Paychex Demonstration Website

Project Plan

Updated 9-Feb-10

Contents

[Paychex Demonstration Website 1](#_Toc256681875)

[Project Plan 1](#_Toc256681876)

[Updated 9-Feb-10 1](#_Toc256681877)

[Overview 1](#_Toc256681878)

[Deliverables 1](#_Toc256681879)

[To RIT: 2](#_Toc256681880)

[Risk Management 2](#_Toc256681881)

[Failure Mode Analysis 3](#_Toc256681882)

[Scheduling & Estimates 4](#_Toc256681883)

[Sprint Schedule 4](#_Toc256681884)

[Sprint Planning 4](#_Toc256681885)

[Measurements & Metrics 5](#_Toc256681886)

[Process Metrics: 5](#_Toc256681887)

[Burndown chart 5](#_Toc256681888)

[Velocity/Average lead time analysis 5](#_Toc256681889)

[Defect Log Analysis 5](#_Toc256681890)

[Project Metrics: 5](#_Toc256681891)

[Test coverage 6](#_Toc256681892)

[Performance testing 6](#_Toc256681893)

[Code complexity 6](#_Toc256681894)

[Technical Process 6](#_Toc256681895)

Overview  
The Amazon Cloud Sales Demonstration Website involves deploying and running several of Paychex’s payroll and human resources products on Amazon EC2. The company’s current demonstration process is currently inefficient to set up for hundreds of sales representatives distributed across the country. Team Sentinel Prime will be creating a demonstration platform that will allow untrained sales reps to effectively demonstrate the Paychex product suite. The project involves several improvements over the current manual way of setting up demonstrations for customers. Data integration across the various applications will allow sales reps to show customers the benefits of using multiple products to manage their payroll and human resource efforts. Cloud computing will allow Paychex sales reps to easily demonstrate their offerings to customers across many industries. In an effort to make the demonstrations seem fresh and relevant, there must be a method of providing and maintaining a set of realistic data for customers to evaluate.  
  
Goals & Scope

The broad scope of the project is to design and deploy a sales demonstration environment hosted on Amazon web services cloud computing environment to host payroll and human resources products for demonstration to Paychex's prospective clients.  Team Sentinel Prime will design and deploy sample applications based on the platforms and technologies used by Paychex's existing applications.  It is important to Paychex to demonstrate the following capabilities of their software to their clients:

* Single sign-on capabilities, allowing a single username and password to access all applications
* Data integration across applications
* Accommodation for up to 40 concurrent users running individual self-contained demonstrations
* An administrative module, allowing users to select an industry type when creating an environment to depict targeted information within each application

Ultimately, the goal of this project for Paychex is to increase sales of their payroll and HR applications by being able to create a more personalized, enticing presentation to potential customers.  This new environment will be more credible and persuasive than the current user experience.  Sales representatives will be able to customize each sales environment to their audience, so sample data seems more realistic, and pull up additional products which clients may be interested in.  Users will be able to manually reset demo and sample data back to its original state, and system administrators will be able to easily manage and monitor the entire Amazon Web services environment.  Administrators will be able to easily update software configurations based on application changes and releases, ensuring sales representatives are always showcasing the most recent version of their products.

Deliverables  
To Paychex:

* Initial version of the operational demonstration website deployed for the users.
* Infrastructure and component design documents relating to the deployment within the Amazon cloud
* User instructions and guides
* A document of lessons learned for EC2

## To RIT:

* Project website (<http://www.se.rit.edu/~sentinelprime/>) holding all non-proprietary work products and project artifacts maintained in the project account on the ***se.rit.edu*** web server.
* Project plan, schedule and process methodology definition prepared by the end of week 3 of the first term.
* Tracking report for time/effort worked by each team member and the team aggregate updated on the project website weekly.
* Tracking report for at least two product/process metrics appropriate to the project and development methodology updated on the project website at least every two weeks.
* Interim status and final project presentations
* Project poster and presentation at "SE Senior Project Day"
* Project technical report
* Interim and final team self-assessment
* Post-mortem curriculum reflection report
* A CD(s) at the conclusion of the project containing all project artifacts.
* Each team member completes a Software Engineering Program Senior survey

