

Report on

# Trade Finance Guarantee Issuance System

-By

Apurva  
Munish  
Maharshi  
Chidera

Contents	
Technologies Used .....	3
Database Schema Overview.....	3
Features .....	4
Deployment of Laravel .....	5

Screenshots .....	9
Conclusion .....	16

### **Trade Finance Guarantee Issuance System**

This project involves developing a Trade Finance Guarantee Issuance System using Laravel. It focuses on creating and managing guarantees such as Bank, Bid Bond, Insurance, and Surety through both manual entry and bulk data transfer. The application integrates core CRUD functionalities, secure user access, and data processing, ensuring scalability and reliability.

## Technologies Used

- **PHP:** Core programming language for backend development.
- **Laravel Framework:** PHP-based framework for building the system with MVC architecture.
- **MySQL:** Database management system for storing guarantees and related data.
- **HTML/CSS (Tailwind):** For building responsive and visually appealing frontend interfaces.
- **JavaScript:** To enhance interactivity in the user interface.
- **MVC Pattern:** Used to organize the project into Model, View, and Controller layers, ensuring a clean architecture.
- **Repository Pattern:** For managing data access logic, providing modular and testable components.
- **Blade Templates:** For creating dynamic and reusable frontend views.

## Database Schema Overview

The database consists of three main tables: **User**, **Guarantee**, and **File**, each designed to support secure operations and maintain data integrity.

### 1. **User Table:**

This table manages user data for authentication and role-based access. It includes fields like id, name, email (unique), role (e.g., admin, applicant), password, and timestamps for tracking creation and updates.

### 2. **Guarantee Table:**

This table stores guarantee-related information. Key fields include corporate\_reference\_number (unique), guarantee\_type (e.g., Bank, Bid Bond), nominal\_amount, currency, expiry\_date,

---

applicant\_id (linked to User.id), status, reviewed\_by, and review\_comments.

3. **File Table:**

This table handles bulk-uploaded files stored as binary blobs. Fields include file\_name, file\_type, file\_blob, and uploaded\_by (linked to User.id).

## Relationships

- **User-Guarantee:** Users can create or review multiple guarantees.
- **User-File:** Users can upload multiple files for bulk data processing.
- **Guarantee-File:** Files are indirectly linked to guarantees through users.

This schema ensures scalability, secure data handling, and seamless role-based interactions.

## Features

1. **Core Operations:**

- Create, review, apply for, issue, and delete guarantees. ◦
- Validation rules ensure data integrity

2. **Authentication and Authorization:**

- Laravel's **Auth middleware** is utilized to secure access to guarantee management functionalities.

3. **Bulk Data Processing:**

- Supports uploading and processing of CSV, JSON, and XML files.
-

- Uploaded files are stored as blobs in the database. ○  
File parsing includes validation for required fields, such as Corporate Reference Number and Expiry Date.

#### 4. **File Management:**

- A user interface lists uploaded files with metadata (e.g., name, upload date, type).
- Files can be deleted with user confirmation to prevent accidental removals.

#### 5. **Dynamic Frontend:**

- Blade templates styled with **Tailwind CSS** to create interactive and user-friendly views

