



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment - 1

Student Name: Keshav Datta

Branch: BE-CSE

Semester: 6th

Subject Name: System Design

Subject Code: 23CSH-314

UID: 23BCS10423

Section/Group: KRG_2B

Date of Performance: 7/1/26

Aim:

To design and analyze a URL Shortener system by identifying its functional and non-functional requirements and representing the system design using a draw.io diagram.

Objectives:

1. To understand the working of a URL Shortener system in real-world applications.
2. To identify and analyze the functional requirements of the URL Shortener system.
3. To study non-functional requirements such as performance, scalability, and reliability.
4. To design a high-level system architecture and data flow using draw.io.
5. To strengthen understanding of practical and real-world system design concepts.

Procedure:

1. Studied the concept and working of URL Shortener systems commonly used in real-world applications.
2. Analysed the core functionalities required for efficient URL shortening and redirection.
3. Identified key functional requirements such as short URL generation, custom alias support, URL expiration handling, and redirection mechanism.
4. Examined important non-functional requirements including low latency, scalability, and system reliability.
5. Designed a well-structured high-level system architecture diagram using draw.io to represent system components and workflows.
6. Reviewed and validated the system diagram to ensure accuracy, clarity, and completeness of the design.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Functional Requirements:

1. Generate a short URL from a given long URL.
2. Allow users to create custom short URLs (aliases).
3. Support both default and user-defined expiration dates for URLs.
4. Redirect users from the short URL to the original long URL.

Non-functional Requirements:

1. Ensure low latency with a response time of less than 200 milliseconds.
2. Scale efficiently to handle up to 100 million daily active users.
3. Support the shortening and storage of up to 1 billion URLs.
4. Maintain high availability and reliability of the system.

Outcome / Result –

1. A high-level design of a URL Shortener system was successfully created using draw.io.
2. Functional and non-functional requirements were analysed and defined in a clear and structured manner.
3. The experiment helped in understanding practical system design aspects such as scalability, performance, and efficiency.
4. Hands-on experience was gained in requirement analysis and conceptual system architecture design.

REQUIRED SYSTEM DESIGN:

