

UML講義 2日目

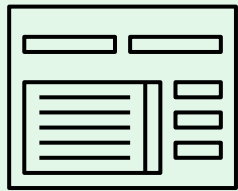
神戸大学

まつ本 真佑 中村 匡秀 佐伯 幸郎
山本 晋太郎 高橋 昂平 大櫛 章裕

復習: ICONIXプロセスの流れ*

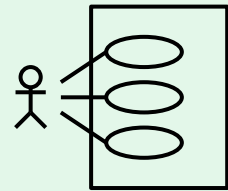
2日目(来週)

夏合宿



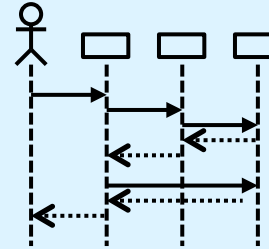
紙芝居

1日目
(今日)

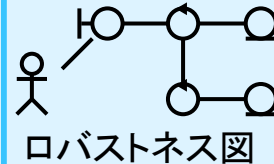


ユースケース図

動的

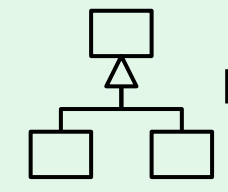


シーケンス図

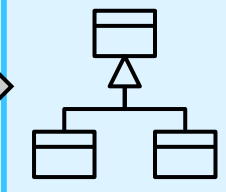


ロバストネス図

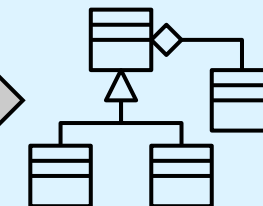
静的



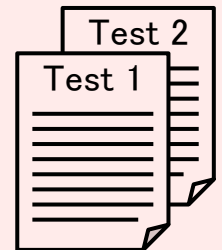
ドメインモデル



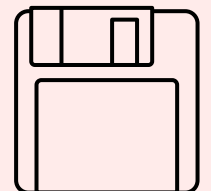
ドメインモデル
(更新済)



クラス図



テスト計画

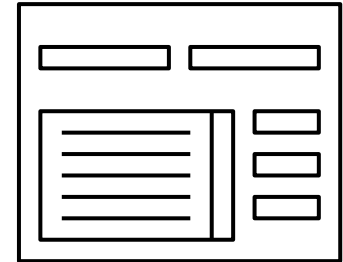


ソースコード
テストコード

前回の復習

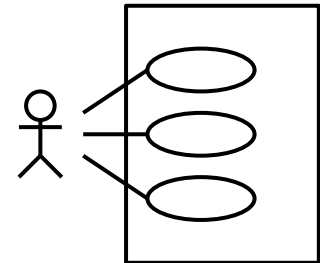
■紙芝居

- プロジェクト内でのwhatの共通認識形成
- 非UML, やり方は様々



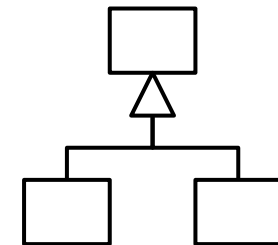
■ユースケース図

- Whatを一定の粒度で整理する



■ドメインモデル図

- システム内で扱う「モノ」を整理する
- 用語集



宿題講評



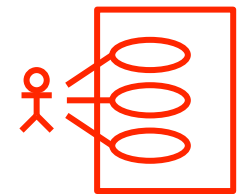
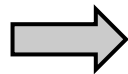
Cloud Specialist Program Initiative for Reality-based Advanced Learning



ICONIX 流れの確認

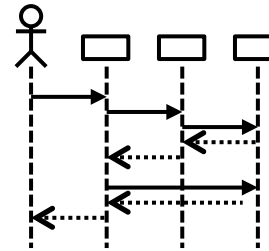
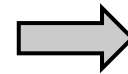


紙芝居

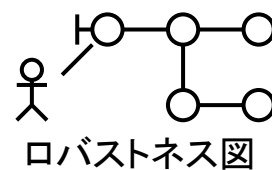


ユースケース図

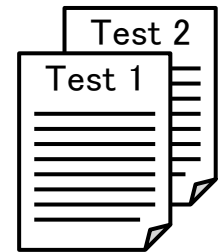
動的



シーケンス図



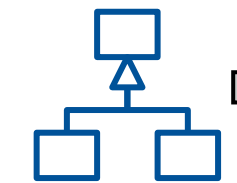
ロバストネス図



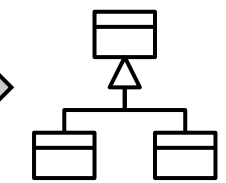
テスト計画



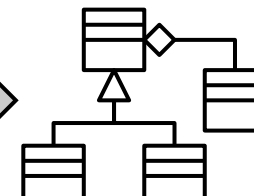
静的



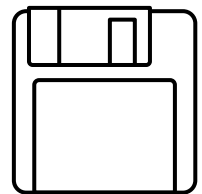
ドメインモデル



ドメインモデル
(更新済)



クラス図



ソースコード
テストコード

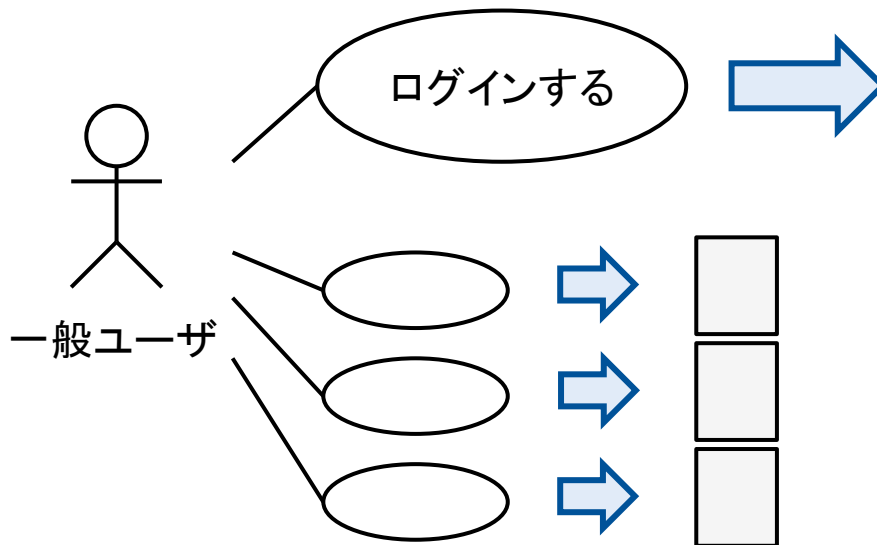
ユースケース記述

■目的:各ユースケースの具体化・詳細化

- ユーザとシステムの相互作用を記述する

ユースケース記述

ユースケース図



UC:ログインする

【基本コース】

- ・ユーザはログインリンクをクリックする.
- ・システムはログインページを表示する.
- ・ユーザはユーザIDとパスワードを入力し...

【代替コース】

- ユーザIDが間違っている場合:
- ・システムは...

2つのシナリオ

■【基本コース】

- 正常系のシナリオ
- 単一

■【代替コース】

- 異常系・分岐系のシナリオ
- 複数

UC: ログインする

【基本コース】

- ・ ユーザはログインリンクをクリックする.
- ・ システムはログインページを表示する.
- ・ ユーザはユーザIDとパスワードを入力し…
- ...

【代替コース】

ユーザIDが間違っている場合:

- ・ システムはエラーメッセージを表示し、直前のページを表示する

パスワードが間違っている場合:

- ・ システムは…

例:「ログインする」のUC記述

① ログインページ

cspiral2013

ログイン

UC: ログインする

UC名

画面操作

画面表示

入力

ログイン処理

画面表示

【基本コース】

- ・ ユーザは任意のページからログインリンクをクリックする.
- ・ システムはログインページを表示する.
- ・ ユーザはユーザIDとパスワードを入力し, ログインボタンをクリックする.
- ・ システムはアカウントリストから指定のユーザIDが存在するかを確認し, 一般ユーザアカウントを取り出す.
- ・ システムは入力パスワードと一般ユーザアカウントのパスワードの一致を確認する.
- ・ システムは歓迎メッセージとともに, 直前のページを表示する.

【代替コース】

ユーザIDが正しくない場合:

- ・ システムは「ユーザ名またはパスワード」が間違っていることを示すメッセージとともに, 直前のページを表示する.

パスワードが正しくない場合:

- ・ システムは「ユーザ名またはパスワード」が間違っていることを示すメッセージとともに, 直前のページを表示する.

UC記述作成時のポイント

■SV0で記述する

- 「ユーザは～を～する」「システムは～を～する」
- 「ユーザは～を～し、～を～する」

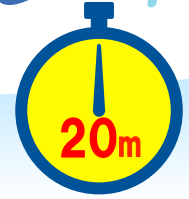
■ユーザとシステムの対話を具体的に書く

- ただしアルゴリズムの詳細を書かない



■ドメインモデル(用語集)の言葉を使う

■代替コースをよく考える



演習：UC記述の作成

■UC記述を作成してください

- 題材：UC「**アカウントを登録する**」
- 基本コース： 

代替コース： 

アカウント登録
ページ

cspiral2013

登録

UC「ログインする」

ユーザは任意のページからログインリンクをクリックする

システムはログインページを表示する

...

ユーザIDが間違っている場合

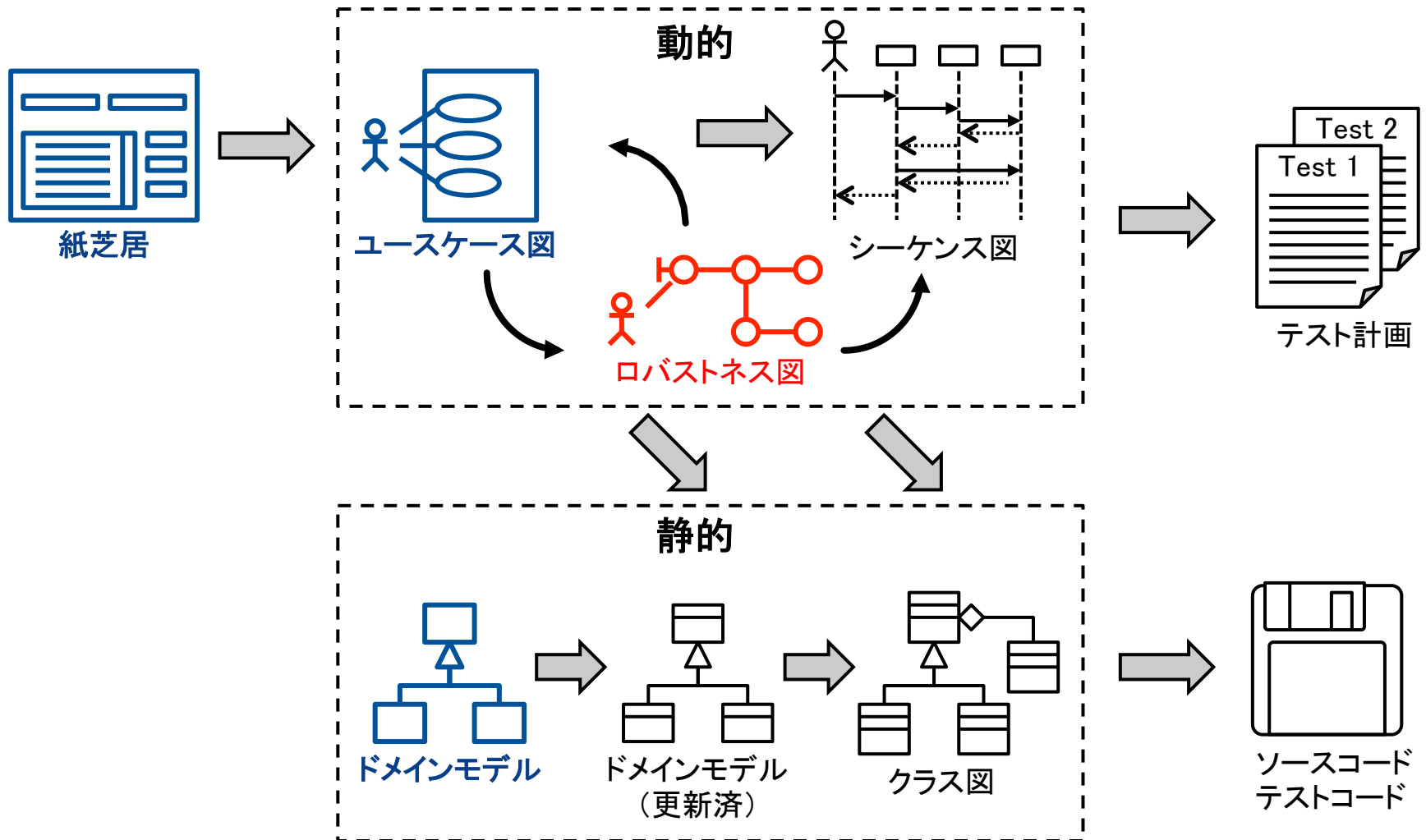
ユーザは「ユーザIDが間違っている」メッセージを表示する

...

パスワードが間違っている場合

...

