

RoR を用いた図書館予約システムの開発 —モデル中心設計とデータベース中心設計—

都丸智弘[†] 片瀬由貴[†] 大谷真[†]

湘南工科大学[†]

1. はじめに

湘南工科大学図書館に全学共有の学習室がある。これを有効活用するためには学内からネットワークを使って自由に予約でき、図書館側で管理するシステムが必要である。このシステムを本論文では図書館予約システムと呼ぶ。

一昨年に開発された図書館予約システムには、機能が不足している、ユーザインタフェースが適切でないなど様々な問題が存在している。そのため現行のシステムの問題点の分析・改良を行い、新たなシステムの設計・開発を行う必要が生じた。

開発には上流工程に UML を適用し、下流工程には RoR (Ruby on Rails) を適用した。これに伴い、UML のモデル中心設計と RoR のデータベース中心設計が一部不整合となり、その対策が課題となった。本論文では、この課題解決を含め、上流設計(要求分析とモデル設計)と RoR による実装開発について述べる。

2. 要求分析とモデル設計

2.1 要求機能

表 1. 新図書館予約システム機能一覧

分類	機能	新機能	改良
管理者の機能	利用者情報の登録・修正・削除・参照		○
	部屋予約情報の登録・修正・削除・参照(全ユーザで可能)		○
	休館日の登録・削除		○
	予約履歴の参照(全ユーザで可能)		○
	予約可能部屋の追加		△
	営業時間の変更(3種類から選ぶ)	○	
	利用履歴のExcel出力	○	
利用者の機能	部屋予約情報の登録・修正・削除・参照(利用者個人のみ)		○
	予約履歴の参照(利用者個人のみ)		○
	利用者情報の参照・修正申請(利用者個人のみ)	○	
全体的な機能	カレンダーによる予約状況の確認		○
	ユーザ認証		△
	予約確認等のメール通知	○	

新機能の営業時間変更とは、図書館の営業時間を変更する機能である。例えば平日の営業時間は 09:00~19:00 だが、学園祭の日なので 09:00~12:30 に変更したい時などに使う。予約確認等のメール通知は、部屋予約や利用者情報修正申請

などがあった時にメールが送られる機能である。

2.2 ユースケース図

UML を用いて作成したユースケース図を図 1 に示す。アクターは管理者と利用者で、管理者は全ての機能が使え、利用者には図 1 のような制限がある。

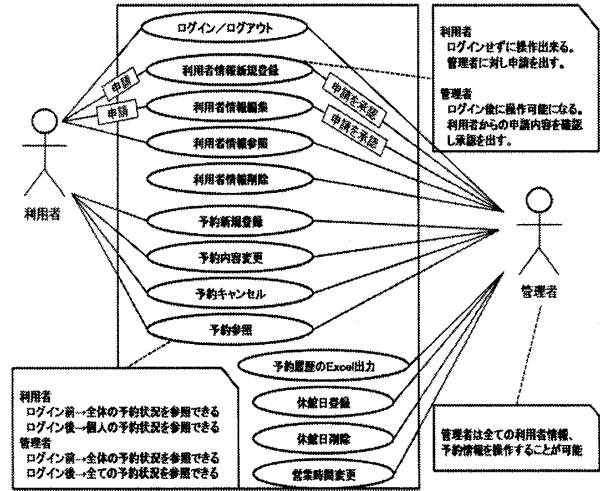


図 1. UML によるユースケース図

2.3 モデル図

モデル図(クラス図)を図 2 に示す。実装する際にはこのモデルに合わせてデータベース等を作成していく事になる。“利用者”は利用者ごとのユーザ名、パスワード、メールアドレスなどのオブジェクト、“予約”は利用者が行った個々の予約情報(日時や人数など)、“学習室”は部屋の特長、“営業情報”は図書館の営業情報(休日、開館時間など)を示す。

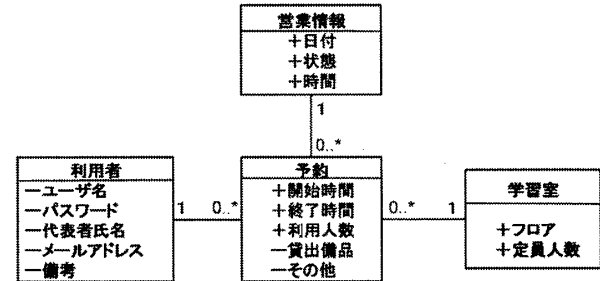


図 2. UML によるモデル図(クラス図)

3. 実装

3.1 UML モデルと RoR モデルの不整合

図 2 のモデル図に RoR の O/R マッピングをそのまま適用すると、これら 4 つのクラスはそれ

Development of library room reservation system applying RoR
[†]Tomohiro Tomaru, Yuki Katase, Makoto Oya, Shonan Institute of Technology

