Some basic corrections:

* Remove start free trail from the three lines drop down – there are enough places they can access the free trial on the home page.
* Remove “starts with welcome & introductions – Script 1 of 9” unneeded.
* In the relaxation tools – we need to remove the options for the user to add their own music, or create a playlist, seems too complicated, otherwise we will need to think about storing these, instead provide two tracks for either of the options (calm, sleep, etc.,) can you add the music here?

Here are some changes to help you understand the flow of the phases - we have got a good base and are close to it being excellent, but we need to be clear on the structure and user experience and how we integrate the different parts of the process.

Firstly, to reiterate here is the order of the phases and the scripts and some changes to how we will name the scripts within the app.

SCRIPT 1: Welcome & Introduction to EMDR

SCRIPT 2: Target Memory Setup

(In the app we will call this: Setting up the Target Memory)

SCRIPT 3: Desensitisation – Initial Setup

(In the app we will call this: Desensitisation and Reprocessing)

SCRIPT 4: After a Set – What Do You Notice Now (loop)?

(In the app we will call this: Desensitisation and Reprocessing Continued)

SCRIPT 6: Installation Phase

(In the app we will call this: Installation of the Positive Belief)

SCRIPT 7: Installation continued (loop)

(In the app we will call this: Installation of the Positive Belief Continued)

SCRIPT 8: Body Scan

SCRIPT 9: Safe Closure

(In the app we will call this: Calm Place)

This app follows the standard eight-phase EMDR protocol, simplified for app-based delivery using video guidance and interactive elements.

Below is the intended user flow and structure, starting from the moment the user engages with the app.

-------------------------------------------------------------

ACCESS POINT TO BEGIN EMDR FLOW

-------------------------------------------------------------

Users can access the Welcome & Introduction phase from any of the following buttons:

• "Sign Up – Start Free Trial"

• "Start Free Trial to Begin"

Clicking either will take them directly to the Welcome & Introduction phase of the app.

-------------------------------------------------------------

1. WELCOME & INTRODUCTION

-------------------------------------------------------------

- This page introduces EMDR and the therapist via a short video.

- The video explains:

• What EMDR is

• What to expect from the app

• How the therapist will guide the process

- After watching, the user clicks a button labelled:

• “Start EMDR” or “I’m Ready to Begin”

This moves them into the structured EMDR session flow.

-------------------------------------------------------------

2. PHASE 2: SETTING UP A TARGET MEMORY

(in-app name: Setting Up the Target Memory)

-------------------------------------------------------------

- User is guided by video to enter the following information:

• Target memory or image

• Negative cognition (e.g., "I am powerless")

• Positive cognition (e.g., "I am in control now")

• VOC (Validity of Cognition) rating (1–7)

• Emotions felt when focusing on the memory

• SUDS rating (0–10)

• Location of distress in the body

- When all fields are completed, user clicks “Continue”.

-------------------------------------------------------------

3. PHASE 4: DESENSITISATION AND REPROCESSING

(in-app name: Desensitisation and Reprocessing)

-------------------------------------------------------------

- This is the core engine of the app.

STRUCTURE:

• Therapist video explains the task

• App runs a first set of BLS (e.g. 22 repetitions)

• App then prompts: "What do you notice now?"

• User types a brief response or reflects internally

LOOP BEGINS:

• Therapist says: “Just notice that.”

• App runs another BLS set

• Repeat the cycle (BLS → prompt → response → BLS)

• Continue looping until user reports low/no distress

When SUDS = 0 (or distress has significantly reduced), app transitions to next phase.

-------------------------------------------------------------

4. PHASE 5: INSTALLATION

(in-app name: Installation of the Positive Belief)

-------------------------------------------------------------

- Therapist instructs user to:

• Focus on the memory + positive belief

• Follow BLS (same as before)

LOOP:

• After each BLS set, prompt: "Is the belief getting stronger, weaker or staying the same?"

• User responds or reflects

• Loop continues until VOC = 7

• Then, app proceeds to next phase

-------------------------------------------------------------

5. PHASE 6: BODY SCAN

-------------------------------------------------------------

- Therapist instructs user to:

• Bring up the memory + positive belief

• Slowly scan their body from head to toe

IF DISTURBANCE IS PRESENT:

• App runs additional BLS

• User is prompted to scan again

LOOP CONTINUES until no disturbance is reported in body scan.