ICONIX 流れの確認



ロバストネス図

■目的: UC記述の洗練

- UC記述を可視化, 分析し**その妥当性を確認する**
- WhatとHowの橋渡し

UC記述

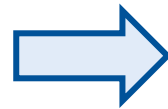
UC: ログインする

【基本コース】

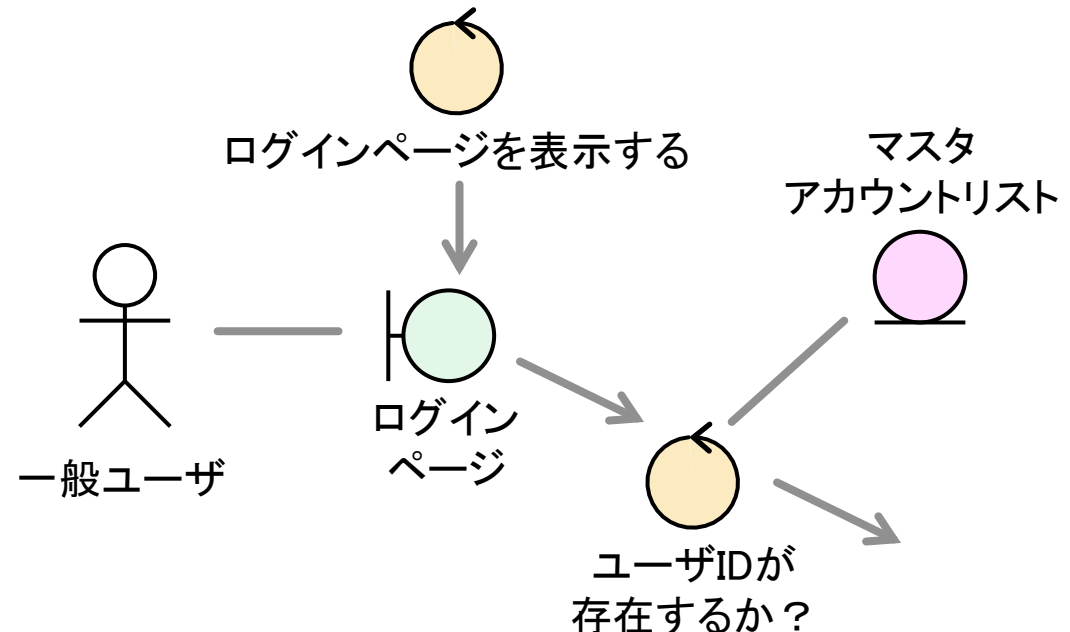
- ・ユーザは…
- ・システムは…

【代替コース】

- ～が間違っている場合:
- ・システムは…






ロバストネス図



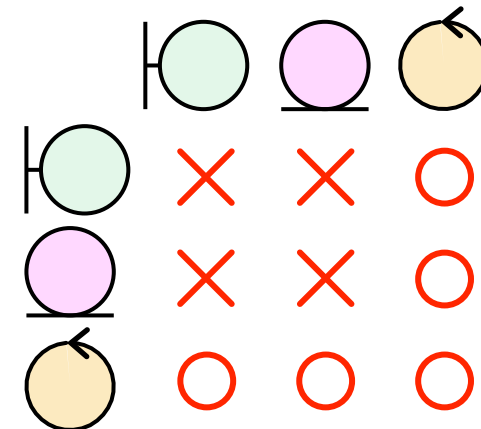
記法

■ステレオタイプ

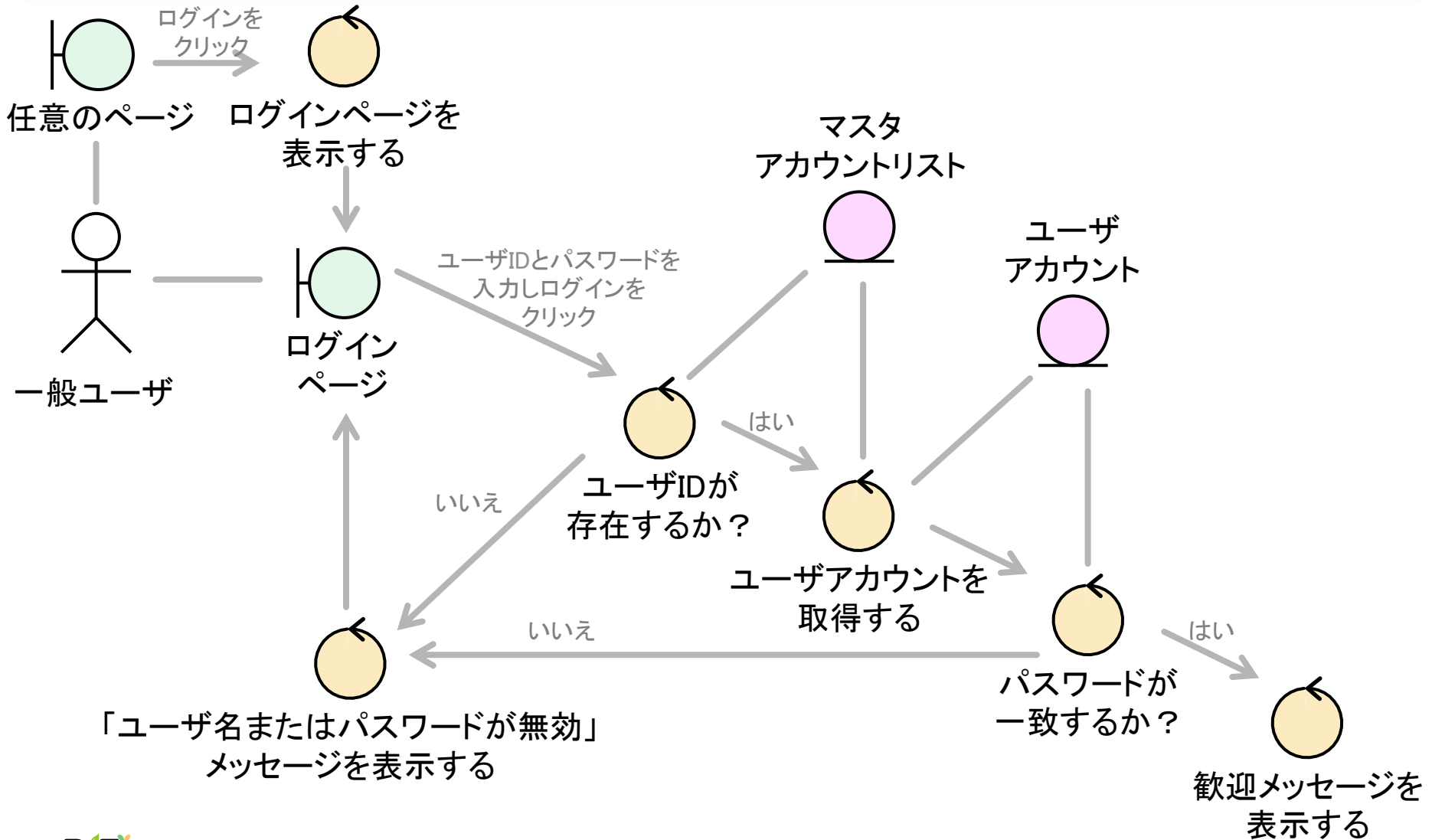
- ◆  バウンダリ: システム内部と外部の境界 = **画面**
- ◆  エンティティ: ドメインモデル上のモノ = **オブジェクト**
- ◆  コントローラ: 上2つの**接着剤**

