Software Requirements Specification

for

Backstage

Version 1.0 approved

Prepared by Jack V. Daly

Backstage Productions

1 April 2022

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

Backstage 1.0 will be an audience-specific **social media platform**. Consisting of a **client**-facing **web application** and a **database** to store user information and interactions; it is intended to be a platform for artists, fans, and industry professionals to network and conduct business.

## Document Conventions

Section text shall be italicized. Glossary terms shall be bold.

## Intended Audience and Reading Suggestions

Designers should read sections 2.1, 2.2, 2.3, and 3.1, referencing the glossary as necessary. Programmers should start with sections 2.5 and 3.3 then move on to sections 2.2 and 3.1.

Other readers should proceed in the order presented.

## Product Scope

Backstage is intended to be a safe space for artists and the industry behind them.

## References

[1] Google Inc., Chrome. https://www.google.com/chrome/

[2] Mozilla Corporation, Firefox. https://www.mozilla.org/en-US/firefox/

[3] Apple Inc., Safari. https://www.apple.com/safari/

[4] Docker Inc., Docker. https://www.docker.com/

[5] Nginx Inc., NGINX. https://www.nginx.com/

[6] Zend, PHP-fpm. https://www.php.net/manual/en/install.fpm.php

[7] MariaDB Foundation, MariaDB. https://mariadb.org/

[8] IETF HTTP working group, HTTP. https://httpwg.org/specs/

[9] Zend, PHP. https://www.php.net/

[10] [WHATWG](https://www.google.com/search?client=firefox-b-1-d&q=WHATWG&stick=H4sIAAAAAAAAAONgVuLUz9U3MK0wMctexMoW7uEYEu4OAH8RVQcWAAAA&sa=X&ved=2ahUKEwi47cbZ5of3AhUlEEQIHfimBtAQmxMoAHoECHMQAg), HTML. https://html.spec.whatwg.org/

[11] [Yehuda Katz](http://twitter.com/wycats), JSON API. https://jsonapi.org/

[12] Oracle, JavaScript. https://www.ecma-international.org/publications-and-standards/

# Overall Description

## Product Perspective

Graphical user interface, application

Description automatically generatedBackstage will allow each user to register or log in, create a personal profile, post content, and connect with other users of the platform.

Home / Login / Registration

Profiles

Content Feed

Search for Other Users

## Graphical user interface, text, application, chat or text message Description automatically generatedProduct Functions

## User Classes and Characteristics

* Artists / Musicians – Primary audience. These users would be looking to the platform to promote themselves, conduct business with industry professionals and connect with fans.
* Industry Professionals – Secondary Audience. Users looking to network with artists and their fans to establish business relationships. These users may desire additional functionality from the platform to conduct business.
* Fans / General Users – Likely the most common type of user. These individuals are those seeking to connect with their favorite artists and potentially discover new favorites. This group contains the potential customer base of the previous two groups.
* Moderators – These Users will have elevated access to the platform. These users will be acting in the interests of Backstage Productions to review user complaints and enforce user guidelines.

## Operating Environment

This application will be designed to run on all modern **browsers** including Chrome 54 [1], Firefox [2], and Safari [3] versions 60+. Ideally the application should provide the same user experience regardless of **operating system**, as long the **browser** is able to function normally.

## Design and Implementation Constraints

Due to foreign regulations regarding data collection and retention Backstage is intended to be used solely within the continental United States.

Initially the application will have support for US English, but with the intention of including other language support as the user base grows.

Backstage will be built in a **Docker** [4] environment, so the application will be able to be served from any operating system with **Docker** installed, using **containerized** versions of **NGINX** [5], **PHP-fpm** [6], and **MariaDB** [7] which will communicate internally and externally using **HTTP** [8] protocols.

Access will be regulated by a password login system.

## User Documentation

Documentation on software operation will be provided by a help page accessible from the site’s **navigation**.

# External Interface Requirements

## User Interfaces

Text

Description automatically generated with medium confidenceThe Home Page will allow users to register with the site, or login. If logged in, this page will redirect to the Content Feed. After registration users will be directed to a page where they can edit their personal profile and create posts. From the **navigation** panel users can Search for other users, navigate to the Content Feed, view site notifications, or seek help.

## Hardware Interfaces

The Backstage Application is intended to run in the **browser**, so is available to operate on any **operating system** that can install and run a modern web **browser**.

## Software Interfaces

**PHP** [9] 7.4-fpm will be used on a server to send **HTML** [10], **JSON** (per JSON API specifications [11]), and **JavaScript** [12] to the browser using **NGINX** and interact with the **MariaDB** database.

## Communications Interfaces

Standard internet access will be required to reach the site and will be served to the browser via **HTTP**.

Appendix A: Glossary

\* Unless noted otherwise, all definitions paraphrased from Wikipedia - <https://en.wikipedia.org/>

**browser -** A web browser (also referred to as an Internet browser or simply a browser) is application software for accessing the World Wide Web or a local website.

**client -** A client is a piece of computer hardware or software that accesses a service made available by a server as part of the client–server model of computer networks. The server is often (but not always) on another computer system, in which case the client accesses the service by way of a network.

**containerized -** Containerized software will always run the same, regardless of the infrastructure. Containers isolate software from its environment and ensure that it works uniformly despite differences, for instance between development and staging. - https://www.docker.com/resources/what-container/

**database -** A database is an organized collection of data stored and accessed electronically.

**Docker -** Docker is a set of platform as a service (PaaS) products that use OS (**operating system**)-level virtualization to deliver software in packages called containers.

**HTML -** The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser.

**JavaScript -** A programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS.

**JSON -** An open standard file format and data interchange format that uses human-readable text to store and transmit data objects consisting of attribute–value pairs and arrays.

**MariaDB –** Acommunity-developed, commercially supported fork of the MySQL relational database management system (RDBMS), intended to remain free and open-source software under the GNU General Public License. Development is led by some of the original developers of MySQL, who forked it due to concerns over its acquisition by Oracle Corporation in 2009.

**navigation -** A section of a graphical user interface intended to aid visitors in accessing information. Navigation bars are implemented in file browsers, web browsers and as a design element of some web site.

**NGINX -** A web server that can also be used as a reverse proxy, load balancer, mail proxy and HTTP cache.

**operating system -** System software that manages computer hardware, software resources, and provides common services for computer programs.

**PHP -** a general-purpose scripting language geared toward web development.

**PHP-fpm -** An alternative FastCGI implementation for PHP, bundled with the official PHP distribution since version 5.3.3.[229] When compared to the older FastCGI implementation, it contains some additional features, mostly useful for heavily loaded web servers.

**social media platform -** Interactive technologies and digital channels that facilitate the creation and sharing of information, ideas, interests, and other forms of expression through virtual communities and networks.

**web application -** Application software that runs on a web server, unlike computer-based software programs that are run locally on the operating system (OS) of the device. Web applications are accessed by the user through a web browser with an active network connection. These applications are programmed using a client–server modeled structure—the user ("client") is provided services through an off-site server that is hosted by a third-party.

*Graphical user interface

Description automatically generated with low confidence*Appendix B: Analysis Models

Appendix C: Errata Report and To Be Determined List