

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 20 日 第 4.3 版
2023 年 7 月 19 日 第 4.2 版
2023 年 7 月 12 日 第 4.1 版
2023 年 7 月 06 日 第 4.0 版
2023 年 7 月 06 日 第 3.3 版
2023 年 6 月 29 日 第 3.2 版
2023 年 6 月 28 日 第 3.1 版
2023 年 6 月 22 日 第 3.0 版
2023 年 6 月 22 日 第 2.2 版
2023 年 6 月 15 日 第 2.1 版
2023 年 4 月 02 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

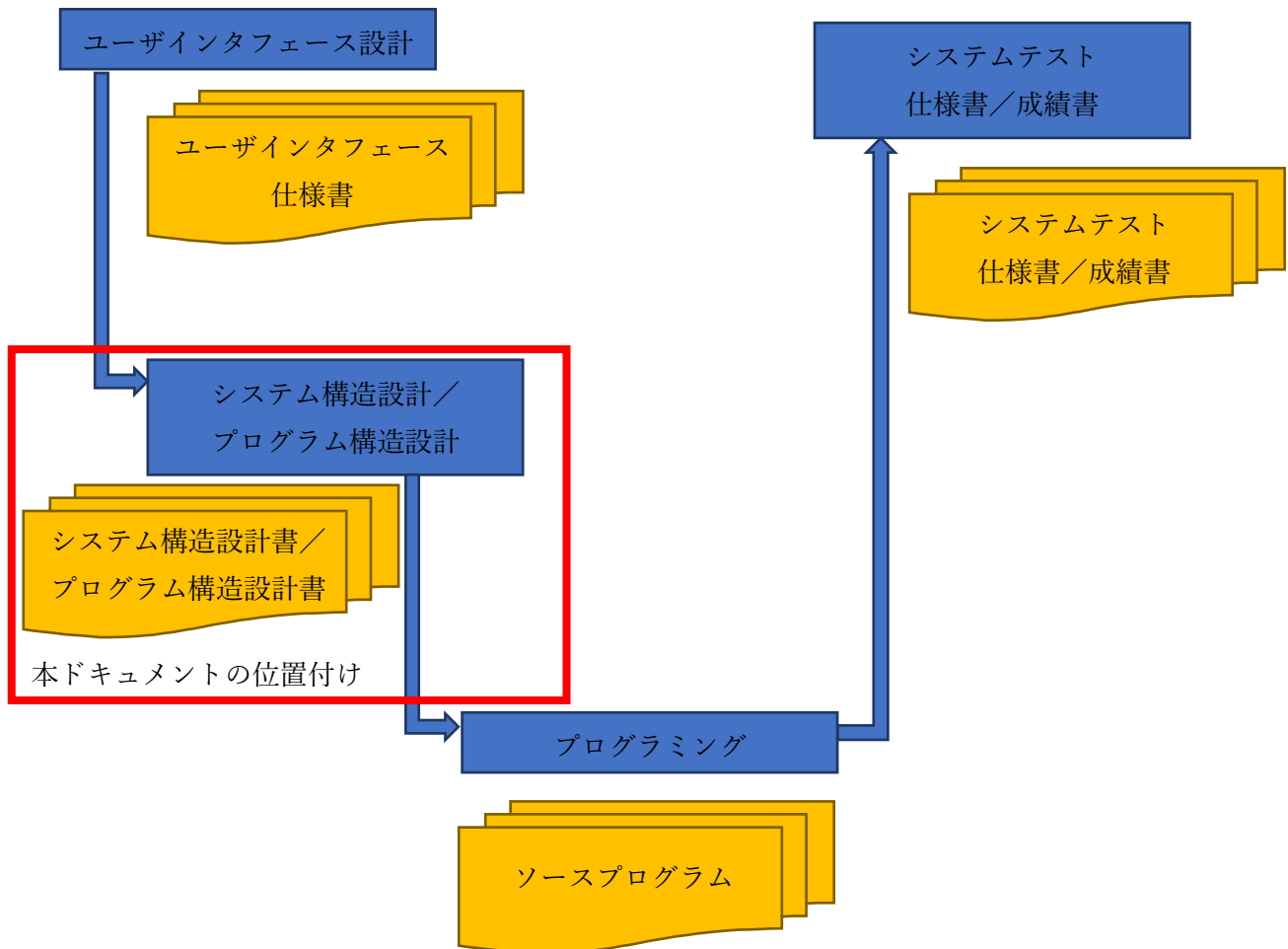
畑咲人

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	Teacher	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	Teacher	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	Teacher	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.		
		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	Teacher	2022/4/24
5	2.1	新規予約機能 (簡易版) 対応, 教室予約状況確認機能 対応. レビュー直前版発行	全項	畑咲人	2023/06/15
		2.1 MainFrame の構成を変更, panel 上への部品配 置について説明追記	2.1		
		2.3 新規予約ダイアログを追加	2.3		
		2.4 教室予約状況確認ダイアログを追加	2.4		
		3. クラス図に ChoiceHour,ChoiceMinute, ReservationDialog,ReservationCheckDialog を追加	3.		
		3.2 MainFrame の追加・変更処理追記	3.2		
		3.3 ReservationControl のメソッド追記	3.3		
		3.6 ChoiceHour 追記	3.6		
		3.7 ChoiceMinute 追記	3.7		
		3.8 ReservationDialog 追記	3.8		
		3.9 ReservationCheckDialog 追記	3.9		
		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. 追加されたクラス を追記.	4.		
6	2.2	レビュー後の変更を実施.	全項	畑咲人	2023/6/22
		2.3 新規予約機能ダイアログの説明文を変更.	2.3		
		2.4 教室予約状況ダイアログの画像を変更.	2.4		
		2.4 教室予約状況ダイアログの戻り値の説明文を訂 正.	2.4		
		3.8 ReservationDialog の記述を訂正.	3.8		

		3.9 ReservationCheckDialog の記述を訂正.	3.9		
		4. シーケンス図を修正.	4.		
		4.1 起動、4.2 ログイン機能 にて画像修正.	4.		
		4.3 教室概要表示 のシーケンス図内文言を修正.	4.3		
7	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,教室予約状況確認対応版ここまで	全頁	畑咲人	2023/6/22
8	3.1	課題1 フィードバックを参照し訂正.	全項	畑咲人	2023/6/28
		1.基本アーキテクチャ にて新規予約ボタン,教室予約 状況ボタン押下時のアクションについて追記	1.		
		2.3. 新規予約ダイアログの説明文を訂正,画像の変更	2.3		
		2.4. 教室予約状況ダイアログの説明文を訂正,画像の 変更	2.4		
		3. クラス図及びクラス内変数メソッドの全体クラス 図を変更	3.		
		3.2.3 actionPerfomed メソッドにて教室概要ボタン押 下時の説明,教室予約状況ボタン押下時の説明を追記	3.2.3		
		3.3.11 makeReservation メソッドにて説明文を訂正.	3.3.11		
		4. シーケンス図にて構成を変更. 4.4.2 新規予約ダイアログ表示不可の追加,元 4.4.4 予 約情報入力～時間不一致を 予約情報入力～予約時間, 予約日不適切 に変更. 元 4.5.3 予約状況確認日付不一致 を 予約状況確認日 付不適切,西暦4桁以上に変更. 4.5.2.2 予約なし を削除 4.5.3 予約状況確認出力失敗を追記	4		
9	3.2	新規予約機能対応,自己予約確認機能対応. レビュー 直前版発行	全項	畑咲人	2023/06/29
		1.基本アーキテクチャ にて自己予約確認ボタン, 自己 予約確認設定ボタン押下時のアクションについて追 記			
		2.1 MainFrame の画像差し替え,構成内容の更新.	2.1		
		2.5 自己予約確認 Option 設定ダイアログを追記.	2.5		
		3 クラス図及びクラス内の変数メソッド にて全体の クラス図を更新.	3.		

		3.2MainFram にて panelSouthSub1,panelSouthSub2, buttonMyReservation,buttonMyReservationSetting について追記.	3.2		
		3.2.3 actionPerdomed メソッド にて自己予約確認設定ボタン押下時,自己予約確認ボタン押下時を追記.	3.2.3		
		3.3 ReservationControl にて Set_date1, Set_date2, facility_id, MyCheckSetting, MyReservation を追記.	3.3		
		3.10 MyReservationSettingDialog を追記.	3.10		
		4 シーケンス図 にて各シーケンス間の繋がりに関する画像を差し替え.	4.		
		4 シーケンス図 にて, 4.1 起動 ~ 4.5 教室予約状況確認機能 のシーケンス図を更新.	4.		
		「4.4.5 予約情報入力~予約時間,予約日不適切」から「4.4.5 予約情報入力~予約情報不適切」に変更	4.5		
		4.4.5 予約情報入力~予約情報不適切 にて予約情報不適切に値する条件の詳細を以下に追記. 4.4.5.1 予約情報入力~予約西暦不適切 4.4.5.2 予約情報入力~予約日不適切 4.4.5.3 予約情報入力~予約日付不適切 4.4.5.4 予約情報入力~予約時間不適切	4.4.5		
		4.4.6 予約情報入力~登録失敗 を追記.	4.4.6		
		4.5 教室予約状況確認機能 にて 4.5.3 予約状況確認~出力失敗,4.5.4 予約情報入力~教室 ID 変更,4.5.5 予約情報入力~予約情報不適切 を追記.	4.5		
		4.6 自己予約確認設定機能 を追記.	4.6		
		4.7 自己予約確認機能 を追記.	4.7		
10	3.3	レビュー後の変更を実施.	全項	畑咲人	2023/07/06
		2.2 ログインダイアログ 部品配置について追記.	2.2		
		2.3 新規予約ダイアログ 部品配置について追記.	2.3		
		2.4 教室予約状況ダイアログ 部品配置について追記.	2.4		
		2.5 自己予約確認 Option 設定ダイアログ部品配置について追記.	2.5		
		3.3.14 makeReservation メソッドにて追記. また、開始時間と終了時刻が同じか終了時刻の方が早くなっていますという表記から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に変更	3.3.14		
		3.3.17 MyCheckSetting メソッドの自己予約確認 Option 設定ダイアログにて、設定完了ボタン押下時の指定期間（開始日~終了日）に問題があった時の説明文を一部変更.	3.3.17		

		3.3.18 MyReservation メソッドにてミニマム仕様とオプション仕様の説明内容の構成を変更.	3.3.18		
11	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,自己予約確認機能対応版ここまで	全項	畑咲人	2023/07/06
12	4.1	課題2 フィードバックを参照し訂正.	全項	畑咲人	2023/07/12
		3.2.1 MainFrame コンストラクタ にて自己予約ボタン, 自己予約確認設定ボタンの記述を追記.	3.2.1		
		3.3 ReservationControl にて 3.3.7 ResetSetting フラグを追記.	3.3.7		
		3.3.15 makeReservation メソッドにて 使用する関数, フラグ, リストについて説明し, 動作について詳細な説明を追記.	3.3.15		
		3.3.17 getFacility_Reservation メソッドにて 使用する関数, フラグ, リストについて説明し, 動作について詳細な説明を追記.	3.3.17		
		3.3.18 MyCheckSetting メソッドにて 使用する関数, フラグ, リストについて説明し, 動作について詳細な説明を追記.	3.3.18		
		3.3.19 MyReservation メソッドにて 使用する関数, フラグ, リストについて説明し, 動作について詳細な説明を追記.	3.3.19		
13	4.2	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	全項	畑咲人	2023/07/19
		1 基本アーキテクチャ にて 予約キャンセルボタン押下時のアクションについて追記.	1.		
		2.6 予約キャンセルダイアログ について追記.	2.6		
		3 クラス図及び各クラス内の変数メソッドにて クラス図を変更.	3.		
		3.2.1 MainFrame コンストラクタにて 予約キャンセルボタンの記述を追記.	3.2.1		
		3.2.3 actionPerfomed メソッドにて 予約キャンセルボタン押下時を追記.	3.2.3		
		3.3 ReservationControl にて 3.3.20 ReservationCancel メソッドを追記.	3.3		
		3.11 CancelDialog を追記.	3.11		
		4 シーケンス図にて 各シーケンス間の繋がりを表す画像を変更.	4.		
		4 シーケンス図 にて 4.8 予約キャンセルを追記.	4.8		
		4 シーケンス図にて 4.8 の追記に伴い, 各シーケンスの画像を変更.	4.		

14	4.3	レビュー後の変更を実施.	全項	畑咲人	2023/07/20
		2.1 MainFrame にて画像差し替え.	2.1		
		2.6 予約キャンセルダイアログにて説明内容を画像差し替え.	2.6		
		3.1.4 itemStateChanged メソッド にて「利用可能時刻」から「利用可能開始時刻」に修正.	3.1.4		
		4.2.5 認証中止 シーケンス図を差し替え.	4.2.5		
		4.7.2.2.1 予約あり と 4.7.2.2.2 予約なし の説明文にて「 ex.4.2.2 ID,パスワード入力～認証完了からの継続 」から「 ex. 4.6.3 設定情報入力～設定完了からの継続 」に変更.	4.7.2		
		4 シーケンス図にて シーケンス間の繋がり図を変更.	4.		
15	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	全項	畑咲人	2023/07/20

目次

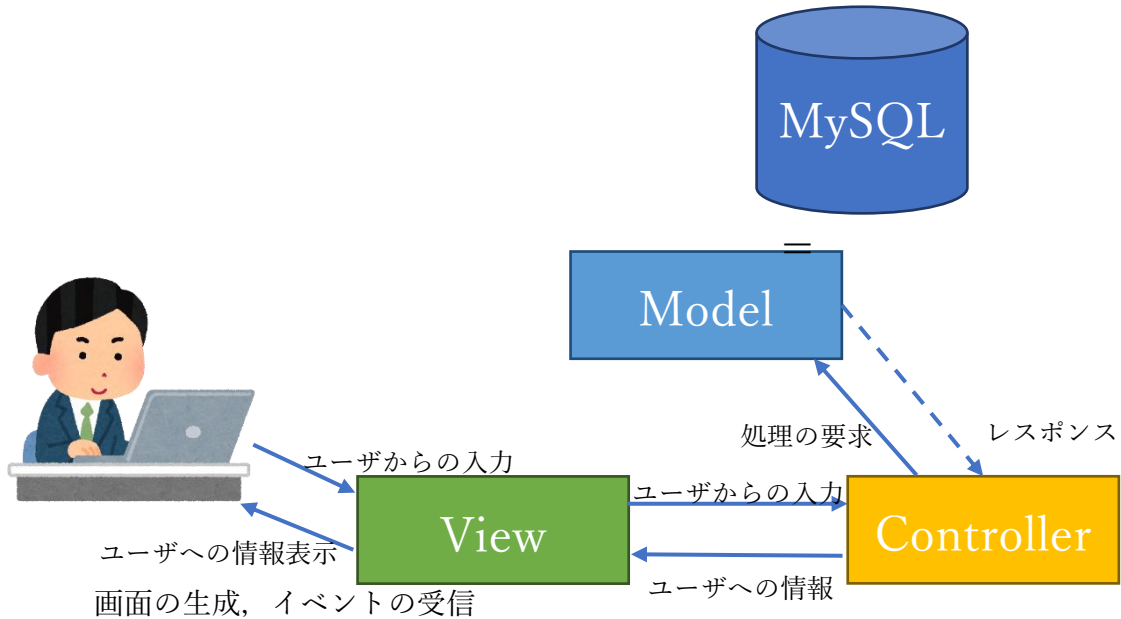
1	基本アーキテクチャ	10
2	Window 構成	12
2.1	MainFrame	12
2.2	ログインダイアログ	12
2.3	新規予約ダイアログ	13
2.4	教室予約状況ダイアログ	13
2.5	自己予約確認 Option 設定ダイアログ	14
2.6	予約キャンセルダイアログ	14
3	クラス図及び各クラス内の変数及びメソッド	15
3.1	ReservationSystem	15
3.1.1	main メソッド	15
3.2	MainFrame	16
3.2.1	MainFrame コンストラクタ	16
3.2.2	windowClosing メソッド	16
3.2.3	actionPerformed メソッド	17
3.3	ReservationControl	18
3.3.1	sqlUserID	18
3.3.2	sqlPassword	18
3.3.3	reservationUserID	18
3.3.4	Set_date1	19
3.3.5	Set_date2	19
3.3.6	facility_id	19
3.3.7	ResetSetting	19
3.3.8	flagLogin	19
3.3.9	ReservationControl コンストラクタ	19
3.3.10	connectDB メソッド	19
3.3.11	closeDB メソッド	19
3.3.12	loginLogout メソッド	20
3.3.13	getFacilityExplanation メソッド	20
3.3.14	getFacilityId メソッド	21
3.3.15	makeReservation メソッド	21
3.3.16	getAvailableTime メソッド	23
3.3.17	getFacility_Reservation メソッド	24
3.3.18	MyCheckSetting メソッド	25
3.3.19	MyReservation メソッド	28
3.3.20	ReservationCancel メソッド	31
3.4	LoginDialog	34
3.4.1	canceled	34

3.4.2	LoginDialog コンストラクタ	34
3.4.3	windowClosing メソッド	34
3.4.4	actionPerformed メソッド	34
3.5	ChoiceFacility.....	35
3.5.1	ChoiceFacility コンストラクタ	35
3.6	ChoiceHour.....	35
3.6.1	ChoiceHour コンストラクタ	35
3.6.2	resetRange メソッド	35
3.6.3	getFirst メソッド	35
3.6.4	getLast メソッド	35
3.7	ChoiceMinute	36
3.7.1	ChoiceMinute コンストラクタ	36
3.8	ReservationDialog.....	36
3.8.1	canceled	37
3.8.2	ReservationDialog コンストラクタ	37
3.8.3	resetTimeRange メソッド	37
3.8.4	itemStateChanged メソッド	37
3.8.5	windowClosing メソッド	37
3.8.6	actionPerformed メソッド	37
3.9	ReservationCheckDialog.....	38
3.9.1	canceled	38
3.9.2	ReservationCheckDialog コンストラクタ	38
3.9.3	windowClosing メソッド	38
3.9.4	actionPerformed メソッド	38
3.10	MyReservationSettingDialog	39
3.10.1	canceled	39
3.10.2	ResetSetting.....	39
3.10.3	MyReservationSettingDialog コンストラクタ	40
3.10.4	windowClosing メソッド	40
3.10.5	actionPerformed メソッド	40
3.11	CancelDialog.....	41
3.11.1	canceled	41
3.11.2	CancelDialog コンストラクタ	41
3.11.3	resetTimeRange メソッド	42
3.11.4	itemStateChanged メソッド	42
3.11.5	windowClosing メソッド	42
3.11.6	actionPerformed メソッド	42
4	シーケンス図	43
4.1	起動	46
4.2	ログイン機能.....	46

4.2.1	ログインダイアログ表示	46
4.2.2	ID, パスワード入力～認証完了	46
4.2.3	ID, パスワード入力～ユーザ ID 未登録	47
4.2.4	ID, パスワード入力～パスワード不一致	47
4.2.5	認証中止	47
4.3	教室概要表示	48
4.4	新規予約機能	48
4.4.1	新規予約ダイアログ表示	48
4.4.2	新規予約ダイアログ表示不可	48
4.4.3	予約情報入力～登録完了	48
4.4.4	予約情報入力～教室 ID, 予約時間変更	49
4.4.5	予約情報入力～予約情報不適切	49
4.4.6	予約情報入力～登録失敗	50
4.4.7	新規予約中止	51
4.5	教室予約状況確認機能	51
4.5.1	教室予約状況ダイアログ表示	51
4.5.2	予約状況確認出力成功	51
4.5.3	予約状況確認出力失敗	52
4.5.4	予約情報入力～教室 ID 変更	52
4.5.5	予約情報入力～予約情報不適切	52
4.5.6	予約状況確認中止	53
4.6	自己予約確認設定機能	53
4.6.1	自己予約確認 Option 設定ダイアログ表示	53
4.6.2	自己予約確認 Option 設定ダイアログ表示不可	53
4.6.3	設定情報入力～設定完了	53
4.6.4	設定情報初期化	54
4.6.5	設定情報入力～教室 ID 変更	54
4.6.6	設定情報入力～設定情報不適切	54
4.6.7	自己予約確認設定中止	56
4.7	自己予約確認機能	56
4.7.1	未ログイン状態の自己予約確認	56
4.7.2	ログイン状態の自己予約確認	56

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。

また、ユーザが「教室概要」ボタンを押下すると、MainFrame はそれを ReservationControl に伝え、ReservationControl クラスでは、MySQL との通信により、教室名、座席数、教室利用可能時間を MainFrame に引き渡す。

また、ユーザが「教室予約状況」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 ReservationCheckDialog クラスのインスタンスを生成し、以降教室予約状況確認処理が終了するまでの間、ReservationControl は ReservationCheckDialog クラスより、ユーザの入力を受け付け、処理を行う。

また、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、以降新規予約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。

また、ユーザが「自己予約確認設定」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 MyReservationSettingDialog クラスのインスタンスを生成し、以降自己予約確認設定処理が終了するまでの間、ReservationControl は

MyReservationSettingDialog クラスより、ユーザの入力を受け付け、処理を行う。

また、ユーザが「自己予約確認」ボタンを押下すると、MainFrame はそれを ReservationControl に伝え、ReservationControl クラスでは、「自己予約確認設定」の設定内容に基づき、ユーザ自身が予約したデータを MySQL との通信により、教室名、予約日、教室利用時間を MainFrame に引き渡す。

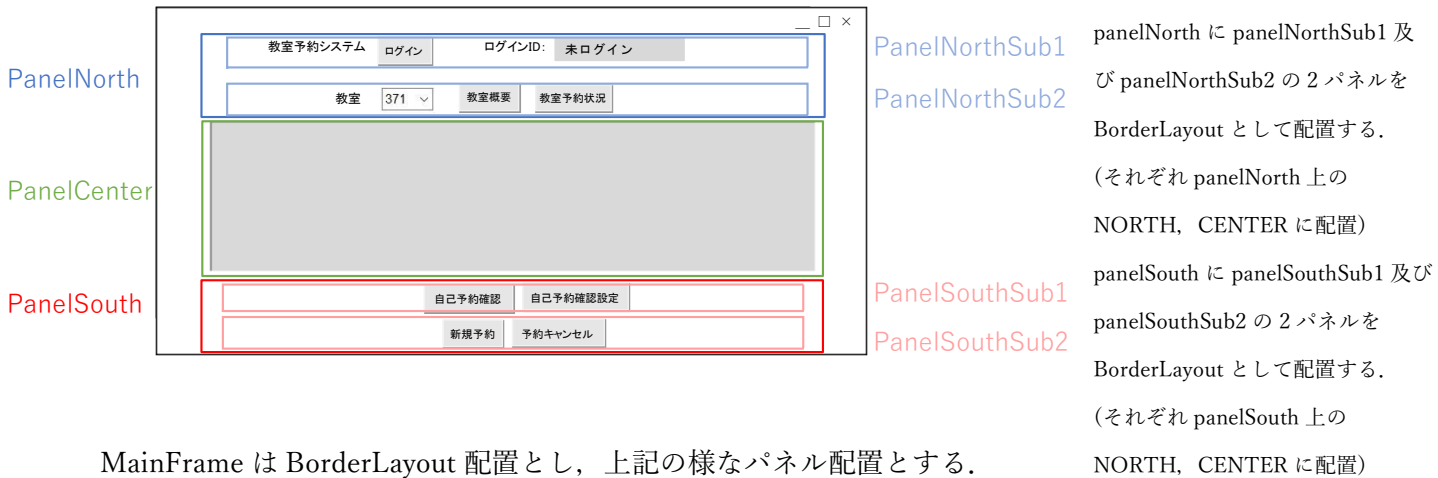
そして、ユーザが「予約キャンセル」ボタンを押下すると、MainFrame はそれを ReservationControl に伝え、ReservationControl クラスでは、もう 1 つの View である CancelDialog クラスのインスタンスを生成し、以降予約キャンセル処理が終了するまでの間、ReservationControl は CancelDialog クラスより、ユーザの入力を受け付け、処理を行う。

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。

2 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.

