

# Matty - System Design Document

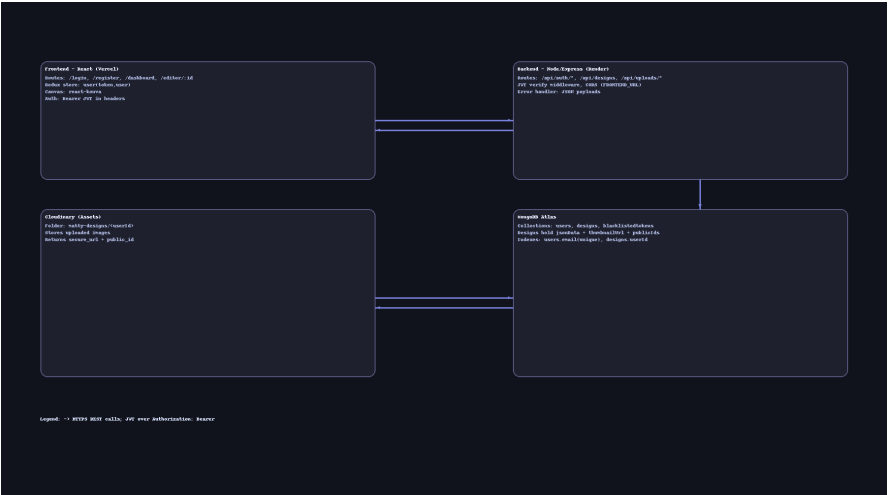
Updated: 2025-08-12

## 1. Introduction

Matty is a web-based graphic design tool built on the MERN stack. Users can create designs using a Konva.js canvas, upload assets to Cloudinary, and save designs as JSON + thumbnail in MongoDB.

## 2. Architecture Overview

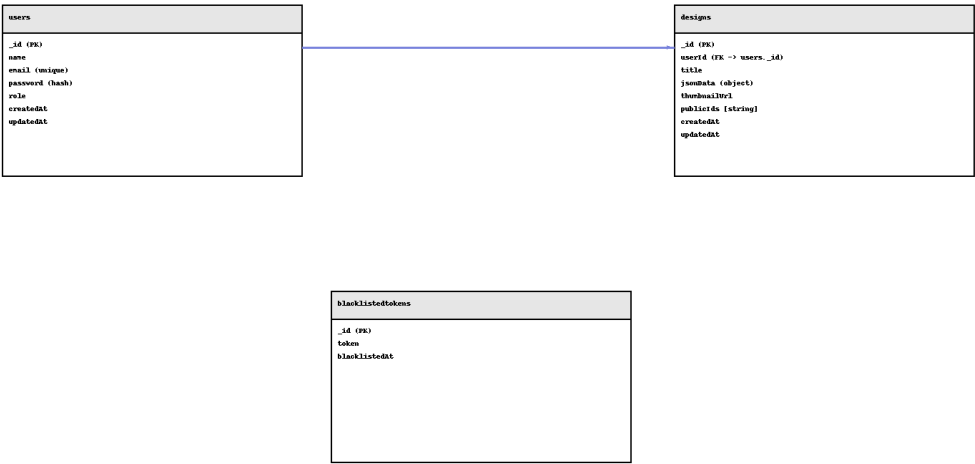
Frontend (Vercel) communicates with Backend (Render) via REST over HTTPS. JWT is sent in Authorization headers. Backend persists users/designs in MongoDB Atlas and stores images in Cloudinary.



