Quality Requirements for BIP:

The portal should be intuitive and usable without any documentation for users familiar with common web platforms (e.g., event booking sites) [Usability].

All interactive elements should perform actions as described by their labels [Reliability].

Each tour guide profile should have a profile photo to be recognized by visitors [Usability].

The system should process updates within 500 milliseconds [Performance].

All displayed information must be accurate and up-to-date. [Reliability].

Registered users and guest users should be informed when the status of their application changes. [Usability].

If a user input error occurs, the portal should clearly identify the issue and highlight the relevant area [Error Handling].

The website layout should be responsive and properly sized on all device screens, including mobile, without overflow [Hardware].

Users should only access their data and public information of others, ensuring privacy [Security].

The portal should efficiently handle up to 100 concurrent users [Performance].

The portal should redirect guest users to verification, if their credentials are out-of-date. [Security]

TechStack:

We plan on using **MongoDB** for storing and accessing our database, as it provides flexibility and scalability when handling unstructured or semi-structured data. MongoDB's document-oriented storage aligns well with modern web application requirements and allows for rapid development and iteration.

For the backend programming language, we chose **JavaScript**, using the **Node.js** runtime environment with the **Express.js** framework to manage server-side operations. Node.js and Express.js offer a lightweight and efficient platform for building scalable network applications.

Using JavaScript on both the front end and back end simplifies development and allows for code reusability, which enhances team productivity.

On the front end, we continue to use **React**, which offers a modular approach to building user interfaces with its component-based structure. The ability to reuse components in React will help streamline development and enhance the application's performance. Additionally, React's virtual DOM improves rendering efficiency, contributing to a smoother user experience.