Seasar 双六

Version:1.0.1

ゴール

(12) 上司へ Seasar 申請通らず (TT

(13)View 担当がソースを破壊。 1 回休み

(14)JSP と HTML と Java と格闘

(15) モックを完成。内部処理実装開始

(16)O/R マッピングツールを選定

(17) マッピングツールの設定記述が面倒

(18) マッピングツール利用クラス作成

(19)DB との接続失敗！ 1 回休み

(20) 状態遷移関連のバグにハマる (T_T

(21) 外部システム連携を SOAP で実装

(22)SOAP の実装が面倒・・・苦しい

(23) 各要素を統合して試験開始

(24) 外部システム連携失敗！ 2 マス戻る

(25) ログ出力処理をひたすら実装

(26) 例外処理をひたすら実装

(27) 性能が出ない。1 回休み

(28) 不具合が直らない。 1 回休み

(29) 公開直前の謎のバグ発生。 1 回休み

(30) お客様に怒られながらも遂にリリース

(15) 公開準備。順調だ！

(14)S2Javelin でリリース前のバグを解決

(13) コンポーネントを統合して試験開始

(12)S2JMS で外部システムと連携！

(11)Kijimuna で実装効率化。 1 マス進む

(10)S2Buri の状態遷移を実装に活用

(9)S2Dao で DB 連携と内部処理実装

(8) モック完成！内部処理実装開始

(7)S2JSF プラグイン便利。 1マス進む

(6)S2JSF で、View 作成担当者と連携

(5)S2JSF で、モック作成効率化

(4) システム分析後、モック作成開始！

(3)S2Buri 導入で、修正を即反映

(2)S2Buri で業務分析実施

(1) 構築するシステムの分析開始！

スタート

(11)View 担当との連携失敗。 2 マス戻る

(10) 顧客から質問と要望の嵐。 1 回休み

(9)JSP 使用の為、動くモックが作れない

(8)JSP を使い、モックを作成開始

(7) システムの分析をやっと終了させた

(6) 顧客の変更依頼に対応できず

(5) 業務フロー作り直し。 1 回休み

(4) 業務フローの作成に無意味さを感じる

(3) 分岐の多用でフローが煩雑化！

(2) 業務フローを作成して流れを把握

(1) システムの分析を精力的に開始！

【S2Javelin】
開発 / 運用を問わず、性能監視と分析が出来、問題原因の特定も容易にできる。

【S2Dao】
XML 記述や JDBC を意識せず、DB とのやり取りが可能。
簡単な SQL は自動生成され。
カスタマイズも完全対応。

【Kijimuna】
S2 での設定ファイルの編集と検証を用意に行える。

【S2JSF】
開発画面が HTML なので、顧客の確認を取りながら画面開発を進められる。

【S2JSF Plugin】
S2JSF のタグを自動補完 & バリデーションが可能。

【S2Buri】
業務フローの分岐を定義ファイルとして分岐。
修正に柔軟に対応可能。

【その他の S2Family】
S2Struts : 画面へのアクションの実装 / テストを用意に。
Teeda : DI と AOP をフルに活用可能。
S2Flex2 : リッチクライアントアプリケーション構築の際、サーバサイドとクライアントサイドの実装を分離可能。
S2Ajax : Ajax の開発に S2 を適用可能。
Kuina : EJB で S2Dao を活用可能。

遊び方 : 通常の開発と Seasar での開発を体感できる双六です。サイコロ1個を皆で振り合いながら、システム開発しましょう♪

2006/5/15 D.T:Y.N:S.K