Certainly! Below is a simplified Java code snippet to get you started on the link shortener project. Note that this is a console-based application, and you might want to enhance it further based on your preferences and requirements.

```java
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;

public class LinkShortener {

    private static Map<String, String> urlMappings = new HashMap<>();

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println("1. Shorten a URL");
            System.out.println("2. Exit");
            System.out.print("Choose an option: ");
            int choice = scanner.nextInt();
            scanner.nextLine();  // Consume newline

            switch (choice) {
                case 1:
                    shortenUrl(scanner);
                    break;
                case 2:
                    System.out.println("Exiting...");
                    System.exit(0);
                default:
                    System.out.println("Invalid option. Please try again.");
            }
        }
    }

    private static void shortenUrl(Scanner scanner) {
        System.out.print("Enter the long URL: ");
        String longUrl = scanner.nextLine();

        // Generate a short link (you can replace this with your shortening algorithm)
        String shortLink = generateShortLink(longUrl);

        // Store the mapping
        urlMappings.put(shortLink, longUrl);

        System.out.println("Shortened URL: " + shortLink);
    }

    private static String generateShortLink(String longUrl) {
        // Replace this with your shortening algorithm (e.g., base conversion or hashing)
        // For simplicity, using a basic hash of the long URL in this example
        return String.valueOf(longUrl.hashCode());
    }
}
```

This is a basic console application. You can expand it by adding error handling, improving the shortening algorithm, implementing persistent storage, and creating a more user-friendly interface if you plan to deploy it outside of the console environment.