

期間	受講生
約 2 営業日	<p>【チーム制作準備】</p> <p><u>機能設計書、テーブル設計書を使って、プログラムする各機能と繋がりを持握する</u></p> <p>※設計書に書かれた機能通りとなるようプログラムしてゆく</p> <p>※設計書のうち、記載されたカテゴリの 2 ～ 4 は変更してもよい</p> <p>⇒カテゴリの 1、即ち、すべてのカテゴリは変更NG</p> <p>※設計書に全く追加して良い機能は管理者用機能</p> <p>※これ以外に追加する場合は、講師と相談の上、更に 1 機能のみ追加しても良い</p> <p>⇒追加したい機能がある場合は、このタイミングで講師と可否を相談する</p> <p>※この時点で追加可否と作成する機能が決定</p> <p>※機能実装が開始した後で機能の追加や変更はNG</p> <p>※ただし講師が機能実装コードレビュー時、機能実装及びチームサイト制作が間に合わないと判断した場合には機能を削除することも有りえる</p> <p><u>各機能及び画面の担当分担</u></p> <p>※機能単位で担当分担を割り振る</p> <p><u>制作スケジュールの作成</u></p> <p>※割り振った担当と各機能に紐づく処理と画面のスケジュールを作成</p> <p>⇒○○入力を 1 日、○○確認を 2 日、○○完了を 2 日のような粒度にする</p> <p>○○入力、○○確認、○○完了いずれも 5 日のような粒度にしない</p> <p><u>チームサイトのプロジェクト名を決定</u></p> <p>※プロジェクト名が決定したら、講師にリポジトリ作成依頼</p> <p><u>環境構築</u></p> <p>※作成されたリポジトリを自身のworkspaceへクローンする</p> <p>(testリポジトリ内はNG)</p>
約7営業日	<p>【機能実装】</p> <p><u>機能実装のプログラム</u></p> <p>以下をスケジュールに従ってプログラムしてゆく</p> <p>※機能毎に担当範囲の画面実装をプログラムしてゆく</p> <p>web.xml作成</p> <p>struts.xml作成</p> <p>Actionクラス作成</p> <p>DAOクラス作成</p> <p>DTOクラス作成</p> <p>JSP作成 (必要な値、プルダウン、チェックボックス、ラジオボタン等を表示する程度、CSSを使ったデザインプログラムは行わない)</p> <p>【機能実装のコードレビュー依頼】</p> <p><u>レビュー依頼</u></p> <p>チームサイトの機能実装が終わったら、チームリーダーは講師にコードレビューを依頼する</p> <p>※設計書通りすべての機能がプログラムできていること</p> <p>※すべての機能について動作確認できること</p> <p>以下の 2 つを並行して進めてゆく</p> <p>【レイアウト検討と可視化】</p> <p>・紙やモックアップツールに各画面のレイアウトを具体化して画面イメージを描いてゆく</p> <p>⇒ここでCSSのpxや%なども決めておく</p> <p>⇒各画面のレイアウト資料が完成したら、講師にレイアウト会議を依頼する</p> <p>【機能実装コードレビューの修正と再確認依頼】</p> <p>・コードレビューの結果、修正箇所があった場合は修正をおこなう</p> <p>・チームで修正箇所がすべて直せたら、コードレビューの再確認を依頼する</p> <p>【レイアウト会議】</p> <p>設計書と紙やモックアップツールで作成した画面イメージをもとに講師に説明する</p> <p>・設計書 (画面遷移図) 通りの表示や設計書の動作ができる画面構成や画面遷移のデザインになっているか</p> <p>・画面実装期間で作成可能なレイアウト構成になっているか</p> <p>・難しい動作 (JavaScriptやjQueryを使った難易度の高いプログラムが必要になる箇所) や画面遷移や値の受け渡しで矛盾している箇所はないかなど</p>

