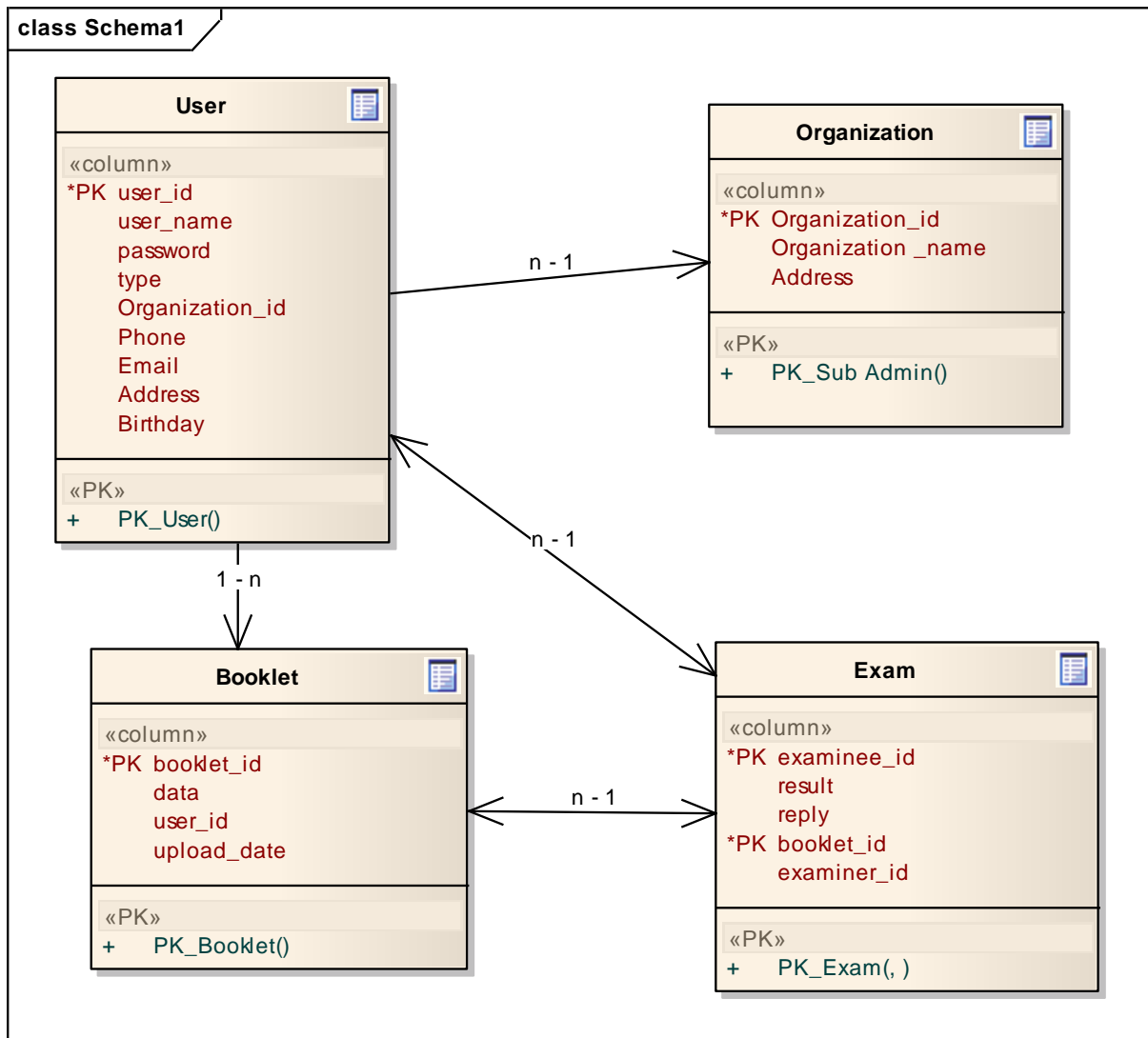


図 1 : システムの ER 図

## 2. テーブルの一覧表

### 2.1 User テーブル

項目	タイプ	主キー	記述
user_id	Char(8)	o	一つのレコードを特定する項目です。自動的に増加します。
Organization_id	Char(8)		ユーザの団体
user_name	Varchar(30)		一つのユーザを特定するための項目。重複できません。
Password	Varchar(20)		ユーザのパスワードを SHA のハッシュ関数でハッシュしましたテキスト

			です。
Type	Char(4)		ユーザのタイプ。 例：1000：回答者 0100：出題者 0010：採点者 0001：団体者
Phone	int		ユーザの携帯電話番号
Email	Varchar(30)		ユーザのメールアドレス
Address	Varchar(30)		ユーザのアドレス
Birthday	Date		ユーザの誕生日

## 2.2 Organization テーブル

項目	タイプ	主キー	記述
Organization_id	Char(8)	o	一つのレコードを特定する項目です。自動的に増加します。
Organization_name	Varchar(30)		団体の名を表示します。
Address	Varchar(20)		団体のアドレス。

## 2.3 Booklet テーブル

項目	タイプ	主キー	記述
booklet_id	Char(8)	o	一つのレコードを特定する項目です。自動的に増加します。
user_id	Char(8)		このテストを作る人の Id
upload_date	Date		CSV をアップロードします。
Data	Longtext		テストの内容

## 2.4 Exam テーブル

項目	タイプ	主キー	記述
examinee_id	Char(8)	o	このテストを解答すとひとです。
booklet_id	Char(8)	o	Exam にテストのテスト Id を使用します。
Examiner_id	Char(8)		採点者の ID
Reply	Longtext		この人のテストの解答です。
Result	Int		この人のテストの結果です。

