

# Software Requirements Specification

**for**

**SubScruple**

**Version <1.0>**

**Prepared by Kevin Black, Alex Sloan, Shane Bellika**

**Group Name: BBS teemMaets**

**Shane Bellika  
Kevin Black  
Alex Sloan**

**11639152  
11480710  
11651363**

**shane.bellika@wsu.edu  
kevin.black@wsu.edu  
alex.sloan@wsu.edu**

**October 7, 2018**

# Contents

<b>REVISIONS.....</b>	<b>III</b>
<b>1 INTRODUCTION.....</b>	<b>1</b>
1.1 DOCUMENT PURPOSE .....	1
1.2 PRODUCT SCOPE .....	1
1.3 INTENDED AUDIENCE AND DOCUMENT OVERVIEW .....	1
1.4 DEFINITIONS, ACRONYMS AND ABBREVIATIONS.....	2
1.5 DOCUMENT CONVENTIONS .....	2
1.6 REFERENCES AND ACKNOWLEDGMENTS .....	2
<b>2 OVERALL DESCRIPTION.....</b>	<b>3</b>
2.1 PRODUCT PERSPECTIVE.....	3
2.2 PRODUCT FUNCTIONALITY .....	3
2.3 USERS AND CHARACTERISTICS .....	4
2.4 OPERATING ENVIRONMENT.....	4
2.5 DESIGN AND IMPLEMENTATION CONSTRAINTS.....	4
2.6 USER DOCUMENTATION.....	5
2.7 ASSUMPTIONS AND DEPENDENCIES .....	5
<b>3 SPECIFIC REQUIREMENTS.....</b>	<b>6</b>
3.1 EXTERNAL INTERFACE REQUIREMENTS .....	6
3.2 FUNCTIONAL REQUIREMENTS .....	6
3.3 BEHAVIOUR REQUIREMENTS.....	9
<b>4 OTHER NON-FUNCTIONAL REQUIREMENTS.....</b>	<b>10</b>
4.1 PERFORMANCE REQUIREMENTS .....	10
4.2 SAFETY AND SECURITY REQUIREMENTS.....	10
4.3 SOFTWARE QUALITY ATTRIBUTES .....	10
<b>5 OTHER REQUIREMENTS.....</b>	<b>12</b>
<b>APPENDIX A – DATA DICTIONARY.....</b>	<b>13</b>
<b>APPENDIX B - GROUP LOG .....</b>	<b>14</b>

## Revisions

Version	Primary Author(s)	Description of Version	Date Completed
<1.0>	Kevin Black, Alex Sloan, Shane Bellika	Title page and Section 1 of the Software Requirements Specification completed, Revisions section will be continually reformatted as new versions release.	09/28/2018
<1.1>	Kevin Black	Section 2 of the Software Requirements Specification completed.	09/29/2018
<1.2>	Kevin Black, Shane Bellika	Section 5, Appendix A, and Appendix B of the Software Requirements Specification completed.	09/30/2018
<1.3>	Kevin Black, Alex Sloan, Shane Bellika	Section 3 of the Software Requirements Specification completed.	10/01/2018
<1.4>	Kevin Black, Alex Sloan	Section 4 of the Software Requirements Specification completed, table of contents updated.	10/01/2018

# 1 Introduction

**SubScruple** is a free, subscription service, web-based application that users can purchase subscriptions through different services offered in their respective location. These subscriptions can be bought at different tier levels and provide extra benefits to the user depending on the tier. In this section, you will be introduced to the purpose of this document, the scope of this product, the intended audience/document overview, definitions/acronyms/abbreviations used throughout this document, document conventions, and references and acknowledgements.

## 1.1 Document Purpose

The purpose of this document is to centralize the ideas and functionalities, provided by each group member, that ultimately define how the system operates. This will consist of an overview of how the system will function, the high-level, web-based user interface, and how this system will resonate with the team members developing said system and the interested audience and/or stakeholders. Revisions made to the document are a possibility depending on the changing vision of the project or the feasibility of certain features, therefore make sure to check the Revisions section on page iii to get an idea of what features are included within the latest version of the system. Reading over this document should provide the reader with a holistic sense of what **SubScruple** does and the benefits it provides to its subscription users.

