Here's a **tailored Software Development Life Cycle (SDLC)** for building your **MERN web app — Adaptive Learning Path (ALP)**, step-by-step:

**🧠 1. Requirement Gathering**

Understand what you're building and for whom.

**Key Activities:**

* Interview **therapists**, **educators**, and **parents**
* Define:
  + Core features (e.g., real-time emotion tracking, game progression logic)
  + Users: therapists, children, admins
* Identify non-functional needs: **data privacy**, **responsiveness**, **low latency**
* Document expected **AI outputs** (emotion labels, engagement levels)

**📐 2. System Design**

Plan how the system will work and look.

**Architecture:**

* **Frontend (React.js)**:
  + UI for kids (simple, fun, responsive)
  + Therapist dashboard
* **Backend (Node.js + Express)**:
  + Game logic
  + User management
  + Interface with AI APIs (e.g., Hugging Face ViT or custom transformer models)
* **Database (MongoDB)**:
  + Store user profiles, session logs, expression data, learning path decisions
* **AI Integration Layer**:
  + Calls expression analysis APIs
  + Maps emotional cues to game state decisions
* **Rule Engine**:
  + Applies therapist’s path with adaptive constraints

**💻 3. Implementation (Coding)**

Break it into modules and start building.

**Suggested Modules:**

* 🧩 Games Module (educational + expression-trackable)
* 👦 Emotion Detection Integration (camera + transformer API)
* 🧭 Adaptive Engine (responds to behavior/emotion + therapist rules)
* 🧑‍⚕️ Therapist Dashboard (recommend paths, track progress)
* 📊 Session Analytics (emotion trend reports, engagement patterns)

**🧪 4. Testing**

Ensure the system behaves as expected.

**Types of Testing:**

* **Unit Testing** (React components, API routes)
* **Integration Testing** (e.g., gameplay → expression → path update)
* **User Acceptance Testing (UAT)** with therapists
* **Security Testing** (facial data handling, GDPR-compliance if needed)

**🚀 5. Deployment**

Make the app live and usable.

**Stack:**

* **Frontend**: Deployed on Vercel or Netlify
* **Backend**: Deployed on Heroku, AWS, or DigitalOcean
* **DB**: MongoDB Atlas
* **AI Model**: Hugging Face APIs or self-hosted inference server

Add observability:

* Logs, performance metrics, crash analytics

**🛠️ 6. Maintenance & Feedback**

Update based on real-world usage.

* Monitor: therapist feedback, child engagement metrics
* Improve model mappings: emotion → path tweaks
* Add games, refine UI, patch bugs

**Bonus:**

You can follow **Agile methodology** for SDLC with:

* **2-week sprints**
* **Daily standups**
* **Demo and feedback sessions with therapists**