約 5 営業日	<p>【画面実装】</p> <p>機能実装のプログラム</p> <p>以下をスケジュールに従ってプログラムしてゆく</p> <p>※機能毎に担当範囲の画面実装をプログラムしてゆく</p> <ul style="list-style-type: none">・ JSP（CSSやJavaScriptを反映する為のプログラム）・ CSS・ JavaScript・ jQuery
	<p>【画面実装のコードレビュー依頼】</p> <p><u>レビュー依頼</u></p> <p>チームサイトの画面実装が終わったら、チームリーダーは講師にコードレビューを依頼する</p> <p>※レイアウト会議時の仕様通り画面実装されていること</p> <p>※設計書通りすべての機能が実行できること</p>
	<p>【画面実装コードレビューの修正と再確認依頼】</p> <ul style="list-style-type: none">・ コードレビューの結果、修正箇所があった場合は修正をおこなう・ チームで修正箇所がすべて直せたら、コードレビューの再確認を依頼する
	<p>【試験準備】</p> <p>⇒チームメンバーのうち3割り程度の方は、単体試験を進めて、完了次第シナリオ試験に合流する</p> <p>⇒チームメンバーのうち7割り程度の方は、先行してシナリオ試験を進めてバグ修正</p> <p>⇒チーム全員でアドホック試験をおこない、単体試験やシナリオ試験と並行してバグ修正をおこなってゆく。</p>
約3営業日	<p>【単体試験】</p> <p>※別紙、「単体試験の手順」資料を参照。</p> <p><u>テストコードプログラムの作成</u></p> <p><u>単体試験テストプログラムの作成完了</u></p> <p><u>JUNITツールを使って単体テスト実施</u></p> <p><u>すべての単体試験項目がOK（緑ゲージ）になったら、単体試験報告書を作成する</u></p> <p><u>単体試験報告書を講師に提出とチェック依頼</u></p> <p><u>・単体試験完了</u></p> <p>※単体試験が終わったら、シナリオ試験の消化を進めてゆく・</p>
約5営業日	<p>【シナリオ試験】</p> <p>※別紙、「シナリオ試験の手順」資料を参照。</p> <p><u>シナリオ試験の実施</u></p> <ul style="list-style-type: none">・ WARファイルを作成する（リーダーもしくはメンバー）・ WARファイルとSQLファイルをマイドライブに保存する（リーダーもしくはメンバー）・ 講師にAWSへアップロードを依頼する（リーダーもしくはメンバー）・ AWSにアップロードされたチームサイトのURLをWebブラウザで開く・ シナリオ試験項目書をもとに試験を進めてゆく <p>各試験結果がOKならばチェック欄に○をつけてゆく</p> <p>各試験結果がNGならばチェック欄に×をつけてプログラムを修正する</p> <p>（NGの場合は、WARファイルとSQLファイルの再アップロード依頼をする）</p> <ul style="list-style-type: none">・ シナリオ試験項目書をもとに再度試験をおこなう<ul style="list-style-type: none">× だった項目がOKならば再チェック欄に○をつけてゆく <p>※すべてがOKになるまでこれを繰り返す</p> <p><u>シナリオ試験項目書（試験済みとなったシート）の内容確認</u></p> <ul style="list-style-type: none">・ シナリオ試験項目書を見直して、漏れがないかを確認する <p><u>シナリオ試験項目書（試験済みとなったシート一式）を講師に提出とチェック依頼</u></p>

	<u>シナリオ試験完了</u>
約5営業日	<p>【アドホック試験】</p> <p>※別紙、「アドホック試験の手順」資料を参照。</p> <p><u>アドホック試験の実施</u></p> <ul style="list-style-type: none">・WARファイルを作成する（リーダーもしくはメンバー）・WARファイルとSQLファイルをマイドライブに保存する（リーダーもしくはメンバー）・講師にAWSへアップロードを依頼する（リーダーもしくはメンバー）・AWSにアップロードされたチームサイトのURLをWebブラウザで開く・決められた日時を使ってアドホック試験をおこなう・<u>バグを発見したら、アドホック試験バグ報告書</u>に記入してゆく・アドホック試験が終了したら、チームサイトで挙がったバグを修正する <p>※すべてがOKになるまで毎日これを繰り返す</p> <p><u>アドホック試験報告書の作成</u></p> <ul style="list-style-type: none">・アドホック試験でバグが出なくなった状態、もしくは最終日の午前中にアドホック試験報告書を作成する <p><u>アドホック試験報告書を講師に提出とチェック依頼</u></p> <p><u>シナリオ試験完了</u></p>
1営業日	<p>【最終確認（リリース）＆発表会】</p> <p><u>最終確認（リリース）</u></p> <ul style="list-style-type: none">・最終版のWARファイルを作成する（リーダーもしくはメンバー）・最終版のWARファイルとSQLファイルをマイドライブに保存する（リーダーもしくはメンバー）・講師にAWSへアップロードを依頼する（リーダーもしくはメンバー）・AWSにアップロードされたチームサイトのURLをWebブラウザで開く・反映されたことを確認する <p><u>発表会</u></p> <ul style="list-style-type: none">・最終日15：30～ チームサイトを発表する