## II.処理設計

### 1.ログインモジュール

#### 1.1. ログイン

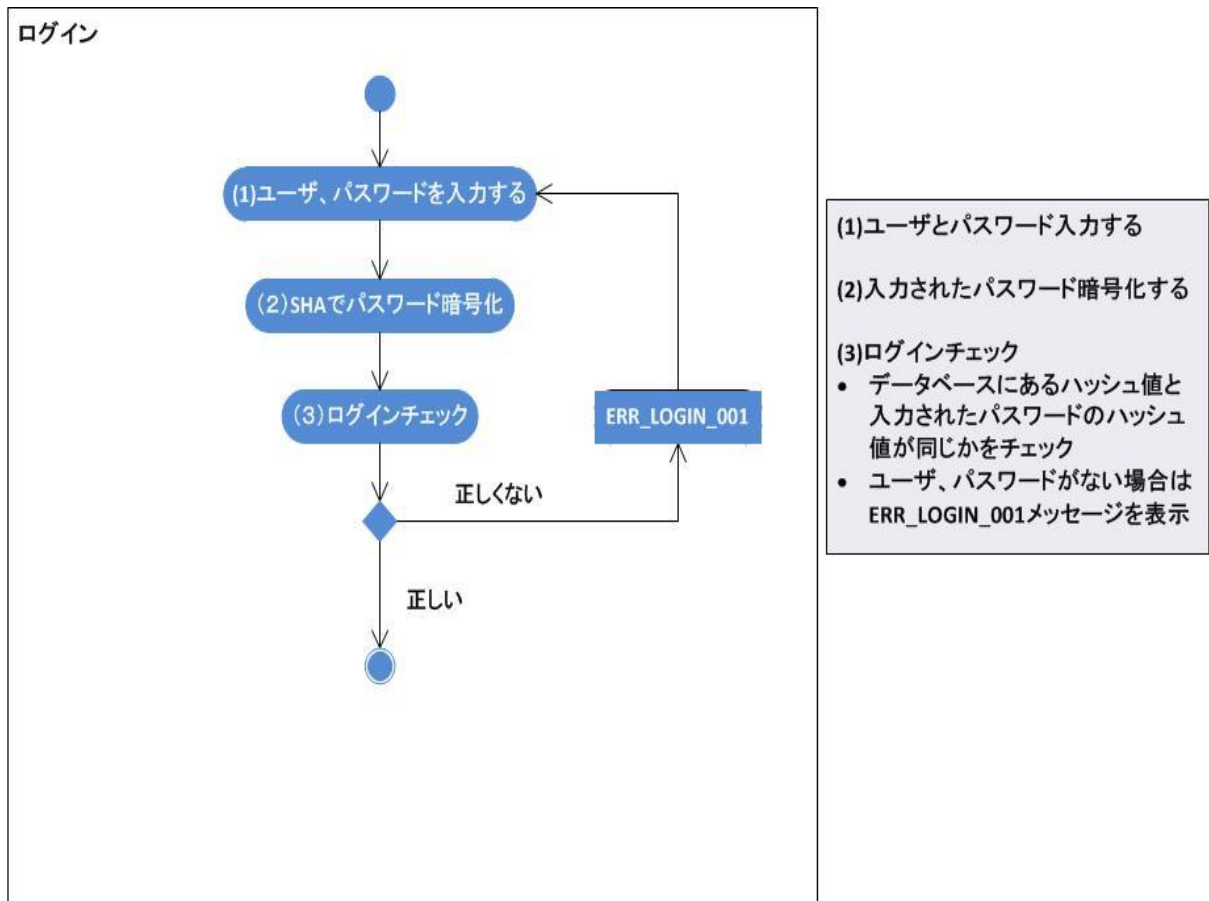


図 2 : ユーザログインアクティビティ



## 1.2. 個人情報変更

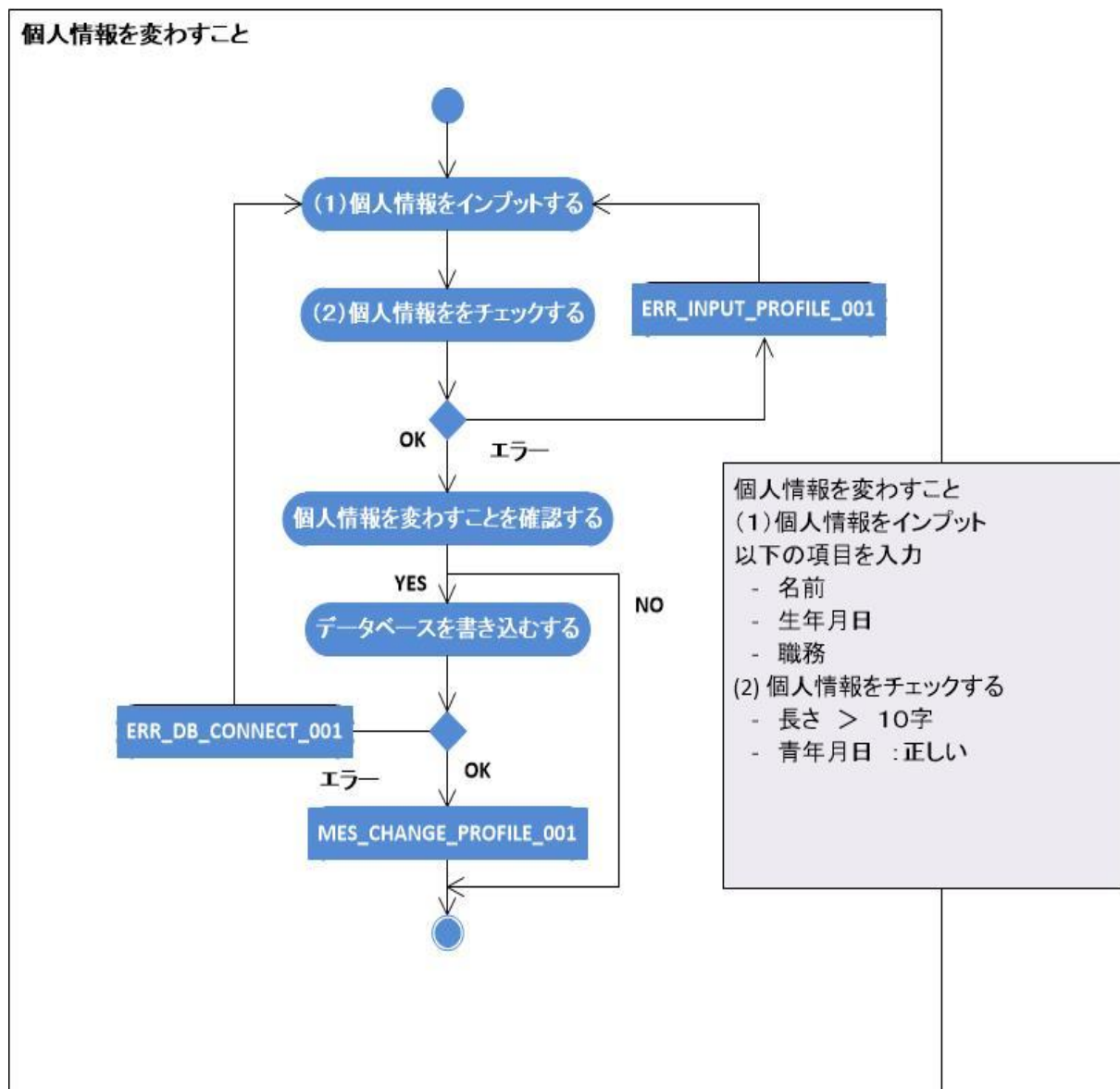


図 3 : 個人情報変更アクティビティ

### 1.3. パスワードリセット

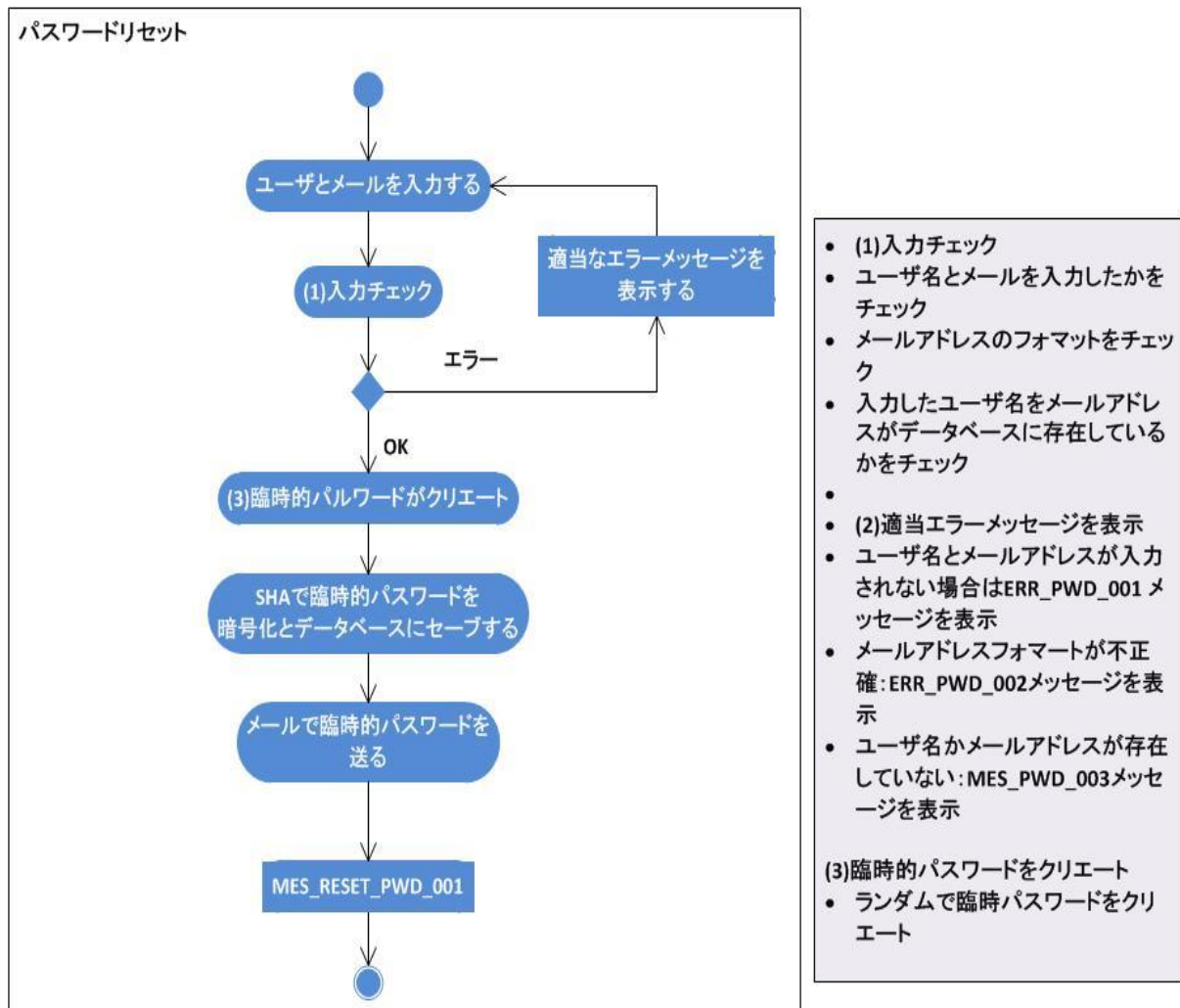


図 4 : パスワードリセットアクティビティ

## 2. テストモジュール

### 2.1 解答者にテストの情報を送る

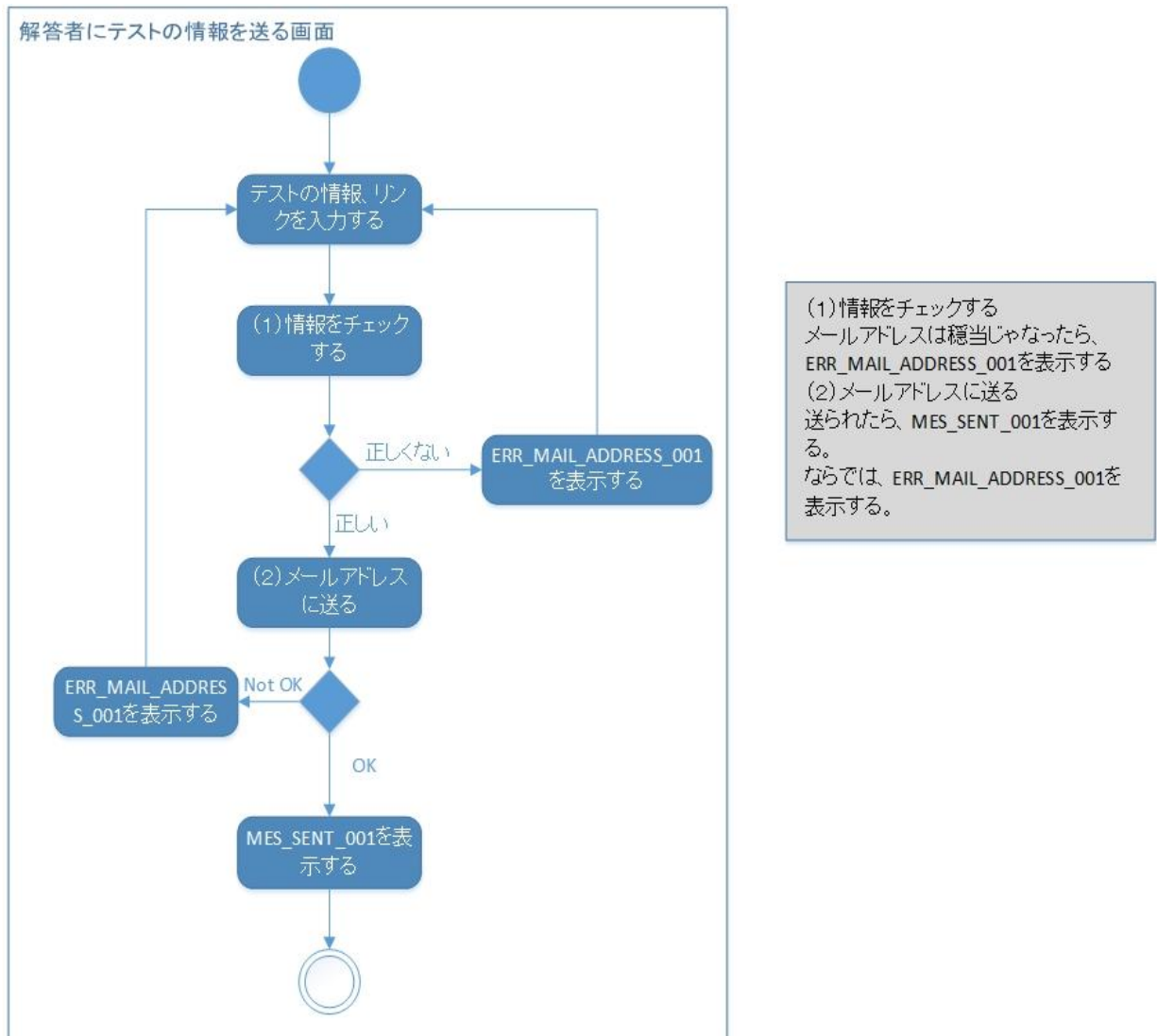


図 5： 解答者にテストの情報を送るアクティビティ

## 2.2 テスト問題を作成する

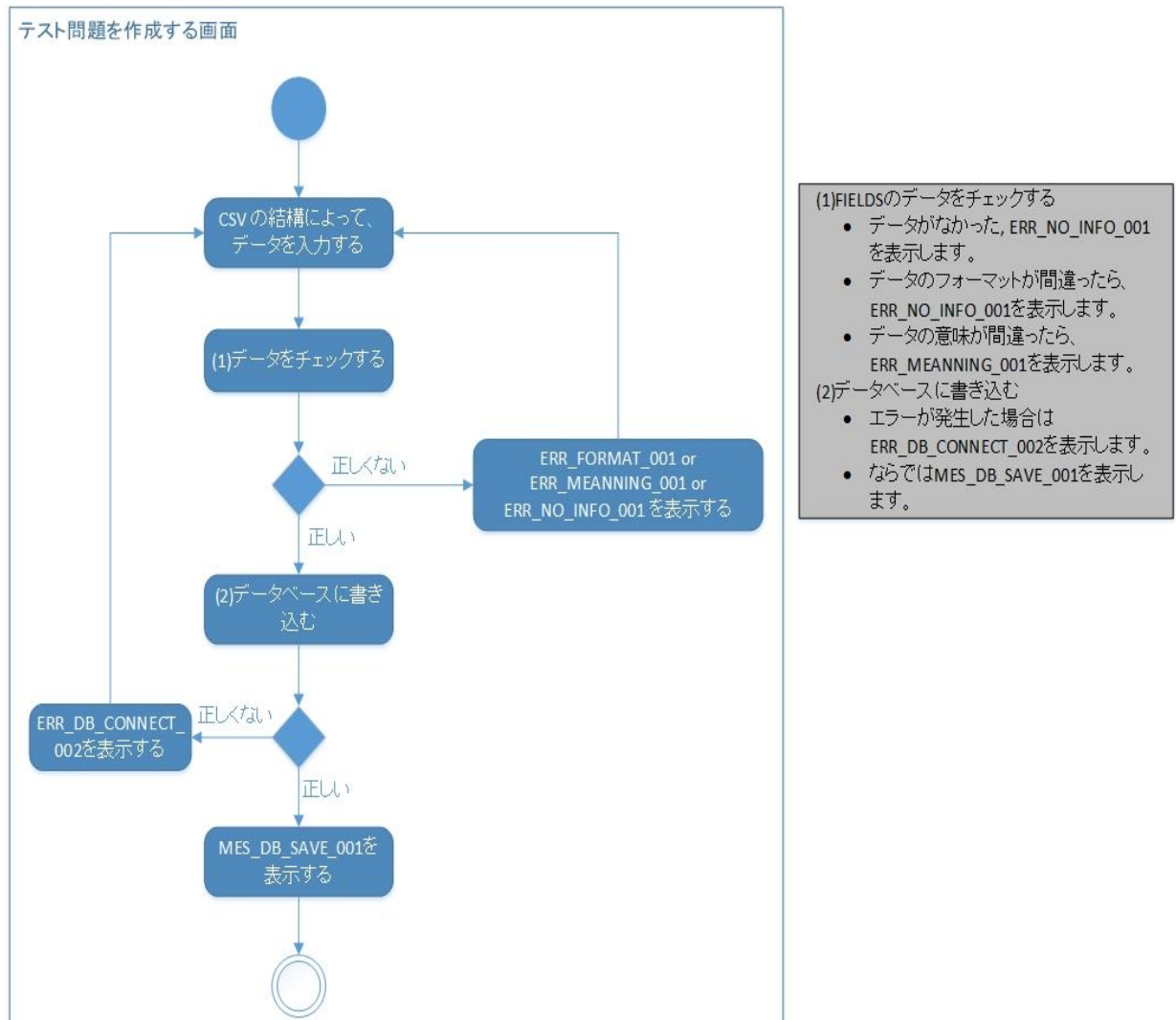


