

```

import React, { useState } from 'react';
import { AlertCircle, Clock, CheckCircle, FileText, Users, DollarSign, UserPlus } from 'lucide-react';

export default function CaseManagerDashboard() {
  const [activeFilter, setActiveFilter] = useState(null);
  const [expandedCase, setExpandedCase] = useState(null);
  const [activeTab, setActiveTab] = useState('cases');

  // Sample data
  const actionCards = [
    { id: 'urgent', label: 'Urgent', count: 3, detail: 'clients waiting for INA booking', color: 'bg-red-50 border-red-200 text-red-800', icon: AlertCircle },
    { id: 'today', label: 'Today', count: 2, detail: 'INA visits scheduled', color: 'bg-yellow-50 border-yellow-200 text-yellow-800', icon: Clock },
    { id: 'ready', label: 'Ready', count: 5, detail: 'consents received - contact clients', color: 'bg-green-50 border-green-200 text-green-800', icon: CheckCircle },
    { id: 'pending', label: 'Pending', count: 2, detail: 'cases awaiting reader assignment', color: 'bg-blue-50 border-blue-200 text-blue-800', icon: FileText }
  ];

  const cases = [
    { id: 1, ref: 'INA-2024-001', client: 'Sarah Johnson', stage: 'Awaiting Consent', readers: ['Dr. Smith'], nextAction: 'Follow up on consent', days: 3, category: 'urgent' },
    { id: 2, ref: 'INA-2024-002', client: 'Michael Chen', stage: '1st Draft', readers: ['Dr. Brown', 'Dr. Lee'], nextAction: 'Review draft', days: 7, category: 'today' },
    { id: 3, ref: 'INA-2024-003', client: 'Emma Davis', stage: 'Consent Received', readers: [], nextAction: 'Contact client', days: 1, category: 'ready' },
    { id: 4, ref: 'INA-2024-004', client: 'James Wilson', stage: '2nd Draft', readers: ['Dr. Smith'], nextAction: 'Final review', days: 12, category: 'pending' },
    { id: 5, ref: 'INA-2024-005', client: 'Olivia Martinez', stage: 'Awaiting Reader', readers: [], nextAction: 'Assign readers', days: 2, category: 'pending' },
    { id: 6, ref: 'INA-2024-006', client: 'Noah Anderson', stage: 'Consent Received', readers: [], nextAction: 'Contact client', days: 1, category: 'ready' }
  ];

  const filteredCases = activeFilter
    ? cases.filter(c => c.category === activeFilter)
    : cases;

  const approvalItems = [
    { id: 1, type: 'Payment Request', detail: 'Dr. Smith - 3 cases reviewed', amount: '£450', status: 'pending' },
    { id: 2, type: 'Case Closure', detail: 'INA-2024-001 - All documents submitted', status: 'pending' }
  ];

  return (
    <div className="min-h-screen bg-gray-50">
      { /* Header */ }
      <div className="bg-white border-b border-gray-200 px-6 py-4">
        <div className="flex items-center justify-between">
          <div>
            <h1 className="text-2xl font-semibold text-gray-900">Case Manager Dashboard</h1>
            <p className="text-sm text-gray-500 mt-1">Welcome back, Liz</p>
          </div>
          <div className="text-sm text-gray-600">
            Friday, October 10, 2025
          </div>
        </div>
      </div>
    </div>
  );
}

```

```

<div className="max-w-7xl mx-auto px-6 py-6">
  {/ * ACTION CENTER */}
  <div className="mb-8">
    <h2 className="text-lg font-semibold text-gray-900 mb-4">Action Center</h2>
    <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-4">
      {actionCards.map(card => {
        const Icon = card.icon;
        return (
          <button
            key={card.id}
            onClick={() => setActiveFilter(activeFilter === card.id ? null : card.id)}
            className={` ${card.color} border-2 rounded-lg p-4 text-left transition-all
              hover:shadow-md ${
                activeFilter === card.id ? 'ring-2 ring-offset-2 ring-blue-500' : ''
              }`}
          >
            <div className="flex items-start justify-between mb-2">
              <Icon className="w-5 h-5" />
              <span className="text-2xl font-bold">{card.count}</span>
            </div>
            <div className="font-medium mb-1">{card.label}</div>
            <div className="text-xs opacity-80">{card.detail}</div>
          </button>
        );
      })}
    </div>
    {activeFilter && (
      <div className="mt-3 text-sm text-gray-600">
        Filtering by: <span className="font-medium">{actionCards.find(c => c.id ===
activeFilter)?.label}</span>
        <button onClick={() => setActiveFilter(null)} className="ml-2 text-blue-600
hover:underline">
          Clear filter
        </button>
      </div>
    )}
  </div>

  {/ * Navigation Tabs for Liz's Extra Sections */}
  <div className="mb-6 border-b border-gray-200">
    <div className="flex gap-6">
      <button
        onClick={() => setActiveTab('cases')}
        className={` pb-3 px-1 border-b-2 font-medium text-sm transition-colors ${
          activeTab === 'cases'
            ? 'border-blue-600 text-blue-600'
            : 'border-transparent text-gray-600 hover:text-gray-900'
        }`}
      >
        My Cases
      </button>
      <button
        onClick={() => setActiveTab('readers')}
        className={` pb-3 px-1 border-b-2 font-medium text-sm transition-colors ${
          activeTab === 'readers'
            ? 'border-blue-600 text-blue-600'
            : 'border-transparent text-gray-600 hover:text-gray-900'
        }`}
      >
    </div>
  </div>

