



# Hard Skills Roadmap (3 Months)



## 1. Product & Design Foundation

- **Figma (UI/UX basics)**
    - Create wireframes & hi-fi mockups
    - Export design specs for developers
    - Build simple design system (buttons, colors, typography)
  - **Outcome:** Can take an idea → design a simple user flow → hand off to dev (himself).
- 



## 2. Core Web Building Blocks

- **HTML5**
    - Semantic elements (`header`, `main`, `footer`, `section`)
    - Accessibility tags (ARIA, alt text, roles)
  - **CSS3**
    - Flexbox, Grid, responsive layouts
    - Animations, transitions, media queries
    - Tailwind CSS utility-first styling
  - **JavaScript (ES6+)**
    - Variables, functions, loops, conditionals
    - DOM manipulation, events, fetch API
    - Async/await, promises, modules
  - **Outcome:** Can build static, responsive websites that look good on any device.
- 



## 3. Frontend Framework & Typing

- **React**
  - JSX, components, props, state
  - Hooks: `useState`, `useEffect`, `useContext`, `useRef`
  - Routing (React Router or Next.js)
- **TypeScript**
  - Types, interfaces, generics
  - Strict typing in React props & states
- **Outcome:** Can build SPAs (single-page apps) with modular, reusable components.



## 4. State & Data Handling

- **State Management**
    - Context API
    - Lightweight libraries: Zustand or Redux Toolkit
  - **Forms & Validation**
    - Controlled vs uncontrolled inputs
    - Libraries: react-hook-form + zod/yup
  - **APIs**
    - REST (fetch/axios)
    - Basics of GraphQL (optional)
  - **Outcome:** Can build apps that fetch/store data, validate forms, handle errors.
- 



## 5. Backend & Infrastructure (Fullstack layer)

- **Node.js + Express / Next.js API routes**
  - **Databases**
    - PostgreSQL or MongoDB
    - ORM: Prisma or Mongoose
  - **Authentication**
    - JWT, sessions, password hashing
    - OAuth basics (Google login, etc.)
  - **Deployment**
    - Vercel/Netlify (frontend)
    - Railway/Render/Heroku (backend)
    - GitHub Actions (CI/CD)
  - **Outcome:** Can ship a fullstack product: login/signup + database + deployed API.
- 



## 6. Testing, Performance & Monitoring

- **Testing**
  - Unit (Jest)
  - Integration/UI (React Testing Library)
  - E2E (Playwright/Cypress)
- **Performance**
  - Lighthouse audits, lazy loading, code splitting
- **Monitoring**
  - Sentry, LogRocket, Google Analytics
- **Outcome:** Can ship production-ready apps with automated tests & monitoring.

## 7. AI-Enabled Frontend Skills

- **AI APIs**
    - OpenAI API basics (chat completions, embeddings)
    - Hugging Face Inference API
  - **Vector DBs**
    - Pinecone / Weaviate / local embeddings
  - **Frameworks**
    - LangChain.js concepts (chains, RAG, agents)
  - **AI UI/UX**
    - Chat UI with streaming responses
    - Token streaming, loading indicators, fallbacks
  - **Outcome:** Can integrate AI features (chatbot, summarizer, RAG search) into apps.
- 

## 8. Advanced (Optional but valuable)

- **DevOps Basics**
    - Docker (containerize frontend + backend)
    - CI/CD pipelines for staging/production
  - **Security**
    - HTTPS, CORS, XSS, CSRF protection
  - **Scaling**
    - Caching, rate limiting, pagination
  - **Outcome:** Prepared to handle production-grade scaling and team workflows.
- 

## Final Checklist (by end of 3 months he should be able to...)

- Design a product flow in Figma
- Convert design → responsive React+TS UI
- Manage app state, forms, API data
- Build backend APIs & connect DB
- Deploy fullstack app with auth & CRUD
- Add tests, improve performance, monitor errors
- Integrate AI features (chatbot, RAG, summarizer)
- Present the product with a clean repo, docs, and demo