図 6 : テスト問題を作成するアクティビティ

## 2.3. テストの選択

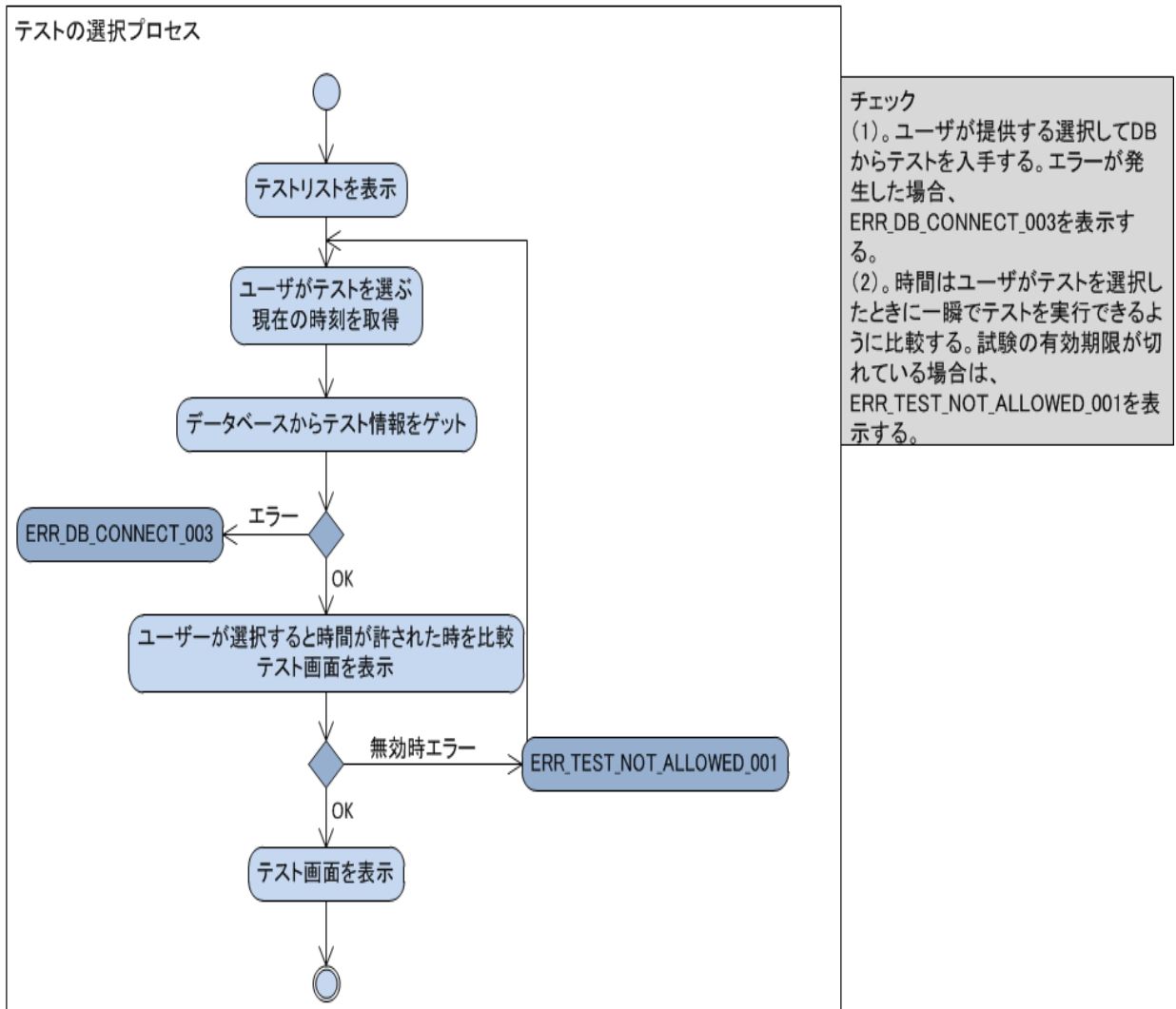


図 7：テストの選択アクティビティ

## 2.4. テスト画面

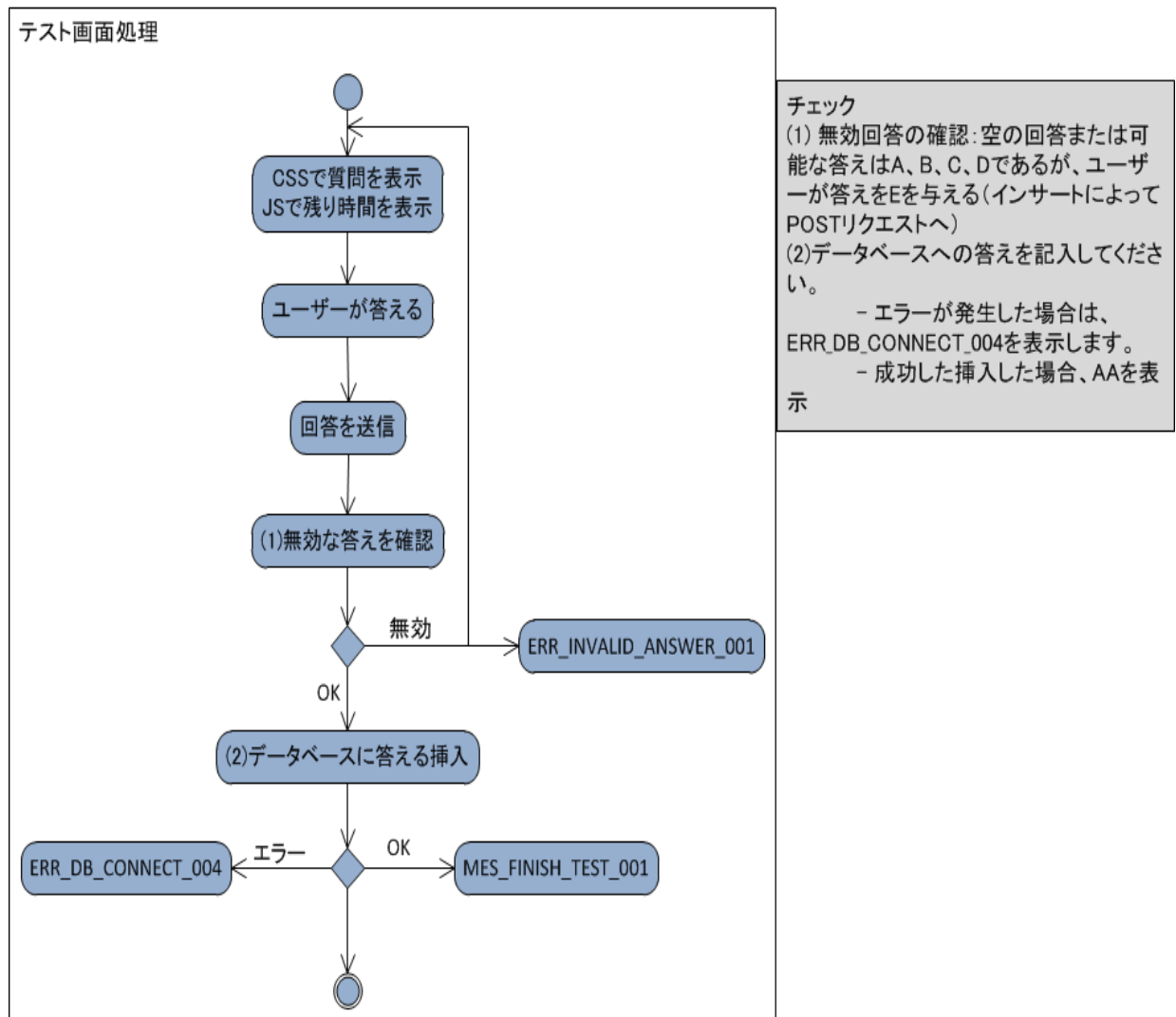


図 8 : テスト画面アクティビティ

## 2.5. テスト結果の送信

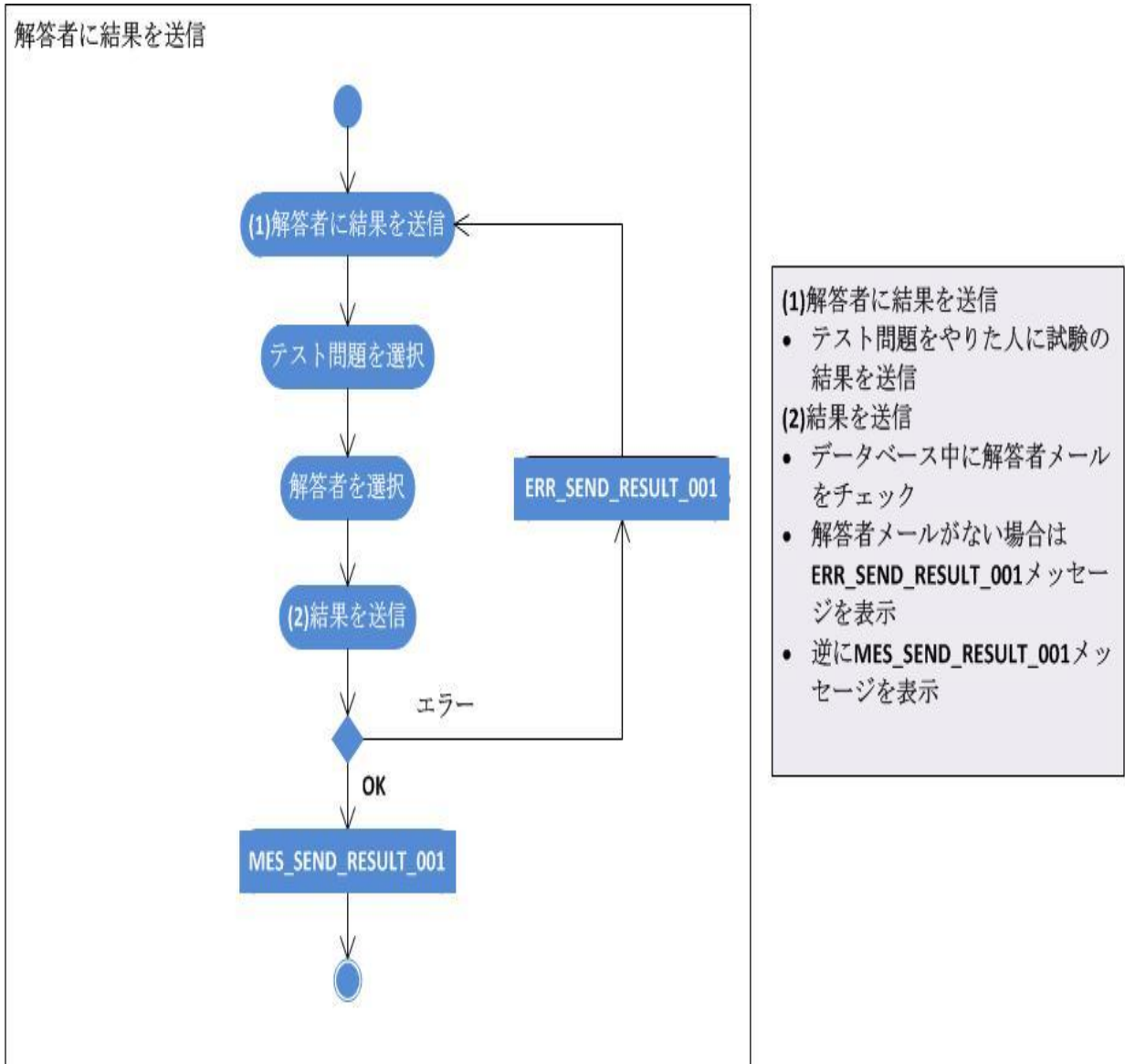


図 9 : テスト結果の送信アクティビティ

## 2.6. テスト問題を採点

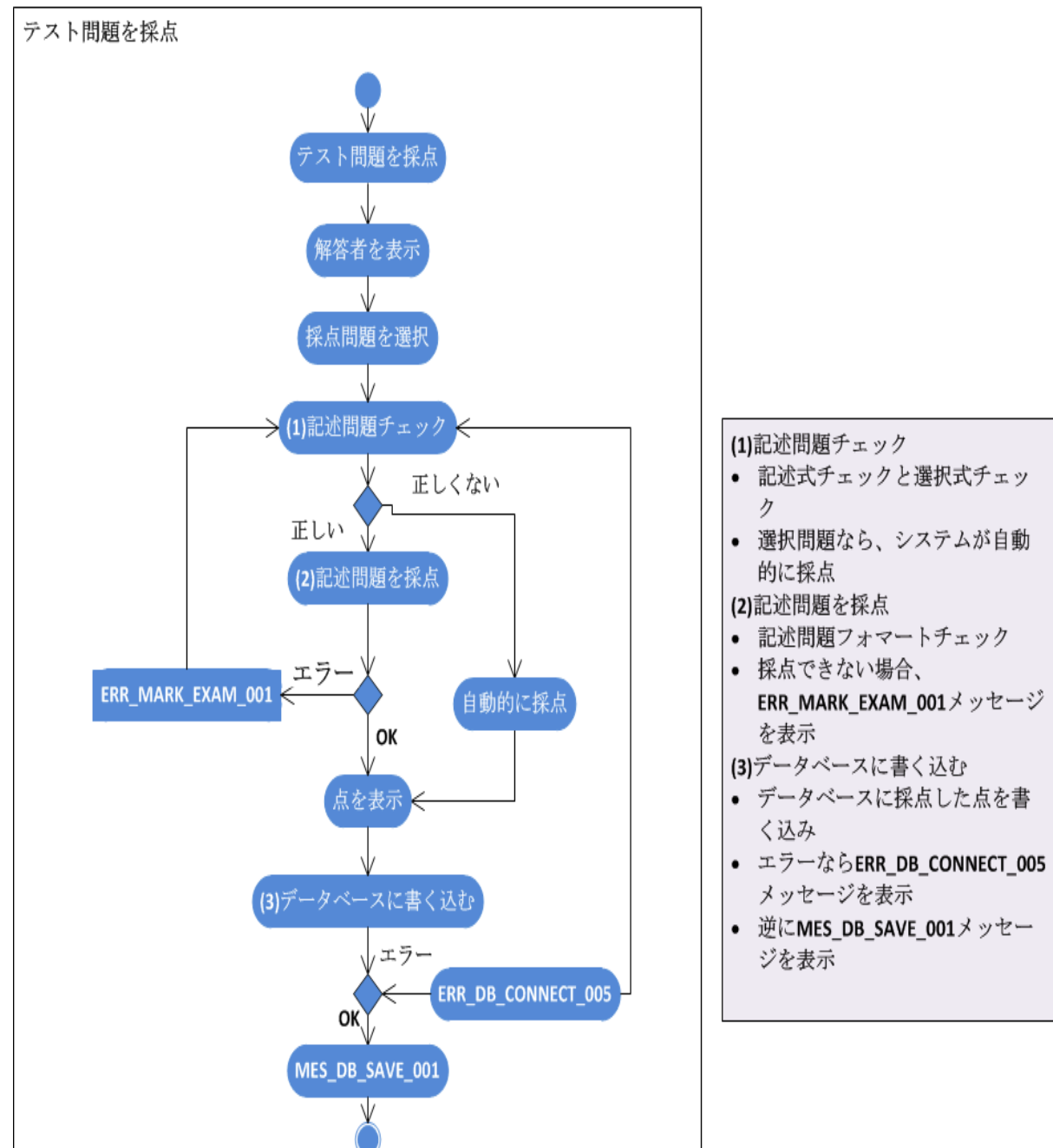


図 10 : テスト問題を採点アクティビティ



### 3. ユーザ管理モジュール

#### 3.1. ユーザ検索

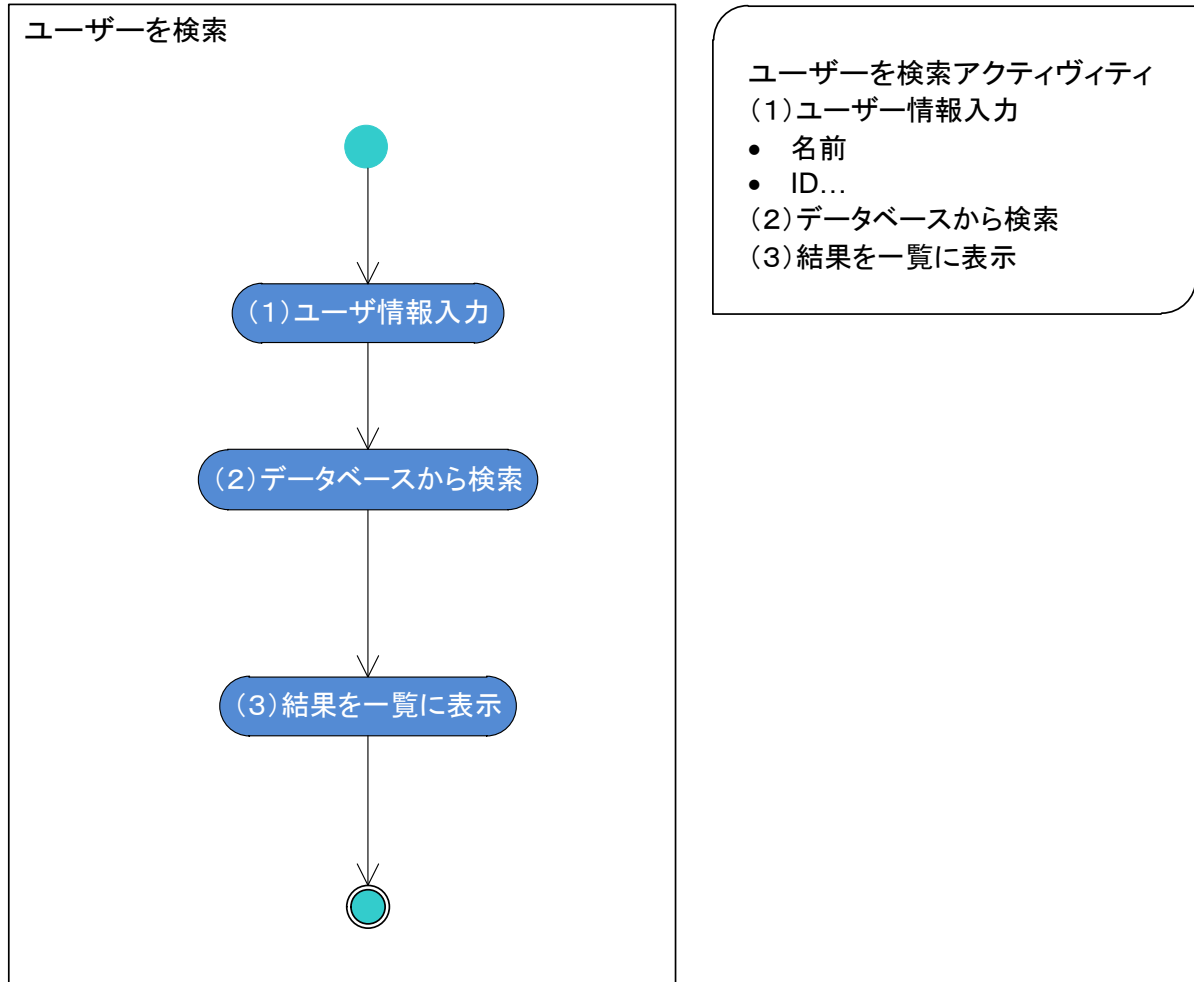


