**📄 TimekinV1 – Tech Stack Overview**

**1. Frontend**

| **Category** | **Choice** | **Reasoning** |
| --- | --- | --- |
| **Framework** | React (w/ Vite) | Fast build setup, rich ecosystem, JSX familiarity |
| **Language** | TypeScript | Type safety, better developer experience |
| **Styling** | Tailwind CSS | Utility-first, fast prototyping, responsive by default |
| **State Mgmt** | Context API | Lightweight; no need for Redux for V1 |
| **Image Handling** | React Dropzone or native input[type=file] | Simpler for MVP |

**2. Backend**

| **Category** | **Choice** | **Reasoning** |
| --- | --- | --- |
| **Runtime** | Node.js | Popular, aligns with front-end language |
| **Framework** | Express | Lightweight, beginner-friendly, fast setup |
| **API Type** | REST | Simple, predictable, easy to document/test |
| **Language** | TypeScript | Shared types between frontend and backend |

**3. Database**

| **Category** | **Choice** | **Reasoning** |
| --- | --- | --- |
| **Type** | Document (NoSQL) | Flexible schema suits journal entries |
| **DB Engine** | MongoDB | Popular, good developer UX, Atlas free tier available |
| **Client** | Mongoose | Schema definition, middleware support, validation |

**4. Media Storage**

| **Category** | **Choice** | **Reasoning** |
| --- | --- | --- |
| **Provider** | AWS S3 | Scalable, widely adopted, well-documented |
| **Upload Flow** | Pre-signed URLs | Secure, efficient media upload from client directly to S3 |

**5. Infrastructure**

| **Category** | **Choice** | **Reasoning** |
| --- | --- | --- |
| **IaC Tool** | Terraform | Industry-standard, reusable for future work |
| **Cloud Host** | AWS | Unified ecosystem for S3, EC2, RDS, etc. |
| **Deployment** | Render (or AWS EC2) | For simplicity unless full container orchestration is needed |

**6. CI/CD**

| **Category** | **Choice** | **Reasoning** |
| --- | --- | --- |
| **Pipeline** | GitHub Actions | Integrated with GitHub, widely used |
| **Steps** | Lint → Build → Test → Deploy | Standard pipeline progression |
| **Env Mgmt** | Dotenv / GitHub Secrets | Manage .env securely and inject into builds |

**7. Other**

| **Category** | **Choice** | **Reasoning** |
| --- | --- | --- |
| **Version Control** | Git (GitHub) | Default standard |
| **Linting** | ESLint + Prettier | Maintain consistent code style |
| **Testing** | Jest (unit), Supertest (API) | Enough for basic confidence in backend |
| **Documentation** | Markdown README + inline JSDoc | Fast, minimal, dev-friendly |