### Deployment of Laravel

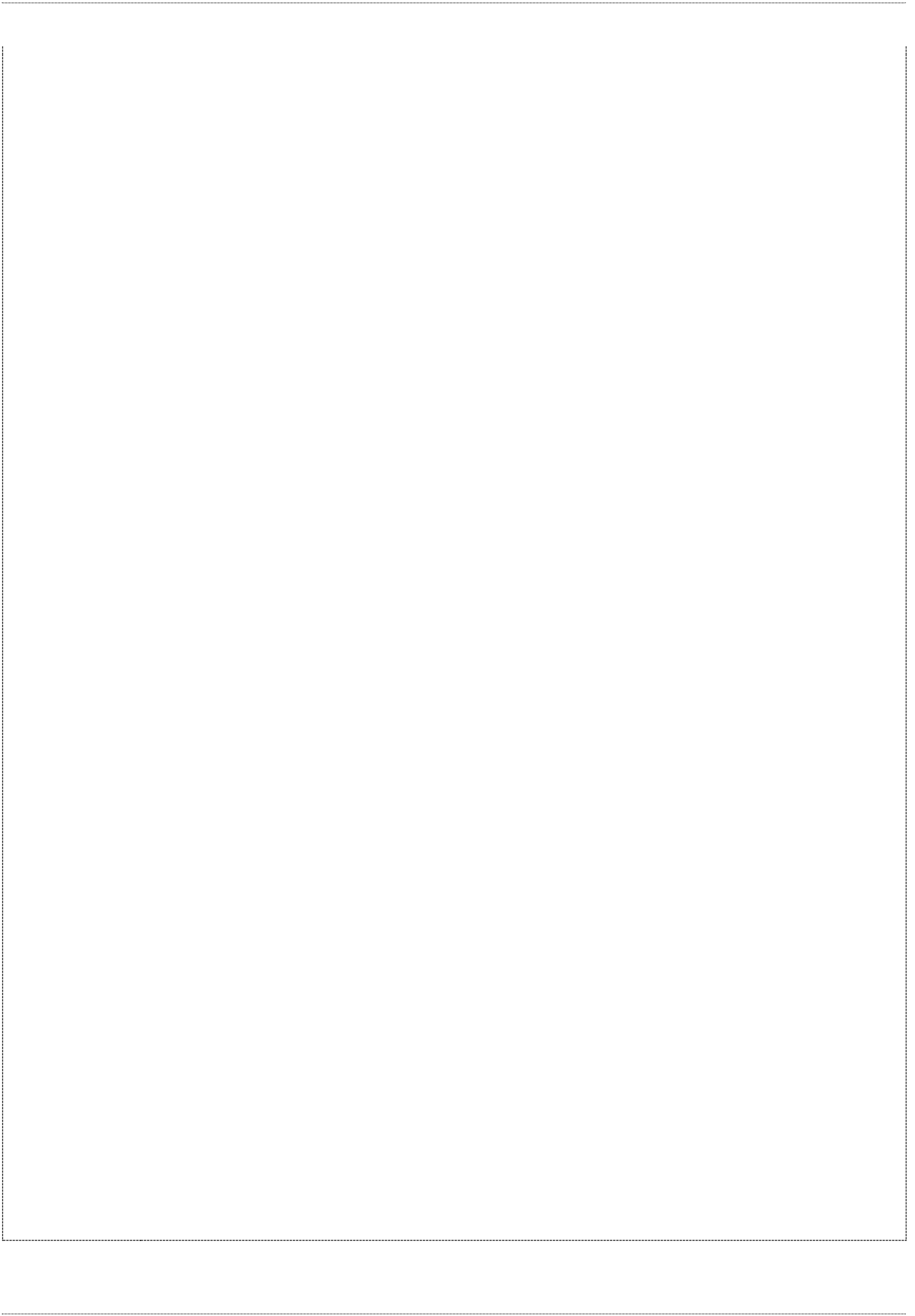
- **Install Dependencies:** ○ Run composer install to set up all required dependencies.
- **Configure the Environment:**
  - Copy .env.example to .env and update it with database credentials and other application settings.
  - Generate the application key with php artisan key:generate.
- **Database Setup:**
  - Create a new database schema in MySQL (or another supported database).

- Run `php artisan migrate` to set up tables and optionally `php artisan db:seed` to populate with sample data.