図 11 : ユーザ検索アクティビティ

### 3.2. ユーザ追加

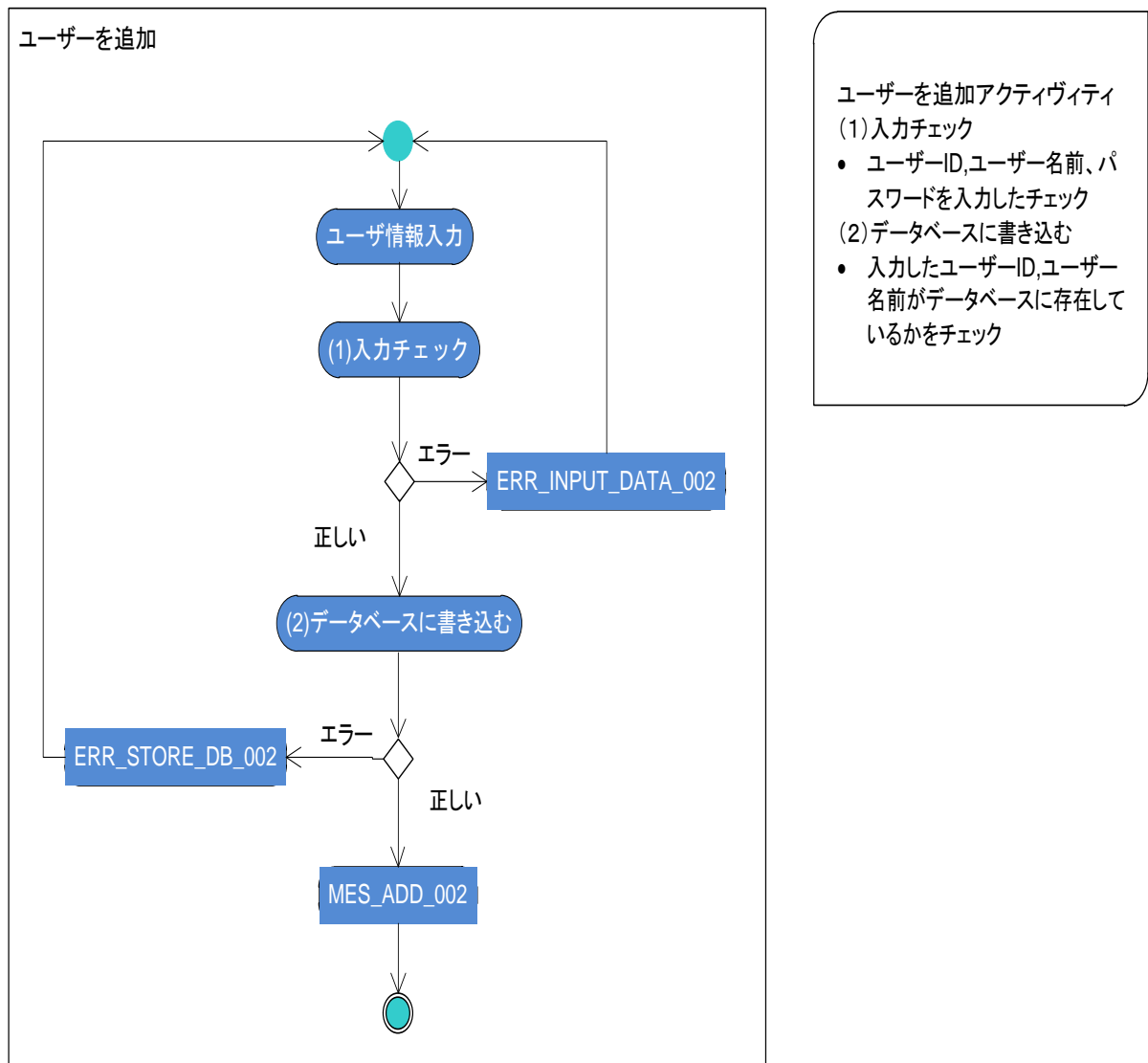


図 12 : ユーザ追加アクティビティ

### 3.3. ユーザ削除

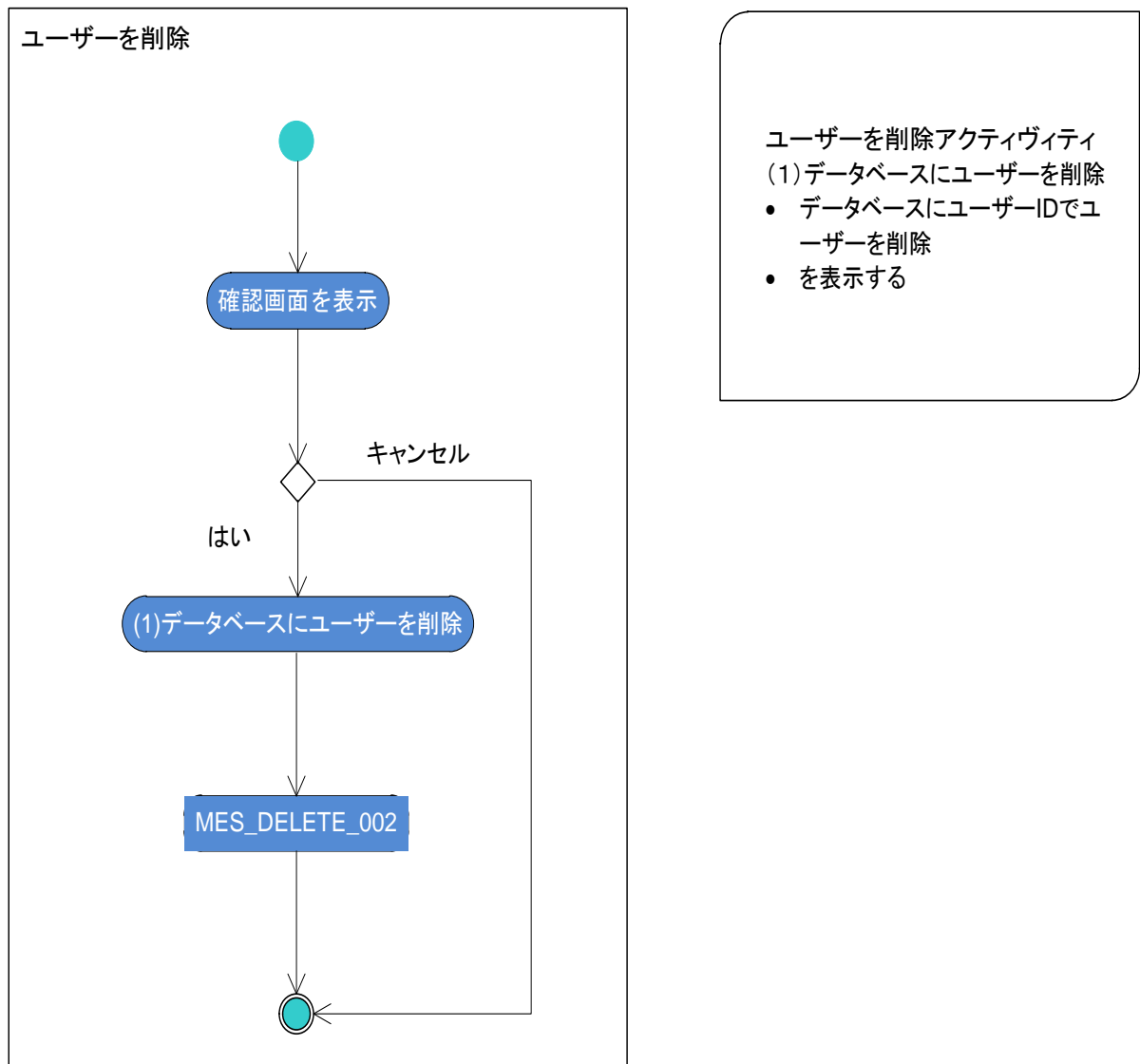


図 13 : ユーザ削除アクティビティ

## 4. 団体者管理モジュール

### 4.1. 一覧団体者を表示

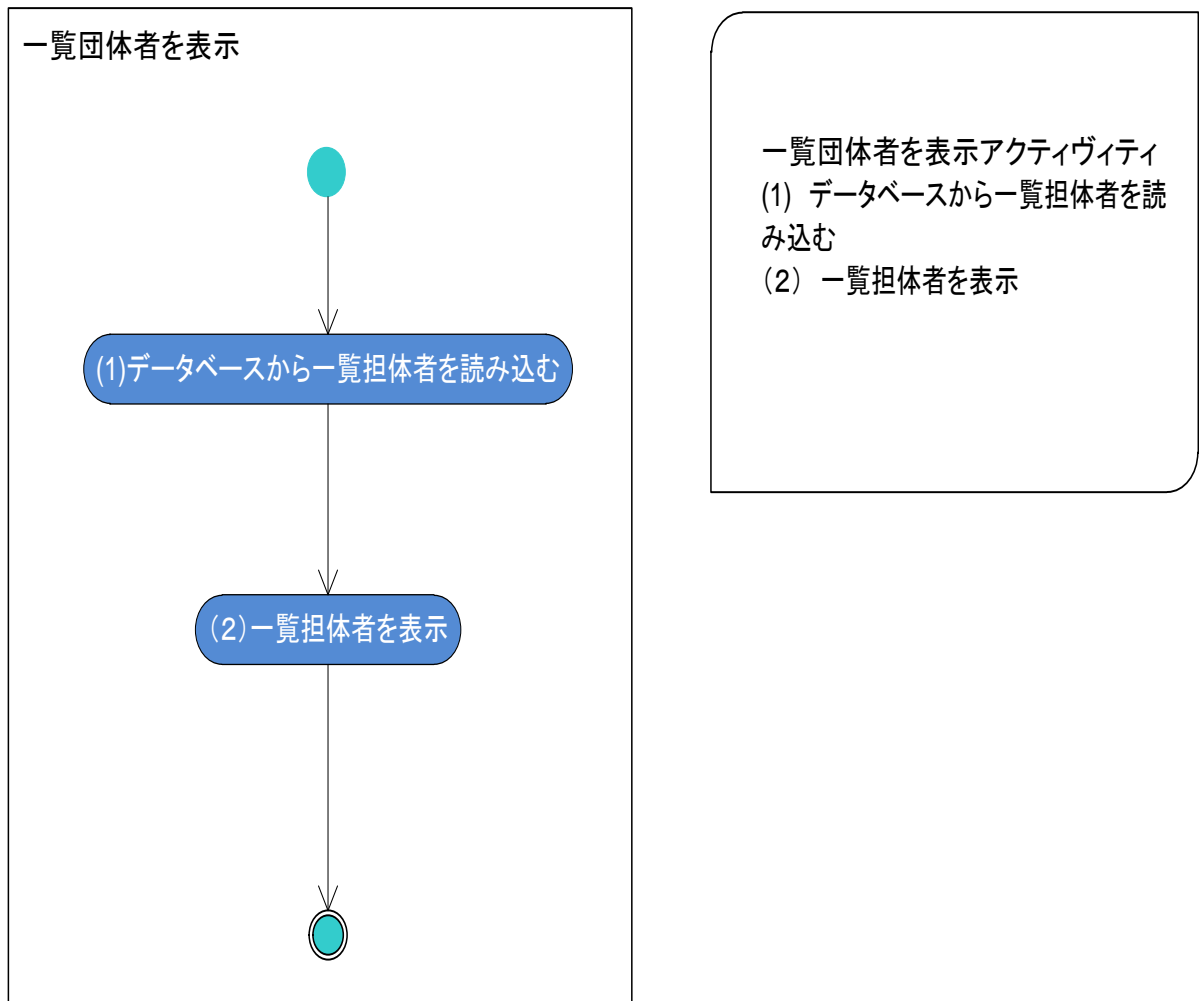


図 14 : 一覧団体者を表示アクティビティ

## 4.2. 団体者検索

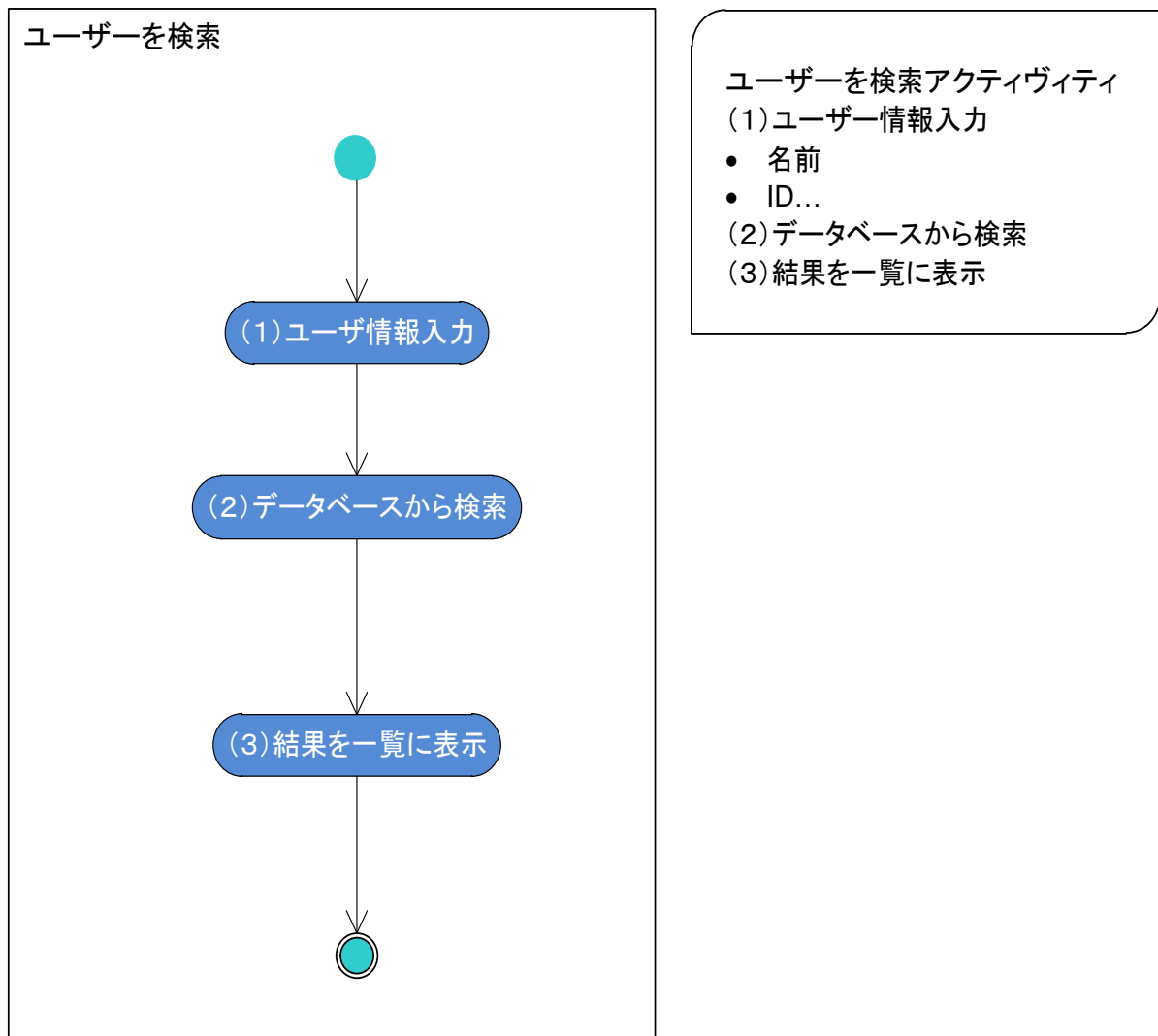


図 15 : 団体者を検索アクティビティ

### 4.3. 団体者追加

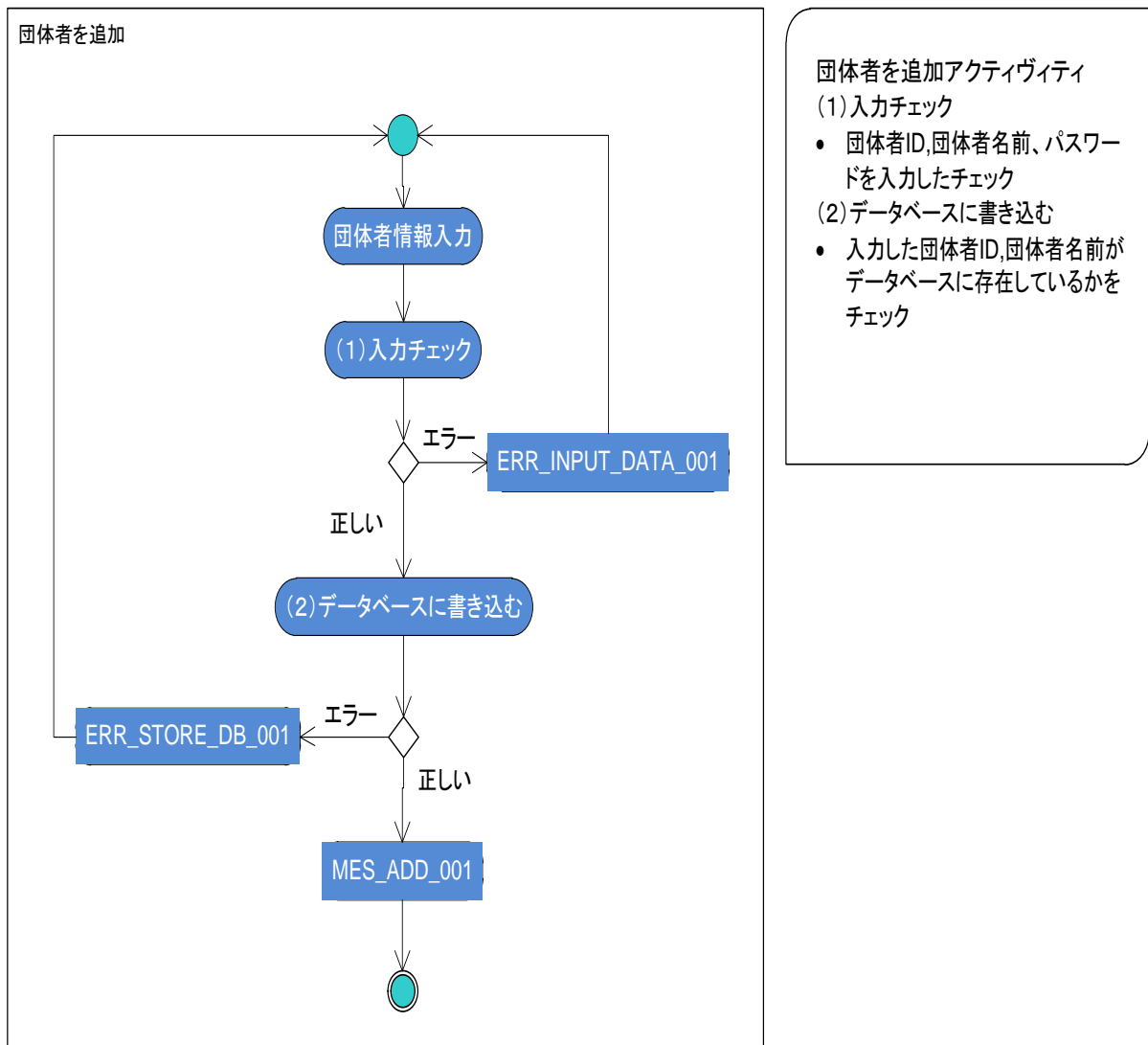


図 16 : 団体者追加アクティビティ