```

```

    <Users className="w-4 h-4 inline mr-1" />
    Reader Management
  </button>
  <button
    onClick={() => setActiveTab('approvals')}
    className={pb-3 px-1 border-b-2 font-medium text-sm transition-colors ${
      activeTab === 'approvals'
        ? 'border-blue-600 text-blue-600'
        : 'border-transparent text-gray-600 hover:text-gray-900'
    }}
  >
    <DollarSign className="w-4 h-4 inline mr-1" />
    Approval Queue ({approvalItems.length})
  </button>
  <button
    onClick={() => setActiveTab('cm')}
    className={pb-3 px-1 border-b-2 font-medium text-sm transition-colors ${
      activeTab === 'cm'
        ? 'border-blue-600 text-blue-600'
        : 'border-transparent text-gray-600 hover:text-gray-900'
    }}
  >
    <UserPlus className="w-4 h-4 inline mr-1" />
    CM Management
  </button>
</div>
</div>

```

```

/* MY CASES Section */
{activeTab === 'cases' && (
  <div className="bg-white rounded-lg shadow">
    <div className="px-6 py-4 border-b border-gray-200">
      <h2 className="text-lg font-semibold text-gray-900">My Cases</h2>
      <p className="text-sm text-gray-500 mt-1">
        {filteredCases.length} {filteredCases.length === 1 ? 'case' : 'cases'} shown
      </p>
    </div>
    <div className="divide-y divide-gray-200">
      {filteredCases.map(casItem => (
        <div key={casItem.id} className="hover:bg-gray-50 transition-colors">
          <button
            onClick={() => setExpandedCase(expandedCase === casItem.id ? null :
casItem.id)}
            className="w-full px-6 py-4 text-left"
          >
            <div className="flex items-center justify-between">
              <div className="flex-1 grid grid-cols-6 gap-4 items-center">
                <div className="font-medium text-gray-900">{casItem.ref}</div>
                <div className="text-gray-700">{casItem.client}</div>
              </div>
                <span className="inline-flex items-center px-2.5 py-0.5 rounded-full text-xs
font-medium bg-blue-100 text-blue-800">
                  {casItem.stage}
                </span>
              </div>
              <div className="text-sm text-gray-600">
                {casItem.readers.length > 0 ? casItem.readers.join(', ') : 'No readers assigned'}
              </div>
              <div className="text-sm text-gray-700">{casItem.nextAction}</div>
              <div className="text-sm text-gray-500">{casItem.days} days</div>
            </div>
          </button>
        </div>
      )

```

```

    </div>
    <div className="ml-4 text-gray-400">
      {expandedCase === caseItem.id ? '-' : '+'}
    </div>
  </div>
</button>

{expandedCase === caseItem.id && (
  <div className="px-6 pb-4 bg-gray-50 border-t border-gray-100">
    <div className="mt-4 grid grid-cols-2 gap-6">
      <div>
        <h4 className="text-sm font-medium text-gray-900 mb-3">Timeline</h4>
        <div className="space-y-2 text-sm">
          <div className="flex items-center gap-2">
            <div className="w-2 h-2 rounded-full bg-green-500"></div>
            <span className="text-gray-600">Oct 7: Consent received</span>
          </div>
          <div className="flex items-center gap-2">
            <div className="w-2 h-2 rounded-full bg-blue-500"></div>
            <span className="text-gray-600">Oct 5: Initial contact made</span>
          </div>
          <div className="flex items-center gap-2">
            <div className="w-2 h-2 rounded-full bg-gray-300"></div>
            <span className="text-gray-600">Oct 1: Case opened</span>
          </div>
        </div>
      </div>
      <div>
        <h4 className="text-sm font-medium text-gray-900 mb-3">Quick Actions</h4>
        <div className="space-y-2">
          <button className="w-full px-3 py-2 text-sm bg-blue-600 text-white rounded
hover:bg-blue-700">
            Contact Client
          </button>
          <button className="w-full px-3 py-2 text-sm border border-gray-300 rounded
hover:bg-gray-50">
            Assign Reader
          </button>
          <button className="w-full px-3 py-2 text-sm border border-gray-300 rounded
hover:bg-gray-50">
            View Documents
          </button>
        </div>
      </div>
    </div>
  )}
</div>
))}
</div>
</div>
)}

{/* READER MANAGEMENT Section */}
{activeTab === 'readers' && (
  <div className="bg-white rounded-lg shadow p-6">
    <div className="flex items-center justify-between mb-6">
      <h2 className="text-lg font-semibold text-gray-900">Reader Management</h2>
      <button className="px-4 py-2 bg-blue-600 text-white rounded hover:bg-blue-700 text-
sm">

