

教室予約システム

## システム構造設計書

## プログラム構造設計書

2023 年 7 月 20 日 第 5.0 版  
2023 年 7 月 18 日 第 4.1 版  
2023 年 6 月 28 日 第 4.0 版  
2023 年 6 月 19 日 第 3.1 版  
2023 年 6 月 15 日 第 3.0 版  
2023 年 6 月 15 日 第 2.2 版  
2023 年 6 月 14 日 第 2.1 版  
2023 年 4 月 24 日 第 2.0 版  
2023 年 4 月 01 日 第 1.1 版  
2023 年 3 月 27 日 第 1.0 版  
2023 年 3 月 24 日 第 0.0 版

東京国際工科専門職大学  
工科学部 情報工学科

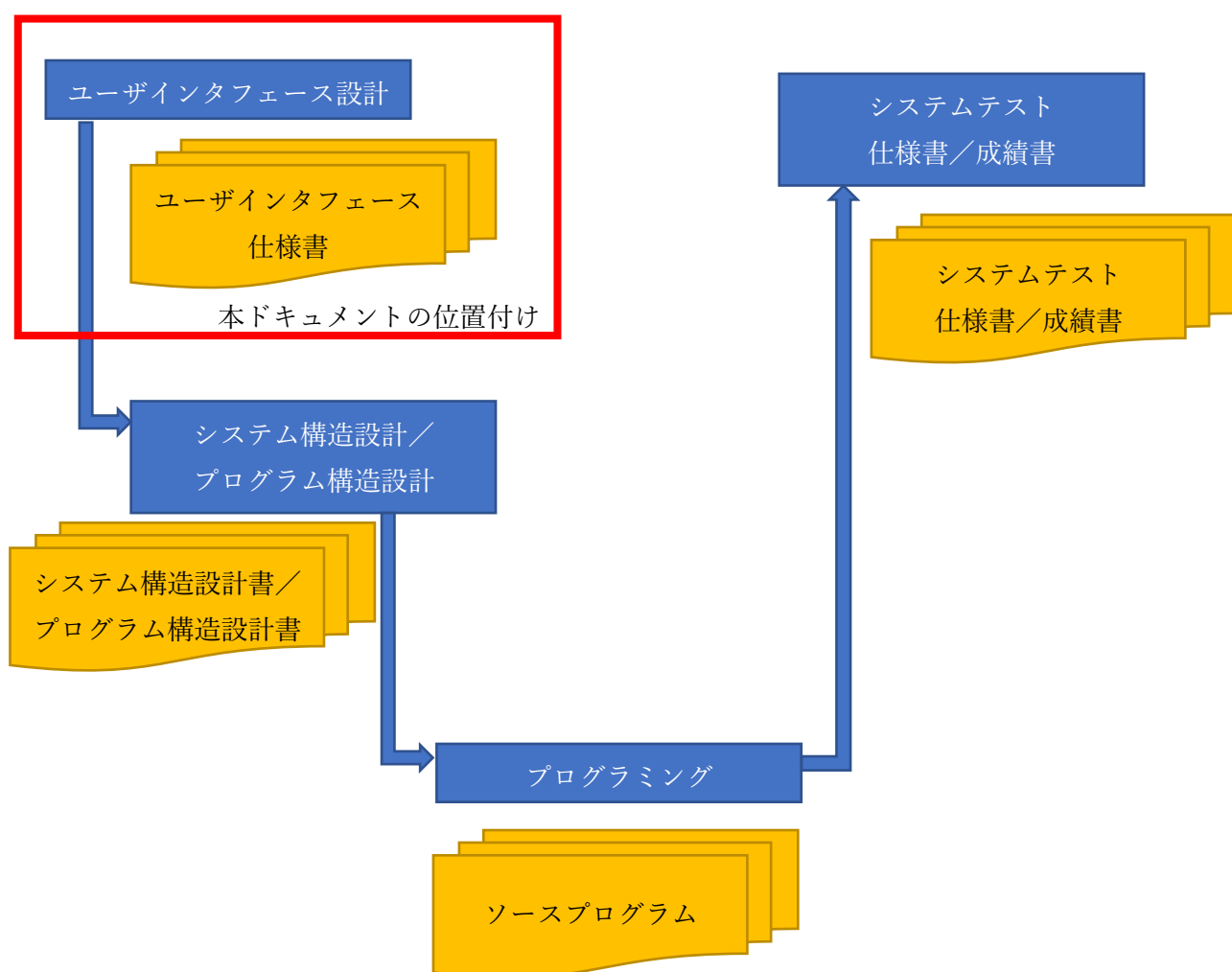
TK210474 田村 輝  
TK210494 森田 李  
TK210505 志水桃子  
TK210508 児玉俊介  
TK210530 齊藤成司  
TK210531 山田達成

はじめに

本ドキュメントは、東京国際工科専門職大学内の教室予約システムに関するソフトウェアシステム構造及びプログラム構造について規定するものである。

なお、本システムは、スパイラルにて開発を行うが、イテレーション毎に本ドキュメントを更新するものとする。ベースとなるユーザインタフェース仕様書の版数については、変更履歴に明記する。

#### 【本プロジェクトにおける工程と成果物】



# 変更履歴

No.	版数	概要	頁	変更者	変更日
1	0.0	ログイン・ログアウト対応版. レビュー直前版発行	全頁	荒金匡徳	2023/3/24
2	1.0	レビュー完了のため正式版としてリリース. ログイン・ログアウト機能対応版ここまで	全頁	荒金匡徳	2023/3/27
3	1.1	教室概要表示機能対応. レビュー直前版発行	全頁	荒金匡徳	2023/4/1
		2. 1 MainFrame の構成を変更, panelNorth の分割と panel 上への部品配置について説明追記	2.1		
		3. クラス図に ChoiceFacility を追加	3.5		
		3.2.MainFrame の追加メソッド・処理追記	3.2		
		3.3.ReservationControl のメソッド追記	3.3		
		3.5.ChoiceFacility 追記	3.5		
		4. シーケンスでMainFrame より起動される機能が増えるため, シーケンスを細分化	4		
4	2.0	レビュー完了のため正式版としてリリース. 教室概要表示対応版ここまで	全頁	荒金匡徳	2022/4/24
	2.1	新規予約クラス(ReservationDialog)のクラス図を作成&メソッド追記	3.6	山田達成 齊藤成司	2023/6/14
		ChoiceHour クラス のクラス図を作成&メソッド追記	3.7		
		ChoiceMinute クラス のクラス図を作成&メソッド追記	3.8		
		予約状況確認(Reservationed1)のクラス図を作成&メソッド追記	3.9		
	2.2	MainFrame のメソッド追記	3.2	山田達成	2023/6/15
		ReservationControl のメソッド追記	3.3		
		MainFrame 等で表示される文字サイズ/	3.2	齊藤成司	

		フォント変更メソッドの追記			
		基本アーキテクチャに新規予約機能(ReservationDialog)と予約状況確認機能(Reservationed1)に関する説明を追記	1	山田達成	2023/6/15
		MainFrame の Window 構成追記	2.1		
		新規予約画面(ReservationDialog)と予約状況確認画面(Reservationed1)の Window 構成作成	2.3		
		全体のクラス図を再構築	3		
		新規予約機能のシーケンス図作成(4.4.1～4.4.5)	4.4		
		予約状況確認機能のシーケンス図作成	4.5		
		4.1～4.3 までのシーケンス図修正/ 各シーケンス図の繋がり修正	4.1～ 4.3		
		予約状況確認機能のシーケンス図作成	4.5	山田達成 齊藤成司	
5	3.0	レビュー完了のため正式版としてリリース。 新規予約機能・予約状況確認機能対応版 ここまで	全頁	山田達成 齊藤成司	2023/6/15
	3.1	MainFrame のメソッド追記	3.2	山田達成 齊藤成司	2023/6/19
		ReservationControl のメソッド追記	3.3	山田達成	2023/6/28
		自己予約確認機能のシーケンス図作成	4.6	山田達成 齊藤成司	
		新規予約機能のシーケンス図の修正	4.4	山田達成	
		各シーケンス図の繋がり修正	4		
6	4.0	レビュー完了のため正式版としてリリース。 新規予約機能・自己予約確認機能対応版 ここまで	全頁	山田達成 齊藤成司	
		基本アーキテクチャに予約キャンセル機能(ReservationCancel)に関する説明を追記	1	山田達成	2023/7/18

		MainFrame の Window 構成追記	2.1		
		予約キャンセル画面(ReservationCancel)の Window 構成作成	2.3		
		全体のクラス図を再構築	3		
		MainFrame クラスのメソッド追記	3.2		
		ReservationControl クラスのメソッド追記	3.3		
		ChoiceReservationId クラスのクラス図を作成&メソッド追記	3.10		
		予約キャンセル(ReservationCancel)のクラス図を作成&メソッド追記	3.11		
7	4.1	MainFrame のメソッド追記	3.2		
		ReservationControl のメソッド追記	3.3		
		自己予約確認機能のシーケンス図作成	4.7		
		新規予約機能のシーケンス図の修正	4.4		
		各シーケンス図の繋がり修正	4		
		構造設計書/プログラム構造設計書の校閲訂正完了		田村輝	2023/7/20
8	5.0	レビュー完了のため正式版としてリリース. 予約キャンセル機能対応版ここまで		山田達成	2023/7/20

## 目次

目次.....	6
1 基本アーキテクチャ.....	9
2 Window 構成.....	11
2.1 MainFrame.....	11
2.2 ログインダイアログ.....	11
2.3 新規予約画面.....	12
2.4 予約状況確認画面.....	12
2.5 予約キャンセル画面.....	13
3 クラス図及び各クラス内の変数及びメソッド.....	13
3.1 ReservationSystem.....	13
3.1.1 main メソッド.....	14
3.2 MainFrame.....	14
3.2.1 MainFrame コンストラクタ.....	15
3.2.2 windowClosing メソッド.....	15
3.2.3 actionPerformed メソッド.....	15
3.3 ReservationControl.....	16
3.3.1 sqlUserID.....	16
3.3.2 sqlPassword.....	16
3.3.3 reservationUserID.....	17
3.3.4 flagLogin.....	17
3.3.5 ReservationControl コンストラクタ.....	17
3.3.6 connectDB メソッド.....	17
3.3.7 closeDB メソッド.....	17
3.3.8 loginLogout メソッド.....	17
3.3.9 getFacilityExplanation メソッド.....	18
3.3.10 getFacilityId メソッド.....	18
3.3.11 getReservationId メソッド.....	18
3.3.12 Reservationed1 メソッド.....	19
3.3.13 make Reservation メソッド.....	19
3.3.14 getAvailableTime メソッド.....	20
3.3.15 getSelfReservationed メソッド.....	20
3.3.16 getReservationCancel メソッド.....	20
3.4 LoginDialog.....	21
3.4.1 canceled.....	21