---

- **Run the Application:**

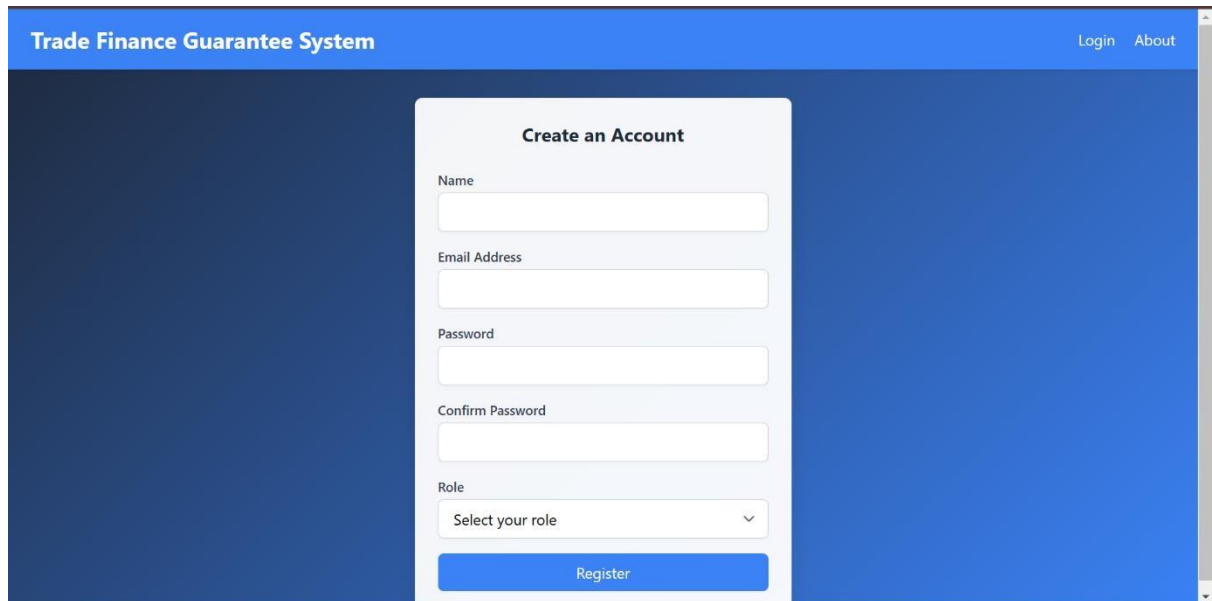
- Start the development server using php artisan serve and access it at <http://127.0.0.1:8000>.





## Screenshots

### Register:



The screenshot shows the 'Create an Account' registration form. The form is centered on a dark blue background. It includes input fields for Name, Email Address, Password, and Confirm Password. There is also a dropdown menu for Role with the text 'Select your role'. A blue 'Register' button is at the bottom of the form. The header of the page is blue with the text 'Trade Finance Guarantee System' and links for 'Login' and 'About'.

Trade Finance Guarantee System [Login](#) [About](#)