# Risk Management

|  |  |  |
| --- | --- | --- |
| **Risk** | **Explanation** | **Mitigation Strategy** |
| Response Bottleneck | Tasks may depend on a response from the client. If the client is unavailable to respond in the desired timeframe, the project may experience delays that it may not be able to recover from. | 1. Several contacts have been made within the client organization. If the primary contact seems unresponsive, the team has other contacts to reach out to.   2. Tri-weekly stand-up meetings will shed light on any potential problems and bottleneck scenarios before the project is in jeopardy. |
| Unfamiliarity with Technology | Amazon EC2 is a new and unfamiliar to most team members. This creates a risk for both initial planning and requirements elicitation and the project of the project itself. | 1. During the 2-week winter recess, team members will research EC2 and explore its capabilities. |
| Outside Commitments | In addition to senior project, all team members are taking more classes, working, or both. This creates reduces the time each team member has to work on the project, and attend meetings. | 1. The team is self-organizing, so most tasks will be voluntarily assigned.   2. Short iteration cycles, velocity-based planning and a static scope during an iteration will limit the work load to what is determined to be manageable by the team.  3. The tri-weekly meetings will provide a way for team members to communicate any time management issues that may arise. |
| Dynamic Scope | As the understanding of the project increases by both the team and clients, scope may change and new features added or removed. This is particularly a risk during requirements elicitation. | 1. The team will provide the client with a series of prototypes to solidify the understanding of the problem.  2. A reactive and adaptive process methodology was chosen - scrum - to accommodate a fluctuating scope. |
| Long iteration zero | The amount of time it takes for the team to get up to speed on the project scope, technologies and desired feature set may eat up too much time. There may not be enough time left to complete the project as stated. | 1. Early planning and utilizing the 2-week winter recess will give more 'Iteration 0' time.   2. The development process chosen is receptive to changing scope. If the team feels iteration 0 is taking too long, it may decide to just start the first iteration and accept any changes as they come as new features.  3. Using story points and team velocity as a guideline, the team will be able to estimate how long a given set of tasks will take to complete. There is a hard deadline for this project, and scope is flexible to a degree. |

# Failure Mode Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk** | **Severity** | **Likelihood** | **Undetectable** | **Score** |
| Long iteration 0 | 8 | 8 | 2 | 128 |
| Outside commitments | 4 | 7 | 3 | 84 |
| Unfamiliarity with EC2 | 9 | 9 | 1 | 81 |
| Response bottleneck | 6 | 5 | 2 | 60 |
| Dynamic scope | 2 | 4 | 1 | 6 |

# Scheduling & Estimates

Week 1: Meet with sponsors, Produce project synopsis

Week 2: Technology/Sales meeting with sponsors, Generate questions based off technology meeting to well-define the scope of the project

Week 3: In-depth Technology meeting(Cancelled), Initial revision of Project Plan

Week 4: Researching EC2, Prototyping based off project synopsis

Week 5: Create mock-ups of sponsor application, Evolve prototype, start iteration 1

Week 6: Finalize Prototype, Set up scrum board, Start initial backlog

Week 7: Continue iteration work

Week 8: End iteration1, retrospective, get Paychex feedback, start iteration 2

Week 9: Continue iteration 2 work

Week 10: Finish iteration 2, retrospective

Week 11-20: Finish Project

## Sprint Schedule

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Start** | **End** | **Story Points** | **Burndown** | **~RIT Weeks** | **Total Points** |
| 0 | 5-Nov | 14-Jan | 0 | 85 | Winter 1-5 | **85** |
| 1 | 15-Jan | 28-Jan | 12 | 73 | Winter 5-7 |  |
| 2 | 29-Jan | 11-Feb | 10 | 63 | Winter 7-9 |  |
| 3 | 8-Mar | 18-Mar | 17 | 46 | Spring 1-2 |  |
| 4 | 19-Mar | 1-Apr | 18 | 28 | Spring 2-4 |  |
| 5 | 2-Apr | 15-Apr | 16 | 12 | Spring 4-6 |  |
| 6 | 16-Apr | 29-Apr | 12 | 0 | Spring 6-8 |  |

