Software Requirements Specification

for

Right-on-Time

Version 1.0

Prepared by Brant DenHerder

Central Washington University

November 10, 2016

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Brant DenHerder	11/16/2016	Initial version 1.0 of the SRS document	1.0

1. Introduction

1.1 Purpose

The purpose of this document is to describe the features and functionality of Right-on-Time. Right-on-Time is a project tracking system for Breedt Production Tooling and Design. The main high level scope of Right-on-Time includes tracking hours and purchases by employees towards individual projects. The goal is to give Breedt a good representation of how and where money is being spent. Right-on-Time will also be used to track how much work individual employees are contributing each day.

1.2 Document Conventions

Each feature will have the following details covered:

1. Description
2. Stimulus Sequence (This section will only be covered on the features that require user input)
3. Validity
4. Consistency
5. Feasibility
6. Error Handling
7. Priority

1.3 Intended Audience and Reading Suggestions

This document is designed to be a contract between Breedt Production Tooling and Design, and the developers of Right-on-Time. Therefore, the main intended audience for this document includes the developers of Right-on-Time and the owners of Breedt. The contents of this document will contain an overall description and scope of Right-on-Time, an overview of hardware and software interfaces, and finally a comprehensive list of required features. To fully grasp this software, I would recommend reading the entire document in order.

1.4 Product Scope

In short, Right-on-Time is a project tracking system designed to keep Breedt Production Tooling and Design on track with both time and budget for their projects. The goal is to be able to determine if a project was actually profitable by tracking how many hours and how much money is being put into it. Another objective for Right-on-Time is to track how many billable hours each employee is actually completing. Finally, Breedt wants to formalize its process for purchasing project materials by requiring all purchases to be tracked through Right-on-Time. Overall, Breedt wants a better understanding of their project flow and expenses.

1.5 References

"IEEE Recommended Practice for Software Requirements ..." 1998. Web. 15 Nov. 2016.

2. Overall Description

2.1 Product Perspective

Right-on-Time will be a self-contained product that is not dependent on any other external software or systems. All requirements of Right-on-Time will be developed into the application itself. There should be no need for another piece of software to work with Right-on-Time, besides a modern browser.

2.2 Product Functions

Major High-level Functions

1. Project Tracking
    a. Tracking hours put towards each project
    b. Tracking bill of materials (BOM) items
2. Project Overview
    a. Visualize statistics on the amount hours dedicated to the project
    b. Project details: Name, customer, deadline, hours invested, money invested, other.
3. To-Do List
    a. Managers can create to-do list for the next day
    b. Employees can view to-do list on dashboard
4. Multiple User Levels

	Create Project	Enter BOM Item	Authenticate BOM Item	Add/Modify Tasks	Modify To-Do list	View Budget Information	Modify Users
Admin	x	x	x	x	x	x	x
Shop Manager	x	x		x	x		
Project Manager		x		x			
Purchasing			x				
Employee				x			

Table: This table shows the different user levels and what functions that user level can do.

2.3 User Classes and Characteristics

Right-on-Time will be developed for anyone working at BPTD. There will not be any features which require specific training or skills to operate. All features will be text input or visual representations of data, which should not require any special technical experience beyond using a browser.

2.4 Operating Environment

Right-on-Time is being developed as a web based application, which means the user interaction and interface can be conducted on any modern web browser connected to the internet. This includes but is not limited:

1. Internet Explorer
2. Chrome
3. Safari
4. Firefox

The database and backend code will be hosted through a service like Amazon Web Services, discussed in section 3.2 Hardware Interfaces.

2.5 Design and Implementation Constraints

Since we are developing Right-on-Time as a web application there are a few design constraints that come along with web development. These constraints are as follows:

1. The application must work on different browsers
    a. Since Right-on-Time must work on different browsers, we are constrained to UI functionality that works cross browser.
    b. Working with different browsers also requires testing the software on every browser we want to be able to access Right-on-Time
2. The application must have responsive design for mobile use
    a. Since browsers can be viewed as different sizes such as desktop, tablet, and mobile, our interfaces need to be designed such that the functionality will still fit no matter the screen size.
    b. This may also require entirely different interfaces for mobile vs. desktop.
3. We are also limited to the functionalities that browsers allow us. For instance, some browsers now support desktop notifications, while others do not.

