

リモートユーザ参加型 Web アプリケーション開発環境の改良

横田 義和[†] 矢野 日高[‡] 山田 敬三[†] 田中 充[†] 佐々木 淳[†] 船生 豊[†]

[†]岩手県立大学ソフトウェア情報学部 ソフトウェア情報学科 〒020-0193 岩手県滝沢村滝沢字巣子 152-52

[‡]岩手県立大学 ソフトウェア情報学研究科 〒020-0193 岩手県滝沢村滝沢字巣子 152-52

E-mail: [†] [‡] {g031b160, g231e036}@edu.soft.iwate-pu.ac.jp, {k-yamada, m-tanaka, jsasaki, funyu}@iwate-pu.ac.jp

あらまし 昨今, アジャイル型システム開発方法論の一つである XP(eXtreme Programming)が注目されている. しかし, XPは開発の現場にユーザが参加しなくてはならないためユーザの負担が大きいいわれている. 著者らはユーザにとって負担の少ないXP開発環境を実現するため, 遠隔地からユーザが開発に参加できる協調開発環境を構築し, 試行実験を行った. その結果, ユーザと開発者間のコミュニケーションや操作性という点で, いくつかの問題点が明らかになった. 本論文では, よりユーザにとって負担の少ないXP開発環境を実現するため, 問題点を解決する新たな機能を構築し, この環境を用いたシステム開発実験, 評価を行った.

キーワード 遠隔地, 協調開発, リモートユーザ, XP, コミュニケーション

Improvement of Web Application Development Environment with Remote-user Participation

Yoshikazu YOKOTA[†] Hidaka YANO[‡] Keizo YAMADA[†] Michiru TANAKA[†] Jun SASAKI[†]
and Yutaka FUNYU[‡]

[†] Faculty of Software and Information Science, Iwate Prefectural University 152-52 Sugo, Takizawa, Takizawa-mura, Iwate, 020-0193 Japan

[‡] Graduate School of Software and Information Science, Iwate Prefectural University 152-52 Sugo, Takizawa, Takizawa-mura, Iwate, 020-0193 Japan

E-mail: [†] [‡] {g031b160, g231e036}@edu.soft.iwate-pu.ac.jp, {k-yamada, m-tanaka, jsasaki, funyu}@iwate-pu.ac.jp

Abstract In recent year, XP(eXtreme Programming) that is one of the agile system-development methodologies is paid to attention. However, it is said that end users' load is large in XP, because the users have to participate in the development site. Therefore, we construct a cooperative development environment that the user is able to participate from the remote place in the development place. In order to compare our proposed method with XP, we carry out an evaluation experiment. As a result, some problems are clarified in the communications between the user and the developer and the usability. To achieve the XP development environment with light load of the user, we construct new functions to solve the problem. This paper describes the conventional problems and evaluation results of our new proposal.

Keyword remote, collaborative-development, view, users in remote site, communication

1. はじめに

インターネットを基盤としたネットワーク社会の到来により, 情報システムには, より多様なニーズに応え, かつ早期に開発されることが求められている[1][2][3].

高品質な情報システムを迅速に構築するため, 様々な開発方法論, 開発技術が提案・実践されているが,

情報システム開発プロジェクトの多くがエンドユーザの要求に応えられず失敗に終わっている[4][5]. プロジェクトが失敗する要因としては, 要求仕様が不完全であること, 頻繁な要求使用の追加・変更等が挙げられているが[6], これは情報システム開発の上流工程である要求定義, 設計工程が不完全であるためと考えられる.

この問題解決を図る方法として、新たな開発方法論としてエクストリームプログラミング (XP) による情報システム開発が注目されている。XP による情報システム開発では本質的な要求を持っているユーザ自身と協調して開発を行うため、真にユーザの要求に沿った情報システム構築が期待される。しかし、XP のプラクティスの一つであるオンサイト顧客では、ユーザは常に開発の現場に同席しなければならない。そのため、ユーザに負担がかかり、実現が困難なプラクティスとなっている。

そこで、我々はユーザに負担の少ない XP 開発の実現を目的に、近年進展が著しい Web アプリケーションを対象とした遠隔地からユーザが開発に参加できる協調開発環境のモデルを提案し、その提案モデルに基づいてプロトタイプシステムを構築した。提案システムの特徴は、Web アプリケーションの特性を生かし、リリースのタイミングだけではなくインターネット上で常時システムの動作確認が可能になることで、これによりユーザが開発に割かれていた時間の軽減と、ユーザが開発の現場 (フロア) に来るといった負担が解消される。

