

ホテルマネジメントシステム 内部設計書

Workers

2020 年 12 月 21 日

目次

1	システム概要	4
1.1	全ユーザ	4
1.2	管理者	4
1.3	従業員・アルバイト	4
2	動作環境	5
3	開発環境	6
4	コーディング規約	6
4.1	コーディングスタイル	6
4.2	命名規約	7
5	モジュール設計書	7
5.1	ログインモジュール	8
5.2	総合 TOP モジュール	9
5.3	フロント担当者モジュール	10
5.4	予約情報入力モジュール	11
5.5	予約検索モジュール	12
5.6	予約情報詳細モジュール	13
5.7	予約情報編集モジュール	14
5.8	追加料金編集モジュール	15
5.9	食事情報閲覧モジュール	16
5.10	清掃情報管理モジュール	17
5.11	マニュアル・質問閲覧 TOP モジュール	18
5.12	管理者専用 TOP モジュール	19
5.13	ユーザ情報管理モジュール	20
5.14	システムログ参照モジュール	21
5.15	よくある質問情報管理モジュール	22
5.16	マニュアル情報管理モジュール	23
6	モジュール詳細	24
6.1	j_general_top.html	24
6.2	j_general_top.php	24
6.3	j_login.html	24
6.4	j_login.php	24
6.5	f_top.html	25
6.6	f_reservation_input.html	25

6.7	f_reservation_confirmation.html	25
6.8	f_reservation_done.html	25
6.9	f_search.html	25
6.10	f_search_result.html	26
6.11	f_information_details.html	26
6.12	f_information_edit.html	26
6.13	f_information_confirmation.html	26
6.14	f_information_done.html	26
6.15	f_addfee_edit.html	26
6.16	f_addfee_confirmation.html	27
6.17	f_addfee_done.html	27
6.18	f_customre.php	27
6.19	r_information_meal.html	28
6.20	r_information_meal.php	28
6.21	r_fee.php	28
6.22	s_clean_management.html	29
6.23	s_clean_edit.html	29
6.24	s_clean_management.php	29
6.25	j_manualquestion_top.html	30
6.26	j_manual_selection.php	30
6.27	j_question_selection.php	30
6.28	j_manual_selection.html	31
6.29	j_manual_reading.html	31
6.30	k_top.html	31
6.31	k_user_management.html	31
6.32	k_user_input.html	31
6.33	k_user_edit.html	31
6.34	k_user_confirmation.html	32
6.35	k_user_done.html	32
6.36	k_user_manegement.php	32
6.37	k_log_list.html	33
6.38	k_log_list.php	33
6.39	k_log_record.php	33
6.40	k_question_management.html	33
6.41	k_question_input.html	34
6.42	k_question_edit.html	34
6.43	k_question_confirmation.html	34
6.44	k_question_done.html	34
6.45	k_question_management.php	34
6.46	k_manual_management.html	35

6.47	k_manual_input.html	35
6.48	k_manual_edit.html	35
6.49	k_manual_confirmation.html	36
6.50	k_manual_done.html	36
6.51	k_manual_management.php	36
7	データベース設計	37
7.1	ユーザ情報テーブル	37
7.2	ログ情報テーブル	38
7.3	マニュアル情報テーブル	40
7.4	よくある質問情報テーブル	40
7.5	顧客情報テーブル	41
7.6	過去顧客情報テーブル	45
7.7	部屋情報テーブル	47
7.8	追加料金情報テーブル	47
付録 A	ER 図	50
付録 B	各メンバーの貢献内容	50

1 システム概要

本システムは、ホテル U の問題点を解決するための WEB システムであり、管理者、従業員、アルバイトに向けたシステムを実装する。従業員には、フロント担当、清掃担当、レストラン担当が存在し、それぞれに専用の機能を実装する。各システムを構成する機能を以下に示す。

1.1 全ユーザ

- ユーザ認証サブシステム
ログイン時に、入力されたユーザ名とパスワードから、ユーザの識別と認証を行う。
ユーザ認証が成功した場合、権限によって本システムの利用可能な機能を制限する。

1.2 管理者

- ユーザ情報管理サブシステム
ユーザ ID、ユーザ名、パスワード、ユーザ権限（複数可）によってユーザ情報の管理を行う。
ユーザ情報の管理では、ユーザ情報の閲覧、登録、編集、削除を行う。
- ログ参照サブシステム
本システムの操作のログを参照し、一覧を表示する。
- 質問情報管理サブシステム
質問 No、質問内容、回答内容、によって質問情報の管理を行う。
質問情報の管理では、質問情報の閲覧、登録、編集、削除を行う。
- マニュアル管理サブシステム
マニュアル No、マニュアル名、マニュアルの URL によってマニュアル情報の管理を行う。
マニュアル情報の管理では、マニュアル情報の閲覧、登録、編集、削除を行う。

1.3 従業員・アルバイト

- 質問情報確認サブシステム (従業員・アルバイト)
質問情報の一覧を表示する。
- マニュアル確認サブシステム (従業員・アルバイト)
マニュアル情報の一覧を表示する。
- 予約一覧表示サブシステム (フロント担当)
予約一覧画面の一覧を表示する。

- 予約情報検索サブシステム (フロント担当)
氏名または、電話番号によって予約情報の検索を行い、結果画面を表示する。
- 予約管理サブシステム (フロント担当)
予約 ID, 宿泊日, 予約日, 宿泊数, 氏名, 住所, 電話番号, 人数, プラン, 食事の有無, 食事メニュー, 部屋番号, チェックイン情報, 備考によって予約情報の管理を行う。
予約情報の管理では、予約情報の閲覧, 登録, 編集, 削除を行う。
- 清掃情報管理サブシステム (フロント担当, 清掃担当)
各部屋毎の清掃状況によって清掃情報の管理を行う。
清掃情報の管理では、清掃状況の編集を行う。
- 清掃情報一覧表示サブシステム (清掃担当)
清掃情報の一覧を表示する。
- 追加料金管理サブシステム (フロント担当, レストラン担当)
宿泊客の情報, 場所, 金額, 内容, 備考によって追加料金の管理を行う。
追加料金の管理では、追加料金の登録を行う。
- 食事情報一覧表示サブシステム (レストラン担当)
食事情報の一覧を表示する。

2 動作環境

本システムの動作環境を表 1, 表 2 に示す。

- ウェブアプリケーション動作環境

表 1 ウェブアプリケーション動作環境

対応 OS	Windows10, Android
推奨ブラウザ	firefox83.0
CPU	1GHz 以上
メモリ	2GB 以上

- サーバ動作環境

表 2 サーバ動作環境

OS	Amazon Linux2 AMI(Amazon Web Service EC2)
vCPU	1
メモリ	1GiB
ストレージ	30GiB

3 開発環境

本システムを開発する際に必要な環境を表 3 に示す。

表 3 開発環境

エディタ	Visual Studio Code, Emacs
OS	Windows10, macOS
開発言語	HTML, JavaScript, CSS, PHP, MySQL
データベース	MySQL
バージョン管理	Git Hub

4 コーディング規約

本システムを開発する上でのコーディング規約を下記に示す。

4.1 コーディングスタイル

- 文字コード
 - UTF-8 を使用する
- インデント
 - インデントには、タブを使用する。(半角スペース 4 文字)
- 括弧
 - 中括弧使用時には改行後に書き始める。
- コンマ
 - コンマの後ろには半角スペースを 1 文字入れる。
- 演算子
 - 使用する際には、前後に半角スペースを 1 文字入れる。

4.2 命名規約

- ファイル名

機能ごとに使用するファイル名の規約を表 4 に示す。

表 4 ファイル名規約

機能ファイル	ファイル名
フロント担当者専用機能	f_機能名_詳細. 拡張子
清掃担当者専用機能	s_機能名_詳細. 拡張子
レストラン担当者専用機能	r_機能名_詳細. 拡張子
従業員共通機能	j_機能名_詳細. 拡張子
管理者専用機能	k_機能名_詳細. 拡張子

- 変数名

- 全て小文字とする。
- 英単語の間にはアンダーバーを使用する。
- 複数の機能ファイルで変数名が重複する場合には、機能ごとにファイル名に使用した先頭の文字を変数名の先頭につけることで区別する。

- 定数名

- 全て大文字とする。
- 英単語の間にはアンダーバーを使用する。

4.2.1 HTML, CSS, JavaScript, PHP

- クラス名, メソッド名

- キャメルケース表記を使用する。(例:getId)

4.2.2 MySQL

- カラム名

- スネークケース表記を使用する。(例:get_id)

5 モジュール設計書

モジュール構成において、各図形の説明を図 1 に示す。

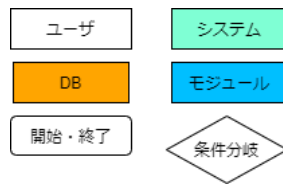


図 1 モジュール構成における各図形の説明

無色の図形は「ユーザの操作」、緑色の図形は「システムの動作」、橙色の図形は「データベース側の動作」、青色の図形は「モジュール」を意味している。角の丸い四角形は「各モジュールの開始、終了時の処理」、ひし形は「条件分岐」を意味している。図及び文章中の【 】はボタンを表している。

また、画面によっては、【戻る】ボタンが存在する。基本的には 1 つ前に表示した画面に遷移する。それ以外の遷移を行う場合は、別途記載を行う。

5.1 ログインモジュール

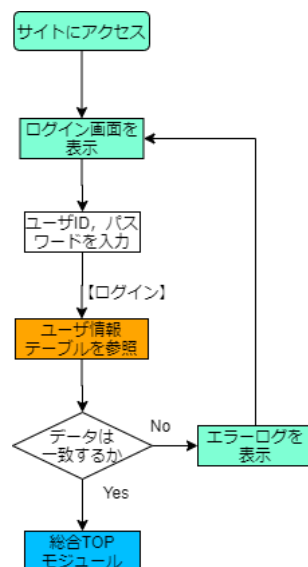


図 2 ログインモジュールのフローチャート

概要

図 2 は、本サイトにアクセスしてからログインし、総合 TOP モジュールまでの流れを示している。

処理

- 本サイトにアクセスした際に、ログイン画面を表示する。
- 入力されたユーザ名、パスワードを、データベースから参照しデータが一致していれば、総合 TOP モジュールを呼び出す。一致しない場合は、エラーログを表示し、ログイン画面に戻る。