### Create an Account

Name

Email Address

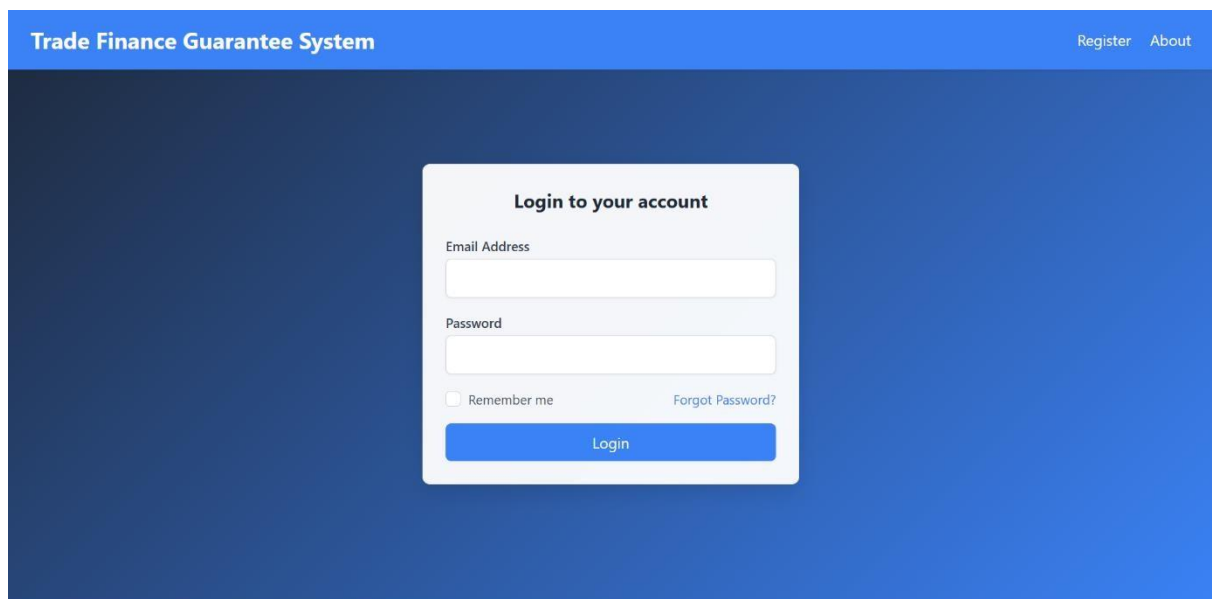
Password

Confirm Password

Role

[Register](#)

### Login:



The screenshot shows the 'Login to your account' form. The form is centered on a dark blue background. It includes input fields for Email Address and Password. There is a checkbox for 'Remember me' and a link for 'Forgot Password?'. A blue 'Login' button is at the bottom of the form. The header of the page is blue with the text 'Trade Finance Guarantee System' and links for 'Register' and 'About'.

Trade Finance Guarantee System [Register](#) [About](#)

### Login to your account

Email Address

Password

☐ Remember me [Forgot Password?](#)

[Login](#)

## Admin Dashboard:

Admin Panel

Dashboard

Manage Users

Manage Guarantees

Uploaded Files

Logout

Admin Dashboard

Total Users

8

Total Guarantees

5

Uploaded Files

0

© 2024 Trade Finance Guarantee System

## Manage Users:

Admin Panel

Dashboard

Manage Users

Manage Guarantees

Uploaded Files

Logout

Manage Users

Search users...

Search

#	NAME	EMAIL	ROLE	ACTIONS
1	Admin User	admin@example.com	admin	<a href="#">Edit</a> <a href="#">Delete</a>
2	Applicant User	applicant@example.com	applicant	<a href="#">Edit</a> <a href="#">Delete</a>
6	Reviewer Name	reviewer@example.com	reviewer	<a href="#">Edit</a> <a href="#">Delete</a>
7	mitali	mitali@example.com	applicant	<a href="#">Edit</a> <a href="#">Delete</a>
8	mitali13	mitali13@example.com	applicant	<a href="#">Edit</a> <a href="#">Delete</a>
9	sam	sam@example.com	applicant	<a href="#">Edit</a> <a href="#">Delete</a>
10	sam13	sam13@example.com	applicant	<a href="#">Edit</a> <a href="#">Delete</a>
11	mit	mit@g	applicant	<a href="#">Edit</a> <a href="#">Delete</a>

## Manage Guarantees:

Admin Panel

Dashboard

Manage Users

Manage Guarantees

Uploaded Files

Logout

Manage Guarantees

+ Add New Guarantee

Corporate Ref. No.	Type	Nominal Amount	Expiry Date	Actions
13	Insurance	12.01 us	2024-12-07	<div>Edit Delete</div>
CRN12345	Bank	10000.00 USD	2025-11-24	<div>Edit Delete</div>
CRN123435	Bid Bond	10230.00 USD	2025-11-24	<div>Edit Delete</div>
23432	Bid Bond	342.00 GBP	2024-12-07	<div>Edit Delete</div>
23456	Insurance	2345.00 EUR	2024-11-29	<div>Edit Delete</div>

© 2024 Trade Finance Guarantee System

## Upload data using File:

Admin Panel

Dashboard

Manage Users

Manage Guarantees

Uploaded Files

Logout

Uploaded Files

File Name	File Type	Uploaded At	Actions
example.csv	csv	2024-11-25 00:55:48	<div>Delete</div>

