**Semester 2 Project - Technical Documentation**

This project, built with PHP, HTML, CSS, and JavaScript, serves as a platform for EPITA International Programs staff to efficiently manage student, course, population, and grade data from the system’s database.

**1. User Interface (HTML, CSS, JavaScript)**

The frontend is designed for ease of use, focusing on seamless navigation and a responsive layout.

* **HTML**: Key files like index.html structure the web pages, displaying student lists, courses, and populations in table formats. The pages also include forms for adding or modifying records.
* **CSS**: Styling is applied through styles.css, ensuring the interface has a uniform, clean, and professional appearance.
* **JavaScript**: The script.js file handles dynamic features, such as appending rows to tables and supporting form submissions for new records (e.g., adding students or courses). It plays a critical role in maintaining a smooth, interactive user experience.

**2. Server-Side Logic (PHP)**

PHP powers the backend, enabling secure interactions with the MySQL database.

* **Database Management**: The db.php file establishes database connections using PHP Data Objects (PDO), ensuring smooth and secure data exchanges between the frontend and backend.
* **Authentication System**: User authentication is managed by auth\_check.php, supported by client-side logic in auth\_checker.js. These scripts validate login credentials, ensuring proper access control.
* **Data Retrieval Scripts**: Several PHP files (e.g., get\_students\_by\_discipline.php, get\_grades.php) are responsible for retrieving specific information such as student data, course details, or grades from the database, tailored for staff use.

**3. Real-Time Data Handling (AJAX)**

AJAX integration allows data to be fetched or updated without requiring page reloads. This ensures that operations like adding, deleting, or updating records (e.g., students, courses) are immediately reflected on the interface, enhancing user responsiveness.

**4. Modular Design**

The application follows a modular structure, allowing for easier maintenance and future scalability. Each part of the system is clearly defined and designed to facilitate straightforward future updates or feature additions.