## 3. Component Diagram



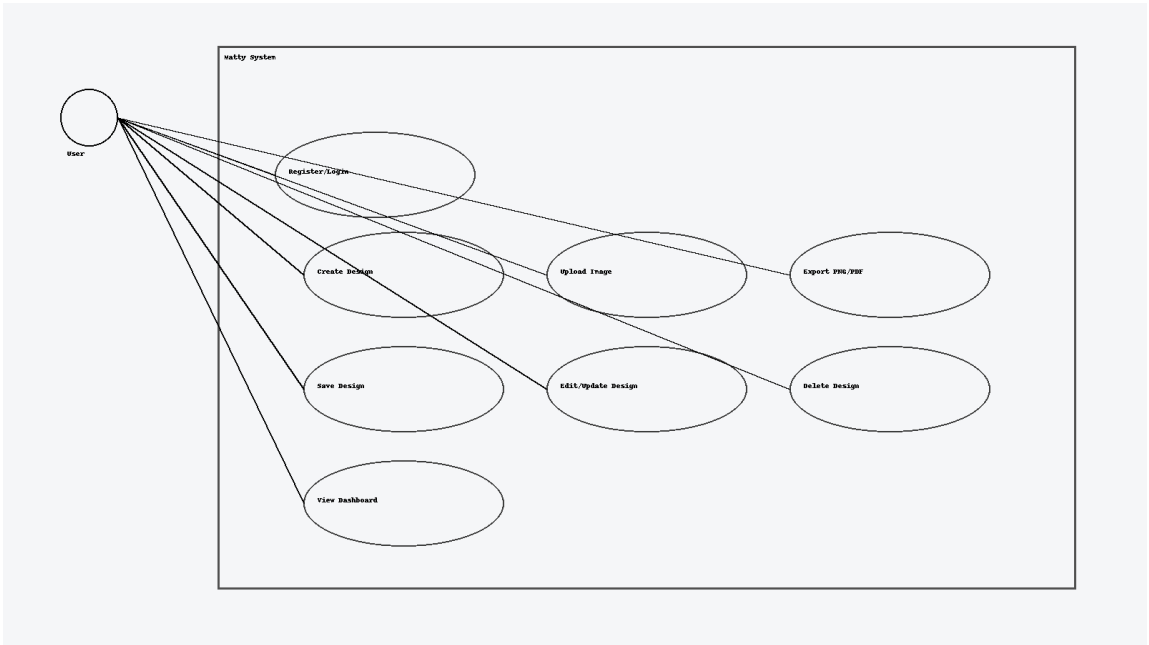
## 4. Data Model & ER Diagram



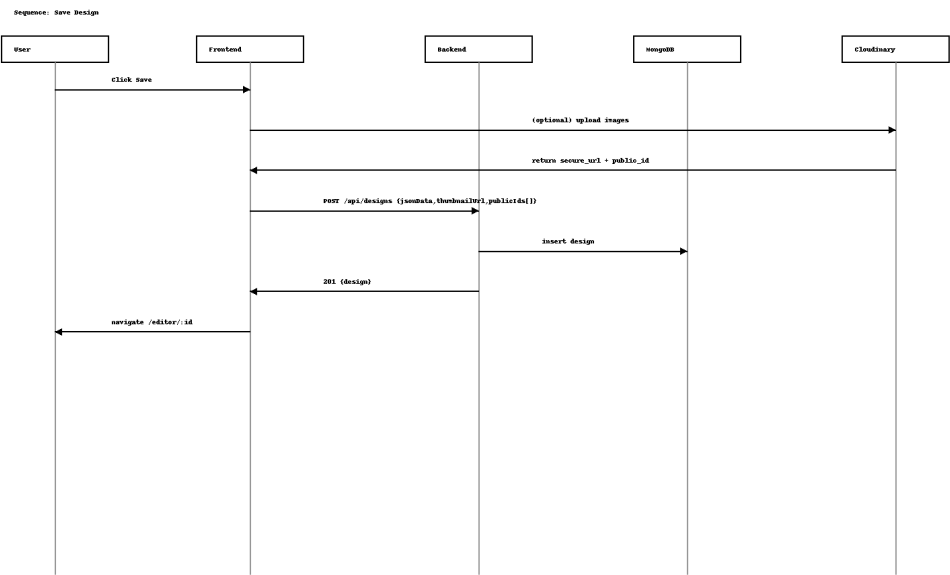
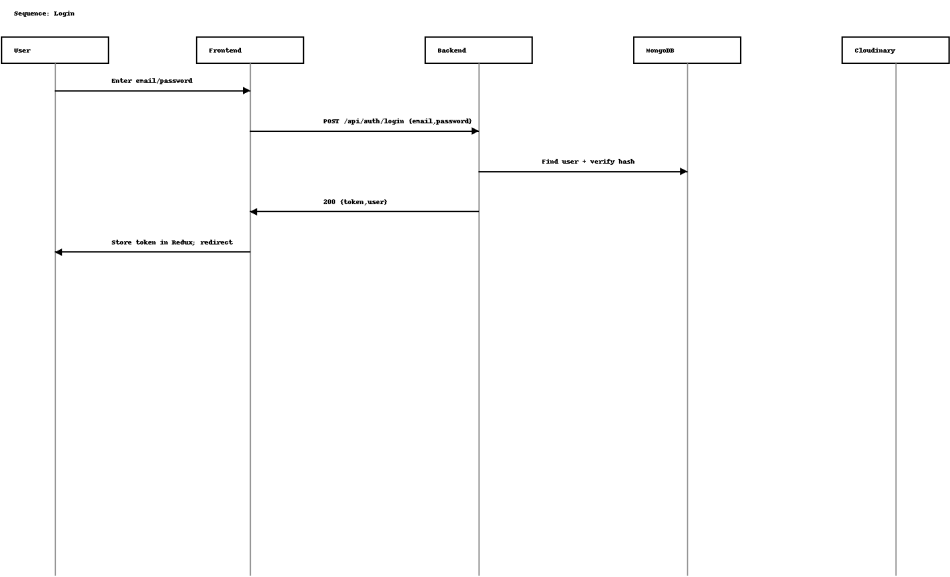
## 5. Data Dictionary

See `Data_Model/Data_Dictionary.xlsx` for all collections, fields, types, and semantics.