5.2 総合 TOP モジュール

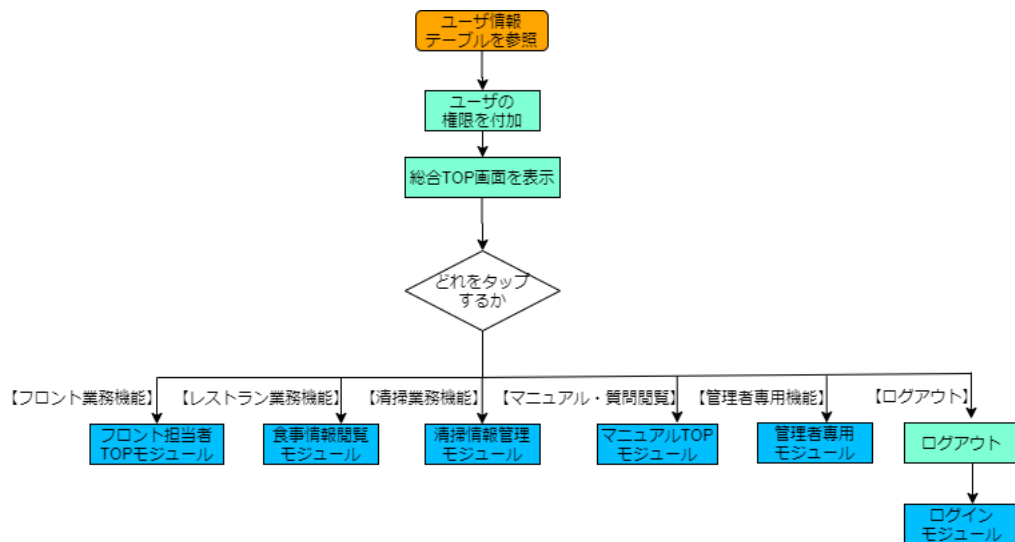


図3 総合 TOP モジュールのフローチャート

概要

図3は、総合 TOP モジュールから各モジュールへの切り替えまでの流れを示している。

処理

- 総合 TOP 画面を表示し、選択されたボタンに応じたモジュールを呼び出す。その際に、テーブルを参照し、権限を持っている場合のみ選択されたモジュールを呼び出す。権限を持っていない場合は、エラーログを表示して総合 TOP 画面に戻る。
- 【フロント業務機能】が押された場合、フロント担当者 TOP モジュールを呼び出す。
- 【レストラン業務機能】が押された場合、食事情報閲覧モジュールを呼び出す。
- 【清掃業務機能】が押された場合、清掃情報管理モジュールを呼び出す。
- 【マニュアル・質問閲覧】が押された場合、マニュアル TOP モジュールを呼び出す。
- 【管理者専用機能】が押された場合、管理者専用モジュールを呼び出す。
- 【ログアウト】が押された場合、アカウントからのログインを行い、ログインモジュールを呼び出す。

5.3 フロント担当者モジュール

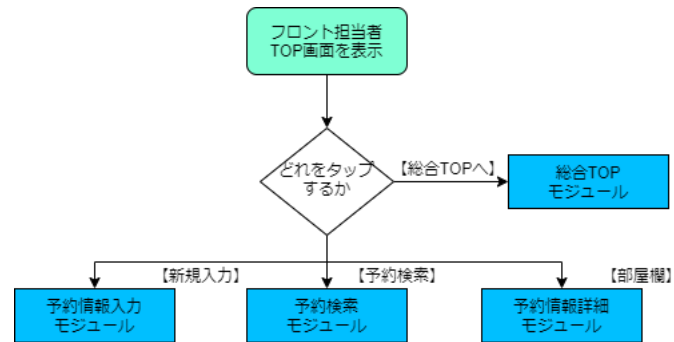


図4 フロント担当者モジュールのフローチャート

概要

図4は、フロント担当者 TOP 画面から各ボタンを押すことにより各モジュールを呼び出すことを示している。

処理

- 【新規入力】を押すと、予約情報入力モジュールを呼び出す。
- 【予約検索】を押すと、予約検索モジュールを呼び出す。
- 【部屋欄】を押すと、予約情報詳細モジュールを呼び出す。
- 【総合 TOP へ】を押すと、総合 TOP モジュールを呼び出す。

5.4 予約情報入力モジュール

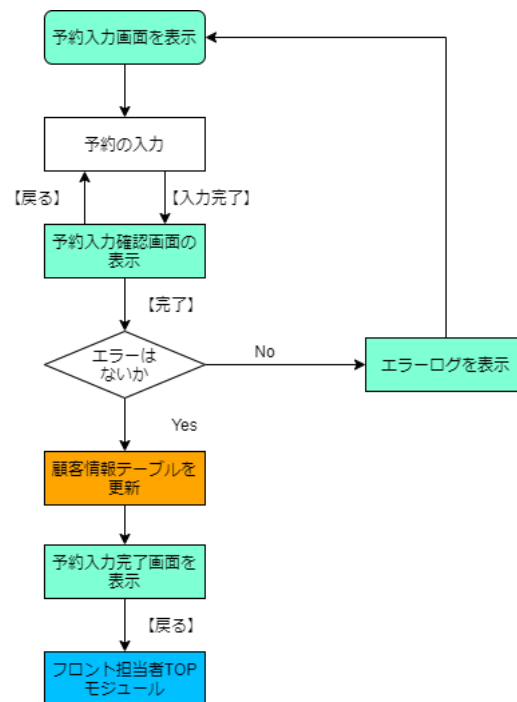


図5 予約情報入力モジュールのフローチャート

概要

図5は、予約入力画面からフロント担当者 TOP モジュールまでの流れを示している。

処理

- 予約入力画面で予約の入力を行う。
- 予約入力確認画面で予約の入力内容の確認を行い、【完了】を押す。
- エラーがなければ、顧客情報テーブルの更新を行い、予約入力完了画面を表示し、【戻る】を押すことでフロント担当者 TOP モジュールを呼び出す。エラーがある場合は、エラーログを表示し、予約入力画面に戻る。

5.5 予約検索モジュール

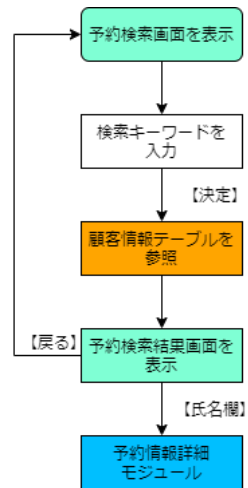


図 6 予約検索モジュールのフローチャート

概要

図 6 は、予約検索画面から予約情報詳細モジュールまでの流れを示している。

処理

- 予約検索画面で検索キーワードの入力を行う。
- データベースから検索キーワードと一致するものを検索し返す。
- 予約検索結果画面を表示し、【戻る】を押すことで予約検索画面を表示する。氏名欄を押すことで予約情報詳細モジュールを呼び出す。

5.6 予約情報詳細モジュール

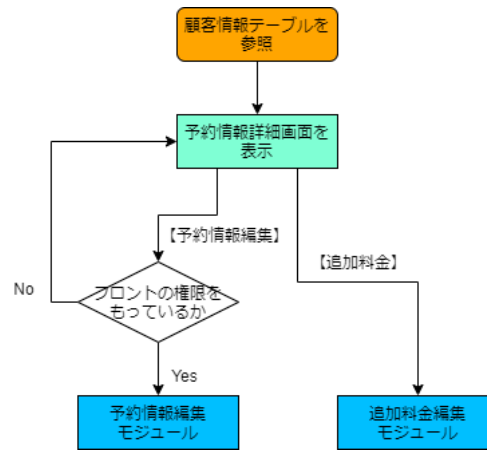


図7 予約情報詳細モジュールのフローチャート

概要

図7は、顧客情報テーブルの参照から予約情報詳細モジュール、追加料金編集モジュールまでの流れを示している。

処理

- 顧客情報テーブルを参照し、予約情報詳細画面を表示する。
- 【予約情報編集】を押すとフロントの権限を持っているかの確認を行い、持っていれば予約情報編集モジュールを呼び出す。権限を持っていない場合は、予約情報詳細画面へ戻る。
- 【追加料金】を押すと追加料金編集モジュールを呼び出す。

5.7 予約情報編集モジュール

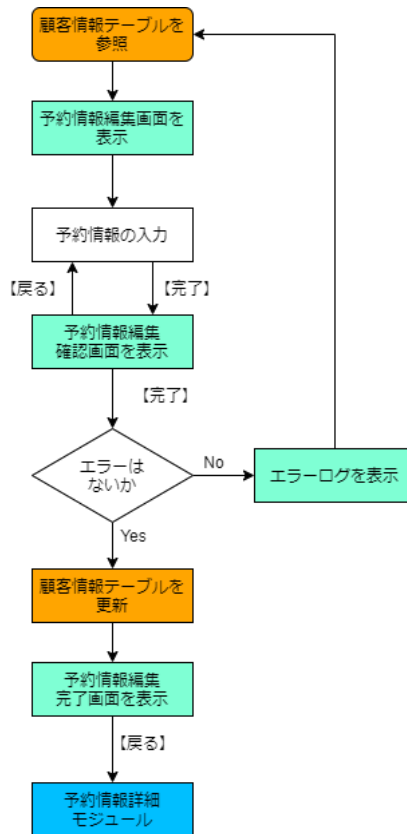


図 8 予約情報編集モジュールのフローチャート

概要

図 8 は、顧客情報テーブルの参照から予約情報詳細モジュールまでの流れを示している。

処理

- 顧客情報テーブルを参照し、予約情報編集画面を表示する。
- 予約情報の入力を行い【完了】を押すと、予約情報編集確認画面が表示される。
- エラーがなければ、入力された情報を顧客情報テーブルに更新し、予約情報編集完了画面を表示され、【戻る】を押すことで予約情報詳細モジュールが呼び出される。エラーがある場合は、顧客情報テーブル参照へ戻る。