2.6 Assumptions and Dependencies

There are very few major assumptions when it comes to software dependencies. However that being said, our group is assuming some about our ability to develop all of these features in a framework we do not have much experience in. The overall assumptions I see at this point include the following:

1. Our database type will work within our framework e.g. Mongo working with ASP.NET
2. User levels can be easily implemented within our database and framework
3. Our application will deploy smoothly to whatever host platform we choose
4. Since we have not been given specifics about the UI design, we are assuming a design based on the requirements as we see fit.

The other assumption being made, which does not relate to the software development, is that users will actually use the tool properly to record their tasks. The software and the data involved will only be as good as the users are willing to provide.

3. External Interface Requirements

3.1 User Interfaces

Right-on-Time will consist of six main views, and in this section we will list those views as well as what functionality will be available in each view. Each view will also contain a menu bar with quick access to the Dashboard View, the Projects List View, and the Settings View.

1. Login View
    a. The user will be able to enter a username and password to login to Right-on-Time
2. Dashboard View
    a. The user will be able to view their currently active tasks
    b. The user will be able to view the current to-do list
    c. The user will be able to view their recently completed tasks
    d. The user will be able to start/pause/complete their tasks
    e. If the user has a high enough user level they will be able to see a list of all active tasks
    f. If the user has a high enough user level, they will be able to edit the to-do list.
3. Projects List View
    a. The user will be able to see a list of all current projects with the newest at the top
    b. The user will be able to click on a project to see the project in the Project View
    c. If the user has a high enough user level, they will be able to add new projects
4. Project View
    a. The user will be able to see the project details include the following:
        i. Project Name
        ii. Project ID
        iii. Project Manager
        iv. Project Description
        v. Project Status (Ready for work, or not ready for work)
        vi. Current Hours Invested vs. Hours Budgeted
        vii. Money invested (If the user level is high enough)
    b. The user will be able to see the projects currently active tasks
    c. The user will be able to see the projects current BOM list
    d. The user will be able to see the currently complete tasks for the project
    e. The user will be able to start/pause/complete their tasks for this project
    f. If the user has a high enough user level, they will be able to request BOM items
    g. If the user has a purchaser user level, they will be able to modify the status of BOM items
    h. If the user has a high enough user level, they will be able to modify the project details
5. Settings View
    a. The user will be able to see their current user information
    b. The user will be able to change their password

c. If the user is an Administrator, they will be able to see a list of all employees and their user levels.
d. If the user is an Administrator, they will be able to modify the user levels of employees.
e. If the user is an Administrator, they will be able to add and remove employees

6. Employee View
    a. If the user has a high enough user level, they will be able to see all the information about a user. This includes the following:
        i. Employee's active and recently completed tasks
        ii. Employee's name, username, and user level

3.2 Hardware Interfaces

The hardware interfaces will be handled by the hosting service we choose for Right-on-Time. The hosts we have discussed may be one of the following options:

1. Amazon Web Services
2. Microsoft Azure

Either option will have similar functionality when it comes to deploying and managing Right-on-Time.

3.3 Software Interfaces

Since Right-on-Time is being developed as a standalone web application, there will be no software interfaces developed for it.

3.4 Communications Interfaces

Since Right-on-Time is a web based application, all communication between the user and the application will be conducted through a web browser. All of the communication between the front-end application and the backend software will be done through HTTP requests. The application will handle all of this communication without user involvement. Also, since we are not dealing with payments, the HTTP communication standard should be secure enough.

