How to create a React frontend for your Spring Boot Java backend using Visual Studio on Windows:

1. Project Setup:

- **Install Node.js and npm:**
  - Download and install Node.js from https://nodejs.org/. This will also include npm, the Node Package Manager.
- **Create a new React project:**
  - Open a command prompt or terminal window.
  - Navigate to the desired directory where you want to create your project.
  - Run the following command:

    Bash
    ```
    npx create-react-app my-react-app
    ```
    Replace my-react-app with your desired project name.

2. Open the Project in Visual Studio:

- **File > Open > Project/Solution:**
  - Navigate to the directory where you created your React project.
  - Select the package.json file and open it.

3. Install Dependencies:

- **Open the Package Manager Console:**
  - Go to **View > Other Windows > Package Manager Console**.
- **Install necessary libraries:**
  - In the Package Manager Console, run the following command:

    PowerShell
    ```
    Install-Package axios
    ```
    axios is a popular HTTP client library for making requests to your backend.

4. Set up Backend Communication:

- **Create a service file:** Create a new file named BackendService.js in a suitable directory (e.g., src/services).
- **Define API endpoints:** Inside the service file, define functions that will make HTTP requests to your backend API endpoints. For example:

  JavaScript
  ```
  import axios from 'axios';

  const baseURL = 'http://localhost:8080'; // Replace with your backend URL

  const getProducts = () => {
    return axios.get(`${baseURL}/products`);
  };
  ```

```
export default {
  getProducts,
};
```

Replace http://localhost:8080 with the actual URL of your backend.

5. Integrate with Components:

- **Import the service:** In your React components, import the service file:
  JavaScript
  ```
  import BackendService from './services/BackendService';
  ```

- **Make API calls:** Use the service functions to make API calls and handle the responses:
  JavaScript
  ```
  import React, { useState, useEffect } from 'react';
  import BackendService from './services/BackendService';

  function ProductsList() {
  const [products, setProducts] = useState([]);

  useEffect(() => {
    const fetchProducts = async () => {
      const response = await BackendService.getProducts();
      setProducts(response.data);
    };
    fetchProducts();
  }, []);

  return (
    <div>
      <h1>Products</h1>
      <ul>
        {products.map((product) => (
          <li key={product.id}>{product.name}</li>
        ))}
      </ul>
    </div>
  );
  ```

}

export default ProductsList;

```
This example fetches a list of products from the backend and displays
them in a list.

**6. Run the Frontend:**

- **Open the Debug menu:**
- Go to **Debug > Start Debugging**.
- The browser will open and display your React app at
`http://localhost:3000`.

**Additional Tips:**

- Consider using a state management library like Redux or Context API
for more complex applications.
- Implement error handling and loading indicators to provide a better
user experience.
- Use a linter to ensure code quality and consistency.
- Explore tools like Storybook for component development and testing.

By following these steps and leveraging the power of Visual Studio and
React, you can effectively create a robust and user-friendly frontend
for your Spring Boot Java backend on Windows.
```

Sources

1. https://mithin.hashnode.dev/creating-a-nlp-app?source=tags_feed_article
2. https://www.bairesdev.com/blog/server-side-rendering-react/