## 1.2 Product Scope

The particular scope of **SubScruple** is to offer subscriptions for various services that are congregated onto this one application. This application will strive to get the best deals possible by being constantly kept up to date with subscription pricing for the various services that the user chooses. The application can also be thought of as a hub for where all user subscription info will be stored, where they can choose to view their subscriptions by logging in to the application. In an ever-expanding market and reliance on technology for most purchases, each user can be certain to receive some benefit out of this web application.

## 1.3 Intended Audience and Document Overview

This document will be written in such a way that the reader is assumed to have a knowledge of how software requirements specification documents are written, whether they are developers, project managers, marketing staff, users, testers, or documentation writers. The intended audience who are to operate **SubScruple** are those typically with an interest in technology, online shopping, or online reading, as the offered services will mainly be centered on these three categories.

The rest of this software requirements specification document past section 1 will delve into the nature of the system functionalities, with sections discussing topics like product perspective, the operating environment, the decision making behind the user-interface for the application, design and implementation constraints, specific/non-functional requirements, and documentation, just to name a few. This SRS is designed in such a way as to introduce the average reader to **SubScruple**, introduce basic functionalities to the user, and burrow all the way to the specific functionalities of how the application works, if one chooses to read that far. If you are a reader with none to a slight knowledge in how web applications operate, then feel free to just read the first section along with the basic operations for how the web application will work. Otherwise, if you are a reader with moderate to extensive knowledge in how web applications operate, please read the whole document to have a comprehensive understanding of the different functions of **SubScruple**.

## 1.4 Definitions, Acronyms and Abbreviations

1. WSUV	Washington State University Vancouver: A public university located in Vancouver, Washington.
2. IDE	Integrated Development Environment.
3. GNU GPL	GNU General Public License is essentially a free software license which allows end users to execute or modify software under this license.
4. CS 320	Fundamentals of Software Engineering, under which this SRS and application is developed at Washington State University under the further tutelage of Professor Xinghui Zhao.
5. SRS	Software Requirements Specification.

## 1.5 Document Conventions

This software requirements specification document follows the writing conventions of the IEEE formatting requirements which includes but is not limited to: the use of size 11/12 Arial font, italics are used for comments, each line of the document is single spaced, the document maintains one inch margins, and the main sections begin at the top of it's own page with a single number followed by subsections with decimal separator format that can span over multiple pages.

