

```

import React, { useState, useEffect } from 'react';
import {
  Heart,
  Activity,
  AlertTriangle,
  CheckCircle2,
  ChevronRight,
  ChevronLeft,
  Syringe,
  Pill,
  FileText,
  RefreshCw,
  ShieldAlert,
  Info
} from 'lucide-react';

// --- Data & Logic based on SDCEP Guidance ---

const DRUG_CATEGORIES = [
  {
    id: 'doac',
    label: 'Direct Oral Anticoagulant (DOAC)',
    sublabel: 'Apixaban, Dabigatran, Rivaroxaban, Edoxaban',
    icon: <Pill className="w-6 h-6 text-blue-500" />
  },
  {
    id: 'vka',
    label: 'Vitamin K Antagonist',
    sublabel: 'Warfarin, Acenocoumarol, Phenindione',
    icon: <Activity className="w-6 h-6 text-amber-500" />
  },
  {
    id: 'antiplatelet',
    label: 'Antiplatelet Drug(s)',
    sublabel: 'Aspirin, Clopidogrel, Ticagrelor, Dipyridamole',
    icon: <Heart className="w-6 h-6 text-red-500" />
  },
  {
    id: 'injectable',
    label: 'Injectable Anticoagulant (LMWH)',
    sublabel: 'Dalteparin, Enoxaparin, Tinzaparin',
    icon: <Syringe className="w-6 h-6 text-purple-500" />
  }
];

const PROCEDURES = {
  lowRisk: [
    "Local anaesthesia (infiltration, intraligamentary, mental block)",
    "Basic Periodontal Examination (BPE)",
    "Supragingival scaling/polishing",
    "Direct/indirect restorations (supragingival margins)",
    "Endodontics (orthograde)",
    "Impressions / Prosthetics",
    "Orthodontic appliance fitting"
  ]
};

```

```

        ],
        higherRisk: [
            "Local anaesthesia (ID block - caution advised but generally low risk)",
            "Simple extractions (1-3 teeth, restricted wound)",
            "Complex extractions (surgical/flap)",
            "Multiple extractions (>3 at once)",
            "Subgingival scaling / RSD",
            "Restorations with subgingival margins",
            "Periodontal surgery / Crown lengthening",
            "Biopsies",
            "Implant surgery"
        ]
    };
}

const RISK_FACTORS = [
    "Liver or kidney disease",
    "Alcohol dependence",
    "Haematological malignancy / Chemotherapy",
    "Taking other bleeding-risk drugs (NSAIDs, SSRIs)",
    "History of bleeding complications"
];
// --- Components ---

const Card = ({ children, className = "" }) => (
    <div className={`${`bg-white rounded-xl shadow-sm border border-slate-200 overflow-hidden ${className}`}`}>
        {children}
    </div>
);

const Button = ({ onClick, variant = "primary", className = "", children, disabled = false }) => {
    const baseStyle = "px-4 py-3 rounded-lg font-medium transition-all duration-200 flex items-center justify-center gap-2 active:scale-95";
    const variants = {
        primary: "bg-blue-600 text-white hover:bg-blue-700 shadow-md hover:shadow-lg disabled:opacity-50 disabled:cursor-not-allowed",
        secondary: "bg-white border border-slate-200 text-slate-700 hover:bg-slate-50 disabled:opacity-50",
        danger: "bg-red-50 text-red-600 border border-red-100 hover:bg-red-100",
        outline: "border-2 border-blue-600 text-blue-600 hover:bg-blue-50"
    };

    return (
        <button onClick={onClick} disabled={disabled} className={`${baseStyle} ${variants[variant]} ${className}`}>
            {children}
        </button>
    );
};

```

```

const FadeIn = ({ children, delay = 0 }) => (
  <div className="animate-in fade-in slide-in-from-bottom-4 duration-500
fill-mode-forwards" style={{ animationDelay: `${delay}ms` }}>
    {children}
  </div>
);

// --- Main App ---

export default function ClinicalWorkup() {
  const [step, setStep] = useState(0);
  const [selection, setSelection] = useState({
    drugType: null,
    drugName: null,
    procedureRisk: null, // 'low' or 'higher'
    procedureName: null,
    inr: null, // for warfarin
    lmwhDose: null, // 'prophylactic' or 'treatment'
    factors: []
  });

  const reset = () => {
    setSelection({
      drugType: null,
      drugName: null,
      procedureRisk: null,
      procedureName: null,
      inr: null,
      lmwhDose: null,
      factors: []
    });
    setStep(0);
  };

  const handleDrugSelect = (drug) => {
    setSelection(prev => ({ ...prev, drugType: drug.id }));
    setStep(1);
  };

  const handleProcedureSelect = (risk, name) => {
    setSelection(prev => ({ ...prev, procedureRisk: risk, procedureName: name }));
  };

  // Skip factor check for simple flows in this demo, or go to specific
  // checks
  if (selection.drugType === 'vka') {
    setStep(2); // INR Check
  } else if (selection.drugType === 'injectable') {
    setStep(2); // Dose Check
  } else {
    setStep(3); // Result
  }
}

```

```

const handleInrSubmit = (val) => {
  setSelection(prev => ({ ...prev, inr: val }));
  setStep(3);
};

const handleLmwhSelect = (doseType) => {
  setSelection(prev => ({ ...prev, lmwhDose: doseType }));
  setStep(3);
};

// --- Result Generation Logic ---
const getRecommendation = () => {
  const { drugType, procedureRisk, inr, lmwhDose } = selection;

  // 1. WARFARIN (VKA)
  if (drugType === 'vka') {
    if (parseFloat(inr) >= 4.0) {
      return {
        status: "DELAY TREATMENT",
        color: "red",
        title: "INR is Unsafe (≥ 4.0)",
        content: (
          <div className="space-y-4">
            <p><strong>Do not proceed</strong> with invasive treatment.</p>
            <ul className="list-disc pl-5 space-y-2 text-sm">
              <li>Refer to GP or anticoagulation clinic to reduce INR.</li>
              <li>For urgent treatment, refer to secondary dental care.</li>
            </ul>
          </div>
        )
      };
    }
    // INR < 4
    return {
      status: "TREAT WITHOUT INTERRUPTION",
      color: "green",
      title: `INR is Safe (${inr})`,
      content: (
        <div className="space-y-4">
          <p className="font-medium">Continue Warfarin. Do not stop medication.</p>
          <div className="bg-blue-50 p-4 rounded-lg text-sm space-y-2">
            <div className="font-semibold text-blue-800 flex items-center gap-2">
              <CheckCircle2 size={16} /> Ensure INR check
            </div>
            <p>Ideally within 24 hours. If stable (no readings > 4 in last 2 months), up to 72 hours is acceptable.</p>
          </div>
          <div className="space-y-2 text-sm">
            <p className="font-semibold">Haemostatic Measures:</p>

```

```

        <ul className="list-disc pl-5 text-slate-600">
            <li>Pack and suture sockets (strongly considered).</li>
            <li>Limit initial treatment area (e.g., 1-3 teeth).</li>
            <li>Stage complex treatments.</li>
        </ul>
    </div>
</div>
)
};

// 2. DOACs
if (drugType === 'doac') {
    if (procedureRisk === 'lowRisk') {
        return {
            status: "TREAT WITHOUT INTERRUPTION",
            color: "green",
            title: "Low Bleeding Risk Procedure",
            content: (
                <div className="space-y-4">
                    <p>Treat without interrupting the DOAC medication.</p>
                    <ul className="list-disc pl-5 text-sm text-slate-600 space-y-1">
                        <li>Plan treatment for early in the day.</li>
                        <li>Limit initial treatment area.</li>
                        <li>Use local haemostatic measures.</li>
                    </ul>
                </div>
            )
        };
    }
    // Higher Risk
    return {
        status: "MODIFY MORNING DOSE",
        color: "amber",
        title: "Higher Bleeding Risk Procedure",
        content: (
            <div className="space-y-4">
                <div className="bg-amber-50 border border-amber-100 p-4 rounded-lg space-y-3">
                    <h4 className="font-bold text-amber-900 flex items-center gap-2">
                        <Activity size={18}/> Intervention Required
                    </h4>
                    <div className="text-sm space-y-2 text-amber-800">
                        <p><strong>Apixaban / Dabigatran (Twice daily):</strong><br/>MISS morning dose. Treat. Restart evening dose (min 4hrs after haemostasis).</p>
                        <hr className="border-amber-200"/>
                        <p><strong>Rivaroxaban / Edoxaban (Once daily - Morning):</strong><br/>DELAY dose. Treat. Take dose 4 hours after haemostasis.</p>
                        <p className="text-xs italic opacity-80">If taken in evening, no change needed.</p>
                    </div>
                </div>
            </div>
        )
    };
}