■ステレオタイプ間の関係

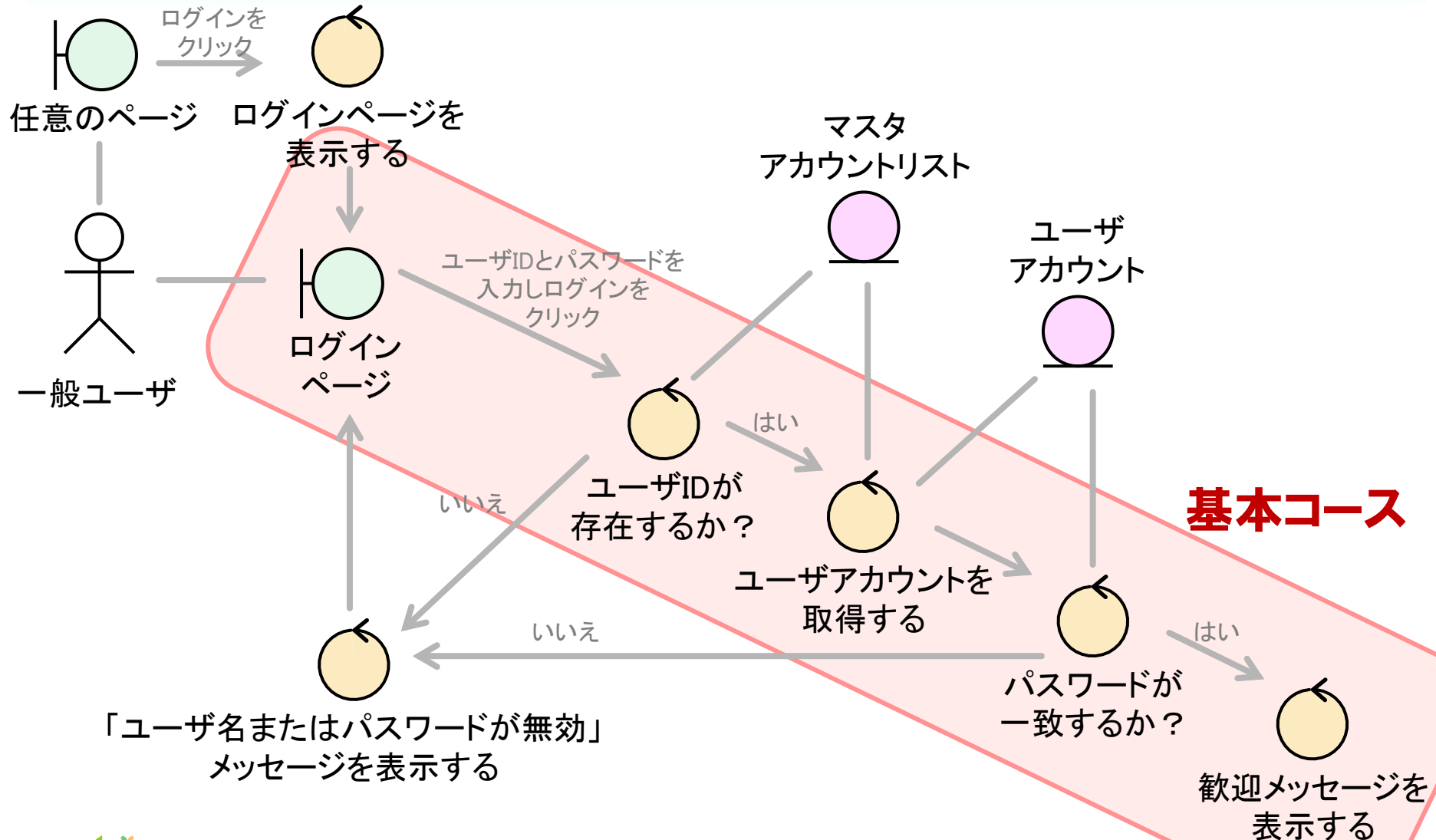
- 必ずコントローラと接続する



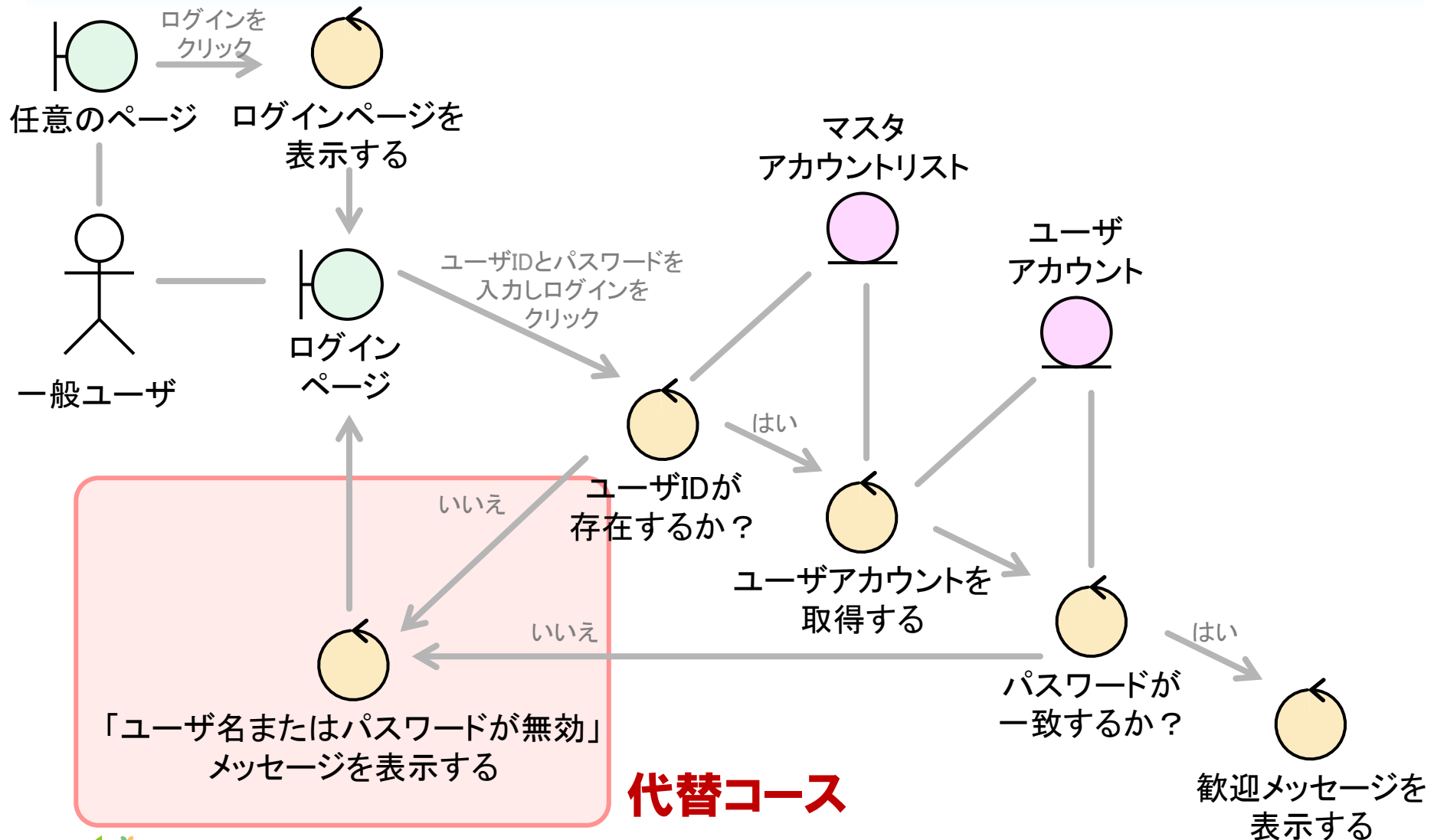
例：UC「ログインする」のロバストネス図



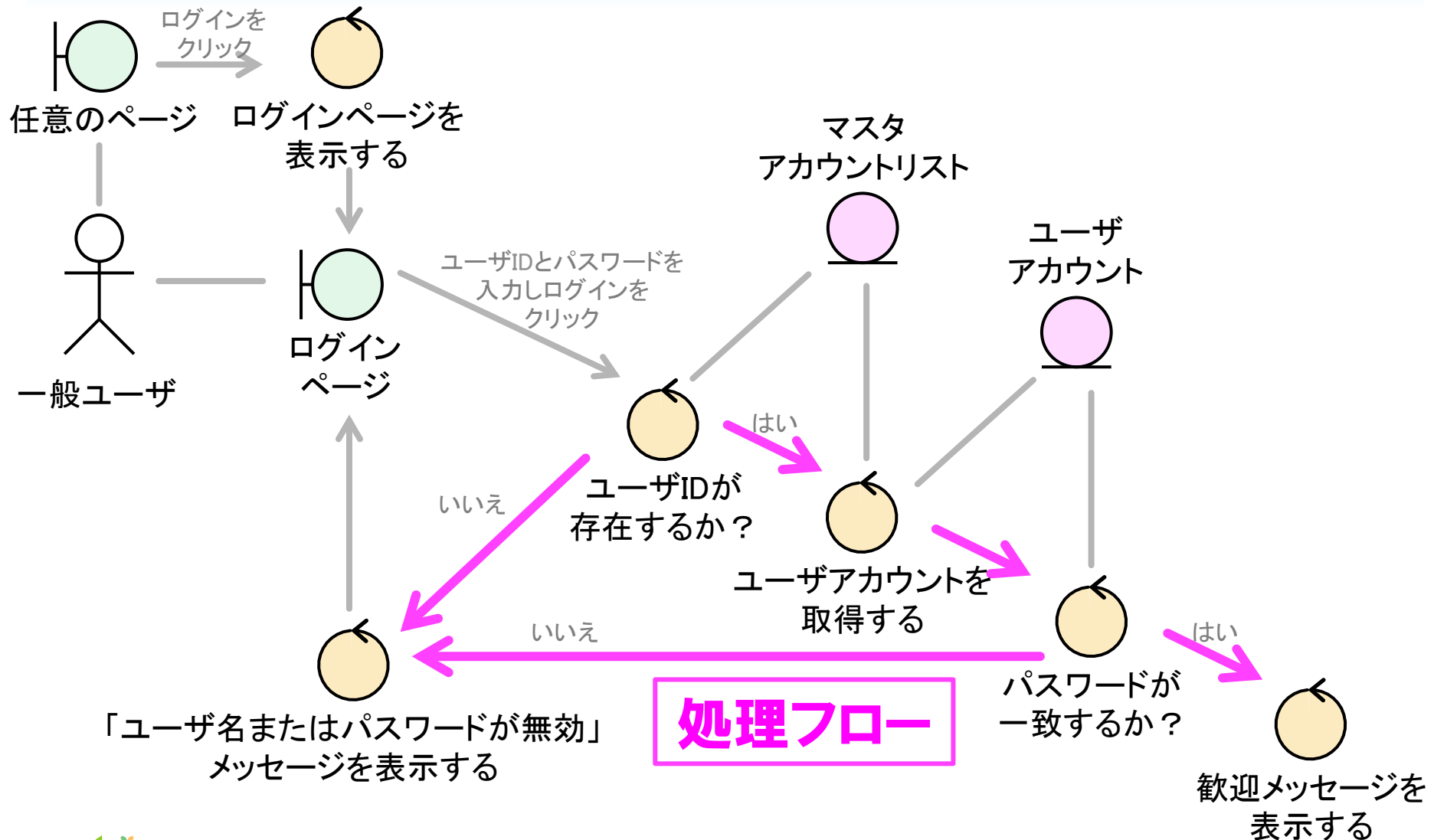
例：UC「ログインする」のロバストネス図



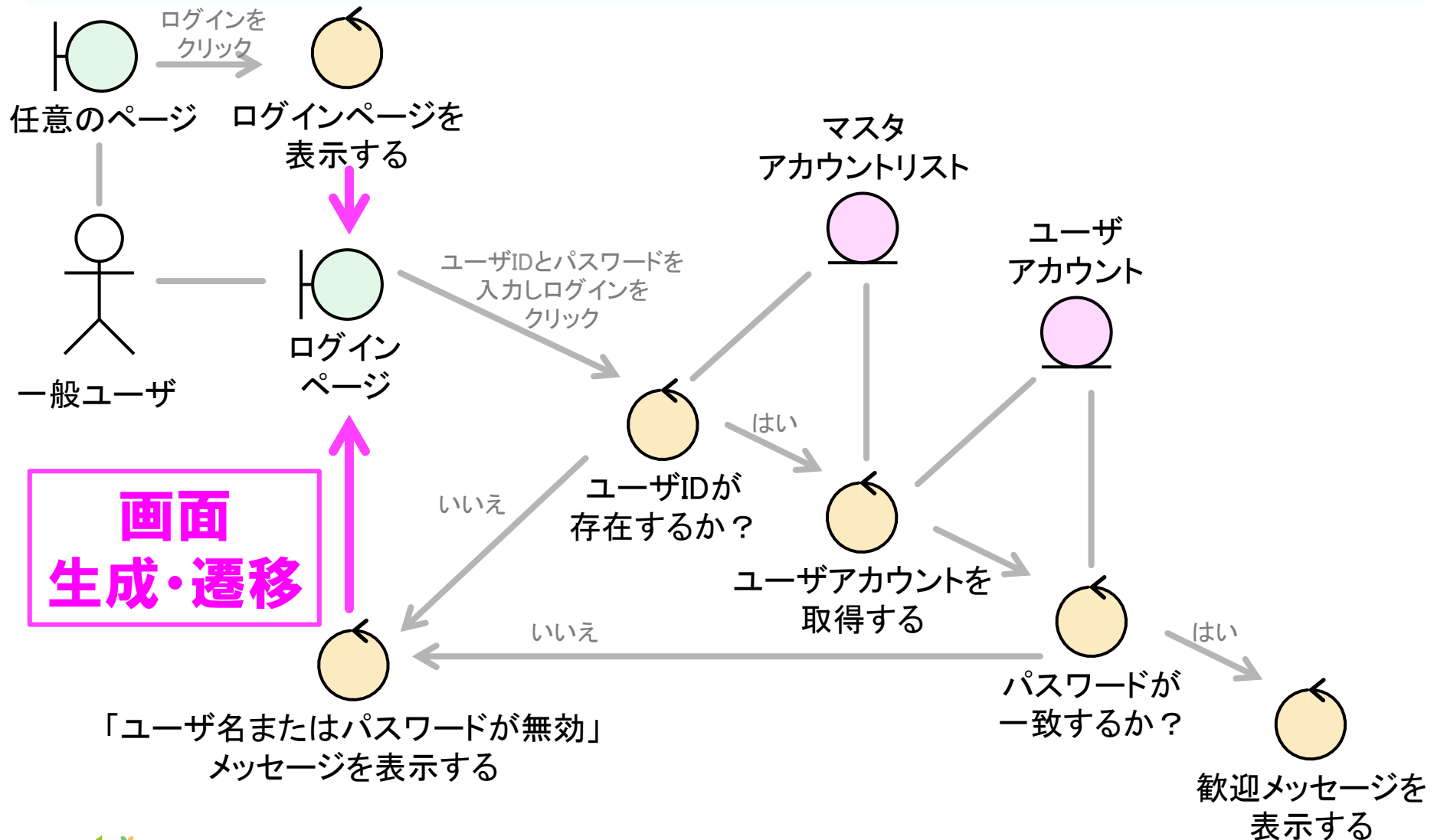
例：UC「ログインする」のロバストネス図



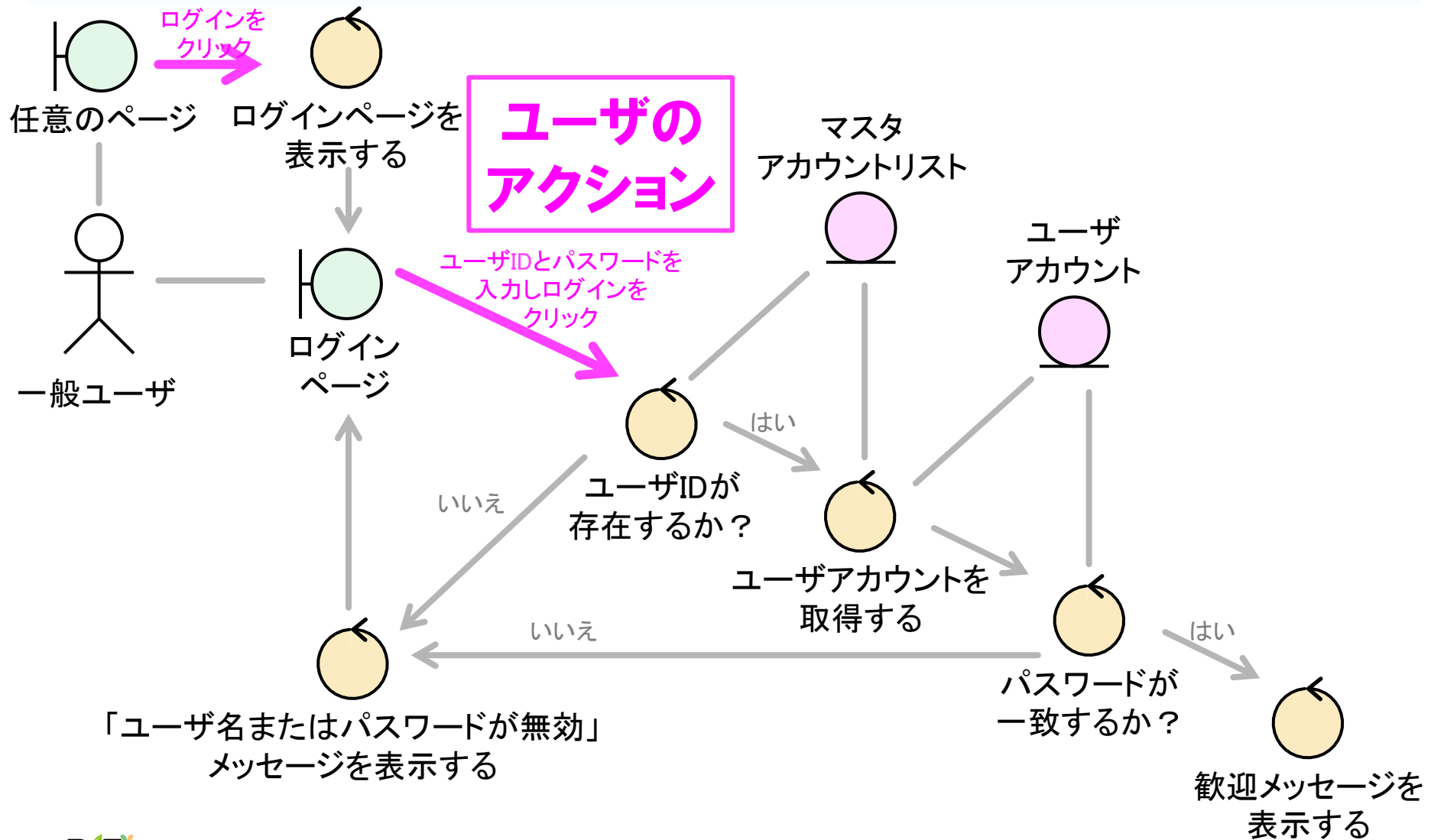
例：UC「ログインする」のロバストネス図



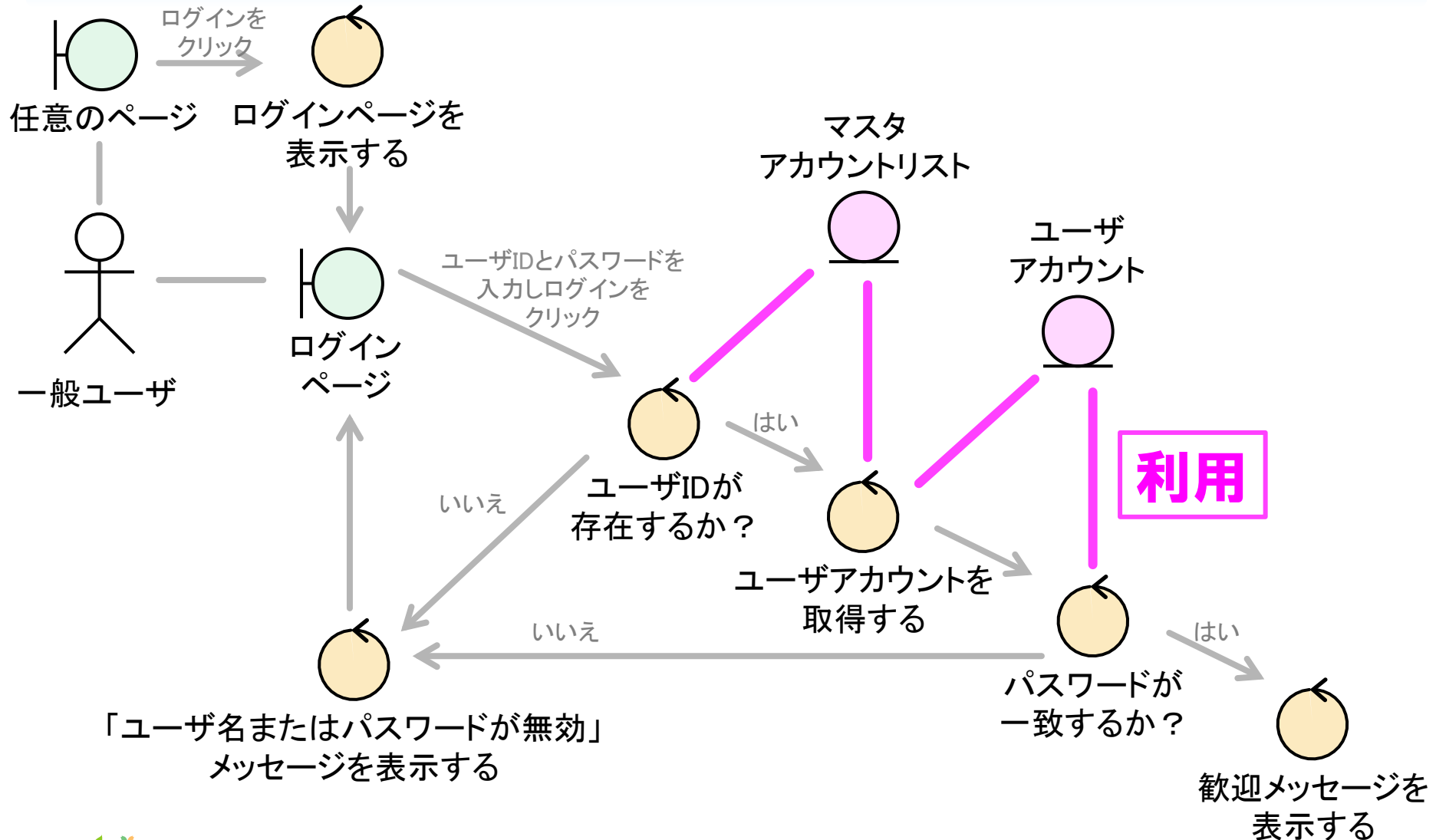
例：UC「ログインする」のロバストネス図



例：UC「ログインする」のロバストネス図



例：UC「ログインする」のロバストネス図



ロバストネス図作成のポイント

■まずUC記述をそのまま貼り付ける

