**Software Requirements Specification**

**for**

**Blogify**

**Version 1.0**

**Prepared by Rishabh, Shouvik, Bhavya**

**Manipal Institution of Technology**

**<19 September 2023>**

Table of Contents

Table of Contents [ii](#__RefHeading___Toc441230970)

Revision History [ii](#__RefHeading___Toc441230971)

1. Introduction [1](#__RefHeading___Toc441230972)

1.1 Purpose [1](#__RefHeading___Toc441230973)

1.2 Intended Audience and Reading Suggestions [1](#__RefHeading___Toc441230975)

1.3 Product Scope [1](#__RefHeading___Toc441230976)

1.4 References [1](#__RefHeading___Toc441230977)

2. Overall Description [2](#__RefHeading___Toc441230978)

2.1 Product Perspective [2](#__RefHeading___Toc441230979)

2.2 Product Functions [2](#__RefHeading___Toc441230980)

2.3 User Classes and Characteristics [2](#__RefHeading___Toc441230981)

2.4 Operating Environment [2](#__RefHeading___Toc441230982)

2.5 Design and Implementation Constraints [2](#__RefHeading___Toc441230983)

2.6 User Documentation [2](#__RefHeading___Toc441230984)

2.7 Assumptions and Dependencies [3](#__RefHeading___Toc441230985)

3. External Interface Requirements [3](#__RefHeading___Toc441230986)

3.1 User Interfaces [3](#__RefHeading___Toc441230987)

3.2 Hardware Interfaces [3](#__RefHeading___Toc441230988)

3.3 Software Interfaces [3](#__RefHeading___Toc441230989)

3.4 Communications Interfaces [3](#__RefHeading___Toc441230990)

4. System Features [4](#__RefHeading___Toc441230991)

4.1 System Feature 1 [4](#__RefHeading___Toc441230992)

4.2 System Feature 2 (and so on) [4](#__RefHeading___Toc441230993)

5. Other Nonfunctional Requirements [4](#__RefHeading___Toc441230994)

5.1 Performance Requirements [4](#__RefHeading___Toc441230995)

5.2 Safety Requirements [5](#__RefHeading___Toc441230996)

5.3 Security Requirements [5](#__RefHeading___Toc441230997)

5.4 Software Quality Attributes [5](#__RefHeading___Toc441230998)

5.5 Business Rules [5](#__RefHeading___Toc441230999)

6. Other Requirements [5](#__RefHeading___Toc441231000)

Appendix A: Glossary [5](#__RefHeading___Toc441231001)

**Revision History**

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

# Introduction

## Purpose

*The purpose of this web-app is to offer a user-friendly digital environment where individuals and entities can effortlessly produce and disseminate blog posts, facilitating seamless communication with their audience, providing tools for content personalization, offering insightful analytical insights, and ensuring accessibility across various devices, all while prioritizing robust security measures and long-term content archiving capabilities.*

## Document Conventions

*Topic headers are represented by bold letters font sized 18 and subtopics by font size 14. Navigation for subtopics have been mentioned for a detailed reading experience along with descriptive subtitles for the same.*

## Intended Audience and Reading Suggestions

*This document is intended towards professors for grading of this project and for users (preferably developers) to get an insight about requirements to run our application or to get an abstract idea on running similar products based on similar ideas and technologies.*

## Product Scope

*The primary objective of this mini project is to make the tedious procedure of hostel booking and management feasible for users avoiding futile confusions and uncertainty regarding the same. This project can be extended to solve management issues for a larger organization and for a larger subset of people offering fresh additional functionalities to ease the cause.*

## References

*For source code and additional resources used in this code:*

* *<https://github.com/RampageousRJ/Blogify>*

*For software dependencies, clone the GitHub repository and using terminal run the command ‘pip install requirements.txt’ and all the dependencies would automatically be downloaded.*

# Overall Description

## Product Perspective

*Blogify functions as a comprehensive content management and communication platform, providing users with intuitive content creation and publication tools. It offers a user-centric interface, adaptable to various user needs, and is designed to be easily integrated with other applications and services through APIs. The application emphasizes scalability and flexibility, allowing for future feature enhancements and customization to accommodate evolving user requirements. It also prioritizes performance, reliability, and security to ensure a seamless and safe experience for both bloggers and their readers.*