```

```

        </div>
    </div>
    <div className="text-sm">
        <p><strong>Clinical Advice:</strong> Plan treatment early
in day. Consider staging treatment.</p>
        </div>
    </div>
)
};

}

// 3. ANTIPLATELETS
if (drugType === 'antiplatelet') {
    return {
        status: "TREAT WITHOUT INTERRUPTION",
        color: "green",
        title: "Single or Dual Therapy",
        content: (
            <div className="space-y-4">
                <p className="font-medium">Do NOT stop Aspirin, Clopidogrel,
Ticagrelor, or Prasugrel.</p>
                <div className="bg-slate-50 p-4 rounded-lg text-sm text-
slate-700 border border-slate-200">
                    <p className="mb-2"><strong>Risk Warning:</strong> Stopping
these drugs (especially dual therapy for stents) carries a high risk of
thrombotic events.</p>
                </div>
                <div className="space-y-2 text-sm">
                    <p className="font-semibold">Management:</p>
                    <ul className="list-disc pl-5 text-slate-600">
                        <li>Expect prolonged bleeding.</li>
                        <li>Treat as normal using local measures.</li>
                        <li><strong>Dual Therapy:</strong> Strongly consider
suturing and packing.</li>
                    </ul>
                </div>
            </div>
        )
    };
}

// 4. INJECTABLES (LMWH)
if (drugType === 'injectable') {
    if (lmwhDose === 'treatment') {
        return {
            status: "CONSULT PRESCRIBER",
            color: "red",
            title: "Therapeutic (High) Dose",
            content: (
                <div className="space-y-4">
                    <p>Patient is on a treatment dose (e.g., for acute
DVT/PE).</p>
                    <div className="bg-red-50 p-4 rounded-lg border border-red-
100 text-red-800 text-sm">

```

```

        <ShieldAlert className="inline mr-2 w-4 h-4"/>
        <strong>Action:</strong> Consult with the prescribing
clinician/haematologist before proceeding with invasive treatment.
    </div>
</div>
)
};

}

return {
    status: "TREAT WITHOUT INTERRUPTION",
    color: "green",
    title: "Prophylactic (Low) Dose",
    content: (
        <div className="space-y-4">
            <p>Treat without interrupting medication.</p>
            <div className="text-sm text-slate-600">
                <p>Prophylactic doses generally pose low bleeding risk.</p>
                <ul className="list-disc pl-5 mt-2">
                    <li>Limit initial treatment area.</li>
                    <li>Suture and pack if higher risk procedure.</li>
                </ul>
            </div>
        </div>
    )
};

}

return { status: "Error", content: "Invalid selection" };
};

const result = selection.drugType ? getRecommendation() : null;

return (
    <div className="max-w-2xl mx-auto p-4 min-h-screen font-sans text-
slate-800">

    {/* Header */}
    <header className="mb-8 flex items-center justify-between">
        <div>
            <h1 className="text-2xl font-bold text-slate-900">SDCEP
Workup</h1>
            <p className="text-slate-500 text-sm">Anticoagulants &
Antiplatelets Guidance (2nd Ed.)</p>
        </div>
        {step > 0 && (
            <button onClick={reset} className="text-slate-400 hover:text-
slate-600 p-2">
                <RefreshCw size={20} />
            </button>
        ) }
    </header>

    {/* STEP 0: MEDICATION */}
    {step === 0 && (