#### 4.4 団体者削除

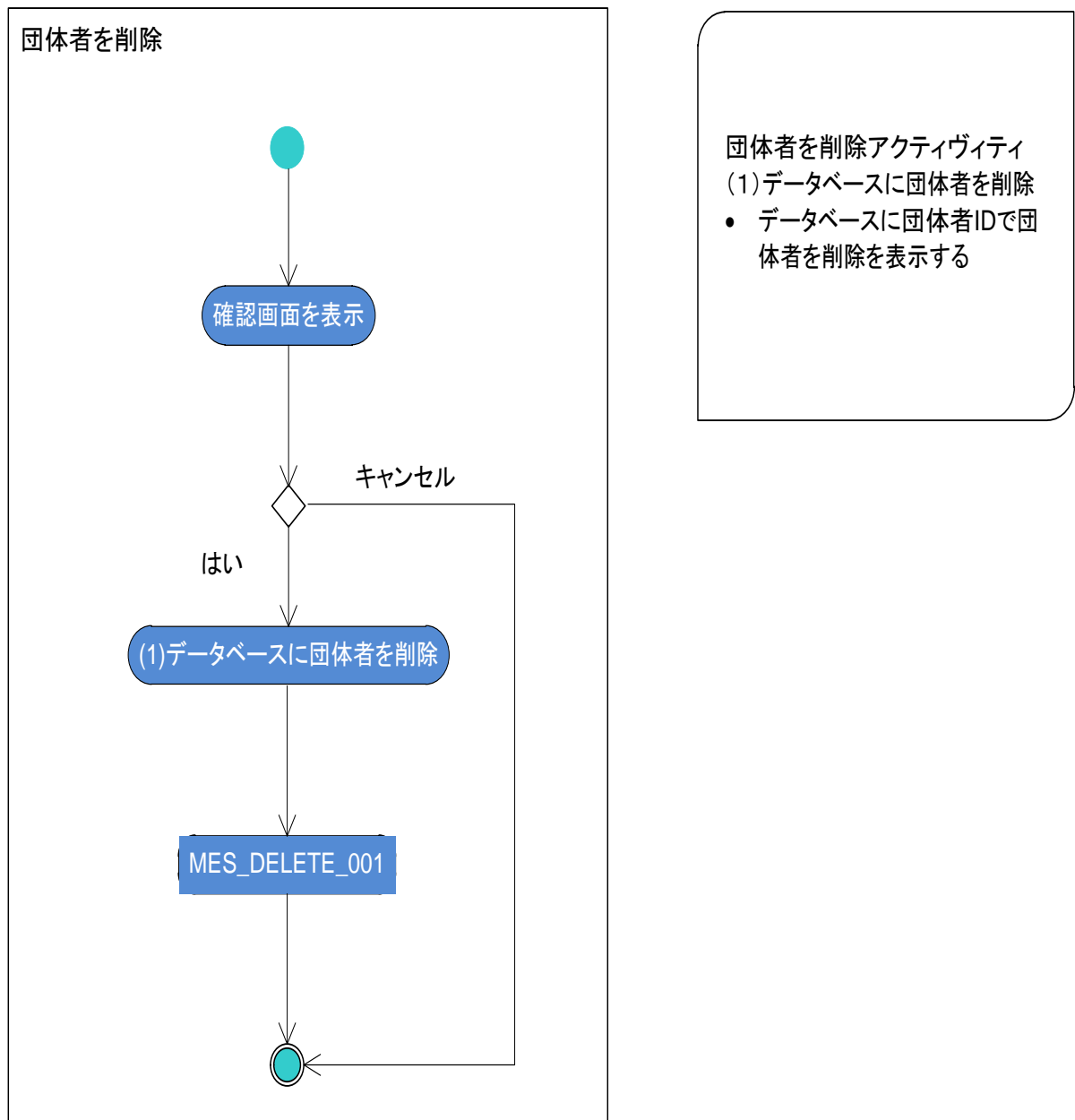


図 17 : 団体者削除アクティビティ

## 5. お知らせ管理モジュール

### 5.1 お知らせ変更

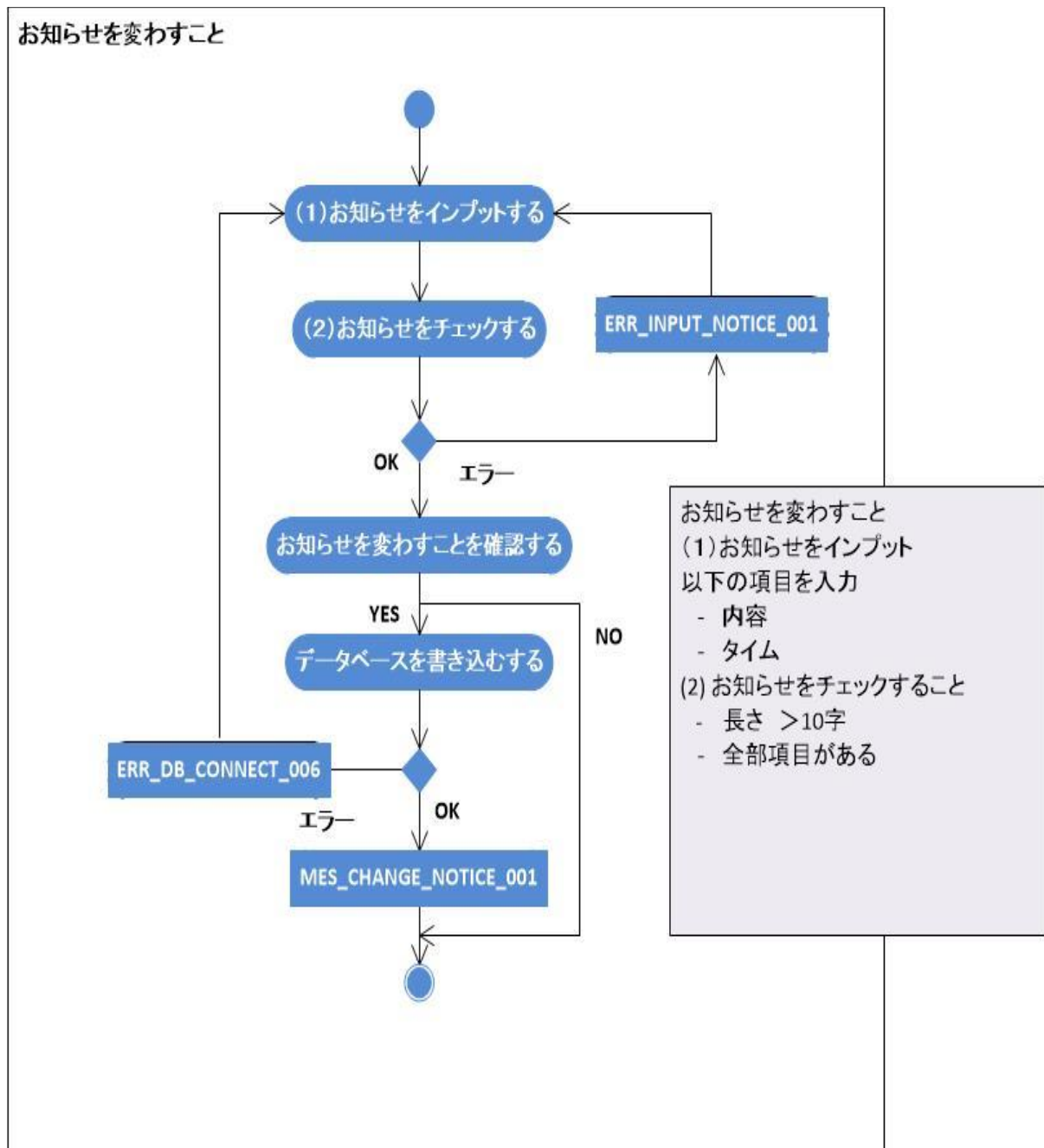


図 18 : 予約参照と状態変更アクティビティ(管理者)



## 5.2 お知らせ追加

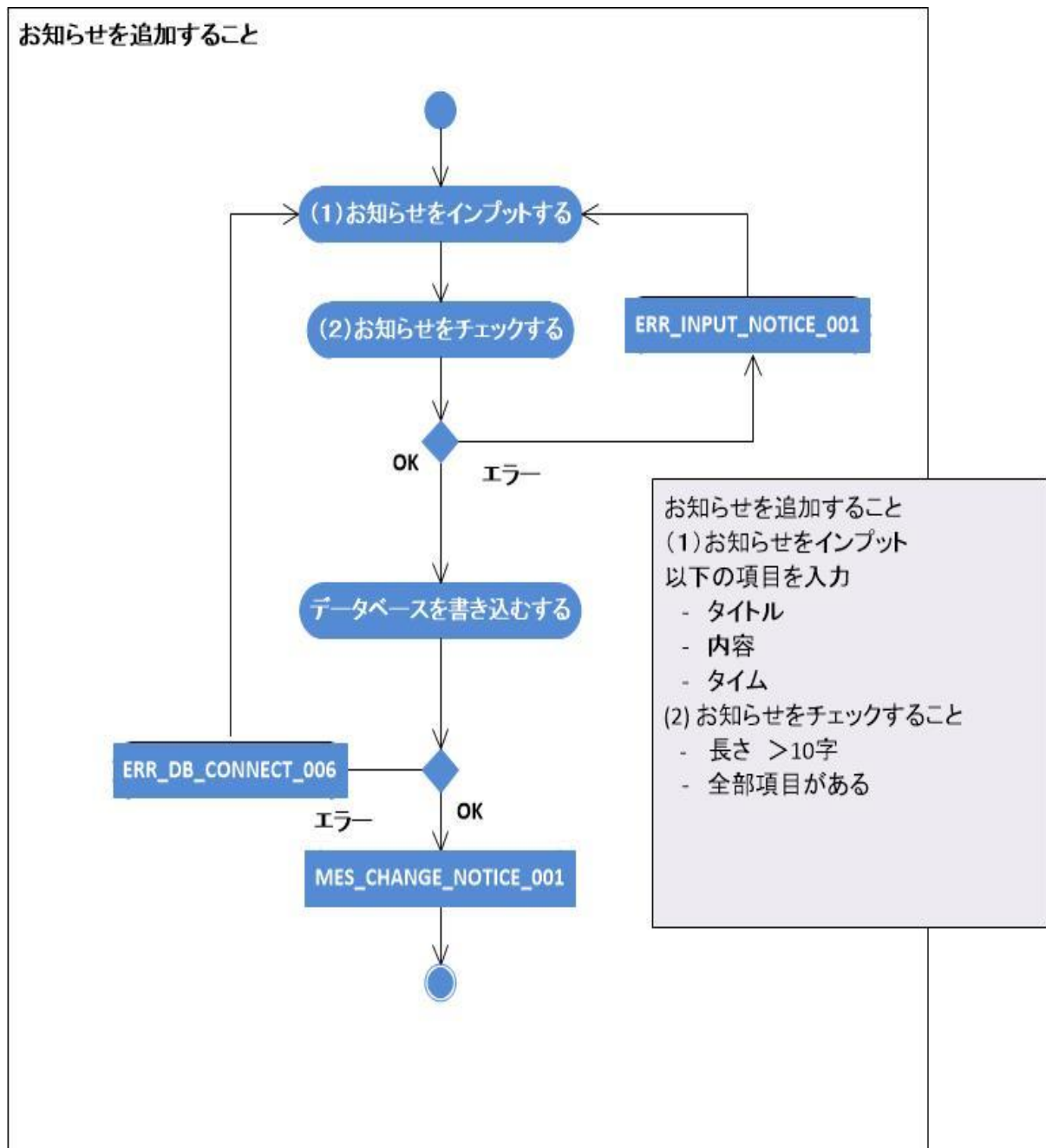


図 19： お知らせ追加アクティビティ

### 5.3 お知らせ削除

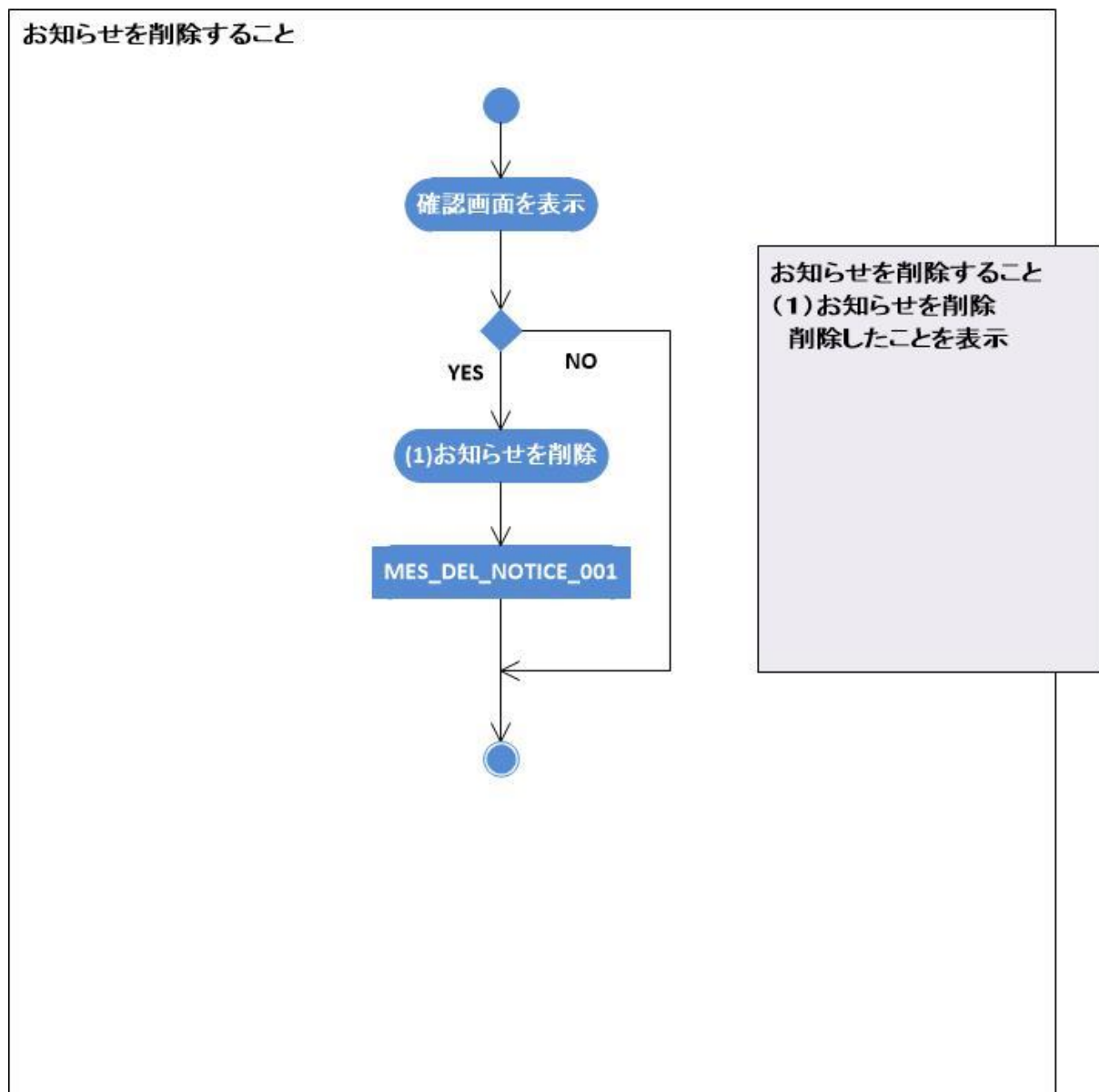


図 20 : お知らせ削除アクティビティ

### III. メッセージ設計

コード	タイプ	内容	チェックレベル	画面
ERR_LOGIN_001	エラー	パスワードが不正確です。	意味レベル	ログイン
ERR_LOGIN_002	エラー	無効なユーザー名です。	意味レベル	
ERR_INPUT_PROFILE_001	エラー	入力したデータがエラーです。	形式レベル	個人情報変更
ERR_DB_CONNECT_001	エラー	個人情報を変わずために、データベースに接続することはできません。	記銘レベル	
MES_CHANGE_PROFILE_001	通知	個人情報を変えました。	形式レベル	
ERR_PWD_001	エラー	ユーザ名かメールアドレスが入力されていません。	意味レベル	パスワード再設定
ERR_PWD_002	エラー	メールアドレスが不正確。	形式レベル	
ERR_PWD_003	エラー	メールアドレスとユーザ名のカップルがデータベースに存在しません。	意味レベル	
MES_RESET_PWD_001	通知	パスワードを送信いたしました。		
ERR_MAIL_ADDRESS_001	エラー	もう一度メールアドレスを入力してください。	形式レベル	解答者にテストの情報