ぞれ 1 つの DB テーブルとして実装されることになる。利用者、予約、学習室の 3 クラスについてはこれで特に問題は無いが、営業情報クラスについてはそれでは効率的ではないとの問題がある。休日など特殊な日だけでなく通常の日まで DB 上にレコードが作成されてしまうからである。これに対処するには、通常の営業日を入れるデフォルトテーブルと、例外的な営業日を入れる特殊テーブルの 2 つのテーブルを準備するのが妥当である。そこで営業情報クラスについては RoR の標準の O/R マッピングを適用せず、Ruby で `find_day` と `find_month` というメソッドを持つクラス(Eigyou)を作成した。`find_day` は、図 3 のように指定された日のレコードが特殊テーブルにある場合はそのレコードを返し、無い場合は例外的な営業日ではないと判断してデフォルトの値を返すメソッドである。`find_month` は指定された 1 カ月分のデータを返すメソッドである。

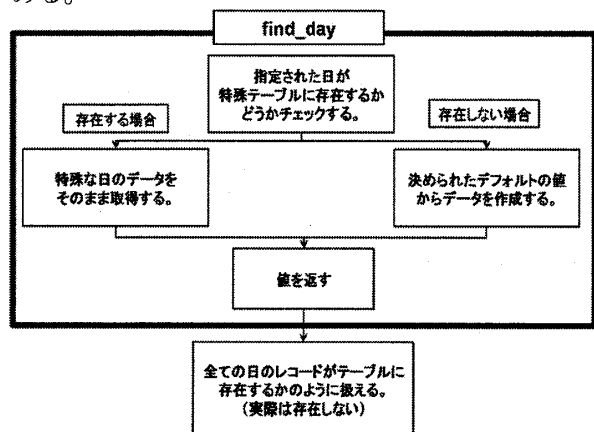


図 3. `find_day` メソッド

完成した Ruby によるモデル図(クラス図)を図 4 に示す。図 4 から分かるように Eigyou クラスを作成したことで UML によるモデル図(図 2)を整合させることができた。

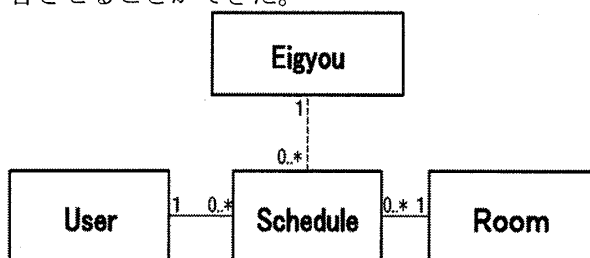


図 4. Ruby によるモデル図(クラス図)

3. 3 RoR のクラス構造

RoR の MVC アーキテクチャに従ってクラス構造をまず設計した。その他のクラス構造の設計方針としては、大まかな機能による分類、一つのクラスの大きさなどを考慮してクラス構造を

決定した。クラス一覧を表 2 に示す。コントローラはデータ処理用、モデルはデータベースに対応するクラスになる。ビューは表示用テンプレートであるため表にはのせていない。

表 2. クラス一覧

種類	クラス名	機能概略
controller	ApplicationController MainController SchedulesController TimesController LoginController RoomsController SpecialdaysController UsersController	全体で共通のエラー処理など ログインしなくても使える機能 部屋予約に関する機能 営業時間を設定 ログイン・ログアウト処理 予約可能部屋の追加・編集など 特殊な日の追加・削除 ユーザ管理
model	Eigyou Room Schedule User	データベースのテーブル に対応するクラス
	Information	メール機能用クラス

3. 4 営業時間変更のユーザインタフェース

この機能のユーザインタフェースは、1 カ月分のデータを一括で更新できるようにとの要望もあり、特別に考える必要があった。要求分析の結果、営業時間は平日、土曜、特殊、の 3 種に分けられることがわかった。なので、1 カ月ごとのカレンダーと、その中の日付ごとにチェックボックスを用意し、チェックした日付を 3 種のどれにするか選ばせるといったようにした。また、カレンダー上での営業時間は色で区別できるようにした。

3. 5 プラグイン利用によるコード簡略化

コードを簡略化し開発を短期間で行うためプラグインを利用した。主に利用したのは、カレンダーを表示する `calendar_helper`、祝日の計算を行う `date2`、エクセル出力を行う `to_xls` となる。

4. まとめ

実装には Ruby 1.8.6、RoR 2.3.4、MySQL 5.1.36-community を使用し、この環境でシステムが稼働することを確認した。機能要求を満たすシステムを開発することができた。RoR を適用することにより 3 カ月程度での開発に成功した。今後の課題は部屋予約時のトランザクション処理や、エラーチェックの強化などが挙げられる。

参考文献

- [1]青木峰郎・後藤裕蔵・高橋征義、Ruby レシピブック、ソフトバンククリエイティブ株式会社、2007 年
- [2]Dave Thomas and David Heinemeier Hansson with Leon Breedt, Mike Clark, Thomas Fuchs, and Andreas Schwarz、Rails によるアジャイル Web アプリケーション開発、オーム社、2006 年