## Product Functions

*This project introduces a user classification system, dividing users into two distinct categories: anonymous users and logged-in users. Anonymous users are granted the privilege to read posts, access individual blog pages for reading, and utilize basic reading features. In contrast, logged-in users enjoy extended privileges, including the ability to create posts, access all their personal posts, engage with posts through interactions, and subscribe to bloggers. Additionally, a comprehensive search functionality has been implemented to cater to the needs of all user types.*

## User Classes and Characteristics

*The different types of users expected to use this product are:*

1. *Bloggers: Access to add blog, interact with blogs and view blog analytics.*
2. *Admins: View blogs and search blogs based on keywords.*

## Operating Environment

*Compatible with any operating system and available 24/7. Initially, a local database needs to be utilized, but efforts will be made to enhance the system for global accessibility by hosting it on a public domain with a worldwide database infrastructure..*

## User Documentation

*For any help regarding understanding of logical code segments, refer to:*

* [*https://flask.palletsprojects.com/en/2.3.x/*](https://flask.palletsprojects.com/en/2.3.x/)
* [*https://dev.mysql.com/doc/*](https://dev.mysql.com/doc/)
* [*https://flask-login.readthedocs.io/en/latest/*](https://flask-login.readthedocs.io/en/latest/)

## Assumptions and Dependencies

*When using an online-hosted blogging website created with Flask and MySQL, it is assumed that users have internet access and web browsers, and the website is reliably available. Data security, database maintenance, and regular backups are expected to be in place, with scalability designed for future growth. Additionally, content moderation should be implemented to ensure a safe and engaging user experience.*

# External Interface Requirements

## User Interfaces

*UI includes support for modern web browsers (e.g., Chrome, Firefox, Safari), responsive design for various screen sizes, and compatibility with mobile devices. The application utilizes HTTPS for secure data transmission, offer well-documented RESTful API endpoints for potential third-party integration, and provide clear user authentication mechanisms, including password hashing and session management. Additionally, the URL structure and outline the procedure for configuring custom domain and DNS settings, ensuring a seamless and secure external user experience has also been accounted for.*

## Hardware Interfaces

*The hardware interface for Blogify entail minimal hardware specifications for the server hosting the application, such as a minimum of 2 GB RAM, a dual-core processor, and sufficient storage capacity for data and backups. The server supports Python and MySQL, ensuring compatibility with the application's technology stack. Additionally, it is equipped with a reliable internet connection for continuous online access. The hardware infrastructure also allows scalability to accommodate increased user traffic and data storage needs, with provisions for load balancing and redundancy as the website grows in popularity..*

## Software Interfaces

*Operating System: Windows 10/11*

*Database: MySQL (Can be migrated to MongoDB or SQLite3 with ease)*

*GUI: HTML, CSS and Bootstrap, ReCaptcha.v2*

*Libraries: Python-based libraries like, Flask, SQLAlchemy, Werkzeug, and Pandas*

*Tools : MySQL,,Visual Studio Code, GitHub, Git, Vercel*

## Communications Interfaces

*The communication interface requirements for Flask-based application ‘Blogify’ includes establishing secure communication protocols, such as HTTPS, to encrypt data transmission between the user's browser and the server. The application also implements RESTful API endpoints to facilitate external communication with potential third-party services or applications. User communication within the platform, including comments and messaging, prioritizes data privacy and security through encryption and secure socket layers (SSL). Furthermore, the application also incorporates email notifications for user interactions and system alerts, requiring SMTP service for outbound communication. Lastly, it also accounts for responsive web design and asynchronous data loading to ensure efficient data exchange and a seamless user experience across various devices and network conditions.*

