Task: Website Database of Unreliable Tenants

Problem Description

Property owners often face the following issues with tenants:

- Failure to pay rent on time or complete evasion of payment.
- Damage to property.
- Violation of rental agreements.
- Conflicts with neighbors.

The lack of transparency and an accessible database makes it difficult to verify future tenants, leading to financial losses and legal disputes.

Project Goal

To create a website where property owners can check tenant reliability and leave reviews and complaints.

Functional Requirements

1. User Registration and Authentication

- ☑ Registration for property owners (landlords).
- ☑ Tenants can register to view their reputation.
- ☑ Authentication via email/phone/social networks.

2. Tenant Information Search

- ☑ Enter name, ID number, phone number, or other details to verify a tenant.
- ☑ Display complaints, debts, and violations.
- ☑ Filter by region and date added to the database.

3. Adding Data on Unreliable Tenants

- ☑ Complaint submission form with specified reasons:
  - Late rent payments.
  - Property damage.
  - Violation of lease terms.
  - Neighbor complaints.
- ☑ Option to upload evidence (photos, contract copies, correspondence screenshots).
- ☑ Information verification before publication (moderation).

4. Rating and Review System

- ☑ Ability to leave reviews and rate tenants.
- ☑ Reviews remain anonymous but can be disputed via moderation.

5. Legal Safety and Privacy

- ☑ Rules to prevent defamation and illegal data sharing.
- ☑ Mechanism for disputing false information.

6. Admin Panel for Moderation

- ☑ Reviewing and approving new entries.
- ☑ Removing false or unverified complaints.
- ☑ Managing users (blocking, issuing warnings).

Technical Requirements

- **Backend:** Python (Django/FastAPI), Node.js (Express/NestJS), PHP (Laravel).
- **Frontend:** React/Vue.js/Angular.
- **Database:** PostgreSQL, MySQL, or MongoDB.
- **Hosting:** AWS, DigitalOcean, VPS.
- **API:** Possible integration with debt-checking services.

Evaluation Criteria (100 points)

1. Functional Capabilities (30 points)

- Are the core functions (search, tenant addition, rating) implemented?
- Is user registration and authentication available?

2. Technical Implementation (20 points)

- Code quality and security.
- Performance and database optimization.

3. User Interface and Usability (15 points)

- How clear and user-friendly is the design?
- Ease of searching for information.

4. Data Protection and Legal Compliance (15 points)

- Compliance with data privacy laws (GDPR, local regulations).

- Implementation of mechanisms to prevent defamation and illegal information.

5. Presentation (10 points)

- Clear explanation of system logic.
- Demonstration with real or test data.

6. Innovation (10 points)

- Are there unique features (e.g., integration with payment services to check debts)?
- Does the project improve the tenant verification experience?

Additional Features

◇ **Tenant verification system via open databases** (e.g., court debt records). ◇ **Integration with real estate agencies** for data exchange. ◇ **Landlord chatroom** (to discuss issues and share experiences). ◇ **Automatic notifications to tenants** if they are added to the database.

Legal Risks and Protection

☑ Data should be anonymized to prevent legal claims.
☑ The platform must provide a data removal function (in compliance with legal requirements).
☑ All complaints should go through moderation.