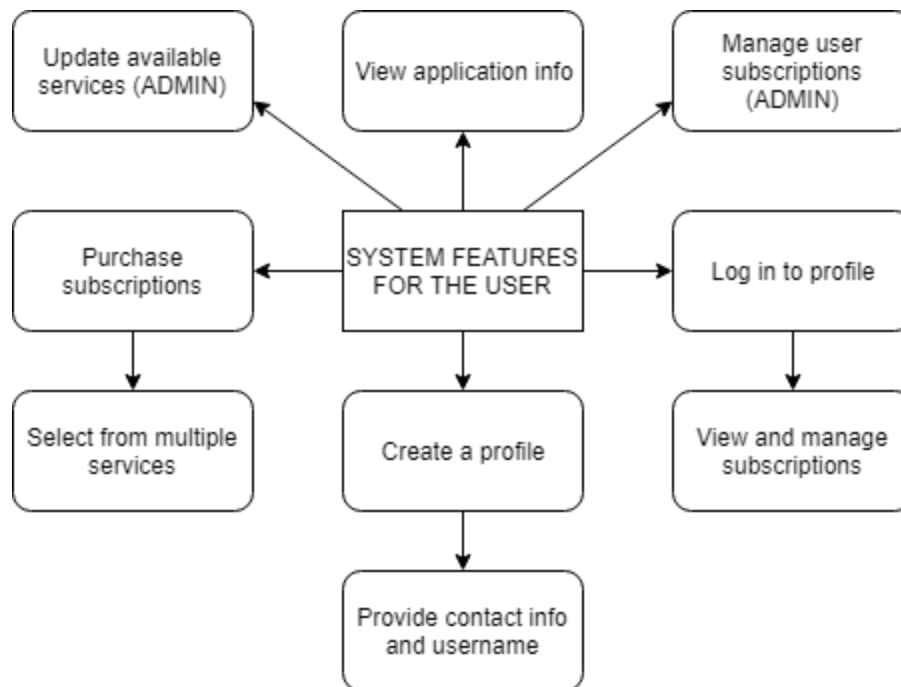
## 1.6 References and Acknowledgments

- Standard IEEE Citation Guide: [https://learn.wsu.edu/bbcswebdav/pid-2795651-dt-content-rid-91333015\\_1/courses/2018-FALL-VANCO-CS-320-7661-LEC/IEEE-Citation-StyleGuide.pdf](https://learn.wsu.edu/bbcswebdav/pid-2795651-dt-content-rid-91333015_1/courses/2018-FALL-VANCO-CS-320-7661-LEC/IEEE-Citation-StyleGuide.pdf)
- Slide reference material provided by Professor Xinghui Zhao of WSUV: [https://learn.wsu.edu/webapps/blackboard/content/listContent.jsp?course\\_id= 189425\\_1&content\\_id= 2750588\\_1&mode=reset](https://learn.wsu.edu/webapps/blackboard/content/listContent.jsp?course_id= 189425_1&content_id= 2750588_1&mode=reset)
- Application coding standard (for JavaScript): <https://eslint.org/>
- Application development environments:  
JSFiddle: <https://jsfiddle.net/>  
IntelliJ: <https://www.jetbrains.com/idea/?fromMenu>

## 2 Overall Description

### 2.1 Product Perspective

As the name may entail, **SubScruple** was formed out of the idea of a subscription based business model that concentrates the subscription purchases of every user in one place. The aim of this product is to collect subscription data from the user and store it internally in such a way that reduces the time and complexity of managing subscription records. This product is simply not a follow-on member of a higher product family, but a new, self-contained product with inspiration drawn from current subscription based models/applications. The following high level diagram showcases all of the features that are to be implemented for the system.



### 2.2 Product Functionality

The ideal goal of **SubScruple** is to make it easy for users to manage current subscriptions/purchase new subscriptions of various services. This will be done through a variety of functions that the user can perform, including but not limited to:

- Returning to the website will prompt the user to log in or create an account.
- If the user doesn't not have any login information to provide, they can create a free profile so as to actually have an account for managing/purchasing subscriptions.
- Purchasing subscriptions from a variety of services, mainly from other websites that contain some form of e-commerce or digital entertainment. The services will also contain URL's to the websites if they wish to have a more indepth look at what kinds of additional benefits the specific service provides.
- Able to view the application info, with information such as: the developers who created the application and their roles, the mission of the application, contact info and location of the developers, and so on.

- If a user logs in and they have administrative privileges, they have the additional benefits of being able to update available services offered by **SubScruple** or being able to manage normal user subscriptions through the internal database.

## 2.3 Users and Characteristics

Generally speaking, there will be a few different, yet similar, classes of users who will operate **SubScruple**: the user who uses this application to keep track of the various subscriptions they currently own, the user who uses this application to purchase subscriptions through different tiers so as to receive extra benefits of some sort, the user who is a combination of the previously mentioned ones, and an administrator who oversees the database of user subscriptions and makes sure the integrity of the application is not compromised. The average user who comes across this application will more than likely have some knowledge when it comes to online purchases and managing digital subscriptions, so the user interface and functionalities are to be developed in such a way that it is to be assumed that the user doesn't have to be initially guided through the process of how to use this application. If the user wishes to know what they can do through this application, they have the option of consulting the 'About' section of the application. Yet at the same time, the characteristics of this application will not be so complicated that a new user will be overwhelmed as how to navigate the system, and will be given a simplistic, slightly modern style.

## 2.4 Operating Environment

Running this application should not be too intensive of a workload for modern computers to handle, although the recommended system requirements are listed below.

### User Requirements – Hardware:

- Operating System: Windows 8 or higher/macOS 10.7 or higher
- CPU: at least 400 MHz
- RAM: at least 512 MB

### User Requirements – Software:

- Browser: Google Chrome, Mozilla Firefox, Microsoft Edge, or Apple Safari.

