

BuildFund Deployment Guide

This document describes how to unpack and run the BuildFund platform with MongoDB persistence and the integrated Private Equity module. Follow these instructions to launch the application on your development or production environment.

Prerequisites

- Python 3.11 or higher with `pip` installed
- Node.js (v16 or later) and `npm`
- A running MongoDB instance (local or cloud)
- OpenAI and Google API keys for underwriting and mapping services

Unpack the Source Code

1. Unzip the provided archive (e.g. `buildfund_pe_mongo.zip`) to a directory of your choice.
2. The archive contains two main folders:
 - **buildfund_webapp** – Django REST API backed by MongoDB
 - **new_website** – React front end

Configure and Run the Backend

1. Navigate to the backend directory:

```
cd buildfund_webapp
```

2. Create and activate a virtual environment and install dependencies:

```
python -m venv venv
```

```
source venv/bin/activate
```

```
pip install -r requirements.txt
```

3. Copy the environment template and fill in required variables:

```
cp .env.example .env
```

Edit `.env` and set:

- `SECRET_KEY` – a unique Django secret
- `DEBUG` – set to `False` in production
- `ALLOWED_HOSTS` – comma-separated hostnames for your deployment
- `DB_ENGINE=djongo` – instructs Django to use MongoDB via Djongo
- `DB_NAME`, `DB_HOST`, `DB_PORT`, `DB_USER`, `DB_PASSWORD` – your MongoDB connection details
- `OPENAI_API_KEY` – your OpenAI secret key for underwriting
- `GOOGLE_API_KEY` – your Google Maps key for address search and geocoding

4. Run the setup script to prepare the database:

```
./scripts/setup_mongo.sh
```

This script applies all database migrations to MongoDB using Djongo.

5. (Optional) create an admin user so you can log in to the Django admin:

```
python manage.py createsuperuser
```

6. Start the backend server for development:

```
python manage.py runserver 0.0.0.0:8000
```

In production you should use Gunicorn or a similar WSGI server behind a reverse proxy.

Configure and Run the Frontend

1. In a new terminal, navigate to the front-end directory:

```
cd new_website
```

2. Install dependencies with npm:

```
npm install
```

3. Create a `.env` file in `new_website` with:

```
REACT_APP_API_BASE_URL=http://localhost:8000
```

Adjust the URL if your backend runs on a different host or port.

4. Start the development server:

```
npm start
```

For production builds, run `npm run build` and serve the static files using a web server such as Nginx.

Private Equity Module

The platform includes a Private Equity module allowing borrowers to list equity opportunities, lenders to invest, and administrators to approve submissions.

- **Borrowers:** Go to *Dashboard* → *Private Equity* to view your existing opportunities and create new ones. Fill in the title, description, industry, funding required, valuation and the equity percentage offered. New submissions are automatically marked as *pending review* and will be visible to lenders once approved by an administrator.
- **Lenders:** Navigate to *Dashboard* → *Private Equity* to view approved opportunities. For each opportunity you can enter an investment amount, the equity share you want and optional notes. Submitted investments will appear in your "My Investments" list.
- **Admins:** In the admin dashboard click *Private Equity* to review new submissions. Approve opportunities to make them available to lenders. Decline functionality can be added if desired.

Security Best Practices

Keep your API keys private by storing them in environment variables and never exposing them in client-side code. The backend enforces token-based authentication, rate limiting and CORS restrictions so only approved domains can call the APIs `637421499612537†L23-L31` `422390506697459†L266-L276` .

With these steps completed the BuildFund platform will be running locally, ready for further styling, testing and deployment. The integrated Private Equity module leverages the same secure architecture as the core project and fits seamlessly into the Borrower, Lender and Admin dashboards.