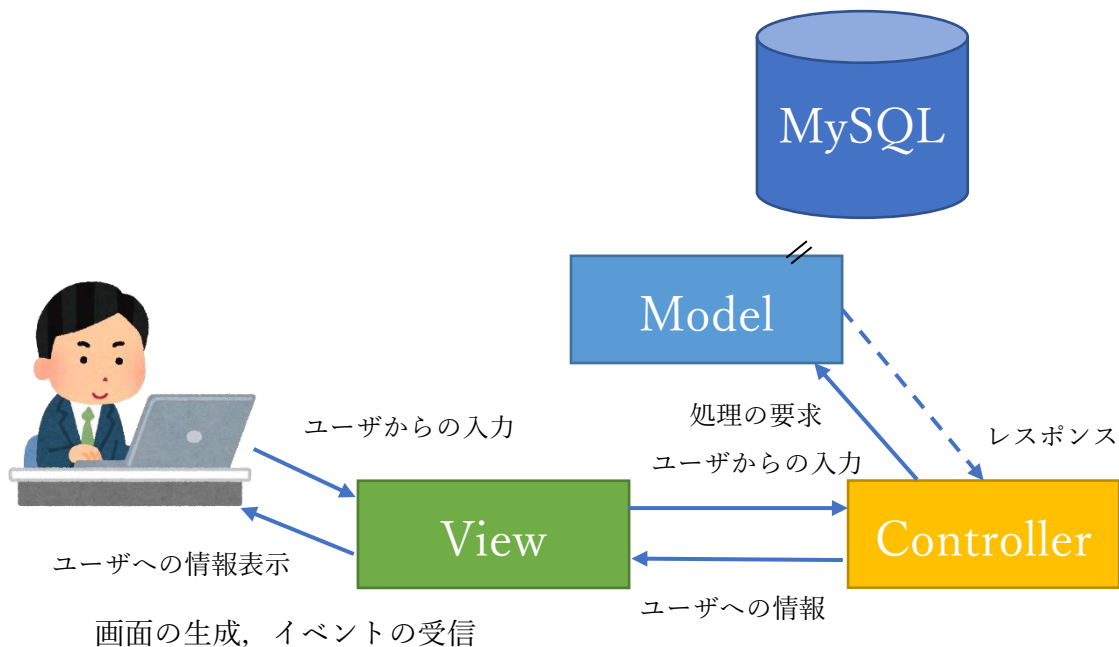
3.4.2	LoginDialog コンストラクタ .....	21
3.4.3	windowClosing メソッド .....	22
3.4.4	actionPerformed メソッド .....	22
3.5	ChoiceFacility .....	22
3.5.1	ChoiceFacility コンストラクタ .....	22
3.6	ReservationDialog .....	23
3.6.1	canceled .....	23
3.6.2	resetTimeRange メソッド .....	24
3.6.3	itemStateChanged メソッド .....	24
3.6.4	WindowClosing メソッド .....	24
3.6.5	actionPerformed メソッド .....	24
3.7	ChoiceHour .....	24
3.7.1	ChoiceHour コンストラクタ .....	24
3.8	ChoiceMinute .....	25
3.8.1	ChoiceHour コンストラクタ .....	25
3.9	Reservationed1 .....	25
3.9.1	canceled .....	26
3.9.2	itemStateChanged メソッド .....	26
3.9.3	WindowClosing メソッド .....	26
3.9.4	actionPerformed メソッド .....	26
3.10	ChoiceReservationId .....	26
3.10.1	ChoiceReservationId コンストラクタ .....	26
3.11	ReservationCancel .....	27
3.9.1	canceled .....	27
3.9.2	itemStateChanged メソッド .....	27
3.9.3	WindowClosing メソッド .....	27
3.9.4	actionPerformed メソッド .....	28
4	シーケンス図 .....	29
4.1	起動 .....	30
4.2	ログイン機能 .....	30
4.2.1	ログインダイアログ表示 .....	30
4.2.2	ID, パスワード入力～認証完了 .....	31
4.2.3	ID, パスワード入力～ユーザ ID 未登録 .....	31
4.2.4	ID, パスワード入力～パスワード不一致 .....	31
4.2.5	認証中止 .....	32
4.3	教室概要表示 .....	32

4.4 新規予約機能.....	32
4.4.1 未ログイン状態からの新規予約ボタン押下.....	32
4.4.2 ログイン後新規予約ボタン押下～予約実行まで.....	33
4.4.3 選択教室変更 .....	33
4.4.4 開始「時」変更～開始「分」変更～予約実行まで.....	33
4.4.5 入力データチェック～日付入力エラー .....	34
4.4.6 入力データチェック～予約時間エラー .....	34
4.4.7 入力データチェック～予約重複エラー .....	35
4.4.8 入力データチェック～過去の日付エラー .....	35
4.4.9 予約中止 .....	35
4.5 予約状況確認機能.....	36
4.5.1 予約確認ダイアログ表示～実行.....	36
4.5.2 入力データチェック～日付入力エラー .....	36
4.5.3 入力データチェック～予約情報なしエラー.....	37
4.5.4 予約確認中止 .....	37
4.6 自己予約確認.....	38
4.6.1 未ログイン状態～自己予約確認ボタン押下.....	38
4.6.2 自己予約確認ボタン押下～自己予約確認実行.....	38
4.6.3 入力データチェック～予約情報なしエラー.....	38
4.7 予約キャンセル .....	39
4.7.1 未ログイン状態～予約キャンセルボタン押下.....	39
4.7.2 予約キャンセルボタン押下～予約キャンセル実行.....	39
4.7.3 選択予約 ID 変更 .....	40
4.7.4 入力データチェック～別ユーザエラー .....	40
4.7.5 予約キャンセル中止 .....	41



## 1 基本アーキテクチャ

本システムは、下図の様な MVC アーキテクチャにより構成する。



ReservationSystem クラスが View である MainFrame クラス及び Controller である ReservationControl クラスのインスタンスを生成する。

MainFrame クラスは MainFrame の描画を行い、以降ユーザからの入力を受け取り、ReservationControl クラスに入力された情報を渡ししながら、そのレスポンスを表示する。

ReservationControl クラスは、インスタンス生成時に Model である MySQL との通信を確立し、以降 MainFrame からユーザからの入力を受け取ると、それに応じた処理を行い、結果を MainFrame に返す。

ユーザが「ログインボタン」を押下すると、MainFrame はそれを ReservationControl に伝え、ReservationControl クラスでは、もう 1 つの View である LoginDialog クラスのインスタンスを生成し、以降ログイン処理が終了するまでの間、ReservationControl は LoginDialog クラスより、ユーザの入力を受け付け、処理を行う。

コンボボックスへのアイテム登録は、MainFrame クラス内で行うと煩雑になるため、choice の派生クラスを作成し、利用する事とする。今回は ChoiceFacility クラスが該当する。

ユーザが「新規予約ボタン」を押下すると、MainFrame はそれを ReservationControl に伝え、ReservationControl クラスでは、もう 1 つの View である ReservationDialog クラスのインスタンスを生成し、以降ログイン処理が終了するまでの間、ReservationControl は ReservationDialog クラスより、ユーザの入力を受け付け、処理

を行う。

コンボボックスへのアイテム登録は、MainFrame クラス内で行うと煩雑になるため、ChoiceFacility, ChoiceHour, ChoiceMinute クラスを利用する。

ユーザが「予約状況確認ボタン」を押下すると、MainFrame はそれをReservationControl に伝え、ReservationControl クラスでは、もう1つの View であるReservationed1 クラスのインスタンスを生成し、以降ログイン処理が終了するまでの間、ReservationControl は Reservationed1 クラスより、ユーザの入力を受け付け、処理を行う。

コンボボックスへのアイテム登録は、MainFrame クラス内で行うと煩雑になるため、ChoiceFacility, ChoiceHour, ChoiceMinute クラスを利用する。

ユーザが「予約キャンセルボタン」を押下すると、MainFrame はそれをReservationControl に伝え、ReservationControl クラスでは、もう1つの View であるReservationCancel クラスのインスタンスを生成し、以降ログイン処理が終了するまでの間、ReservationControl は ReservationCancel クラスより、ユーザの入力を受け付け、処理を行う。

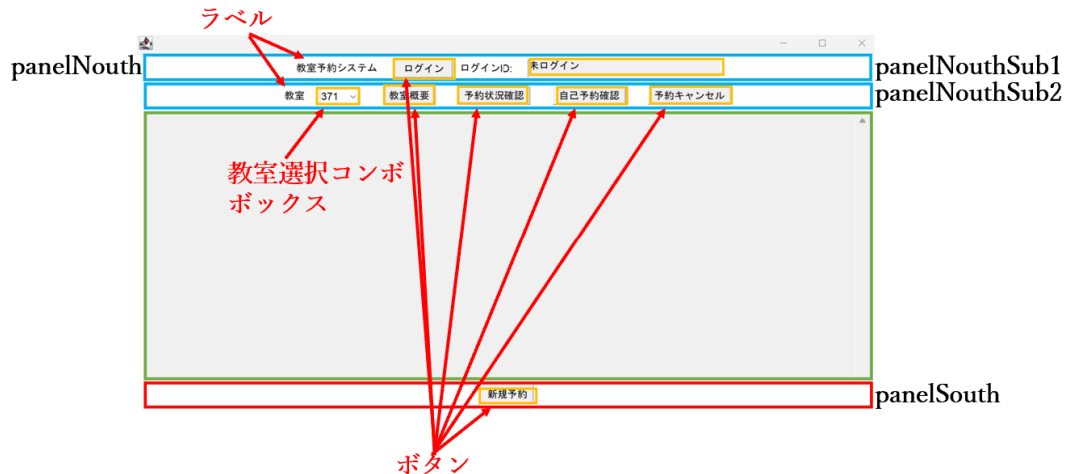
コンボボックスへのアイテム登録は、MainFrame クラス内で行うと煩雑になるため、ChoiceReservationId クラスを利用する。

2

### 3 Window 構成

本システムで使用する Window の部品配置構成を以下に示す.

#### 3.1 MainFrame



MainFrame は BorderLayout 配置とし, 上記の様なパネル配置とする.

panelNorthSub1 への配置部品

教室予約システムラベル, ログインボタン, ログイン ID ラベル, ログイン ID テキストフィールド

panelNorthSub2 への配置部品

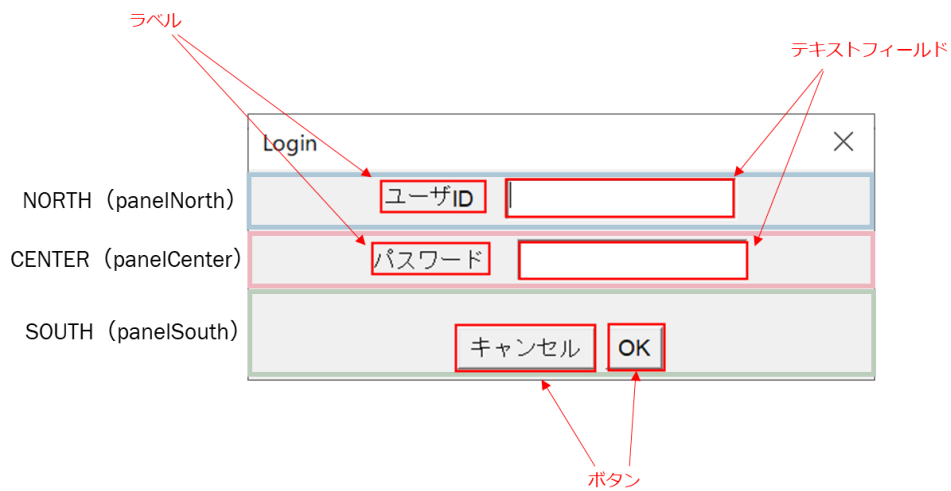
教室ラベル, 教室選択コンボボックス, 教室概要ボタン, 予約情報確認ボタン, 自己予約確認ボタン

panelCenter への配置部品

テキストエリア (20 行×30 文字)

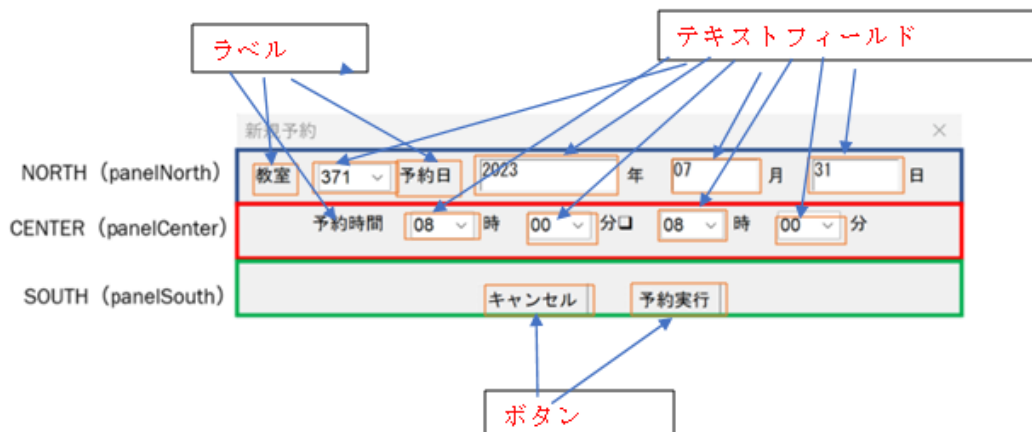
#### 3.2 ログインダイアログ

LoginDialog は, MainFrame と同様 BorderLayout 配置とし, 下記の様なパネル配置及び部品配置とする.