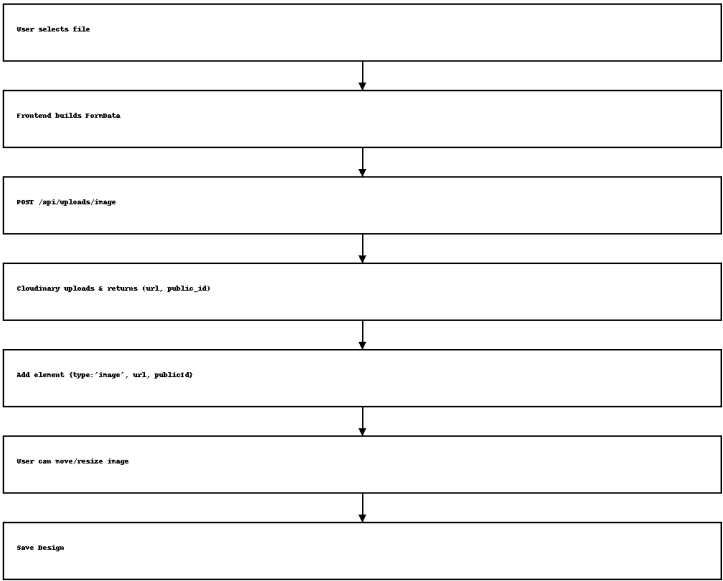
## 6. UML: Use Case



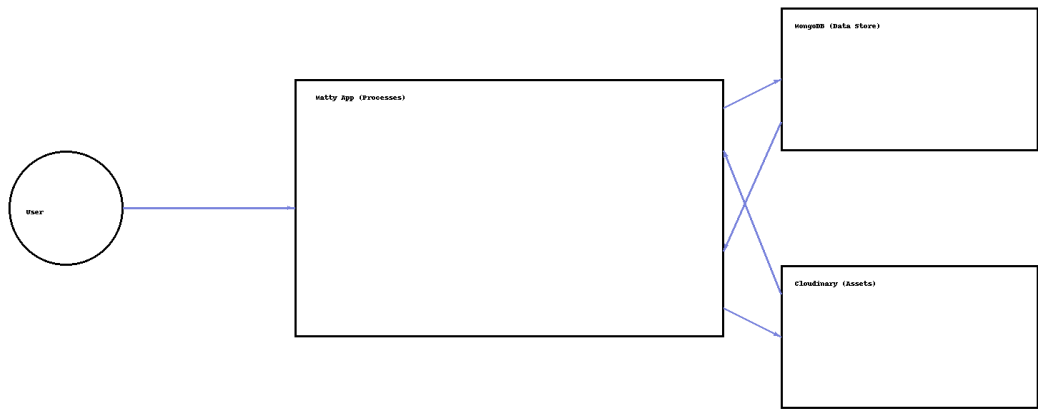
# 7. UML: Sequence (Login / Save Design)



# 8. Activity Diagram (Upload Image)



9. Data Flow Diagram (Level 0)



## 10. API Contracts

Endpoints grouped under `/api/auth`, `/api/designs`, `/api/uploads`. See `System_Design/API/API_Contracts.md` for full request/response schemas and auth requirements.

## 11. State Management

- Redux slice: `user { token, user }`
- Selectors: `useAuth()` returns `{ token, user, isLoggedIn }`
- Axios instance attaches Authorization header dynamically

## 12. Security Design

- JWT signed with `SERVER JWT_SECRET`; 1d expiry
- Auth middleware verifies token; `userId` injected into req
- CORS restricts origins to `FRONTEND_URL`
- Cloudinary uses signed API keys on server; `publicId`-only exposed
- Server-side validation on payloads; limit file size & types

## 13. Performance & Reliability

- Thumbnails stored as data URLs reduce initial latency from Cloudinary
- Konva stage uses minimal layers; avoid unnecessary re-renders
- Indexes on `users.email` and `designs.userId` improve queries
- Graceful error handler returns `JSON {message}`; logs server-side

## 14. Deployment Topology

- Frontend: Vercel build (Vite) with env var `VITE_REACT_APP_API_URL`
- Backend: Render, Node 18+, env vars (`JWT_SECRET`, `MONGO_URI`, `CLOUDINARY_*`)
- DB: MongoDB Atlas (shared cluster)
- DNS: Custom domain optional; HTTPS enforced