

専攻名	情報アーキテクチャ専攻	必修・選択	選択	単位	2	学期	1Q
科目群	システム開発系科目群	科目名	フレームワーク開発特論			教員名	安川 要平
		(英文表記)	Framework Development				

概要	<p>実用的な Web アプリケーション開発においては、高度なフレームワークを活用することで求めるシステムを迅速に開発し、利用者に提供するためのスキルが求められる。本講義では、代表的な Web アプリケーションフレームワークとして、Ruby on Rails を題材に取り上げる。具体的には、このフレームワークを活用して SNS (Social Networking Service) を作りながら、Web アプリケーションの実践的な開発技術を学ぶ。</p> <p>教材: Ruby on Rails チュートリアル https://railstutorial.jp/</p>			
目的・狙い	<p>本講義ではオブジェクト指向型のプログラミング言語である Ruby と、その代表的なフレームワーク Ruby on Rails を用いて、以下に記述する知識項目と Web アプリケーションを開発するためのスキルセットを体系的に学ぶ。具体的には、フレームワークの特徴を活かした Web アプリケーションの開発や、クラウド IDE (cloud9) を使った開発方法、Git/GitHub を使ったバージョン管理、Heroku を使った本番環境へのデプロイ、Rails の設計思想とリソース設計・実装、Rails を用いた静的/動的ページの作り方、Ruby の設計思想と基本的な構文、Twitter Bootstrap を使ったレイアウトデザイン、モデルによるデータベース操作、MVC アーキテクチャを使ったユーザー登録、セッションを使ったログイン機構、RESTful なリソースと CRUD 処理、Ruby/Rails のエコシステムを活用した画像投稿、多対多のモデリング方法などを取り上げる。</p> <p>修得できる知識単位:</p> <p>(A1) K-02-05-04 開発ツール (レベル 3) 設計ツール、構築ツール、テストツール、言語処理ツール</p> <p>(A4) K-04-12-01 システム要件定義 (レベル 2) システム要件定義、システム要件</p> <p>(A4) K-04-12-02 システム方式設計 (レベル 4) システムの最上位レベルでの方式確立</p> <p>(A4) K-04-12-03 ソフトウェア要件定義 (レベル 3) ユースケース、プロトタイプ、UML</p> <p>(A4) K-04-12-04 ソフトウェア方式設計・ソフトウェア詳細設計 (レベル 3) ソフトウェア構造とコンポーネントの設計、ソフトウェアユニットのテストの設計、ソフトウェア結合テストの設計、ソフトウェア品質、オブジェクト指向設計</p> <p>(A4) K-04-12-05 ソフトウェアコード作成及びテスト (レベル 3) ソフトウェアコード作成、コーディング基準、コーディング支援手法、テスト手法、テスト準備 (テスト環境、テストデータほか)、テストの実施、テスト結果の評価</p> <p>(A4) K-04-12-06 ソフトウェア結合・ソフトウェア適格性確認テスト (レベル 2) テスト計画、テスト準備 (テスト環境、テストデータほか)、テストの実施、テスト結果の評価</p> <p>(A4) K-04-12-07 システム結合・システム適格性確認テスト (レベル 2) テスト計画、テスト準備 (テスト環境、テストデータほか)、テストの実施、テスト結果の評価</p> <p>(A4) K-04-13-01 開発プロセス・手法 (レベル 4) ソフトウェア開発モデル、アジャイル開発、ソフトウェア再利用</p>			
前提知識 (履修条件)	<ul style="list-style-type: none"> ・ Ruby 又は他のオブジェクト指向型プログラミング言語によるプログラミング経験 ・ HTML/CSS/JavaScript に関する基本的な知識、UNIX コマンドに関する基本的な知識 			
到達目標	上位到達目標			
	フレームワークを活用した Web アプリケーション開発を自分で行えるようになること。			
到達目標	最低到達目標			
	フレームワークを活用した Web アプリケーション開発の全体像を理解すること。			
授業の形態	形態	実施	特徴・留意点	
	録画・対面混合授業	◎	本講義期間中は、教材の解説動画 (36 時間) を受講生向けに限定公開する。Web 上の教材と解説動画、及び講義時間中の質疑応答を通して、シラバスに書かれた内容の習得を目指す。シラバス通りに学習できたかどうかは、毎回の講義で出題される演習を通して確認する。	
	対面授業	講義 (双方向)	—	
		実習・演習 (個人)	◎	毎回の講義で演習を出題する。演習のほとんどはプログラミング (実装) となるため、開発環境は初回の講義中又は事前に整えておくこと。本講義では実践的な演習 (プログラミング) とそれに関連する質疑応答を重視しているため、主体的に学ぶ力が強く求められる。プログラミングに苦手意識がある方は、講義時間中の質疑応答を積極的に活用すること。
		実習・演習 (グループ)	—	
サテライト開講授業	—			
その他	—			
授業外の学習	<p>復習として、授業中に示す課題に取り組むこと。予習として、事前に教材を 1 周していることが望ましい。少なくとも、授業内で取り扱うコードは授業前に実装しておくこと (当日講義を聞きながら実装することは困難なため)。</p> <p>参考: 過去の受講生の感想</p> <p>https://docs.google.com/spreadsheets/d/1OI0XxcCYwLN8G91XOT84phtUWdFLfuDjrMM06qtzi-g/</p>			
授業の内容	<p>教材「Ruby on Rails チュートリアル (第 4 版、全 14 章)」のうち、省略可能な章を除いた 11 章分を授業で進める。また、授業で取り扱う 11 章分についても、時間の関係から一部のトピックを省略する場合がある。なお、学習状況が順調であれば、比較的应用的なトピックかつ省略予定の第 9 章 (発展的なログイン機構)、第 11 章 (アカウントの有効化)、第 12 章 (パスワードの再設定) の一部又は全部についても授業内で取り扱う。</p> <p>授業で取り扱う資料の詳細については、下記の「教材・教科書」及び「参考図書」を参照すること。</p>			