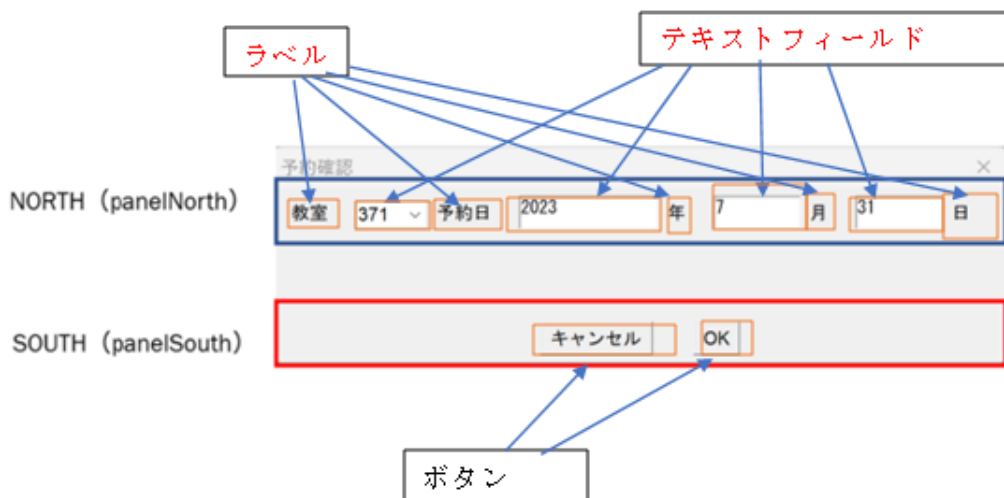
### 2.3 新規予約画面

新規予約画面(ReservationDialog)は下記のような部品配置とする。



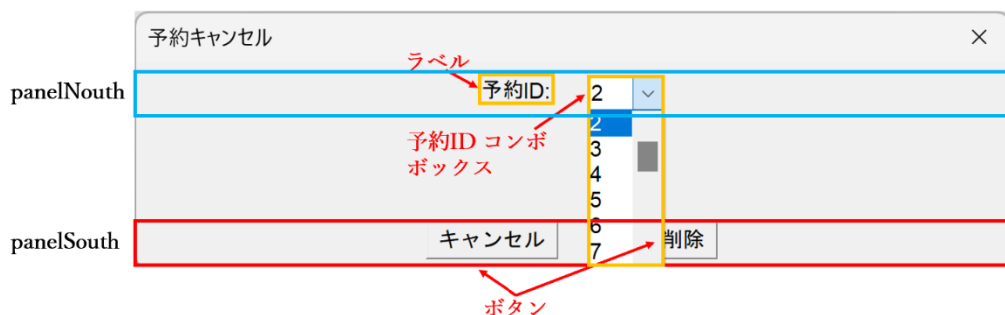
### 2.4 予約状況確認画面

予約状況確認(Reservationed1)は下記のような部品配置とする。



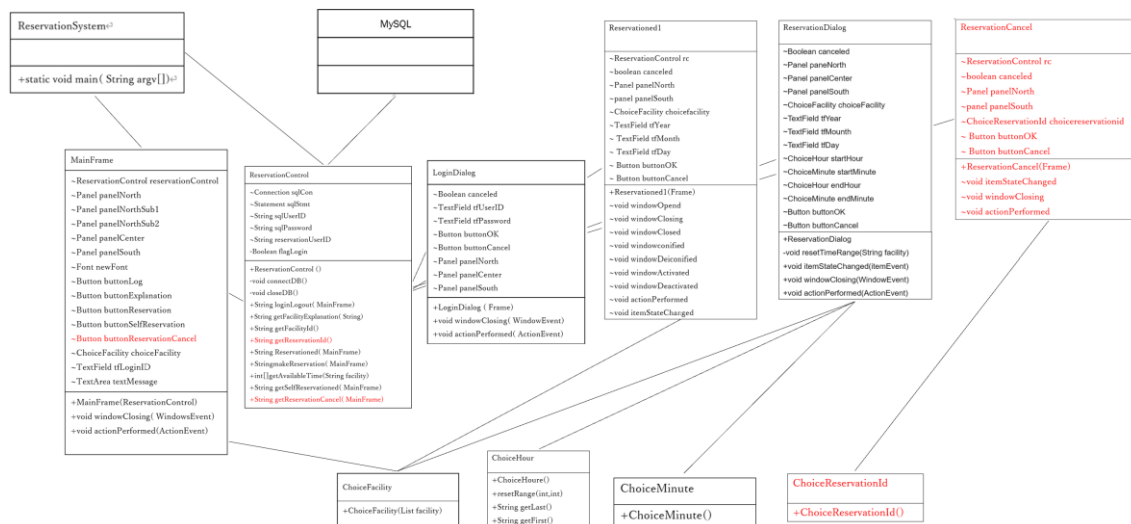
## 2.5 予約キャンセル画面

予約キャンセル(ReservationCancel)は下記のような部品配置とする。

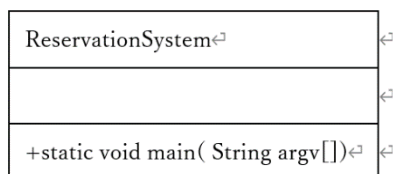


## 4 クラス図及び各クラス内の変数及びメソッド

全体のクラス図を以下に示す。



### 4.1 ReservationSystem



ReservationSystem クラスは、本システムを起動するためのクラスで、MainFrame クラス及び ReservationControl クラスを起動する役割を担う。それ以外の機能は持たない。

#### 4.1.1 main メソッド

MainFrame クラス及び ReservationControl クラスのインスタンスを生成し，MainFrame の Window を表示，可視化を行う．

### 4.2 MainFrame

MainFrame
~ReservationControl reservationControl ~Panel panelNorth ~Panel panelNorthSub1 ~Panel panelNorthSub2 ~Panel panelCenter ~Panel panelSouth ~Font newFont ~Button buttonLog ~Button buttonExplanation ~Button buttonReservation ~Button buttonSelfReservation ~Button buttonReservationCancel ~ChoiceFacility choiceFacility ~TextField tfLoginID ~TextArea textMessage
+MainFrame(ReservationControl) +void windowClosing( WindowsEvent) +void actionPerformed(ActionEvent)

MainFrame class は，本システムの基本となる Window を司るクラスである．MainFrame の各部品配置を行い，ユーザからのデータ入力及びボタン押下等のアクションに対して，該当する ReservationControl class のメソッドを呼び出すことで，対応する処理を行う．

#### 4.2.1 MainFrame コンストラクタ

main より ReservationControl クラスのインスタンスを引数として受け取る.

コンストラクタ内で, MainFrame の Window 及び部品配置を行う. 教室選択のコンボボックスは, Choice クラスの派生クラスとして作成する ChoiceFacility クラスのインスタンスを生成し, 実現する. ChoiceFacility クラスのインスタンス生成時, ReservationControl の getFacilityExplanation メソッドにより全ての facility\_id を取得し, ChoiceFacility のインスタンスに渡す. また, ログインボタン, 教室概要ボタン, 新規予約ボタン, 予約状況確認ボタン, 自己予約確認ボタン及び終了ボタンを有効にするため, これらを ActionListener 及び WindowListener に登録する.

#### 4.2.2 windowClosing メソッド

windowClosing メソッドは, 終了ボタン押下時に呼び出され, 本システムを終了させる.

#### 3.2.3 actionPerformed メソッド

actionPerformed メソッドは, ボタン押下時に呼出される.

ログインボタン押下時: reservationControl クラスの loginLogout メソッドを呼び出す. この時, 戻り値を結果表示用テキストエリアに表示させる.

教室概要ボタン押下時: reservationControl インスタンスの getFacilityExplanation メソッドを呼び出す. この時, 戻り値を結果表示エリアに表示させる.

新規予約ボタン押下時: reservationControl インスタンスの makeReservation メソッドを呼び出す. この時, 戻り値を結果表示エリアに表示させる.

予約状況確認ボタン: reservationControl インスタンスの Reservationed1 メソッドを呼び出す. この時, 戻り値を結果表示エリアに表示させる.

自己予約確認ボタン: reservationControl インスタンスの getSelfReservationed メソッドを呼び出す. この時, 戻り値を結果表示エリアに表示させる.

### 4.3 ReservationControl

ReservationControl
~Connection sqlCon ~Statement sqlStmt ~String sqlUserID ~String sqlPassword ~String reservationUserID -Boolean flagLogin
+ReservationControl () -void connectDB() -void closeDB() +String loginLogout( MainFrame) +String getFacilityExplanation( String) +String getFacilityId() <b>+String getReservationId()</b> +String Reservationed( MainFrame) +String makeReservation( MainFrame) +int[]getAvailableTime(String facility) +String getSelfReservationed( MainFrame) <b>+String getReservationCancel( MainFrame)</b>

ReservationControl class は、本システムにてユーザが求める機能に対する処理を行うクラスである。MainFrame から、ボタン押下等の機能実行要求があった際に、本クラスのメソッドが呼出され、処理に必要な情報を MainFrame の TextField 等の部品や MySQL との通信を行うことで取得する。また処理した結果を MainFrame に表示する際は、本クラスのメソッドからの戻り値として文字列を返すことで実現する。その他、必要に応じて、MySQL に対してデータの書き込みも行う。

#### 4.3.1 sqlUserID

MySQL にログインするための userID 格納変数. "puser"を入れておく。

#### 4.3.2 sqlPassword

MySQL にログインするための userID に対応するパスワード格納変数, "1234"を入れておく。



#### 4.3.3 reservationUserID

本システムにログイン中のユーザ ID を格納する変数. ログイン中のみ有効となる.

#### 4.3.4 flagLogin

本システムにユーザがログインしている状態かどうかを表す変数. ログイン中は true の値を取る.

#### 4.3.5 ReservationControl コンストラクタ

flagLogin に false を設定し, 非ログイン状態とする.

#### 4.3.6 connectDB メソッド

MySQL のドライバを Load し, その後 MySQL との connection を確立する.

connection 確立後, Statement Object を生成する.

上記いずれかの際に exception が発生した場合は StackTrace をコンソールに表示する.

#### 4.3.7 closeDB メソッド

Statement Object を Close し, MySQL との connection も close する.

上記いずれかの際に exception が発生した場合は StackTrace をコンソールに表示する.

#### 4.3.8 loginLogout メソッド

MainFrame のログインボタン押下時に MainFrame により呼出される.

ログイン状態時の処理:

flagLogin に false を設定し, 非ログイン状態にする.

また, ログインボタンの表記を「ログイン」に変更する.

ログイン ID テキストエリアの表示を「未ログイン」に変更する.

非ログイン状態時の処理:

ログインダイアログを生成し, 制御をログインダイアログに移す.

ログイン操作キャンセルでダイアログが閉じられた時:

戻り値の文字列に NULL を設定し, 処理を終了させる.

ログイン操作実行でダイアログが閉じられた時:

ログインダイアログで入力されたユーザ ID とパスワードを取得す

る.

MySQL との Connection を確立する.

ログインダイアログに入力されたユーザ ID と一致する情報を user テーブルより取得.

MySQL に該当する情報がない時:

戻り値に「登録されていません」を設定する.

ログインダイアログに入力されたパスワードと user テーブルのパスワードを比較する.

パスワードが一致:

ログイン状態 (flagLogin) に true を設定する.

ログインボタンの表記を「ログアウト」に変更する.

ログイン ID テキストフィールドにログインダイアログに入力したログイン ID を設定する.

戻り値の文字列に NULL を設定する.

パスワードが不一致: 戻り値に「ID またはパスワードが違います。」を設定する.

MySQL との Connection を Close する.

MySQL アクセス時に exception が発生した場合は, StackTrace をコンソールに表示する.

#### 4.3.9 getFacilityExplanation メソッド

MainFrame の教室概要ボタン押下時に MainFrame により呼び出され, 引数として教室 ID を受け取る.

本メソッドでは, MySQL に接続し, 受け取った教室 ID で facility テーブルを検索し, 該当するデータの explanation, open\_time, close\_time より, 表示用の教室概要データを作成し, 戻り値として返す.

MySQL との通信で exception が発生した場合は, コンソールに StackTrace を表示する.

#### 3.3.10 getFacilityId メソッド

このメソッドを呼び出すと, データベースからすべての facility\_id が取得され, それらが格納されたリストが返される.