5.8 追加料金編集モジュール

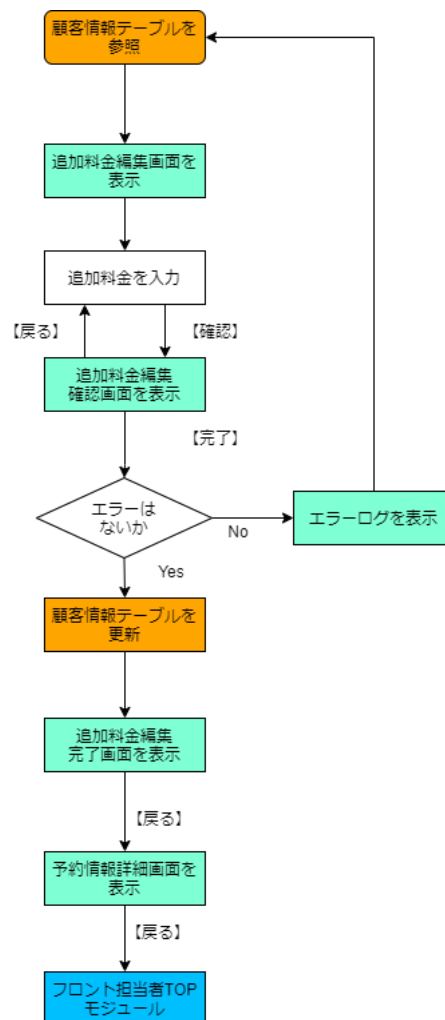


図9 追加料金編集モジュールのフローチャート

概要

図9は、顧客情報テーブルの参照からフロント担当者 TOP モジュールまでの流れを示している。

処理

- 顧客情報テーブルを参照し、追加料金画面を表示する。
- 追加料金の入力を行い、【確認】を押すことで追加料金編集確認画面が表示される。
- 【完了】を押すことで、エラーがなければ、顧客情報テーブルを更新し、追加料金編集完了画面が表示され、【戻る】を押すことで予約情報詳細画面が表示される。さらに、【戻る】を押すことでフロント担当者 TOP モジュールが呼び出される。エラーがある場合は、エラーログを表示し、顧客情報テーブル参照へ戻る。

5.9 食事情報閲覧モジュール

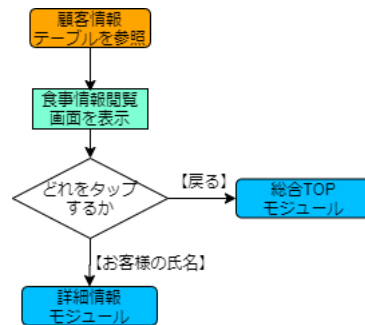


図 10 食事情報閲覧モジュールのフローチャート

概要

図 10 は、食事情報閲覧モジュールを呼び出し、詳細情報モジュール、総合 TOP モジュールまでの流れを示している。

処理

- 食事情報閲覧モジュールを呼び出した際に、顧客情報テーブルを参照し、食事情報を取得する。その後、食事情報閲覧画面を表示する。
- お客様の氏名が選択された場合、詳細情報モジュールを呼び出す。なお、図 10 では【お客様の氏名】と表示しているが、ボタンにはお客様の氏名を名称として画面に表示する。
- 【戻る】が選択された場合、総合 TOP モジュールを呼び出す。

5.10 清掃情報管理モジュール

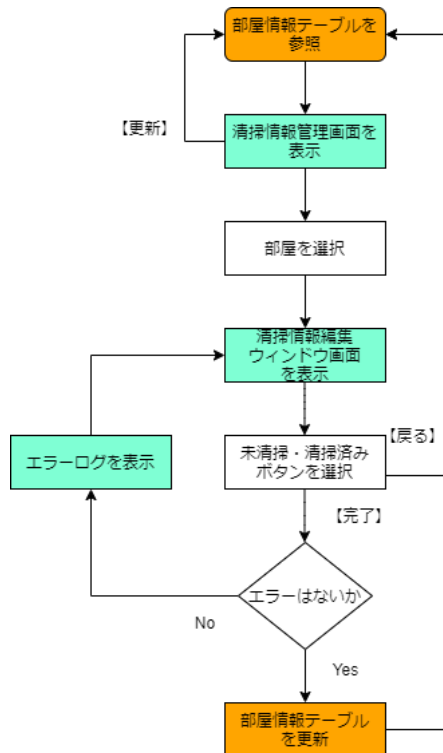


図 11 清掃情報管理モジュールのフローチャート

概要

図 11 は、清掃情報を管理する流れを示している。

処理

- 部屋情報テーブルを参照し清掃情報管理画面を表示する。
- 【更新】を押すと、部屋情報テーブルを参照し、清掃情報管理画面を表示する。
- 【部屋】を選択すると、清掃情報編集ウィンドウ画面を表示する。
- 清掃情報編集ウィンドウ画面から、【未清掃】または【清掃済み】を選択し、【完了】を押す。【戻る】を押すと、編集内容を破棄し部屋情報テーブルを参照へ戻る。
- 【完了】を押すと、エラーの確認を行い、エラーがなければ部屋情報テーブルを更新する。エラーがある場合は、エラーログを表示し、清掃情報編集ウィンドウ画面を表示する。

5.11 マニュアル・質問閲覧 TOP モジュール

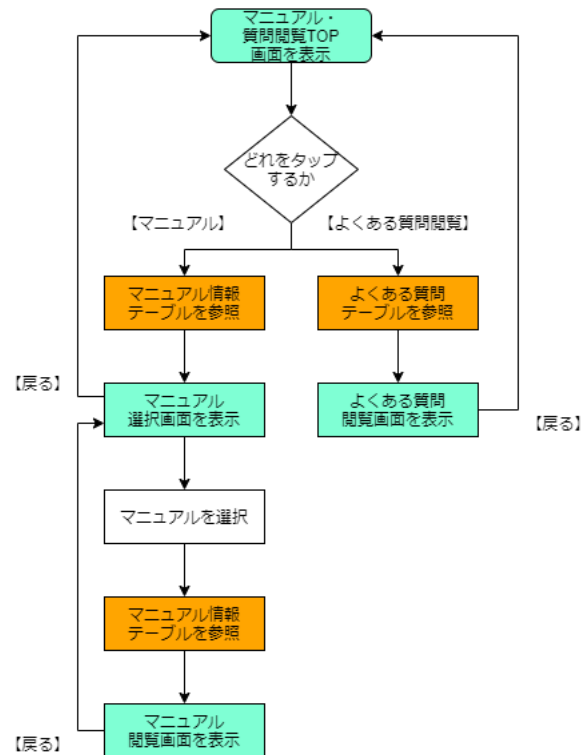


図 12 マニュアル・質問閲覧 TOP モジュールのフローチャート

概要

図 12 は、マニュアル・質問閲覧の流れを示している。

処理

- マニュアル・質問閲覧 TOP 画面から、【マニュアル】または【よくある質問閲覧】を押す。
- 【マニュアル】を押すと、マニュアル情報テーブルを参照しマニュアル選択画面を表示する。
- マニュアルを選択すると、マニュアル情報テーブルを参照しマニュアル閲覧画面を表示する。
- 【よくある質問閲覧】を押すと、よくある質問テーブルを参照しよくある質問閲覧画面を表示する。

5.12 管理者専用 TOP モジュール

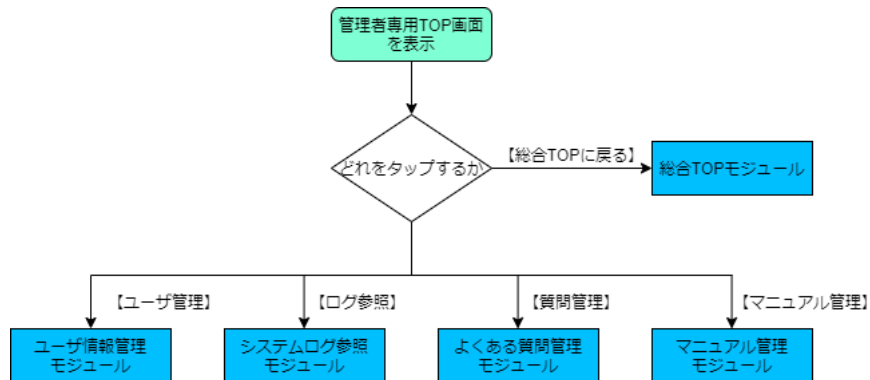


図 13 管理者専用 TOP のフローチャート

概要

図 13 は、管理者情報画面から各ボタンを押すことにより各モジュールを呼び出すことを示している。

処理

- 【ユーザ情報管理】を押すとユーザ情報管理モジュールを呼び出す。
- 【システムログ参照】を押すとシステムログ参照モジュールを呼び出す。
- 【質問情報管理】を押すとよくある質問管理モジュールを呼び出す。
- 【マニュアル情報管理】を押すとマニュアル情報管理モジュールを呼び出す。
- 【総合 TOP に戻る】を押すと総合 TOP モジュールを呼び出す。