4. System Features

The following section will contain two main types of features, those that allow the user to view information, and those that allow users to add/modify information in the database. Those features are listed as the following:

4.1 System Display Features

4.1.1 View Project Details

Description	This feature will allow the user to view the high level details about the project they are viewing. These details will be viewed on the specific project's page. These details include the following: 1. Project Name 2. Customer 3. Project ID/PO 4. Project Manager 5. Current Hours Invested 6. Budgeted Hours 7. Project Description 8. Current Budget Invested (Depending on User Level) 9. Projects Status (Either not ready to be worked on, or ready for work)
Validity	The employees need to be able to see information about the project they are adding tasks to, as well as who they need to talk to about the project if they have questions. These details will also show the employees if they are on track and within the budgeted hours.
Consistency	This requirement doesn't conflict with any other specified requirement.
Feasibility	This will be possible by storing information in the database for each project and retrieving it when a user goes to the project page.
Error Handling	The only error possible would be if the project does not exist in the database. If that is

	the case, then we will send the user back to the projects list view with an error about missing project.
Priority	Required

4.1.2 View Active and Completed Tasks

Description	This feature will allow users to view their tasks as well as all open tasks. Tasks will be shown on the users dashboard view, the project view, and individual employee view. Tasks can be shown in the following lists: 1. Employee's Currently Active Tasks 2. Employee's Completed Tasks 3. All Active Tasks 4. Project's Active Tacks 5. Project's Completed Tasks
Validity	Not only do employees need to be able to see what tasks they are currently working on, but they also need to be able to see what tasks they have completed for the day to judge how productive they have been. It is also important to be able to see what tasks are being worked on for each project, and for project managers and admins to be able to see how much work is being tracked each day.
Consistency	This requirement doesn't conflict with any other specified requirement.
Feasibility	This will be possible by storing the tasks in the database for each project and applying queries to receive the proper lists of tasks.
Error Handling	There should not be any errors here as only tasks that are found in the database with the correct conditions for each list will be shown.
Priority	Required

4.1.3 View Bill of Material Items

Description	This feature will show all of the current bill of material items that have been requested and purchased for a specific project. There are a few ways to filter this list by the following conditions: 1. Requested BOM items 2. Ordered BOM Items 3. Delivered BOM Items
Validity	One of the requirements to start working on a project is for the materials of the project to be available. It is easier to start a task if you know the material is ready to be worked on. Also, one of the major things to be tracked in this system in the amount of money being invested in each project. Tracking the cost of purchased materials will allow that to be possible.
Consistency	This requirement doesn't conflict with any other specified requirement.
Feasibility	This will be possible by storing the BOM items in the database for each project and applying queries to receive the proper lists BOM items.
Error Handling	There should not be any errors here, as only BOM items which are in the database for the specific project will be shown.
Priority	Required

4.1.4 View To-do List

Description	This feature will allow employees to see what the shops main goals are. This could include projects the shop is focusing on or tasks that are imperative to be completed soon.
Validity	This feature allows for BPTD to easily notify their employees about to the current priorities of the shop.
Consistency	This requirement doesn't conflict with any other specified requirement.
Feasibility	This can be accomplished by keeping a list of to-do items in the database and retrieving them when a user access the dashboard.
Error Handling	There should not be any errors here, as only to-do items which are in the database will be shown. If there are no to-do list items in the database, then the to-do list will not be visible.
Priority	Ideal

4.2 System Functional Features

4.2.1 Start/Pause/Finish Task Items

Description	This feature will allow employees to add and modify their tasks. There will be a few different functionality's surrounding adding and modifying tasks. Those functions are as follows: 1. Start Task 2. Pause Task 3. Complete Task 4. Delete Task
Stimulus Sequence	Each possible function surrounding tasks will have its own user input requirements as follows: • Start Task: User must input the project code if they are not adding the task from the project view. The user must also input a description of the task. • Pause Task: The user must input a reason for pausing the task • Complete Task: The user just has to click complete on the task • Delete Task: The user will just have to click delete on the task
Validity	This feature relates to the main reason the application was proposed. Assuming employees enter tasks properly, this will allow BPTD to easily track the productivity of their employees as well as see how many hours are being invested in each project.
Consistency	This feature does not conflict with any other features, as it is one of the pain purposes for this application.
Feasibility	These features can be implemented by storing the tasks with their corresponding project in the database. The user level as well as the creator of the task will determine if the task can be edited by a user.
Error Handling	We will force the user to input all of the required fields when using any of the functions above. This will hopefully prevent any errors that could occur.
Priority	Required