しかし、実際に Web アプリケーションの開発、実験評価を行った結果、プロトタイプシステムに対していくつかの問題点が明らかになった。

そこで本論文では、現行の協調開発モデルにおいて、よりユーザにとって負担の少ない XP 開発環境を実現するために新たな機能を構築し、再び Web アプリケーションの開発、実験評価を行ったので、その結果について述べる。

2. 現行の協調開発環境の概要

2.1. 現行の協調開発モデル

まず、我々がこれまで提案した遠隔地からユーザが開発に参加できる協調開発モデルのプロセス (図 1) について説明する。図 1 において、インターネットを介して、(1)開発者は実装済みの成果物をユーザに見せ、(2)ユーザはその時点でのシステムの動作確認を行い、(3)要求の追加・修正がある場合は開発者に通知する、(4)開発者はユーザからの要求を確認し、(5)システムの修正を行い、(1)その成果物をユーザに見せるという一連の作業を繰り返す。

本モデルでは、非同期的にユーザがシステムへの要求を通知したり、動作確認を行えるので、ユーザの都合に合わせて、より負担の少ない協調開発が可能となる。また、開発者はユーザの要求の追加・修正が即座に把握できるため、ユーザの意図を迅速に反映したシステム開発が可能となる。

2.2. 現行のプロトタイプシステムの概要

我々は、前述のモデルに基づいて、PHP, Smarty を用いた Web アプリケーション開発を対象とした遠隔協調開発環境支援システム「RCDESS (Remote Collaboration Development Environment Support System)」を構築した。そのシステムの構成を図 2 に示す。

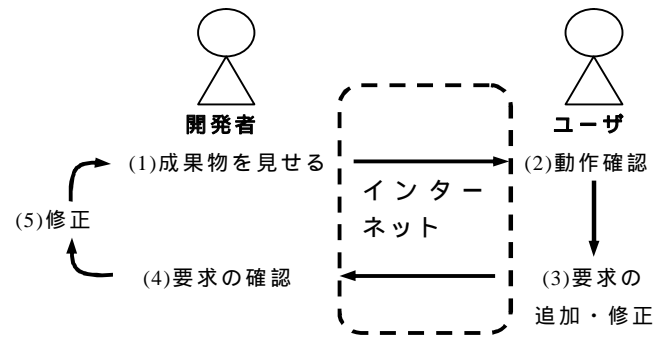


図 1 現行の協調開発モデル

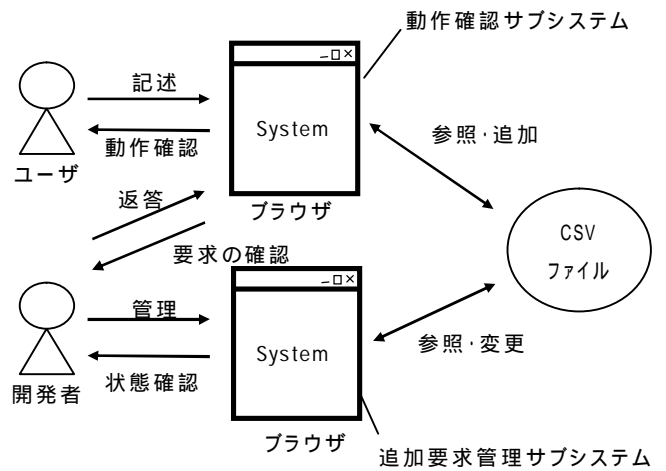


図 2 システム構成図

RCDESS は、開発者が開発対象システムの成果物をユーザに見せ、ユーザが動作確認を行う「動作確認サブシステム」と、ユーザからの修正・追加の要求を管理する「追加要求管理サブシステム」からなる。また、開発履歴を残すため追加・修正の要求を格納する CSV ファイルを設ける。

2.2.1. 動作確認サブシステム

本サブシステムの画面構成は図 3 に示すように、フレームによって 2 つに分割されている。動作確認フレーム (画面右) では、開発中システムの動作確認を行う。要求記述フレーム (画面左) は、開発中システム