# System Features

*Blogify prides itself on providing an immersive and user-centric environment for diving into the captivating realm of blogging. Irrespective of their user type, all visitors can seamlessly explore a diverse array of blogs, delve into the nuanced realm of likes and dislikes associated with individual posts, and employ a powerful keyword-based search functionality to unearth posts of particular interest. For those seeking an elevated experience, Blogify extends the opportunity to create a logged-in user account through a straightforward registration process using their Email ID. Once logged in, users unlock an array of enhanced features, including access to a personalized dashboard where they can fine-tune their profile details, peruse their own collection of blogs, and delve into comprehensive blog analytics. The power to craft and edit posts, specifically those they own, adds to the allure of the platform. Moreover, users can actively engage with the vibrant blogging community by expressing their appreciation through likes or subscribing to their favorite bloggers, fostering a sense of connection and community within the Blogify ecosystem.*

**Functional Requirement****s**

## Blog Analytics

4.1.1 Description and Priority

*Blog analytics on the blogging site provides bloggers with comprehensive insights into their content's performance and audience engagement. It encompasses essential metrics such as likes and subscriptions, allowing bloggers to identify popular posts and measure their and assess the impact of their content. The analytics feature serves as a valuable tool for bloggers to refine their content strategy, make data-driven decisions, and foster a deeper connection with their readership.*

4.1.2 Stimulus/Response Sequences

*By leveraging these insights, you can refine your content strategy, create more of what your audience enjoys, and strengthen your connection with your readership, ultimately enhancing your overall blogging experience and impact.*

## Anonymity

4.1.1 Description and Priority

*This feature of the interface allows users to anonymously view and surf through Blogify without caring about information leak as any personal information is never visible to other users. Information collected while registration is only used for mailing purposes and never actually used.*

4.1.2 Stimulus/Response Sequences

*This feature allows the users to freely use and post blogs on Blogify.*

## Mailing Feature

4.1.1 Description and Priority

*This feature of the interface allows users subscribe to their favorite blogger so that whenever they post, these people receive notifications through mails.*

4.1.2 Stimulus/Response Sequences

*This feature allows the users to never miss any updates.*

# Other Nonfunctional Requirements

## Performance Requirements

*For optimum performance, the minimum system requirements are as follows:*

* *1.8 GHz or faster processor. Quad-core or better recommended.*
* *2 GB of RAM; 8 GB of RAM recommended (2.5 GB minimum if running on a virtual machine)*
* *Hard disk space: Minimum of 800MB up to 210 GB of available space, depending on features installed; typical installations require 20-50 GB of free space.*
* *Hard disk speed: to improve performance, install Windows and Visual Studio Code on a solid state drive (SSD).*
* *Video card that supports a minimum display resolution of 720p (1280 by 720); Visual Studio will work best at a resolution of WXGA (1366 by 768) or higher.*

## Safety Requirements

*Safety requirements for Blogify encompass critical measures to protect user data and the application's integrity. These requirements include implementing SSL/TLS encryption for secure data transmission, robust user authentication and authorization mechanisms, secure password storage, input validation, and session management. Additional safeguards involve data validation, rate limiting, and CAPTCHA to prevent automated attacks, error handling with secure logging, regular data backups, and software updates. Security headers, CORS policies, and Content Security Policy (CSP) should be set up to mitigate common web vulnerabilities. Regular dependency scanning, penetration testing, and compliance with privacy regulations are essential, as is continuous monitoring and a well-defined incident response plan to address security threats effectively.*

## Security Requirements

*Security requirements for Blogify encompass robust user authentication, secure password storage, data encryption, input validation, protection against XSS and SQL injection, security headers, regular testing and monitoring, and compliance with privacy regulations. An incident response plan is essential for prompt mitigation of security incidents.*

# Other Requirements

*No external requirements other than the ones already mentioned.*

**Appendix A: Glossary**

*The required terminologies for better reading purposes are as follows:*

*SRS: Software Requirements Specification*

*MySQL: My Structured Query Language*

*RAM: Random Access Memory*

*GUI: Graphical User Interface*

*GPA: Grade Point Average*

*CGPA: Cumulative Grade Point Average*

*GHz: Gigahertz*

*GB: Gigabytes*

*ID: Identification*

*XSS: Cross Site Scripting*

*SQL: Structured Querying Language*

*SSL: Secure Socket Layer*

*TLS: Transport Layer Security*

*API: Application Programming Interface*