#### 3.3.11 getReservationId メソッド

このメソッドを呼び出すと, データベースからすべての reservation\_id が

取得され、それらが格納されたリストが返される。

### 3.3.12 Reservationed1 メソッド

MainFrame の予約状況確認ボタン押下時に MainFrame により呼出される。  
予約確認画面の生成、表示。  
予約確認画面から年月日を取得。  
月と日が一桁だった場合に、前に「0」を付加。  
入力された日付の書式が違う場合：  
戻り値に「日付の書式を修正してください（年:西暦 4 桁, 月:1～12, 日:1～31(毎月末日まで))」を設定する。  
入力された日付の値が違う場合：  
戻り値に「日付の値を修正してください」を設定する。  
予約確認画面から教室名・開始時刻・終了時刻を取得。

### 3.3.13 make Reservation メソッド

新規予約画面の生成・表示。  
新規予約画面から年月日の取得。  
月と日が一桁だった場合に、前に「0」を付加。  
入力された日付の書式が違う場合：  
戻り値に「日付の書式を修正してください（年:西暦 4 桁, 月:1～12, 日:1～31(毎月末日まで))」を設定する。  
入力された日付の値が違う場合：  
戻り値に「日付の値を修正してください」を設定する。  
開始時刻の方が早く設定した場合：  
戻り値に「開始時刻と終了時刻が同じか終了時刻の方が早くなっています」を設定する。  
過去の日付を設定した場合：  
戻り値に「過去の日付は予約できません」を設定する。  
予約情報を SQL の reservation テーブルに書き込む場合：  
戻り値に「予約されました」を設定する。  
予約情報を SQL の reservation テーブルに書き込みができなかった場合：  
戻り値に「データベース接続または SQL 実行中にエラーが発生しました」を設定。  
ログインしていない場合：

戻り値に「ログインしてください」を設定.

#### 3.3.14 getAvailableTime メソッド

SQL の facility テーブルの open\_Time, close\_Time から指定された教室の利用可能開始・終了時間を取得する

#### 3.3.15 getSelfReservationed メソッド

ログインしていない場合:

戻り値に「ログインしてください」を設定.

SQL の reservation テーブルからログインしているユーザの予約情報を記載する.

#### 3.3.16 getReservationCancel メソッド

ログインしていない場合:

戻り値に「ログインしてください」を設定.

SQL の reservation テーブルから選択した reservation\_id の予約情報をキャンセルした内容を記載する.

#### 4.4 LoginDialog

LoginDialog
~Boolean canceled ~TextField tfUserID ~TextField tfPassword ~Button buttonOK ~Button buttonCancel ~Panel panelNorth ~Panel panelCenter ~Panel panelSouth
+LoginDialog ( Frame) +void windowClosing( WindowEvent) +void actionPerformed( ActionEvent)

LoginDialog class は、ユーザがログインする際にユーザ ID やパスワードを入力するためのダイアログについて、View の役割を果たす。ユーザ ID 及びパスワード入力後、本ダイアログが閉じられた要因に従い、変数 canceled にログイン処理を行うかキャンセルするかを設定する。

##### 4.4.1 canceled

この変数はログインダイアログを閉じる際に、ユーザがログイン処理を行うかログイン処理をキャンセルするかを決めるためのものである。ログインダイアログが終了ボタンもしくはキャンセルボタンで閉じられた場合は true が設定され、以降ログイン処理はキャンセル扱いとなる。また、OK ボタンにて閉じられた場合は false が設定され、ログイン認証動作に移行する。

##### 4.4.2 LoginDialog コンストラクタ

本クラスは、Dialog クラスの派生クラスで、ActionListener 及び WindowListener を利用する。引数で MainFrame のインスタンスを取得し、親クラスである Dialog のコンストラクタを呼び出し、本ダイアログの所有者、タイトル、モーダル指定を行う。

(所有者：MainFrame, タイトル：Login, モーダル指定：有効 (true))  
ダイアログの各種部品配置及び ActionListener や WindowListener へのボタ

ン登録を行う。(ActionListener へは, キャンセルボタン及び OK ボタンが設定される.)

#### 4.4.3 windowClosing メソッド

windowClosing メソッドは, 終了ボタン押下時に呼出され, ログインダイアログを非表示に設定, canceled 変数に true を設定, ダイアログを削除する.

#### 4.4.4 actionPerformed メソッド

actionPerformed メソッドは, キャンセルボタンもしくは OK ボタン押下時に呼出される.

canceled 変数に, キャンセルボタン押下時は true, OK ボタン押下時は false が設定され, その後ログインダイアログを非表示にし, ダイアログを削除する.

### 4.5 ChoiceFacility

ChoiceFacility
+ChoiceFacility(List facility)

ChoiceFacility class は, Choice クラスを継承した派生クラスで, 教室を選択するコンボボックスの機能を提供する. 本クラス独自のメソッドは持たず, 選択された教室 ID は Choice クラスの getSelectedItem メソッドにて取得することが出来る.

#### 4.5.1 ChoiceFacility コンストラクタ

呼出元から受け取ったリストにある facility\_id を全て choice のアイテムに加える.

### 3.6 ReservationDialog

ReservationDialog
<div>~Boolean canceled</div> <div>~Panel paneNorth</div> <div>~Panel panelCenter</div> <div>~Panel panelSouth</div> <div>~ChoiceFacility choiceFacility</div> <div>~TextField tfYear</div> <div>~TextField tfMounth</div> <div>~TextField tfDay</div> <div>~ChoiceHour startHour</div> <div>~ChoiceMinute startMinute</div> <div>~ChoiceHour endHour</div> <div>~ChoiceMinute endMinute</div> <div>~Button buttonOK</div> <div>~Button buttonCancel</div>
<div>+ReservationDialog</div> <div>-void resetTimeRange(String facility)</div> <div>+void itemStateChanged(itemEvent)</div> <div>+void windowClosing(WindowEvent)</div> <div>+void actionPerformed(ActionEvent)</div>

ReservationDialog class は、予約ダイアログの表示と操作を行うための GUI コンポーネントが配置されている。ダイアログ内の入力や選択肢の変更に応じて、適切な処理が実行されるような仕組みになっている。「キャンセル」ボタンがクリックされた場合は、ダイアログを非表示にし、リソースを解放する。「OK」ボタンがクリックされた場合は、変数 canceled の値を false に設定し、ダイアログを非表示にし、リソースを解放する。

#### 3.6.1 canceled

この変数は Reservation ダイアログを閉じる際に、ユーザがログイン処理を行うかログイン処理をキャンセルするかを決めるためのものである。Reservation ダイアログ面が終了ボタンもしくはキャンセルボタンで閉じられた場合は true が設定され、以降ログイン処理はキャンセル扱いとなる。また、OK ボタンにて閉じられ

た場合は false が設定され、ログイン認証動作に移行する。

### 3.6.2 resetTimeRange メソッド

resetTimeRange メソッドは、指定された教室の利用可能な開始時刻と終了時刻を取得する。取得した時刻の範囲を元に、開始時刻と終了時刻の選択ボックスの範囲を設定する。

### 3.6.3 itemStateChanged メソッド

itemStateChanged メソッドは、選択された教室・利用開始時刻・利用終了時刻が変更された時に値や選択コンボボックスをリセットし、範囲を調整する。

### 3.6.4 WindowClosing メソッド

windowClosing メソッドは、終了ボタン押下時に呼出され、Reservationed1 ログを非表示に設定、canceled 変数に true を設定、ダイアログを削除する。

### 3.6.5 actionPerformed メソッド

actionPerformed メソッドは、キャンセルボタンもしくは OK ボタン押下時に呼出される。

canceled 変数に、キャンセルボタン押下時は true、OK ボタン押下時は false が設定され、その後 Reservation ログを非表示にし、ダイアログを削除する。

## 3.7 ChoiceHour

ChoiceHour
+ChoiceHoure() +resetRange(int,int) +String getLast() +String getFirst()

ChoiceHour class は、Choice クラスを継承した派生クラスで、指定できる「時間」の範囲を設定する機能を提供する。

### 3.7.1 ChoiceHour コンストラクタ

選択された時刻の範囲は Choice クラスの getSelectedItem メソッドにて取得することが出来る。



### 3.8 ChoiceMinute

ChoiceMinute
+ChoiceMinute()

ChoiceHour class は, Choice クラスを継承した派生クラスで, 指定できる「分」の範囲を設定する機能を提供する.

#### 3.8.1 ChoiceHour コンストラクタ

選択された時刻の範囲は Choice クラスの `getSelectedItem` メソッドにて取得することが出来る.

### 3.9 Reservationed1

Reservationed1
~ReservationControl rc ~boolean canceled ~Panel panelNorth ~panel panelSouth ~ChoiceFacility choicefacility ~TextField tfYear ~ TextField tfMonth ~ TextField tfDay ~ Button buttonOK ~ Button buttonCancel
+Reservationed1(Frame) ~void itemStateChanged ~void windowClosing ~void actionPerformed

Reservationed1 class は、予約状況確認ダイアログの表示と操作を行うための GUI コンポーネントが配置されている.ダイアログ内の入力や選択肢の変更に応じて、適切な処理が実行されるような仕組みになっている。「キャンセル」ボタンがクリックされた場合は、ダイアログを非表示にし、リソースを解放する。「OK」ボタンがクリックされた場合は、変数 `canceled` の値を `false` に設定し、ダイアログを非表示にし、リソースを解放する.

### 3.9.1 canceled

この変数は Reservationed1 ダイアログを閉じる際に、ユーザがログイン処理を行うかログイン処理をキャンセルするかを決めるためのものである。Reservationed1 ダイアログ面が終了ボタンもしくはキャンセルボタンで閉じられた場合は true が設定され、以降ログイン処理はキャンセル扱いとなる。また、OK ボタンにて閉じられた場合は false が設定され、ログイン認証動作に移行する。

### 3.9.2 itemStateChanged メソッド

itemStateChanged メソッドは、選択された教室・利用開始時刻・利用終了時刻が変更された時に値や選択コンボボックスをリセットし、範囲を調整する。

### 3.9.3 WindowClosing メソッド

windowClosing メソッドは、終了ボタン押下時に呼出され、Reservationed1 ログを非表示に設定、canceled 変数に true を設定、ダイアログを削除する。

### 3.9.4 actionPerformed メソッド

actionPerformed メソッドは、キャンセルボタンもしくは OK ボタン押下時に呼出される。

canceled 変数に、キャンセルボタン押下時は true、OK ボタン押下時は false が設定され、その後 Reservationed1 ログを非表示にし、ダイアログを削除する。

## 3.10 ChoiceReservationId

ChoiceReservationId
+ChoiceReservationId()

ChoiceReservationId class は、Choice クラスを継承した派生クラスで、DB にある reservation\_id の範囲を設定する機能を提供する。

### 3.10.1 ChoiceReservationId コンストラクタ

選択された時刻の範囲は Choice クラスの getSelectedItem メソッドにて取得することが出来る。

### 3.11 ReservationCancel

ReservationCancel
~ReservationControl rc ~boolean canceled ~Panel panelNorth ~panel panelSouth ~ChoiceReservationId choicereservationid ~ Button buttonOK ~ Button buttonCancel
+ReservationCancel(Frame) ~void itemStateChanged ~void windowClosing ~void actionPerformed

ReservationCancel class は、予約キャンセルダイアログの表示と操作を行うための GUI コンポーネントが配置されている。ダイアログ内の入力や選択肢の変更に応じて、適切な処理が実行されるような仕組みになっている。「キャンセル」ボタンがクリックされた場合は、ダイアログを非表示にし、リソースを解放する。「予約削除」ボタンがクリックされた場合は、変数 canceled の値を false に設定し、ダイアログを非表示にし、リソースを解放する。

#### 3.9.1 canceled

この変数は ReservationCancel ダイアログを閉じる際に、ユーザが予約キャンセル処理を行うか予約キャンセル処理をキャンセルするかを決めるためのものである。ReservationCancel ダイアログ画面が終了ボタンもしくはキャンセルボタンで閉じられた場合は true が設定され、以降予約キャンセル処理はキャンセル扱いとなる。また、OK ボタンにて閉じられた場合は false が設定され、予約キャンセル動作に移行する。

#### 3.9.2 itemStateChanged メソッド

itemStateChanged メソッドは、選択された予約 ID が変更された時に選択コンボボックスをリセットし、範囲を調整する。

#### 3.9.3 WindowClosing メソッド

WindowClosing メソッドは, 終了ボタン押下時に呼出され, ReservationCancel ログを非表示に設定, canceled 変数に true を設定, ダイアログを削除する.

#### 3.9.4 actionPerformed メソッド

actionPerformed メソッドは, キャンセルボタンもしくは OK ボタン押下時に呼出される.

canceled 変数に, キャンセルボタン押下時は true, OK ボタン押下時は false が設定され, その後 ReservationCancel ログを非表示にし, ダイアログを削除する.