## 2.5 Design and Implementation Constraints

As this application is in the process of being developed while the developers are attending university, the application may only be updated as a result of this application only being required for the CS 320 course at WSUV. Therefore system rollouts and system validation will more than likely only be executed as a result of what's required in the relative time frame of the project. The constraint of also finding instances to collaborate and actively further develop the application will also be limited to the fact that the developers may possibly be preoccupied with other projects at the time. On the development side of the application, **SubScruple** will be developed using the ESLint tool and the Airbnb coding standard mainly through the JSFiddle IDE. In regards to hardware or software limitations and as mentioned in section 2.4, any modern computer created past 2010 should have little to no issue running this application. As the current state of the database is private to just to the developers at the moment, the amount of users to interact with this application may be limited to around five users for testing purposes, and work its way up to however many users the developers see fit. The application is to be designed only in the English language and will therefore possess a user interface only depicting the English language. As for browser limitation, most computer users access the internet through the stated browsers in section 2.4, so the application will only be accessible through those browsers.

## 2.6 User Documentation

**SubScruple** will not contain any extensive user manuals or much dedicated tutorials from the developers to aid the user in how to navigate and operate the application, with the reasonings being listed in section 1.3 and section 2.3. However, if help is needed for how certain parts of application work or how to perform a certain action, there will be an 'About' section listed on the website that will provide a general overview on what to do and how to operate the application. Additionally, any further questions can be redirected to the emails of the developers, which can also be found at the bottom of the 'About' section. The administration team and developers of **BBS teemMaets** may support the evolution of the system after the university project has concluded, but feel free to contact the developers through the email addresses provided on the cover page of this SRS document or refer to the 'About' section listed on the application if any help is needed.

## 2.7 Assumptions and Dependencies

The development team behind **SubScruple** generally assume that the application will still be accessible post-project amongst other assumptions, however it is not guaranteed that the application will possess a dedicated evolution. Other general assumptions or dependencies can be found listed below:

- Users are assumed to possess an adequate, modern computer that is capable of running this application and possess a stable network connection to access the application at any time.
- The vastness of the application is developed on the assumption that numerous services are able to be discovered, obtained, and updated regularly for the duration of the project.
- A basic, stable database of user subscriptions is to be initially provided by the developers and can only truly be accessible by said developers and the administration team, and any efforts made by unauthorized users to access this database will be dealt with swiftly relative to the magnitude of the issue.
- Typical users are assumed to just operate the service for its intended purpose, although if any users wish to perform some quality assurance efforts, any bug reporting can be sent directly to the developers through the given emails on the front cover of the SRS document and will try to be fixed as soon as possible. The development team depends on user feedback to sustain and improve the application, so all opinions about **SubScruple** are welcome.
- Whether a normal user, a developer, or a part of the administration team, it is assumed that every person that interacts with this system will do so in a non-malicious manner.



## 3 Specific Requirements

### 3.1 External Interface Requirements

#### 3.1.1 User Interfaces

There will be two main user interfaces, one that is designed for the average user, and one that is designed for a user with administrator privileges, which will be essentially the same in look but will contain an identifier somewhere on the screen specifying that the current user is an administrator. The introduction user interface will task the user with creating an account or to log in to an existing profile with 'create profile' or 'log in' buttons, respectively. Other user interfaces include: displaying the homescreen that showcases different services offered for subscriptions with 'purchase' and 'profile' buttons, an interface that allows users to manage their profile with 'edit', 'save', and 'home' buttons, an interface that allows users to manage their subscriptions with a 'delete' button, and interfaces that allows administrators to access the available services or the subscription information of the users with 'edit' and 'save' buttons.

#### 3.1.2 Hardware Interfaces

- Display Monitor: Basic piece of hardware to have information displayed to stdout.
- Input Devices: A properly functioning keyboard and mouse are sufficient enough pieces of hardware to operate this application.
- CPU Usage: Necessary for carrying out the instructions of the application.
- RAM Usage: Necessary for current use of data and machine code that is to be processed by the application.
- Network Usage: The user won't require top-of-the-line internet speed to run this application, so a simple computer to internet connect will suffice.

#### 3.1.3 Software Interfaces

As far as the development process goes, the **SubScurple** application will contain only an uninterrupted integration with the operating/hardware system. While and after the user creates an profile or logs in to the system, each interface accessed by the user will operate the local input, output, and/or operating system devices as needed by the application.

