

Rubyによるアジャイル開発 および Ruby on rails によるアジャイル開発

カリキュラムとシラバス

平成29年2月

学校法人吉田学園
吉田学園情報ビジネス専門学校

■カリキュラムとスケジュール

			月曜日 1月16日	火曜日 1月17日	水曜日 1月18日	木曜日 1月19日	金曜日 1月20日
1	9:30 - 11:30	内容	1. Rubyを用いたアジャイル開発 1-1. アジャイル開発基礎 ①アジャイル開発プロセス (1) * スクラム/XP概要 * ブラックスや開発の流れ * ふりかえり	1. Rubyを用いたアジャイル開発 1-2. Ruby基礎 ①おさらい (1)	1. Rubyを用いたアジャイル開発 1-3. Rubyによるボウリングプログラム ②ボウリングプログラムの作成 (1) * ボウリングの基本的な概念を作る * ボーナス計算の組み込みでリアクティング	2. Ruby on Railsを用いたアジャイル開発 2-1. Rails基礎(1コマ) * Railsの設計思想など * テストについても触れておく	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ② 管理画面の作成 (1) * デザインテンプレートの適用 * 商品マスタの作成 * テスト修正
昼休み							
2	12:30 - 14:30	内容	1. Rubyを用いたアジャイル開発 1-1. アジャイル開発基礎 ①アジャイル開発プロセス (2) * スクラム/XP概要 * ブラックスや開発の流れ * ふりかえり	1. Rubyを用いたアジャイル開発 1-2. Ruby基礎 ①おさらい (2)	1. Rubyを用いたアジャイル開発 1-3. Rubyによるボウリングプログラム ②ボウリングプログラムの作成 (2) * ボウリングの基本的な概念を作る * ボーナス計算の組み込みでリアクティング	2. Ruby on Railsを用いたアジャイル開発 2-2. Railsアプリケーションの作成 (1) * Cloud9上でRailsアプリケーションを作成 * Railsの基本 * 簡単なTODOとかBlog等を作成する	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ② 管理画面の作成 (2) * デザインテンプレートの適用 * 商品マスタの作成 * テスト修正
3	14:30 - 16:30	内容	1. Rubyを用いたアジャイル開発 1-1. アジャイル開発基礎 ②コード管理基礎 * Git基礎 * GitHub基礎	1. Rubyを用いたアジャイル開発 1-2. Ruby基礎 ②おさらいのテストプログラミング	1. Rubyを用いたアジャイル開発 1-3. Rubyによるボウリングプログラム ②ボウリングプログラムの作成 (3) * ボウリングの基本的な概念を作る * ボーナス計算の組み込みでリアクティング	2. Ruby on Railsを用いたアジャイル開発 2-2. Railsアプリケーションの作成 (2) * Cloud9上でRailsアプリケーションを作成 * Railsの基本 * 簡単なTODOとかBlog等を作成する	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ③ 管理者ログイン認証の作成 (1) * ユーザ管理画面の作成 * Gemを利用したログイン認証 * ログイン認証必須に伴うテスト修正 * 管理ユーザ/一般ユーザのロール権限作成
4	16:30 - 18:30	内容	1. Rubyを用いたアジャイル開発 1-1. アジャイル開発基礎 ③テスト基礎 * テスト駆動開発とは * Test::Unit/Rspec	1. Rubyを用いたアジャイル開発 1-3. Rubyによるボウリングプログラム ①アジャイルによる計画 * プログラミングツールの作成 * スプリントツールの作成 * プランニングポーカーによる見積り	予備	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ①バックログの作成 * バックログの作成と見積り	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ③ 管理者ログイン認証の作成 (2) * ユーザ管理画面の作成 * Gemを利用したログイン認証 * ログイン認証必須に伴うテスト修正 * 管理ユーザ/一般ユーザのロール権限作成
時間			月曜日 1月23日	火曜日 1月24日	水曜日 1月25日	木曜日 1月26日	金曜日 1月27日
1	9:30 - 11:30	内容	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ③ 管理者ログイン認証の作成 (3) * ユーザ管理画面の作成 * Gemを利用したログイン認証 * ログイン認証必須に伴うテスト修正 * 管理ユーザ/一般ユーザのロール権限作成	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-2. アプリケーションの公開 ①Seedデータの作成	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ②要求変更の対応 (1) * 音楽商品の取扱い追加	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ④管理画面: オーダー一覧 (1) * 注文状況の確認 * 注文ステータス(発送済み等)の管理 * 状態管理Gemの利用	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ⑥. 各種検索機能の追加 * 商品の検索 * オーダー情報の検索
昼休み							
2	12:30 - 14:30	内容	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ④商品一覧ページの作成 (1) * 商品を一覧表示する画面 * テストコード作成	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-2. アプリケーションの公開 ②Herokuへの公開	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ②要求変更の対応 (2) * 音楽商品の取扱い追加	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ④管理画面: オーダー一覧 (2) * 注文状況の確認 * 注文ステータス(発送済み等)の管理 * 状態管理Gemの利用	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ⑦画面のAjax化 (1) * CoffeeScript基礎 * 購入処理のAjax化 * 注文ステータス管理のAjax化
3	14:30 - 16:30	内容	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-1. 1stリリース ④商品一覧ページの作成 (2) * 商品を一覧表示する画面 * テストコード作成	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ①バックログの作成 * 要求の変更 * バックログの作成と見積り	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ③注文画面の作成 (1) * 商品のオーダー情報モデルの作成 * オーダー画面の作成 * メール送信	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ⑤ カート機能の追加 (1) * 購入処理の変更 * オーダー管理画面の変更	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ⑦画面のAjax化 (2) * CoffeeScript基礎 * 購入処理のAjax化 * 注文ステータス管理のAjax化
4	16:30 - 18:30	内容	予備	予備	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ③注文画面の作成 (2) * 商品のオーダー情報モデルの作成 * オーダー画面の作成 * メール送信	2. Ruby on Railsを用いたアジャイル開発 2-3. ECサイト アプリケーションの作成 2-3-3. 2ndリリース ⑤ カート機能の追加 (2) * 購入処理の変更 * オーダー管理画面の変更	まとめ ふりかえり