5.13 ユーザ情報管理モジュール

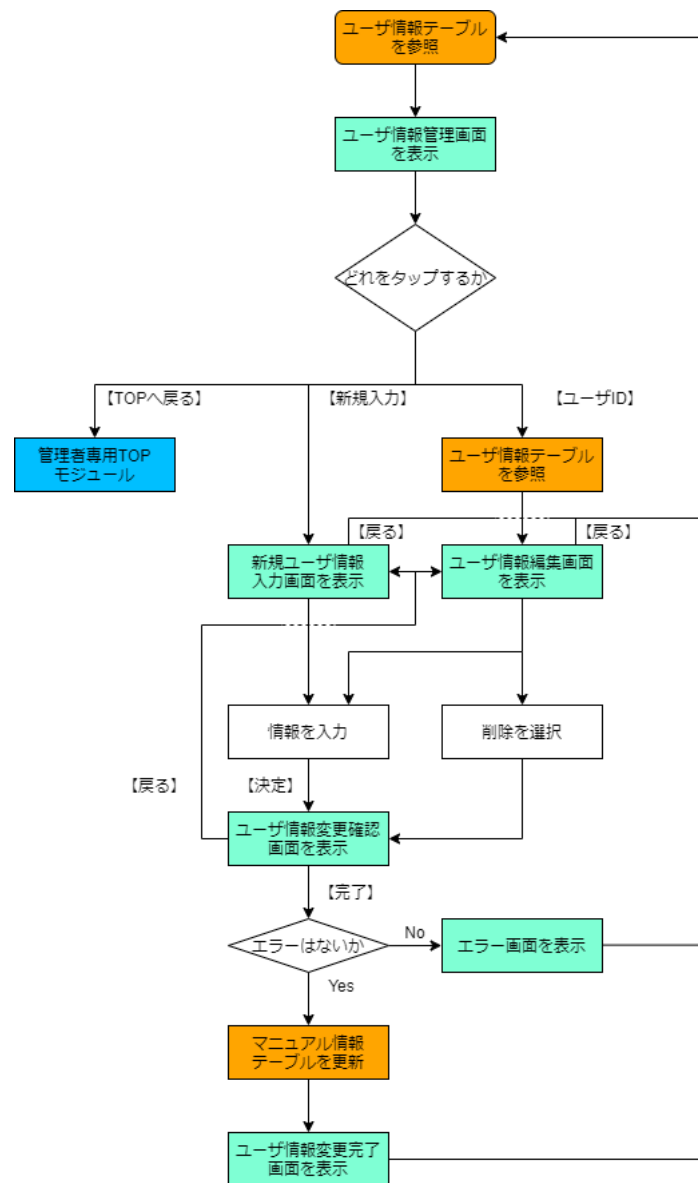


図 14 ユーザ情報管理モジュールのフローチャート

概要

図 14 は、ユーザ情報を管理する流れを示している。

処理

- ユーザ情報テーブルを参照しユーザ情報管理画面を表示する。
- 【新規入力】を選択すると新規ユーザ情報入力画面を表示する。【ユーザ ID】を選択するとユーザ情報

テーブルを参照しユーザ情報編集画面を表示する。【TOP へ戻る】を押すと管理者専用 TOP モジュールを呼び出す。

- ユーザ情報の入力を行い完了ボタンを押すと、ユーザ情報変更確認画面を表示される。また、【削除】を選択した場合にも同じくユーザ情報変更確認画面が表示される。
- エラーが無ければ変更された情報をユーザ情報テーブルに更新し、ユーザ情報変更完了画面が表示され、【ユーザ情報管理へ】を押すことでモジュールの開始に戻る。エラーがある場合は、エラー画面を表示しモジュールの開始に戻る。エラーが発生した場合にはテーブルの更新は行われない。

5.14 システムログ参照モジュール

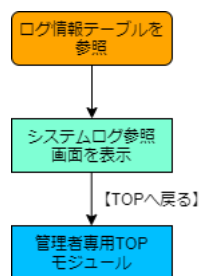


図 15 システムログ参照モジュールのフローチャート

概要

図 15 は、システムログを参照する流れを示している。

処理

- ログ情報テーブルを参照し、システムログ参照画面を表示する。
- 【TOP へ戻る】を押すと管理者専用 TOP モジュールを呼び出す。

5.15 よくある質問情報管理モジュール

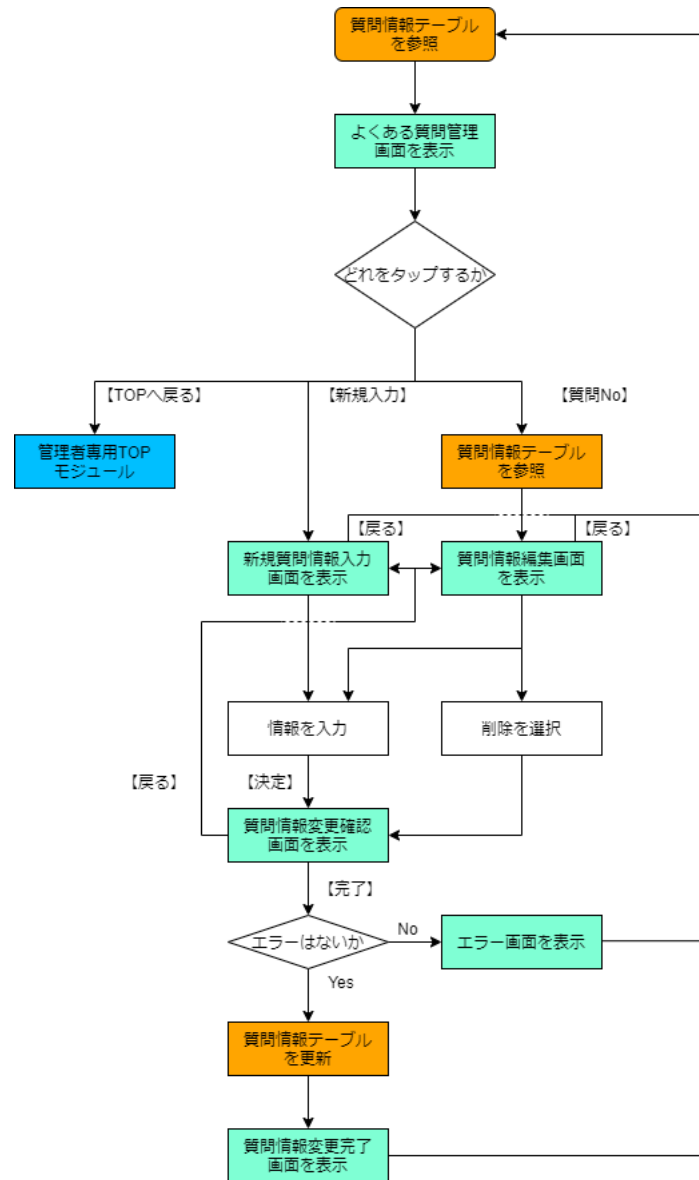


図 16 よくある質問情報管理モジュールのフローチャート

概要

図 16 は、よくある質問情報を管理する流れを示している。

処理

処理の流れはユーザ情報管理モジュールと同様である。

5.16 マニュアル情報管理モジュール

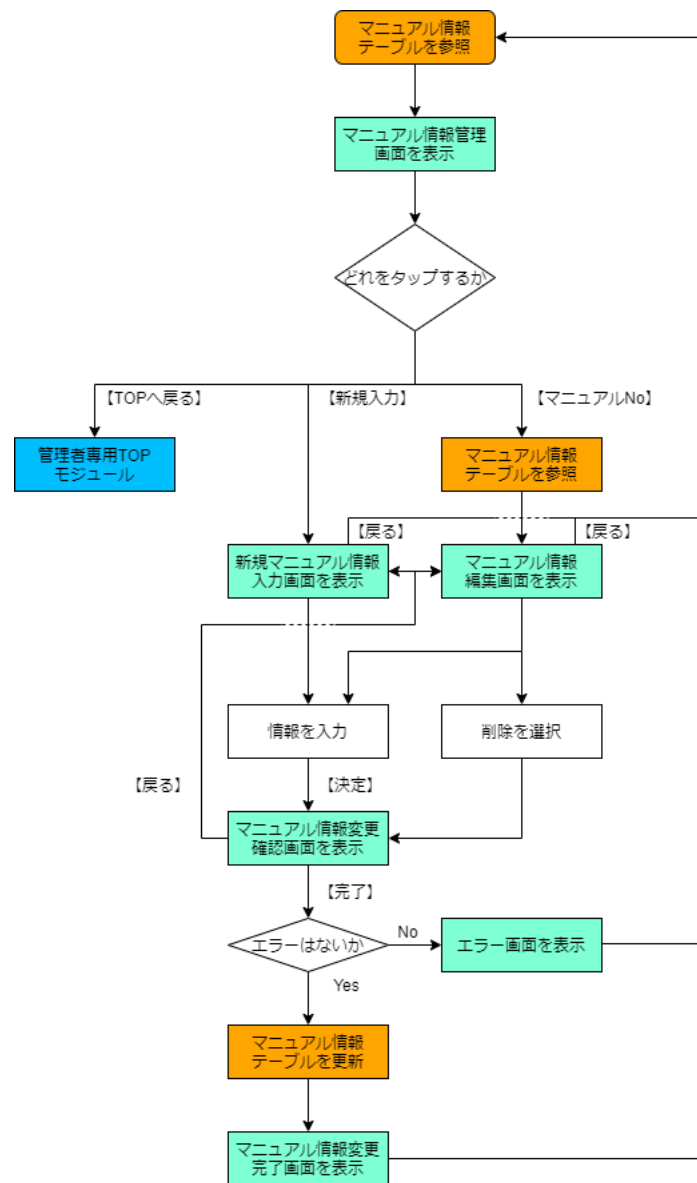


図 17 マニュアル管理モジュールのフローチャート

概要

図 17 は、マニュアル情報を管理する流れを示している。

処理

処理の流れはユーザ情報管理モジュールと同様である。

6 モジュール詳細

本システムのモジュールの詳細を下記に示す。

6.1 j_general_top.html

総合 TOP 画面を表示するファイルである。

1. 【フロント業務機能】を押すと、f_top.html に遷移する。
2. 【レストラン業務機能】を押すと、r_information_meal.html に遷移する。
3. 【清掃業務機能】を押すと、s_clean_management.html に遷移する。
4. 【マニュアル・質問閲覧】を押すと、j_manualquestion_top.html に遷移する。
5. 【管理者専用機能】を押すと、k_top.html に遷移する。