## Sprint Planning

|  |  |  |
| --- | --- | --- |
| **Sprint** | **Stories** | **Points** |
| 3 | Admin: add new industry option | 5 |
|  | User: reset password | 2 |
|  | User: signs out | 2 |
|  | User: set up demo | 8 |
|  |  | **17** |
| 4 | sales: sign into app/user access permissions | 3 |
|  | sales: demo single sign on | 2 |
|  | admin: destroy any demo instance | 5 |
|  | sales: destroy demo instance | 5 |
|  | admin: view active created demos | 3 |
|  |  | **18** |
| 5 | admin: update baseline data | 5 |
|  | admin: remove app on ex2 | 5 |
|  | sales: select industry in environment | 3 |
|  | sales: reset data | 3 |
|  |  | **16** |
| 6 | admin: update app on ec2 | 8 |
|  | sales: demo data integration | 2 |
|  | admin: find old apps | 2 |
|  |  | **12** |

# Measurements & Metrics

Team Sentinel Prime will be tracking several measurements and metrics throughout the lifecycle of the project in an effort to show improvement over time for both how the project is run and with how it is implemented. The results will be published on the course website for the project sponsors and RIT representatives to view.

## Process Metrics:

### Burndown chart

The SCRUM process breaks up tasks for each iteration, or sprint, into separate backlogs. Each backlog item has a set number of hours that the team has estimated that the task will take. This chart provides a view into how many hours are left in the sprint according to how many tasks have been completed so far. As it is updated each day, it's easy to gauge when the sprint will be done.

### Velocity/Average lead time analysis

This measures how much work was done over an iteration, which is usually counted by the number of story points assigned to each feature. This metrics helps with determining how much can get done in one iteration and helps with planning future work. Also, tracking sprint task and how long it takes to complete it will be essential in determining how the team is progressing with implementation and testing.

### Defect Log Analysis

The team plans on keeping an extensive log of defects that come up throughout the project. By tracking when they come up and when (or not) they are resolved, it will be possible to determine which parts of the iteration, such as integration testing or requirements elicitation, that needs more attention.

## Project Metrics:

### Test coverage

Metrics on how much of the code is covered by tests from both unit and integration levels will show how committed the team was to solid testing of the implementation and provide insight into problem areas that may be less reliable than others. The team will not be shooting for 100% code coverage since that is usually a goal that provides little benefits for too much effort, but instead will be seeking a maintainable level of coverage that still assures reliability and dependability for the codebase.

### Performance testing

One of the main requirements the project sponsors require is that the applications can withstand a certain amount of concurrent users, and this can be proven by using performance testing tools on the website the team creates. A community accepted tool to do this is Apache Bench, which can give an accurate rating of how many requests per second the server can handle.

### Code complexity

Team Sentinel Prime realizes that the end products of this project will eventually be used and developed by Paychex employees, and the goal is to provide as maintainable and extendable codebase as possible. By looking into several metrics and tools that can root out code smells such as cyclomatic complexity and duplication, the project sponsor can be further assured they will be able to build on top of our team's work in the future.

# Technical Process

Scrum, an agile and iterative process, will be used as the technical process. The following describes the adaption of scrum team Sentinel Prime will undertake:

* A ’10 minute standup’ meeting at least 3 times a week
* 2 week iterations
* A working backlog for the project and current iteration, kept on the pivotal.com tracker.
* User stories presented and approved by Paychex
* Story points assigned by the entire team
* A sprint retrospective after each iteration to facilitate process improvement