第1コマ	
タイトル	アジャイル開発基礎 アジャイル開発プロセス（1）
目標	アジャイル開発についての基礎知識を習得する
概要	1. アジャイル開発の起源についての紹介 2. アジャイルソフトウェア開発宣言およびその背後にある原則の紹介 3. ウォーターフォール型開発とアジャイル開発の違いについて解説 4. エクストリームプログラミング、スクラムの概要説明
座学・演習	座学
使用教材	スライド
事前学習	なし
宿題	なし
特記事項	なし
所用時間	120分(休憩時間20分含む)

学習達成度評価指標	
0～20分	アジャイル開発の起源とアジャイルが解決しようとしている問題について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
20～50分	アジャイルソフトウェア開発宣言、アジャイル宣言の背後にある原則について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～90分	ウォーターフォール型開発とアジャイル開発の違いについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
90～110分	アジャイル開発の代表的な手法であるエクストリームプログラミングとスクラムの概要 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第2コマ	
タイトル	アジャイル開発基礎 アジャイル開発プロセス（2）
目標	アジャイル開発についての基礎知識を習得する
概要	1. エクストリームプログラミングのプラクティスの説明 2. スクラムのイベントの説明 3. アジャイル開発でよく用いられる、その他のプラクティスの紹介 4. ふりかえりの目的と手法の紹介
座学・演習	座学
使用教材	スライド
事前学習	なし
宿題	なし
特記事項	なし
所用時間	120分(休憩時間20分含む)

