**🧱 1. Architecture & Structure**

* **Web Architecture:**  
  Client-server architecture (MVC-like separation using Flask or FastAPI backend).
* **Frontend-Backend Separation:**  
  HTML templates (Jinja2) rendered by backend with potential for future decoupling (e.g., Vue.js or React frontend).
* **Folder Structure:**
  + /templates – HTML pages
  + /static – CSS, JS, images
  + /modals – Generated HTML for previews
  + /db – SQLite or other DB storage
  + /admin – Admin interface modules

**🧪 2. Technologies Used**

**Backend**

* **Python 3.10+**
* **Flask** *(or FastAPI)*
* **SQLite** *(development DB)*
  + Future-ready for PostgreSQL/MySQL

**Frontend**

* **HTML5, CSS3, JavaScript (Vanilla)**
* **Jinja2** (templating engine)
* **AJAX / Fetch API** for dynamic modal loading
* **Bootstrap / Tailwind CSS** *(optional)*

**Tools & Libraries**

* **Jinja2** – HTML rendering
* **Flask-WTF** *(optional)* – form handling
* **SQLAlchemy** – ORM for DB models
* **Flask-Login** *(optional)* – user authentication
* **PyTest** – for testing
* **Git** – version control
* **Docker** *(optional)* – for deployment

**🖥️ 3. Hosting & Deployment**

* **Localhost / PythonAnywhere / Render / Vercel**
* **Gunicorn + Nginx** *(for production Linux servers)*
* **Environment configuration via .env files**
* **Database migrations** via Alembic or Flask-Migrate *(optional)*

**🛡️ 4. Security Tools**

* **CSRF protection** – Flask-WTF or built-in middleware
* **Content sanitization** – Bleach or similar for user HTML
* **HTTPS configuration** – via certbot or hosting provider
* **Password hashing** – werkzeug.security or bcrypt

**📦 5. Scalability & Future Readiness**

* Upgrade to:
  + PostgreSQL / MySQL for larger datasets
  + Redis for caching modal HTML
  + React/Vue frontend if SPA is required
* REST or GraphQL API layer for external integration
* CI/CD via GitHub Actions, GitLab CI, or Jenkins

**🧰 6. Development Tools**

* **IDE:** VSCode, PyCharm
* **Version Control:** Git + GitHub/GitLab
* **Testing Tools:** PyTest, Coverage.py
* **Task Tracker:** Trello / Notion / GitHub Projects
* **API Testing:** Postman or Insomnia