#### 3.1.4 Communications Interfaces

**SubScurple** will mainly utilize HTTP protocol for communication over the internet for most of the application's functions. This can be done through the optimal web browsers Google Chrome, Microsoft Edge, Mozilla Firefox, or Apple Safari. Users may need to have pop-up windows enabled and to make sure that JavaScript files are able to run properly on their respective browsers if there are any issues with application functionality or continuity. There should be little to no issues for operating this application when it comes to modern download/upload data transfer rates, as well as not requiring a direct communication to a security software.

### 3.2 Functional Requirements

The following subsections contain the functions that are to be encapsulated within the **SubScurple** application. The order of these subsections is written in such a manner of how we assume the user will go through the available features of this application sequentially, but they are free to go in

whatever order they wish. Each subsection will describe any inputs or outputs required by the function. Refer technical information can be found within Appendix A – Data Dictionary on page 13.

### 3.2.1 Introduction

#### 3.2.1.1 Accessing SubScruptle

- The system shall be accessed by a shareable link provided by the developers after initial development has concluded and version 1.0 of **SubScruptle** is released.

#### 3.2.1.2 Account Creation

- The system shall display a window prompting the user to make an account or log in.
- The system shall allow the user to create an account if they possess no account or it is their first time on the application.
- The system shall allow the user to input their contact information in various fields needed for account creation.
- The system shall display a message saying that their account was successfully created.
- The system shall associate a randomly generated, unique 8-digit ID for the user.
- The system shall display an error message if there is an issue with creating an account with the application.

#### 3.2.1.3 Account Login

- The system shall display a window prompting the user to make an account or log in.
- The system shall allow the user to input their username and password to safely access their account.
- The system shall display the main interface of the application if log in is successful.
- The system shall display an error message if there is an issue with logging in.

### 3.2.2 Application Info

#### 3.2.2.1 About

- The system shall display information regarding the mission statement of the application.
- The system shall display information regarding how to contact the developers.
- The system shall display information regarding how to operate and navigate the various functions of the application.

### 3.2.3 Menu

#### 3.2.3.1 Displaying Current Offered Services and Tiers

- The system shall display various service subscriptions offered to the user for optional purchase.
- The system shall display various tiers of the services offered to the user for optional purchase.
- The system shall log out the user and return to the log in screen after 10 minutes of inactivity has elapsed.

### 3.2.4 Subscriptions

#### 3.2.4.1 Purchasing Subscriptions

- The system shall display a different interface showing the various service subscriptions offered by **SubScruptle**.
- The system shall allow the user to search through various service subscriptions offered by **SubScruptle**.

- The system shall allow the user to select which service subscription and tier they wish to purchase.
- The system shall display a checkout window allowing them to enter their payment and billing information for purchase.
- The system shall return to the main interface after the checkout process is complete.
- The system shall display a confirmation message that their subscription was successfully purchased.
- The system shall display an error message if there was an issue with purchasing their specified subscription.

#### **3.2.4.2 Manage Subscriptions**

- The system shall display a different interface showing the subscriptions purchased by the user.
- The system shall display information about a subscription when selected.
- The system shall display an unsubscribe button next to each subscription.
- The system shall allow a user to unsubscribe to any subscription they wish.
- The system shall display a running total for how much the user spends monthly and yearly on all of their subscriptions.

#### **3.2.4.3 Manage User Subscriptions**

- The system shall allow a user with administrative privileges to delete a subscription from a specific user in their database.

### **3.2.5 Services**

#### **3.2.5.1 Update Available Services**

- The system shall allow a user with administrative privileges to add a new service and associated tiers to their available services.
- The system shall allow a user with administrative privileges to delete a service and associated tiers from their available services.



## 4 Other Non-functional Requirements

### 4.1 Performance Requirements

Given the small workload of the application and the scale of the accompanying databases, the response time of various application functions should not take longer than a few seconds to complete, whether the current step in the system ended up passing or failing. In the scenario where a wealth of users operate the application at once, there may be slight delay in response time, but the latter should more than likely be ideal. As for user interaction with specific buttons of the system, action-response time should contain no immediate delays. The operating system should not have a large impact on application performance as well, but the browser on which the application operates may influence delays depending on browser choice and upload/download speeds.

### 4.2 Safety and Security Requirements

#### 4.2.1 Safety Requirements

- Simply ensure that your internet connection to the application is stable so as to prevent the loss of progress.
- In the event of a lost transaction or other application issues, please refer to the 'About' section on the application and contact the developers directly through email.