6.2 j_general_top.php

本システム利用者の総合 TOP 画面を実装するファイルである。

- メソッド名：JGeneralAuthorityP
ログインしているユーザ権限から、押すことのできるボタンを設定するメソッドである。
 - 引数：authority
 - 戻り値：なし
 - 処理：ログインしているユーザ権限を受け取る。
ユーザ権限からボタンを表示する。

6.3 j_login.html

ログイン画面を表示するファイルである。

1. 【ログイン】を押し、ログインに成功すると j_general_top.html に遷移する。
2. 【ログイン】を押し、ログインに失敗すると j_login.html にとどまる。

6.4 j_login.php

本システム利用者のログイン画面を実装するファイルである。

- メソッド名：JLoginP
入力されたユーザ ID とパスワードが、DB に登録されているものと一致するかを確認するメソッドである。
 - 引数：user_id, user_pass
 - 戻り値：なし

- 処理：入力されたユーザ ID とパスワードを受け取る。
ユーザ ID とパスワードをユーザ情報テーブルに要求する。
ユーザ ID とパスワードが入力されたものと一致しているかを確認する。
ログイン状態にする。

6.5 f_top.html

フロント担当者 TOP 画面を表示するファイルである。

1. 【新規入力】を押すと、f_reservation_input.html に遷移する。
2. 【予約検索】を押すと、f_search.html に遷移する。
3. 【部屋欄】を押すと、f_information_details.html に遷移する。

6.6 f_reservation_input.html

予約入力画面を表示するためのファイルである。

1. 【入力完了】を押すと、f_reservation_confirmation.html に遷移する。
2. 【戻る】を押すと、f_top.html に遷移する。

6.7 f_reservation_confirmation.html

予約入力確認画面を表示するためのファイルである。

1. 【完了】を押し、エラーがなければ、f_top.html に遷移する。
2. 【完了】を押し、エラーがあれば、f_reservation_input.html に遷移する。
3. 【戻る】を押すと、f_reservation_input.html に遷移する。

6.8 f_reservation_done.html

予約入力完了画面を表示するためのファイルである。

1. 【戻る】を押すと、f_top.html に遷移する。

6.9 f_search.html

予約検索画面を表示するファイルである。

1. 【完了】を押すと、f_search_result.html に遷移する。
2. 【戻る】を押すと、f_top.html に遷移する。

6.10 f_search_result.html

予約検索結果画面を表示するファイルである。

1. 【氏名欄】を押すと、f_information_details.html に遷移する。
2. 【戻る】を押すと、f_search.html に遷移する。

6.11 f_information_details.html

予約情報詳細画面を表示するファイルである。

1. 【予約情報編集】を押すと、f_information_edit.html に遷移する。
2. 【追加料金】を押すと、f_addfee_edit.html に遷移する。
3. 【戻る】を押すと、f_top.html から遷移してきた場合、f_top.html に遷移する。 r_information_meal.html から遷移してきた場合、r_information_meal.html に遷移する。

6.12 f_information_edit.html

予約情報編集画面を表示するファイルである。

1. 【完了】を押すと、f_information_confirmation.html に遷移する。
2. 【戻る】を押すと、f_information_details.html に遷移する。

6.13 f_information_confirmation.html

予約情報編集確認画面を表示するファイルである。

1. 【完了】を押し、エラーがなければ、f_information_done.html に遷移する。
2. 【完了】を押し、エラーがあれば、f_information_edit.html に遷移する。
3. 【戻る】を押すと、f_information_edit.html に遷移する。

6.14 f_information_done.html

予約情報編集完了画面を表示するファイルである。

1. 【戻る】を押すと、f_top.html に遷移する。

6.15 f_addfee_edit.html

追加料金編集画面を表示するファイルである。

1. 【完了】を押すと、f_information_details.html に遷移する。

2. 【戻る】を押すと、f_addfee_confirmation.html に遷移する。

6.16 f_addfee_confirmation.html

追加料金編集確認画面を表示するファイルである。

1. 【完了】を押し、エラーがなければ、f_addfee_done.html に遷移する。
2. 【完了】を押し、エラーがあれば、f_addfee_edit.html に遷移する。
3. 【戻る】を押すと、f_addfee_edit.html に遷移する。

6.17 f_addfee_done.html

追加料金編集完了画面を表示するファイルである。

1. 【戻る】を押すと、f_information_details.html に遷移する。

6.18 f_customre.php

- メソッド名：FrontInsertP

顧客情報テーブルに情報を入力する。

- 引数 : reservation_id, stay_date, reservation_date, stay_count, customer_name, customer_address, phone_number, adult, child, customer_plan, dinner, is_dinner, dinner_menu, is_breakfast, breakfast_menu, room_1, room_2, room_3, customer_checkin, customer_remark
- 戻り値：なし
- 処理 : 入力画面から必要な情報を受け取り、顧客情報テーブルに登録する。

- メソッド名：FrontPdateP

顧客情報テーブルの内容を編集する。

- 引数 : reservation_id, stay_date, reservation_date, stay_count, customer_name, customer_address, phone_number, adult, child, customer_plan, dinner, is_dinner, dinner_menu, is_breakfast, breakfast_menu, room_1, room_2, room_3, customer_checkin, customer_remark
- 戻り値：なし
- 処理 : FrontInsertP と FrontDeleteP を関数呼び出しする。

- メソッド名：FrontDeleteP

顧客情報テーブルの情報を削除する。

- 引数 : reservation_id
- 戻り値：なし
- 処理 : 選択した顧客情報を顧客情報テーブルから削除する。

- メソッド名：FrontSelectP

顧客情報テーブルから情報を取得する。

- 引数 : stay_date, customre_name
- 戻り値 : reservation_id, stay_date, reservation_date, stay_count, customer_name, customer_address, phone_number, adult, child, customer_plan, dinner, is_dinner, dinner_menu, is_breakfast, breakfast_menu, room_1, room_2, room_3, customer_checkin, suctomer_remark
- 処理 : 顧客情報テーブルに接続し、検索条件の情報を取得する。

6.19 r_information_meal.html

食事情報閲覧画面を実装するファイルである。

1. 【氏名欄】を押すと、FrontSelectP を参照し、f_information_details.html に遷移する。
2. 【戻る】を押すと、j_general_top.html に遷移する。

6.20 r_information_meal.php

食事情報閲覧画面を実装するファイルである。

- メソッド名：RInformationMealP

食事情報閲覧画面に必要な情報を、顧客情報テーブルから受け取るメソッドである。

- 引数 : なし
- 戻り値 : name, adult, child, dinner_menu
- 処理 : 顧客情報テーブルに接続する
顧客情報テーブルから転送された情報を受け取る。

6.21 r_fee.php

- メソッド名：FeeInsertP

追加料金情報テーブルに情報を追加する

- 引数 : fee_id, fee_date, fee_name, fee_place, fee_add, fee_contents, fee_remark
- 戻り値 : なし
- 処理 : 入力画面から必要な情報を受け取り、追加料金情報テーブルに登録する。

- メソッド名：FeeEditP

追加料金情報テーブルの内容を編集する。

- 引数 : fee_id, fee_date, fee_name, fee_place, fee_add, fee_contents, fee_remark
- 戻り値 : なし

- 処理 : FeeInsertP と FeeDeliteP を関数呼び出しする.
- メソッド名: FeeDeliteP
 追加料金情報テーブルの情報を削除する.
 - 引数 : fee_id
 - 戻り値: なし
 - 処理 : 選択した顧客情報を顧客情報テーブルから削除する.
- メソッド名: FeeSelectP
 追加料金情報テーブルから情報を取得する.
 - 引数 : fee_id
 - 戻り値: fee_place, fee_add, fee_contents, fee_remark
 - 処理 : 追加料金情報テーブルに接続し, 情報を取得する.

6.22 s_clean_management.html

清掃情報管理画面を表示するファイルです.

1. SCleanManagementP, SCleanNumberP から受け取った情報を表示する.
2. 【部屋】を選択することで, s_clean_edit.html へ遷移する.

6.23 s_clean_edit.html

清掃情報編集画面を表示するファイルです.

1. 【(ステイ) 未清掃・清掃済み】を選択し, 【完了】を押すことで, s_clean_management.html へ遷移する.
2. 【戻る】を押すことで, s_clean_management.html へ遷移する.

6.24 s_clean_management.php

清掃情報管理画面を実装するファイルである.

- メソッド名: SCleanManagementP
 清掃情報管理画面に必要な情報を, 部屋情報テーブルから受け取るメソッドである.
 - 引数: なし
 - 戻り値: room_number, room_clean
 - 処理: 部屋情報テーブルに必要な情報を要求する.
 部屋情報テーブルから転送された情報を受け取る.
- メソッド名: SCleanNumberP
 清掃情報管理画面の中の宿泊している人数を顧客情報テーブルから受け取るメソッドである.