■UC記述をそのまま図に書き直す

- 基本コース・代替コースの両方を書き込む

■UC記述も同時に修正する

- ここが目的

■矢印の意味を考えすぎない



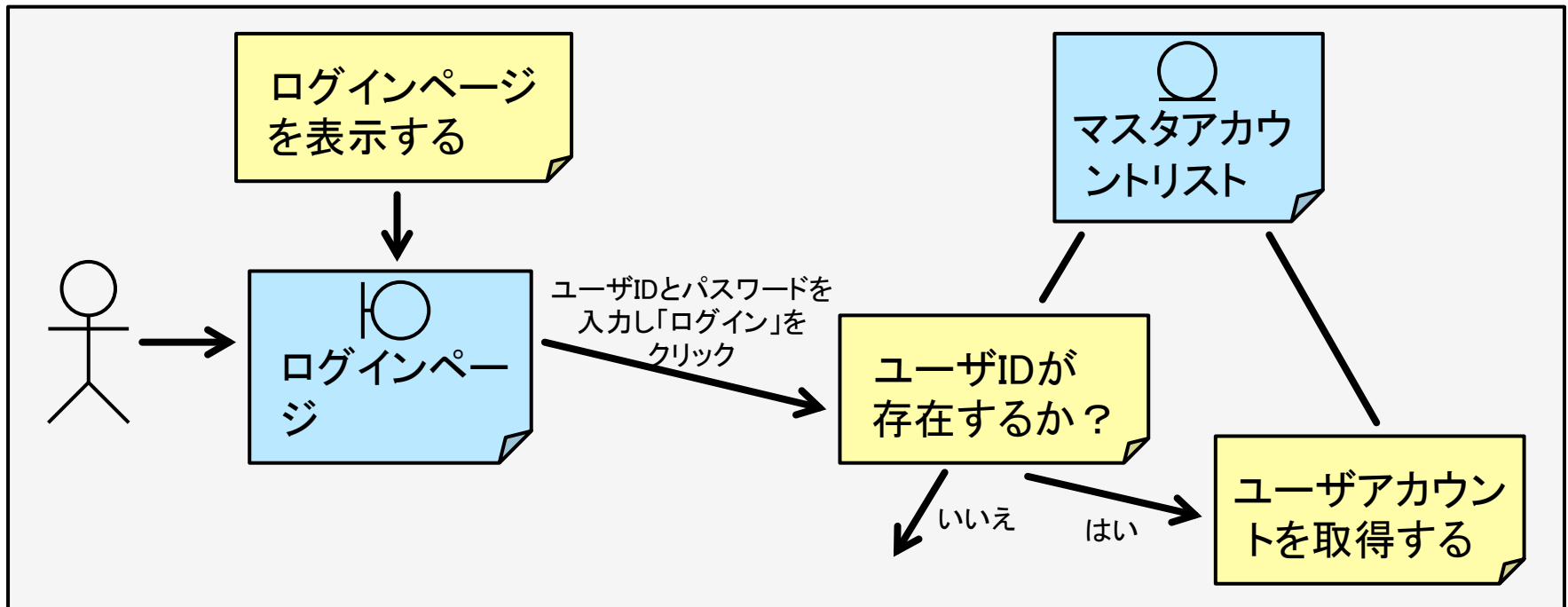


演習：ロバストネス図の作成

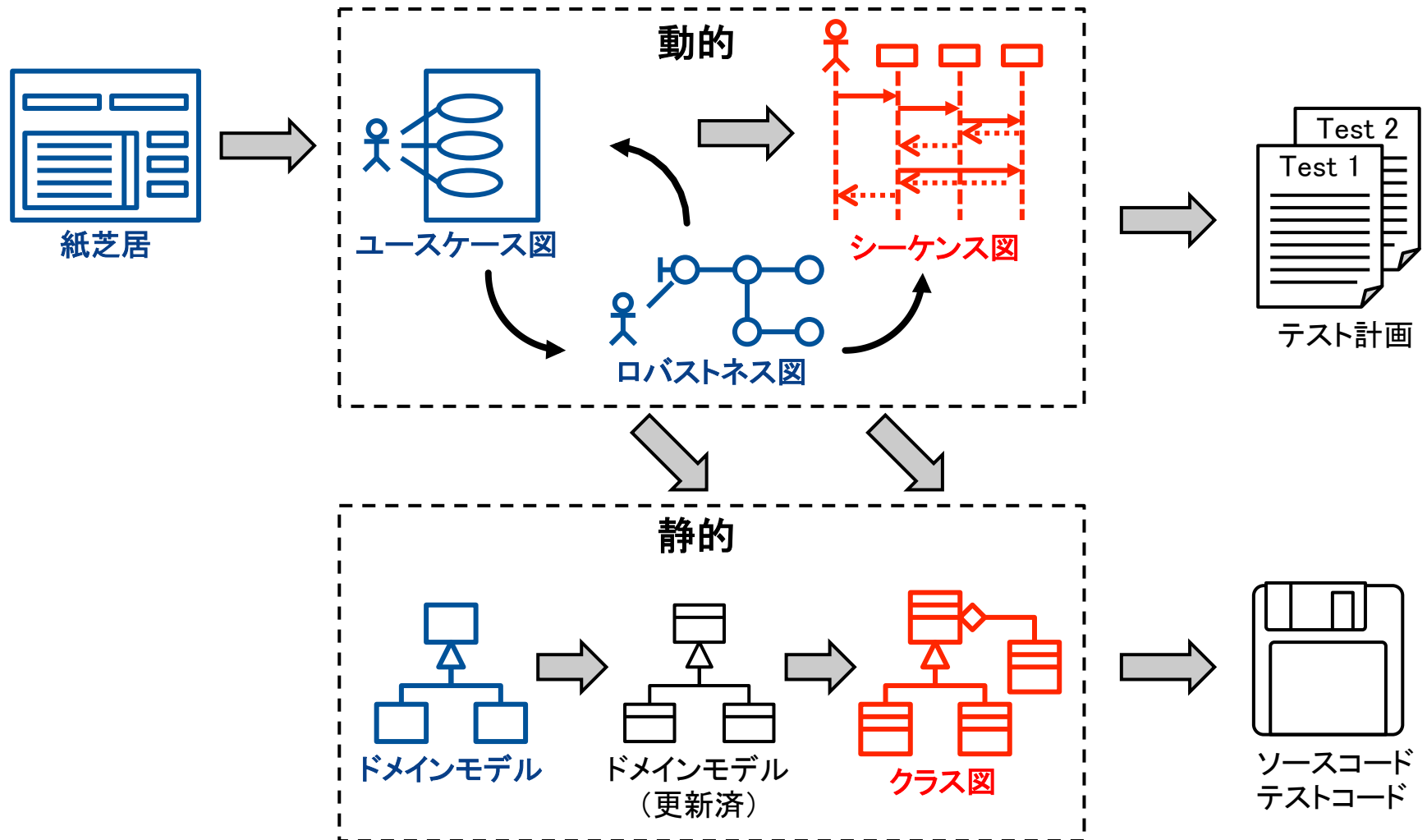
■ロバストネス図を作成してください

- 題材：UC「**アカウントを登録する**」

- コントローラ：  バウンダリ・エンティティ： 



ICONIX 流れの確認



シーケンス図

■目的: クラスの責務割り当て

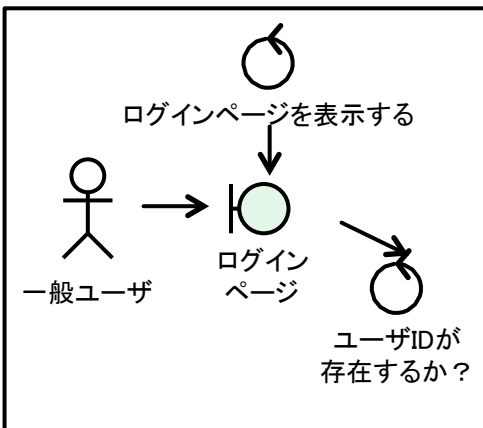
- 目的は処理フローの記述ではない
- クラスの責務 = 属性と振る舞い



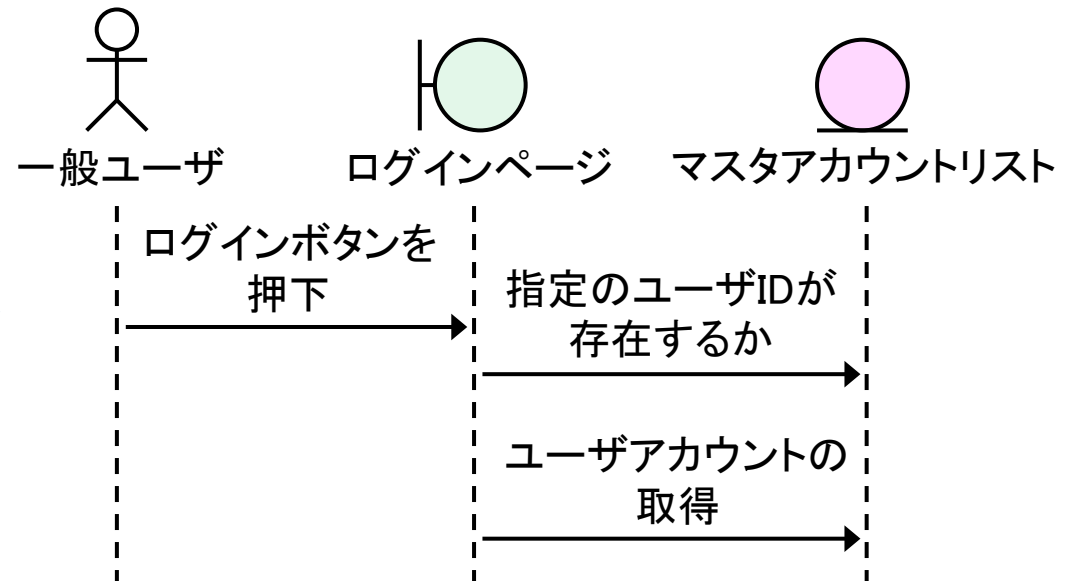
クラス名
属性
振る舞い

シーケンス図

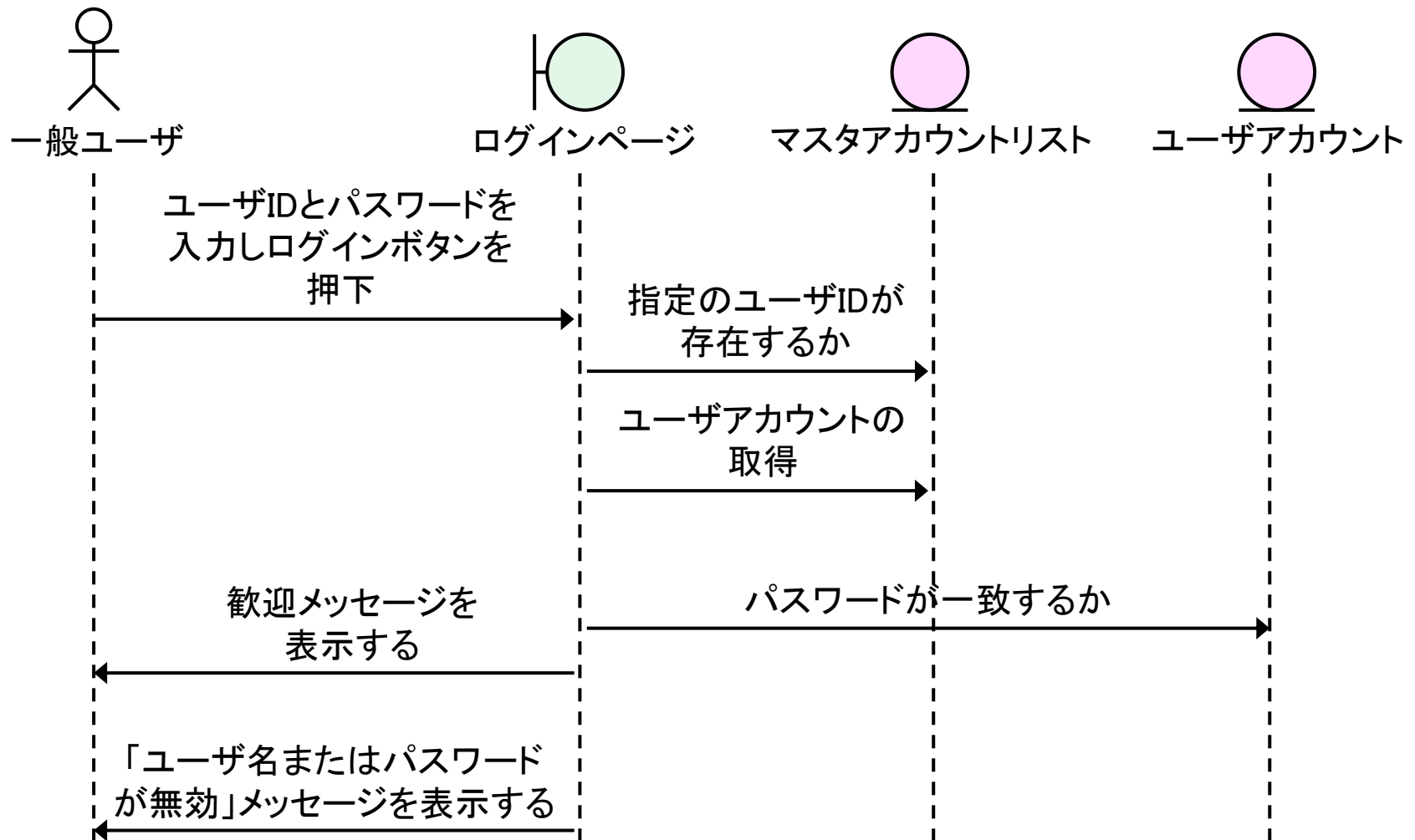
ロバストネス図