```

```

<FadeIn>
  <div className="space-y-4">
    <h2 className="text-lg font-semibold mb-4 flex items-center gap-2">
      <span className="bg-blue-100 text-blue-600 w-8 h-8 rounded-full flex items-center justify-center text-sm">1</span>
      Select Medication Type
    </h2>
    <div className="grid gap-3">
      {DRUG_CATEGORIES.map((drug) => (
        <button
          key={drug.id}
          onClick={() => handleDrugSelect(drug)}
          className="flex items-start gap-4 p-4 bg-white border border-slate-200 rounded-xl hover:border-blue-400 hover:shadow-md transition-all text-left group"
        >
          <div className="bg-slate-50 p-3 rounded-lg group-hover:bg-blue-50 transition-colors">
            {drug.icon}
          </div>
          <div>
            <div className="font-semibold text-slate-900">{drug.label}</div>
            <div className="text-sm text-slate-500">{drug.sublabel}</div>
          </div>
          <ChevronRight className="ml-auto text-slate-300 group-hover:text-blue-400" />
        </button>
      )))
    </div>
  </div>
</FadeIn>
) }

/* STEP 1: PROCEDURE */
{step === 1 && (
  <FadeIn>
    <button onClick={() => setStep(0)} className="mb-4 text-sm text-slate-500 flex items-center hover:text-slate-800">
      <ChevronLeft size={16} /> Back
    </button>

    <h2 className="text-lg font-semibold mb-4 flex items-center gap-2">
      <span className="bg-blue-100 text-blue-600 w-8 h-8 rounded-full flex items-center justify-center text-sm">2</span>
      Select Dental Procedure
    </h2>

    <div className="space-y-6">
      /* Low Risk Group */
      <div className="space-y-2">

```

```

        <h3 className="text-sm font-bold text-green-700 uppercase tracking-wider px-1">Unlikely to bleed / Low Risk</h3>
        <div className="bg-white rounded-xl border border-slate-200 overflow-hidden divide-y divide-slate-100">
            {PROCEDURES.lowRisk.map((proc, idx) => (
                <button
                    key={idx}
                    onClick={() => handleProcedureSelect('lowRisk', proc)}>
                    {proc}
                    <ChevronRight size={16} className="opacity-0 group-hover:opacity-100 text-green-600" />
                </button>
            )))
        </div>
    </div>

    {/* Higher Risk Group */}
    <div className="space-y-2">
        <h3 className="text-sm font-bold text-amber-700 uppercase tracking-wider px-1">Likely to bleed / Higher Risk</h3>
        <div className="bg-white rounded-xl border border-slate-200 overflow-hidden divide-y divide-slate-100">
            {PROCEDURES.higherRisk.map((proc, idx) => (
                <button
                    key={idx}
                    onClick={() => handleProcedureSelect('higherRisk', proc)}>
                    {proc}
                    <ChevronRight size={16} className="opacity-0 group-hover:opacity-100 text-amber-600" />
                </button>
            )))
        </div>
    </div>
    </div>
</FadeIn>
) }

/* STEP 2: SPECIFIC CHECKS (INR or DOSE) */
{step === 2 && (
    <FadeIn>
        <button onClick={() => setStep(1)} className="mb-4 text-sm text-slate-500 flex items-center hover:text-slate-800">
            <ChevronLeft size={16} /> Back
        </button>
)
}