-------------------------------------------------------------

6. PHASE 7: SAFE CLOSURE

(in-app name: Calm Place)

-------------------------------------------------------------

- Therapist guides user into a calming visualisation

- Option to revisit calm place or end session

- End screen confirms memory processed or marked as incomplete

-------------------------------------------------------------

ADDITIONAL NOTES

-------------------------------------------------------------

- All therapist scripts are delivered via video

- After each phase, user clicks a button like “Next” or “I’m ready”

- BLS types (visual, audio, tapping) must be selectable and available during desensitisation, installation, and body scan

- Visual BLS = ball animation

- Audio BLS = stereo beeps

- Tapping BLS = user taps chest or thighs guided by cues

- Users must be able to pause or exit safely at any point

-------------------------------------------------------------

LOOPS TO SUPPORT IN CODE

-------------------------------------------------------------

• Reprocessing Loop (BLS → prompt → user input → repeat)

• Installation Loop (BLS → belief strength check → repeat)

• Body Scan Loop (Scan → detect disturbance → BLS → rescan)

There are some further changes to the structural flow between the phases which I will paste here:

1. **TargetSetupScreen.tsx**

import React, { useState } from "react";

export default function TargetSetupScreen({ onContinue }: { onContinue: () => void }) {

const [image, setImage] = useState("");

const [negCog, setNegCog] = useState("");

const [posCog, setPosCog] = useState("");

const [suds, setSuds] = useState(0);

const [voc, setVoc] = useState(1);

return (

<div style={{ padding: "2rem", maxWidth: "600px", margin: "0 auto" }}>

<h2>Set Up Your Target Memory</h2>

<label>Image Associated with Memory</label>

<input type="text" value={image} onChange={(e) => setImage(e.target.value)} />

<label>Negative Cognition</label>

<input type="text" value={negCog} onChange={(e) => setNegCog(e.target.value)} />

<label>Positive Cognition</label>

<input type="text" value={posCog} onChange={(e) => setPosCog(e.target.value)} />

<label>Validity of Positive Cognition (VOC, 1–7)</label>

<input

type="number"

min={1}

max={7}

value={voc}

onChange={(e) => setVoc(parseInt(e.target.value))}

/>

<label>Subjective Units of Distress (SUDS, 0–10)</label>

<input

type="number"

min={0}

max={10}

value={suds}

onChange={(e) => setSuds(parseInt(e.target.value))}

/>

<button onClick={onContinue} style={{ marginTop: "2rem" }}>

Start Processing

</button>

</div>

);

}

2. **BLSSession.tsx**

import React, { useState } from "react";

export default function BLSSession({ onInstallPhase }: { onInstallPhase: () => void }) {

const [setCount, setSetCount] = useState(1);

const [userInput, setUserInput] = useState("");

const [phase, setPhase] = useState<"bls" | "prompt" | "complete">("bls");

const handleBLSComplete = () => {

setPhase("prompt");

};

const handleNextSet = () => {

setSetCount(setCount + 1);

setUserInput("");

setPhase("bls");

};

const handleEndSession = () => {

setPhase("complete");

};

return (

<div style={{ padding: "2rem", maxWidth: "700px", margin: "0 auto" }}>

<h2>Desensitisation & Reprocessing</h2>

<p>Set {setCount}</p>

{phase === "bls" && (

<div>

<p>Bilateral stimulation running (visual/audio/tapping)...</p>

<button onClick={handleBLSComplete}>End BLS Set</button>

</div>

)}

{phase === "prompt" && (

<div style={{ marginTop: "2rem" }}>

<p>What do you notice now?</p>

<textarea

rows={4}

value={userInput}

onChange={(e) => setUserInput(e.target.value)}

style={{ width: "100%" }}

/>

<div style={{ marginTop: "1rem" }}>

<button onClick={handleNextSet}>Just Notice That & Continue</button>

<button onClick={handleEndSession} style={{ marginLeft: "1rem" }}>

No Further Distress – Move to Installation

</button>

</div>

</div>

)}

{phase === "complete" && (

<div>

<p>Great. Let's move on to the next stage.</p>

<button onClick={onInstallPhase}>Start Installation Phase</button>

</div>

)}

</div>

);

}

3. **InstallationPhase.tsx**

import React, { useState } from "react";

export default function InstallationPhase({ onNext }: { onNext: () => void }) {

const [voc, setVoc] = useState(1);

const [setCount, setSetCount] = useState(1);

const [userInput, setUserInput] = useState("");

const [phase, setPhase] = useState<"bls" | "prompt" | "complete">("bls");

const handleBLSComplete = () => {

setPhase("prompt");

};

const handleNextSet = () => {

setSetCount(setCount + 1);

setUserInput("");

setPhase("bls");

};

const handleVOCChange = (e: React.ChangeEvent<HTMLInputElement>) => {

setVoc(parseInt(e.target.value));

};

const handleEndSession = () => {

if (voc === 7) {

setPhase("complete");

} else {

alert("Please continue until the VOC reaches 7.");