の1画面毎にコメントが入力できるテキストボックスが存在する。このテキストボックスにおいてユーザは追加・修正要求を、開発者は回答をそれぞれ記述することができる。また、開発者側からユーザへの意見・質問等を記述することもできる。

本サブシステムの機能は以下の6つである。

- ・ **動作確認機能**：開発中システムの動作確認を行う。開発の進捗度をリアルタイムに把握することができる。
- ・ **コメント機能**：要求記述フレームのテキストエリアに、ユーザの要求記述、または開発者からの意見・質問を記述する。
- ・ **返信機能**：記述されたコメントに対して返信を行う。各コメントをクリックすると、返信を記述するための別 Window が表示される。
- ・ **チャット機能**：会話形式にユーザと開発者がやり取りをする場合に利用する。要求記述フレーム下の「チャット」ボタンをクリックすると、チャットをするための別 Window が表示される。
- ・ **過去ログ参照機能**：過去のコメントとそれに対応する返信コメントの参照を行う。要求記述フレーム下の「ログ」ボタンをクリックすると、過去ログを参照するための別 Window が表示される。
- ・ **RSS 生成機能**：新たなコメント又は返信が入力された場合 RSS を生成する。そのため、ユーザ・開発者共に RSS リーダーに登録すれば、新着コメントの有無を常に把握することができる。

本サブシステムの機能は以下の3つである。

- ・ **状態別閲覧機能**：新着・既読・修正中・修正済の4つの状態別に要求の一覧を表示する。他の開発者が修正を行っている要求の確認ができるため、開発者間の協調関係が図れる。
- ・ **機能別閲覧機能**：対象システムの機能別に要求の一覧を表示する。機能別の項目は対象システムに依存するが、各機能と階層関係の抽出は、要求が格納されている CSV ファイルから読み取るため、システムごとに設定をする必要は無い。
- ・ **状態変更機能**：各要求に対し既読・修正中・修正済へ状態を変更できる。また、ここで修正済に変更された要求は、動作確認サブシステムでは過去ログに格納される。

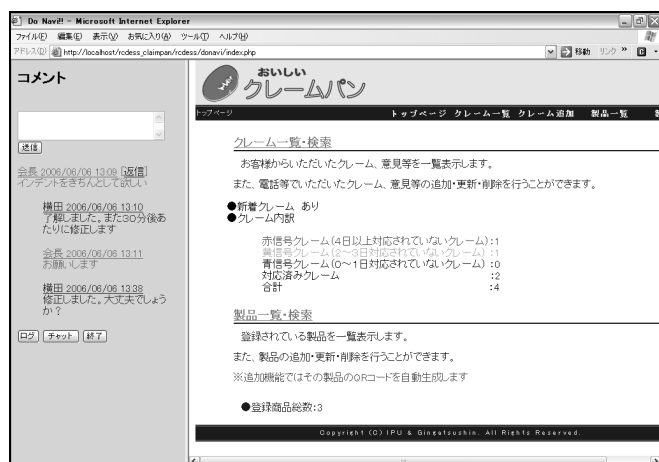


図3 動作確認システム画面例

2.2.2. 追加要求管理サブシステム

本サブシステムは動作確認サブシステムによって記述された要求の管理を行う。図4はその画面例である。



図4 追加要求管理サブシステム画面例

3. 試行実験の概要と確認された問題点

3.1. 概要

試行実験（以下、第1次開発実験）では、本学の古川氏らが提案する「食育サポートシステム」の開発を対象とし、RCDESSを導入した場合（以下 YP: Yokota Process）としない場合（以下 XP）とで開発効率を比較することにした。本実験では、すべての機能の開発ではなく、上記システムの「認証機能」「生産者管理機能」「ユーザ管理機能」の3つの機能の開発に評価対象を絞った。なお、システムの内部仕様上の特徴は、演算処理よりもDBへのデータの挿入、抽出が多い点であった。両開発実験においては、できるだけ共通した開発体制と環境を用いることとした(表2)。両開発実験ともユーザは同一人物（古川氏）である。開発要員は同等のスキルレベルを持つ別々の学生3人である。

