1. Project Name

- **DasBuch**

---

2. General Description

A web platform for recording and managing book reviews. The goal is to build a professional system with a **RESTful API** to connect the PostgreSQL database with the front-end. Refined search capabilities and data manipulation via CRUD operations will be implemented.

---

3. Key Features

1. **Full CRUD** for book reviews (Create, Read, Update, Delete).
2. **Search Filters**:
    - Search by **Name**.
    - Search by **Genre**.
    - Search by **Author**.
    - Search by **Reading Period**.
    - Search by **Rating** (using operators: >=, <=, =).
    - **Sorting Options**: Book Name, Author, Rating, and Reading Date.
3. **Data Visualization**:
    - Paginated list of reviews.
    - Detailed view of each review.
4. **Internationalization (i18n)**:
    - Multi-language support implemented from the start.
5. **Web Interface**:
    - EJS for Node.js.
    - Razor Pages for .NET.
    - Front-end built using BootstrapStudio.

---

4. Data Structure (Review Object)

Each review will have the following attributes:

- **Book Name** (String)
- **Author Name** (String)
- **Genre** (String)

- **Comment** (String)
- **Rating** (Integer: 1-5)
- **Reading Date** (Date)

---

5. System Screens (Views)

1. **Home Page (Reviews List)**
    - Displays the complete list of reviews.
    - Allows filtering by:
        - Genre
        - Author
        - Reading Period
        - Rating (>=, <=, =)
        - Sorting Options
    - Actions:
        - Add Review (link to creation page).
        - View Details (link to detail page).
        - Edit and Delete directly from the list.
2. **Review Creation Page**
    - Form to add a new review:
        - Book Name.
        - Author Name.
        - Genre.
        - Comment.
        - Rating (1-5).
        - Reading Date.
3. **Review Details Page**
    - Complete display of all review information.
    - Buttons to edit or delete the review.
4. **Review Edit Page**
    - Form similar to the creation page but pre-filled with current data.
5. **Advanced Search Page**

   - Dedicated page for detailed search options.
   - Results displayed as a list.

6. **Error/404 Page**
   - Generic error page (e.g., "Review not found", invalid routes, etc.).

---

6. Technologies Used

1. **Management and Organization**:
   - Trello
2. **Back-end**:
   - NodeJS/Express
   - .NET (C#)
   - PostgreSQL Database
3. **Front-end**:
   - EJS for NodeJS
   - Razor Pages for .NET
   - BootstrapStudio
4. **Internationalization**:
   - i18n standard for multi-language support.
5. **Version Control**:
   - Git/GitHub
6. **Agile Methodology**:
   - Sprint planning with iterative deliveries.