Let’s Poll People

CLC Assessment

Contents

[Instructions 2](#_Toc173864884)

[Instructions: Building a Polling Website 2](#_Toc173864885)

[Choose Your Front-End Framework 2](#_Toc173864886)

[Create a User Authentication System 2](#_Toc173864887)

[Design the User Interface 2](#_Toc173864888)

[Create Essential Pages 2](#_Toc173864889)

[Poll Voting Page 2](#_Toc173864890)

[Poll Results Page 2](#_Toc173864891)

[Implement Routing 3](#_Toc173864892)

[Utilize State Management 3](#_Toc173864893)

[Data Storage 3](#_Toc173864894)

[Unit Testing (Optional but Advantageous) 3](#_Toc173864895)

[Evaluation Criteria 3](#_Toc173864896)

# Instructions

Welcome to CLC for your technical test.

In this assessment you will demonstrate your skills as a developer by **building a real-world polling website**. This exercise is designed to assess your ability to create interactive web applications using either **Blazor or Angular** as the front-end framework. The backend of the website will be built using C# and .NET 7 or higher, while the frontend will be developed using either Blazor or Angular.

This real-world assessment will test your proficiency in **user authentication, routing, state management, and responsive design.** You will also have the opportunity to demonstrate your knowledge of best practices in web development.

Furthermore, we encourage you to **implement unit testing for both the front-end and back-end components** to showcase your skills in ensuring the quality of your code.

## Instructions: Building a Polling Website

### Choose Your Front-End Framework

You have the option to choose either Blazor or Angular for the front end. You should be familiar with the chosen framework and follow its best practices for implementation.

I’ve chosen to use **Blazor Web Assembly** because it can be downloaded as an app. I’m also going to use **Mud Blazor** as a modern material design.

Complete

### Create a User Authentication System

Implement a **secure user authentication system** that allows users to **register** and **log in**. User authentication is a critical component of the website, and it must be **robust** and **secure**.

I’ve created a system admin user

Username: Mlando

Password: Password1

Complete

### Design the User Interface

Design a clean and intuitive user interface for the polling website. Ensure that the website is responsive, functioning well on both desktop and mobile devices.

In Progress

### Create Essential Pages

Poll Creation Page: Implement a page where users can **create new polls**. Users should be able to **define poll questions and options**.

### Poll Voting Page

Develop a page that **allows users to vote on existing polls**. Users should be able to **select an option and submit their vote**.

### Poll Results Page

Create a page that **displays the results of a poll**. This page should **show the question, options,** and the **number of votes for each option.**

### Implement Routing

Date: 06/08/2024

Time: N.A.

Venue: Remote

Set up proper routing for your website to enable navigation between different pages. Ensure that users can **access the login, poll creation, poll voting, and poll results pages** seamlessly.

### Utilize State Management

Implement state management solutions appropriate to your chosen front-end framework (Blazor components or

Angular services) to manage the state of the application effectively.

### Data Storage

You have the flexibility to choose the data storage method. You can use a database, in-memory storage, or any other data store of your choice to save poll data and user information. Make sure that your choice is appropriate for the task.

### Unit Testing (Optional but Advantageous)

We encourage you to write unit tests to verify the functionality of both the front-end and back-end components.

While this is not mandatory, it will be advantageous and highly appreciated.

Submission

Submit your source code for the polling website in a zipped file along with clear instructions for running the application. If you used a database, provide documentation on how to set it up. Additionally, include any test cases and documentation for unit testing. You can send this to carinp@clc.co.za.

### Evaluation Criteria

Your submission will be evaluated based on several criteria, including code quality, functionality, user authentication, responsiveness, unit testing, best practices, and documentation.

Please note that this is a hands-on exercise designed to assess your practical skills as a front-end developer. We encourage you to showcase your best work and adhere to industry best practices in web development. Good luck, and we look forward to reviewing your submission!