表 2 開発体制と環境

開発要員	3人
OS	MacOS X
スクリプト言語	PHP 4
DB	MySQL 4.0.27
フレームワーク	Smarty

3.2. 確認された問題点とその考察

第1次開発実験の結果、アンケートから主に以下のような問題点が確認された。

(1) ユーザと開発者間におけるコミュニケーション

RCDESS でユーザと開発者間のコミュニケーションが円滑に行えなかったという意見が多かった。その原因として、リアルタイムに会話形式でコミュニケーションを取れるチャット機能を一度も利用せず、コメント機能とその返信機能のみでやり取りをしていたからと考えられる。これはお互いにどのタイミングでチャットが利用できるか分からなかったためである。そのため、お互いにチャットを利用する時間を事前に打ち合わせできる機構が必要である。

(2) ユーザ自身による要求の状態確認の負担

現行の RCDESS では要求の状態を確認できるのは追加要求管理サブシステムを利用できる開発者のみであり、追加要求の状態変更はユーザに知らされない。そのため、ユーザは自身の出した要求に対し、それが開発者に伝わっているか、また、それが反映されたかの確認をする必要があり、その時間が負担となっていた。このことから、ユーザに対する要求の状態通知の機構が必要である。

(3) どの状態の定義にも属さないコメントの発生

要求に対し、新規、既読、修正中、修正済といった状態の定義がされているが、例外のコメントが存在することが判明した。例えば、「この画面の ID & パスワードは～です」「管理者メニューを再開しました」など、その画面に対する要求ではなく、お互いの意思疎通を図るためのコメントも見受けられた。これらはシステム修正に関係なく、修正中、修正済といった状態に属せず、新たなコメントの状態を定義する必要がある。

4. 新機能の構築

本章では、前章で挙げた RCDESS に対する問題点を解決するために、動作確認サブシステム、追加要求管理サブシステムに新たな機能を追加して再構築した。

4.1. 動作確認サブシステムの追加機能

本サブシステムに対し、新たに構築した機能について、その概要と予想される効果について述べる。

(1) 打合せ調整機能

あらかじめチャットを行いたい日付、時間の指定を行う。単位は1時間ごとに指定可能で、相手はその要請に対し参加の有無を返答する。要求記述フレーム下の「打合せ調整」ボタンをクリックすると時間指定、返答のための別 Window が表示される。

(2) 状態表示機能

各々の要求に対し、その左側に状態を表示する。

機能(1)から予想される効果: あらかじめチャットをする日程を指定しておくことにより、チャット利用が促進され、コミュニケーションが円滑に出来る。

機能(2)から予想される効果: 本サブシステムからも要求の状態を確認できるため、ユーザは開発者に要求が伝わったか、現在修正中なのかなどの開発状況を把握できる。

4.2. 追加要求管理サブシステムの追加機能

本サブシステムに対し、新たに構築した機能について、その概要と予想される効果について述べる。

(1) 状態通知機能

本サブシステムによって追加要求の状態がされた場合 RSS を生成・配信する。

(2) 状態の追加

これまで新着、既読、修正中、修正済だった状態に「お知らせ」を追加した

機能(1)から予想される効果: これまで、ユーザからの要求が修正された場合その通知は無く、ユーザ自身が確認作業を行う必要があり、その作業が負担となっていた。開発者によって状態が変更される度にその旨を RSS で通知することにより、ユーザ自身による確認作業の負担を軽減できる。

機能(2)から予想される効果: 前述したようにどの状態の定義にも属さないコメントに対し、「お知らせ」という状態で統一することが可能になる。

5. 実験評価

5.1. 概要

開発実験(以下、第2次開発実験)では、本学の浜田氏らが提案する「食事指導支援システム」の開発を対象とし、第1次開発実験と同じく YP と XP とで開発効率を比較することにした。本実験では、すべての機能の開発ではなく、上記システムの「認証機能」「料理管理機能」「ユーザ管理機能」「クライアント管理機能(一部除く)」の4つの機能の開発に評価対象を絞った。なお、システムの内部仕様上の特徴は、「食育サポートシステム」と同じく演算処理よりも DB へのデータの