4.2.2 Add/Remove To-do List Items

Description	This feature will allow a user with the correct user level to add and remove items from the to-do list, which will be viewable on every user's dashboard. The two functions these users will have access to include: 1. Adding To-do List Items 2. Removing To-do List Items
Stimulus Sequence	Each of the feature functions above will have its own user input requirements as follows: • Add to-do list item: User must input a description of the to-do list item as well as a priority for that goal. • Remove to-do list item: User will be able to click a delete button on the list item to remove it.
Validity	To have a to-do list available, there needs to be a way to add and remove items from the list as the priorities of the shop changes over time.
Consistency	This requirement doesn't conflict with any other specified requirement.
Feasibility	This will be possible by modifying the list of items in the to-do list which is stored in the database.
Error Handling	There should be no chance for an error as we will require the user to enter all of the specified inputs to continue.
Priority	Ideal

4.2.3 Create/Modify Project Details

Description	This feature will allow employees of the correct user level to create projects for BPTD. The two possible functions these users will be able to take are as follows: 1. Add a Project 2. Modify the Project Details

Stimulus Sequence	Each of the feature functions above will have its own user input requirements as follows:
	• When adding a project, the user will have to input the following details:
	  ○ Project Name
	  ○ Customer
	  ○ Project Type
	  ○ Project Manager
	  ○ Budgeted hours for the project
	  ○ Project Description
	• When modifying the project details, the user will be able to edit any of the details above, besides project type, and will also be able to edit the following additional detail:
	  ○ Status (Whether the project can or cannot be worked on yet.

Validity	Since one of the reasons Right-on-Time is being developed is to track how much time

	employees are putting into a project, there needs to be a way to track projects. This will allow the owners of BPTD to understand better what projects are being worked on.
Consistency	This feature does not conflict with any other features, as it is one of the pain purposes for this application.
Feasibility	Each project will get its own entry in the database with the project details above, as well as lists of tasks and BOM items that are assigned to the project. Since this feature is one of the core pillars of Right-on-Time, many other features will be based on this implementation.
Error Handling	When adding projects, the information above will be required before submission, which should prevent any possible errors.
Priority	Required

4.2.4 Request/Modify Bill of Material Items

Description	This functionality will allow users, of the correct user level, to request bill of material items for specific projects. The possible functions are as follows: 1. Request BOM items 2. Modify the status of BOM items
Stimulus Sequence	Each of the feature functions above will have its own user input requirements as follows: • When requesting a BOM item, the user must have a user level higher than project manager and must enter a description and estimated cost of the requested item. • When modifying the BOM status, the user must have a user level of admin or purchasing and must choose a status from the following list of options: o Requested o Ordered o Delivered The user may also change the cost of the BOM item if the actual cost was different from the estimated cost.
Validity	Since one of Right-on-Time's goals was to track the amount of money going into projects, this features is rather imperative.
Consistency	This feature does not conflict with any other features, as it is one of the pain purposes for this application.
Feasibility	The BOM list will be stored in the database under the project which it belongs to. In other words, each project will have a list of BOM items associated to it.
Error Handling	When adding or modifying the status of BOM items, the information above will be required before submission, which should prevent any possible errors.
Priority	Required

4.2.5 Add/Remove Employee

Description	This feature will allow a user, of the user level admin, to add and remove users from the database of company employees. The two functions of this feature include the following: 1. Add user to the Right-on-Time user database

	2. Remove user from the Right-on-Time user database
Stimulus Sequence	Each of the feature functions above will haves its own user input requirements as follows: • When adding a user, the following information will be required: o Name o Username o User level o Temporary password • When removing a user, the admin will have to authenticate with their password.
Validity	It is very imperative of the application to track different users so that quality information about BPTD can be monitored. For this to be possible, it must also be possible for users to be added and removed from Right-on-Time. It is also important to restrict users to functions specified by BPTD, and user accounts with user levels will allow for that functionality.
Consistency	This feature does not conflict with any other features.
Feasibility	This feature will be accomplished by having a list of users and their information in the database.
Error Handling	When adding or removing users, the above inputs will be required which should reduce the chance of anything going wrong.
Priority	Required