}

};

return (

<div style={{ padding: "2rem", maxWidth: "700px", margin: "0 auto" }}>

<h2>Installation of Positive Belief</h2>

<p>Set {setCount}</p>

{phase === "bls" && (

<div>

<p>Follow the BLS (visual/audio/tapping) while holding the memory and the new belief in mind.</p>

<button onClick={handleBLSComplete}>End BLS Set</button>

</div>

)}

{phase === "prompt" && (

<div style={{ marginTop: "2rem" }}>

<label>VOC (1–7): </label>

<input type="number" value={voc} min={1} max={7} onChange={handleVOCChange} />

<p>Is the belief getting stronger, weaker, or staying the same?</p>

<textarea

rows={3}

value={userInput}

onChange={(e) => setUserInput(e.target.value)}

style={{ width: "100%" }}

/>

<div style={{ marginTop: "1rem" }}>

<button onClick={handleNextSet}>Continue BLS</button>

<button onClick={handleEndSession} style={{ marginLeft: "1rem" }}>

VOC = 7 – Continue to Body Scan

</button>

</div>

</div>

)}

{phase === "complete" && (

<div>

<p>Great. Let's move on to the body scan.</p>

<button onClick={onNext}>Start Body Scan</button>

</div>

)}

</div>

);

}

4. **BodyScanPhase.tsx**

import React, { useState } from "react";

export default function BodyScanPhase({ onNext }: { onNext: () => void }) {

const [disturbance, setDisturbance] = useState(false);

const [blsRound, setBlsRound] = useState(0);

const [phase, setPhase] = useState<"scan" | "bls" | "complete">("scan");

const handleStartBLS = () => {

setPhase("bls");

setBlsRound(blsRound + 1);

};

const handleBLSComplete = () => {

setDisturbance(false);

setPhase("scan");

};

const handleComplete = () => {

if (!disturbance) {

setPhase("complete");

} else {

alert("Please continue BLS until there is no disturbance.");

}

};

return (

<div style={{ padding: "2rem", maxWidth: "700px", margin: "0 auto" }}>

<h2>Body Scan</h2>

{phase === "scan" && (

<div>

<p>Scan your body slowly from head to toe. Are there any lingering sensations or tension?</p>

<label>

<input

type="checkbox"

checked={disturbance}

onChange={(e) => setDisturbance(e.target.checked)}

/>

I notice some disturbance or discomfort

</label>

<div style={{ marginTop: "1rem" }}>

<button onClick={handleStartBLS}>Do More BLS</button>

<button onClick={handleComplete} style={{ marginLeft: "1rem" }}>

Body Scan Clear – Go to Calm Place

</button>

</div>

</div>

)}

{phase === "bls" && (

<div>

<p>Running BLS set {blsRound} to clear body tension...</p>

<button onClick={handleBLSComplete}>Stop BLS</button>

</div>

)}

{phase === "complete" && (

<div>

<p>Your body is clear. Well done.</p>

<button onClick={onNext}>Go to Calm Place</button>

</div>

)}

</div>

);

}

5. **CalmPlacePhase.tsx**

import React, { useState } from "react";

export default function CalmPlacePhase() {

const [sessionComplete, setSessionComplete] = useState(false);

const handleFinish = () => {

setSessionComplete(true);

};

return (

<div style={{ padding: "2rem", maxWidth: "700px", margin: "0 auto" }}>

<h2>Calm Place</h2>

{!sessionComplete ? (

<div>

<p>

Let's return to your calm place. Hold the image in your mind and focus on the feelings of safety and ease.

Follow the ball, tones, or tapping for a few rounds.

</p>

<p>

When you're ready, click below to finish your session.

</p>

<button onClick={handleFinish}>Finish Session</button>

</div>

) : (

<div>

<p>Your session is now complete.</p>

<p>Feel free to take a few moments to rest or reflect before closing the app.</p>

</div>

)}

</div>

);

}

SessionFlow.tsx

import React, { useState } from "react";

// Import each phase component

import TargetMemorySetup from "./TargetMemorySetup";

import ReprocessingPhase from "./ReprocessingPhase";

import InstallationPhase from "./InstallationPhase";

import BodyScanPhase from "./BodyScanPhase";

import CalmPlacePhase from "./CalmPlacePhase";

export default function SessionFlow() {

const [phaseIndex, setPhaseIndex] = useState(0);

const nextPhase = () => {

setPhaseIndex(prev => prev + 1);

};

const phases = [

<TargetMemorySetup onNext={nextPhase} />,

<ReprocessingPhase onNext={nextPhase} />,

<InstallationPhase onNext={nextPhase} />,

<BodyScanPhase onNext={nextPhase} />,

<CalmPlacePhase />

];

return (

<div>

{phases[phaseIndex]}

</div>

);

}