学習達成度評価指標	
0～40分	エクストリームプログラミングの各プラクティスの目的と効果について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
40～50分	スクラムの各種イベントについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～90分	アジャイル開発でよく用いられるプラクティスについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
90～110分	ふりかえりの目的とよく用いられる手法について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第3コマ	
タイトル	コード管理基礎
目標	Git・GitHubに関する基礎的な知識を身に付け、ソースコードのバージョン管理方法を理解する
概要	1. Gitに関する説明・解説 2. GitHubに関する説明・解説
座学・演習	座学
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～55分	Gitの基本的な知識について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～115分	GitHubの基本的な知識について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第4コマ	
タイトル	テスト基礎
目標	テスト駆動開発の概念を理解し、実際のテストツールの使い方を学ぶ。
概要	1. テスト駆動開発についての説明・解説 2. 使用するテストツールの説明・解説
座学・演習	座学
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～55分	テスト駆動開発の考え方や概念について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～115分	使用するテストツールについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第5コマ	
タイトル	Ruby基礎 - おさらい(1)
目標	Rubyの基本的な文法や、それぞれの使い方を再確認する
概要	1. Rubyの基礎や文法を説明・解説
座学・演習	座学
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～55分	1. 変数 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 2. 定数 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 3. 真偽値 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～115分	4. 配列 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 5. ハッシュ A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 6. 演算子 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第6コマ	
タイトル	Ruby基礎 - おさらい(2)
目標	Rubyの基本的な文法や、それぞれの使い方を再確認する
概要	1. Rubyの基礎や文法を説明・解説
座学・演習	座学
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～55分	7. 条件分岐 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 8. 繰り返し A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 9. メソッド A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～115分	10. クラス A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 11. モジュール A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第7コマ	
タイトル	Ruby基礎 - おさらいのテストプログラミング
目標	解説したRubyの基本的な文法を用いてプログラミングを行い、知識がどれだけ身に付いたかを確認する
概要	1.Rubyでの簡単なプログラミング ・課題（Q1～Q5）の提示と、それに対するプログラミング
座学・演習	演習
使用教材	各自ノートPC（Windows, Mac）
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分（休憩10分含む）

学習達成度評価指標	
0～55分	Q1のプログラミングについて A：完全にできた B：大体できた C：あまりできなかった D：できなかった Q2のプログラミングについて A：完全にできた B：大体できた C：あまりできなかった D：できなかった Q3のプログラミングについて A：完全にできた B：大体できた C：あまりできなかった D：できなかった
60～115分	Q4のプログラミングについて A：完全にできた B：大体できた C：あまりできなかった D：できなかった Q5のプログラミングについて A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第8コマ	
タイトル	Rubyによるボウリングプログラム - アジャイルによる計画
目標	アジャイル開発での計画の立て方を知る
概要	1. 不確実性コーンの紹介 2. 計画を立てる際に使われるプラクティス 3. ボウリングプログラムのバックログ作成 4. プランニングポーカーを使用したバックログの見積もり
座学・演習	座学と演習
使用教材	スライド、付箋、ペン、プランニングポーカー
事前学習	なし
宿題	なし
特記事項	バックログの作成と見積もりは、5名前後のチームになって実施する
所用時間	120分(休憩時間20分含む)

学習達成度評価指標	
0～10分	計画の不確実性を知り、計画の継続的な見直しについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
10～40分	アジャイル開発で計画を立てる際に使われるプラクティスについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
50～80分	ボウリングプログラムにどんな要件があるかを考え話し合いながら、バックログを作る A：完全にできた B：大体できた C：あまりできなかった D：できなかった
80～110分	相対見積もりの利点について理解し、プランニングポーカーを用いて見積もる A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第9コマ	
タイトル	Rubyによるボウリングプログラム - ボウリングプログラムの作成（1）
目標	実際にプログラミングを行い、その過程でこれまでに学習した内容を取り入れることで、実戦的な能力を身に付ける。
概要	<ol style="list-style-type: none"> 1. テスト駆動開発の流れを再確認 2. Git・GitHubの準備 3. ボウリングプログラムの基礎部分を作成
座学・演習	演習
使用教材	各自ノートPC（Windows, Mac）
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分（休憩10分含む）

学習達成度評価指標	
0～30分	テスト駆動開発の流れを再確認 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
30～55分	GitとGitHubの準備 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
60～115分	ボウリングプログラムの基礎部分を作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第10コマ	
タイトル	Rubyによるボウリングプログラム - ボウリングプログラムの作成（2）
目標	実際にプログラミングを行い、その過程でこれまでに学習した内容を取り入れることで、実戦的な能力を身に付ける。
概要	1. スペアを取った時のボーナスを計算する機能を追加 2. ストライクを取った時のボーナスを計算する機能を追加
座学・演習	座学と演習
使用教材	各自ノートPC（Windows, Mac）
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分（休憩10分含む）