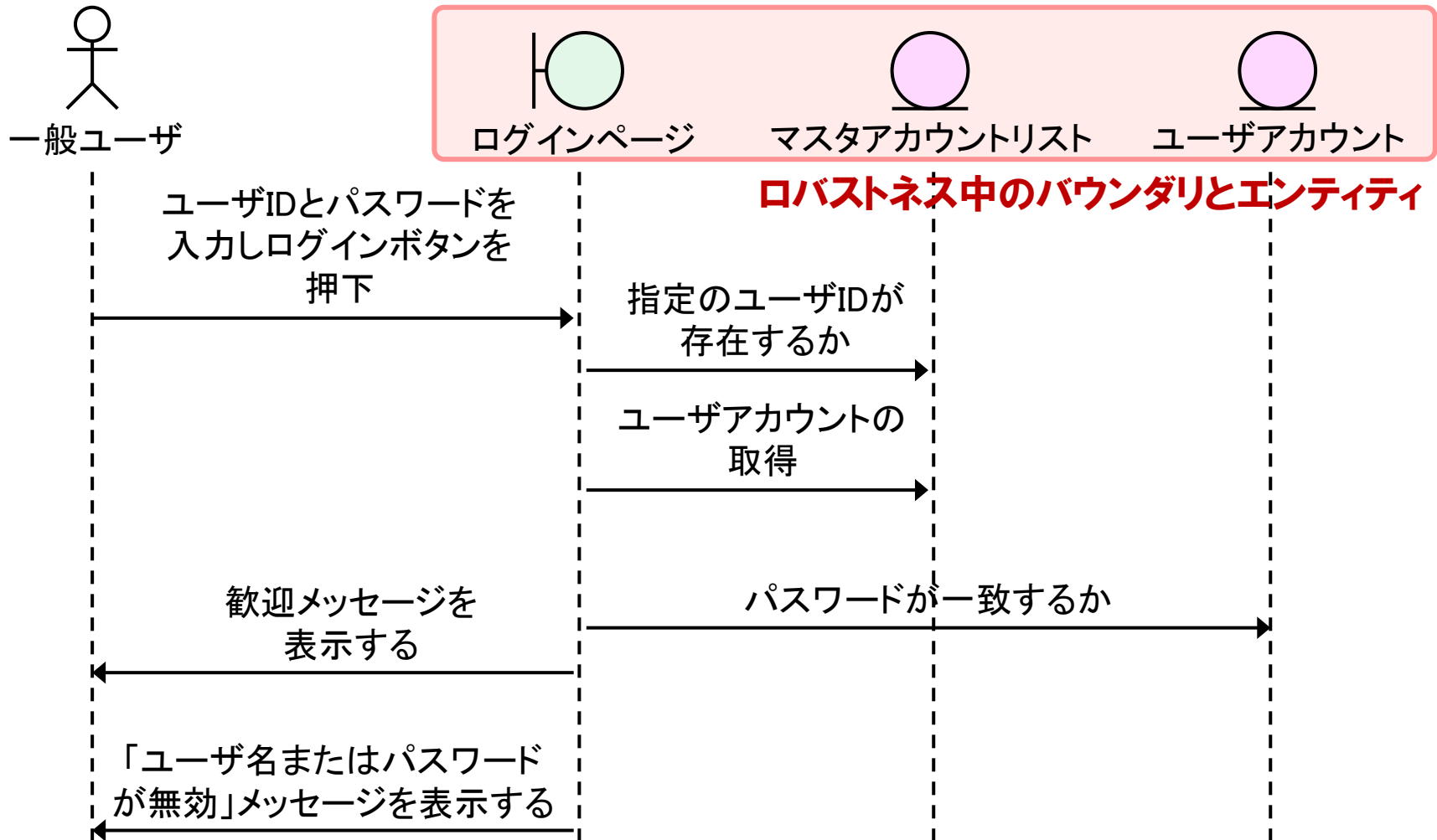
自動化
可能



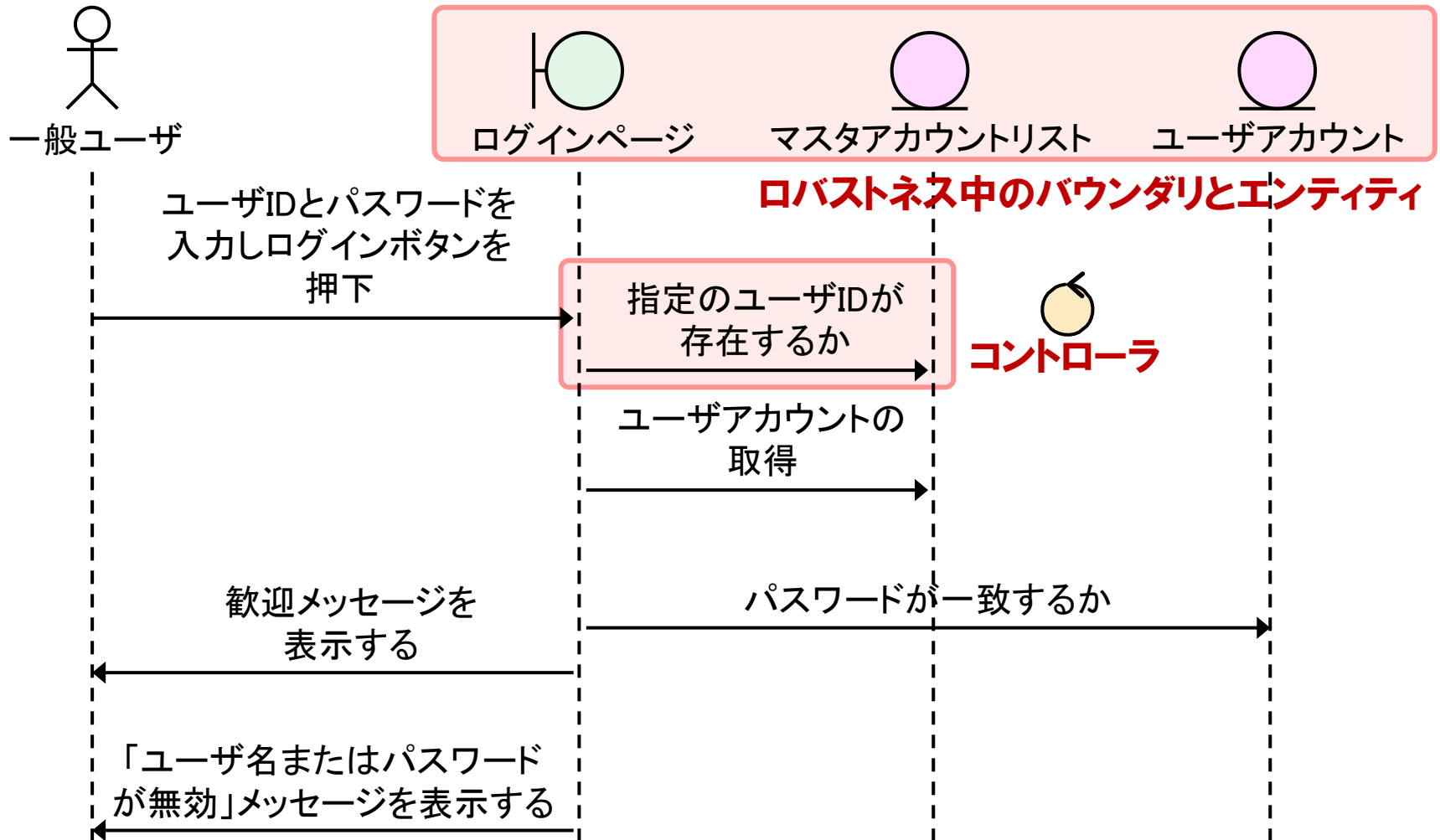
例：UC「ログインする」のシーケンス図



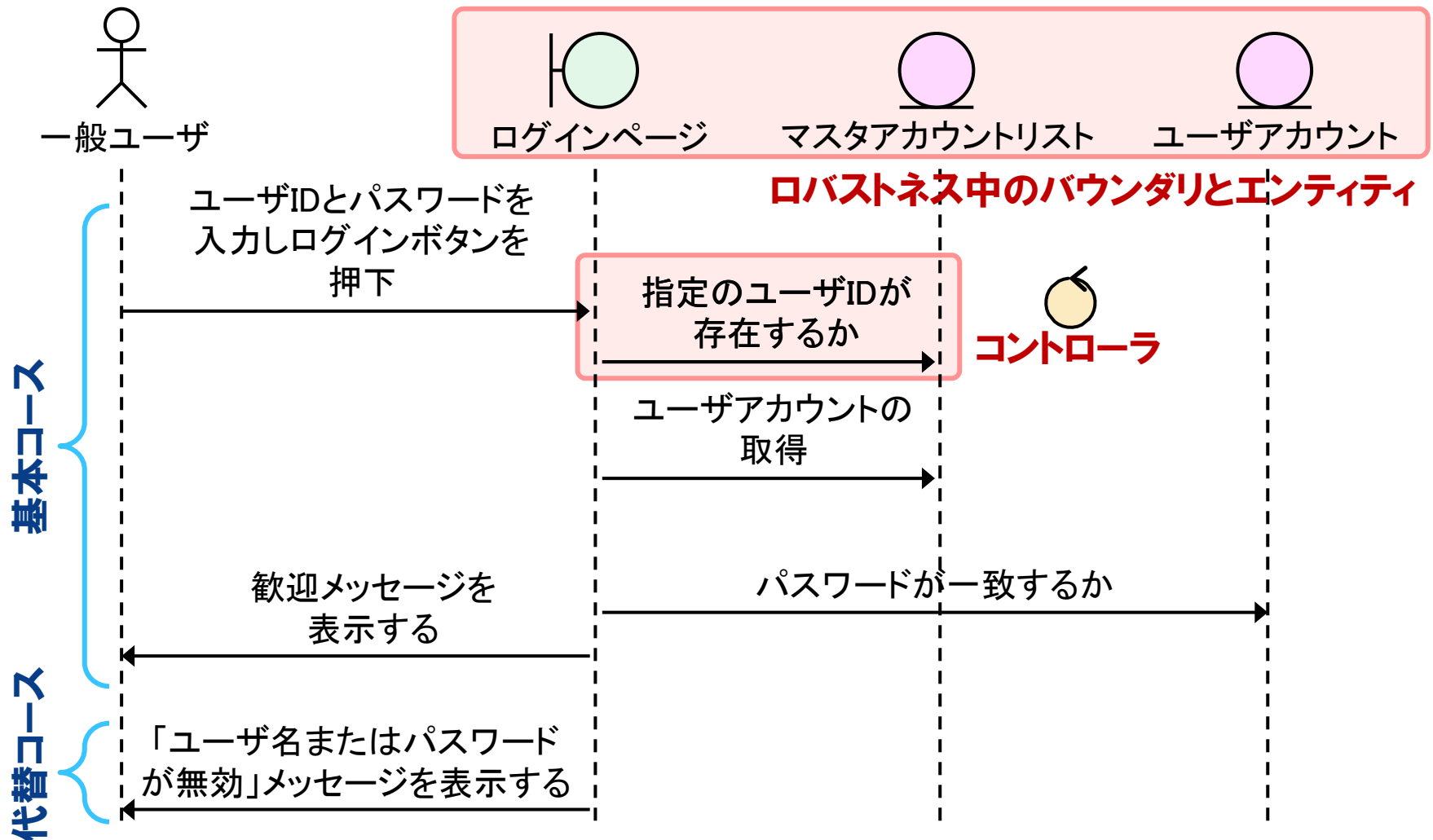
例：UC「ログインする」のシーケンス図



例：UC「ログインする」のシーケンス図



例：UC「ログインする」のシーケンス図



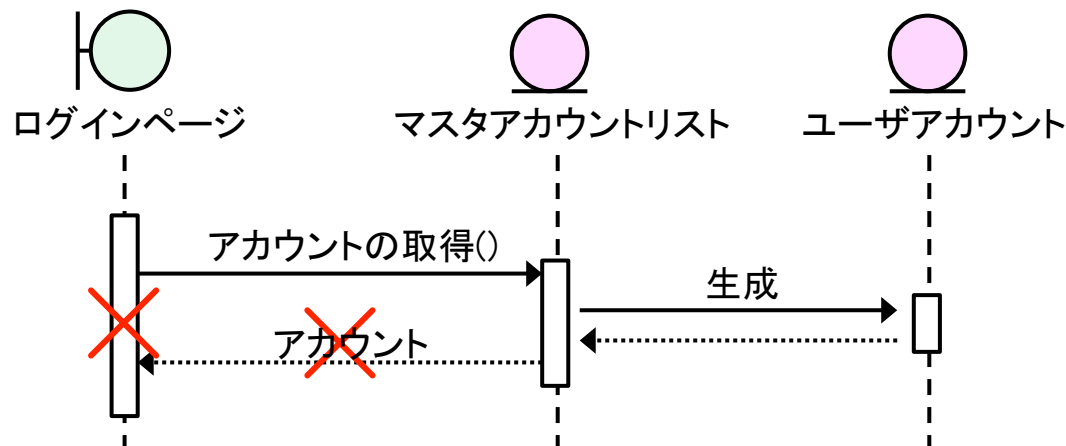
シーケンス図作成のポイント

■まずロバストネス図をそのまま貼り付ける

- ロバストネス図のバウンダリとエンティティを書き込む
- コントローラを矢印に書き換える

■活性区間，返回值，分岐を書かない

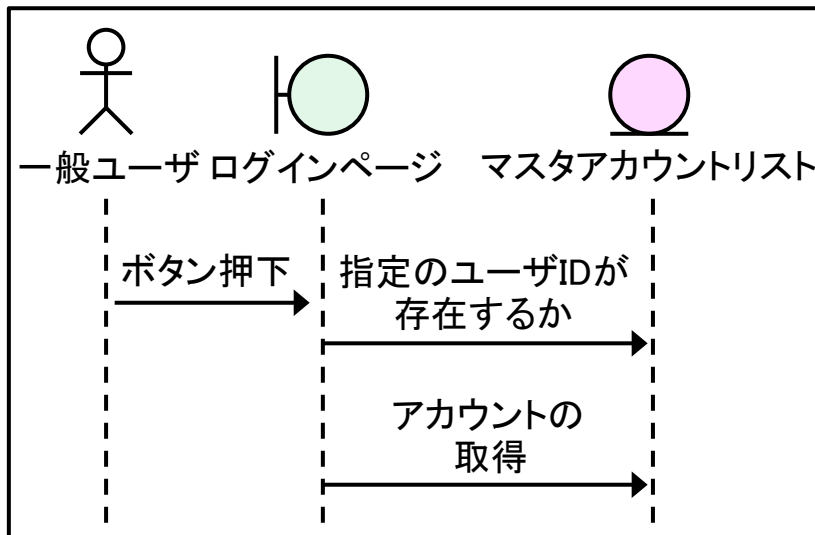
- あるいは無視する



クラス図

■クラスの責務(属性と振る舞い)を構造化する

シーケンス図

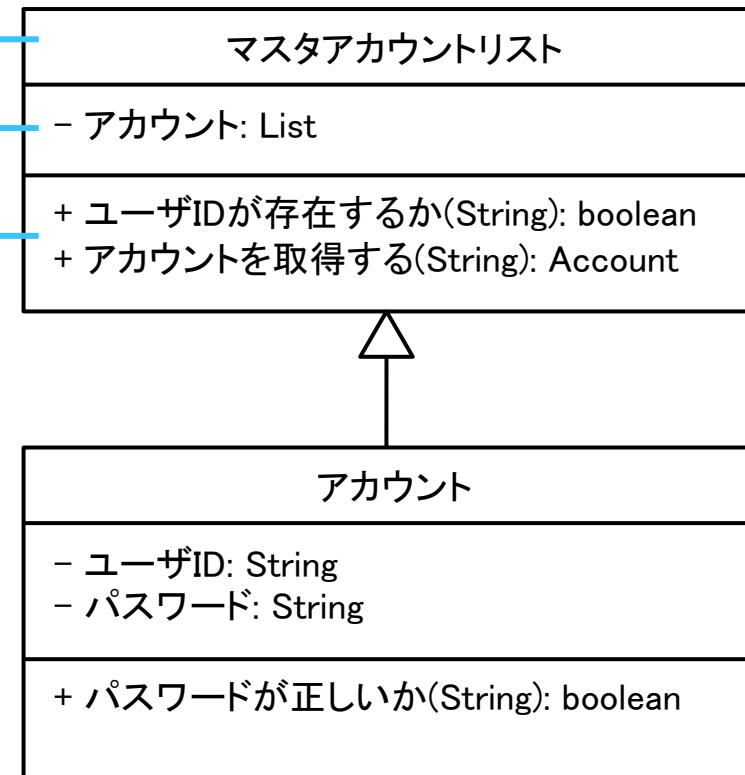


