SOFTWARE REQUIREMENTS SPECIFICATION

for

Online Postage Ordering System

Release 0.1.0

Version 0.1.0 approved

Prepared by Team Moon Moon

Team Moon Moon

Contents

1 Introduction **7**
- 1.1 Purpose . . . 7
- 1.2 Project Scope and Product Features . . . 7
- 1.3 Defintions, Acronyms, and Abbreviations . . . 7
- 1.4 References . . . 7
- 1.5 Overview . . . 7

2 Overall Description **8**
- 2.1 Product Perspective . . . 8
- 2.2 Product Functions . . . 8
  - 2.2.1 Create An Account . . . 8
  - 2.2.2 Print Postage . . . 8
  - 2.2.3 Buy Postage . . . 8
  - 2.2.4 List Postage Transactions . . . 8
  - 2.2.5 Get Postage Balance . . . 8
  - 2.2.6 Calculate Postage . . . 8
  - 2.2.7 Track/Confirm Package . . . 8
  - 2.2.8 Login . . . 9
  - 2.2.9 Refund Request . . . 9
  - 2.2.10 Validate Address . . . 9
- 2.3 User Characteristics . . . 9
- 2.4 Constraints . . . 9
- 2.5 Assumptions and Dependencies . . . 9

3 Specific Requirements **10**
- 3.1 External Interface Requirements . . . 10
  - 3.1.1 System Interfaces . . . 10
  - 3.1.2 User Interfaces . . . 10
  - 3.1.3 Hardware Interfaces . . . 10
  - 3.1.4 Software Interfaces . . . 10
  - 3.1.5 Communication Interfaces . . . 10
- 3.2 Functional Requirements . . . 10
  - 3.2.1 Create an Account . . . 10
  - 3.2.2 Print Postage . . . 11
  - 3.2.3 Buy Postage . . . 11
  - 3.2.4 List Postage Transactions . . . 11

3.2.5 Get Postage Balance 11
3.2.6 Calculate Postage 11
3.2.7 Track/Confirm Package 11
3.2.8 Login 11
3.2.9 Refund Request 11
3.2.10 Validate Address 11
3.3 Performance Requirements 11
3.4 Logical Structure of the Data 11
3.5 Design Constraints 12
3.6 Software System Attributes 12

Revision History

1. Team Moon Moon Needs No Revisions.
2. Team Moon Moon Never Makes No Mistakes.
3. Correction. Team Moon Moon "fails fast." Isn't that the new buzzword or buzzphrase?
4. The skeleton has been updated to reflect the actual requirements provided by Dr. Seaman.

1 Introduction

Team Moon Moon has been tasked with developing a SRS for their Software Engineering class at Texas State University. If Team Moon Moon doesn't do this project, Team Moon Moon will fail D: Team Moon Moon wants to graduate already and doesn't feel like taking this class again.

1.1 Purpose

The purpose of this project is to get our C. Because C's get degrees!

1.2 Project Scope and Product Features

BLAH BLAH BLAH

1.3 Defintions, Acronyms, and Abbreviations

What is the airspeed velocity of an unladen swallow?

1.4 References

OMG OMG OMG

1.5 Overview

Aren't "Introduction" and "Overview" kinda the same thing? Just sayin'...

2 Overall Description

Many description. Much adjective. Wow.

2.1 Product Perspective

Such perspective. So edgy.

2.2 Product Functions

BRO DO YOU EVEN FUNCTION?

2.2.1 Create An Account

Lol please shoot me now nobody does software development like this anymore.

2.2.2 Print Postage

DAE hate waterfall?

2.2.3 Buy Postage

TO THE MOON!!!

2.2.4 List Postage Transactions

herpderp

2.2.5 Get Postage Balance

has anyone ever been so far as even decided.

2.2.6 Calculate Postage

A Stamp? What's a stamp, grampa?

2.2.7 Track/Confirm Package

Here, you can incessantly check on the state of your package inbetween meals and trips to the loo! Never be out of the loop again. You really needed to know that your package left the seller's facility.

2.2.8 Login

I can't let you do that Dave.
I can't let you do that, Dave.
Commas.

2.2.9 Refund Request

It's been denied.

2.2.10 Validate Address

Validating addresses is actually a royal PITA. Just let Google Maps API handle it.

2.3 User Characteristics

Our users are in the 19-29 demographic. They love iProducts of all sorts and can't wait to buy postage stamps with our hip new website!

2.4 Constraints

Batman has no constraints.

2.5 Assumptions and Dependencies

I'm starting to run out of steam here guys.

3 Specific Requirements

As opposed to general requirements.

3.1 External Interface Requirements

aryweguiahergearpguhestrgperasguheasrgpewuhr

3.1.1 System Interfaces

This product will interface with a Gibson. You know? Like, supercomputers they use to do physics and search for oil? Wouldn't you love to get one of those Gibsons?

3.1.2 User Interfaces

CLI fo LYFE. Firefox is for nubs. Real haxors use lynx!

3.1.3 Hardware Interfaces

awrygaeghsrethsegwearfawfewaf. I feel better now.

3.1.4 Software Interfaces

gbsetghbsetgesrgfawefawefwaefawfawefawefwaf. Much better.

3.1.5 Communication Interfaces

Does the internet constitute a communication, software, or hardware interface? All of the above? All of the others? Some? Who knows.

3.2 Functional Requirements

Ok. Now I'm really running low on ideas.

3.2.1 Create an Account

The system shall spam you. Violently.

PLEASE GIVE US YOUR EMAIL SO WE CAN SPAM YOU! I mean... OFFER YOU EXCLUSIVE PROMOTIONS ON INSIDER DEALS!

3.2.2 Print Postage

The system shall print your postage. Ironically.

3.2.3 Buy Postage

The system shall take your money and run. Maniacally.

3.2.4 List Postage Transactions

The system shall list all the times it serviced your account. Suggestively.

3.2.5 Get Postage Balance

The system shall present your balance. And laugh.

3.2.6 Calculate Postage

The system shall calculate how much postage is required to mail your package. Too much.

3.2.7 Track/Confirm Package

The system shall give you your tracking number. A day later.

3.2.8 Login

The system shall let you in upon uttering the magic word. Please.

3.2.9 Refund Request

The system shall take your request and pipe it to /dev/null. Cheerfully.

3.2.10 Validate Address

The system shall throw its hands in the air and just flip a coin to determine whether the address is valid.

3.3 Performance Requirements

Don't fail.

3.4 Logical Structure of the Data

That would imply "logic" and "structure," neither of which are currently present. The logic is back-ordered and the structure was never there to begin with.

In any case, this is where the main diagram goes.

3.5 Design Constraints

Why would we put design constraints all the way at the bottom? You'd think the design constraints might illuminate pretty much all of the subsequent design decisions that were made and expounded upon in the entire SRS...

3.6 Software System Attributes

Is it at-tri-bute? Or a-trib-ute? Because one is a noun and one is a verb. Yeah, it's kinda the English language's way of saying FUCK YOU to foreigners.