2.1 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, 教室予約状況ボタン

panelCenter への配置部品

テキストエリア (20 行×80 文字)

panelSouth1 への配置部品

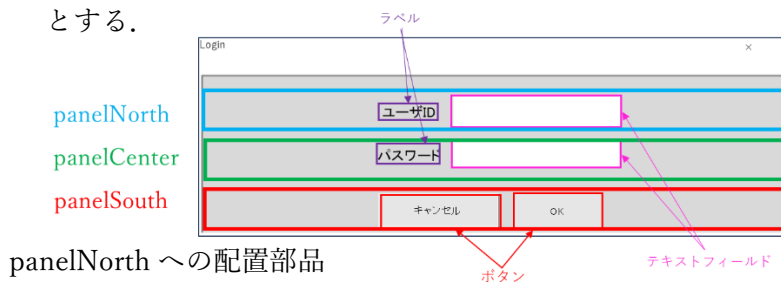
自己予約確認ボタン, 自己予約確認設定ボタン

panelSouth2 への配置部品

新規予約ボタン, 予約キャンセルボタン

2.2 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.



panelNorth への配置部品

ユーザ ID ラベル, ユーザ ID テキストフィールド

panelCenter への配置部品

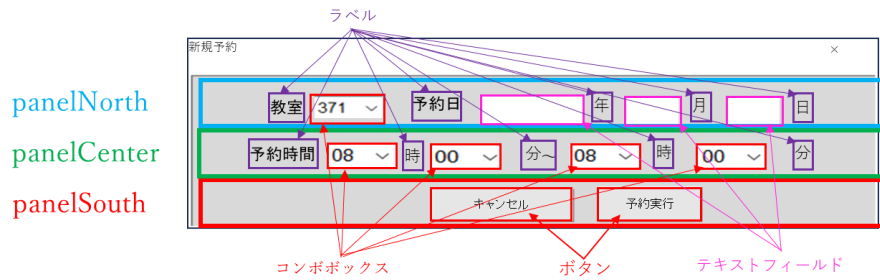
パスワードラベル, パスワードテキストフィールド

panelSouth への配置部品

キャンセルボタン, OK ボタン

2.3 新規予約ダイアログ

ReservationDialog は BorderLayout の配置とする。



panelNorth への配置部品

教室ラベル, 教室選択コンボボックス, 予約日ラベル, 年テキストフィールド, 年ラベル, 月テキストフィールド, 月ラベル, 日テキストフィールド, 日ラベル

panelCenter への配置部品

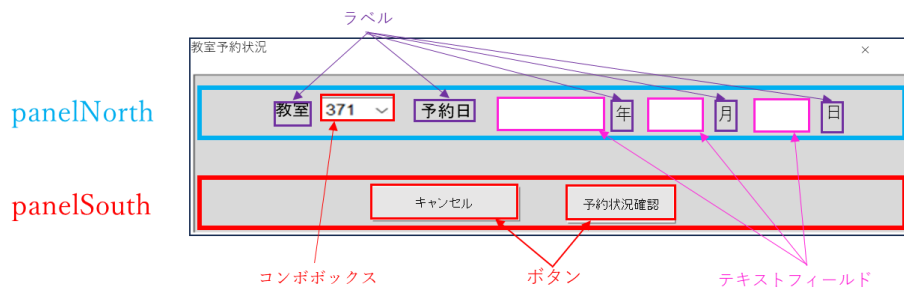
予約時間ラベル, 開始時コンボボックス, 時ラベル, 開始分コンボボックス, 分〜ラベル, 終了時コンボボックス, 時ラベル, 終了分コンボボックス, 分ラベル

panelSouth への配置部品

キャンセルボタン, 予約実行ボタン

2.4 教室予約状況ダイアログ

ReservationCheckDialog は、BorderLayout 配置とする。



panelNorth への配置部品

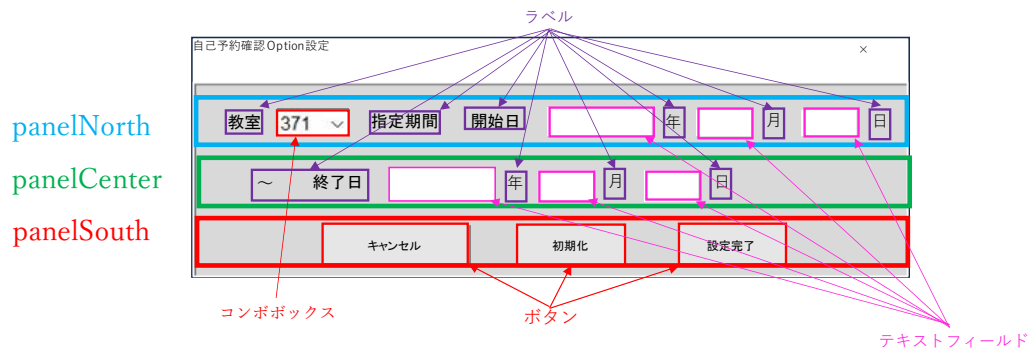
教室ラベル, 教室コンボボックス, 予約日ラベル, 年テキストフィールド, 年ラベル, 月テキストフィールド, 月ラベル, 日テキストフィールド, 日ラベル

panelSouth への配置部品

キャンセルボタン, 予約状況確認ボタン

2.5 自己予約確認 Option 設定ダイアログ

MyReservationSettingDialog は, BorderLayout 配置とする.



panelNorth への配置部品

教室ラベル, 教室コンボボックス, 指定期間ラベル, 開始日ラベル, 年テキストフィールド, 年ラベル, 月テキストフィールド, 月ラベル, 日テキストフィールド, 日ラベル

panelCenter への配置部品

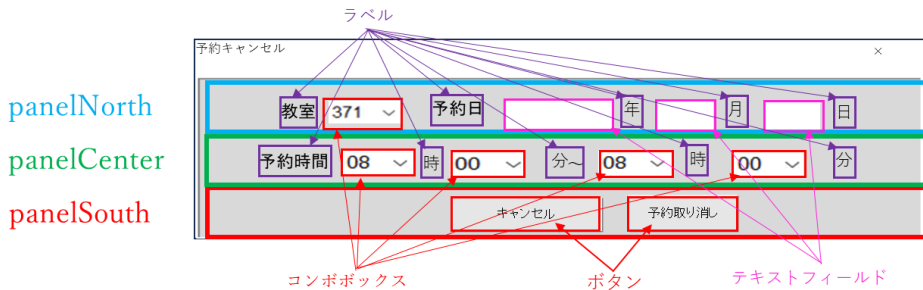
~ 終了日ラベル, 年テキストフィールド, 年ラベル, 月テキストフィールド, 月ラベル, 日テキストフィールド, 日ラベル

panelSouth への配置部品

キャンセルボタン, 初期化ボタン, 設定完了ボタン

2.6 予約キャンセルダイアログ

CancelDialog は, BorderLayout 配置とする



panelNorth への配置部品

教室ラベル, 教室選択コンボボックス, 予約日ラベル, 年テキストフィールド, 年ラベル, 月テキストフィールド, 月ラベル, 日テキストフィールド, 日ラベル

panelCenter への配置部品

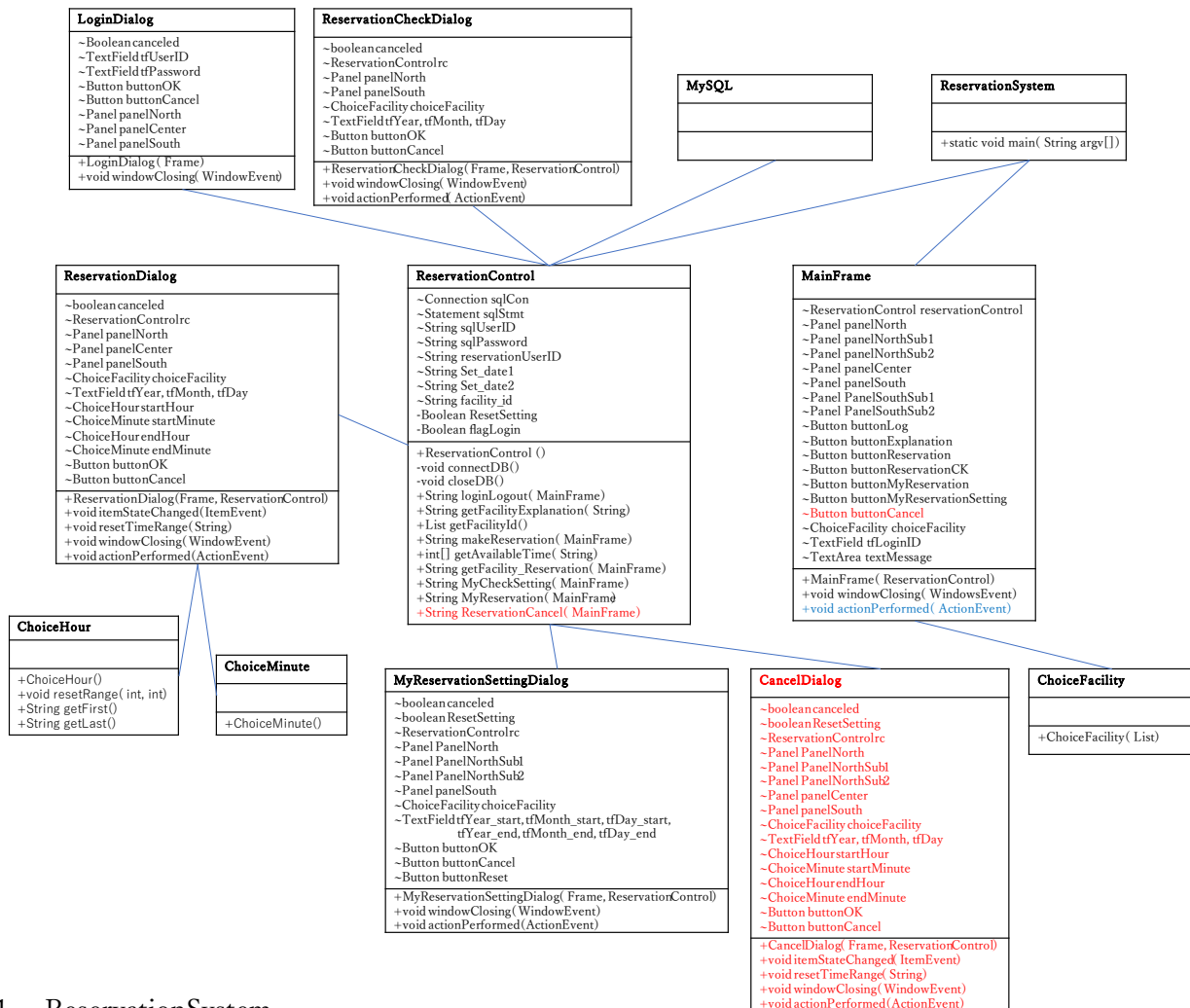
予約時間ラベル, 開始時コンボボックス, 時ラベル, 開始分コンボボックス, 分~ラベル, 終了時コンボボックス, 時ラベル, 終了分コンボボックス, 分ラベル

panelSouth への配置部品

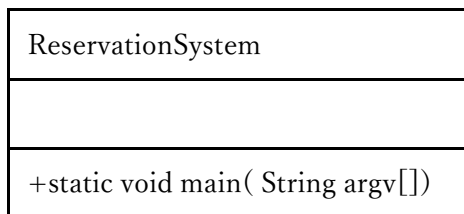
キャンセルボタン, 予約取り消しボタン

3 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。



3.1 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。

3.1.1 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。

3.2 MainFrame

MainFrame
~ReservationControl reservationControl ~Panel panelNorth ~Panel panelNorthSub1 ~Panel panelNorthSub2 ~Panel panelCenter ~Panel panelSouth ~Panel PanelSouthSub1 ~Panel PanelSouthSub2 ~Button buttonLog ~Button buttonExplanation ~Button buttonReservation ~Button buttonReservationCK ~Button buttonMyReservation ~Button buttonMyReservationSetting ~Button buttonCancel ~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。

3.2.1 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、自己予約ボタン,自己予約確認設定ボタン,予約キャンセルボタン,教室概要ボタン、新規予約ボタン、教室予約状況ボタン,ログイン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。

3.2.2 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。

教室予約状況ボタン押下時：

reservationControl インスタンスの getFacility_Reservation 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。

自己予約確認設定ボタン押下時：

reservationControl インスタンスの MyCheckSetting メソッドを呼び出す。この時、戻り値を表示結果エリアに表示させる。

自己予約確認ボタン押下時：

reservationControl インスタンスの MyReservation 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。

予約キャンセルボタン押下時：

reservationControl インスタンスの ReservationCancel 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。

3.3 ReservationControl

ReservationControl
<div>~Connection sqlCon</div> <div>~Statement sqlStmt</div> <div>~String sqlUserID</div> <div>~String sqlPassword</div> <div>~String reservationUserID</div> <div>~String Set_date1</div> <div>~String Set_date2</div> <div>~String facility_id</div> <div>-Boolean ResetSetting</div> <div>-Boolean flagLogin</div>
<div>+ReservationControl ()</div> <div>-void connectDB()</div> <div>-void closeDB()</div> <div>+String loginLogout(MainFrame)</div> <div>+String getFacilityExplanation(String)</div> <div>+List getFacilityId()</div> <div>+String makeReservation(MainFrame)</div> <div>+int[] getAvailableTime(String)</div> <div>+String getFacility_Reservation(MainFrame)</div> <div>+String MyCheckSetting(MainFrame)</div> <div>+String MyReservation(MainFrame)</div> <div>+String ReservationCancel(MainFrame)</div>

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。

3.3.1 sqlUserID

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

3.3.2 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。

3.3.3 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。

3.3.4 Set_date1

本システムの自己予約確認機能でユーザが設定した,指定期間開始日を格納する変数.ログイン中のみ有効となる.

3.3.5 Set_date2

本システムの自己予約確認機能でユーザが設定した,指定期間終了日を格納する変数.ログイン中のみ有効となる.

3.3.6 facility_id

本システムの自己予約確認機能でユーザが設定した,教室 id を格納する変数.ログイン中のみ有効となる.

3.3.7 ResetSetting

本システムの自己予約確認設定機能でユーザが設定したオプション仕様が有効になっているかを表す変数.ログイン中のみ有効となる.

3.3.8 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.

3.3.9 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.

3.3.10 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.

3.3.11 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.

3.3.12 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「ID が違います」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。

3.3.13 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。

3.3.14 getFacilityId メソッド

MainFrame のボタン押下時に MainFrame に呼び出される, ResrvationControl クラスのメソッドにより呼び出される.

本メソッドでは, MySQL に接続し, 該当テーブルの facility_id を参照し, そのデータからリスト作成し, 必要に応じて戻り値として返す.

3.3.15 makeReservation メソッド

MainFrame の新規予約ボタン押下時に MainFrame により呼出される.

本メソッドでは, ログイン時は MySQL に接続し, 新規予約ダイアログより入力された値を reservation テーブルの facility_id, user_id, date, day, start_time, end_time にデータとして送り, 戻り値を返す. また, 未ログイン時は「ログインしてください」を戻り値に設定する.

以下は, 本メソッドで使用する変数, フラグ及びリストである.

justNow : 本メソッドにて過去に対する予約を防止すべく, 現在時刻を java 内にある Calendar クラスから現在の日時を取得する.

resDate : 指定された値を java 内にある SimpleDateFormat クラスを用いて西暦 4 桁, 月は 1~12 月, 日は 1~31(毎月末日まで), 時間, 分, 秒の形である「yyyy-MM-dd HH:mm:ss」に指定された値をフォーマットする.

now : justNow で受け取った値を resDate に引き渡しフォーマットする.

facility : ReservationDialog にて選択された教室 ID を本メソッドに引き渡す.

st : ReservationDialog にて選択された開始時刻(startHour, startMinute)を「startHour : startMinute」の形にする.

et : ReservationDialog にて選択された終了時刻(endHour, endMinute)を「endHour : endMinute」の形にする.

ryear_str : ReservationDialog にて入力された予約日の年を本メソッドに引き渡す.

rmonth_str : ReservationDialog にて入力された予約日の月を本メソッドに引き渡す.
また, 引き取った値が一桁だった場合, 頭に 0 を足す.
例) 入力された値が 5 の時, 「05」にする.

rday_str : ReservationDialog にて入力された予約日の日を本メソッドに引き渡す.
また, 引き取った値が一桁だった場合, rmonth_str と同様に頭に 0 を足す.

rdate : 本メソッドにて受け取った ryear_str, rmonth_str, rday_str を各間に“-”を入れ,「ryear_str + "-" + rmonth_str + "-" + rday_str」の形にする.

convData : java 内にある SimpleDateFormat クラスを用いて西暦 4 桁, 月は 1~12 月, 日は 1~31(毎月末日まで)の形である「yyyy-MM-dd」に rdate をフォーマットする.

reservationTimeCK : convData と st の値を引き継ぎ, 各間にスペースを入れ,「convData + " " + st」の形にする.

rt : reservationTimeCK で整形した値を resDate に引き渡しフォーマットする.

startTime : reservation テーブルから該当する start_time を取得する.

endTime : reservation テーブルから該当する end_time を取得する.

CheckList_start : startTime の取得データをリストデータとして追記する.