© 2024 Trade Finance Guarantee System

## Edit Guarantee:

Admin Panel

Dashboard

Manage Users

Manage Guarantees

Uploaded Files

Logout

Edit Guarantee

Corporate Reference Number

13

Guarantee Type

Insurance

Nominal Amount

12.01

Nominal Amount Currency

us

Expiry Date

07-12-2024

Applicant Name

Mitali

## Add new Guarantee:

Admin Panel

Dashboard

Manage Users

Manage Guarantees

Uploaded Files

Logout

Add New Guarantee

Corporate Reference Number

Guarantee Type

Select a Type

Nominal Amount

Currency

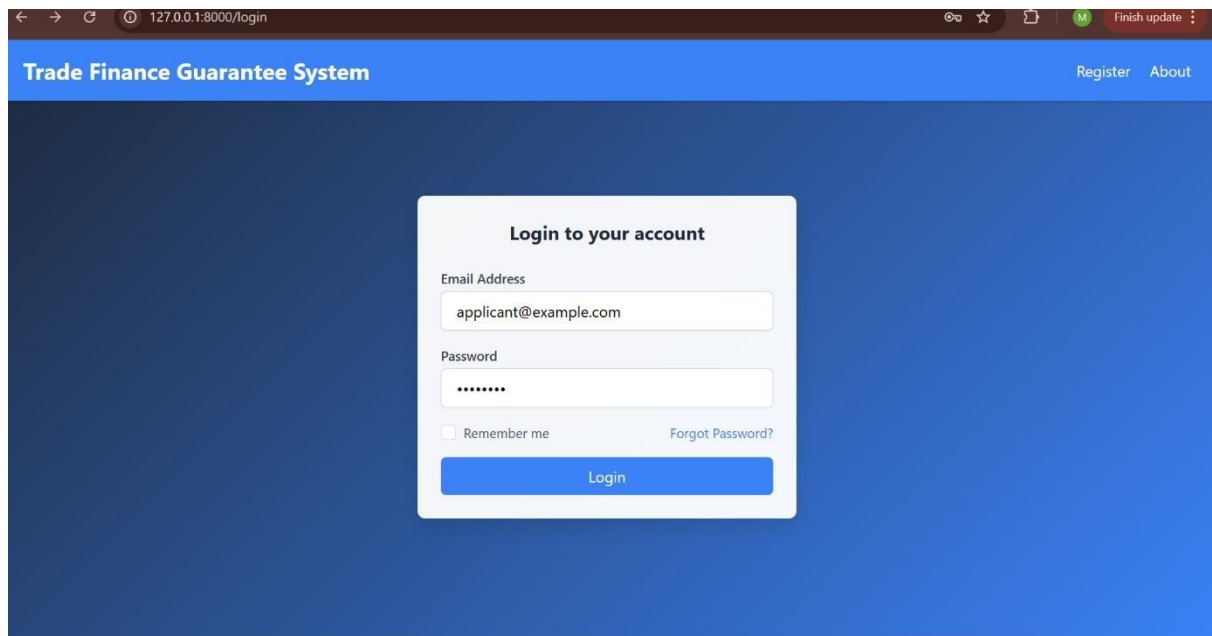
Expiry Date

dd-mm-yyyy

Applicant Name

Applicant Address

## Applicant Login:



Trade Finance Guarantee System [Register](#) [About](#)

