**Eagleslist**

**Software Requirements Specification**

**CEN 3031, Fall, 2015**

**Modification history:**

|  |  |  |  |
| --- | --- | --- | --- |
| Version  | Date  | Who  | Comment  |
| v1.0  | 09/17/2015  | Michael MacCallum | Initial content added and form filled out. |
| ...  |  |  |    |

**Team Name:** 5Bit Studio

**Team Members:**

* Mostofa Ansary – msansary6258@eagle.fgcu.edu
* Hermilo Calderon - hcalder@eagle.fgcu.edu
* Michael MacCallum - msmaccallum5438@eagle.fgcu.edu
* Andrew Owen – adowen5642@eagle.fgcu.edu
* John Zachariah – jfzachariah3593@eagle.fgcu.edu
* <5-bit.github.io>

**Contents of this Document**

[Introduction](#Introduction)

* [Software to be Produced](#SoftwareToBeProduced)
* [Definition, Acronyms, and Abbreviations](#DefinitionsAcronymsAbbreviations)

[Product Overview](#ProductOverview)

* [Assumptions](#Assumptions)
* [Use Case Diagram](#UseCaseDiagram)
* [Use Case Descriptions](#UseCaseDescriptions)

[Specific Requirements](#SpecificRequirements)

**SECTION 1: Introduction**

Software to be Produced:

Eagleslist is a textbook focused trading program for FGCU students that will allow users to make accounts, verify their student email addresses and post used textbooks and other goods that they’re looking to buy or sell. We plan on adding special features for textbooks, like the ability to lookup books by CRN, or ISBN. Professors will also be able to verify their faculty email addresses and have the ability to supply the ISBNs for books that they will require for specific classes. Students will be allowed to comment and/or review course books to rate them by relevance/necessity/etc. We will not be handling money, all transactions will be determined by the students who buy and sell products on our service.

 Definitions, Acronyms, and Abbreviations:

* <include any that are needed to read this document (or "None" if document is self-explanatory and no acronyms or abbreviations will be used>
* GUI
* IDE
* SQL
* DB

**SECTION 2:** **Product Overview**

Assumptions:

* <List all the assumptions the developers are making. For example: assumptions about other systems this product will interface with; assumptions about the technological environment in which the product will operate (how much memory, what type of processor, ...); assumptions about availability and capability of COTS, GOTS, or other re-used products, ...>
* ...
* The desktop application will require
	+ Windows 7+ to run.
	+ Communication with a webservice, and will therefore require an internet connection to operate.
	+ A system will at least 1GB of RAM, and at least a 1 GHz processor.
	+ C# and .NET 4.5
	+ Google Books API
* Web Service
	+ PostgreSQL
	+ Go
	+ Ubuntu 14.04
	+ Digital Ocean
* Journal Website
	+ HTML
	+ CSS
	+ Javascript
	+ Jekyl
	+ Liquid

Use Case Diagram:

* <Include a use case diagram here. see materials presented in class and textbook>

Use Case Descriptions:

* <Briefly describe each use case included in the above diagram. Please make sure you describe the main steps of each use case, making sure that exceptions are considered. For reference, see materials presented in class and textbook >

**SECTION 3: Specific Requirements**

**< The following template must be used for each requirement; you should have about 15 requirements total: >**

|  |
| --- |
| **No:** <Each requirement must have a unique requirement number> |
| **Statement:** <the "shall" statement of the requirement; each requirement must be specific, unambiguous, precise, and testable (this means it should be obvious during product testing if the product satisfies this requirement (passes) or not) >  |

(need 15-20 of these)

* The system shall be able to store information entered by users about textbooks.
* The system shall be able to verify whether or not a user is a FGCU student by requiring them to enter their FGCU email address, and click a link in a verification email.

Template created by G. Walton (GWalton@mail.ucf.edu) on Aug 30, 1999 and last updated Aug 15, 2000; updated by A. Koufakou, Aug 2014

This page last modified by <your name here> (<your e-mail address text and link here> ) on <modification date here>