クラス名

属性

振る舞い

クラス図

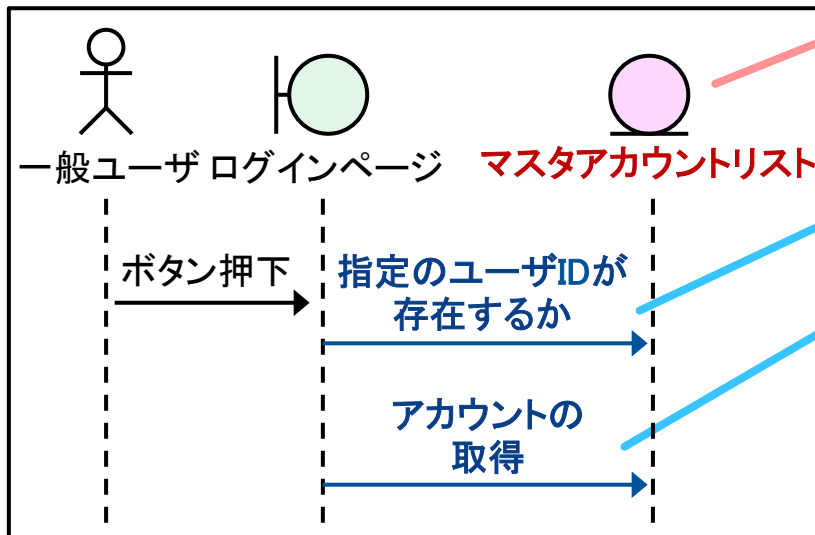


クラス図の作成

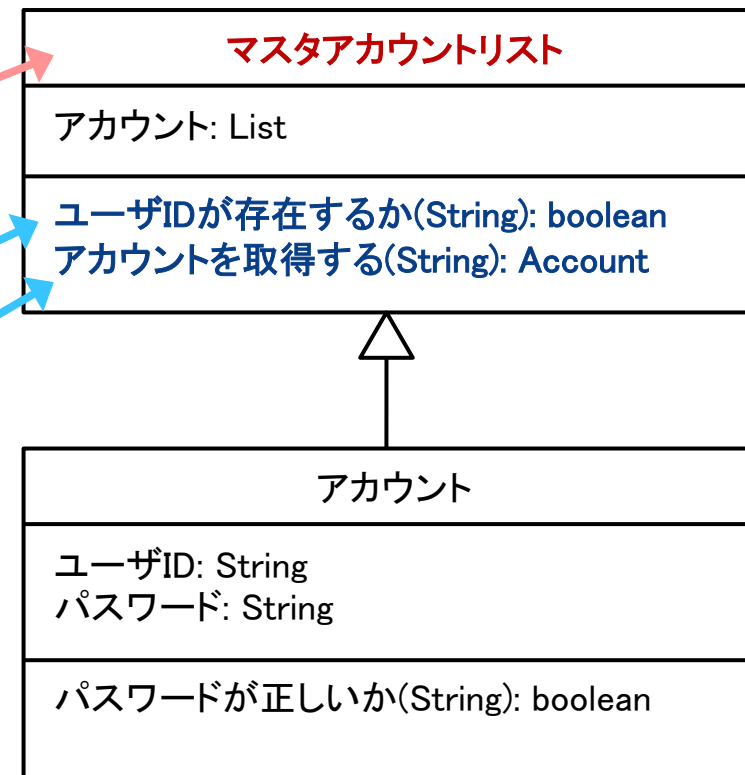
■シーケンス図からほぼ自動的に作成できる

- エンティティをクラスに
- 矢印を振る舞いに

シーケンス図



クラス図

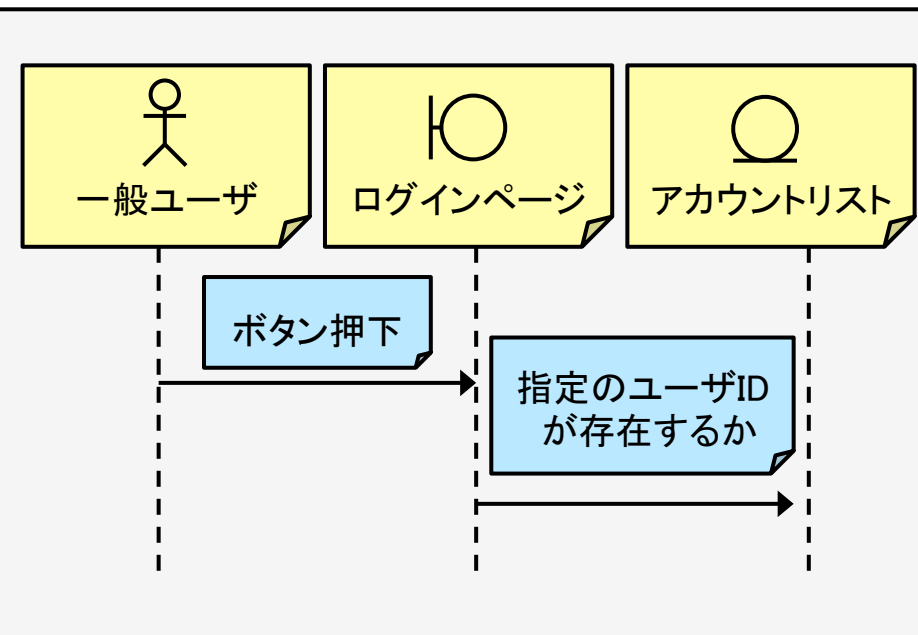


演習：クラス図・シーケンス図の作成

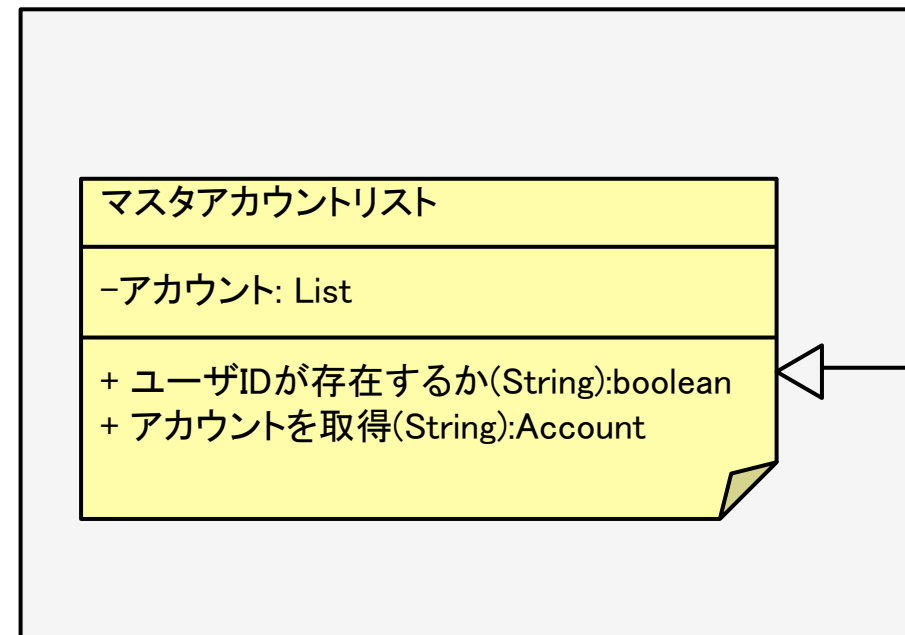


■2つの図を作成してください

シーケンス図



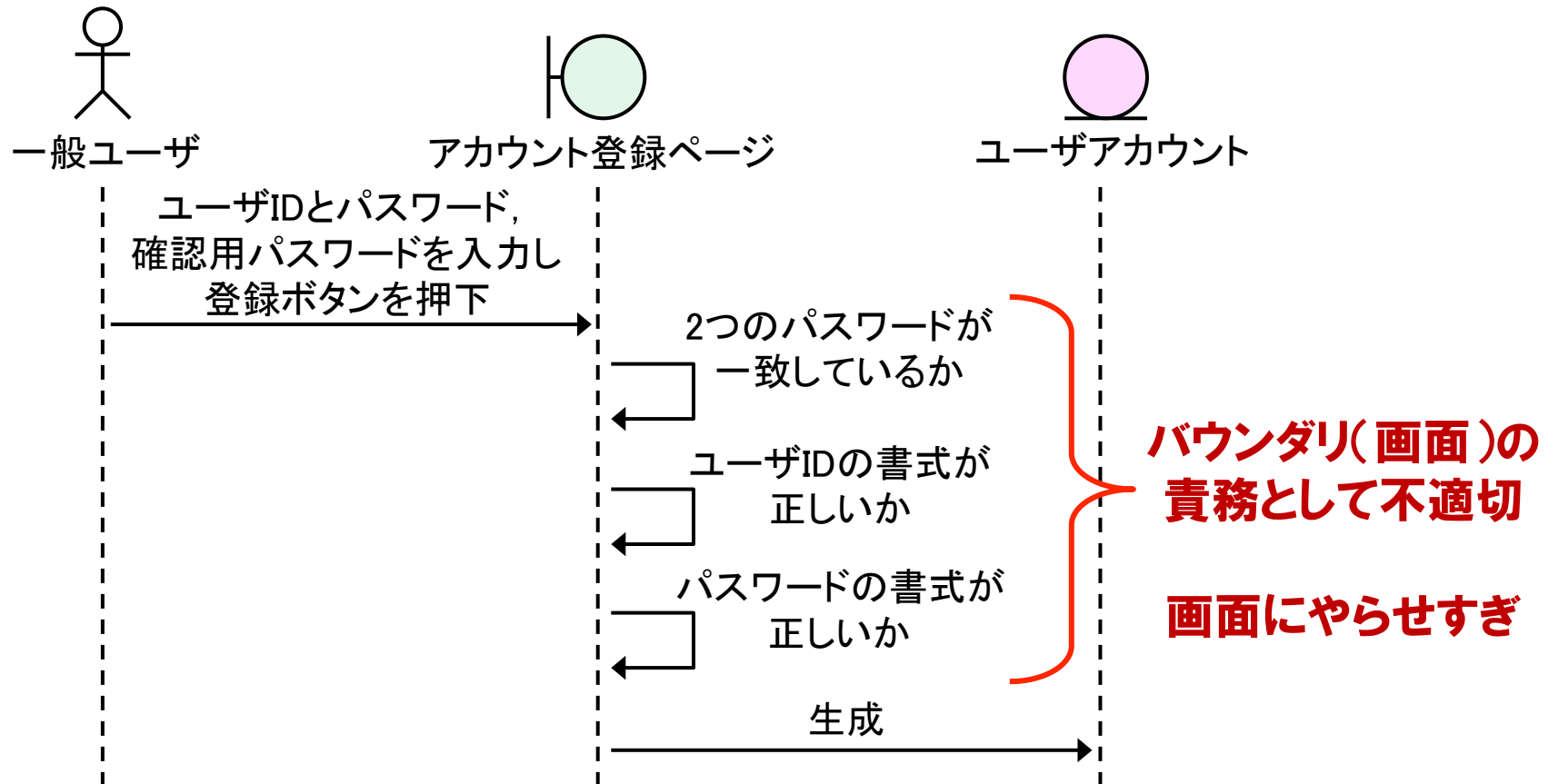
クラス図



さらなるステップ

■責務の割り当てを再検討する

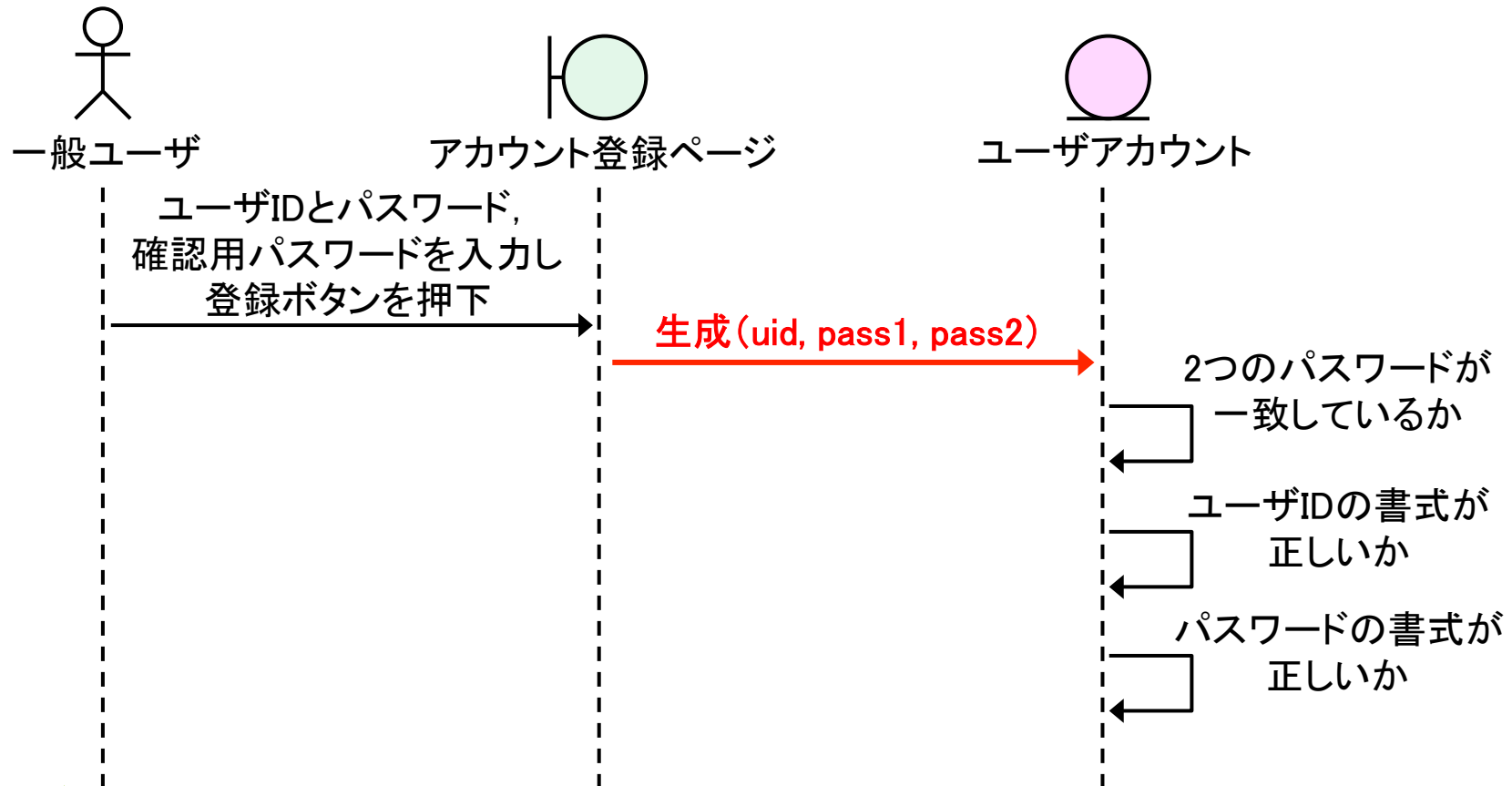
● UC「アカウントを登録する」の例



さらなるステップ