CheckList_end : endTime の取得データをリストデータとして追記する。
count : CheckList_start に格納されているデータの数記憶する。
isOverlap : boolean 形式のフラグ.初期状態では false が設定されており,新規予約内容と reservation テーブルにある予約が競合しないかのチェックに用いる。

以下は,本メソッドの動作である。

ログイン時 :

MainFrame の新規予約ボタン押下時 :

ReservationDialog を呼び出す。

ReservationDialog のキャンセルボタン、または×ボタン押下時 :

戻り値に「」を設定。

ReservationDialog の予約実行ボタン押下時 :

ryear_str, rmonth_str, rday_str を ReservationDialog から取得。

Calender クラスから justNow を取得し,取得したデータを resDate でフォーマットし,その結果を now に引き渡す。

facility, st, et を ReservationDialog から取得。

rmonth_str と rday_str において,各々の値が 1 桁であるとき :

前に 0 を付加し,再度各変数に引き渡す。

rdate を ryear_str, rmonth_str, rday_str から作成。

convData で rdate の値を「yyyy-MM-dd」にフォーマット。

convData が rdate と等しくない,もしくは ryear_str に入力された西暦が 4 桁以外のとき : 戻り値に「日付の書式を修正して下さい(年 : 西暦 4 桁, 月 : 1~12, 日 : 1~31(毎月末日まで))」を設定。

reservationTimeCK を convData と st の値から整形。

rt を reservationTimeCK から作成。

rt(予約日時)と now (現在時刻)を比較して rt の方が小さいとき :

戻り値に「過去に対する予約はできません」を設定。

convData の値が存在しない日付であったとき :

convData の値が「月は 1~12 月, 日は 1~31(毎月末日まで)」を守っていない,もしくは「2025 年 2 月 29 日」のような,うるう年でないのに関わらず「2 月 29 日」を設定してしまっているとき,

戻り値に「日付の値を修正してください」を設定。

st(開始時刻)と et(終了時刻)を比較して同じ,もしくは st が et より大きいとき :

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。

MySQL に接続。

MySQL の reservation テーブルから day と rdate, facility_id と facility が一致する start_time, end_time を startTime, endTime にて取得し,各値を CheckList_start, CheckList_end に格納する。

isOverlap フラグを初期化(false)にする。

count にて CheckList_start の格納しているデータの数を取得する.

count の値の分だけ, CheckList_start に格納されている値と et(終了時刻), CheckList_end に格納されている値と st(開始時刻)を順番に比較する.

CheckList_start の値が et より小さく, CheckList_end の値が st より大きいとき :

isOverlap フラグを true にする.

isOverlap が true のとき :

新規予約内容が既に予約済みだったとみなし, 戻り値に「既にその時間帯は予約されています」を設定.

isOverlap が false のとき :

MySQL の reservation テーブルに facility_id に facility, user_id に

reservationUserID, date に now, day に rdate, start_time に st, end_time に et を

引き渡し, 新規予約を実行する. 戻り値に「予約されました」を設定.

MySQL を解除.

非ログイン時 :

戻り値に「ログインしてください」を設定.

3.3.16 getAvailableTime メソッド

MainFrame の新規予約ボタン押下時に, ResrvationControl により ReservationDialog から呼び出され, 予約該当教室の利用可能開始・終了時間を取得する.

本メソッドでは, MySQL に接続し, facility から facility_id を取得することで該当教室の open_time, close_time を参照することが可能であり, これらは他クラスに引き渡すための戻り値に設定される.

3.3.17 getFacility_Reservation メソッド

MainFrame の教室予約状況ボタン押下時に, MainFrame により呼び出され, 引数として予約状況を受け取る.

本メソッドでは, ログイン・非ログインにかかわらず教室予約状況を確認可能である.

以下は, 本メソッドで使用する変数, フラグ及びリストである.

ryear_str : ReservationCheckDialog にて入力された予約日の年を本メソッドに引き渡す.

rmonth_str : ReservationCheckDialog にて入力された予約日の月を本メソッドに引き渡す.

また, 引き取った値が一桁だった場合, 頭に 0 を足す.

例) 入力された値が 5 の時, 「05」にする.

rday_str : ReservationCheckDialog にて入力された予約日の日を本メソッドに引き渡す.

また, 引き取った値が一桁だった場合, rmonth_str と同様に頭に 0 を足す.

rdate : 本メソッドにて受け取った ryear_str, rmonth_str, rday_str を各間に“-”を入れ,

「ryear_str + "-" + rmonth_str + "-" + rday_str」の形にする.

convData : java 内にある SimpleDateFormat クラスを用いて西暦 4 桁, 月は 1~12 月, 日は

1~31(毎月末日まで)の形である「yyyy-MM-dd」に rdate をフォーマットする.

facility : ReservationCheckDialog にて選択された教室 ID を本メソッドに引き渡す.

date : reservation テーブルから該当する day を取得する.

また, 当変数は reservation テーブル上に予約があるかの確認にも用いる.

startTime : reservation テーブルから該当する start_time を取得する.

endTime : reservation テーブルから該当する end_time を取得する.

TimeList : startTime と endTime の時間・分までを取得し以下の形にする.

「startTime.substring(0,5)+"~"+endTime.substring(0,5)」

整形したデータはリストデータとして格納される.

TimeListString : TimeList にて全てのデータ格納後, 戻り値に使用するため, 各データ間に“, ”
を入れ, String 型に変換する.

以下は, 本メソッドの動作である.

MainFrame の教室概要ボタン押下時 :

ReservationCheckDialog を呼び出す.

ReservationCheckDialog のキャンセルボタン、または×ボタン押下時 :

戻り値に「予約状況の確認をキャンセルしました」を設定.

ReservationCheckDialog の教室予約状況ボタン押下時 :

ryear_str, rmonth_str, rday_str, facility を新規予約ダイアログから取得.

rmonth_str と rday_str において, 各々の値が 1 桁であるとき :

前に 0 を付加し, 再度各変数に引き渡す.

rdate を ryear_str, rmonth_str, rday_str から作成.

convData で rdate の値を「yyyy-MM-dd」にフォーマット.

convData が rdate と等しくない,もしくは ryear_str に入力された西暦が 4 桁以外のとき: 戻り値に「日付の書式を修正して下さい(年: 西暦 4 桁, 月: 1~12, 日: 1~31(毎月末日まで))」を設定.

convData の値が存在しない日付であったとき:

convData の値が「月は 1~12 月, 日は 1~31(毎月末日まで)」を守っていない,もしくは「2025 年 2 月 29 日」のような,うるう年でないのに関わらず「2 月 29 日」を設定してしまっているとき, 戻り値に「日付の値を修正してください」を設定.

MySQL に接続.

MySQL の reservation テーブルから day と rdate, facility_id と facility が一致する day, start_time, end_time を date, startTime, endTime にて取得する.

TimeList にて, 取得した startTime と endTime の時間・分までを取得し以下の形にする.
「startTime.substring(0,5)+"~"+endTime.substring(0,5)」

整形したデータはリストデータとして格納される.

rdate と date の値が等しいとき:

入力された予約日と reservation テーブル上の予約日が一致しているとみなし, 同一教室, 同一日に予約があったと判断し以下の戻り値を設定する.

「facility + "教室、" + rdate + " に以下の時間帯が予約されています: " + "\n" + TimeListString」

rdate と date の値が等しくないとき:

入力された予約日と reservation テーブル上の予約日が一致していないとみなし, 同一教室, 同一日に予約がないと判断し以下の戻り値を設定する.

「facility + "教室は予約可能です"」

MySQL を解除.

3.3.18 MyCheckSetting メソッド

MainFrame の自己予約確認設定ボタン押下時に, MainFrame からの呼び出しに応じ, 本メソッドが終了した要因に従い戻り値を設定する.

以下は, 本メソッドで使用する変数, フラグ及びリストである.

justNow: 本メソッドにて過去に対する設定を防止すべく, 現在時刻を java 内にある Calender クラスから現在の日時を取得する.

resDate: 指定された値を java 内にある SimpleDateFormat クラスを用いて西暦 4 桁, 月は 1~12 月, 日は 1~31(毎月末日まで), 時間, 分, 秒の形である「yyyy-MM-dd HH:mm:ss」に指定された値をフォーマットする.

now: justNow で受け取った値を resDate に引き渡しフォーマットする.

ryear_str1: MyReservationSettingDialog にて入力された開始日の年を本メソッドに引き渡す.

rmonth_str1: MyReservationSettingDialog にて入力された開始日の月を本メソッドに引き渡す.

また, 引き取った値が一桁だった場合, 頭に 0 を足す.

例) 入力された値が 5 の時, 「05」にする.

rday_str1 : MyReservationSettingDialog にて入力された開始日の日を本メソッドに引き渡す.

また,引き取った値が一桁だった場合,rmonth_str1 と同様に頭に 0 を足す.

ryear_str2 : MyReservationSettingDialog にて入力された終了日の年を本メソッドに引き渡す.

rmonth_str2 : MyReservationSettingDialog にて入力された終了日の月を本メソッドに引き渡す.

また,引き取った値が一桁だった場合,rmonth_str1 と同様に頭に 0 を足す.

rday_str2 : MyReservationSettingDialog にて入力された終了日の日を本メソッドに引き渡す.

また,引き取った値が一桁だった場合,rmonth_str1 と同様に頭に 0 を足す.

rdate1 : 本メソッドにて受け取った ryear_str1, rmonth_str1, rday_str1 を各間に“-”を入れ,
「ryear_str1 + "-" + rmonth_str1 + "-" + rday_str1」の形にする.

rdate2 : 本メソッドにて受け取った ryear_str2, rmonth_str2, rday_str2 を各間に“-”を入れ,
「ryear_str2 + "-" + rmonth_str2 + "-" + rday_str2」の形にする.

convData : java 内にある SimpleDateFormat クラスを用いて西暦 4 桁, 月は 1~12 月, 日は
1~31(毎月末日まで)の形である「yyyy-MM-dd」に rdate1 及び,rdate2 を
フォーマットする.

day1 : rdate1 にて整形した開始日を引き渡す.

day2 : rdate2 にて整形した終了日を引き渡す.

Set_date1 : 3.3.4 のものと同一である,ReservationControl にて設定されている開始日格納変
数.day1 の値を引き渡す.

Set_date2 : 3.3.5 のものと同一である,ReservationControl にて設定されている終了日格納変
数.day2 の値を引き渡す.

facility_id : 3.3.6 のものと同一である,ReservationControl 内にて設定されている
選択教室格納変数.

MyReservationSettingDialog にて選択された教室を記憶する.

ResetSetting : 3.3.7 のものと同一である,ReservationControl 内にて設定されている
オプション仕様切り替えフラグ.

以下は,本メソッドの動作である.

MainFrame の自己予約確認設定ボタン押下時 :

MyReservationSettingDialog を呼び出す.

Calender クラスから justNow を取得し,取得したデータを resDate でフォーマットし,
その結果を now に引き渡す.

ログイン時 :

MyReservationSettingDialog のキャンセルボタン、または×ボタン押下時 :

戻り値に「自己予約確認 Option 設定をキャンセルしました」を設定.

MyReservationSettingDialog の初期化ボタン押下時 :

Set_date1,Set_date2 に値が入っているか否かに拘わらず MyReservationSetting
Dialog 側で ResetSetting フラグが true になる.

また,ResetSetting を本メソッドに引き渡す.

ResetSetting が true のとき：

ReservationControl 内の MyReservationSettingDialog にて入力した結果を記憶する「Set_date1（指定期間開始日）」、「Set_date2（指定期間終了日）」、「ResetSetting（初期化フラグ）」にそれぞれ「null」、「false」を渡し、戻り値に「初期化成功」を設定。

また、初期化成功に伴い自己予約確認機能をミニマム仕様に切り替え成功したものとする。

ryear_str1, rmonth_str1, rday_str1, ryear_str2, rmonth_str2, rday_str2 を MyReservationSettingDialog から取得。

rmonth_str1 と rday_str1 において、各々の値が 1 桁であるとき：

前に 0 を付加し、再度各変数に引き渡す。

rdate1 を ryear_str1, rmonth_str1, rday_str1 から作成。

rmonth_str2 と rday_str2 において、各々の値が 1 桁であるとき：

前に 0 を付加し、再度各変数に引き渡す。

rdate2 を ryear_str2, rmonth_str2, rday_str2 から作成。

convData で rdate1 の値を「yyyy-MM-dd」にフォーマット。

convData が rdate1 と等しくない、もしくは ryear_str1 に入力された西暦が 4 桁以外
のとき： 戻り値に「期間開始日付の書式を修正して下さい(年：西暦 4 桁, 月：1~12,
日：1~31(毎月末日まで))」を設定。

rdate1 と now（現在時刻）を比較して rdate1 の方が小さいとき：

開始日の設定が設定当日以前であるとみなし、戻り値に「期間開始日が本日
以降でないため設定できません」

convData(rdate1)の値が存在しない日付であったとき：

convData の値が「月は 1~12 月, 日は 1~31(毎月末日まで)」を守っていない、
もしくは「2025 年 2 月 29 日」のような、うるう年でないのに関わらず
「2 月 29 日」を設定してしまっているとき、戻り値に「期間開始日付の値
を修正してください」を設定。

convData で rdate2 の値を「yyyy-MM-dd」にフォーマット。

convData が rdate2 と等しくない、もしくは ryear_str1 に入力された西暦が 4 桁以外
のとき： 戻り値に「期間終了日付の書式を修正して下さい(年：西暦 4 桁, 月：1~12,
日：1~31(毎月末日まで))」を設定。

convData(rdate2)の値が存在しない日付であったとき：

convData の値が「月は 1~12 月, 日は 1~31(毎月末日まで)」を守っていない、
もしくは「2025 年 2 月 29 日」のような、うるう年でないのに関わらず
「2 月 29 日」を設定してしまっているとき、戻り値に「期間終了日付の値
を修正してください」を設定。

facility_id を MyReservationSettingDialog から取得。

MyReservationSettingDialog の指定期間が開始日～終了日に沿った入力するとき：

Set_date1 に day1, Set_date2 に day2 を引き渡し、戻り値に「設定完了」を
設定。また、これに伴い自己予約確認設定完了とみなし、自己予約確認機能をオ

プッシュ仕様へ切り替える。

MyReservationSettingDialog の指定期間が開始日～終了日に沿った入力でないとき：戻り値に「指定期間が開始日～終了日に沿った入力でないため設定できません」を設定。

未ログイン時：

戻り値に「ログインしてください」を設定。

3.3.19 MyReservation メソッド

MainFrame の自己予約確認ボタン押下時に、MainFrame により呼び出しに応じ、本メソッドが終了した要因に従い戻り値を設定する。

また、MyReservationSettingDialog の設定に伴い、ミニマム仕様、オプション仕様へ切り替わり、これにより遷移が変化し戻り値が異なる。

以下は、本メソッドで使用する変数、フラグ及びリストである。

user_id：3.3.3 reservationUserID の値を引き渡す。

justNow：本メソッドにて過去の予約表示を防止すべく、現在時刻を java 内にある Calendar クラスから現在の日時を取得する。

resDate：指定された値を java 内にある SimpleDateFormat クラスを用いて西暦 4 桁、月は 1~12 月、日は 1~31(毎月末日まで)、時間、分、秒の形である「yyyy-MM-dd HH:mm:ss」に指定された値をフォーマットする。

now：justNow で受け取った値を resDate に引き渡しフォーマットする。

facility_id：3.3.18 MyCheckSetting にてオプション仕様設定時に取得した、MyReservation SettingDialog にて選択された教室を取得する。

facility：自己予約確認機能がミニマム仕様時、reservation テーブルから該当する facility_id を取得する。

date：reservation テーブルから該当する day を取得する。

また、当変数は reservation テーブル上に予約があるかの確認にも用いる。

startTime：reservation テーブルから該当する start_time を取得する。

endTime：reservation テーブルから該当する end_time を取得する。

MyList：facility_id, date, startTime と endTime の時間・分までを取得し以下の形にする。

「facility_id + "教室, " + date + ", " + startTime.substring(0,5) +
"~"+endTime.substring(0,5)」

整形したデータはリストデータとして格納される。

count：MyList に格納されているデータの数を取得する。

i：for 文にて MyList の各要素を取り出す際に用いる。初期値に“0”を設定されている。

以下は、本メソッドの動作である。

自己予約確認ボタン押下時：

ログイン時：

現在ログインしているユーザ ID を user_id で取得。

Calender クラスから justNow を取得し,取得したデータを resDate でフォーマットし,
その結果を now に引き渡す.

オプション仕様に切り替わっており, ResetSetting が false の時:

Set_date1 に値が格納されているため,オプション仕様であるとみなす.

MyCheckSetting にてオプション仕様設定時に取得した facility_id を取得する.

MySQL に接続する.

MySQL の reservation テーブルから Set_date1 以上,かつ Set_date2 以下の day,
本メソッドで取得した user_id と reservation テーブル上の user_id, 本メソッドで取得した facility_id と reservation テーブル上の facility が一致する day, start_time, end_time を date, startTime, endTime にて取得する.

MyList にて,facility_id, date, startTime と endTime の時間・分までを取得し以下の形にする.

「 facility_id + " 教室 , " + date + " , " +startTime.substring(0,5)+" ~ "+endTime.substring(0,5)」

整形したデータはリストデータとして格納される.

MyList を予約日の時間順にソートする.

count にて MyList に格納されているデータの数を取得する.

date が null のとき:

reservation テーブルに該当する情報がなかったとみなし,

戻り値に「指定された期間には," + user_id + "様の予約がありませんでした」を設定.

date に値が引き渡されている時:

reservation テーブルに該当する情報があったとみなし,

戻り値に「user_id + "様の指定された," + Set_date1 + " ~ " + Set_date2 + "の期間の予約が見つかりました" + "¥n"」を追加.

for 文で MyList から要素を取り出すごとに, i に“1”を足し,i が count と同一の値になるまで繰り返す:

戻り値に「MyList.get(i)+ "¥n"」を追加.

最終的に戻り値には「user_id + "様の指定された," + Set_date1 + " ~ " + Set_date2 + "の期間の予約が見つかりました" + "¥n" + MyList.get(0)+ "¥n" + MyList.get(1)+ "¥n" + … + MyList.get(i)+ "¥n"」が設定される.

ミニマム仕様に切り替わっているとき:

MySQL に接続する.

reservation テーブルから, 本メソッドで取得した user_id と reservation テーブル上の user_id が一致し, 本メソッドで取得した now 以下の reservation テーブル上の day を所有している row の day, facility_id, start_time, end_time を取得し, それぞれ date, facility, startTime, endTime に格納する.

MyList にて,facility, date, startTime と endTime の時間・分までを取得し以下の形にする.

「 facility + " 教室 , " + date + " , " + startTime.substring(0,5)+" ~ "+endTime.substring(0,5)」

整形したデータはリストデータとして格納される.

date が now より大きいとき :

MyList のデータを予約日の時間順にソートする.

count にて MyList に格納されているデータの数を取得する.

確認当日以降の予約があるとみなし,

戻り値に「"ユーザーID:" + user_id + "様が登録された本日以降の予約は以下の通りです"+ "\n"」を追加.

for 文で MyList から要素を取り出すごとに, i に“1”を足し,i が“20”になるまで繰り返す :

戻り値に「MyList.get(i)+ "\n"」を追加.

戻り値に「"" + "\n"」を追加.

count が 20 以上のとき :

戻り値に「" 予約表示件数 (20/" + count + " 件) "」を追加.

最終的に戻り値には「"ユーザーID:" + user_id + "様が登録された本日以降の予約は以下の通りです"+ "\n"+ MyList.get(0)+ "\n"+ MyList.get(1)+ "\n"+ ... + MyList.get(i)+ "\n" + "" + "\n" +"

予約表示件数 (20/" + count + " 件) "」が設定される.

count が 20 未満のとき :

戻り値に「" 予約表示件数 (" +count+"/" + count + " 件) "」を追加.

最終的に戻り値には「"ユーザーID:" + user_id + "様が登録された本日以降の予約は以下の通りです"+ "\n"+ MyList.get(0)+ "\n"+ MyList.get(1)+ "\n"+ ... + MyList.get(i)+ "\n" + "" + "\n" +"

予約表示件数 (" +count+"/" + count + " 件) "」

が設定される.

MySQL を解除.

未ログイン時 :

戻り値に「ログインしてください」を設定.

3.3.20 ReservationCancel メソッド

MainFrame の予約キャンセルボタン押下時に, MainFrame により呼び出しに応じ,本メソッドが終了した要因に従い戻り値を設定する.

以下は, 本メソッドで使用する変数,フラグ及びリストである.

ryear_str : CancelDialog にて入力された予約日の年を本メソッドに引き渡す.

rmonth_str : CancelDialog にて入力された予約日の月を本メソッドに引き渡す.

また,引き取った値が一桁だった場合,頭に 0 を足す.

例) 入力された値が 5 の時,「05」にする.

rday_str : CancelDialog にて入力された予約日の日を本メソッドに引き渡す.

また,引き取った値が一桁だった場合,rmonth_str と同様に頭に 0 を足す.

justNow : 本メソッドにて過去に対する予約を防止すべく,現在時刻を java 内にある Calender クラスから現在の日時を取得する.

resDate : 指定された値を java 内にある SimpleDateFormat クラスを用いて西暦 4 桁, 月は 1~12 月, 日は 1~31(毎月末日まで), 時間,分,秒の形である「yyyy-MM-dd HH:mm:ss」に指定された値をフォーマットする.

now : justNow で受け取った値を resDate に引き渡しフォーマットする.

facility : CancelDialog にて選択された教室 ID を本メソッドに引き渡す.

st : CancelDialog にて選択された開始時刻(startHour, startMinute)を「startHour : startMinute」の形にする.

et : CancelDialog にて選択された終了時刻(endHour, endMinute)を「endHour : endMinute」の形にする.

rdate : 本メソッドにて受け取った ryear_str, rmonth_str, rday_str を各間に“-”を入れ,「ryear_str + "-" + rmonth_str + "-" + rday_str」の形にする.

convData : java 内にある SimpleDateFormat クラスを用いて西暦 4 桁, 月は 1~12 月, 日は 1~31(毎月末日まで)の形である「yyyy-MM-dd」に rdate をフォーマットする.

reservationTimeCK : convData と st の値を引き継ぎ,各間にスペースを入れ,「convData + " "+ st」の形にする.

rt : reservationTimeCK で整形した値を resDate に引き渡しフォーマットする.

startTime : reservation テーブルから該当する start_time を取得する.

endTime : reservation テーブルから該当する end_time を取得する.