学習達成度評価指標	
0～55分	スペアを取った時のボーナスを計算する機能を追加<演習> A：完全にできた B：大体できた C：あまりできなかった D：できなかった
60～90分	実習内容の解説について<座学> A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
90～115分	ストライクを取った時のボーナスを計算する機能を追加<演習> A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第11コマ	
タイトル	Rubyによるボウリングプログラム - ボウリングプログラムの作成 (3)
目標	実際にプログラミングを行い、その過程でこれまでに学習した内容を取り入れることで、実戦的な能力を身に付ける。
概要	1. ストライクを取った時のボーナスを計算する機能を追加 2. フレーム毎の合計を計算する機能を追加
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～30分	ストライクを取った時のボーナスを計算する機能を追加<演習> A：完全にできた B：大体できた C：あまりできなかった D：できなかった
30～55分	実習内容の解説について<座学> A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～100分	フレーム毎の合計を計算する機能を追加<演習> A：完全にできた B：大体できた C：あまりできなかった D：できなかった
100～115分	実習内容を解説について<座学> A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第12コマ	
タイトル	Rails基礎
目標	Ruby on Railsに関する基本的な知識や、アプリケーションに対するテストを行う際の注意点などを学ぶ
概要	1. Railsについての解説 2. Railsでのテストについての解説
座学・演習	座学
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～55分 60～80分	Railsについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
80～115分	Railsでのテストについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第13コマ

タイトル	Railsアプリケーションの作成（1）
目標	Ruby on Railsで実際に簡単なアプリケーションを作成し、Railsアプリの構造や仕組みを理解する
概要	1. Cloud9上でRailsアプリケーションを作成
座学・演習	座学と演習
使用教材	各自ノートPC（Windows, Mac）
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分（休憩10分含む）

学習達成度評価指標

0～55分	Cloud9上でRailsアプリケーションを作成<演習> A：完全にできた B：大体できた C：あまりできなかった D：できなかった
60～115分	Railsのファイル構造の解説について<座学> A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第14コマ	
タイトル	Railsアプリケーションの作成（2）
目標	Ruby on Railsで実際に簡単なアプリケーションを作成し、Railsアプリの構造や仕組みを理解する
概要	1. 簡単な自作の機能を追加する
座学・演習	座学と演習
使用教材	各自ノートPC（Windows, Mac）
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分（休憩10分含む）

学習達成度評価指標	
0～55分	ユーザの登録などの簡単な機能を追加<演習> A：完全にできた B：大体できた C：あまりできなかった D：できなかった
60～115分	Railsの動作の原理などの解説について<座学> A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第15コマ	
タイトル	ECサイト アプリケーションの作成 1stリリース バックログの作成
目標	ECサイトのバックログを作成する
概要	1. ECサイトのプロダクトバックログの作成 2. プランニングポーカーを使用したプロダクトバックログの見積もり 3. スプリントバックログの作成と工数見積もり
座学・演習	座学と演習
使用教材	スライド、付箋、ペン、プランニングポーカー
事前学習	なし
宿題	なし
特記事項	バックログの作成と見積もりは、5名前後のチームになって実施する
所用時間	120分(休憩時間20分含む)

学習達成度評価指標	
0～60分	ECサイトのプロダクトバックログの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった 顧客やチームで会話しながら共通認識を持つことの重要性について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
70～110分	プランニングポーカーを使用したプロダクトバックログの見積もり A：完全にできた B：大体できた C：あまりできなかった D：できなかった スプリントバックログの作成と工数見積もり A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第16コマ	
タイトル	ECサイト アプリケーションの作成 1stリリース 管理画面の作成 デザインテンプレートの適用
目標	ECサイトのデータベースを設計し、管理画面のデザインテンプレートを 適応する
概要	1. データベース設計 2. プロジェクトの作成 3. MVCの作成 4. データベースの作成 5. デザインテンプレート(bootstrap)の準備 6. デザインの適用
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成 したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～30分	データベース設計について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～60分	プロジェクトの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった MVCの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

60～120分	<p>データベースの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p> <p>デザインテンプレート(bootstrap)の準備 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p> <p>デザインの適用 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p>
---------	---