- 引数: stay_date, room_number
- 戻り値: adult, child
- 処理: 大人と子供の情報を顧客情報テーブルに要求する。
顧客情報テーブルから転送された情報を受け取る。
- メソッド名: SCleanEditP
清掃情報テーブルを編集するテーブルである。
 - 引数: room_number, room_clean
 - 戻り値: なし
 - 処理: 入力された部屋の掃除状況に、部屋情報テーブルを更新する。

6.25 j_manualquestion_top.html

マニュアル・よくある質問閲覧 TOP 画面を表示するファイルです。

1. 【マニュアル】を押すと、j_manual_selection.php に遷移する。
2. 【よくある質問閲覧】を押すと、j_question_selection.php に遷移する。

6.26 j_manual_selection.php

従業員用のマニュアル選択画面を実装するファイルである。

- メソッド名: JManualSelectionP
マニュアル選択画面に必要な情報を、マニュアル情報テーブルから受け取るメソッドである。
 - 引数: なし
 - 戻り値: manual_number, manual_name, manual_url
 - 処理: マニュアル情報テーブルに必要な情報を要求する。
マニュアル情報テーブルから転送された情報を受け取る。

6.27 j_question_selection.php

従業員用のよくある質問選択画面を実装するファイルである。

- メソッド名: JQuestionSelectionP
よくある質問選択画面に必要な情報を、マニュアル情報テーブルから受け取るメソッドである。
 - 引数: なし
 - 戻り値: question_number, question_name, question_result
 - 処理: よくある質問情報テーブルに必要な情報を要求する。
よくある質問情報テーブルから転送された情報を受け取る。

6.28 j_manual_selection.html

マニュアル選択画面を表示するファイルです。

1. 【マニュアル】を選択すると、j_manual_reading.html に遷移する。

6.29 j_manual_reading.html

マニュアル閲覧画面を表示するファイルです。

1. データベースに格納されているマニュアルを表示する。

6.30 k_top.html

管理者専用機能 TOP 画面を表示するファイルである。

1. 【ユーザ情報管理】を選択すると、k_user_management.html に遷移する。
2. 【システムログ参照】を選択すると、k_log_list.html に遷移する。
3. 【質問情報管理】を選択すると、k_question_management.html に遷移する。
4. 【マニュアル情報管理】を選択すると、k_manual_management.html に遷移する。
5. 【総合 TOP に戻る】を選択すると、j_general_top..html に遷移する。

6.31 k_user_management.html

ユーザ情報管理画面を表示するファイルである。

1. 【TOP へ戻る】を押すとすると、k_top.html に遷移する。
2. 【新規入力】を押すと、k_user_input.html に遷移する。
3. 【ユーザ ID】を押すと、k_user_edit.html に遷移する。

6.32 k_user_input.html

ユーザ情報新規入力画面を表示するファイルである。

1. 【戻る】を押すと、k_top.html に遷移する。
2. 【完了】を押すと、k_user_confirmation.html に遷移する。

6.33 k_user_edit.html

ユーザ情報編集画面を表示するファイルである。

1. 【戻る】を押すと、k_top.html に遷移する。

2. 【完了】または【削除】を押すと、k_user_confirmation.html に遷移する。

6.34 k_user_confirmation.html

ユーザ情報変更確認画面を表示するファイルである。

1. 【戻る】を選択すると、k_user_edit.html に遷移する。
2. 【完了】を押す、エラーが無ければ、k_user_done.html に遷移する。
3. 【完了】を、エラーがあれば、k_user_edit.html に遷移する。

6.35 k_user_done.html

ユーザ情報変更完了画面を表示するファイルである。

1. 【ユーザ情報管理へ】を選択すると、k_user_management.html に遷移する。

6.36 k_user_manegement.php

ユーザ情報管理画面とユーザ情報の管理を実装するファイルである。

- メソッド名：KUserManagementP
ユーザ情報管理画面に必要な情報を、ユーザ情報テーブルから受け取るメソッドである。
 - 引数：なし
 - 戻り値：user_id, user_name
 - 処理：ユーザ情報テーブルに接続する。
ユーザ情報テーブルから転送された情報を受け取る。
- メソッド名：KUserInputP
ユーザ情報の登録を行うメソッドである。
 - 引数：user_id, user_name, user_pass, authority
 - 戻り値：なし
 - 処理：入力された情報をユーザ情報テーブルに登録する。
- メソッド名：KUserEditP
ユーザ情報の編集を行うメソッドである。
 - 引数：user_id, user_name, user_pass, authority
 - 戻り値：なし
 - 処理：入力された情報にユーザ情報テーブルを更新する。
- メソッド名：KUserDelP
ユーザ情報の削除を行うメソッドである。

- 引数：user_del
- 戻り値：なし
- 処理：ユーザ情報をユーザ情報テーブルから削除する。

6.37 k_log_list.html

ログ情報一覧画面を表示するファイルである。

1. 【TOP へ戻る】を選択すると、k_top.html に遷移する。

6.38 k_log_list.php

ログ情報一覧画面を実装するファイルである。

- メソッド名：KLogListP

ログ情報一覧画面に必要な情報を、ログ情報テーブルから受け取るメソッドである。

- 引数：なし
- 戻り値：log_date, log_name, log_work, log_table, log_line, log_attribute, log_befor, log_after
- 処理：ログ情報テーブルに接続する。
ログ情報テーブルから転送された情報を受け取る。

6.39 k_log_record.php

ログ情報の記録を行うファイルである。

- メソッド名：KLogRecordP

他のテーブルの内容が変更された際に、ログ情報テーブルに変更内容を記録を行うメソッドである。

- 引数：user_name, log_work, log_table, log_line, log_attribute, log_befor, log_after
- 戻り値：なし
- 処理：ログ情報テーブルにログ情報の記録を行う。

6.40 k_question_management.html

質問情報管理画面を表示するファイルである。

1. 【TOP へ戻る】を押すとすると、k_top.html に遷移する。
2. 【新規入力】を押すと、k_question_input.html に遷移する。
3. 【質問 No】を押すと、k_question_edit.html に遷移する。

6.41 k_question_input.html

質問情報新規入力画面を表示するファイルである。

1. 【戻る】を押すと、k_top.html に遷移する。
2. 【完了】を押すと、k_question_confirmation.html に遷移する。

6.42 k_question_edit.html

質問情報編集画面を表示するファイルである。

1. 【戻る】を押すと、k_top.html に遷移する。
2. 【完了】または【削除】を押すと、k_question_confirmation.html に遷移する。

6.43 k_question_confirmation.html

質問情報変更確認画面を表示するファイルである。

1. 【戻る】を選択すると、k_question_edit.html に遷移する。
2. 【完了】を押す、エラーが無ければ、k_question_done.html に遷移する。
3. 【完了】を、エラーがあれば、k_question_edit.html に遷移する。

6.44 k_question_done.html

質問情報変更完了画面を表示するファイルである。

1. 【質問情報管理へ】を選択すると、k_question_management.html に遷移する。

6.45 k_question_management.php

管理者用のよくある質問管理画面と質問情報の管理を実装するファイルである。

- メソッド名：KQuestionManagementP
よくある質問管理画面に必要な情報を、質問情報テーブルから受け取るメソッドである。
 - 引数：なし
 - 戻り値：question_number, question_name, question_result
 - 処理：質問情報テーブルに接続する
質問情報テーブルから転送された情報を受け取る。
- メソッド名：KQuestionInputP
質問情報の登録を行うメソッドである。

- 引数：question_name, question_result
- 戻り値：なし
- 処理：入力された情報を質問情報テーブルに登録する。

- メソッド名：KQuestionEditP

質問情報の編集を行うメソッドである。

- 引数：question_number, question_name, question_result
- 戻り値：なし
- 処理：入力された情報に質問情報テーブルを更新する。

- メソッド名：KQuestionDelP

質問情報の削除を行うメソッドである。

- 引数：question_del
- 戻り値：なし
- 処理：質問情報を質問情報テーブルから削除する。

6.46 k_manual_management.html

マニュアル情報管理画面を表示するファイルである。

1. 【TOP へ戻る】を押すとすると、k_top.html に遷移する。
2. 【新規入力】を押すと、k_manual_input.html に遷移する。
3. 【マニュアル No】を押すと、k_manual_edit.html に遷移する。

6.47 k_manual_input.html

マニュアル情報新規入力画面を表示するファイルである。

1. 【戻る】を押すと、k_top.html に遷移する。
2. 【完了】を押すと、k_manual_confirmation.html に遷移する。

6.48 k_manual_edit.html

マニュアル情報編集画面を表示するファイルである。

1. 【戻る】を押すと、k_top.html に遷移する。
2. 【完了】または【削除】を押すと、k_manual_confirmation.html に遷移する。

6.49 k_manual_confirmation.html

マニュアル情報変更確認画面を表示するファイルである。

1. 【戻る】を選択すると、k_manual_edit.html に遷移する。
2. 【完了】を押し、エラーが無ければ、k_manual_done.html に遷移する。
3. 【完了】を、エラーがあれば、k_manual_edit.html に遷移する。

6.50 k_manual_done.html

マニュアル情報変更完了画面を表示するファイルである。

1. 【マニュアル情報管理へ】を選択すると、k_manual_management.html に遷移する。

6.51 k_manual_management.php

管理者用のマニュアル情報管理画面とマニュアルの管理を実装するファイルである。

- メソッド名：KManualManagementP
マニュアル情報管理画面に必要な情報を、マニュアル情報テーブルから受け取るメソッドである。
 - 引数：なし
 - 戻り値：manual_number, manual_name, manual_url
 - 処理：マニュアル情報テーブルに接続する
マニュアル情報テーブルから転送された情報を受け取る。
- メソッド名：KManualInputP
マニュアル情報の登録を行うメソッドである。
 - 引数：manual_name, manual_url
 - 戻り値：なし
 - 処理：入力された情報をマニュアル情報テーブルに登録する。
- メソッド名：KManualEditP
マニュアル情報の編集を行うメソッドである。
 - 引数：manual_number, manual_name, manual_url
 - 戻り値：なし
 - 処理：入力された情報にマニュアル情報テーブルを更新する。
- メソッド名：KManualDelP
マニュアル情報の削除を行うメソッドである。
 - 引数：manual_del

- 戻り値：なし
- 処理：マニュアル情報をマニュアル情報テーブルから削除する。

7 データベース設計

本システムで用いるデータベースの構成と詳細を記述する。各テーブル表の下には、データベースを作成するための CREATE 文と、各属性の説明を記述する。また、PRIMARYKEY は PK, FOREIGNKEY は FK, COMPOSITE PRIMARYKEY は CPK と省略して記述する。

7.1 ユーザ情報テーブル

ユーザ情報テーブルは、本システムのユーザ認証を行う際に利用するユーザ情報の管理を行う。また、ユーザの登録時に入力された情報の管理を行う。テーブルの詳細は表 5 に示す。