CancelList_start : startTime の取得データをリストデータとして追記する.

CancelList_end : endTime の取得データをリストデータとして追記する.

count : CancelList_start に格納されているデータの数を記憶する.

isExist : boolean 形式のフラグ,初期状態では false が設定されており,予約キャンセル対象が reservation テーブルにあるかのチェックに用いる.

以下は、本メソッドの動作である。

ログイン時：

CancelDialog を呼び出す。

CancelDialog のキャンセルボタン、または×ボタン押下時：

戻り値に「予約取り消しをキャンセルしました」を設定。

CancelDialog の予約取り消しボタン押下時：

ryear_str, rmonth_str, rday_str を CancelDialog から取得。

Calender クラスから justNow を取得し、取得したデータを resDate でフォーマットし、その結果を now に引き渡す。

facility, st, et を CancelDialog から取得。

rmonth_str と rday_str において、各々の値が 1 桁であるとき：

前に 0 を付加し、再度各変数に引き渡す。

rdate を ryear_str, rmonth_str, rday_str から作成。

convData で rdate の値を「yyyy-MM-dd」にフォーマット。

convData が rdate と等しくない、もしくは ryear_str に入力された西暦が 4 桁以外のとき：戻り値に「日付の書式を修正して下さい(年：西暦 4 桁, 月：1~12, 日：

1~31(毎月末日まで))」を設定。

reservationTimeCK を convData と st の値から整形。

rt を reservationTimeCK から作成。

rt(予約日時)と now (現在時刻)を比較して rt の方が小さいとき：

戻り値に「過去に対する予約キャンセルはできません」を設定。

convData の値が存在しない日付であったとき：

convData の値が「月は 1~12 月, 日は 1~31(毎月末日まで)」を守っていない、もしくは「2025 年 2 月 29 日」のような、うるう年でないのに関わらず「2 月 29 日」を設定してしまっているとき、戻り値に「日付の値を修正してください」を設定。

st(開始時刻)と et(終了時刻)を比較して同じ、もしくは st が et より大きいとき：

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。

MySQL に接続。

MySQL の reservation テーブルから day と rdate, facility_id と facility が一致する start_time, end_time を startTime, endTime にて取得し、各値を CancelList_start, CancelList_end に格納する。

isExist フラグを初期化(false)にする。

count にて CancelList_start の格納しているデータの数を取得する。

for 文で count の値の分だけ,CheckList_start に格納されている値と et(終了時刻), CheckList_end に格納されている値と st(開始時刻)を順番に比較する：

CheckList_start の値が et より小さく,CheckList_end の値が st より大きいとき：

isExist フラグを true にする。

isExist が true のとき：

予約キャンセル対象が reservation テーブル上にあったとみなし,
MySQL の reservation テーブルから day と rdate, facility_id と facility,
start_time と st, end_time と et が一致する row の予約を削除し,予約を取り
消す.また, 戻り値に「facility+"教室 "+rdate+", "+st.substring(0,5)+"~
"+et.substring(0,5)+" 枠の予約キャンセルに成功しました"」を設定.

isExist が false のとき：

予約キャンセル対象が reservation テーブル上にないとみなし,戻り値に「キ
ャンセル対象が存在しません」を設定.

MySQL を解除.

非ログイン時：

戻り値に「ログインしてください」を設定.

3.4 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於いて、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。

3.4.1 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。

3.4.2 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。)

3.4.3 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。

3.4.4 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。

3.5 ChoiceFacility

ChoiceFacility
+ChoiceFacility(List<String>)

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。

3.5.1 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 ChoiceHour

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

ChoiceHour class は、Choice クラスを継承した派生クラスで、予約時を指定する選択項目を登録する。

3.6.1 ChoiceHour コンストラクタ

選択項目に 9～21 までの値を登録する。

3.6.2 resetRange メソッド

resetRange メソッドは、開始時と終了時を指定すると、その間の時間を選択項目として登録する。

3.6.3 getFirst メソッド

getFirst メソッドは、選択可能な最初の選択項目（選択項目の一番小さい数字）を返す。

3.6.4 getLast メソッド

getLast メソッドは、選択可能な最終の選択項目（選択項目の一番大きい数字）を返す。

3.7 ChoiceMinute

ChoiceMinute
+ChoiceMinute()

ChoiceMinute class は、Choice クラスの派生クラスで、予約分（「00」「15」「30」「45」）を指定する選択項目を登録する。本クラス独自のメソッドは持たない。

3.7.1 ChoiceMinute コンストラクタ

選択項目に、「00」「15」「30」「45」という値を登録する。

3.8 ReservationDialog

ReservationDialog
~boolean canceled ~ReservationControl rc ~Panel panelNorth ~Panel panelCenter ~Panel panelSouth ~ChoiceFacility choiceFacility ~TextField tfYear, tfMonth, tfDay ~ChoiceHour startHour ~ChoiceMinute startMinute ~ChoiceHour endHour ~ChoiceMinute endMinute ~Button buttonOK ~Button buttonCancel
+ReservationDialog(Frame, ReservationControl) +void itemStateChanged(ItemEvent) +void resetTimeRange(String) +void windowClosing(WindowEvent) +void actionPerformed(ActionEvent)

ReservationDialog class は、ユーザが新規予約する際に指定教室、予約日、予約時間を入力するためのダイアログに於いて、View の役割を果たす。必要要件入力後、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。

3.8.1 canceled

この変数は新規予約ダイアログを閉じる際に、ユーザが予約登録処理を行うか予約登録処理をキャンセルするかを決めるためのものである。新規予約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、以降の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、予約登録動作に移行する。

3.8.2 ReservationDialog 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：新規予約, 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予約実行ボタンが設定される。)

3.8.3 resetTimeRange メソッド

本メソッドは、新規予約ダイアログに於ける予約時間の選択ボックスの項目において、ユーザの任意の教室から、その教室の open_time から close_time までの時間を取得し、各選択ボックスに入力を行う。

3.8.4 itemStateChanged メソッド

本メソッドは、新規予約ダイアログに於ける、予約時間の選択ボックスで選択項目が変化した場合に呼び出され、各項目が変更される度、呼び出される resetTimeRange メソッドにより、設定されている初期値を読み込んだ後、改めて情報を取得する。これによりユーザの任意の教室の利用可能開始時刻・終了時刻を読み込み、適宜予約が可能になる。

3.8.5 windowClosing メソッド

本メソッドは、終了ボタン押下時に呼出され、新規予約ダイアログを非表示に設定、canceled 変数は初期設定の true のまま、ダイアログを削除する。

3.8.6 actionPerformed メソッド

本メソッドは、キャンセルボタンもしくは予約実行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初期設定の true のまま、予約実行ボタン押下時は false が設定され、その後新規ダイアログを非表示にし、ダイアログを削除する。

3.9 ReservationCheckDialog

ReservationCheckDialog
~boolean canceled ~ReservationControl rc ~Panel panelNorth ~Panel panelSouth ~ChoiceFacility choiceFacility ~TextField tfYear, tfMonth, tfDay ~Button buttonOK ~Button buttonCancel
+ReservationCheckDialog(Frame, ReservationControl) +void windowClosing(WindowEvent) +void actionPerformed(ActionEvent)

ResrvationCheckDialog class は、ユーザが教室予約状況確認する際に指定教室、予約日を入力するためのダイアログに於いて、View の役割を果たす。必要要件入力後、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。

3.9.1 canceled

この変数は教室予約状況ダイアログを閉じる際に、ユーザが予約状況確認処理を行うか予約状況確認処理をキャンセルするかを決めるためのものである。教室予約状況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、以降の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、予約状況確認動作に移行する。

3.9.2 ReservationCheckDialog 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：教室予約状況, 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予約状況確認ボタンが設定される。)

3.9.3 windowClosing メソッド

本メソッドは、終了ボタン押下時に呼出され、教室予約状況ダイアログを非表示に設定、canceled 変数は初期設定の true のまま、ダイアログを削除する。

3.9.4 actionPerformed メソッド

本メソッドは、キャンセルボタンもしくは予約状況確認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初期設定の true のまま、予約状況確認ボタン押下時は false が設定され、その後教室予約状況ダイアログを非表示にし、ダイアログを削除する。

MyReservationSettingDialog
~boolean canceled ~boolean ResetSetting ~ReservationControl rc ~Panel PanelNorth ~Panel PanelNorthSub1 ~Panel PanelNorthSub2 ~Panel panelSouth ~ChoiceFacility choiceFacility ~TextField tfYear_start, tfMonth_start, tfDay_start, tfYear_end, tfMonth_end, tfDay_end ~Button buttonOK ~Button buttonCancel ~Button buttonReset
+ MyReservationSettingDialog (Frame, ReservationControl) +void windowClosing(WindowEvent) +void actionPerformed(ActionEvent)

MyReservationSettingDialog class は、ユーザが自己予約確認設定（自己予約確認機能のオプション仕様化）する際に指定教室, 指定期間（開始日・終了日）を入力するためのダイアログに於いて、View の役割を果たす。必要要件入力後、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、または変数 ResetSetting に初期化処理を行うか設定する。

3.10.1 canceled

この変数は自己予約確認 Option 設定ダイアログを閉じる際に、ユーザが自己予約確認設定処理を行うか自己予約確認設定処理をキャンセルするかを決めるためのものである。自己予約確認 Option 設定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、以降の処理はキャンセル扱いとなる。また、設定完了ボタン, 初期化ボタンにて閉じられた場合は false が設定され、自己予約確認設定動作に移行する。

3.10.2 ResetSetting

この変数は自己予約確認 Option 設定ダイアログの初期化ボタンを押下し、ダイアログを閉じる際に、初期化フラグを true に設定し、ReservationControl class の MyReservation メソッドにてミニマム仕様に切り替えるかを定めるためのものである。また、ResetSetting は基本的に初期化ボタンが押下されない限り、false に設定されており、押下された場合は上記の通り、true に設定し自己予約確認設定動作に移行する。

3.10.3 MyReservationSettingDialog 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：自己予約確認 Option 設定, 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,初期化ボタン及び設定完了ボタンが設定される.)

3.10.4 windowClosing メソッド

本メソッドは、終了ボタン押下時に呼出され、自己予約確認 Option 設定ダイアログを非表示に設定、canceled 変数は初期設定の true のまま、ダイアログを削除する。

3.10.5 actionPerformed メソッド

本メソッドは、キャンセルボタン,初期化ボタン,または予約状況確認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初期設定の true のまま、予約状況確認ボタン,初期化ボタン押下時は false が設定され、ResetSetting 変数には、予約状況確認ボタン,キャンセルボタン押下時は false のまま,初期化ボタン押下時は true が設定され、その後自己予約確認 Option 設定ダイアログを非表示にし、ダイアログを削除する。

3.11 CancelDialog

CancelDialog
<div>~boolean canceled</div> <div>~ReservationControl rc</div> <div>~Panel PanelNorth</div> <div>~Panel PanelNorthSub1</div> <div>~Panel PanelNorthSub2</div> <div>~Panel panelCenter</div> <div>~Panel panelSouth</div> <div>~ChoiceFacility choiceFacility</div> <div>~TextField tfYear, tfMonth, 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>+CancelDialog(Frame, ReservationControl)</div> <div>+void itemStateChanged(ItemEvent)</div> <div>+void resetTimeRange(String)</div> <div>+void windowClosing(WindowEvent)</div> <div>+void actionPerformed(ActionEvent)</div>

CancelDialog class は、ユーザが予約キャンセルする際に指定教室, 予約日, 予約時間を入力するためのダイアログに於いて、View の役割を果たす。必要要件入力後、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。

3.11.1 canceled

この変数は予約キャンセルダイアログを閉じる際に、ユーザが予約取り消し対象登録処理を行うか予約取り消し対象登録処理をキャンセルするかを決めるためのものである。予約キャンセル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、以降の処理はキャンセル扱いとなる。また、予約取り消しボタンにて閉じられた場合は false が設定され、予約取り消し動作に移行する。

3.11.2 CancelDialog 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：予約キャンセル, 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予約取り消しボタンが設定される。)

3.11.3 resetTimeRange メソッド

本メソッドは, 予約キャンセルダイアログに於ける予約時間の選択ボックスの項目において, ユーザの任意の教室から, その教室の open_time から close_time までの時間を取得し, 各選択ボックスに入力を行う.

3.11.4 itemStateChanged メソッド

本メソッドは, 予約キャンセルダイアログに於ける, 予約時間の選択ボックスで選択項目が変化した際に呼び出され, 各項目が変更される度, 呼び出される resetTimeRange メソッドにより, 設定されている初期値を読み込んだ後, 改めて情報を取得する. これによりユーザの任意の教室の利用可能開始時刻・終了時刻を読み込み, 適宜予約キャンセル登録が可能になる.

3.11.5 windowClosing メソッド

本メソッドは, 終了ボタン押下時に呼出され, 予約キャンセルダイアログを非表示に設定, canceled 変数は初期設定の true のまま, ダイアログを削除する.

3.11.6 actionPerformed メソッド

本メソッドは, キャンセルボタンもしくは予約取り消し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初期設定の true のまま, 予約取り消しボタン押下時は false が設定され, その後予約キャンセルダイアログを非表示にし, 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 新規予約ダイアログ表示

4.4.2 新規予約ダイアログ表示不可

4.4.3 予約情報入力～登録完了

4.4.4 予約情報入力～教室 ID, 予約時間変更

4.4.5 予約情報入力～予約情報不適切

4.4.5.1 予約情報入力～予約西暦不適切

4.4.5.2 予約情報入力～予約日不適切

4.4.5.3 予約情報入力～予約日付不適切

4.4.5.4 予約情報入力～予約時間不適切

4.4.6 予約情報入力～登録失敗

4.4.7 新規予約中止

4.5 教室予約状況確認機能

4.5.1 教室予約状況ダイアログ表示

4.5.2 予約状況確認～出力成功

4.5.3 予約状況確認～出力失敗

4.5.4 予約情報入力～教室 ID 変更

4.5.5 予約情報入力～予約情報不適切

4.5.5.1 予約状況確認～西暦 4 桁以上

4.5.5.2 予約状況確認～日付不適切

4.5.6 予約状況確認中止

4.6 自己予約確認設定機能

4.6.1 自己予約確認 Option 設定ダイアログ表示

4.6.2 自己予約確認 Option 設定ダイアログ表示不可

4.6.3 設定情報入力～設定完了

4.6.4 設定情報初期化

4.6.5 設定情報入力～教室 ID 変更

4.6.6 設定情報入力～設定情報不適切

4.6.6.1 設定情報入力～設定開始日西暦不適切

- 4.6.6.2 設定情報入力～設定開始日不適切
- 4.6.6.3 設定情報入力～設定開始日日付不適切
- 4.6.6.4 設定情報入力～設定終了日西暦不適切
- 4.6.6.5 設定情報入力～設定終了日日付不適切
- 4.6.6.6 設定情報入力～設定開始日と終了日が逆に入力
- 4.6.7 自己予約確認設定中止

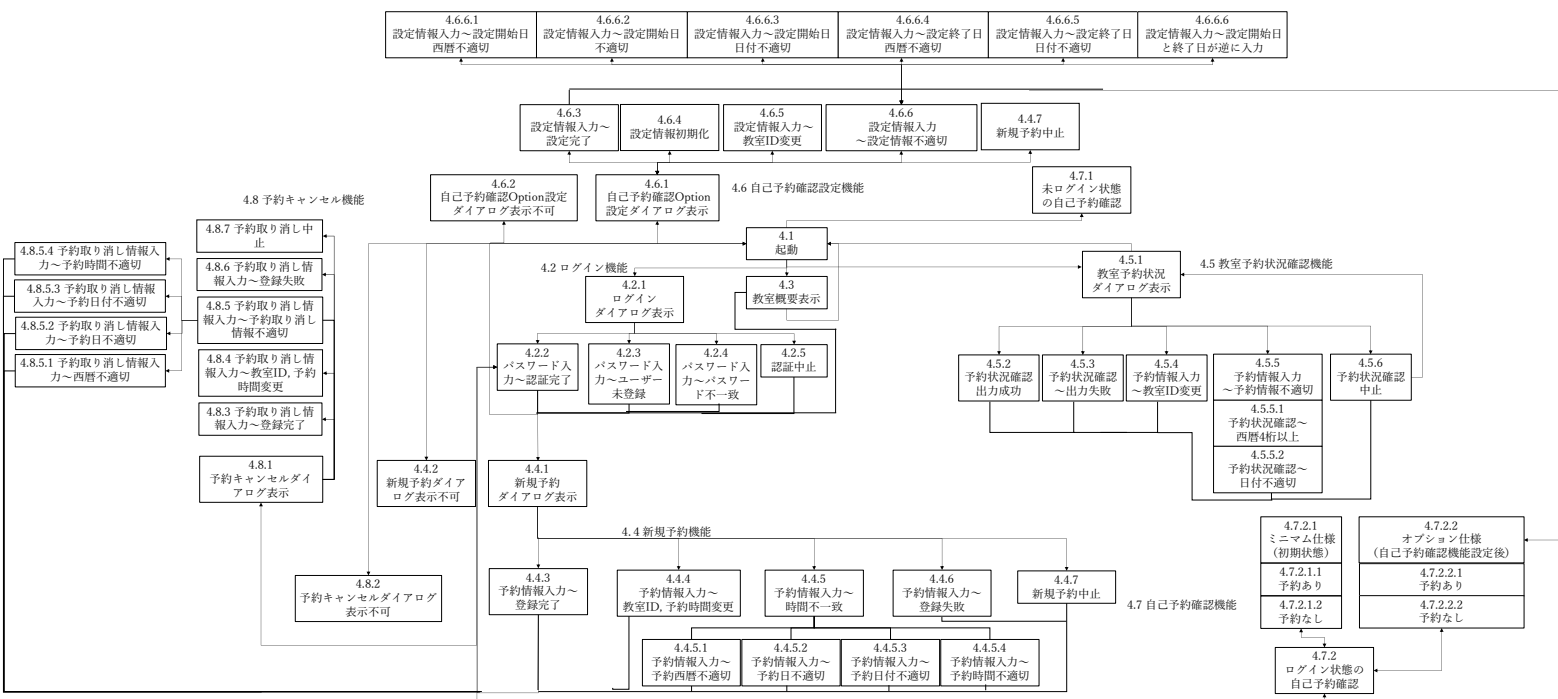
4.7 自己予約確認機能

- 4.7.1 未ログイン状態の自己予約確認
- 4.7.2 ログイン状態の自己予約確認
 - 4.7.2.1 ミニマム仕様（初期状態）
 - 4.7.2.1.1 予約あり
 - 4.7.2.1.2 予約なし
 - 4.7.2.2 オプション仕様（自己予約確認機能設定後）
 - 4.7.2.2.1 予約あり
 - 4.7.2.2.2 予約なし

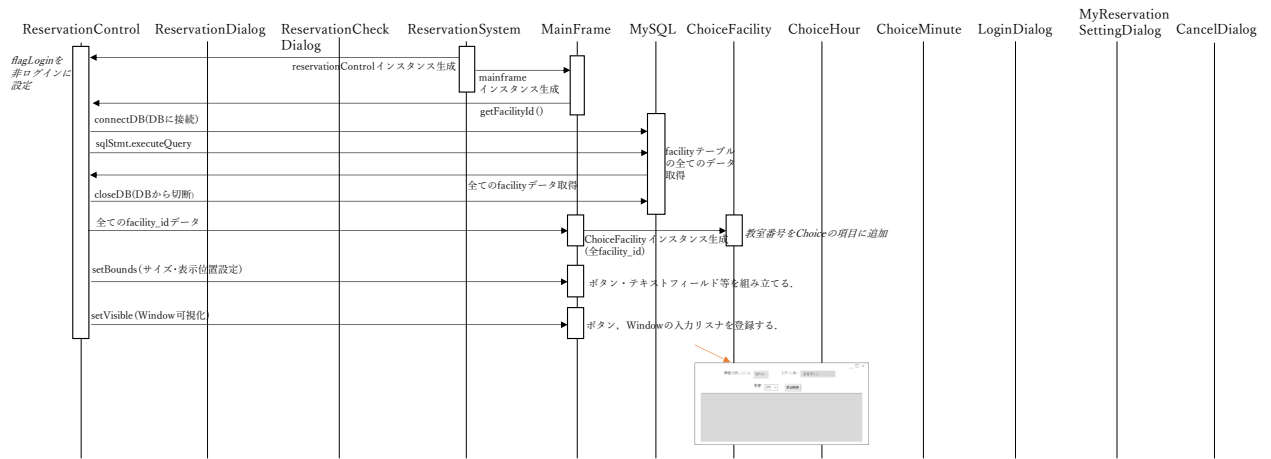
4.8 予約キャンセル機能

- 4.8.1 予約キャンセルダイアログ表示
- 4.8.2 予約キャンセルダイアログ表示不可
- 4.8.3 予約取り消し情報入力～登録削除完了
- 4.8.4 予約取り消し情報入力～教室 ID, 予約時間変更
- 4.8.5 予約取り消し情報入力～予約取り消し情報不適切
 - 4.8.5.1 予約取り消し情報入力～西暦不適切
 - 4.8.5.2 予約取り消し情報入力～予約日不適切
 - 4.8.5.3 予約取り消し情報入力～予約日付不適切
 - 4.8.5.4 予約取り消し情報入力～予約時間不適切
- 4.8.6 予約取り消し情報入力～登録削除失敗
- 4.8.7 予約取り消し中止

各シーケンス間の繋がりを以下に示す.