**Login to your account**

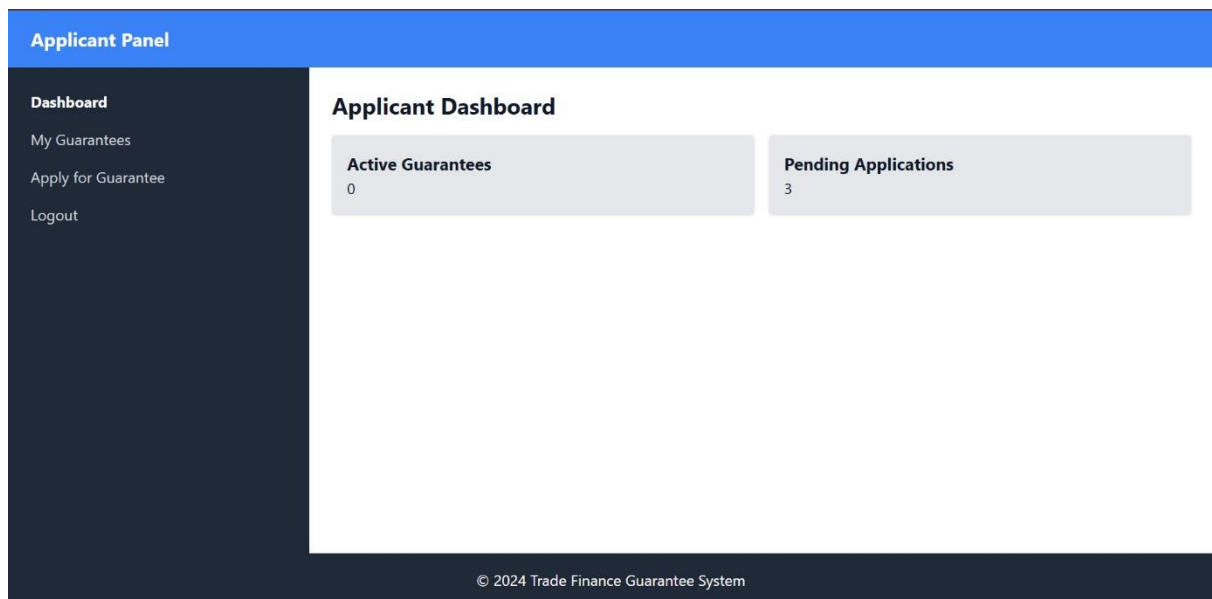
Email Address  
applicant@example.com

Password  
\*\*\*\*\*

☐ Remember me [Forgot Password?](#)

Login

## Applicant Dashboard:



**Applicant Panel**

**Dashboard**

- My Guarantees
- Apply for Guarantee
- Logout

**Applicant Dashboard**

**Active Guarantees**  
0

**Pending Applications**  
3

© 2024 Trade Finance Guarantee System

Submitted Guranatee:

Applicant Panel

Dashboard

My Guarantees

Apply for Guarantee

Logout

My Guarantees

Corporate Reference Number	Guarantee Type	Status	Actions
CRN123435	Bid Bond	pending	<a href="#">Edit</a> <a href="#">Delete</a>
23432	Bid Bond	pending	<a href="#">Edit</a> <a href="#">Delete</a>
23456	Insurance	pending	<a href="#">Edit</a> <a href="#">Delete</a>

© 2024 Trade Finance Guarantee System

Edit Guarantee:

Applicant Panel

Dashboard

My Guarantees

Apply for Guarantee

Logout

Edit Guarantee

Corporate Reference Number

CRN123435

Guarantee Type

Bid Bond

Nominal Amount

10230.00

Expiry Date

24-11-2025

Beneficiary Name

Jane Beneficiary

Beneficiary Address

456 Beneficiary Avenue

## Apply for new Guarantee:

Applicant Panel

Dashboard

**My Guarantees**

**Apply for Guarantee**

Logout

**Apply for a Guarantee**

Corporate Reference Number

Applicant Name

Applicant User

Applicant Address

Guarantee Type

Select Guarantee Type

Nominal Amount

## Conclusion

The Laravel-based **Trade Finance Guarantee Issuance System** showcases an efficient and secure solution for managing guarantees with features like authentication, CRUD operations, and bulk data uploads. The use of modern technologies, clear architecture, and thorough local deployment ensures the application is reliable, functional, and ready for real-world use. This project underscores the value of organized development practices and robust testing in building scalable web applications.