表 5 ユーザ情報テーブル (user)

属性	変数名	データ型	制約条件	備考
ユーザ ID	user_id	INTEGER	PK, NOT NULL	-
ユーザ氏名	user_name	NVARCHAR(16)	NOT NULL	-
パスワード	user_pass	CHAR(128)	NOT NULL	ハッシュ化
ユーザ権限	authority	CHAR(5)	NOT NULL	-

```
CREATE TABLE user (
  user_id      INTEGER NOT NULL ,
  user_name    NVARCHAR(16) NOT NULL ,
  user_pass    CHAR(128) NOT NULL ,
  authority    CHAR(5) NOT NULL ,
  PRIMARY KEY (user_id)
);
```

- ユーザ ID(user_id)
ユーザ情報を識別するための番号である。また、ユーザ情報テーブルの主キーであり、半角数字で構成する。
- ユーザ氏名 (user_name)
ユーザの氏名を全角 16 文字 (漢字を含む) で構成する。
- パスワード (user_pass)
ハッシュ化されたパスワードを半角英数字 128 文字で構成する。

- ユーザ権限 (authority)

チェックボックスより与えられた情報を結合し、半角英数字 5 文字以下で構成する。例えば、管理者権限のみの場合、1 という文字列が記録される。フロント担当と清掃担当の権限が付与される場合には、23 という文字列によって権限を記録する。権限が 3 つの場合は 234 として権限を記録する。そのため、桁数は権限の数と対応する。各権限毎に与える数字は、表 6 に示す。

表 6 権限割当て対応表

権限内容	割当てる数字
管理者	1
フロント担当	2
清掃担当	3
レストラン担当	4
アルバイト	5

7.2 ログ情報テーブル

ログ情報テーブルは、本システムのユーザが各テーブル内容を変更した際の、記録と管理を行う。テーブルの詳細は表 7 に示す。

表 7 ログ情報テーブル (log)

属性	変数名	データ型	制約条件	備考
日時	log_date	DATE	CPK, NOT NULL	-
変更者	log_name	VARCHAR(16)	CPK, NOT NULL	-
変更業務	log_work	TEXT	NOT NULL	-
変更テーブル	log_table	TEXT	NOT NULL	-
変更行	log_line	TEXT	NOT NULL	-
変更属性	log_attribute	TEXT	NOT NULL	-
変更前	log_befor	TEXT	NOT NULL	-
変更後	log_after	TEXT	NOT NULL	-

```
CREATE TABLE log (
  log_date      DATE NOT NULL ,
  log_name      NVARCHAR(16) NOT NULL ,
  log_work      TEXT NOT NULL ,
  log_table     TEXT NOT NULL ,
  log_line      TEXT NOT NULL,
  log_attribute TEXT NOT NULL ,
  log_befor     TEXT NOT NULL ,
  log_after     TEXT NOT NULL ,
  PRIMARY KEY (log_date, log_name),
  FOREIGN KEY (log_name) REFERENCES user(user_name)
);
```

- 日時 (log_date)
各テーブルの内容を変更した際の時間である。ログ情報テーブルの主キーの一つであり、DATE 型で構成する。
- 変更者 (log_name)
テーブルを変更したユーザ氏名である、ログ情報テーブルの主キーの一つであり、全角 16 文字で構成する。
- 変更業務 (log_work)
変更した業務名であり、TEXT 型で構成する。
- 変更テーブル (log_table)
変更したテーブル名であり、TEXT 型で構成する。
- 変更行 (log_line)
変更したテーブルの主キーであり、TEXT 型で構成する。
- 変更属性 (log_attribute)
変更したテーブルの属性であり、TEXT 型で構成する。
- 変更前 (log_befor)
変更する前の業務内容であり、TEXT 型で構成する。
- 変更後 (log_after)
変更した後の業務内容であり、TEXT 型で構成する。

7.3 マニュアル情報テーブル

マニュアル情報テーブルは、ホテルの業務マニュアルの記録と管理を行う。テーブルの詳細を 8 に示す。

表 8 マニュアル情報テーブル (manual)

属性	変数名	データ型	制約条件	備考
マニュアル番号	manual_number	CHAR(4)	PK, NOT NULL	-
ファイル名	manual_name	VARCHAR(16)	NOT NULL	-
URL	manual_url	TEXT	NOT NULL	-

```
CREATE TABLE manual(  
  manual_number CHAR(4) NOT NULL,  
  manual_name VARCHAR(16) NOT NULL,  
  manual_url TEXT NOT NULL,  
  PRIMARY KEY(manual_number)  
);
```

- マニュアル番号 (manual_number)
マニュアル情報テーブルの主キーであり、半角英数字 4 文字で構成する。
- ファイル名 (manual_name)
マニュアルが記述されている PDF のファイル名であり、半角英数字 16 文字で構成する。
- URL(manual_url)
マニュアルが記述されている PDF の URL であり、TEXT 型で構成する。

7.4 よくある質問情報テーブル

よくある質問とその解答例の記録と管理を行う。テーブルの詳細を 9 に示す。

表 9 よくある質問情報テーブル (question)

属性	変数名	データ型	制約条件	備考
質問番号	question_number	CHAR(4)	PK, NOT NULL	-
よくある質問	question_name	TEXT	NOT NULL	-
解答例	question_result	TEXT	NOT NULL	-

```
CREATE TABLE question{
question_number CHAR(4) NOT NULL,
question_name TEXT NOT NULL,
question_result TEXT NOT NULL,
PRIMARY KEY(question_number)
};
```

- 質問番号 (question_number)
よくある質問テーブルの主キーであり，半角英数字 4 文字で構成する．
- よくある質問 (question_name)
よくある質問内容であり，TEXT 型で構成する．
- 解答例 (question_result)
よくある質問の解答例であり，TEXT 型で構成する

7.5 顧客情報テーブル

顧客情報テーブルは，お客様の予約情報の管理を行う．テーブルの構成を表 10 に示す．

表 10 顧客情報テーブル (customer)

属性	変数名	データ型	制約条件	備考
予約 ID	reservation_id	CHAR(11)	PK, NOT NULL	-
宿泊日	stay_date	DATE	NOT NULL	(yyyy/mm/dd)
予約日	reservation_date	DATE	NOT NULL	(yyyy/mm/dd)
宿泊数	stay_count	INTEGER	NOT NULL	-
氏名	customer_name	NVARCHAR(16)	NOT NULL	-
住所	customer_address	TEXT	NOT NULL	-
電話番号	phone_number	TEXT	NOT NULL	-
大人	adult	INTEGER	NOT NULL	-
子供	child	INTEGER	-	-
プラン	customer_plan	TEXT	NOT NULL	-
夕食の有無	is_dinner	INTEGER	NOT NULL	-
夕食のメニュー	dinner_menu	TEXT	-	-
朝食の有無	is_breakfast	INTEGER	NOT NULL	-
朝食のメニュー	breakfast_menu	TEXT	-	-
部屋番号 1	room_1	INTEGER	FK, NOT NULL	references room
部屋番号 2	room_2	INTEGER	FK	references room
部屋番号 3	room_3	INTEGER	FK	references room
チェックイン	customer_checkin	INTEGER	NOT NULL	-
備考	customer_remark	INTEGER	-	-

```

CREATE TABLE customer(
    reservation_id    CHAR(11) NOT NULL,
    stay_date         NOT NULL,
    reservation_date   DATE NOT NULL,
    stay_count        INTEGER NOT NULL,
    customer_name      NVARCHAR(16) NOT NULL,
    customer_address   TEXT NOT NULL,
    phone_number       TEXT NOT NULL,
    adult             INTEGER NOT NULL,
    child             INTEGER,
    customer_plan      TEXT NOT NULL,
    is_dinner          INTEGER NOT NULL,
    dinner_menu        TEXT,
    is_breakfast       INTEGER NOT NULL,
    breakfast_menu     TEXT,
    room_1            INTEGER NOT NULL,
    room_2            INTEGER,
    room_3            INTEGER,
    customer_checkin   INTEGER NOT NULL,
    customer_remark    INTEGER,
    PRIMARY KEY(reservation_id),
    FOREIGN KEY(room_1) REFERENCES room(room_number),
    FOREIGN KEY(room_2) REFERENCES room(room_number),
    FOREIGN KEY(room_3) REFERENCES room(room_number)
);

```

- 予約 ID(reservation_id)
顧客情報テーブルの主キーであり、宿泊年月日 8 桁と当日宿泊予定の部屋番号 3 桁を組み合わせた半角数字 11 桁で構成する。
- 宿泊日 (stay_date)
お客様の宿泊する初日の日付であり、年は西暦表記とする。
- 予約日 (reservation_date)
新規予約登録をした日付である。
- 宿泊数 (stay_count)
お客様の宿泊数であり、整数 2 桁で構成する。
- 氏名 (customer_name)
お客様の氏名であり、全角 16 文字以内の漢字表記で構成する。苗字と名前の間は、半角スペースを挿入して格納する。
- 住所 (customer_address)

お客様の住所であり、データ型を TEXT 型とし、日本語表記で構成する。

- 電話番号 (phone_number)

お客様の電話番号を、データ型を、TEXT 型とし、ハイフンなしの数字のみで構成する。

- 大人 (adult)

大人の人数を半角数字で構成する。

- 子供 (child)

子供の人数を半角数字で構成する。

- プラン (customer_plan)

プランの内容を日本語表記で構成する。

- 夕食の有無 (is_dinner)

夕食の有無を半角数字で構成し、以下のように定義する。

数値	内容
0	食事なし
1	食事あり

- 夕食メニュー (dinner_menu)

夕食のメニューを全角かな文字の TEXT 型で構成する。数泊する場合は、すべての食事を入力する。

- 朝食の有無 (is_breakfast)

朝食の有無を半角数字で構成し、以下のように定義する。

数値	内容
0	食事なし
1	食事あり

- 朝食メニュー (breakfast_menu)