#### 4.2.2 Security Requirements

- The application takes the necessary steps to make sure the accounts of any authenticated user is kept from malicious intent.
- **SubScruple** developers will ensure the privacy of their users stay concealed from general view and that each part of the system performs their instructions to the fullest extent of the application's/developers intentions.
- The integrity of the system is in full control by the team behind the system, so any unauthorized alteration or reallocation of application data will be swiftly resolved to the best of the abilities of the developers and/or administrators alike.

### 4.3 Software Quality Attributes

#### 4.3.1 Adaptability

- The system contains relatively simple adaptability as the application will have a general sense of behavior to each individual since most input from users, for most of the functions, should be about the same. In the case of an error from gathering information about a user or it's environment, the application will address and not progress until the error is resolved. The previous statement can also be applied to the robustness of the system.

#### 4.3.2 Usability

- The system usability is one of the primary concerns for the developers at **BBS teemMaets**, which is why the application contains simple to understand user interfaces that makes navigating and maintaining the website a trivial task. In the case of an error with an interface or a clarity issue that needs to be reaffirmed, notify the developers through the 'About' section.

#### 4.3.3 Reliability

- The system reliability will be kept to the highest standard by the developers performing regular checks on the system to ensure that the latter is in good standing

at all times. In the case of a fault in the integrity of part or whole of the system, the administrators will notify the developers, and the developers will expeditiously resolve the problem.

#### **4.3.4 Testability**

- The system stability shall be occasionally reassured through various tests to make sure each function is kept up to an appropriate standard by the developers. Only after the system confirms its stability with reaching test goals, methods, and resources, will any additional functions be made available to the general public. Any issues regarding testability can be directed to the administrators or developers.

#### **4.3.5 Portability**

- The system portability shall be another prioritized attribute of our application, as being able to operate this system through different environments allows a diversity of users. Since all that the application requires is a stable HTTP protocol for communication over the internet for most of the application's functions, most of, if not all, operating systems and environments should be able to execute the application correctly. Any issues regarding portability can be directed to the administrators or developers.

## 5 Other Requirements

### Licensing Requirements

**SubScruple** will be developed under a GNU GPL license and will be open-source following the initial release, assuming the user wishes to partake in the modification of the source code, which will go through validation of the **BBS teemMaets** development team before an official release is put out given the sensitive information contained within the subscription database.

### Legal, Copyright, and Other Notices

**SubScruple** is a copyrighted trademark of **BBS teemMaets** protected and maintained under the project requirements and solely the project members for the CS 320 course of WSUV and the academic integrity standards upheld by WSUV.

## Appendix A – Data Dictionary

Name	Input	Output	Description
createProfile	Stdin: Username, email address, first name, last name, password	Stdout: Confirmation screen, main menu	Creates a profile for the user, adding their info to a database of users on the application.
loginProfile	Stdin: Username, password	Stdout: Confirmation screen, main menu	Checks the database and logs the user into the application if their credentials are located within the database.
about	N/A	Stdout: Application info page	Redirects the user to an information page containing information about the application and contact information for the developers.
purchaseSubs	Stdin: Search bar for different services	Stdout: List of different subscriptions	User can search for specific services, purchase subscriptions through different tiers one at a time, or simply view the different subscriptions offered by <b>SubScurple</b> .
manageSubs	Search bar for different services	Stdout: List of different subscriptions	User can search for specific services, cancel subscriptions one at a time, or simply view their history of subscription purchases.
updateServices (ADMIN Function)	N/A	Stdout: Database of available services	Allows a user with administrator privileges to update the available services offered by <b>SubScurple</b> .
updateSubs (ADMIN Function)	N/A	Stdout: Database of subscriptions	Allows a user with administrator privileges to update the subscriptions of users.



## Appendix B - Group Log

Date	Duration	Members	Description
09/28/2018	20 min	Shane Bellika, Kevin Black, Alex Sloan.	Very quickly discussed the group name, application name, and basic functionalities of the application. Also worked briefly on section 1 together.
Multiple dates	Multiple instances	Shane Bellika, Kevin Black, Alex Sloan.	Most of the brainstorming and SRS updating was done through the applications Discord and GitHub, respectively.