# **Toll Calculator Application**

As a Frontend Engineer, you will be responsible for developing an advanced Toll Calculator application using React, Next.js, Hooks, Redux, and memoization. The application should not only consume the TollGuru API to calculate toll costs between two waypoints but also provide users with an immersive experience in understanding how toll calculations work. In addition to the original and updated requirements, the candidate is expected to demonstrate the following:

## **Prerequisites:**

**API Understanding:**

1. Candidates must thoroughly go through the [TollGuru API documentation](https://tollguru.com/toll-api-docs) to understand how the API works.
2. Create an account on the TollGuru website and obtain a free trial API key following the guidelines [here](https://tollguru.com/toll-api).
3. Candidates should visit [TollGuru Toll Calculator](https://tollguru.com/toll-calculator) to gain insights into how toll calculations are performed.

## **Additional Criteria:**

#### **Note:** *As a frontend engineer, it's essential to assign meaningful IDs to components for improved accessibility and collaboration. Use clear, descriptive IDs that reflect the component's purpose, maintain consistency, avoid generic terms, consider accessibility, and ensure uniqueness. Adopt camel case for multi-word IDs. For example:*

| function Header() {  return <**header** id="tollCalculatorHeader">...</**header**>; }  function CalculateTollButton() {  return <**button** id="calculateToll">Calculate Toll</**button**>; }  function renderPolylineFromOSRM() {  return <**div** id="leafletMapContainer">...</**div**>; } |
| --- |

#### **User Education (15 points):**

1. Implement a feature that educates users on how toll calculations are performed.
2. Create an interactive guide or tooltip system within the application to explain the factors affecting toll costs.

#### **Route Visualization (25 points):**

1. Implement route visualization using React-Leaflet.
2. Display the calculated route by sending a polyline to the TollGuru API (using only [Route Encoded Polyline](https://tollguru.com/toll-api-docs#tag/Tolls-for-complete-route-polyline-from-a-mapping-service)) [***Note\****: *Using any endpoint other than the one mentioned will result in the cancellation of the submission*].
3. Integrate markers along the route with details of toll information.

**Polyline Decoding (15 points):**

1. Utilize the [@googlemaps/polyline-codec](https://www.npmjs.com/package/@googlemaps/polyline-codec) library to decode polylines received from the TollGuru API.
2. Ensure accurate decoding for route visualization.

**Toll Details Display (15 points):**

1. Integrate toll details on the markers along the route.
2. Display relevant toll information, such as cost and additional details.

**Documentation and Testing (7 + 8 points):**

1. Comprehensive documentation covering setup and configurations.
2. Troubleshooting tips and best practices are included.
3. **Writing Test Cases (Using** [**cypress**](https://docs.cypress.io/guides/overview/why-cypress)**):**
   1. The candidate is required to write comprehensive unit test cases to ensure the application's functionality is as expected. This is critical for several reasons:
      1. **Verification of Functionality:** Tests should verify that each component, including toll calculations, API integration, route visualization, and toll detail display, works correctly.
      2. **Code Quality and Maintainability:** Tests should be written in a way that encourages clean and maintainable code structures.
      3. **Ease of Future Enhancements:** The test suite should facilitate future enhancements or modifications, ensuring that new changes don’t break existing functionalities.

**GitHub Code Submission and Access (8 points):**

1. Commit your code to a private GitHub repository.
2. Share the repository access with [[Anish Ahlawat](mailto:anishahlawat@mapup.ai)[Asijit Paul](mailto:asijitp@mapup.ai)[Vishal Balaji D](mailto:vishalbd@mapup.ai) [Ajay Arvind Padwal](mailto:ajayap@mapup.ai)]. [***Note\****: *Sharing the access is mandatory*].
3. Practice good version control habits by making frequent and descriptive commits. Each commit should have a clear and informative message that summarizes the changes made in that commit.

**Hosting Requirement (7 points):**

1. The candidate is expected to host the Toll Calculator application on [Vercel](https://vercel.com/mapup-team).
2. Provide the link for deployed app and github repo.

**End Goal:**

1. Demonstrate a fully functional Toll Calculator application that not only efficiently calculates toll costs but also educates users on the factors influencing toll calculations.
2. Provide well-documented code, clear instructions, and a link to the hosted application.