# Software Requirements Specification (SRS)

## Project Name: Domain Generation Algorithm (DGA)

## Revision History

| **Version** | **Date of Release** | **Pages Affected** | **Reasons for Change** | **Signature** |
| --- | --- | --- | --- | --- |
| 1.0 |  | All | First Release |  |
| 1.1 |  | Appendix N | Test Cases added in the Requirements Traceability matrix |  |
| 2.0 |  | All | Version Release |  |
| 2.1 |  | Page 6 | Change in Management executive |  |

**Table of Contents**

1.0 Project Code 6  
2.0 Title of the project 6  
3.0 Introduction 6  
3.1 Purpose 6  
3.2 Document Conventions 6  
3.3 Intended Audience and Reading Suggestions 6  
3.4 Project/Product Scope 6  
3.5 References 6  
4.0 Overall Description 6  
4.1 Project/Product Perspective 6  
4.2 Project/Product Functions 7  
4.3 User Classes and Characteristics 7  
4.4 Operating Environment 7  
4.4.1 Server 7  
4.4.2 Client: 7  
Browser-based clients either in Microsoft Windows or Linux Environment. 7  
Android-based mobile client. 7  
4.5 Design and Implementation Constraints 7  
4.6 User Documentation 7  
4.7 Assumptions and Dependencies 8  
5.0 External Interface Requirements 8  
5.1 User Interfaces 8  
6.0 System Features 9  
7.0 Other Nonfunctional Requirements 20  
Performance Requirements 20  
Safety Requirements 20  
Security Requirements 21  
Software Quality Attributes 21  
7.1 Other Requirements 21  
7.2 Applicable Standards 21  
8.0 Acceptance Criteria 22  
9.0 Deliverables 22

**1.0 Project Code**

* Project Code: DGA

**2.0 Title of the project**

* Project Title: Domain Generation Algorithm

## 3.0 Introduction

### 3.1 Purpose

The purpose of this document is to outline the software requirements for the Domain Generation Algorithm (DGA) project. This document provides a detailed description of the project's objectives, functionalities, and constraints, serving as a reference for developers, stakeholders, and users.

### 3.2 Document Conventions

None

### 3.3 Intended Audience and Reading Suggestions

* Developers
* Project Managers
* Stakeholders
* End Users

### 3.4 Project/Product Scope

The DGA project aims to develop an algorithm capable of generating domain names based on specified criteria and validating these domains against predefined standards. This document covers the functionalities, limitations, performance expectations, security measures, and compatibility requirements of the DGA.

### 3.5 References

#### 3.5.1 Ref Docs

* ICANN Domain Name Registration Guidelines
* GDPR Compliance Guidelines

## 4.0 Overall Description

### 4.1 Project/Product Perspective

The DGA project is a new initiative aimed at providing an efficient and automated solution for generating and validating domain names. The project will focus on creating a robust algorithm that can handle various inputs and generate valid domain names quickly and accurately.

### 4.2 Project/Product Functions

1. Generate Domains: Users can generate domain names using different algorithms.
2. Validate Domains: Users can validate generated domains against specific criteria.
3. Handle Edge Cases: The system handles invalid or unusual input gracefully.

### 4.3 User Classes and Characteristics

* **Administrators**: Manage system settings and user access.
* **Developers**: Integrate the DGA into applications and services.
* **End Users**: Use the DGA to generate and validate domain names.

### 4.4 Operating Environment

#### 4.4.1 Server

* Server OS: Linux
* Language: Python, JavaScript
* Database: PostgreSQL
* Application Server: Apache or Nginx

#### 4.4.2 Client

* Browser-based clients either in Microsoft Windows or Linux Environment.
* Android-based mobile client.

### 4.5 Design and Implementation Constraints

* The system must operate efficiently within the specified server and client environments.
* Limit the length of generated domains to comply with DNS standards.

### 4.6 User Documentation

User manual and online help documents will be delivered along with the software. Installation instructions will be available in the Installation manual.

### 4.7 Assumptions and Dependencies

* Assumes stable network connectivity.
* Dependent on third-party libraries and tools for certain functionalities.

## 5.0 External Interface Requirements

### 5.1 User Interfaces

* The targeted browsers are Mozilla Firefox 12.0, Google Chrome, and MS Edge.

### 5.2 Hardware Interfaces

The proposed system configuration is as follows:

| **Sl. No.** | **ITEM** | **Server 1** | **Server 2** |
| --- | --- | --- | --- |
| 01 | Processor | Quad core Processor (64 bit) | Quad core Processor (64 bit) |
| 03 | No. of processor | One | One |
| 04 | Memory | 8 GB expandable to 24 GB | 8 GB expandable to 24 GB |
| 10 | HDD capacity | 1TB | 1TB |
| 12 | Network | 2 Gigabit RJ45 connectors | 2 Gigabit RJ45 connectors |
| 13 | USB ports | Four | Four |
| 17 | OS | Linux | Linux |

### 5.3 Communications Interfaces

* Automated e-Mail sending facility

## 6.0 System Features

* **Generate Domains**: The algorithm will generate domain names based on specified algorithms (e.g., random, algorithmic).
* **Validate Domains**: The system will validate generated domains against certain criteria (e.g., length, character set).
* **Handle Edge Cases**: The system will handle invalid input and unusual inputs gracefully.

## 7.0 Other Nonfunctional Requirements

### 7.1 Performance Requirements

* Response time for domain generation should not exceed 2 seconds.
* The system should support at least 1000 concurrent users without performance degradation.
* Throughput should allow for the generation of at least 10,000 domains per minute.

### 7.2 Safety Requirements

* The system should ensure data integrity and reliability.

### 7.3 Security Requirements

* Implement input validation to prevent injection attacks.
* Ensure secure transmission of data using HTTPS.
* Implement rate limiting to prevent abuse of the DGA.

### 7.4 Software Quality Attributes

* Maintainability: The system should be easy to maintain and update.
* Usability: The user interface should be intuitive and user-friendly.

### 7.5 Other Requirements

* Compliance with industry standards and regulations.

### 7.6 Applicable Standards

* Follow relevant industry standards and best practices.

## 8.0 Acceptance Criteria

* The system meets all functional and non-functional requirements.
* Successful completion of testing phases.
* Positive feedback from end-users during the User Acceptance Testing (UAT) phase.

## 9.0 Deliverables

* Software application
* User manual
* Installation manual
* Technical documentation
* Training materials