MES_SENT_001	通知	メールアドレスに送りました。		を送る
ERR_FORMAT_001	エラー	データのフォーマットが間違っています。 もう一度入力してください。	形式レベル	テスト問題 を作成
ERR_MEANNING_001	エラー	データの意味が間違っています。もう一度 入力してください。	意味レベル	
ERR_NO_INFO_001	エラー	データがありません。入力してくださ い。	形式レベル	
ERR_DB_CONNECT_002	エラー	テストを作成する情報を書く時、デー タベースに接続できません。		
ERR_DB_CONNECT_003	エラー	試験を取得するために、データベース に接続できません。		テストの選 択
ERR_TEST_NOT_ALLOWED_010	エラー	その時点でテストを行うことは許され ません。	意味レベル	
ERR_INVALID_ANSWER_001	エラー	答えは無効です。	意味レベル	テスト画面
ERR_DB_CONNECT_004	エラー	答えを提出するために、データベース に接続することはできません。		
MES_FINISH_TEST_001	通知	テストを終えた。		
MES_SEND_RESULT_001	通知	解答者にテスト結果を送りました。		解答者に結

ERR_SEND_RESULT_001	エラー	テスト結果を送れません。	意味レベル	果を送信
ERR_MARK_EXAM_001	エラー	技術問題が採点できません。	形式レベル	テスト問題を採点
ERR_DB_CONNECT_005	エラー	データベースに接続できません。	意味レベル	
MES_DB_SAVE_001	通知	データベースに採点した点が登録されました。		
ERR_INPUT_DATA_002	エラー	入力したデータがエラーです。	形式レベル	ユーザ管理
ERR_STORE_DB_002	エラー	データベースにデータを格納できません。	記銘レベル	
MES_ADD_002	通知	ユーザーを追加しました。		
MES_DELETE_002	通知	ユーザーを削除しました。		
ERR_INPUT_DATA_001	エラー	入力したデータがエラーです。	形式レベル	団体者管理
ERR_STORE_DB_001	エラー	データベースにデータを格納できません。	記銘レベル	
MES_ADD_001	通知	団体者を追加しました。		
MES_DELETE_001	通知	団体者を削除しました。		

ERR_INPUT_NOTICE_001	エラー	入力したデータがエラーです	形式レベル	お知らせ管理
ERR_DB_CONNECT_006	エラー	お知らせを追加するために、データベースに接続することはできません。	記銘レベル	
MES_CHANGE_NOTICE_001	通知	お知らせを変えました	形式レベル	
MES_ADD_NOTICE_001	通知	お知らせを追加しました	形式レベル	
MES_DEL_NOTICE_001	通知	お知らせを削除しました	形式レベル	

## IV.ログ設計

### 1. 作成法

#### 1.1. システムが各ログタイプ

システムは三つのログタイプを作成する：

- オペレーションログ
  - システムが動かすためにシステムの操作をログします。
  - 障害が発生した場合は原因を見つけるために使います。
- エラーログ
  - システムがエラーが発生した状態をログします。
  - 障害が発生原因を見つけて、障害の回復のためです。
- デバッグログ
  - 開発者が主動てきに作成するログです。
  - システムがデバッグために使います。

#### 1.2. ログ作成法

システムの全部のログはテキストのファイルに書きます。ログファイルの名前は” .log” で終了します。

管理が簡単にするために、各ログファイルは年と月と日によって保存します。

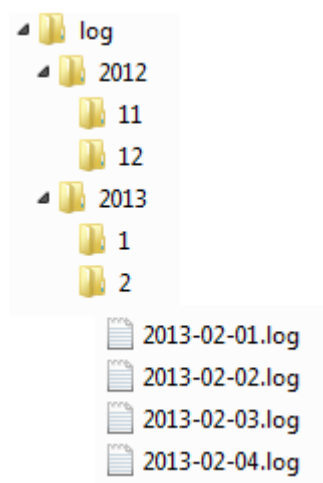


図 21：ログディレクトリ

たとえば、21 図の場合は：

- log というディレクトリはプロジェクトの中に設置します。このディレクトリは各ログファイルを保存します。

- log ディレクトリの中には 2012 ディレクトリと 2013 ディレクトリあります。
  - 2012 ディレクトリ：2012 年のログファイルを保存します。
  - 2013 ディレクトリ：2013 年のログファイルを保存します。
- 2013 ディレクトリは各月ディレクトリがあります
  - 01, 02 のディレクトリは 1 月、2 月のログを保存します。
- 02 ディレクトリは 2 月の各ログファイルがあります。
  - 一つのファイルは一日のログを保存します。
  - ファイル名のフォーマット：YYYY-MM-DD.log

### 1.3. ログを保存する時間

テストを解答の管理とテストを採点の管理は一年間に繰り替えするので、一年前のログを保存することは必要です。つまり去年と今年のログだけを保存します。

## 2. ログレコードの一覧表

### 2.1. ログインモジュールのログ

#### ❖ ログインログ

- タイプ：オペレーションログ
- 書く時：システムにユーザがログインする時
- 形：
 

```
OPERATION:user_name がシステムにログインしました:log_time
```

  - user\_name:ユーザのログイン名
  - log\_time:ユーザが追加時間 (YYYY/MM/DD HH:MM:SS)

例：OPERATION:bongis がシステムにログインしました:2013/02/23 21:01:11

#### ❖ ログインエラーログ

- タイプ：エラーログ
- 書く時：システムにユーザがログインできない場合
- 形：
 

```
ERROR:user_name ログインできません:cause:log_time
```

  - user\_name:ユーザのログイン名
  - cause:ログインエラーの原因
  - log\_time: ユーザがパスワードリセットする時間  
(YYYY/MM/DD HH:MM:SS)



例：ERROR:bongis ログインできません:パスワードが間違い:2013/02/24  
09:12:01

#### ❖ 個人情報を変更ログ

- タイプ：オペレーションログ
- 書く時：ユーザの個人情報を変更時
- 形：

OPERATION:(title,userID)の個人情報を変更した:log\_time

- userID: 個人情報を作った人
- password: userID の パスワード
- name : userID の名前
- birthday : userID の誕生日
- 職務 : userID の仕事
- log\_time:個人情報を変更時間 (YYYY/MM/DD HH:MM:SS)

#### ❖ 個人情報を変更エラーログ

- タイプ：エラーログ
- 書く時：個人情報を変更時場合、エラーが発生する時
- 形：

OPERATION:(title,userID)の個人情報を変更エラーした:log\_time

- userID: 個人情報を作った人
- password: userID の パスワード
- name : userID の名前

#### ❖ パスワードリセットログ

- タイプ：オペレーションログ
- 書く時：ユーザがパスワードリセットする時
- 形：

OPERATION:user\_name がパスワードリセットを要求した:log\_time

- user\_name:ユーザのログイン名
- log\_time: ユーザがパスワードリセットする時間  
(YYYY/MM/DD HH:MM:SS)

例：OPERATION:bongis がパスワードリセットを要求した:2013/02/21  
10:12:11

#### ❖ パスワードリセットエラーログ

- タイプ：エラーログ
- 書く時：ユーザがパスワードリセットできない時
- 形：

ERROR:user\_name のパスワードをリセットできない:cause:log\_time

- user\_name: ユーザのログイン名
- cause: パスワードリセットエラーの原因
- log\_time: ユーザがパスワードリセットする時間  
(YYYY/MM/DD HH:MM:SS)

例: ERROR:namlt のパスワードをリセットできない:namlt というユーザ名が存在していない:2013/02/23 08:11:23

## 2.2. テストモジュールのログ

### ❖ 解答者にテストの情報を送るログ

- タイプ: オペレーションログ
- 書く時: 解答者にテストの情報を送る時
- 形:

OPERATION:(senderId, startExaminationTime, endExaminationTime,toMailAddress)の解答者にテストの情報を送るした:log\_time

- id: 会議室の ID
- startExaminationTime: テスト時間のスタート
- endExaminationTime: テスト時間のエンド
- toMailAddress: 解答者のメールアドレス
- log\_time: 送る時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION:(1, 09:45, 11:45, tuongthenghia@gmail.com)の情報を送るした:2013/02/03 20:45:23

### ❖ 解答者にテストの情報を送るエラーログ

- タイプ: エラーログ
- 書く時: 解答者にテストの情報を送らない時
- 形:

ERROR:(senderId, startExaminationTime, endExaminationTime,toMailAddress)の解答者にテストの情報を送りませんでした: cause : log\_time

- id: 会議室の ID
- startExaminationTime: テスト時間のスタート
- endExaminationTime: テスト時間のエンド
- toMailAddress: 解答者のメールアドレス
- cause: エラーが発生する原因
- log\_time: 送らない時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR:(1, 09:45, 11:45, tuongthenghia@gmail.co)の情報を送りませんでした: メールアドレスのエラー: 2013/02/03 20:45:23

#### ❖ テスト問題を作成するログ

- **タイプ:**オペレーションログ
- **書く時:** テスト問題を作成する時
- **形:**

OPERATION: (teacherId、テスト Id) のテスト問題を作成した:log\_time

- teacherId:作成する人
- テスト Id: テストの Id
- log\_time:作成する時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION: (1,23 ) のテスト問題を作成した:2013/02/03 20:45:23

#### ❖ テスト問題を作成するエラーログ

- **タイプ:**エラーログ
- **書く時:** テスト問題を作成するエラー時
- **形:**

OPERATION: (teacherId、テスト Id) のテスト問題を作成できませんでした:  
cause:log\_time

- teacherId:作成する人
- テスト Id: テストの Id
- log\_time:作成しない時間 (YYYY/MM/DD HH:MM:SS)
- cause:エラーが発生する原因

例: OPERATION: (1,23 ) のテスト問題を作成できませんでした:FIELD のエラー  
2013/02/03 20:45:23

#### ❖ ユーザがテストを選択したときにログ

- **タイプ:**オペレーションログ
- **書く時:** ユーザーがリストからテストを選択する
- **形:**

OPERATION(test\_id,examiner\_id,marker\_id,email)は  
user\_idによって選ばれた: log\_time.

- test\_id:テストの ID
- examiner\_id: 出題者の ID
- marker\_id: 採点者の ID
- email: システムに解答者が登録しておいたメール
- user\_id: 解答者の ID
- log\_time: テストが開始時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION(Mat17022013, E29, M177, [windytime@gmail.com](mailto:windytime@gmail.com))  
は HUST20082913 によって選ばれた:2013/02/17 16:40:50

❖ ユーザーが無効な時にテストを取得するときにエラーログ

- **タイプ:**エラーログ
- **書く時:** ユーザが期限切れのテストを選択したとき
- **形:**

ERROR(test\_id,examiner\_id,marker\_id,email)は満了時に user\_id によって選ばれた : log\_time.

- test\_id:テストの ID
- examiner\_id: 出題者の ID
- marker\_id: 採点者の ID
- email: システムに解答者が登録しておいたメール
- user\_id: 解答者の ID
- log\_time: テストが開始時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR(Mat17022013, E29, M177, [windytime@gmail.com](mailto:windytime@gmail.com)) は満了時に HUST20082913 によって選ばれた:2013/02/17 16:40:50

❖ ユーザがテストを受けるが、システムがテストを取得できないときにエラーログ

- **タイプ:**エラーログ
- **書く時:** ユーザがテストを受けるが、システムがテストを取得できないとき
- **形:**

ERROR: user\_id のための (test\_id,examiner\_id,marker\_id,email)のテストを取得することはできない:log\_time.

- test\_id:テストの ID
- examiner\_id: 出題者の ID
- marker\_id: 採点者の ID
- email: システムに解答者が登録しておいたメール
- user\_id: 解答者の ID
- log\_time: テストが開始時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR: HUST20082913 のための (Mat17022013, E29, M177, [windytime@gmail.com](mailto:windytime@gmail.com)) のテストを取得することはできない: 2013/02/17 16:40:50

❖ ユーザーがテストを終了したときログ

- **タイプ:**オペレーションログ
- **書く時:** ユーザーが終了し、その結果を提出するとき

▪ **形:**

user\_id が (test\_id, examiner\_id, marker\_id, email) のテストを提出する : log\_time.