第17コマ	
タイトル	ECサイト アプリケーションの作成 1stリリース 管理画面の作成 商品マスタの作成、テスト修正
目標	ECサイトの商品マスタを作成し、rspecテストを修正する
概要	<ol style="list-style-type: none"> 1. 今回のアプリの画面遷移とルーティングについて Railsの動作モードとseeds機能について 2. 国際化 (I18n) とタイムゾーンの設定 3. 商品マスタの作成 4. 画面遷移とルーティングの設定 5. リダイレクトの変更 6. 中間テーブルの実装 7. テストデータとrspecの実装準備とテスト修正
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0~50分	<p>アプリの画面遷移とルーティングについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった</p> <p>Railsの動作モードとseeds機能について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった</p>

50～90分	<p>国際化 (I18n) とタイムゾーンの設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p> <p>商品マスタの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p> <p>画面遷移とルーティングの設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p> <p>リダイレクトの変更 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p>
90～100分	<p>中間テーブルの実装 A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p>
100～120分	<p>テストデータとrpsecの実装準備とテスト修正のプログラミングについて A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p>

第18コマ

タイトル	ECサイト アプリケーションの作成 1stリリース 管理者ログイン認証の作成 Gemを利用したログイン認証
目標	ECサイトのログイン認証としてgem deviseを適応する
概要	1. gem deviseの主な機能とカスタム方法について 2. gem deviseのインストールとユーザーモデルの設定 3. コントローラへの認証設定 4. ログイン画面等のカスタマイズ
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～60分	gem deviseの主な機能とカスタム方法について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～120分	gem deviseのインストールとユーザーモデルの設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった コントローラへの認証設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった ログイン画面等のカスタマイズ A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第19コマ	
タイトル	ECサイト アプリケーションの作成 1stリリース 管理者ログイン認証の作成 ユーザ管理画面の作成、テスト修正
目標	ユーザ管理画面の作成と、ログイン認証必須に伴うテストの修正を行う
概要	1. ユーザモデルの共有方法と注意点について 2. ユーザ管理画面の作成 3. ユーザ管理画面のデザイン適応 4. gem deviseおよびユーザ管理画面のテスト修正
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～50分	ユーザモデルの共有方法と注意点について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
60～120分	ユーザ管理画面の作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった ユーザ管理画面のデザイン適応 A：完全にできた B：大体できた C：あまりできなかった D：できなかった gem deviseおよびユーザ管理画面のテスト修正 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第20コマ	
タイトル	ECサイト アプリケーションの作成 1stリリース 管理者ログイン認証の作成 管理/一般ユーザのロール権限作成
目標	管理ユーザ、一般ユーザを区別するためのロール権限を追加する
概要	1. 既存テーブルへのフィールドと追加と設定の追加について 2. ユーザーモデルへのフィールド追加 3. ユーザ管理画面の修正 4. 管理者ログイン認証の作成についてのまとめと質疑応答
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～50分	既存テーブルへのフィールドと追加と設定の追加について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった ユーザーモデルへのフィールド追加 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
60～100分	ユーザ管理画面の修正 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
100～120分	管理者ログイン認証の作成についてのまとめと質疑応答 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第21コマ	
タイトル	ECサイト アプリケーションの作成 1stリリース 商品一覧ページの作成 商品を一覧表示する画面
目標	商品閲覧するための商品一覧ページを作成する
概要	1. 認証ページの構成とデザインレイアウトを分割する方法について 2. 商品一覧ページのルーティングとコントローラの作成 3. 商品一覧ページのデザインの適応
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～40分	認証ページの構成とデザインレイアウトを分割する方法について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
40～120分	商品一覧ページのルーティングとコントローラの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった 商品一覧ページのデザインの適応 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第22コマ	
タイトル	ECサイト アプリケーションの作成 1stリリース 商品一覧ページの作成 テストコード作成
目標	商品一覧ページのテストコードの作成と、商品の表示・非表示用のメソッドを追加する
概要	1. コントローラでのデータ操作とモデルへのリファクタリングについて 2. 商品一覧ページのコントローラテストの作成と修正 3. 商品一覧の表示・非表示の機能実装 4. テスト全般について修正箇所の確認 5. アプリ全体の動作確認と不具合修正
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～50分	コントローラでのデータ操作とモデルへのリファクタリングについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった 商品一覧ページのコントローラテストの作成と修正 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
50～90分	商品一覧の表示・非表示の機能実装 A：完全にできた B：大体できた C：あまりできなかった D：できなかった テスト全般について修正箇所の確認 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