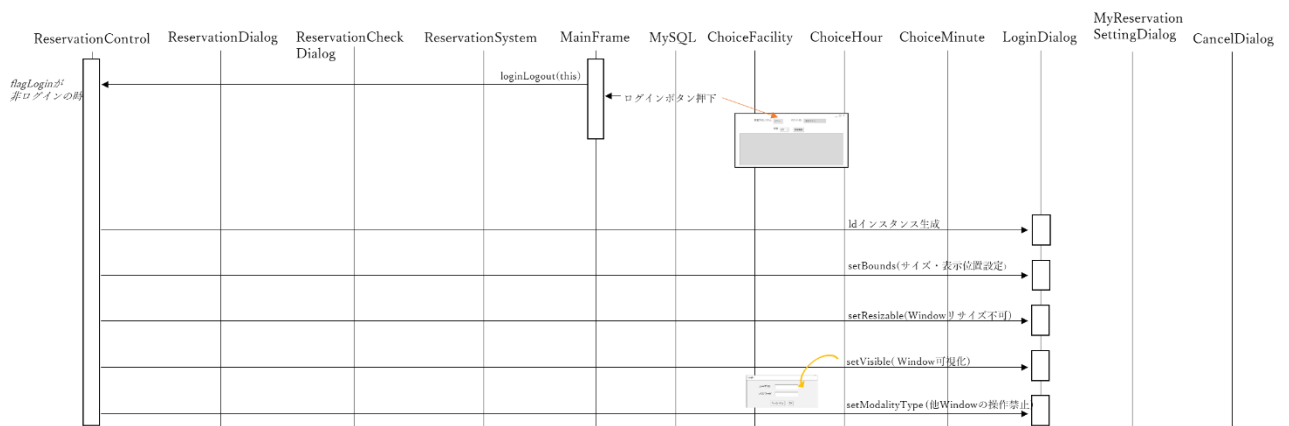
4.1 起動



4.2 ログイン機能

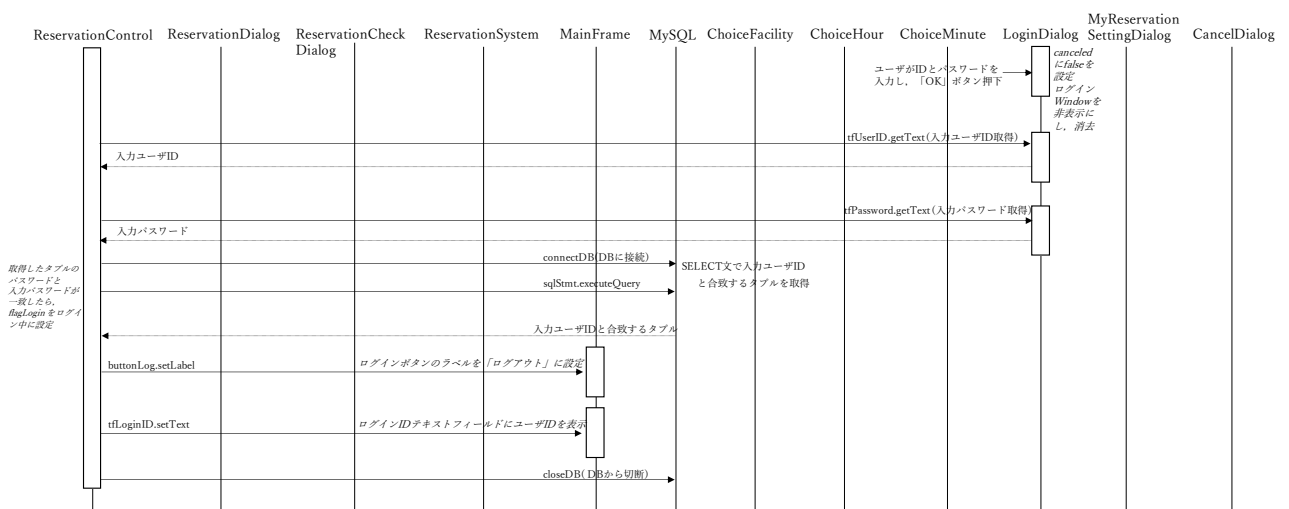
4.2.1 ログインダイアログ表示

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）



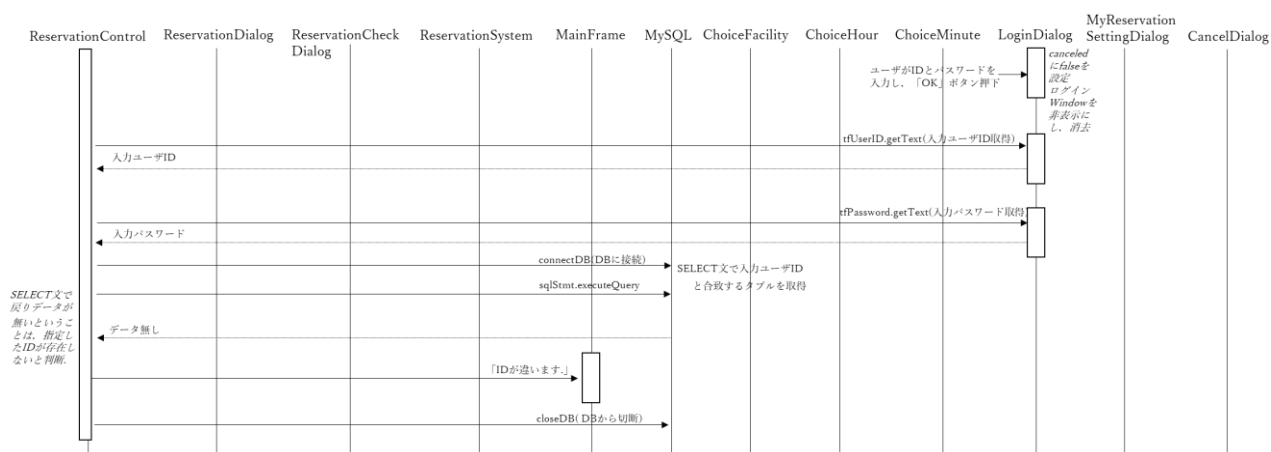
4.2.2 ID、パスワード入力～認証完了

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）



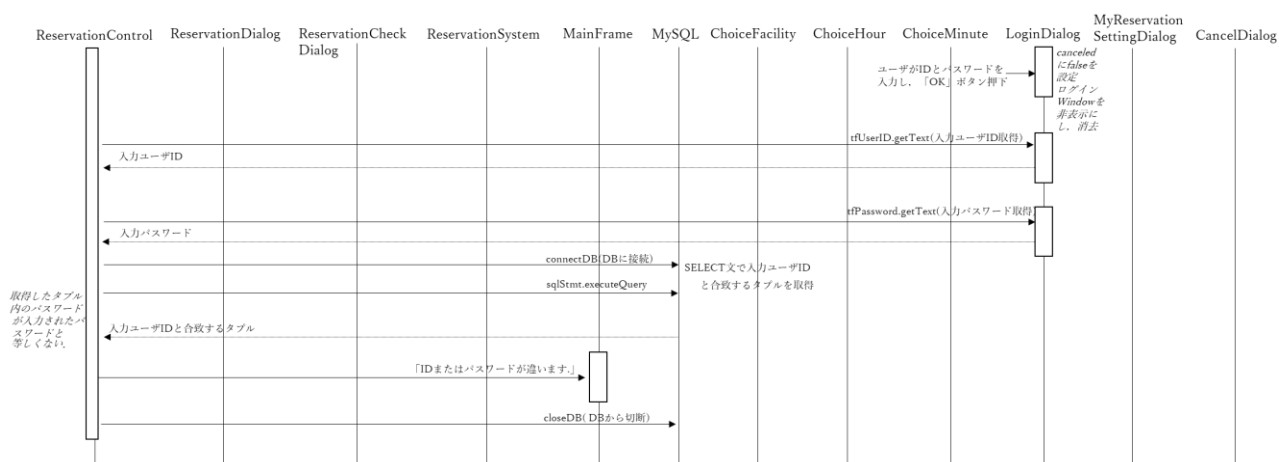
4.2.3 ID、パスワード入力～ユーザ ID 未登録

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)



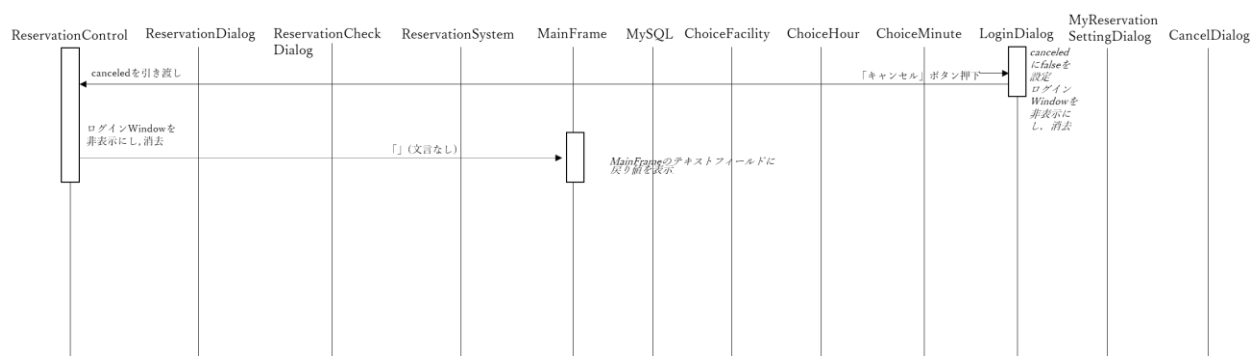
4.2.4 ID、パスワード入力～パスワード不一致

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)



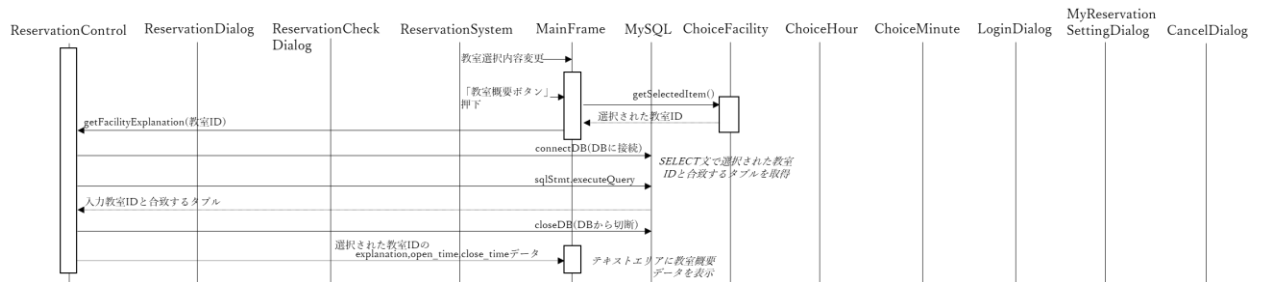
4.2.5 認証中止

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

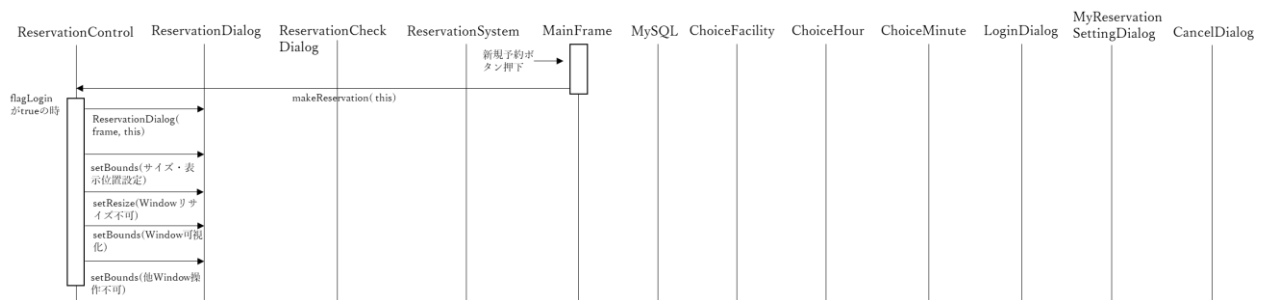
本シーケンスは、MainFrame のみ表示中の状態で、コンボボックスより教室選択を行い、教室概要ボタンを押下した時のものである。(ex.4.1 起動後からの継続, 4.2.2 ID, パスワード入力～認証完了)



4.4 新規予約機能

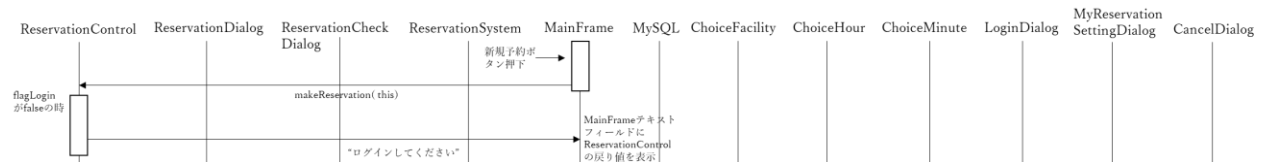
4.4.1 新規予約ダイアログ表示

本シーケンスは、MainFrame 表示中かつログイン状態で、新規予約ボタンを押下した時のものである。(ex.4.2.2 ID, パスワード入力～認証完了からの継続)



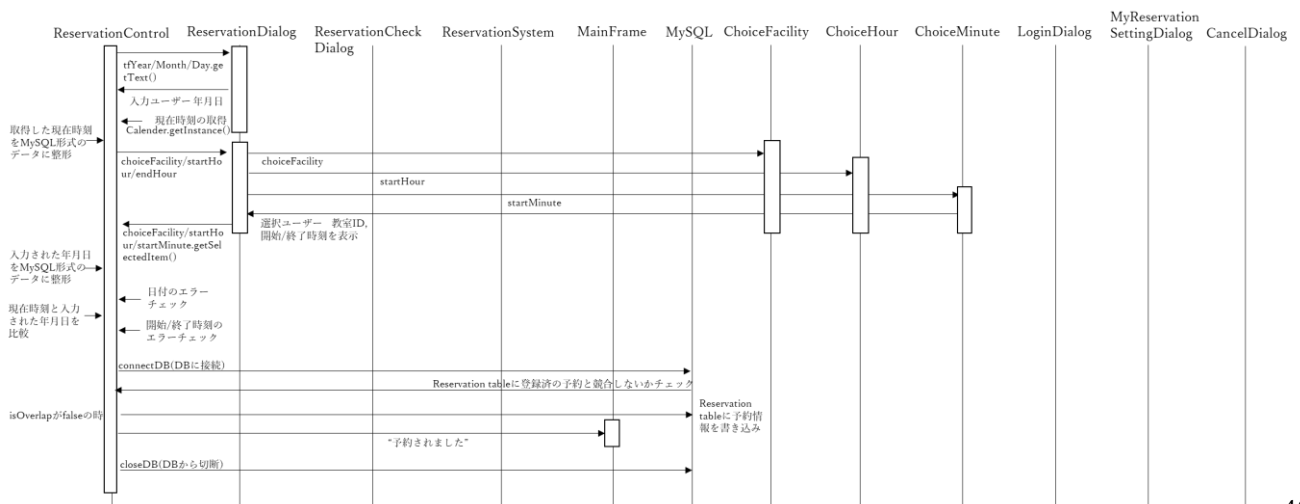
4.4.2 新規予約ダイアログ表示不可

本シーケンスは、MainFrame 表示中かつ非ログイン状態で、新規予約ボタンを押下した時のものである。(ex.4.1 起動からの継続)



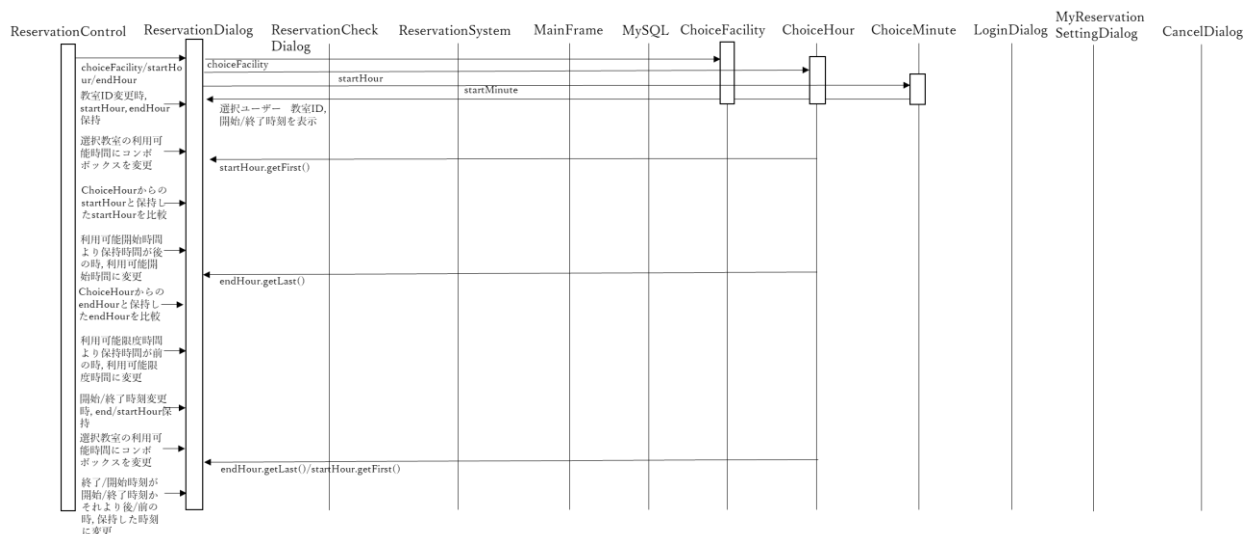
4.4.3 予約情報入力～登録完了

本シーケンスは、新規予約ダイアログ表示中にコンボボックスより教室選択, 予約時間選択を行い, 予約日を入力し, 入力に問題なく OK ボタンを押下した時のものである。(ex.4.4.1 新規予約ダイアログ表示からの継続)



4.4.4 予約情報入力～教室 ID, 予約時間変更

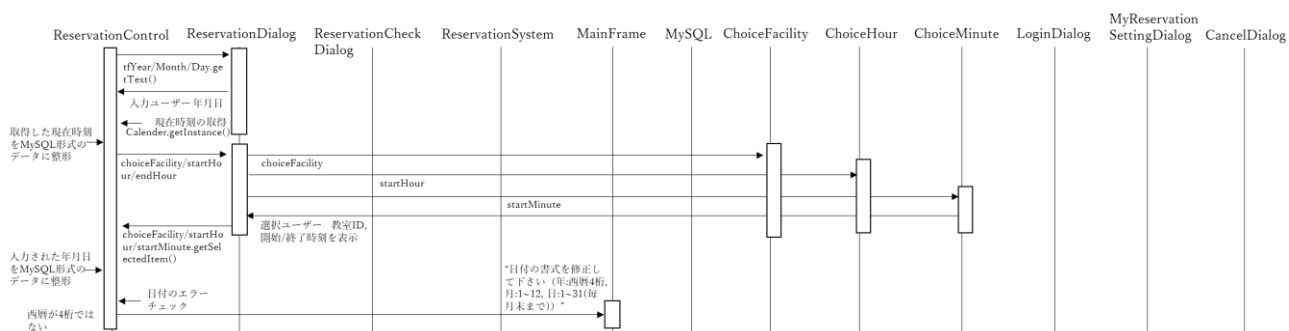
本シーケンスは、新規予約ダイアログ表示中にコンボボックスより教室選択,予約時間選択を行い、再度各コンボボックスの選択を変更した時のものである。(ex.4.4.1 新規予約ダイアログ表示からの継続)