- test\_id: テストの ID
- examiner\_id: 出題者の ID
- marker\_id: 採点者の ID
- email: システムに解答者が登録しておいたメール
- user\_id: 解答者の ID
- log\_time: テストが開始時間 (YYYY/MM/DD HH:MM:SS)

例: HUST20082913 が (Mat17022013, E29, M177, [windytime@gmail.com](mailto:windytime@gmail.com)) のテストを提出する: 2013/02/17 16:40:50

❖ ユーザがテストを提出エラーログ

▪ **タイプ:** エラーログ

- **書く時:** ユーザがテストを提出するが、システムはデータベースに接続できないとき

▪ **形:**

ERROR: user\_id の (test\_id, examiner\_id, marker\_id, email) のテストを投入することはできない : log\_time

- test\_id: テストの ID
- examiner\_id: 出題者の ID
- marker\_id: 採点者の ID
- email: システムに解答者が登録しておいたメール
- user\_id: 解答者の ID
- log\_time: テストが開始時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR: HUST20082913 の (Mat17022013, E29, M177, [windytime@gmail.com](mailto:windytime@gmail.com)) のテストを投入することはできない: 2013/02/17 16:40:50

❖ **結果の送信ログ**

▪ **タイプ:** オペレーションログ

- **書く時:** テストの結果を解答者に送る時

▪ **形:**

OPERATION(user\_id, send\_id, examinee\_id, email) の情報でテスト結果を送信した log\_time

- user\_id: 解答者の ID
- send\_id: 送信者の ID

- `examinee_id`: テスト問題の ID
- `email`: システムに解答者が登録しておいたメール
- `log_time`: 結果を送る時間 (YYYY/MM/DD HH:MM:SS)

例: `OPERATION(HUST20081637,ADMIN123,TEST1,bizongcntt1990@gmail.com)` テスト結果を送信した: 2013/02/01 08:03:12

#### ❖ 結果の送信エラーログ

- **タイプ**: エラーログ
- **書く時**: 解答者のテストの結果を送信できない場合はエラーを発生する時
- **形**:

`ERROR(user_id,send_id,examinee_id,email)` の情報でテスト結果を送信した: `cause:log_time`

- `user_id`: 解答者の ID
- `send_id`: 送信者の ID
- `examinee_id`: テスト問題の ID
- `email`: システムに解答者が登録しておいたメール
- `cause`: エラーが発生した原因
- `log_time`: 結果を送る時間 (YYYY/MM/DD HH:MM:SS)

例: `ERROR(HUST20081637,ADMIN123,TEST1,bizongcntt1990@gmail.com)` テスト結果を送信した: 解答者のメールのフォーマットが正しくない: 2013/02/01 08:03:12

#### ❖ テスト問題の採点ログ

- **タイプ**: オペレーションログ
- **書く時**: テスト問題を採点する時
- **形**:

`OPERATION(user_id,examinee_id,mark)` の情報でテスト問題を採点した `log_time`

- `user_id`: 採点者の ID
- `examinee_id`: テスト問題の ID
- `mark`: 採点した点数
- `log_time`: テスト問題を採点した時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION (MARKER123, TEST1, 100) テスト問題を採点した :  
2013/02/02 08:50:01

#### ❖ テスト問題の採点エラーログ

- タイプ: エラーログ
- 書く時: テスト問題を採点する時、エラーが発生する
- 形:

ERROR (user\_id, examinee\_id, mark) の情報でテスト問題を採点した: cause: log\_time

- user\_id: 採点者の ID
- examinee\_id: テスト問題の ID
- mark: 採点した点数
- cause: エラーが発生した原因
- log\_time: テスト問題を採点した時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION (MARKER123, TEST1, 100) テスト問題を採点した : データベースに接続できない: 2013/02/02 08:50:03

### 2.3. ユーザ管理モジュールのログ

#### ❖ ユーザ検索ログ

- タイプ: オペレーションログ
- 書く時: ユーザを検索時
- 形:

OPERATION: (user\_id, user\_name) の条件でユーザを検索した: result\_count: log\_time

- user\_id: ユーザの ID
- user\_name: ユーザの名前
- result\_count: 検索結果のレコード数
- log\_time: ユーザが検索時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION: (12, nam) のユーザを検索した: 20: 2013/02/28 20: 45:23

#### ❖ ユーザ追加ログ

- タイプ: オペレーションログ
- 書く時: 新しいユーザを追加時
- 形:

OPERATION: (user\_id,user\_name,password,type) のユーザを追加した:log\_time

- user\_id:ユーザの ID
- user\_name: ユーザの名前
- password:ユーザのパスワード
- type: ユーザの種類

例: OPERATION: (12,yutafukuoka ,1313,1) のユーザを追加した:2013/02/28  
20:45:23

#### ❖ ユーザ追加エラーログ

- タイプ: エラーログ
- 書く時: 新しいユーザを追加できない時。
- 形:

ERROR: 新しいユーザの名前を追加できない:cause:log\_time

- user\_name: 新しいユーザの名前
- cause: ユーザ追加エラーの原因
- log\_time: 新しいユーザを追加する時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR:honda のユーザを追加できない:honda というユーザ名が存在する:2013/02/11 08:12:23

#### ❖ ユーザ削除ログ

- タイプ: オペレーションログ
- 書く時: ユーザを削除時
- 形:

OPERATION: (user\_id, user\_name) のユーザを削除した:log\_time

- user\_id:ユーザの ID
- user\_name: ユーザのログイン名
- log\_time:ユーザが削除時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION: (12,yutafukuoka) のユーザを削除した:2013/02/21  
20:45:23

#### ❖ ユーザ削除エラーログ

- タイプ: エラーログ
- 書く時: ユーザを削除できない時。
- 形:

ERROR: ユーザの名前を削除できない:cause:log\_time

- user\_name: ユーザの名前
- cause: ユーザ削除エラーの原因



- log\_time: ユーザを削除する時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR:honda のユーザを削除できない:honda というユーザ名が存在しない:2013/02/11 08:12:23

## 2.4. 団体者管理モジュールログ

### ❖ 団体者検索ログ

- タイプ: オペレーションログ
- 書く時: 団体者を検索時
- 形:

OPERATION:(subadmin\_id, subadmin\_name)の条件でユーザを検索した:result\_count:log\_time

- subadmin\_id: 団体者の ID
- subadmin\_name: 団体者の名前
- result\_count: 検索結果のレコード数
- log\_time: 団体者が検索時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION:(12,nam)の団体者を検索した:20: 2012/02/28 20:45:23

### ❖ 団体者追加ログ

- タイプ: オペレーションログ
- 書く時: 新しい団体者を追加時
- 形:

OPERATION:(subadmin\_id,user\_name,password)の団体者を追加した:log\_time

- subadmin\_id: 団体者の ID
- user\_name: 団体者の名前
- password: 団体者のパスワード

例: OPERATION:(12,yutafukuoka ,1313)の団体者を追加した:2013/02/28 20:45:23

### ❖ 団体者追加エラーログ

- タイプ: エラーログ
- 書く時: 新しい団体者を追加できない時。
- 形:

ERROR: 新しい団体者の名前を追加できない:cause:log\_time

- user\_name: 新しい団体者の名前
- cause: 団体者追加エラーの原因

- log\_time: 新しい団体者を追加する時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR:honda の団体者を追加できない:honda という団体者名が存在する:2013/02/11 08:12:23

## ❖ 団体者削除ログ

- タイプ: オペレーションログ
- 書く時: 団体者を削除時
- 形:

OPERATION:( subadmin\_id, user\_name)の団体者を削除した:log\_time

- subadmin\_id: 団体者の ID
- user\_name: 団体者のログイン名
- log\_time: 団体者が削除時間 (YYYY/MM/DD HH:MM:SS)

例: OPERATION:(12,yutafukuoka)の団体者を削除した:2012/02/28 20:45:23

## ❖ 団体者削除エラーログ

- タイプ: エラーログ
- 書く時: 団体者を削除できない時
- 形:

ERROR: 団体者の名前を削除できない:cause:log\_time

- user\_name: 団体者の名前
- cause: 団体者削除エラーの原因
- log\_time: 団体者を削除する時間 (YYYY/MM/DD HH:MM:SS)

例: ERROR:honda の団体者を削除できない:honda という団体者名が存在しない:2013/02/11 08:12:23

## 2.5 お知らせ管理ログ

### ❖ お知らせを追加ログ

- タイプ: オペレーションログ
- 書く時: 新しいお知らせを追加時
- 形:

OPERATION:(title,userID)のお知らせを追加した:log\_time

- title:お知らせを追加のタイトル
- userID: お知らせを作った人
- log\_time:お知らせを追加時間 (YYYY/MM/DD HH:MM:SS)

#### ❖ お知らせを追加エラーログ

- タイプ：エラーログ
- 書く時：新しいお知らせを追加時場合、エラーが発生する時
- 形：

OPERATION: (title,userID) のお知らせを追加エラーした:log\_time

- title:お知らせを追加のタイトル
- userID: お知らせを作った人
- log\_time:お知らせを追加エラー時間 (YYYY/MM/DD HH:MM:SS)
- cause:エラーが発生した原因

#### ❖ お知らせを変更ログ

- タイプ：オペレーションログ
- 書く時：新お知らせを変更時
- 形：

OPERATION: (title,userID) のお知らせを変更した:log\_time

- title:お知らせを追加のタイトル
- userID: お知らせを作った人
- log\_time:お知らせを変更時間 (YYYY/MM/DD HH:MM:SS)

#### ❖ お知らせを変更エラーログ

- タイプ：エラーログ
- 書く時：お知らせを変更時場合、エラーが発生する時
- 形：

OPERATION: (title,userID) のお知らせを変更エラーした:log\_time

- title:お知らせを変更のタイトル
- userID: お知らせを作った人
- log\_time:お知らせを変更エラー時間 (YYYY/MM/DD HH:MM:SS)
- cause:エラーが発生した原因

#### ❖ お知らせを削除ログ

- タイプ：オペレーションログ
- 書く時：お知らせを削除時
- 形：

OPERATION: (title,userID) のお知らせを削除した:log\_time

- title:お知らせを削除のタイトル
- userID: お知らせを作った人
- log\_time:お知らせを削除時間 (YYYY/MM/DD HH:MM:SS)

#### ❖ お知らせを削除エラーログ

- タイプ：エラーログ

- 書く時：お知らせを削除時場合、エラーが発生する時

- 形：

OPERATION:(title,userID)のお知らせを削除エラーした:log\_time

- title:お知らせを削除のタイトル
- userID: お知らせを作った人
- log\_time:お知らせを削除エラー時間 (YYYY/MM/DD HH:MM:SS)
- cause:エラーが発生した原因

## 2.6. デバッグログ

- タイプ：デバッグログ
- 書く時：開発者がデバッグしたい場合
- 形：

DEBUG:class\_name:function\_name:content:log\_time

- class\_name:クラス名
- function\_name:ファクション名
- content:デバッグ内容
- log\_time:ログを書いた時間 (YYYY/MM/DD HH:MM:SS)

例：DEBUG:User:add:check 123:2012/02/28 20:45:23

## V.例外設計

### 1. 二つのユーザシステムにログインの例外

- 概要：同時に二つのユーザ名がシステムにログインしたことはセキュリティの問題です。
- 解決法：後にログインするユーザはシステムに入れません。

### 2. テストを解答するとき、インターネットをコネクションできない例外

- 概要：テストを解答するとき、インターネットをコネクションできないことテストの結果の問題です。
- 解決法：後にこのときまでのテストの解答をデータベースに格納して、それから、インターネットをコネクションできるときに解答者が解答続けます。

### 3. 採点者がテストを採点するとき、インターネットをコネクションできない例外

- 概要：採点者がテストを採点するとき、インターネットをコネクションできないことテストの採点の問題です。

- 解決法：後にこのときまでのテストの解答の採点をデータベースに格納して、それから、インターネットをコネクションできるときに採点者が採点続けます。

## VI. 定数・コード設計

### 1. 定数設計

価値	記述
REQUEST_TIMEOUT = 3000	クライアントがサーバにリクエストしてから、その時間が過ぎて RESPOND に受けなと TIMEOUT になります。(ミリー秒)
MAX_IN_A_TEST = 300	最大の解答者がテストを受けられます。
LATEST_TIME_ALLOW_TO_GET_TEST = 20	テストが始まってから、遅く時間以上はテストを受けられません。(分)
MAX_TIME_TO_DO_TEST = 180	テストの時間の最大です。(分)
LATEST_TIME_ALLOW_TO_SUBMIT = 3	テストが終わったら、その時間が過ぎて、テストの答えを送ることができません。
STATUS_DOING_TEST	テスト問題をしている状態
STATUS_TEST_COMPLETED	テスト問題が終了した状態
STATUS_SEND_TEST_SUCCESS	テスト問題が終わったら、解答者がサーバに送った状態
TRY_LOGIN_SYSTEM = 3	ユーザが3回以下システムにログインできます。その以外はログインできません。

### 2. コード設計

コード	例	記述
ユーザコード	1, 2, 3 ...	ユーザコードは整数です。重複が起こないように自動的に増加知ることを設計しました。
サブ Admin コード	20072018...	現在 Admin の ID をシステムが配ります。
テストコード	1, 2, 3 ...	テストコードは整数です。重複が起こないように自動的に増加知ることを設計しました。