90～120分

アプリ全体の動作確認と不具合修正

A：完全にできた B：大体できた

C：あまりできなかった D：できなかった

第23コマ

タイトル	ECサイト アプリケーションの作成 アプリケーションの公開 Seedデータの作成
目標	Herokuへリリースするための準備を行う
概要	1. Heroku構成について 2. Herokuアカウントの作成とアプリ側のHeroku対応設定 3. インフラおよびデプロイの質疑応答
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～20分	Heroku構成について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
20～60分	Herokuアカウントの作成とアプリ側のHeroku対応設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	インフラおよびデプロイの質疑応答 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第24コマ	
タイトル	ECサイト アプリケーションの作成 アプリケーションの公開 Herokuへの公開
目標	Herokuへリリースを行う
概要	1. 運用で利用するHerokuコマンドと注意点について 2. Herokuコマンドでのデプロイ 3. 1stリリース全体のまとめと質疑応答
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～30分	運用で利用するHerokuコマンドと注意点について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～60分	Herokuコマンドでのデプロイ A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	1stリリース全体のまとめと質疑応答 A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった

第25コマ	
タイトル	ECサイト アプリケーションの作成 2ndリリース バックログの作成
目標	顧客からの要求の変更とアジャイル開発での適応方法を知る
概要	1. 変更された要求の確認 2. プロダクトバックログの見積もり 3. スプリントバックログの作成と見積もり
座学・演習	座学と演習
使用教材	スライド、付箋、ペン、プランニングポーカー、タスクボード、バーン ダウンチャート
事前学習	なし
宿題	なし
特記事項	バックログの作成と見積もりは、5名前後のチームになって実施する
所用時間	120分(休憩時間20分含む)

学習達成度評価指標	
0～20分	変化に適応することの重要性と、アジャイル開発での変化への適応方法について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
20～60分	プロダクトバックログの見積もり A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～110分	スプリントバックログの作成と工数見積もり A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第26コマ	
タイトル	ECサイト アプリケーションの作成 2ndリリース 要求変更の対応（1）
目標	STI(単一テーブル継承)を理解し、テーブル作成・ECサイトの修正をする
概要	1. STI（単一テーブル継承） 2. テーブルの作成とデータ移行 3. 既存ソースの修正 4. rspecでの既存画面の動作テスト
座学・演習	座学と演習
使用教材	各自ノートPC（Windows, Mac）
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分（休憩10分含む）

学習達成度評価指標	
0～30分	STI（単一テーブル継承）について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～60分	テーブルの作成とデータ移行 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	既存ソースの修正 A：完全にできた B：大体できた C：あまりできなかった D：できなかった rspecでの既存画面の動作テスト A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第27コマ	
タイトル	ECサイト アプリケーションの作成 2ndリリース 要求変更の対応 (2)
目標	STI(単一テーブル継承)を理解し、ECサイトで音楽商品を扱えるようにする
概要	1. 音楽商品の管理画面作成 2. 商品一覧画面に音楽商品を表示 3. テストデータの準備
座学・演習	演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～30分	音楽商品の管理画面作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
30～60分	商品一覧画面に音楽商品を表示 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	テストデータの準備 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第28コマ

タイトル	ECサイト アプリケーションの作成 2ndリリース 注文画面の作成(1)
目標	商品一覧画面から商品を選択し、注文できるようにする。
概要	1. テーブルの設計 2. テーブルの作成 3. 注文画面の作成
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～30分	テーブルの設計について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～60分	テーブルの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	注文画面の作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第29コマ	
タイトル	ECサイト アプリケーションの作成 2ndリリース 注文画面の作成(2)
目標	商品を注文した際に、メールが送信されるようにする
概要	1. ActionMailerの仕組み 2. Mailerクラスの作成 3. ActionMailerの設定 4. 動作環境に応じた環境変数の設定
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～30分	ActionMailerの仕組みについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～60分	Mailerクラスの作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	ActionMailerの設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった 動作環境に応じた環境変数の設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第30コマ