## 4 シーケンス図

以下にシーケンスのパターンを列挙し，次ページよりシーケンス図を示す．

### 4.1 起動

### 4.2 ログイン機能

#### 4.2.1 ログインダイアログ表示

#### 4.2.2 ID,パスワード入力～認証完了

#### 4.2.3 ID, パスワード入力～ユーザ ID 未登録

#### 4.2.4 ID, パスワード入力～パスワード不一致

#### 4.2.5 認証中止

### 4.3 教室概要表示

### 4.4 新規予約機能

#### 4.4.1 未ログイン状態～新規予約ボタン押下

#### 4.4.2 ログイン後新規予約ボタン押下～予約実行まで

#### 4.4.3 選択教室変更

#### 4.4.4 開始「時」変更～開始「分」変更～予約実行まで

#### 4.4.5 入力データチェック～日付入力エラー

#### 4.4.6 入力データチェック～予約時間エラー

#### 4.4.7 入力データチェック～予約重複エラー

#### 4.4.8 入力データチェック～過去の日付エラー

#### 4.4.9 予約中止

### 4.5 予約状況確認機能

#### 4.5.1 予約確認ダイアログ表示

#### 4.5.2 入力データチェック～日付入力エラー

#### 4.5.3 入力データチェック～予約情報なしエラー

#### 4.5.4 予約確認中止

### 4.6 自己予約確認機能

#### 4.6.1 未ログイン状態～自己予約確認ボタン押下

#### 4.6.2 自己予約確認ボタン押下～自己予約確認実行

#### 4.6.3 入力データチェック～予約情報なしエラー

### 4.7 予約キャンセル機能

#### 4.7.1 未ログイン状態～予約キャンセルボタン押下

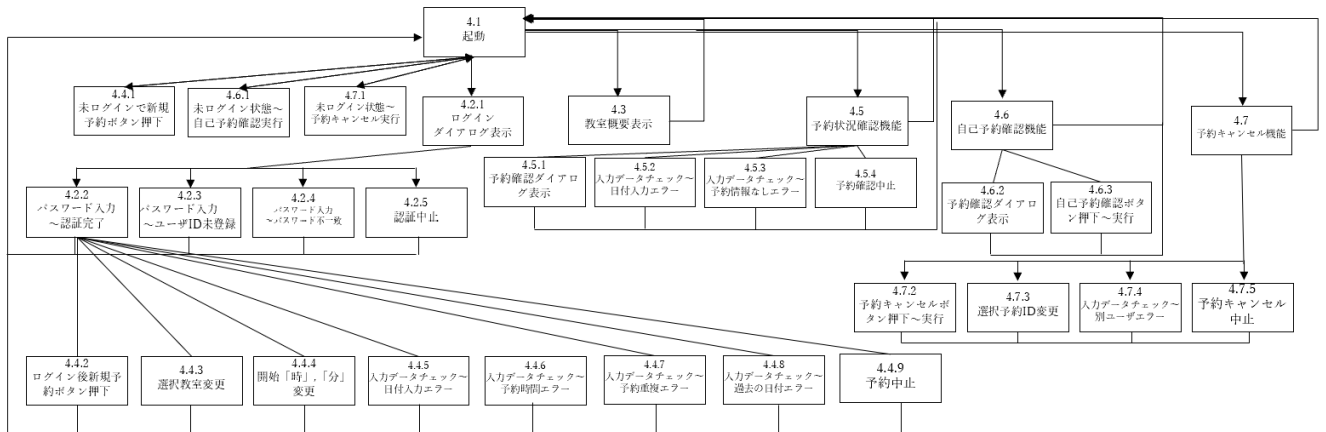
#### 4.7.2 予約キャンセルボタン押下～予約キャンセル実行

#### 4.7.3 選択予約 ID 変更

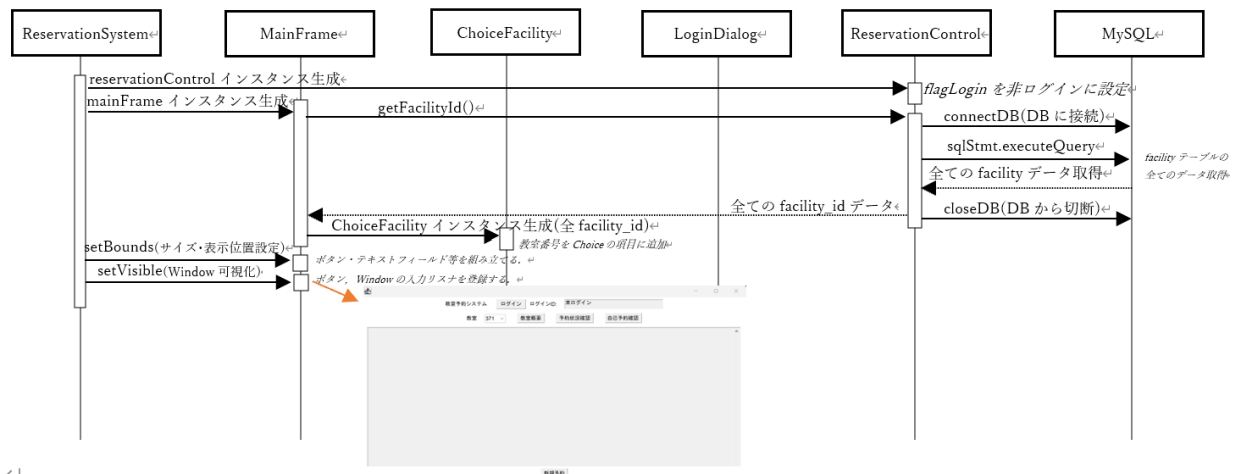
#### 4.7.4 入力データチェック～別ユーザエラー

## 4.7.5 予約キャンセル中止

各シーケンス間の繋がりを以下に示す。



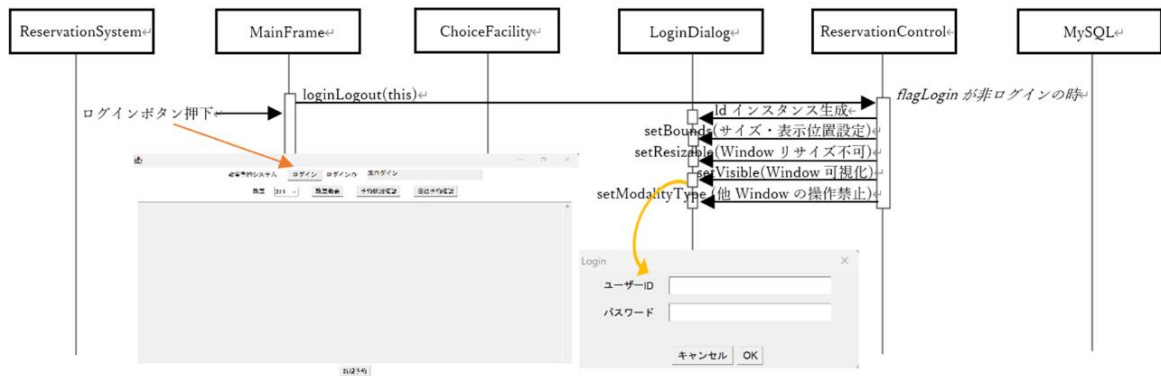
## 4.1 起動



## 4.2 ログイン機能

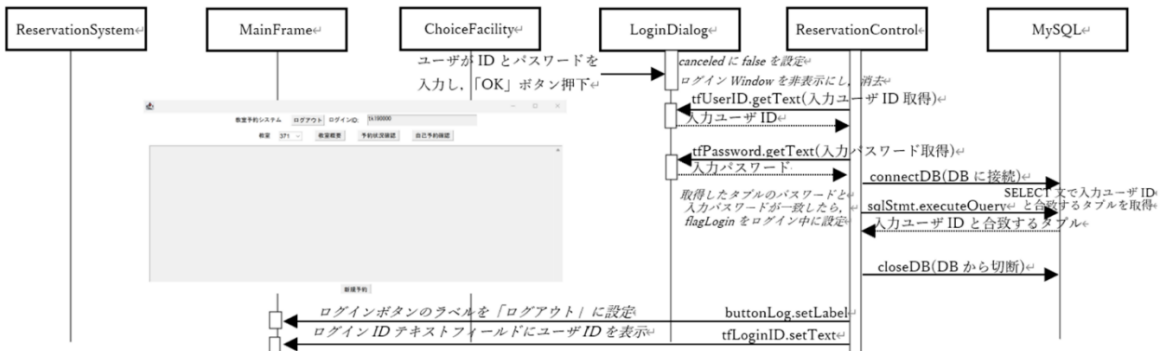
### 4.2.1 ログインダイアログ表示

本シーケンスは、MainFrame 表示中かつ非ログイン状態のものである。(ex.4.1 起動からの継続)



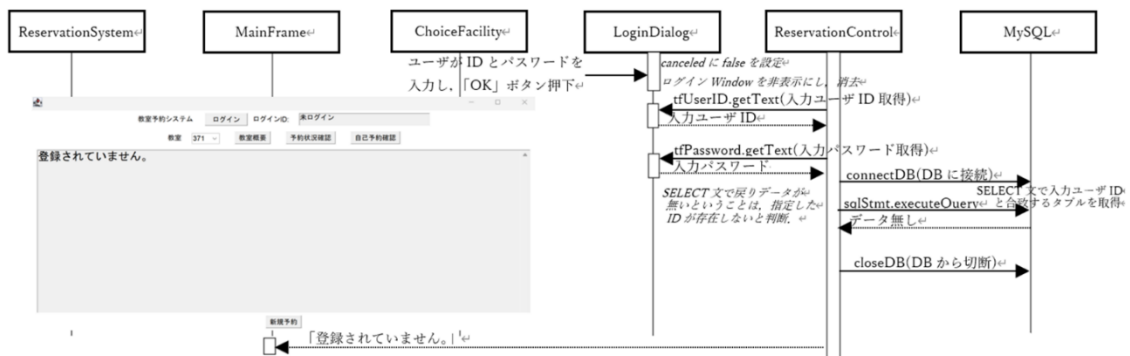
#### 4.2.2 ID, パスワード入力～認証完了

本シーケンスは、ログインダイアログ表示中に正しいパスワードを入力し、OK ボタンを押下した時のものである。(ex.4.2 ログインダイアログ表示からの継続)



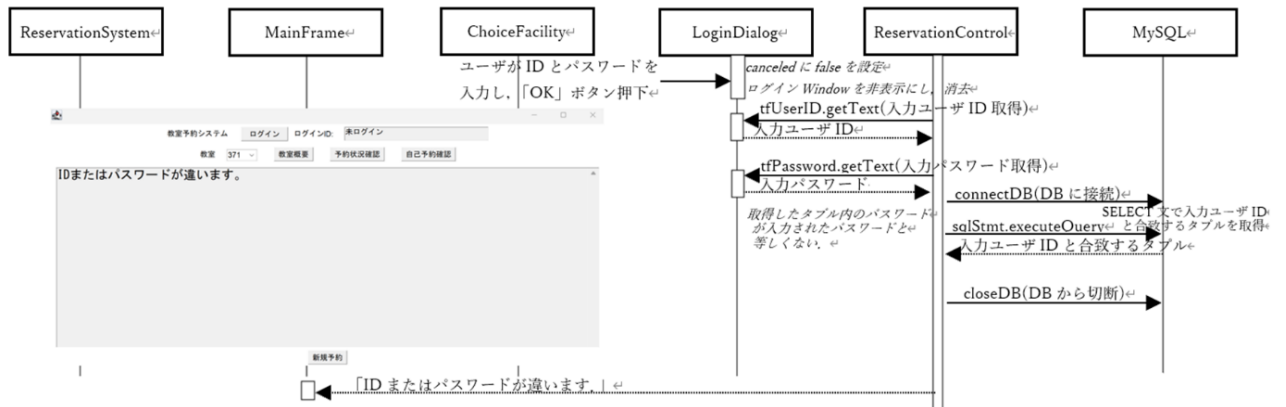
#### 4.2.3 ID, パスワード入力～ユーザ ID 未登録

本シーケンスは、ログインダイアログ表示中に未登録のユーザ ID を入力し、OK ボタンを押下した時のものである。(ex.4.2 ログインダイアログ表示からの継続)