4.4.5 予約情報入力～予約情報不適切

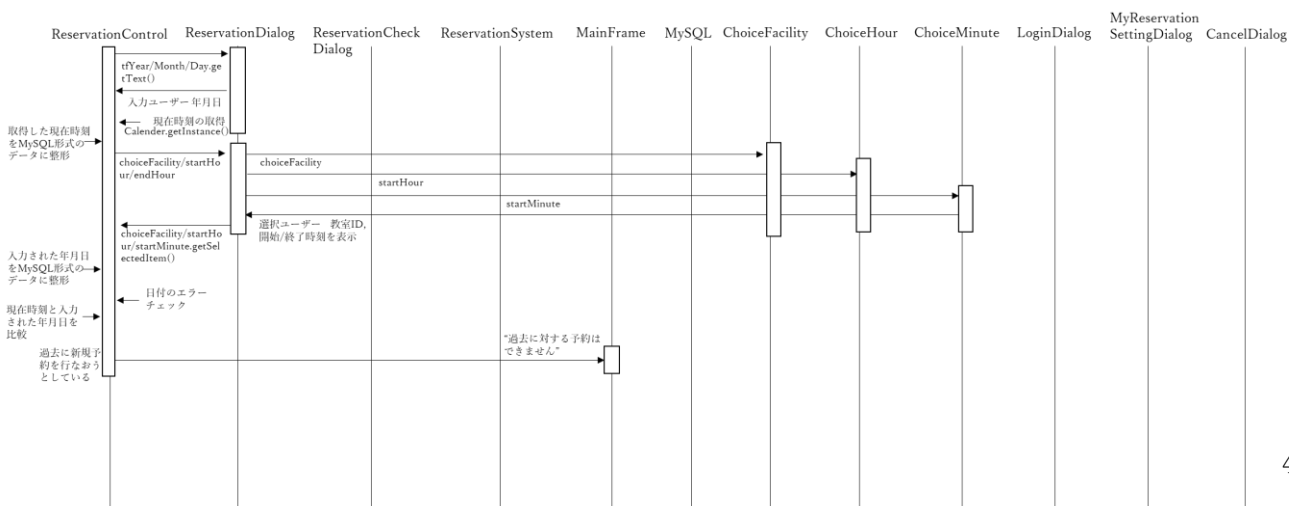
4.4.5.1 予約情報入力～予約西暦不適切

本シーケンスは、新規予約ダイアログ表示中に予約時間選択、及び予約日入力を行い、入力された予約日が西暦 4 桁以外時のものである。(ex.4.4.1 新規予約ダイアログ表示からの継続)



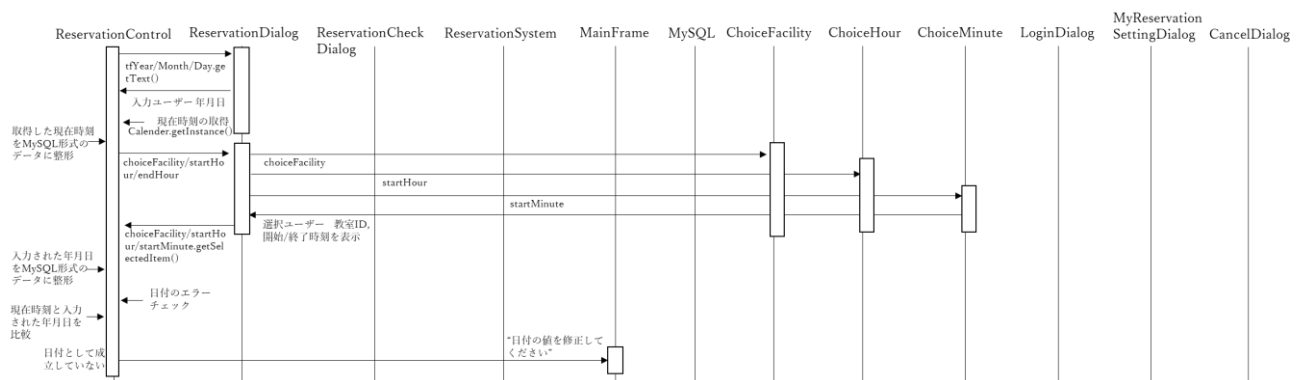
4.4.5.2 予約情報入力～予約日不適切

本シーケンスは、新規予約ダイアログ表示中に予約時間選択、及び予約日入力を行い、入力された予約日が過去だった時のものである。(ex.4.4.1 新規予約ダイアログ表示からの継続)



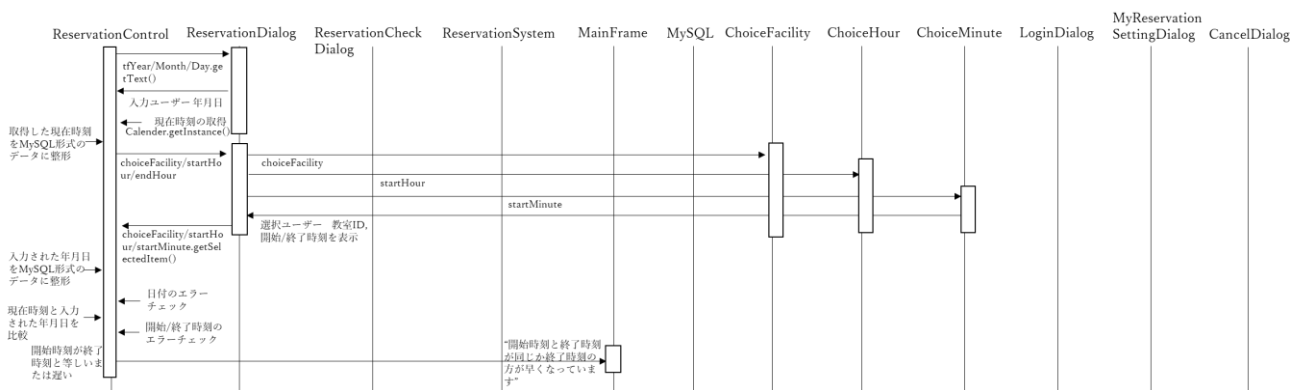
4.4.5.3 予約情報入力～予約日付不適切

本シーケンスは、新規予約ダイアログ表示中に予約時間選択、及び予約日入力を行い、入力された日付が存在しない日付だった時のものである。(ex.4.4.1 新規予約ダイアログ表示からの継続)



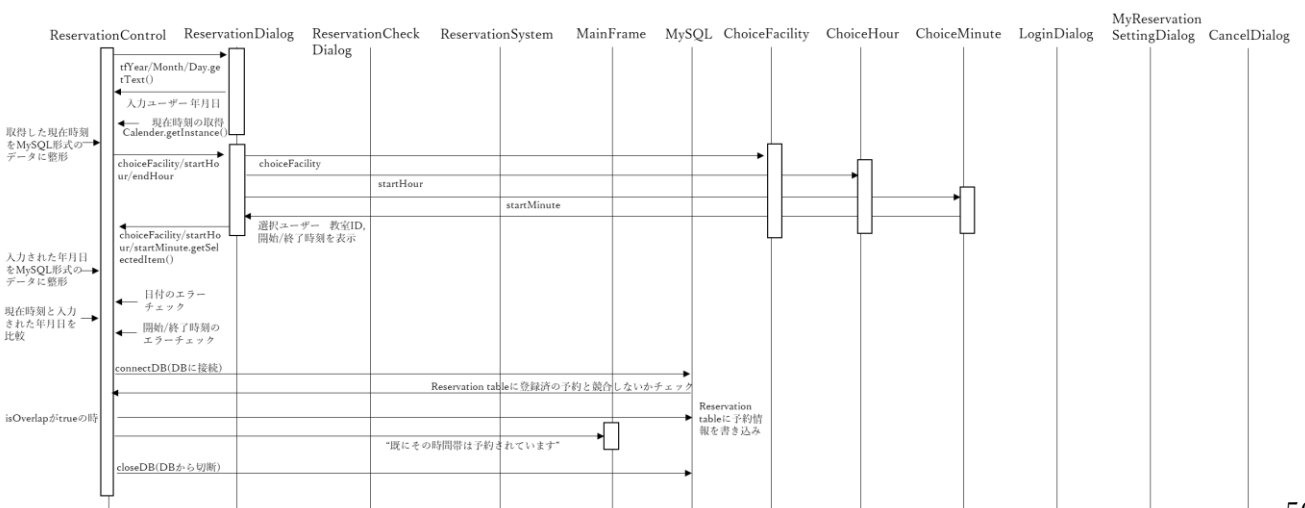
4.4.5.4 予約情報入力～予約時間不適切

本シーケンスは、新規予約ダイアログ表示中に予約時間選択、及び予約日入力を行い、選択内容が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る時のものである。(ex.4.4.1 新規予約ダイアログ表示からの継続)



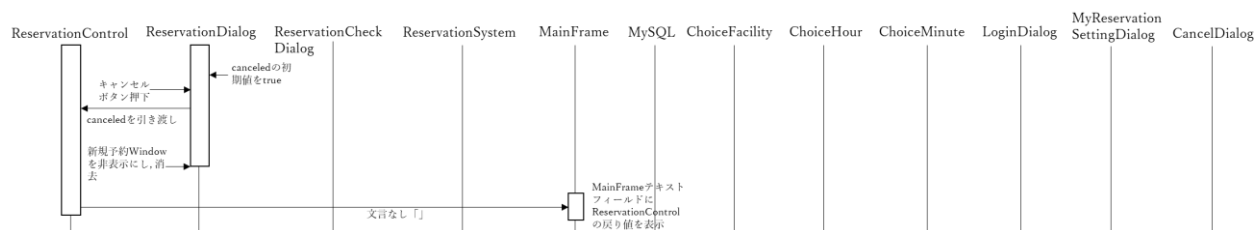
4.4.6 予約情報入力～登録失敗

本シーケンスは、新規予約ダイアログ表示中にコンボボックスより教室選択,予約時間選択を行い、予約日を入力し、入力に問題なく OK ボタンを押下したが,新規予約内容が Reservation テーブルに登録されている予約済みのものと競合してしまった時のものである。(ex.4.4.1 新規予約ダイアログ表示からの継続)



4.4.7 新規予約中止

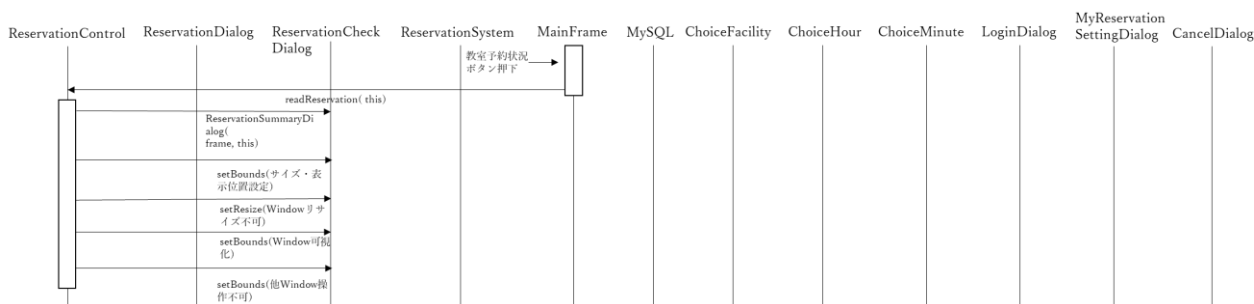
本シーケンスは、新規予約ダイアログ表示中にキャンセルボタンを押下した時のものである。
(ex.4.4.1 新規予約ダイアログ表示からの継続)



4.5 教室予約状況確認機能

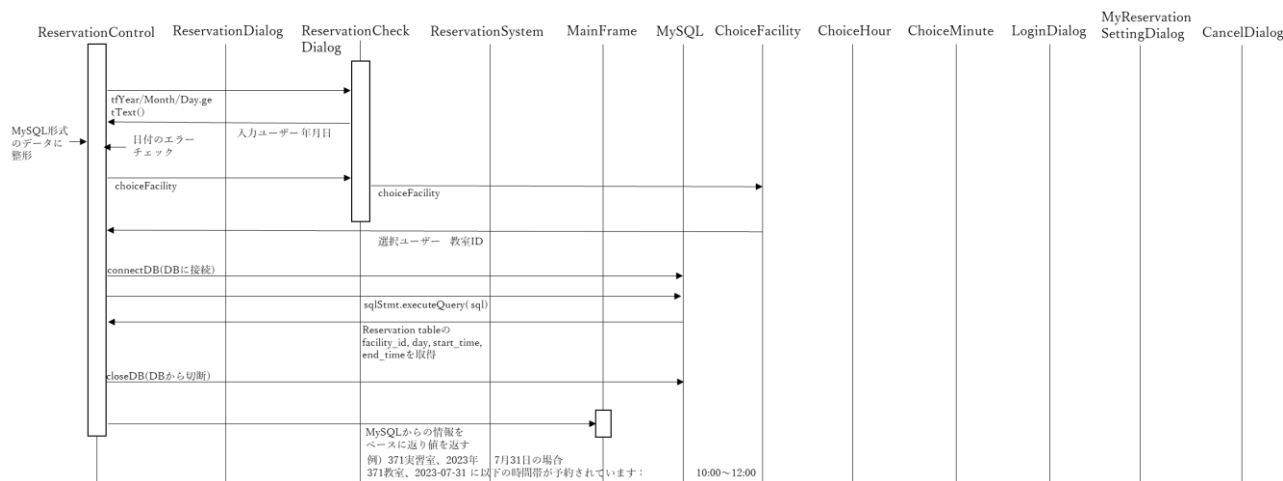
4.5.1 教室予約状況ダイアログ表示

本シーケンスは、MainFrameのみ表示中の状態で、教室予約状況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)



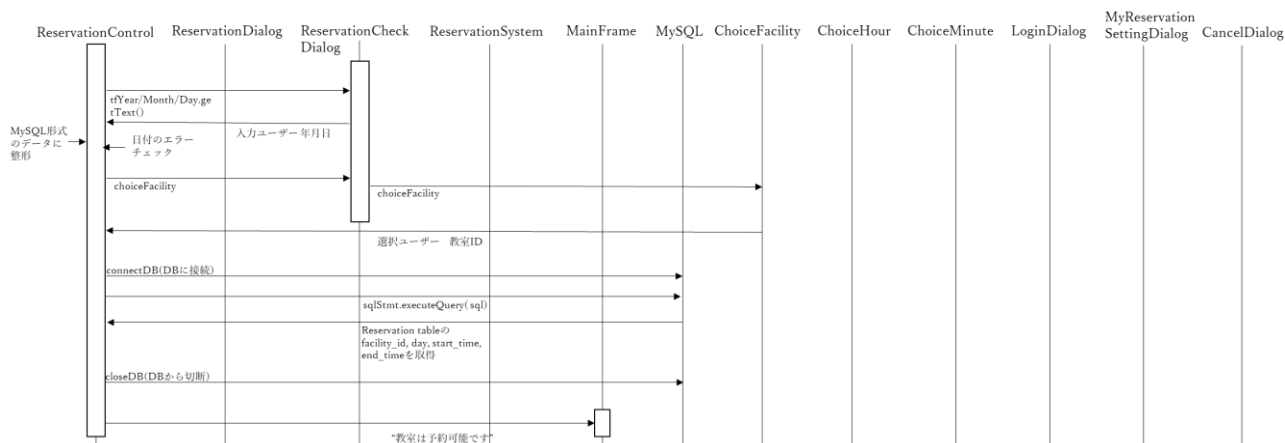
4.5.2 予約状況確認出力成功

本シーケンスは、教室予約状況ダイアログ表示中にコンボボックスより教室選択を行い、予約日を入力し、MySQLに該当データがあり、入力に問題なくOKボタンを押下した時のものである。
(ex.4.5.1 教室予約状況ダイアログ表示からの継続)



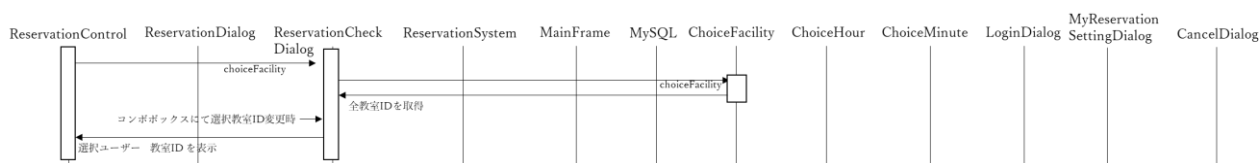
4.5.3 予約状況確認出力失敗

本シーケンスは、教室予約状況ダイアログ表示中にコンボボックスより教室選択を行い、予約日を入力し、MySQL に該当データがなく、入力に問題なく OK ボタンを押下した時のものである。
(ex.4.5.1 教室予約状況ダイアログ表示からの継続)



4.5.4 予約情報入力～教室 ID 変更

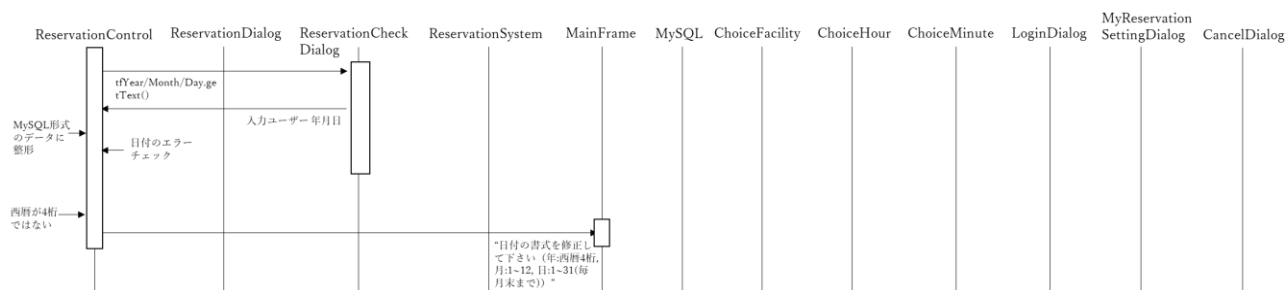
本シーケンスは、教室予約状況ダイアログ表示中に教室選択ボックスにて選択内容の変更を行った時のものである。(ex.4.5.1 教室予約状況ダイアログ表示からの継続)



4.5.5 予約情報入力～予約情報不適切

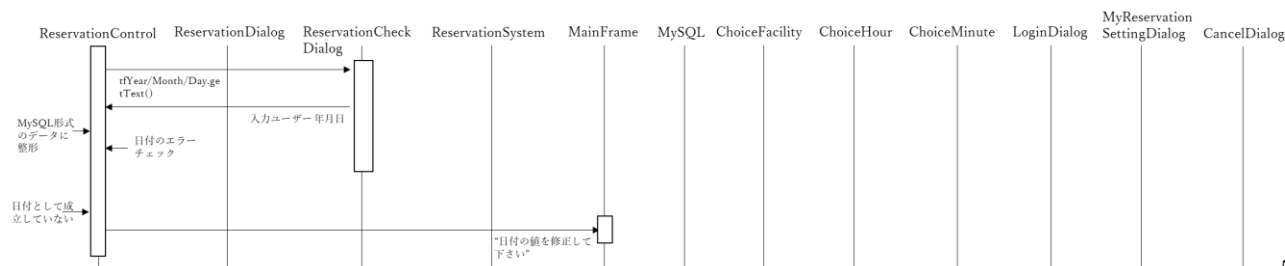
4.5.5.1 予約状況確認～西暦 4 桁以外

本シーケンスは、教室予約状況ダイアログ表示中に予約日入力を行い、入力された予約日が西暦 4 桁以外の時のものである。(ex.4.5.1 教室予約状況ダイアログ表示からの継続)



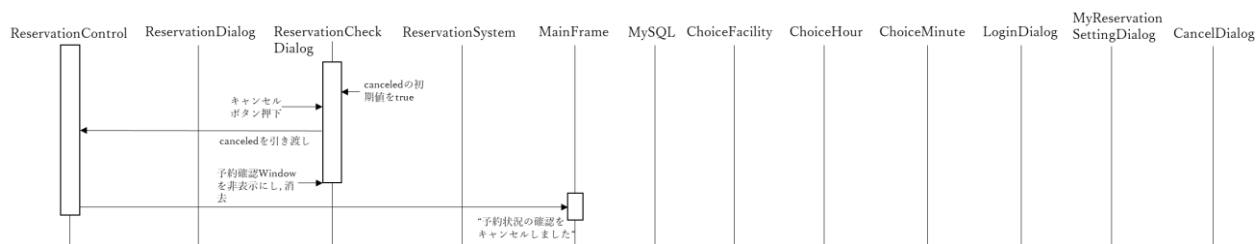
4.5.5.2 予約状況確認～日付不適切

本シーケンスは、教室予約状況ダイアログ表示中に予約日入力を行い、入力された予約日が存在しない日付だった時のものである。(ex.4.5.1 教室予約状況ダイアログ表示からの継続)