	回数	内容	サテライト 開講	対面/録画
授業の計画	第 1 回	ガイドランス、第 1 章 ゼロからデプロイまで ・本授業の目的・目標、開発環境 (Cloud9 など) の準備、Git/Heroku のセットアップ ・演習課題: 開発環境の準備、Git/Heroku のセットアップ、GitHub のセットアップ	—	対面
	第 2 回	第 2 章 Toy アプリケーション ・ Rails の設計思想、Rails を使ったリソース設計と実装デモ	—	対面
	第 3 回	第 3 章 ほぼ静的なページの作成 ・ minitest を使ったテストの書き方、テスト駆動開発、静的/動的なページの生成	—	録画
	第 4 回	第 4 章 Rails 風味の Ruby ・ Ruby の設計思想、Ruby の基本的な構文、Ruby のデータ構造とクラスの実装	—	対面
	第 5 回	第 5 章 レイアウトを作成する。 ・ Twitter Bootstrap を使ったデザインとレイアウト、Sass とアセットパイプライン	—	録画
	第 6 回	第 6 章 ユーザーのモデルを作成する。 ・ Active Record を使ったモデルの実装、バリデーション、セキュアなパスワード	—	対面
	第 7 回	第 7 章 ユーザー登録 ・ Rails のデバッグ方法、MVC アーキテクチャを使ったユーザー登録フォームの実装	—	録画
	第 8 回	第 8 章 基本的なログイン機構 ・ セッションを使ったログイン/ログアウト機能の実装、リファクタリング	—	対面
	第 9 回	第 9 章 ユーザーの更新・表示・削除 (前半) ・ 編集フォームの実装、before アクションを使ったセキュリティモデルの構築	—	録画
	第 10 回	第 10 章 ユーザーの更新・表示・削除 (後半) ・ Rake によるサンプルデータの生成、ページネーション、RESTful なリソースの完成	—	対面
	第 11 回	第 11 章 ユーザーのマイクロポスト (前半) ・ Micropost のモデリング、モデル同士の関連付け、Micropost の MVC 設計と実装	—	録画
	第 12 回	第 12 章 ユーザーのマイクロポスト (後半) ・ Micropost のセキュリティモデルの構築、Ruby Gems を使った画像投稿の実装デモ	—	対面
	第 13 回	第 13 章 ユーザーをフォローする。 (前半) ・ 多対多のモデリングと実装、[Follow] ボタンの基本的な実装とテスト	—	録画
	第 14 回	第 14 章 ユーザーをフォローする。 (後半) ・ [Follow] ボタンの Ajax 実装、ステータスフィールドの実装、応用トピックの紹介	—	対面
	第 15 回	これまでのまとめ、今回取り扱わなかった範囲の紹介、今後の学習方法、実技試験 又はレポートに関する質疑応答など。	—	録画
	試験	授業の理解度を測る実技試験を行う。 ・ レポート提出希望者は、当日の授業が終わるまでにレポートを提出すること。	—	対面
成績評価	次の 2 つのいずれかの方法で評価する。基本的には演習課題と実技試験で評価し、希望する受講生については、上位到達目標 (フレームワークを活用した Web アプリケーション開発を自分で行えるようになること) の達成をレポートで示すことを認める。レポートを提出する場合は、演習課題や実技試験に取り組む必要はない。 ・ 演習課題と実技試験 (第 2 回以降の演習課題については、授業中に適宜指定する。) ・ レポート: サンプルアプリケーションの拡張機能の実装、又は Rails を使った新規 Web アプリケーションの実装 サンプルアプリケーションの拡張機能の例: 返信機能、メッセージ機能、フォローの通知、検索機能、など			
教科書・教材	教員作成スライド資料: http://www.slideshare.net/yasulab/presentations 『Ruby on Rails チュートリアル: 実例を使って Rails を学ぼう (第 4 版)』Michael Hartl 著、安川 要平・八田 昌三 訳 https://railstutorial.jp/ Web 版は無料。有償の解説動画があるが、本講義のために購入する必要はない。			
参考図書	各章ごとのサンプルコード集: https://github.com/yasslab/sample_apps (デバッグなどで活用)、各種 Web サイト、『Ruby on Rails ガイド: 体系的に Rails を学ぼう』Ruby on Rails Community 著、八田 昌三・安川 要平・他 訳 https://railsguides.jp/ Web 版は無料。有償の電子書籍があるが、本講義のために購入する必要はない。			