■ 対処1:ドメインオブジェクトに責務を割り当てる

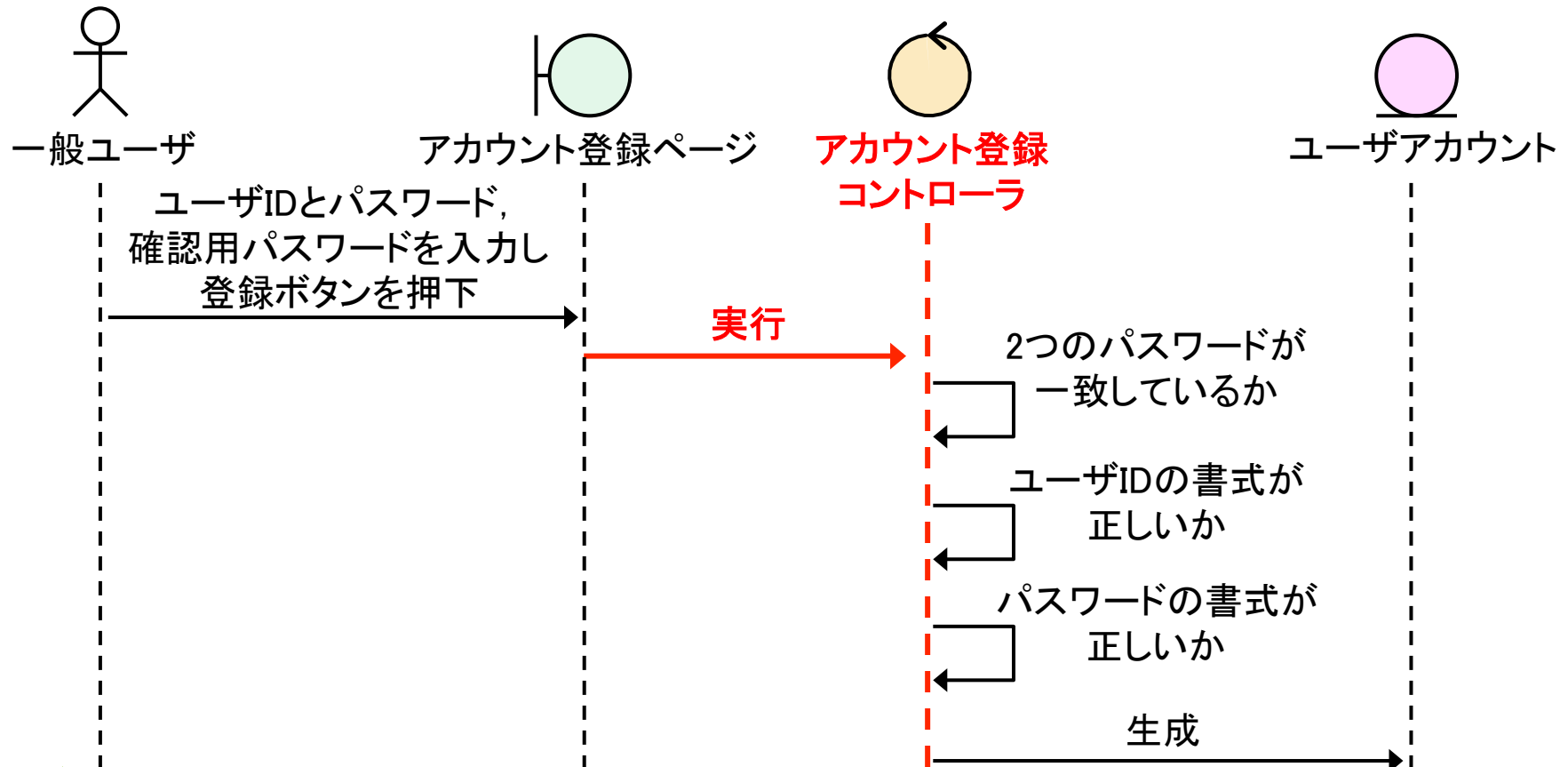
● Domain Model



さらなるステップ

■対処2: 専用コントローラに責務を割り当てる

● Transaction Script



その後の流れ

■詳細設計レビューの実施

- 責務を割り当ての再検討
 - ✓ 使用フレームワークやパターンとの兼ね合い(Spring, Seaser, Struts, ...)
- 全ての機能要求が満たされているか確認
- ...

■テスト計画の作成

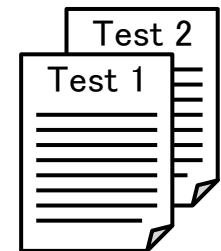
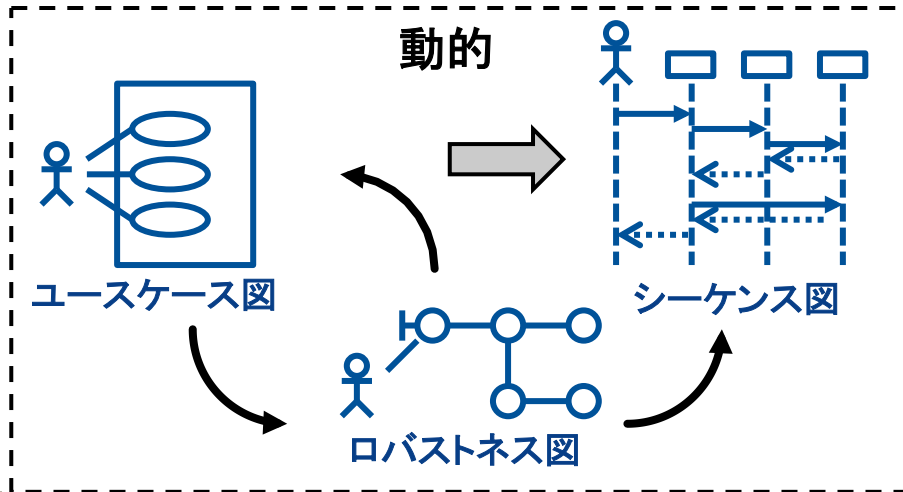
■クラス図からコードのひな形生成

■実装, レビュー, テスト, ...

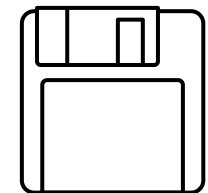
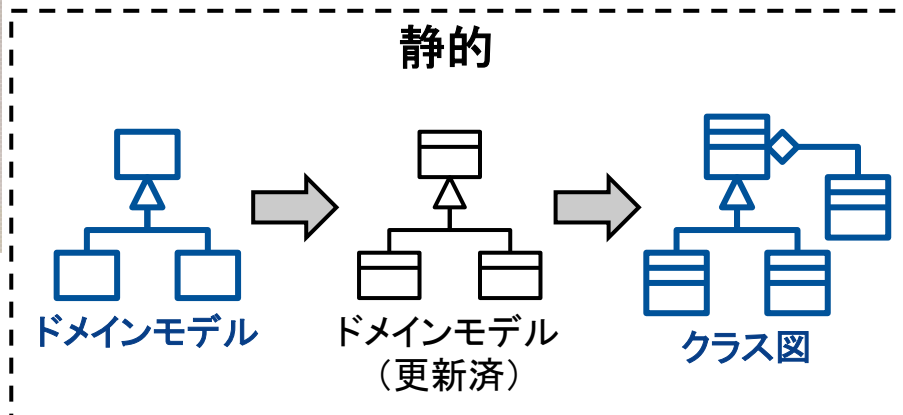
ICONIX おさらい