```

```

{selection.drugType === 'vka' && (
  <Card className="p-6">
    <h2 className="text-lg font-semibold mb-2">Latest INR
  Reading</h2>
    <p className="text-sm text-slate-500 mb-6">Please enter the
patient's most recent INR (ideally within 24h, max 72h).</p>

    <div className="space-y-4">
      <input
        type="number"
        step="0.1"
        placeholder="e.g. 2.5"
        className="w-full p-4 text-2xl border border-slate-300
rounded-lg focus:ring-2 focus:ring-blue-500 outline-none"
        onKeyDown={(e) => {
          if(e.key === 'Enter')
        handleInrSubmit(e.target.value);
        }}
        onChange={(e) => setSelection(prev => ({...prev, inr:
e.target.value}))}
      />
      <Button
        className="w-full"
        disabled={!selection.inr}
        onClick={() => handleInrSubmit(selection.inr)}
      >
        Check INR
      </Button>
    </div>
  </Card>
) }

{selection.drugType === 'injectable' && (
  <Card className="p-6">
    <h2 className="text-lg font-semibold mb-4">LMWH Dose
  Type</h2>
    <p className="text-sm text-slate-500 mb-6">Check the
patient's prescription history or discharge letter.</p>

    <div className="grid gap-4">
      <button
        onClick={() => handleLmwhSelect('prophylactic')}
        className="p-4 border border-slate-200 rounded-lg
hover:border-green-500 hover:bg-green-50 text-left"
      >
        <div className="font-bold text-slate-900">Prophylactic
      (Low) Dose</div>
        <div className="text-sm text-slate-500">e.g.,
      Enoxaparin 20-40mg OD</div>
      </button>

      <button
        onClick={() => handleLmwhSelect('treatment')}

```

```

        className="p-4 border border-slate-200 rounded-lg
        hover:border-red-500 hover:bg-red-50 text-left"
    >
        <div className="font-bold text-slate-900">Treatment
        (Therapeutic) Dose</div>
        <div className="text-sm text-slate-500">e.g.,
        Enoxaparin 1.5mg/kg OD or 1mg/kg BD</div>
        </button>
    </div>
</Card>
)
)
</FadeIn>
) }

/* STEP 3: RESULTS */
{step === 3 && result && (
<FadeIn>
<div className="space-y-6">

/* Summary Pill */
<div className="flex items-center gap-2 text-sm text-slate-500 justify-center">
    <span className="bg-slate-100 px-2 py-1 rounded">{DRUG_CATEGORIES.find(d=>d.id ===
selection.drugType)?.label}</span>
    <ChevronRight size={14} />
    <span className="bg-slate-100 px-2 py-1 rounded truncate max-w-[150px]">{selection.procedureName}</span>
    {selection.inr && <
        <ChevronRight size={14} />
        <span className="bg-slate-100 px-2 py-1 rounded">INR:
        {selection.inr}</span>
    </>}
</div>

<div className={`rounded-xl border-2 overflow-hidden shadow-lg ${

        result.color === 'green' ? 'border-green-100 bg-white' :
        result.color === 'amber' ? 'border-amber-100 bg-white' :
        'border-red-100 bg-white'
    }`}>
    /* Header Status */
    <div className={`p-6 text-center ${

        result.color === 'green' ? 'bg-green-50 text-green-800' :
        result.color === 'amber' ? 'bg-amber-50 text-amber-800' :
        'bg-red-50 text-red-800'
    }`}>
        <div className="font-black tracking-wider text-sm uppercase mb-1 opacity-70">Guidance Recommendation</div>
        <h2 className="text-2xl font-bold flex items-center justify-center gap-2">
            {result.color === 'red' && <ShieldAlert />}

```

```

        {result.status}
    </h2>
</div>

{/* Content */}
<div className="p-6">
    <h3 className="text-xl font-bold text-slate-900 mb-4">{result.title}</h3>
    <div className="text-slate-700 leading-relaxed">
        {result.content}
    </div>
</div>

{/* Standard SDCEP Footer Advice */}
<div className="bg-slate-50 p-4 border-t border-slate-100 text-xs text-slate-500 space-y-1">
    <p className="font-semibold text-slate-700 flex items-center gap-1">
        <Info size={12} /> General SDCEP Advice:
    </p>
    <p>Always treat atraumatically.</p>
    <p>Only discharge once haemostasis is confirmed.</p>
    <p>Provide written post-op instructions and emergency contacts.</p>
    <p>Avoid NSAIDs for pain relief (use Paracetamol).</p>
    </div>
</div>

<Button onClick={reset} variant="secondary" className="w-full">
    Start New Patient Workup
</Button>
</div>
</FadeIn>
) }

</div>
);
}

```