タイトル	ECサイト アプリケーションの作成 2ndリリース 管理画面: オーダー一覧(1)
目標	注文された内容を管理画面で確認できるようにする
概要	1. Ordersテーブルの変更 2. Railsにおけるenum 3. 注文一覧画面の作成
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～30分	Ordersテーブルの変更について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～60分	Railsにおけるenumについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
70～120分	注文一覧画面の作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第31コマ

タイトル	ECサイト アプリケーションの作成 2ndリリース 管理画面: オーダー一覧(2)
目標	注文された内容のステータス管理が行えるようになる
概要	1. 状態遷移とaasm 2. aasmの設定 3. 注文管理画面の作成
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～30分	状態遷移とaasmについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～60分	aasmの設定 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	注文管理画面の作成 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第32コマ

タイトル	ECサイト アプリケーションの作成 2ndリリース カート機能の追加 購入処理の変更 オーダー管理画面の変更
目標	セッション機能を理解して、カート機能の作成を行う
概要	1. セッション機能 2. カートのセッション管理機能の準備 3. セッション登録機能の追加 4. カート登録機能の追加 5. カート削除機能の追加
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～30分	セッション機能について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった カートのセッション管理機能の準備 A：完全にできた B：大体できた C：あまりできなかった D：できなかった セッション登録機能の追加 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
30～60分	カート登録機能の追加 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	カート削除機能の追加 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第33コマ

タイトル	ECサイト アプリケーションの作成 2ndリリース カート機能の追加 購入処理の変更 オーダー管理画面の変更
目標	変更対応要求に対して、機能変更対応方法を学ぶ
概要	1. 複数明細の購入処理の準備 2. 利用者画面の複数明細の購入処理への変更 3. オーダー管理画面の複数明細管理への変更
座学・演習	演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～30分	複数明細の購入処理の準備 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
30～60分	利用者画面の複数明細の購入処理への変更 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	オーダー管理画面の複数明細管理への変更 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第34コマ

タイトル	ECサイト アプリケーションの作成 2ndリリース 各種検索機能の追加 商品検索の検索 オーダー情報の検索
目標	gemを利用して検索機能を学ぶ
概要	1. ransack (gem) の仕組み・使い方 2. ransackの準備 3. 商品検索の対応 4. オーダー情報検索
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～30分	ransack (gem) の仕組み・使い方について A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった ransackの準備 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
30～60分	商品検索の対応 A：完全にできた B：大体できた C：あまりできなかった D：できなかった
70～120分	オーダー情報検索 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第35コマ

タイトル	ECサイト アプリケーションの作成 2ndリリース 画面のAjax化 CoffeeScript基礎
目標	CoffeeScriptの使い方を学ぶ
概要	1. CoffeeScriptとJavaScriptの違い 2. CoffeeScriptの文法基礎
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標

0～30分	CoffeeScriptとJavaScriptの違いについて A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった
30～120分	CoffeeScriptの文法基礎 A：完全にできた B：大体できた C：あまりできなかった D：できなかった

第36コマ	
タイトル	ECサイト アプリケーションの作成 2ndリリース 画面のAjax化 購入処理のAjax化 注文ステータス管理のAjax化
目標	Ajaxを利用して、同期と非同期の使い方を学ぶ
概要	1. Ajax 2. HTTP 3. 同期と非同期の違い 4. カート追加を非同期に変更 5. オーダー管理の非同期のステータス変更を機能追加
座学・演習	座学と演習
使用教材	各自ノートPC (Windows, Mac)
事前学習	Ruby初級セミナー修了と同等の知識があり、Railsで1度はアプリを作成したことがあること
宿題	なし
特記事項	インターネットへ接続できる環境であること
所用時間	120分 (休憩10分含む)

学習達成度評価指標	
0～30分	<p>Ajaxについて</p> <p>A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった</p> <p>HTTPについて</p> <p>A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった</p> <p>同期と非同期の違いについて</p> <p>A：十分理解できた B：大体理解できた C：少し理解できた D：理解できなかった</p>
30～60分	<p>カート追加を非同期に変更</p> <p>A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p>
70～120分	<p>オーダー管理の非同期のステータス変更を機能追加</p> <p>A：完全にできた B：大体できた C：あまりできなかった D：できなかった</p>