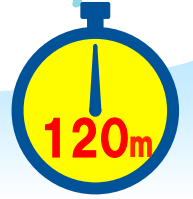
紙芝居



テスト計画

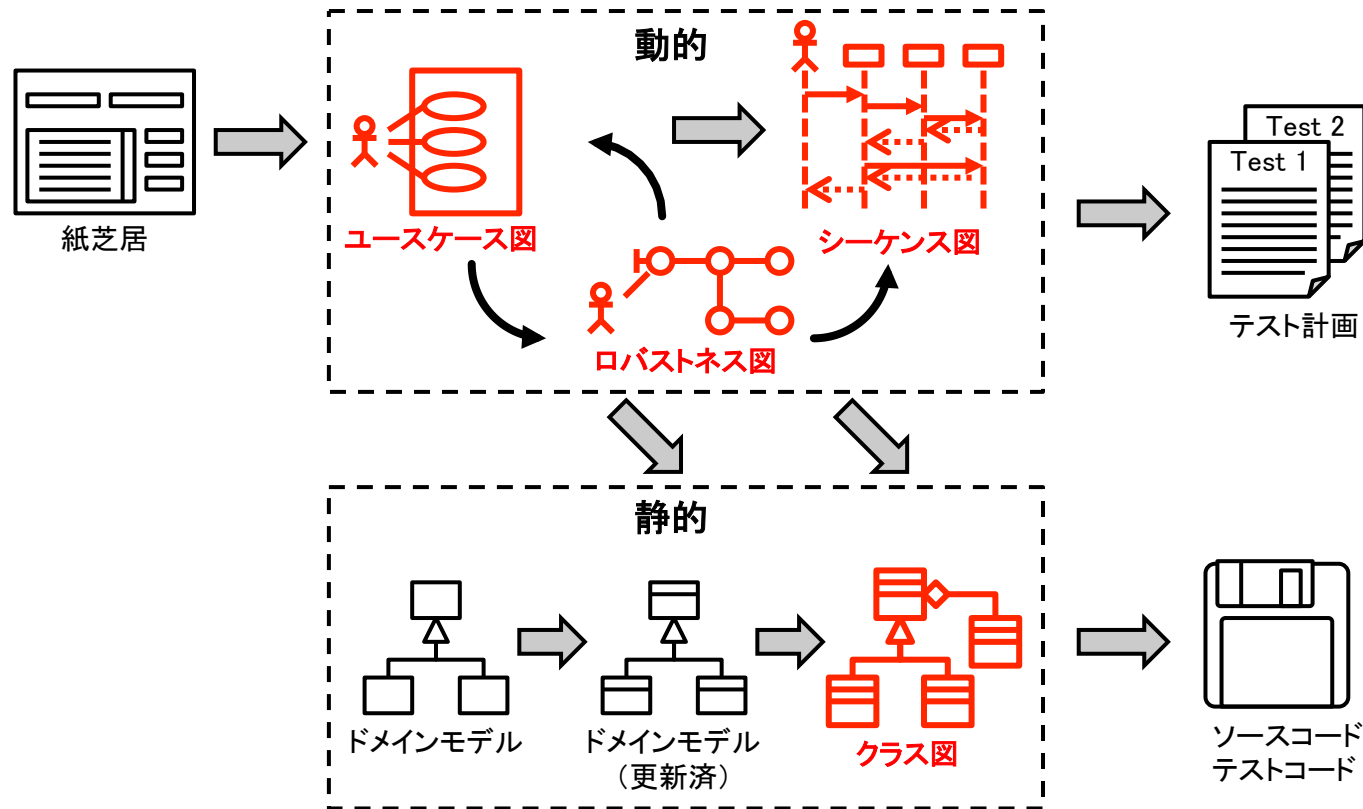
ソースコード
テストコード

総合演習



■UC記述～クラス図までの作成を実践してください

- 題材: UC「チケットを予約する」

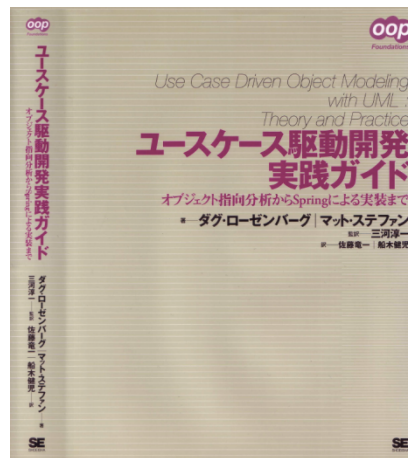


その他資料



参考図書

- 『はじめて学ぶUML』
- 『ユースケース駆動開発実践ガイド』
- 『ユースケース入門—ユーザマニュアルからプログラムを作る』
- 『オブジェクト指向でなぜつくるのか』
- OMGによるUMLの仕様書
- ◆ <http://www.uml.org/>



破棄

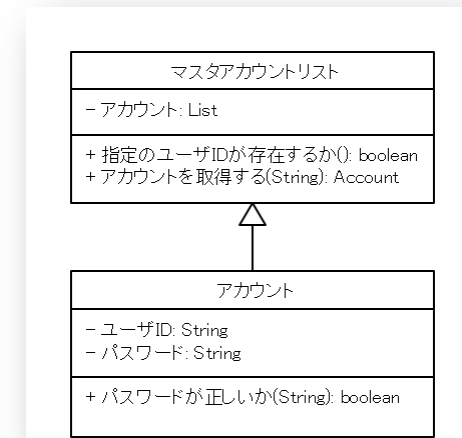
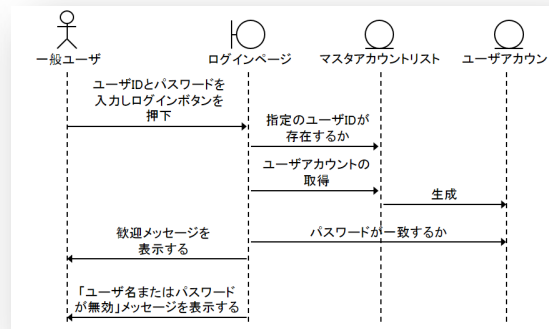
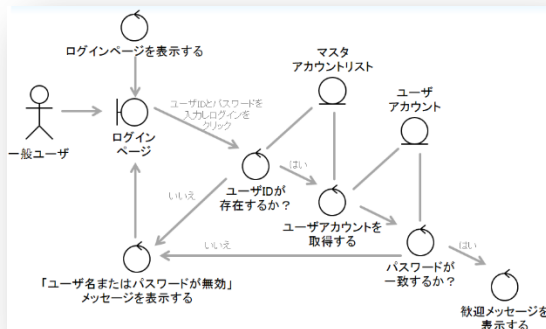
Cloud  *Spiral*

Cloud Specialist Program Initiative for Reality-based Advanced Learning

内容

■UC:予約をキャンセルする を対象に

- ◆ UC記述
- ◆ ロバストネス図
- ◆ シーケンス図
- ◆ クラス図



を各自で書いてください

作り方

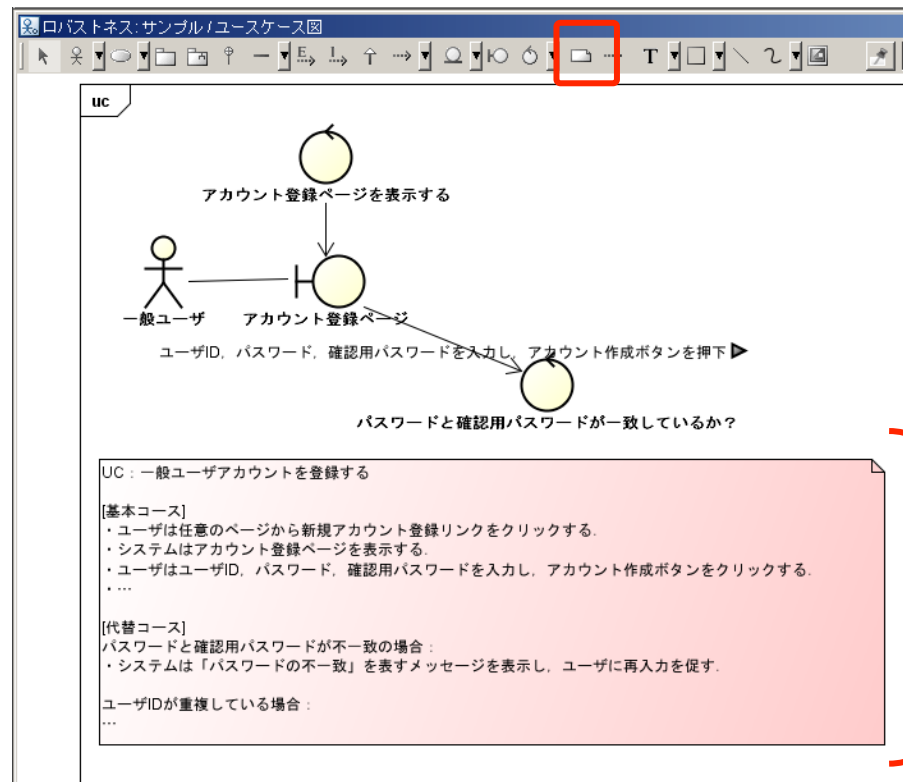
■astah*を使い、それぞれの図を作成します

- ◆ロバストネス図: astah*のユースケース図
- ◆シーケンス図: astah*のシーケンス図
- ◆クラス図: astah*のクラス図

■どうしてもastah*の使い方が分からない場合は pptでも手書きでも可とします

UC記述の添付

■ロバストネス図・シーケンス図には、UC記述をノート形式で貼り付けてください



提出

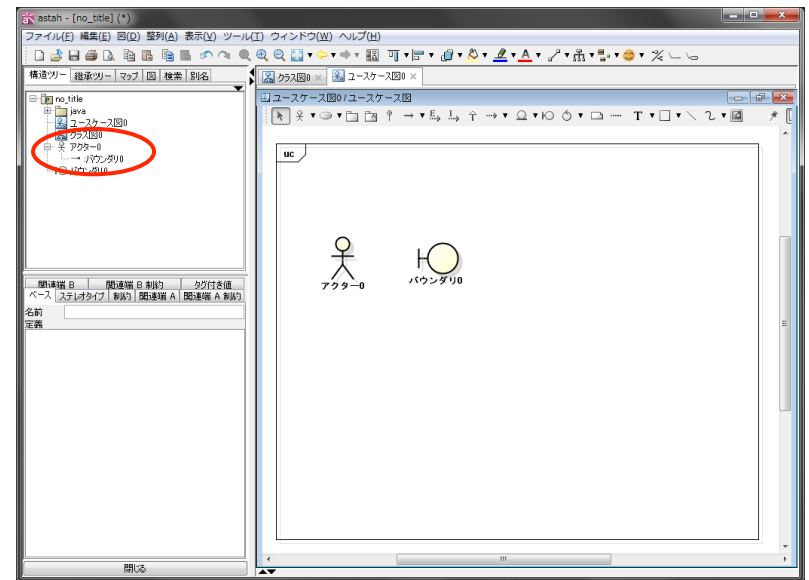
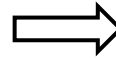
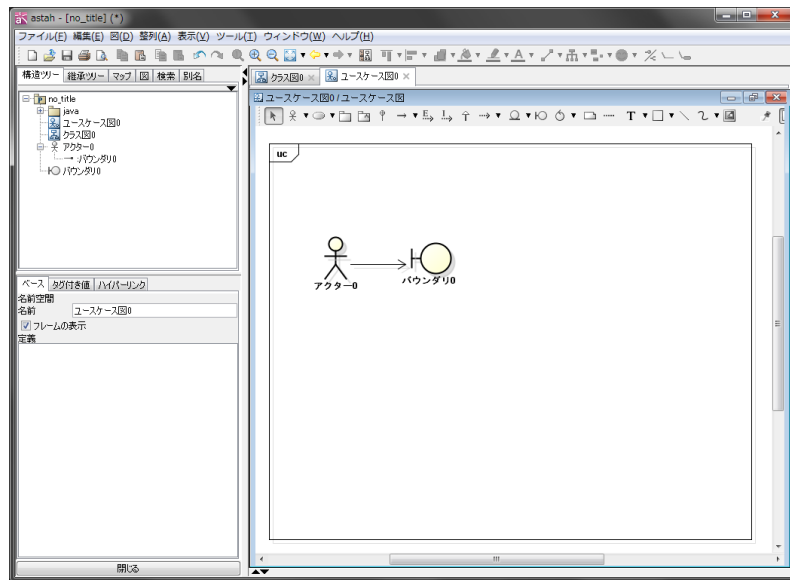
- それぞれを [ツール] → 図を画像ファイルに出力でPNG形式で保存してください。
- ファイル名: dropbox内の reports に
[20130524] フォルダを作成し, その中に
 - ◆ 20130XX-ロバストネス図.png
 - ◆ 20130XX-シーケンス図.png
 - ◆ 20130XX-クラス図.pngを提出して下さい

提出期限: 2013/5/30(木) 16:00

astahで作る場合の注意点

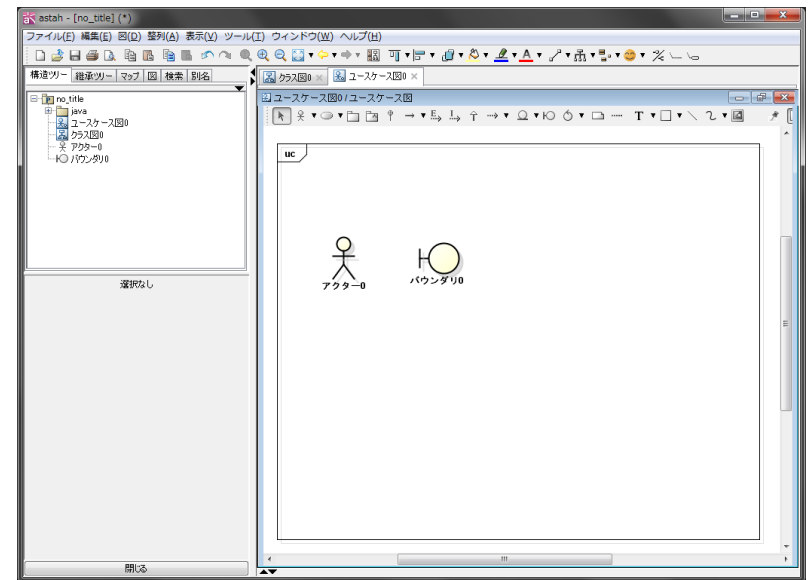
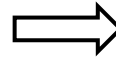
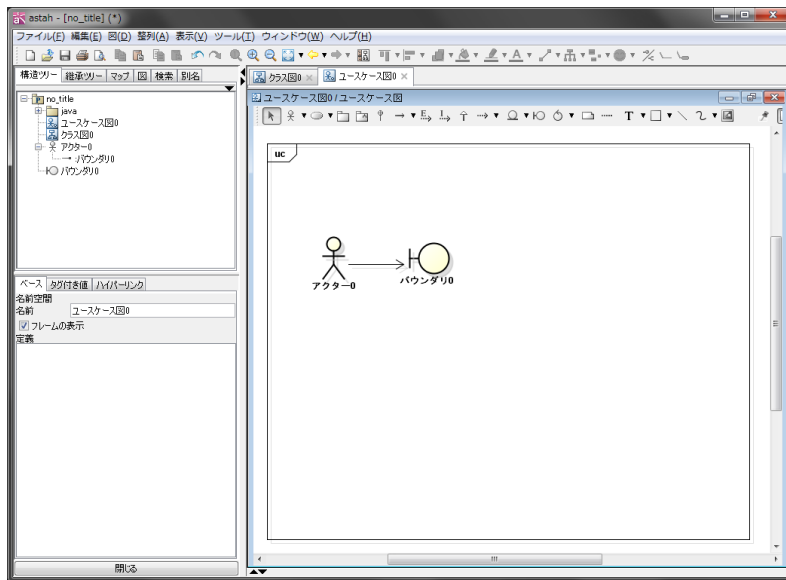
- 関係を結ぶ線を消すときは [ctrl + D] で消す
 - ◆ deleteで消すと、表示上消えているだけです
- ◆ 作成を進める上でdeleteで消しているだけだと色々問題が起きる可能性があるので注意

deleteで消した場合



表示上消えるだけで、関係が残っています

Ctrl + D（もしくは削除）で消した場合



関係ごと消えます