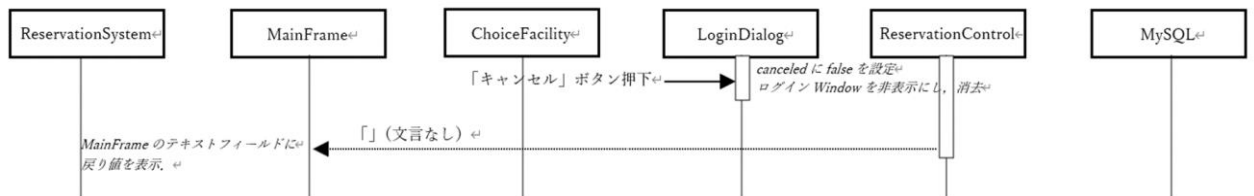
#### 4.2.4 ID, パスワード入力～パスワード不一致

本シーケンスは、ログインダイアログ表示中に登録済みユーザ ID 及び誤ったパスワードを入力し、OK ボタンを押下した時のものである。(ex.4.2 ログインダイアログ表示からの継続)



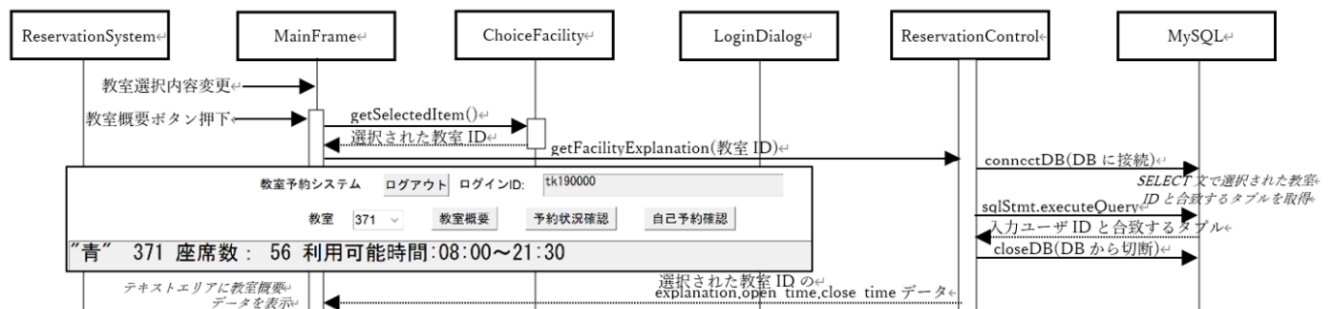
#### 4.2.5 認証中止

本シーケンスは、ログインダイアログ表示中にキャンセルボタンを押下した時のものである。(ex.4.2 ログインダイアログ表示からの継続)



#### 4.3 教室概要表示

本シーケンスは、MainFrame のみ表示中の状態で、コンボボックスより教室選択を行い、教室概要ボタンを押下した時のものである。(ex.4.1 起動後からの継続)

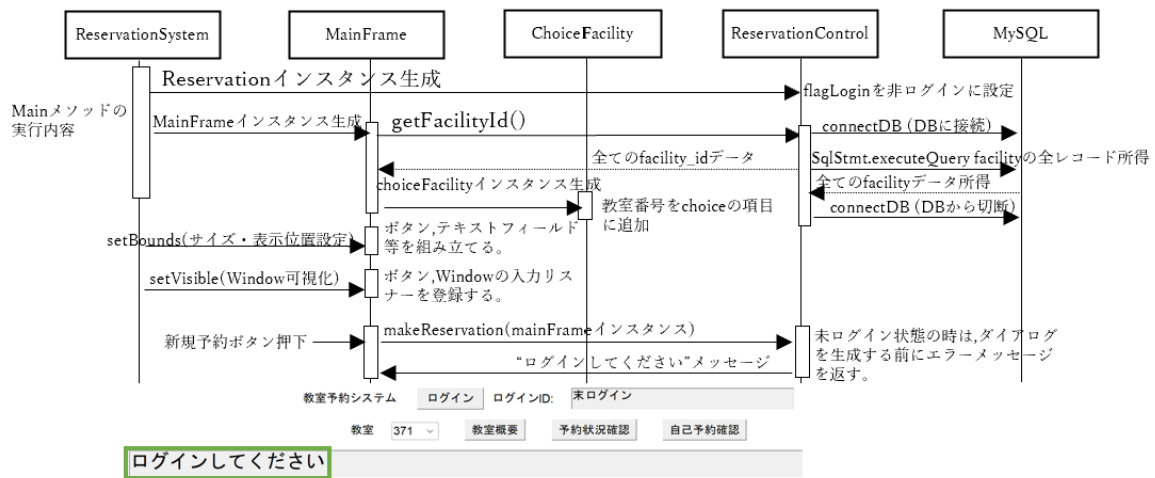


#### 4.4 新規予約機能

##### 4.4.1 未ログイン状態からの新規予約ボタン押下

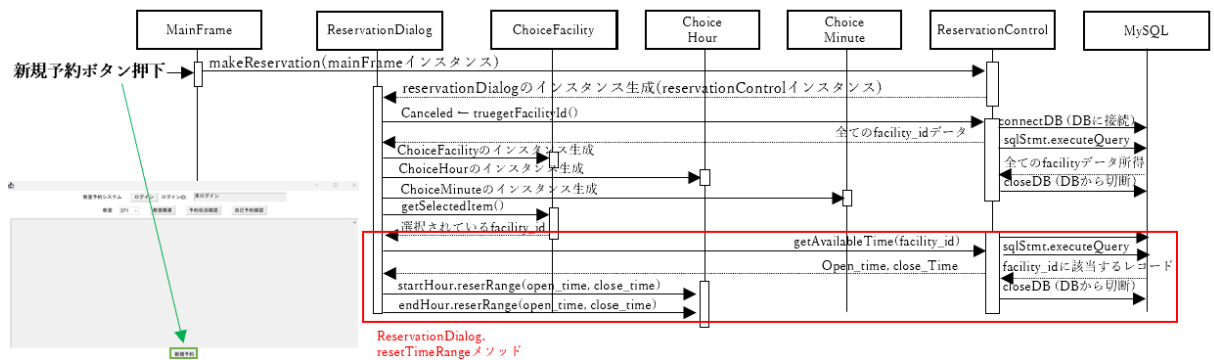
本シーケンスは、未ログイン状態で新規予約ボタンを押下した時のものである。





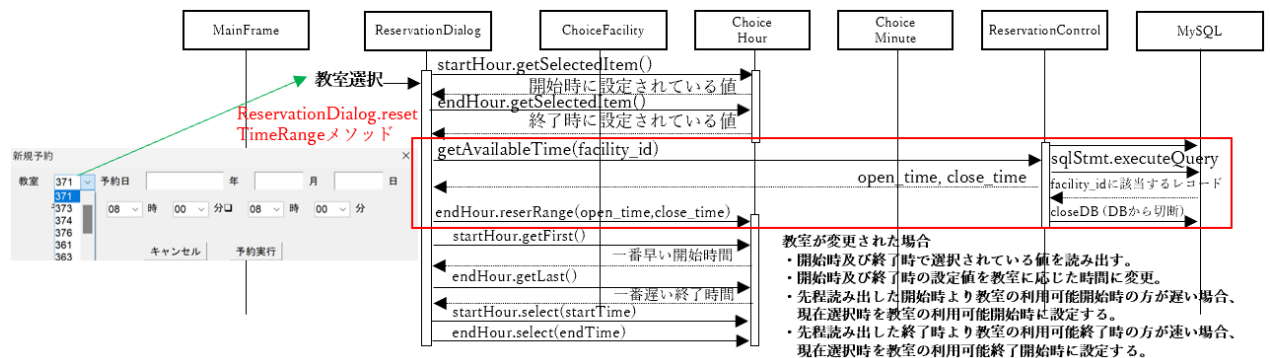
#### 4.4.2 ログイン後新規予約ボタン押下～予約実行まで

本シーケンスは、ログイン後に新規予約ボタンを押下した時のものである。



#### 4.4.3 選択教室変更

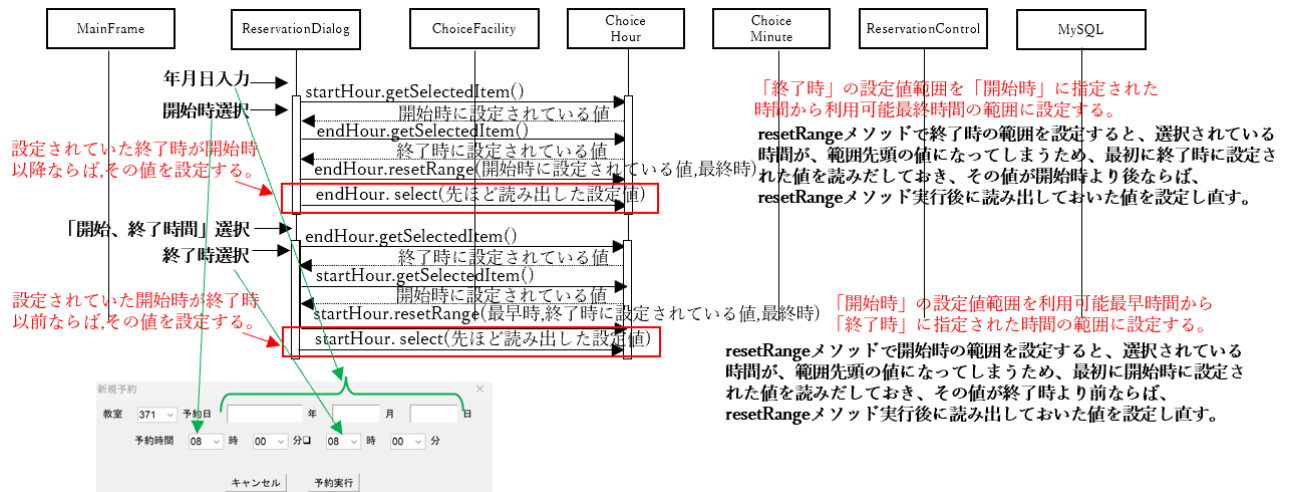
本シーケンスは、ログイン状態で新規予約ボタンを押下した後に教室を変更している時のものである。