4.5.6 予約状況確認中止

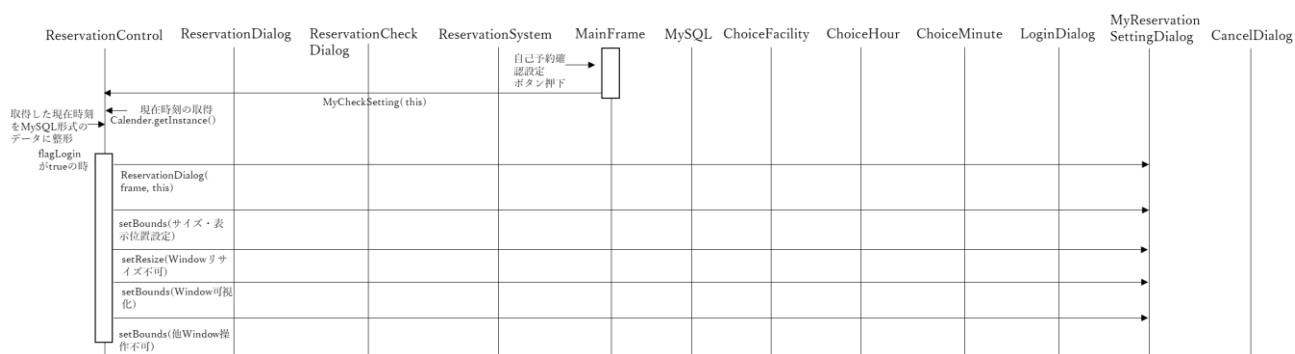
本シーケンスは、教室予約状況ダイアログ表示中にキャンセルボタンを押下した時のものである。
(ex.4.5.1 教室予約状況ダイアログ表示からの継続)



4.6 自己予約確認設定機能

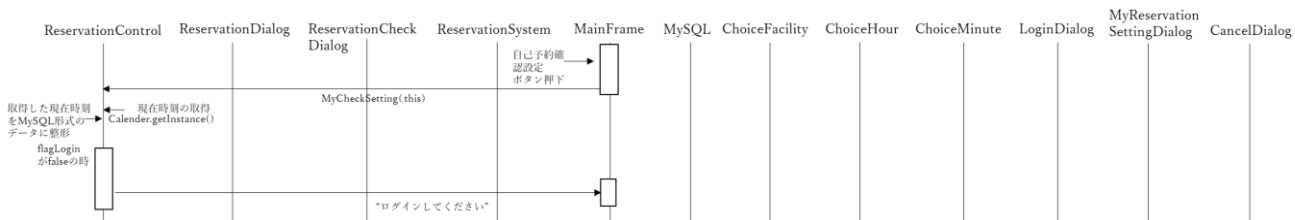
4.6.1 自己予約確認 Option 設定ダイアログ表示

本シーケンスは、MainFrame のみ表示中の状態で、教室予約状況ボタンを押下した時のものである。
(ex.4.2.2 ID,パスワード入力～認証完了からの継続)



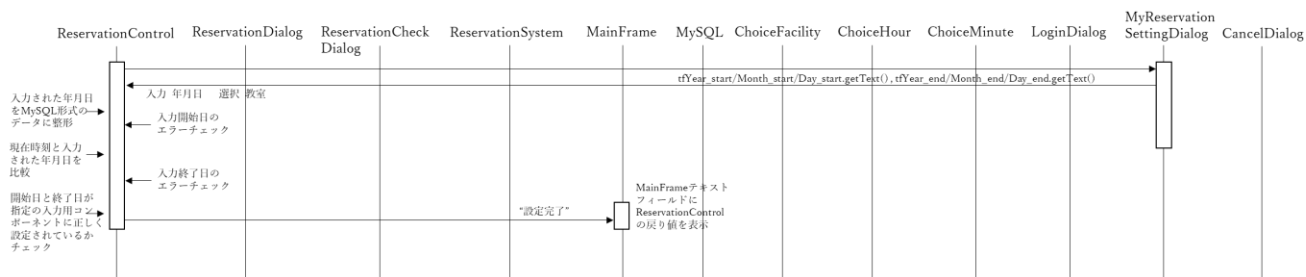
4.6.2 自己予約確認 Option 設定ダイアログ表示不可

本シーケンスは、MainFrame のみ表示中の状態で、教室予約状況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)



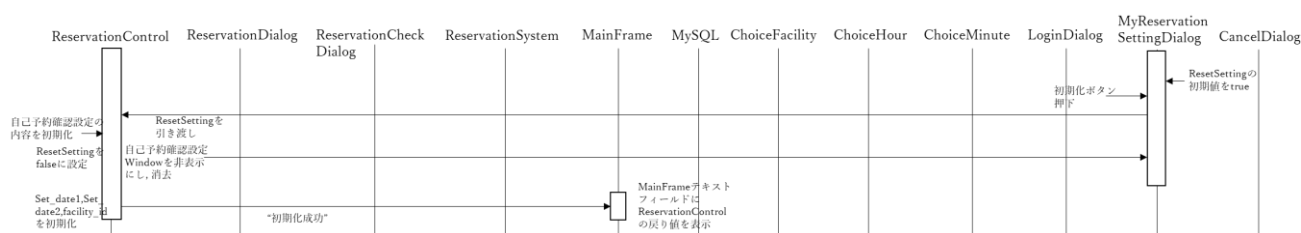
4.6.3 設定情報入力～設定完了

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中にコンボボックスより教室選択を行い,指定期間の開始日,終了日を入力し, 入力に問題なく設定完了ボタンを押下した時のものである。
(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



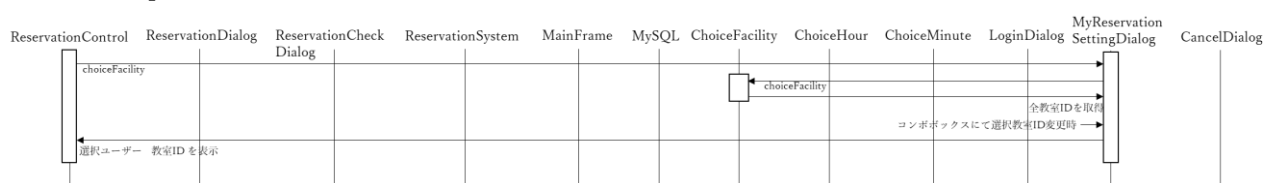
4.6.4 設定情報初期化

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中に初期化ボタンを押下した時のものである。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



4.6.5 設定情報入力～教室 ID 変更

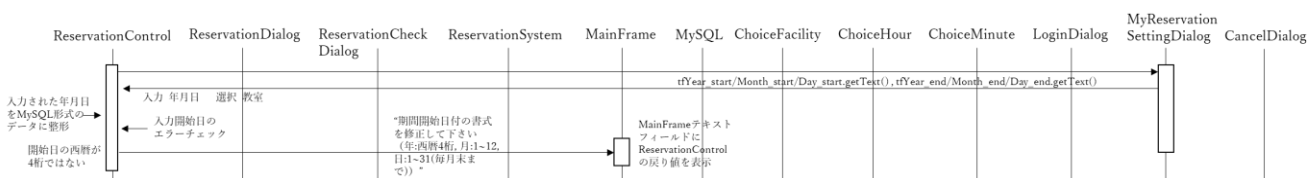
本シーケンスは、自己予約確認 Option 設定ダイアログ表示中にコンボボックスより教室選択にて、選択教室を変更した際の自己予約確認 Option 設定ダイアログ上の変化である。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



4.6.6 設定情報入力～設定情報不適切

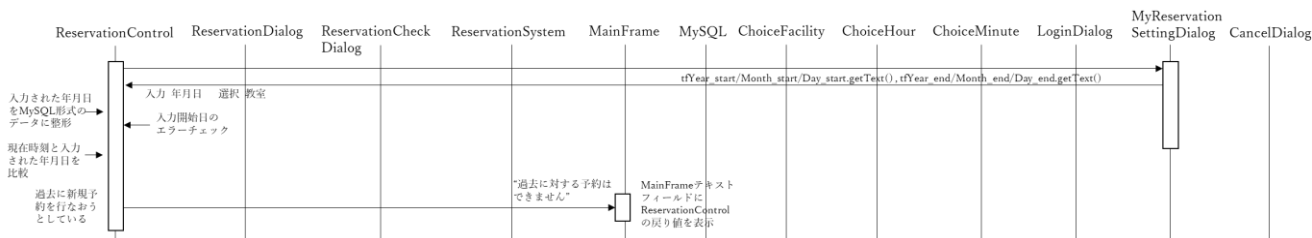
4.6.6.1 設定情報入力～設定開始日西暦不適切

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中に設定を行い、設定完了ボタンを押下した際、入力された指定期間の開始日が西暦 4 桁以外の時のものである。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



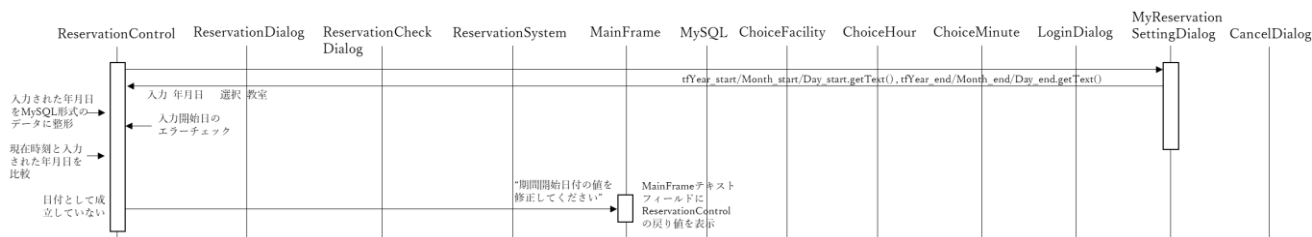
4.6.6.2 設定情報入力～設定開始日不適切

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中に設定を行い、設定完了ボタンを押下した際、入力された指定期間の開始日が設定当日以前に設定された時のものである。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



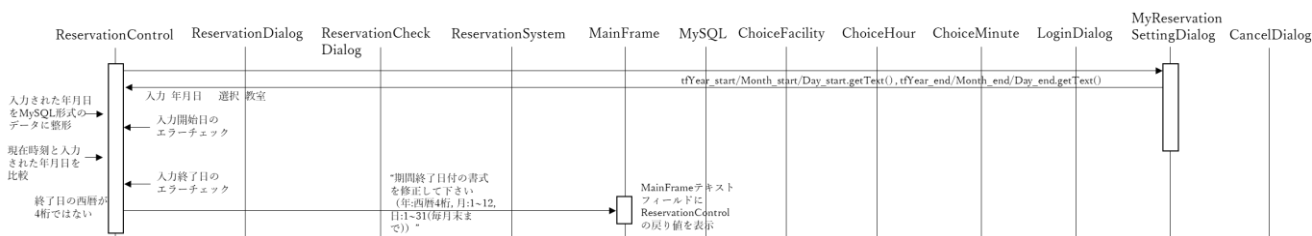
4.6.6.3 設定情報入力～設定開始日日付不適切

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中に設定を行い,設定完了ボタンを押下した際,入力された指定期間の開始日が存在しない日付だった時のものである。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



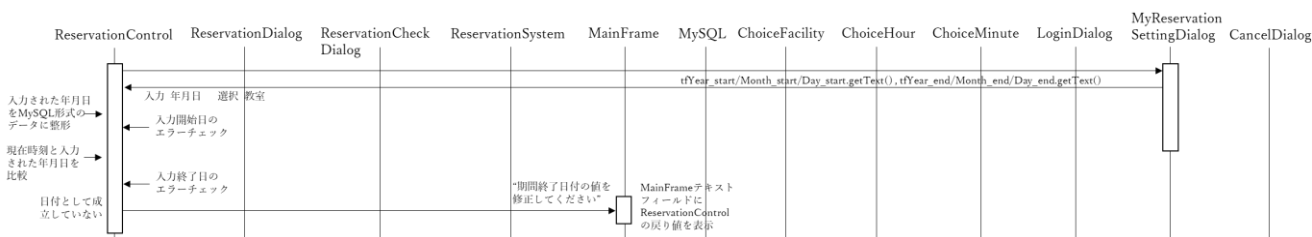
4.6.6.4 設定情報入力～設定終了日西暦不適切

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中に設定を行い,設定完了ボタンを押下した際,入力された指定期間の終了日が西暦 4 桁以外の時のものである。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



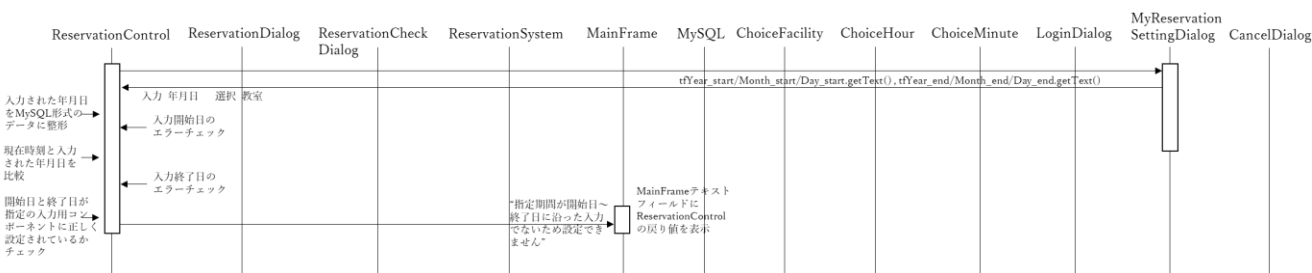
4.6.6.5 設定情報入力～設定終了日日付不適切

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中に設定を行い,設定完了ボタンを押下した際,入力された指定期間の終了日が存在しない日付だった時のものである。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



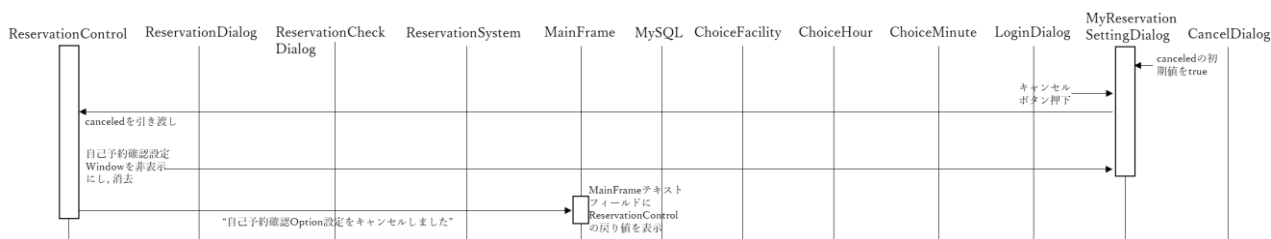
4.6.6.6 設定情報入力～設定開始日と終了日が逆に入力

本シーケンスは、自己予約確認 Option 設定ダイアログ表示中に設定を行い,設定完了ボタンを押下した際,入力された指定期間の開始日,終了日が指定された入力用コンポーネントに入力されていない時のものである。(ex.4.6.1 自己予約確認 Option 設定ダイアログ表示からの継続)



4.6.7 自己予約確認設定中止

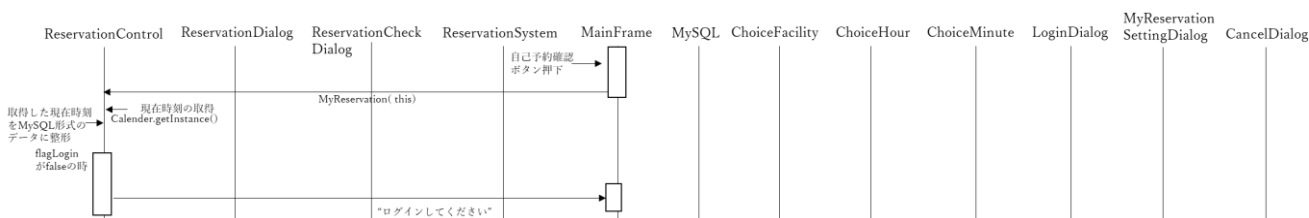
本シーケンスは、自己予約確認 Option 設定ダイアログ表示中にキャンセルボタンを押下した時のものである。(ex.4.5.1 教室予約状況ダイアログ表示からの継続)



4.7 自己予約確認機能

4.7.1 未ログイン状態の自己予約確認

本シーケンスは、MainFrameのみ表示中の状態で、自己予約確認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)

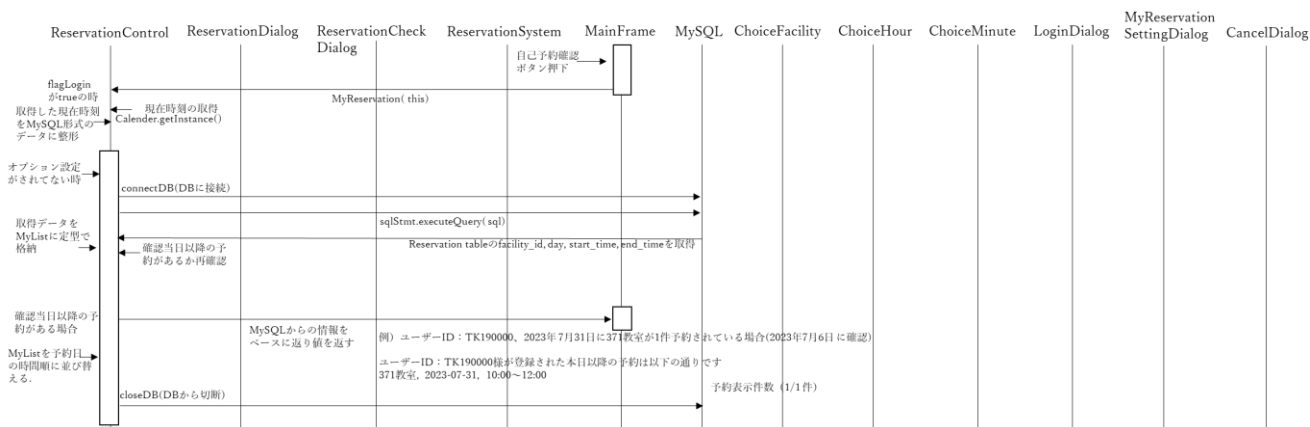


4.7.2 ログイン状態の自己予約確認

4.7.2.1 ミニマム仕様（初期状態）

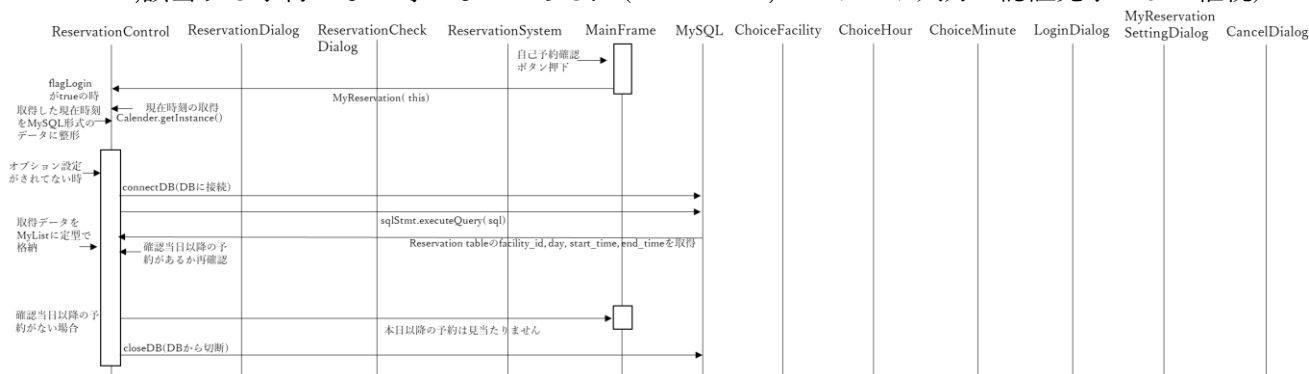
4.7.2.1.1 予約あり

本シーケンスは、MainFrameのみ表示中の状態で、ミニマム仕様の際、自己予約確認ボタンを押下し、該当する予約がある時のものである。(ex.4.2.2 ID,パスワード入力～認証完了からの継続)



4.7.2.1.2 予約なし

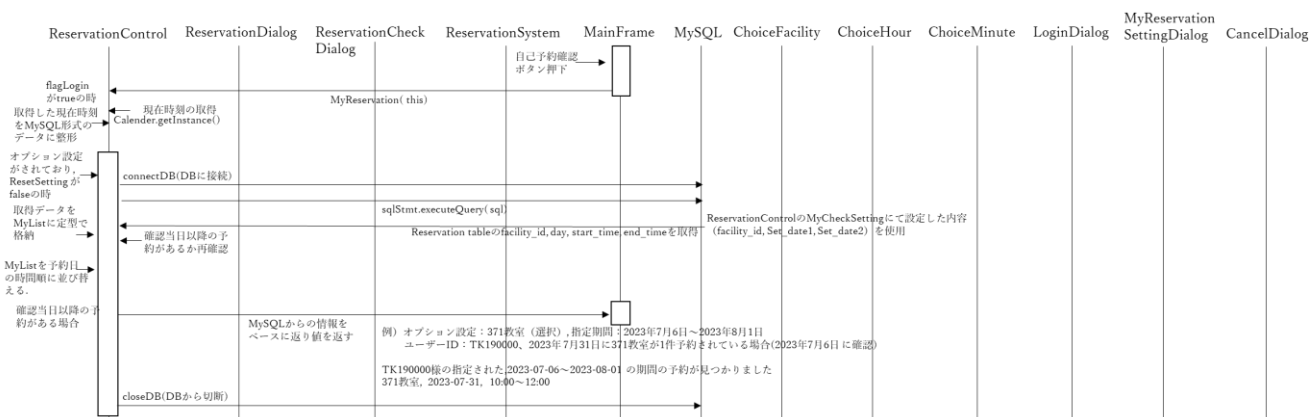
本シーケンスは、MainFrame のみ表示中の状態で、ミニマム仕様の際、自己予約確認ボタンを押下し、該当する予約がない時のものである。（ex.4.2.2 ID,パスワード入力～認証完了からの継続）