挿入、抽出が多い点であった。両開発実験においては、開発要員を4人とし、その他は第1次開発実験と共通した開発体制と環境を用いることとした。

YP、XP両開発実験ともユーザは同一人物（浜田氏）である。開発要員は同等のスキルレベルを持つ別々の学生4人である。

両開発実験では、開発手法の比較材料として以下のデータを取得することとした。

(1) 追加要求量

XPの特徴であるユーザの要求を反映した開発がYPでも行われたかを評価するため、両開発実験においてユーザからの追加・修正の要求数（以下追加要求量）を測定する。

(2) コミュニケーション時間

XPは現場で会話をするのみだが、YPはコメントを入力する時間が発生するため、コミュニケーションに要する時間（以下コミュニケーション時間）を比較する。

(3) 拘束時間

オンライン顧客では開発の現場に同席していてもユーザはコーディング待ちなど何もできない時間（以下、拘束時間と呼ぶ）が発生し、ユーザへの大きな負担となるため、その時間の違いを比較する。

以下、XPとYPにおける測定内容の詳細を述べる。

5.2. XPでの開発実験

(1) 追加要求量

XPの場合、ユーザと顧客の話合いによって発生した追加要求についてドキュメントは残さないが、今回の実験では追加要求量を計測するため、追加要求が発生する毎にその内容を記述する追加要求レポートを作成することとした。記述するタイミングとしては、ユーザと開発者の話合いが終了した時点で、記述する内容は話合いにかかった時間、日付、追加要求内容である。

(2) コミュニケーション時間

開発中、ユーザからの追加要求、開発者からの仕様についての質問、提案について話合いが行われるが、その全ての話し合いにかかった時間を記録する。

(3) 拘束時間

基本的にユーザと開発者の話合いの時間以外ユーザはコーディングが終わるまで何もしないで待つ。その時間をユーザの拘束時間とする。拘束時間は、全体の開発時間からコミュニケーション時間を差し引いて求めた。

5.3. YPでの開発実験

(1) 追加要求量

追加要求管理サブシステムのデータから追加要求

量を求める。対象データはユーザからの要求かつ修正が必要なコメントで、その件数を追加要求量とする。

(2) コミュニケーション時間

ユーザと開発者のコミュニケーションは、全てコメント機能、チャット機能で行われるのでテキストベースとなる。そのため、コメント機能にてやり取りされた時間と、チャット機能を利用した時間を足したものをコミュニケーション時間とする。コメント機能にてやり取りされた時間は、ユーザのコメント、開発者のコメントを考えて、タイピングするのにかかった時間とする。チャットを利用した時間は動作確認サブシステムから求める。

(3) 拘束時間

YPでは、ユーザは自分の都合に合わせて動作確認、要求の記述が行えるので、非同期的に開発に参加していた場合拘束時間は発生しない。しかし、打合せ調整機能により、開発者からチャットの要請があり、それにユーザが承諾した場合、打合せ調整機能により指定された時間は同期的に開発に参加していたことになる。そのため、打合せ調整機能によって指定された時間から、その間コミュニケーションが行われたチャット時間を差し引いた時間を拘束時間とする。

6. 評価

第2次開発実験の計測結果を表2に示す。

表2 第2次開発実験の計測結果

	YP	XP
追加要求量（件）	13件	8件
コミュニケーション時間	3時間3分 (9.7%)	1時間2分 (2.8%)
拘束時間	17時間59分 (57.1%)	35時間38分 (97.2%)
開発時間	31時間30分	36時間40分

追加要求量については若干YPの方が多いものの、YP、XP共に大きな差は無かったといえる。コミュニケーション時間はYPの方が多くかかることが判明した。YPのコミュニケーション時間は、追加要求管理システムから抽出されたチャット以外のコメントの文字数（1560文字）を考えながらタイピングした時間（1時間2分）と、チャット機能を利用した時間（2時間1分）を合計した3時間3分とした。YPにおけるユーザの拘束時間は、打合せ調整機能により指定され、ユーザが同期的に開発に参加した合計時間（20時間）からチャット機能を利用した時間（2時間1分）を差し引いた17時間59分とし、開発時間全体において57.1%

を占める割合となった。一方、XPにおけるユーザの拘束時間は、開発時間全体からコミュニケーション時間を差し引いた35時間38分となり、開発時間全体において97.2%を占める割合となった。