#### 4.4.4 開始「時」変更～開始「分」変更～予約実行まで

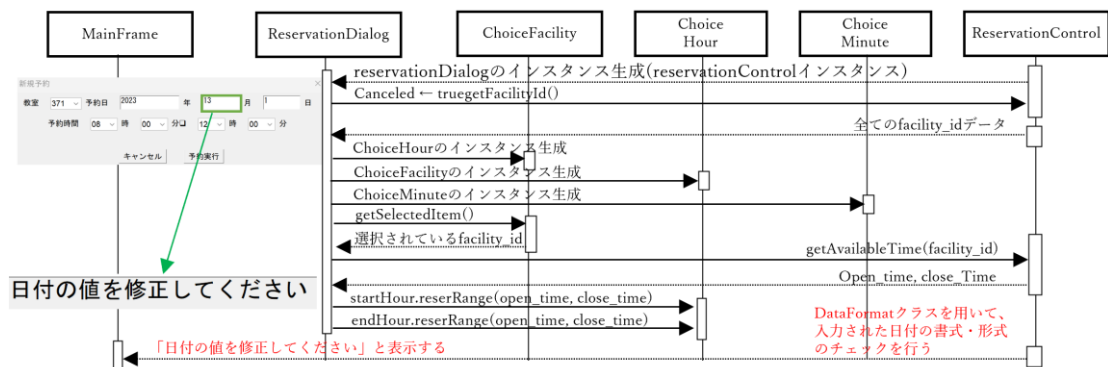
本シーケンスは、ログイン状態で新規予約ボタンを押下した後に開始時間等を変更

した時のものである。



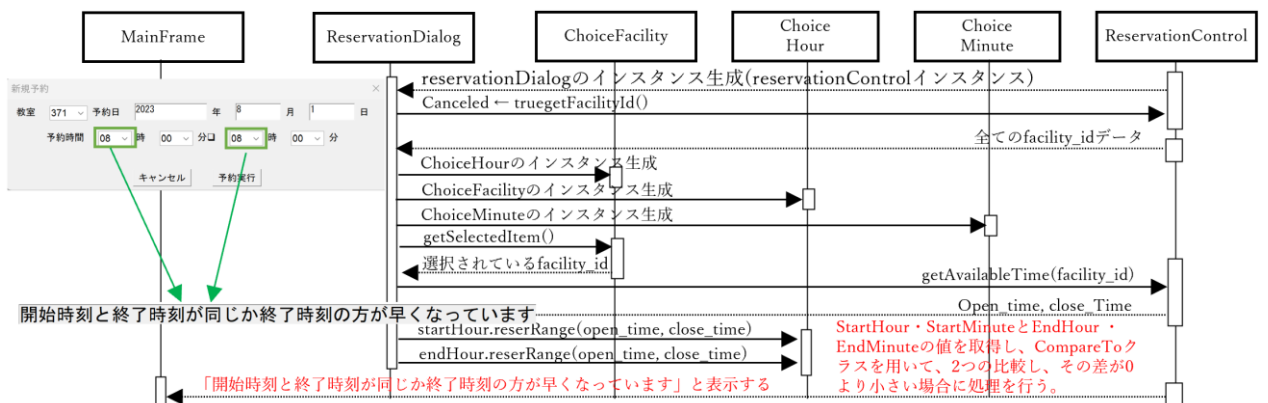
#### 4.4.5 入力データチェック～日付入力エラー

本シーケンスは、ログイン状態で新規予約ボタンを押下した後に、予約の日付を「13月」とし、予約実行ボタンを押下時のものである。



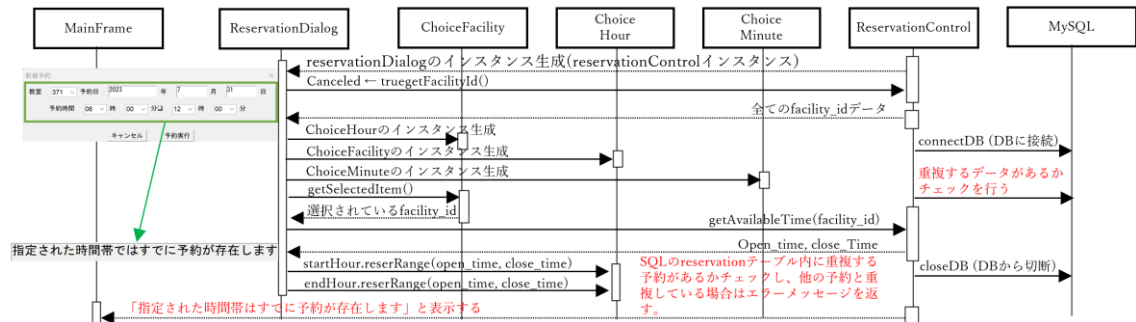
#### 4.4.6 入力データチェック～予約時間エラー

本シーケンスは、ログイン状態で新規予約ボタンを押下した後に、予約の開始時刻と終了時刻を同じ値にした時のものである。



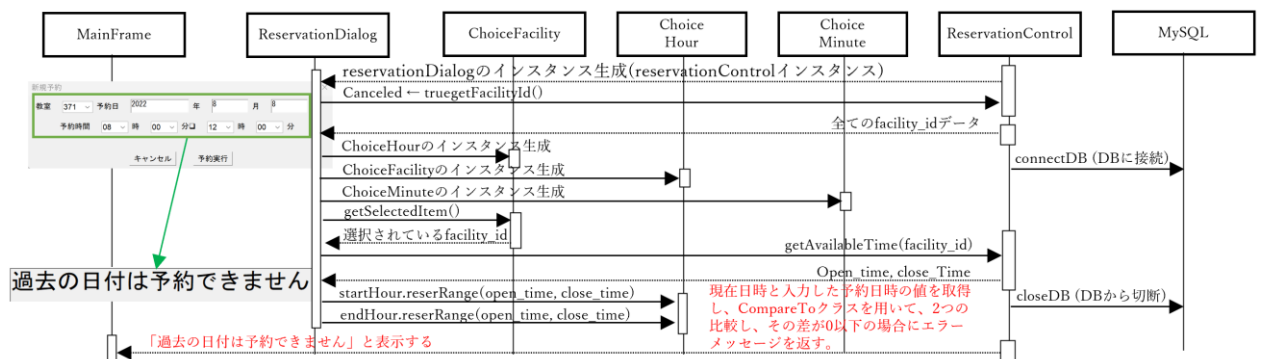
#### 4.4.7 入力データチェック～予約重複エラー

本シーケンスは、ログイン状態で新規予約ボタンを押下した後に、すでに予約した予約情報を再度予約した時のものである。



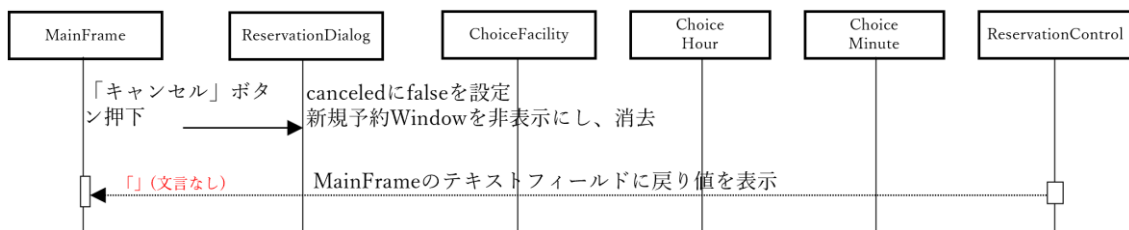
#### 4.4.8 入力データチェック～過去の日付エラー

本シーケンスは、ログイン状態で新規予約ボタンを押下した後に、過去の日付を入力し、予約した時のものである。



#### 4.4.9 予約中止

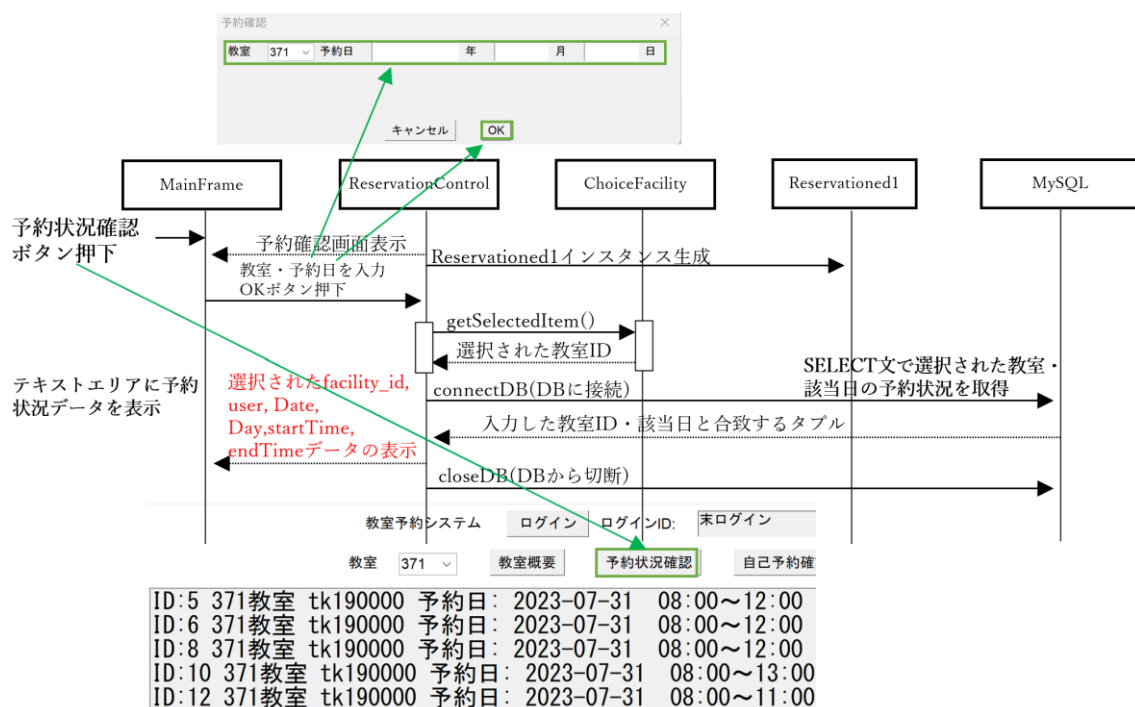
本シーケンスは、新規予約ダイアログ表示中にキャンセルボタンを押下した時のものである。