4.7.2.2 オプション仕様（自己予約確認機能設定後）

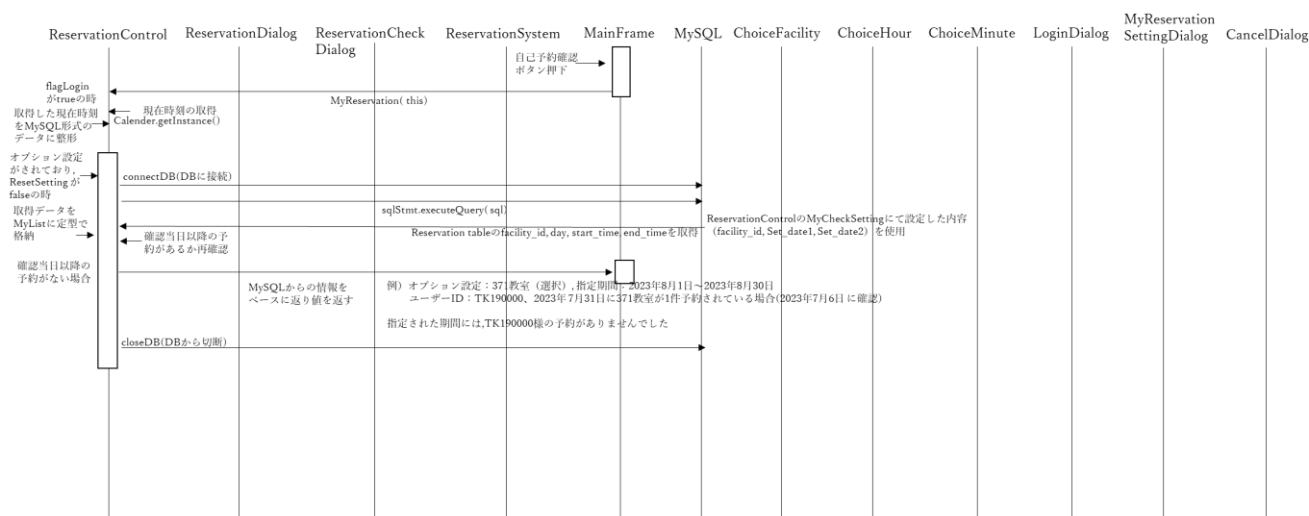
4.7.2.2.1 予約あり

本シーケンスは、MainFrame のみ表示中の状態で、オプション仕様の際、自己予約確認ボタンを押下し、該当する予約がある時のものである。（ex. 4.6.3 設定情報入力～設定完了からの継続）



4.7.2.2.2 予約なし

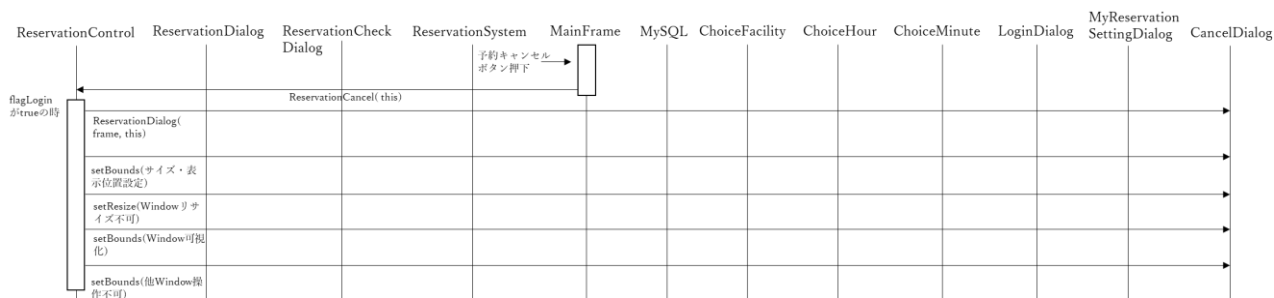
本シーケンスは、MainFrame のみ表示中の状態で、オプション仕様の際、自己予約確認ボタンを押下し、該当する予約がない時のものである。（ex. 4.6.3 設定情報入力～設定完了からの継続）



4.8 予約キャンセル機能

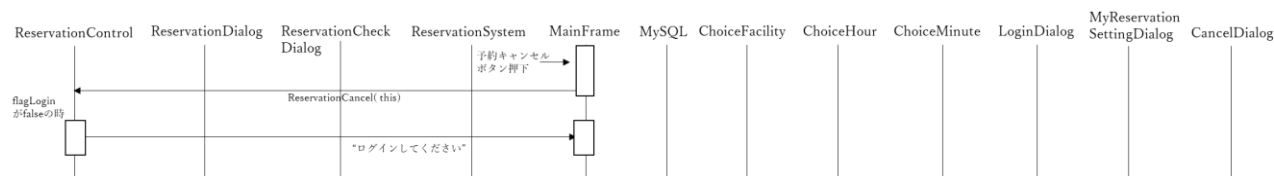
4.8.1 予約キャンセルダイアログ表示

本シーケンスは、MainFrame のみ表示中の状態で、予約キャンセルボタンを押下した時のものである。(ex.4.2.2 ID,パスワード入力～認証完了からの継続)



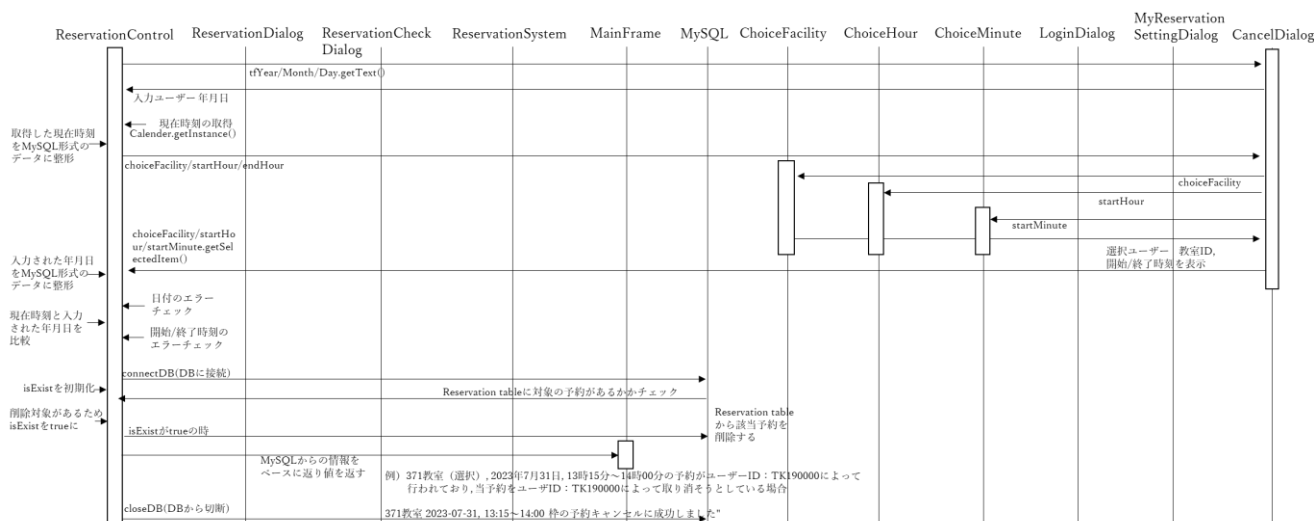
4.8.2 予約キャンセルダイアログ表示不可

本シーケンスは、MainFrame のみ表示中の状態で、予約キャンセルボタンを押下した時のものである。(ex.4.1 起動後からの継続)



4.8.3 予約取り消し情報入力～登録削除完了

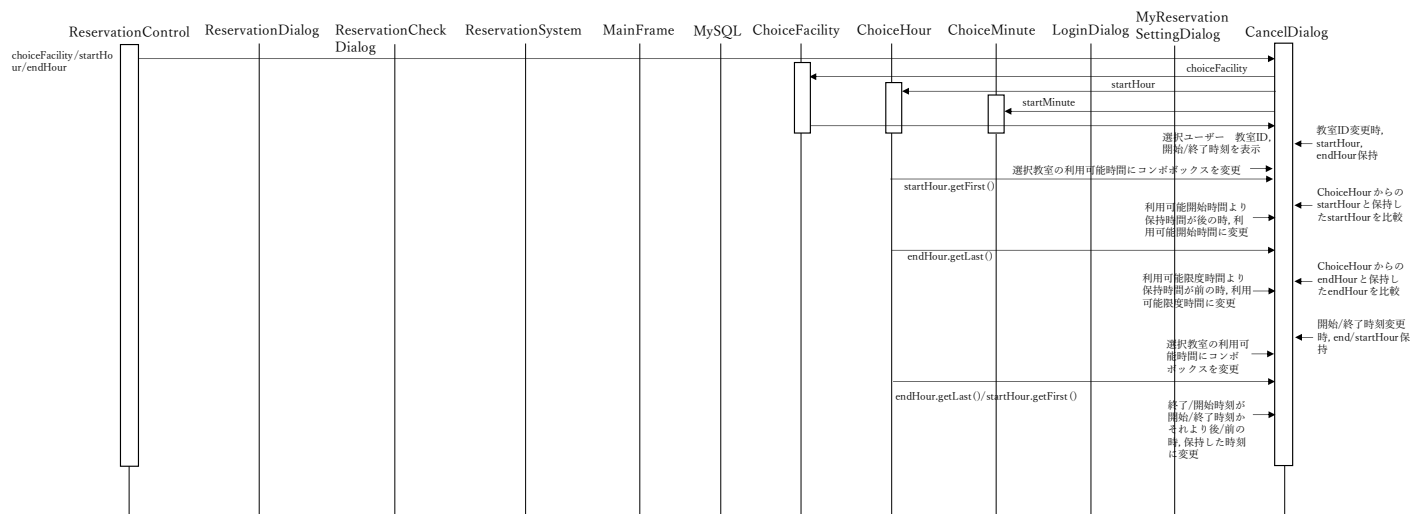
本シーケンスは、予約キャンセルダイアログ表示中にコンボボックスより教室選択,キャンセル対象の予約時間選択を行い,キャンセル対象の予約日を入力し, 入力に問題なく予約取り消しボタンを押下した時のものである。(ex. 4.8.1 予約キャンセルダイアログ表示からの継続)



4.8.4 予約取り消し情報入力～教室 ID, 予約時間変更

本シーケンスは、予約キャンセルダイアログ表示中にコンボボックスより教室選択,キャンセル対象の予約時間選択を行い,再度各コンボボックスの選択を変更した時のものである。

(ex. 4.8.1 予約キャンセルダイアログ表示からの継続)

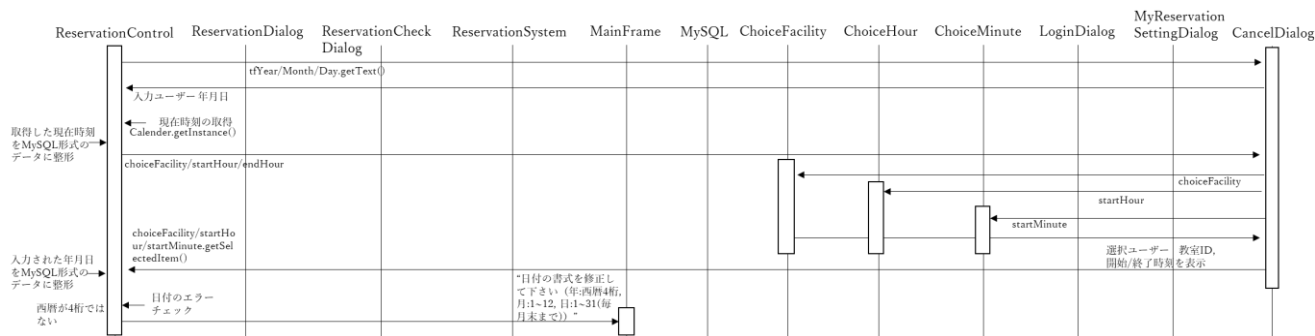


4.8.5 予約取り消し情報入力～予約取り消し情報不適切

4.8.5.1 予約取り消し情報入力～西暦不適切

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対象の予約時間選択、及び予約日入力を行い、入力された 予約日が西暦 4 桁以外時のものである。

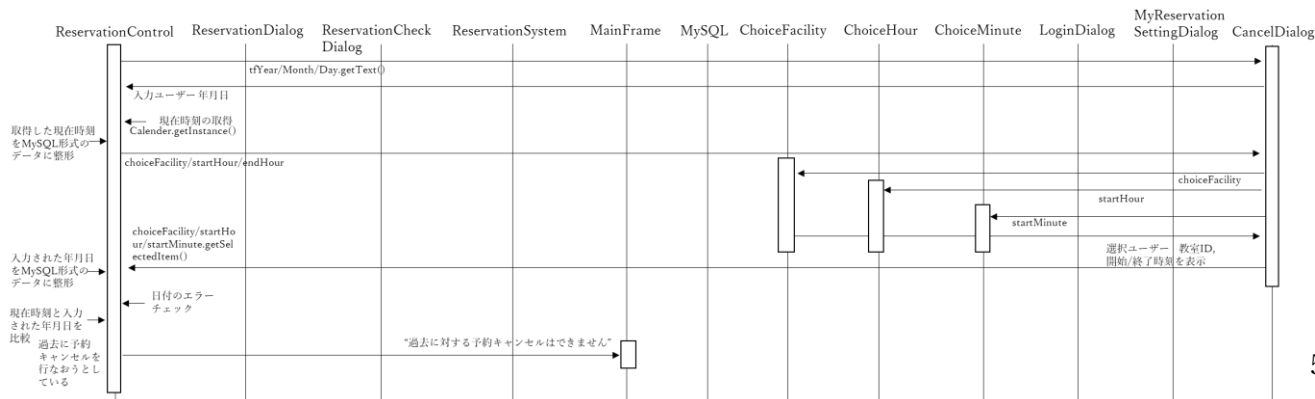
(ex. 4.8.1 予約キャンセルダイアログ表示からの継続)



4.8.5.2 予約取り消し情報入力～予約日不適切

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対象の予約時間選択、及び予約日入力を行い、入力された日付が過去だった時のものである。

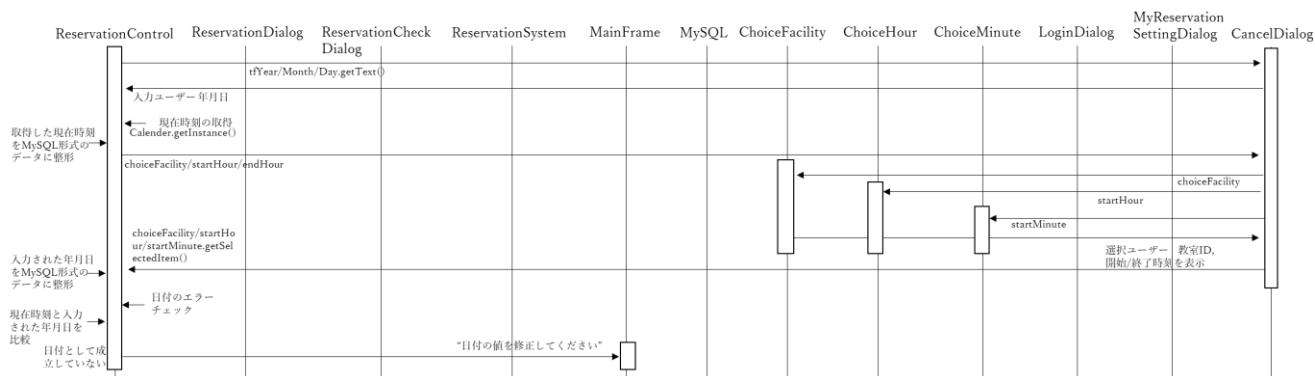
(ex.4.8.1 予約キャンセルダイアログ表示からの継続)



4.8.5.3 予約取り消し情報入力～予約日付不適切

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対象の予約時間選択、及び予約日入力を行い、入力された日付が存在しない日付だった時のものである。

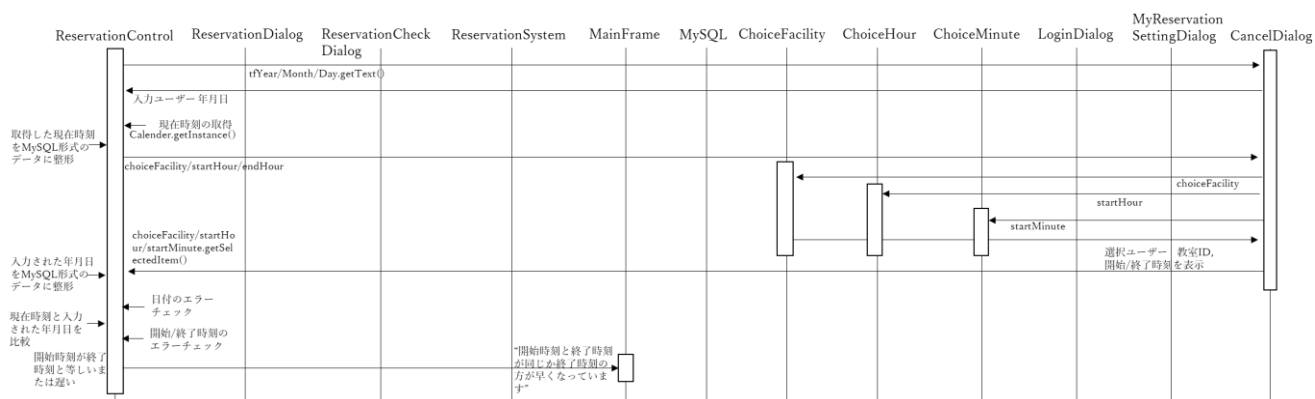
(ex.4.8.1 予約キャンセルダイアログ表示からの継続)



4.8.5.4 予約取り消し情報入力～予約時間不適切

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対象の予約時間選択、及び予約日入力を行い、選択内容が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る時のものである。

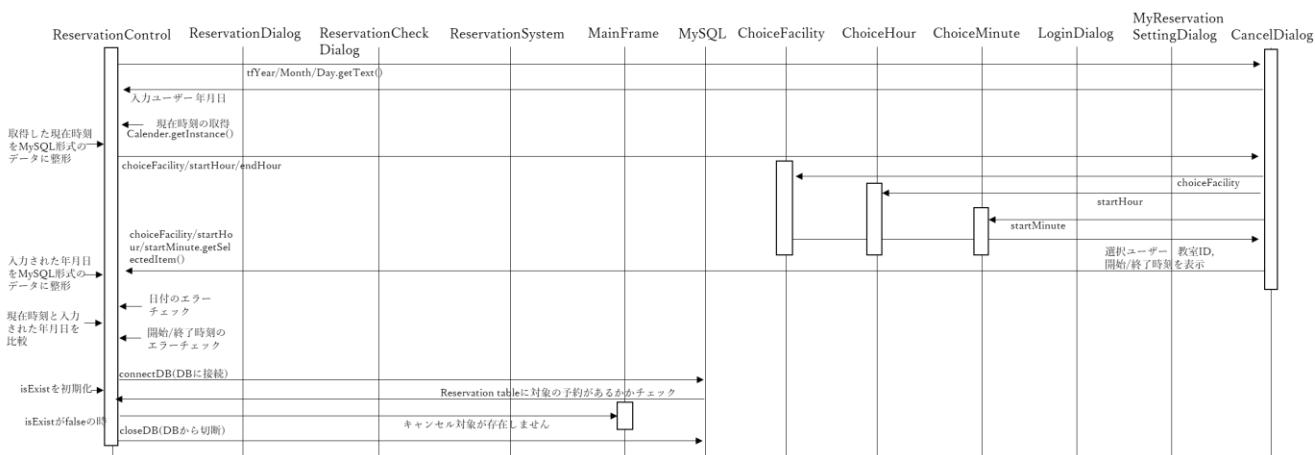
(ex.4.8.1 予約キャンセルダイアログ表示からの継続)



4.8.6 予約取り消し情報入力～登録削除失敗

本シーケンスは、予約キャンセルダイアログ表示中にコンボボックスより教室選択,キャンセル対象の予約時間選択を行い,キャンセル対象の予約日を入力し, 入力に問題なく予約取り消しボタンを押下したが,予約キャンセル内容と一致する予約が Reservation テーブルに登録されてない時のものである。

(ex.4.8.1 予約キャンセルダイアログ表示からの継続)



4.8.7 予約取り消し中止

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。(ex.4.8.1 予約キャンセルダイアログ表示からの継続)

