



CertTracker Next Development Steps



Phase 1: Authentication System (NEXT - 30 mins)

1.1 Sign Up / Login Pages

- Beautiful authentication forms
- Split-screen design with imagery
- Form validation and error handling
- "Sign In" and "Get Started" button functionality

1.2 AWS Cognito Integration

- User registration with email verification
- Login/logout functionality
- Password reset flow
- Session management

1.3 Protected Routes

- Route guards for authenticated users
 - Redirect logic (login → dashboard, logout → landing)
-



Phase 2: Dashboard & Core Features (45 mins)

2.1 Main Dashboard

- Stats overview (active, expiring, renewed)
- Recent certifications list
- Quick action buttons
- Search and filter functionality

2.2 Certification Management

- Add new certification form
- Edit existing certifications
- Delete certifications
- Upload certificate files/images

2.3 Expiration Tracking

- Visual calendar view
 - Expiration status indicators
 - Sorting by expiry date
 - Renewal reminder system
-

Phase 3: Advanced Features (30 mins)

3.1 Notifications System

- Email reminder preferences
- In-app notification center
- Customizable reminder intervals
- Notification history

3.2 Reporting & Analytics

- Certification portfolio overview
 - Expiration timeline charts
 - Certification value tracking
 - Export to PDF/CSV
-

Phase 4: Backend Integration (Optional)

4.1 AWS Infrastructure

- API Gateway setup
- Lambda functions
- DynamoDB tables
- S3 file storage

4.2 Real Data Integration

- Connect frontend to real APIs
- File upload functionality
- Data persistence
- User profile management

Immediate Next Steps (Let's Start Now!)

Step 1: Authentication Pages (15 mins)

1. Login Form Component

- Email/password inputs
- Remember me checkbox
- Forgot password link
- Professional styling

2. Sign Up Form Component

- Name, email, password fields
- Terms of service checkbox
- Email verification flow
- Success messaging

3. Route Integration

- Wire up "Sign In" and "Get Started" buttons
- Add navigation between auth pages
- Form submission handling

Step 2: Dashboard Shell (15 mins)

1. Dashboard Layout

- Header with user info and logout
- Sidebar navigation
- Main content area
- Responsive design

2. Basic Data Display

- Mock certification data
- Statistics cards
- Recent activity feed
- Quick action buttons





- Use same color scheme (blue primary)
 - Maintain professional typography
 - Consistent button and form styling
 - Mobile-responsive design
 - Smooth transitions and animations
-

Ready to Build?

I recommend we start with the authentication pages since:

- Users expect to click those buttons and see something
- Sets up the foundation for the rest of the app
- Gives immediate functionality to test
- Most visually impactful next step

Which component would you like to build first?

1.  **Login Page** (quick win)
2.  **Sign Up Page** (complete auth flow)
3.  **Dashboard Layout** (see the full app vision)
4.  **Wire up navigation** (make buttons work)

Let's keep the momentum going! 