4.2.6 Change Employee User Level

Description	This feature will allow a user, of the user level admin, to modify another employee's user level. The only function involved with this feature is as follows: 1. Change an Employee's User Level
Stimulus Sequence	The functionality above will have the following input requirements: • When changing an employee's user level, the admin must choose from the following list of user levels: o Administrator o Shop manager o Project manager o Purchasing o Employee
Validity	To restrict what users will be able to do within Right-on-Time, there needs to be a way to track what their access level is. This method of assigning a user level to each employee will give BPTD that functionality.
Consistency	This feature does not conflict with any other features.
Feasibility	This will simply be done by changing the user level field in the database for the user in question.
Error Handling	When modify an employee's user level, the administrator will be required to choose from the list of user levels defined above. This should prevent any errors.
Priority	Required

4.2.7 Change Password

Description	This feature will allow users to change their password. The only function involved with this feature is as follows: 1. Change a user's password
Stimulus Sequence	The functionality above will have the following input requirements. • The user must enter their new password two times
Validity	This is a function that is almost always required when dealing with user logins. It is imperative for the users to be able to change their password for security reasons alone. Also when a new employee is created, the admin knows their password, so the new employee needs a way to change it.
Consistency	This feature does not conflict with any other features.
Feasibility	This will be done by first checking to make sure the two new password inputs are the same, then modify the password field for the user in the database.
Error Handling	The only error is if the two password inputs are not the same, then the application will warn the user and let them try again.
Priority	Required

5. Other Nonfunctional Requirements

5.1 Performance Requirements

There are not too many performance requirements for Right-on-Time. Since this application does not deal with real time data, such as data that is changing at a rapid speed, timing will not be as important. That being said, Right-on-time should run fast enough that the user never notices any lag time between when they use a functionality and the results appear to them. Right-on-Time should also be able to run non-stop for an indefinite length of time. Finally, the database queries should be simple enough to not require a large amount of data processing to function.

5.2 Safety Requirements

There are no safety concerns for users of Right-on-Time. The application will not affect the user besides requiring them to spend time entering data.

5.3 Security Requirements

Since the data that Right-on-Time will be handling for BPTD is essential to their business, a few security requirements will be enforced. Those requirements are as follows:

1. No person will be able to login to Right-on-Time without a valid user account
2. All data will be stored in a database that is only accessible and modifiable based on the user's level.
3. If requested, we can implement a system to prompt users to log in again if they have not recently authenticated with Right-on-Time.

5.4 Software Quality Attributes

There are a few quality Attributes that we will take into consideration while developing Right-on-Time. Those attributes are as follows:

Portability: Right-on-Time will be able to run on any server that which can host web applications. It will also be possible to create an instance of the application and transfer it to another server.

Reliability: Right-on-Time should be able to run without hitch for an indefinite amount of time, as long as proper maintenance has been conducted. Also, the front end user interface should work on most modern browsers without breaking.

Maintainability: It should be possible for Right-on-Time to be maintained without much effort from Breedt Production Tooling and Design. All issues that occur must be easy to fix for someone without access to the source code. Any issues that require changes to the code will probably need to go through one of Right-on-Time's developers.

Usability: Right-on-Time should have a consistent user interface that allows users to very easily pick up the software and start adding tasks to projects. There should be little training required for users to operate Right-on-Time.

Scalability: Right-on-Time should never slow or fail to work as the amount of projects and data begins to expand over time. In other words, Right-on-Time's speed and functionality should never depend on the amount of data stored in its database. Also, it should be possible to add features to Right-on-Time without breaking the current features.

Other Requirements

The developers of Right-on-Time will implement the application and provide resources to the employees of Breedt Production Tooling and design as they are learning the tool.

Appendix: Glossary

BOM stands for Bill of Material
BPTD stands for Breedt Production Tooling and Design.
"User" and "Employee" are usually synonymous for a user of the software, except when referring to user levels.
Right-on-Time is the name of the application.