次にRCDESSの利便性、開発されたプロダクトへの満足度について、ユーザと開発者それぞれに評価するためのアンケート調査を行った。

コミュニケーションはスムーズに行えたかという問いに対し、YP、XPともに「当てはまる」「やや当てはまる」という回答を得られ(9名/9名)、ユーザの要求はシステムに反映されたかという問いには全員が「当てはまる」「やや当てはまる」と回答した(7名/7名)。開発されたプロダクトへの満足度としては、XP、YP共に「満足」という回答を得られた(7名/7名)。開発中、ユーザ自身にとって無駄に過ぎたと思われる 때가あったかという問いに対してユーザからは「当てはまらない」という回答が得られた。

また、ユーザから「YPは時間的、場所的拘束が少ないのが魅力的。しかし、要求の伝え方とか伝わりやすさはXPの方がいい」といった意見があった。

7. 考察

追加要求量については若干YPの方が多いものの、YP、XP共に大きな差は無かったことから、遠隔からでもユーザの要求を反映したシステム開発が可能であることを確認した。YPはコミュニケーションでは多少負担になるという結果になったが、ユーザの拘束時間を大幅に削減できたことの効果が大きく、総合的にはユーザの負担が少ないシステム開発環境であるといえる。

今回、YPのコミュニケーションの問題を改善すべくチャット機能利用を促進する打合せ調整機能を構築した。その効果は大きく、第1次開発実験の際には1度も利用されなかったチャット機能が、今回の実験では22回程度の利用が見られ、「コミュニケーションがスムーズにできた」というアンケート結果の大きな要因となった。また、第1次開発実験時にYPでは「どの要求が修正完了したか確認する作業が無駄」という意見があったが、今回の実験では無駄と感じる時間は無かったという結果が得られ、各追加要求の状態表示機能、状態通知機能を構築した効果があったと考えられる。

これらのことから、本研究の目的であるユーザに負担の少ないXP開発の実現可能性を確認できたといえる。

しかし、今回の実験開発はユーザ、開発者共に全て同一研究室内のメンバーで構成されており、XPでは実際の現場にいるユーザは追加要求を出すことに遠慮感

があったこと、YPでは最初からチャットでコミュニケーションを取ることに抵抗感が無かったことなどが今回の結果に多少影響があったことを考慮すべきである。

8. おわりに

本稿では、現行の協調開発環境に対し、よりユーザに負担の少ないXP開発環境の実現を目指し、現行の協調開発環境の問題点を挙げ、それに対応する新たな機能を実装した。さらに、本システムを用いた場合と、用いない場合でのWebアプリケーション開発実験を行い、本提案の有効性を確認した。

今後は、YPを利用したXPの開発プロセスの確立の必要性がある。YP、XPにはそれぞれメリット、デメリットが存在し、今までのXP開発にYPを取り入れるという両開発プロセスのハイブリット型が実際には有効であると考えられる。また、YPは実装段階前の要求分析、設計段階の上流工程での利用を考慮していない。そのため、上流工程での遠隔協調開発環境の提案も必要になってくる。その他に、よりコミュニケーションの負担を軽減するため、音声やビデオチャットを利用したコミュニケーション手段も考慮すべきである。

謝辞

最後に、開発実験にご協力いただいた、本学同研究室の古川恵理奈氏、浜田優子氏、小田雅貴氏、本田創大氏、高橋克也氏、北川愛優美氏、矢野日高氏、三田村唯氏、山下徹氏に感謝致します。

文 献

- [1] 総務省，“平成17年度版情報化白書”，<http://www.johotsusintokei.soumu.go.jp/whitepaper/ja/h17/index.html>，2004。
- [2] 財団法人インターネット協会監修，“インターネット白書2005”，インプレス，2005。
- [3] 経済産業省，“平成16年情報処理実態調査結果報告”，http://www.meti.go.jp/policy/it_policy/statistics/jyojitu.htm，2005。
- [4] The Standish Group，“2004 CHAOS Demographics and Project Resolution”，<http://www.standishgroup.com/>，2004。
- [5] 中村建助，矢口竜太郎著，“プロジェクト成功率は26.7% 2003年情報化実態調査”，日経コンピュータ2003年11月17日号，2003。
- [6] The Standish Group，“The CHAOS Report”，<http://www.standishgroup.com/>，2003。
- [7] 横田義和，矢野日高，田中充，佐々木淳，船生豊，“リモートユーザ参加型Webアプリケーション開発環境の構築”，電子情報通信学会，KBSE2006-11。