```

```

        Register New Reader
    </button>
</div>
<div className="space-y-4">
    {[ 'Dr. Smith', 'Dr. Brown', 'Dr. Lee' ].map((reader, idx) => (
        <div key={idx} className="border border-gray-200 rounded-lg p-4 flex items-center
justify-between">
            <div>
                <div className="font-medium text-gray-900">{reader}</div>
                <div className="text-sm text-gray-500">Available • 12 cases completed • Last
payment: Sep 2025</div>
            </div>
            <div className="flex gap-2">
                <button className="px-3 py-1 text-sm border border-gray-300 rounded hover:bg-
gray-50">
                    View Details
                </button>
                <button className="px-3 py-1 text-sm border border-gray-300 rounded hover:bg-
gray-50">
                    Process Payment
                </button>
            </div>
        </div>
    )]}
</div>
</div>
))

{/* APPROVAL QUEUE Section */}
{activeTab === 'approvals' && (
    <div className="bg-white rounded-lg shadow p-6">
        <h2 className="text-lg font-semibold text-gray-900 mb-6">Approval Queue</h2>
        <div className="space-y-4">
            {approvalItems.map(item => (
                <div key={item.id} className="border border-yellow-200 bg-yellow-50 rounded-lg
p-4">
                    <div className="flex items-start justify-between mb-3">
                        <div>
                            <div className="font-medium text-gray-900">{item.type}</div>
                            <div className="text-sm text-gray-600 mt-1">{item.detail}</div>
                            {item.amount && <div className="text-lg font-semibold text-gray-900
mt-2">{item.amount}</div>}
                        </div>
                        <div>
                            <div className="flex gap-2">
                                <button className="px-4 py-2 bg-green-600 text-white rounded hover:bg-
green-700 text-sm">
                                    Approve
                                </button>
                                <button className="px-4 py-2 border border-gray-300 rounded hover:bg-gray-50
text-sm">
                                    Review Details
                                </button>
                                <button className="px-4 py-2 border border-red-300 text-red-700 rounded
hover:bg-red-50 text-sm">
                                    Reject
                                </button>
                            </div>
                        </div>
                    </div>
                </div>
            )]}
        </div>
    )}
)

```

```

    </div>
  </div>
})

{/* CM MANAGEMENT Section */}
{activeTab === 'cm' && (
  <div className="bg-white rounded-lg shadow p-6">
    <div className="flex items-center justify-between mb-6">
      <h2 className="text-lg font-semibold text-gray-900">Case Manager Management</
h2>
      <button className="px-4 py-2 bg-blue-600 text-white rounded hover:bg-blue-700 text-
sm">
        Onboard New CM
      </button>
    </div>
    <div className="space-y-4">
      {[ 'Emma Thompson', 'David Park', 'Rachel Green' ].map((cm, idx) => (
        <div key={idx} className="border border-gray-200 rounded-lg p-4 flex items-center
justify-between">
          <div>
            <div className="font-medium text-gray-900">{cm}</div>
            <div className="text-sm text-gray-500">Active • {8 + idx * 2} cases assigned •
Joined Mar 2024</div>
          </div>
          <div className="flex gap-2">
            <button className="px-3 py-1 text-sm border border-gray-300 rounded hover:bg-
gray-50">
              View Workload
            </button>
            <button className="px-3 py-1 text-sm border border-gray-300 rounded hover:bg-
gray-50">
              Reassign Cases
            </button>
          </div>
        </div>
      )]}
    </div>
  </div>
)}
);
}

```