## 4.5 予約状況確認機能

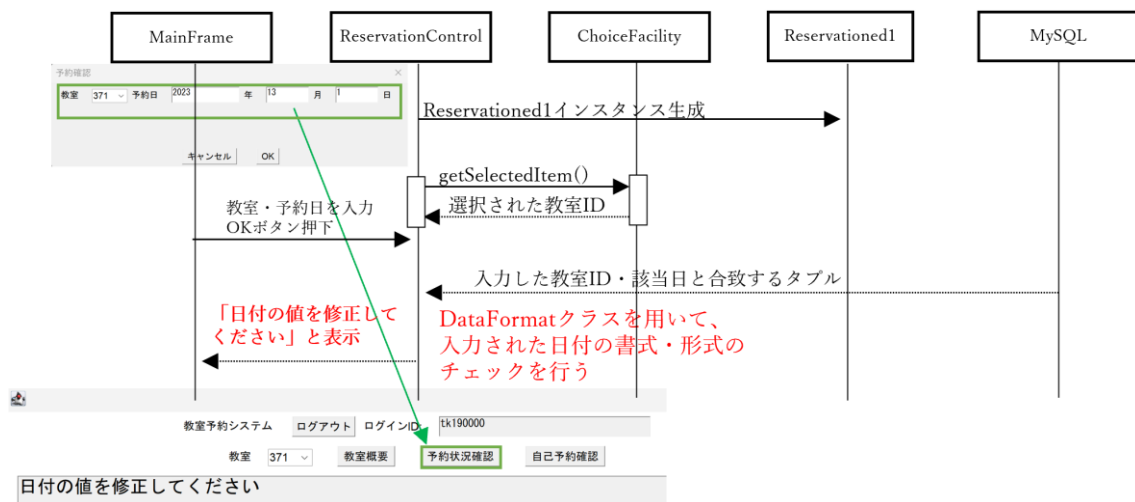
### 4.5.1 予約確認ダイアログ表示～実行

本シーケンスは、予約状況確認ボタンを押下した時のものである。



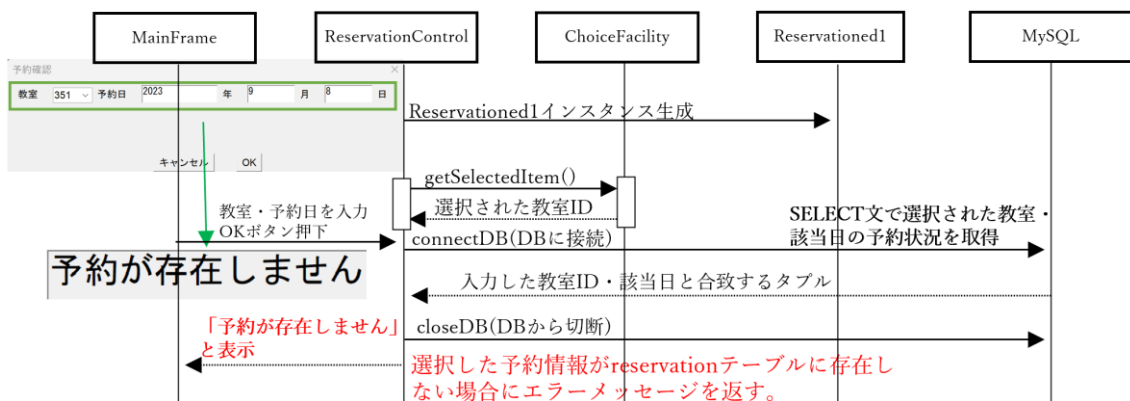
### 4.5.2 入力データチェック～日付入力エラー

本シーケンスは、予約状況確認ボタンを押下した後に、予約の日付を「13 月」とし、予約実行ボタンを押下時のものである。



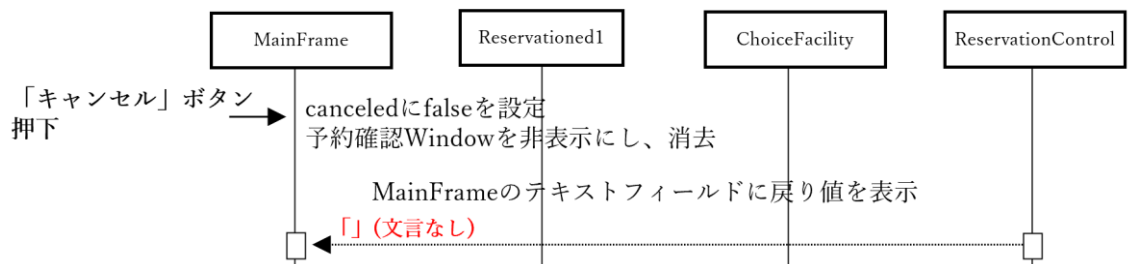
#### 4.5.3 入力データチェック～予約情報なしエラー

本シーケンスは、予約状況確認ボタンを押下した後に、予約していない日付を入力し、OK ボタンを押下時のものである。



#### 4.5.4 予約確認中止

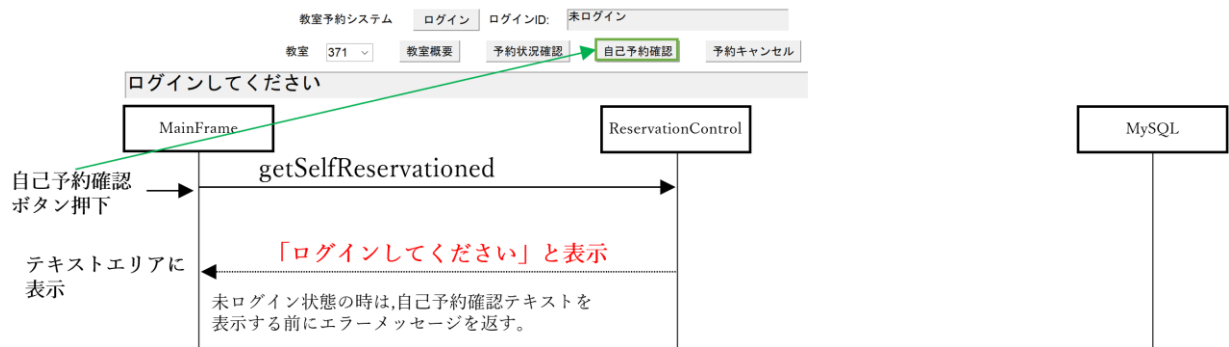
本シーケンスは、予約確認ダイアログ表示中にキャンセルボタンを押下した時のものである。



## 4.6 自己予約確認

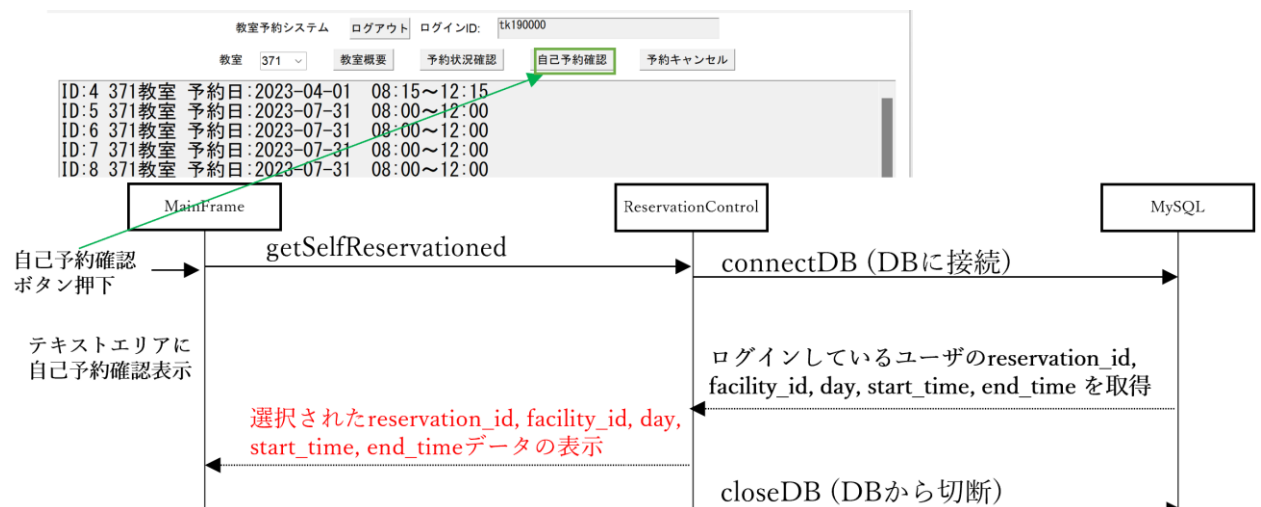
### 4.6.1 未ログイン状態～自己予約確認ボタン押下

本シーケンスは、未ログイン状態で自己予約確認ボタンを押下した時のものである。



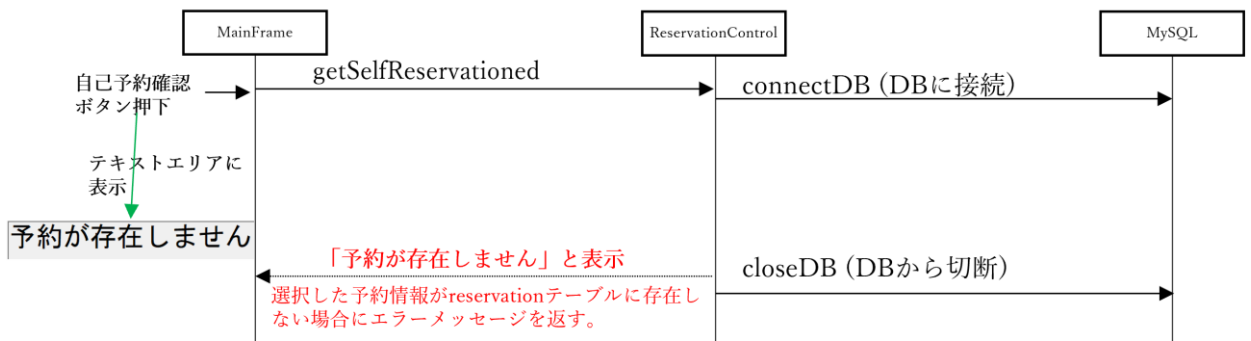
### 4.6.2 自己予約確認ボタン押下～自己予約確認実行

本シーケンスは、ログイン状態で自己予約確認ボタンを押下した時のものである。



### 4.6.3 入力データチェック～予約情報なしエラー

本シーケンスは、自己予約確認ボタンを押下した後に、予約していない日付を入力し、OK ボタンを押下時のものである。



## 4.7 予約キャンセル

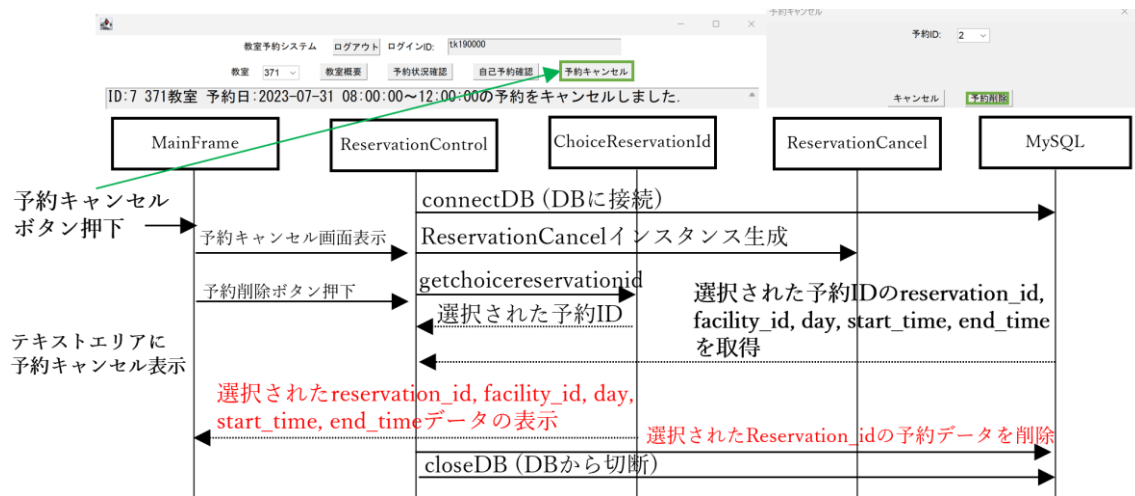
### 4.7.1 未ログイン状態～予約キャンセルボタン押下

本シーケンスは、未ログイン状態で予約キャンセルボタンを押下した時のものである。



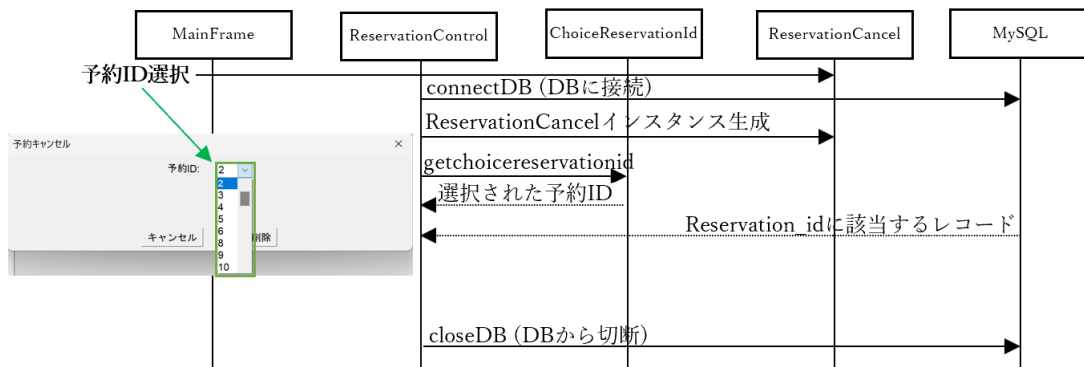
### 4.7.2 予約キャンセルボタン押下～予約キャンセル実行

本シーケンスは、ログイン状態で予約キャンセルボタンを押下した時のものである。



#### 4.7.3 選択予約 ID 変更

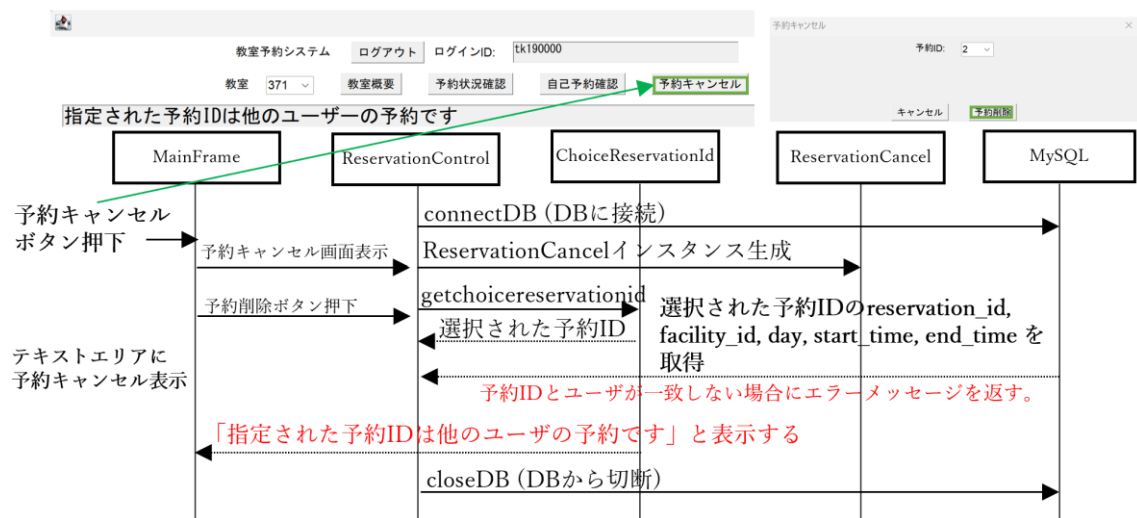
本シーケンスは、ログイン状態で予約キャンセルボタンを押下した後に予約 ID を変更している時のものである。



#### 4.7.4 入力データチェック～別ユーザエラー

本シーケンスは、ログイン状態で予約キャンセルボタンを押下した後、別のユーザの予約 ID を選択して、予約削除ボタンを押下した時のものである。





#### 4.7.5 予約キャンセル中止

本シーケンスは、予約キャンセルダイアログ表示中にキャンセルボタンを押下した時のものである。