朝食のメニューを全角かな文字の TEXT 型で構成する。数泊する場合は、すべての食事を入力する。

- 部屋番号 1(room_1)

お客様の利用する部屋番号を半角数字で構成する。

- 部屋番号 2(room_2)

お客様の利用する部屋番号を半角数字で構成する。宿泊人数が多い場合のみに利用する。

- 部屋番号 3(room_3)

お客様の利用する部屋番号を半角数字で構成する。宿泊人数が多い場合のみに利用する。

- チェックイン (customer_checkin)

お客様のチェックインの情報を半角数字で構成する。以下のように定義する。

数値	内容
0	未チェックイン
1	在泊中
2	外出中

- 備考 (customer_remark)
備考を全角かな文字で構成する。

7.6 過去顧客情報テーブル

過去顧客情報テーブルは、過去のお客様の情報の管理を行うテーブル構成の詳細を表 11 に示す。

表 11 過去顧客情報テーブル (past_customer)

属性	変数名	データ型	制約条件	備考
予約 ID	reservation_id	CHAR(11)	PK, NOT NULL	-
宿泊日	stay_date	DATE	NOT NULL	(yyyy/mm/dd)
予約日	reservation_date	DATE	NOT NULL	(yyyy/mm/dd)
宿泊数	stay_count	INTEGER	NOT NULL	-
氏名	customer_name	NVARCHAR(16)	NOT NULL	-
住所	customer_address	TEXT	NOT NULL	-
電話番号	phone_number	TEXT	NOT NULL	-
大人	adult	INTEGER	NOT NULL	-
子供	child	INTEGER	-	-
プラン	customer_plan	TEXT	NOT NULL	-
備考	customer_remark	INTEGER	-	-

```
CREATE TABLE customer(
    reservation_id    CHAR(11) NOT NULL,
    stay_date DATE     NOT NULL,
    reservation_date   DATE NOT NULL,
    stay_count         INTEGER NOT NULL,
    customer_name      NVARCHAR(16) NOT NULL,
    customer_address   TEXT NOT NULL,
    phone_number       TEXT NOT NULL,
    adult              INTEGER NOT NULL,
    child              INTEGER,
    customer_plan       TEXT NOT NULL,
    PRIMARY KEY(reservation_id)
);
```

- 予約 ID(reservation_id)
顧客情報テーブルの主キーであり、宿泊年月日 8 桁と当日宿泊予定の部屋番号 3 桁を組み合わせた半角数字 11 桁で構成する。
- 宿泊日 (stay_date)
お客様の宿泊する初日の日付であり、年は西暦表記とする。
- 予約日 (reservation_date)
新規予約登録をした日付である。
- 宿泊数 (stay_count)
お客様の宿泊数であり、整数 2 桁で構成する。
- 氏名 (customer_name)
お客様の氏名であり、全角 16 文字以内の漢字表記で構成する。苗字と名前の間は、半角スペースを挿入して格納する。
- 住所 (customer_address)
お客様の住所であり、データ型を TEXT 型とし、日本語表記で構成する。
- 電話番号 (phone_number)
お客様の電話番号を。データ型を、TEXT 型とし、ハイフンなしの数字のみで構成する。
- 大人 (adult)
大人の人数を半角数字で構成する。
- 子供 (adult)
子供の人数を半角数字で構成する。
- プラン (customer_plan)
プランの内容を日本語表記で構成する。

7.7 部屋情報テーブル

部屋情報テーブルは、ホテルの掃除状況の管理を行う。テーブルの詳細は表 12 に示す。

表 12 部屋情報テーブル (room)

属性	変数名	データ型	制約条件	備考
部屋番号	room_number	CHAR(4)	PK, NOT NULL	-
掃除状況	room_clean	INTEGER	NOT NULL	-

```
CREATE TABLE room(  
  room_number CHAR(4) NOT NULL,  
  room_clean INTEGER NOT NULL,  
  PRIMARY KEY(room_number)  
);
```

- 部屋番号 (room_number)
部屋情報テーブルの主キーであり、部屋番号を半角数字 4 文字で構成する。
- 掃除状況 (room_clean)
掃除状況を半角数字 1 文字で構成する。例えば、お客さまがチェックインされた部屋の状況を 1 という数字で記録する。各掃除状況に割当て数字を表 13 に示す。

表 13 掃除状況の割当て対応表

掃除状況	割当て数字
掃除していないもしくは予約なしの状況	0
お客様がチェックインしている状態	1
掃除済み状態	2

7.8 追加料金情報テーブル

追加料金情報テーブルは、本システムのフロント担当、またはレストラン担当のユーザが、宿泊客に追加料金が発生した際の、記録と管理を行う。テーブルの詳細は表 14 に示す。

表 14 追加料金情報テーブル (fee)

属性	変数名	データ型	制約条件	備考
予約 ID	fee_id	CHAE(11)	CPK, FK, NOT NULL	-
日付	fee_date	DATE	CPK, NOT NULL	-
氏名	fee_name	NVARCHAR(16)	FK, NOT NULL	-
場所	fee_place	TEXT	NOT NULL	-
追加料金	fee_add	INTEGER	NOT NULL	-
内容	fee_contents	TEXT	NOT NULL	-
備考	fee_remark	TEXT	-	-

```
CREATE TABLE fee (
    fee_id          CHAR(11) NOT NULL ,
    fee_date        DATE NOT NULL ,
    fee_name        NVARCHAR(16) NOT NULL ,
    fee_place       TEXT NOT NULL ,
    fee_add         INTEGER NOT NULL,
    fee_contents    TEXT NOT NULL ,
    fee_remark      TEXT NOT NULL ,
    PRIMARY KEY (fee_id,fee_date),
    FOREIGN KEY (fee_id) REFERENCES customer(reservation_id),
    FOREIGN KEY (fee_name) REFERENCES customer(name),
);
```

- 予約 ID (fee_id)
追加料金の発生するお客様の予約 ID である、また、追加料金情報テーブルの主キーの一つであり、顧客情報テーブルからの外部キーである。
- 日付 (fee_date)
追加料金を登録した際の時間である。また、追加料金情報テーブルの主キーの一つであり、DATE 型で構成する。
- 氏名 (fee_name)
追加料金が発生したお客様の氏名である、また、顧客情報テーブルからの外部キーである。
- 場所 (fee_place)
追加料金が発生した場所であり、TEXT 型で構成する。
- 追加料金 (fee_add)

追加料金の金額であり、INTEGER 型で構成する。

- 内容 (fee_contents)
追加料金の内容であり、TEXT 型で構成する。
- 備考 (fee_remark)
追加料金が発生した際に、上記以外の情報から TEXT 型で構成する。

付録 A ER 図

本システムで使用するデータベースの ER 図を図 18 に示す。

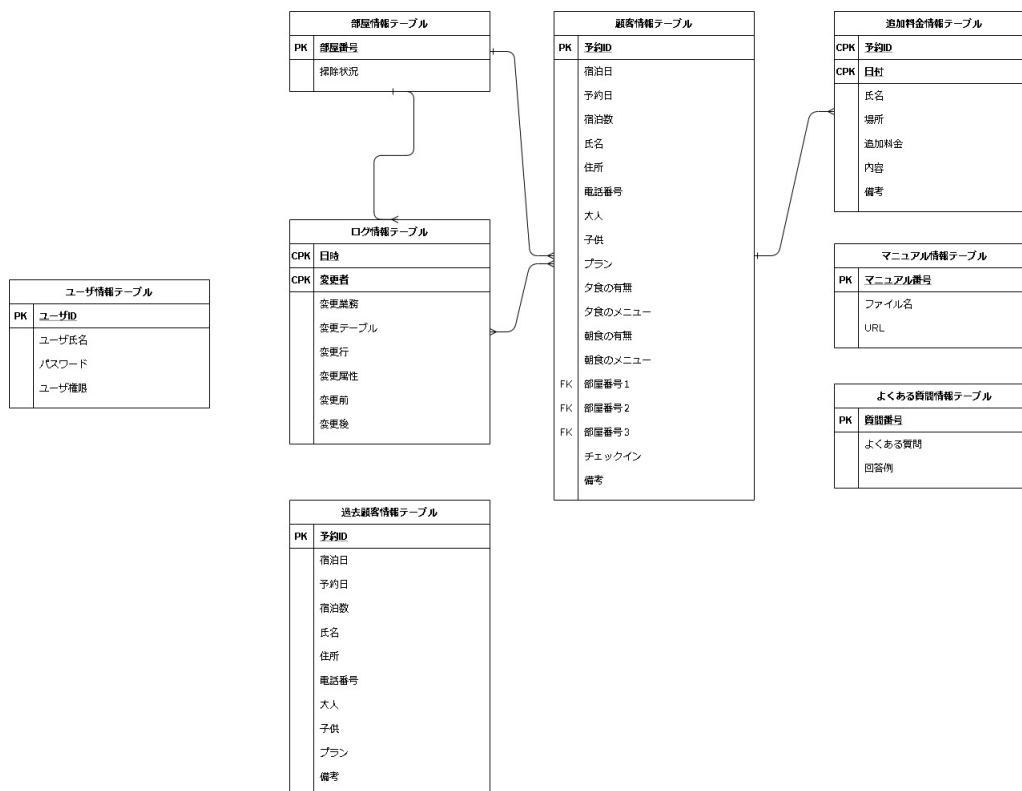


図 18 ER 図

付録 B 各メンバーの貢献内容

- ・ 今村 モジュール設計, モジュール詳細
- ・ 稲村 動作環境, 開発環境, データベース設計, ER 図, モジュール詳細
- ・ 海野 コーディング規約, モジュール設計, モジュール詳細
- ・ 菊森 データベース設計, モジュール詳細
- ・ 杉野 システム概要, データベース設計, モジュール詳細
- ・ 杉本 モジュール設計, モジュール詳細
- ・ 山下 モジュール設計, モジュール詳細