Quotes and Adjudications System Implementation

☑ COMPLETED - Backend Implementation

Database Tables Created

1. **vendor_quotes** - Stores uploaded quote PDFs (max 3 per requisition)
    - Fields: requisition_id, vendor_id, vendor_name, quote_number, quote_amount, currency, file path, uploader, notes
2. **adjudications** - Stores procurement's comparison summary
    - Fields: requisition_id, recommended_vendor, amounts, summary, evaluation criteria, technical compliance, pricing analysis, delivery/payment terms, rationale
3. **requisitions table** - Added flags:
    - `has_quotes` - Boolean flag
    - `has_adjudication` - Boolean flag

Backend API Endpoints (All Functional)

Quote Management:

- `POST /api/requisitions/:id/quotes` - Upload vendor quote PDF (max 3)
- `GET /api/requisitions/:id/quotes` - Get all quotes for a requisition
- `GET /api/quotes/:id/download` - Download quote PDF
- `DELETE /api/quotes/:id` - Delete a quote

Adjudication Management:

- `POST /api/requisitions/:id/adjudication` - Create adjudication summary
- `GET /api/requisitions/:id/adjudication` - Get adjudication for a requisition

Access Control:

- Procurement: Can upload quotes, create adjudications
- Finance & MD: Can view quotes and adjudications
- Admin: Full access

File Upload Configuration

- Uses multer for PDF uploads
- Maximum 3 quotes per requisition
- 10MB file size limit
- Stored in: `backend/uploads/quotes/`
- Only PDF files accepted

🔨 TODO - Frontend Implementation

1. Add Frontend API Functions (`frontend/app.js`)

Add to the `api` object around line 200-300:

```
// Quotes API
uploadQuote: async (requisitionId, formData) => {
  const res = await fetch(`${API_URL}/requisitions/${requisitionId}/quotes`, {
    method: 'POST',
    headers: { 'Authorization': `Bearer ${getToken()}` },
    body: formData // FormData object with file
  });
  if (!res.ok) throw new Error('Failed to upload quote');
  return res.json();
},

getQuotes: async (requisitionId) => {
  const res = await fetch(`${API_URL}/requisitions/${requisitionId}/quotes`, {
    headers: getHeaders()
  });
  if (!res.ok) throw new Error('Failed to fetch quotes');
  return res.json();
},

downloadQuote: (quoteId) => {
  window.open(`${API_URL}/quotes/${quoteId}/download`, '_blank');
},

deleteQuote: async (quoteId) => {
  const res = await fetch(`${API_URL}/quotes/${quoteId}`, {
    method: 'DELETE',
    headers: getHeaders()
  });
  if (!res.ok) throw new Error('Failed to delete quote');
  return res.json();
},

// Adjudication API
createAdjudication: async (requisitionId, data) => {
  const res = await fetch(`${API_URL}/requisitions/${requisitionId}/adjudication`, {
    method: 'POST',
    headers: getHeaders(),
    body: JSON.stringify(data)
  });
  if (!res.ok) throw new Error('Failed to create adjudication');
  return res.json();
},

getAdjudication: async (requisitionId) => {
  const res = await fetch(`${API_URL}/requisitions/${requisitionId}/adjudication`, {
    headers: getHeaders()
  });
  if (!res.ok) throw new Error('Failed to fetch adjudication');
  return res.json();
}
```

2. Create Quote Upload Component

Create a new component for procurement to upload up to 3 vendor quotes:

Component: QuoteUploadSection

- Shows list of uploaded quotes (max 3)
- Upload form for each quote:
  - Vendor selection dropdown
  - Quote number (optional)
  - Quote amount
  - Currency
  - PDF file upload
  - Notes (optional)
- Download/Delete buttons for each uploaded quote
- Progress indicator (e.g., "2 of 3 quotes uploaded")

3. Create Adjudication Summary Form

Create a comprehensive adjudication form component:

Component: AdjudicationForm

- Only shows after all 3 quotes are uploaded
- Form fields:
  - Recommended Vendor (dropdown from uploaded quotes)
  - Recommended Amount
  - Currency
  - **Executive Summary** (textarea, required)
  - **Evaluation Criteria** (textarea)
  - **Technical Compliance** (textarea)
  - **Pricing Analysis** (textarea) - compare 3 quotes
  - **Delivery Terms** (textarea)
  - **Payment Terms** (textarea)

    - **Recommendation Rationale** (textarea, required)

- Submit button creates adjudication

4. Add Menu Items to Sidebar

Update sidebar menu items around line 1400-1450:

```
// Add to procurement/admin menu
{
  id: 'rfq-management',
  label: 'RFQ & Quotes',
  icon: '📝',
  show: user.role === 'procurement' || user.role === 'admin',
  isGroup: true,
  children: [
    { id: 'upload-quotes', label: 'Upload Quotes', icon: '📤', show: true },
    { id: 'adjudications', label: 'Adjudications', icon: '⚖️', show: true }
  ]
}
```

5. Integrate with Requisition Review Flow

For Procurement:

- After HOD/other approvals, show "Upload Quotes" button on requisition
- After 3 quotes uploaded, show "Create Adjudication" button
- Show status badges: "Quotes Pending", "Quotes Uploaded", "Adjudication Complete"

For Finance & MD:

- Show "View Quotes" button on requisitions with quotes
- Show "View Adjudication" button on requisitions with adjudications
- Display quote comparison table
- Allow downloading all quote PDFs
- Show adjudication summary before making approval decision

6. Create View Components

QuotesViewer Component (for Finance/MD):

- Table showing all 3 quotes side-by-side
- Columns: Vendor, Quote #, Amount, Currency, Upload Date
- Download button for each PDF
- Highlight recommended vendor

AdjudicationViewer Component (for Finance/MD):

- Display all adjudication fields in organized sections
- Show recommended vendor prominently
- Show comparison summary
- Comments section for Finance/MD feedback

Implementation Priority

1. **HIGH**: Add API functions to frontend
2. **HIGH**: Create QuoteUploadSection component
3. **HIGH**: Create AdjudicationForm component
4. **MEDIUM**: Add sidebar menu items
5. **MEDIUM**: Integrate with requisition approval workflow
6. **LOW**: Create viewer components for Finance/MD

Usage Flow

1. Procurement receives responses to RFQ from 3 vendors
2. Procurement uploads each vendor's quote PDF (with amount, vendor info)
3. System enforces maximum 3 quotes per requisition
4. After 3 quotes uploaded, procurement creates adjudication summary
5. Adjudication includes comparison and recommendation
6. Finance Manager reviews quotes + adjudication before approval
7. MD reviews quotes + adjudication before final approval
8. Both can download quote PDFs for offline review

Security & Validation

☑ Role-based access control implemented
☑ PDF-only file validation
☑ File size limits (10MB)
☑ Maximum 3 quotes enforced
☑ Required fields validated
☑ Adjudication requires quotes first

Notes

- Quote PDFs stored in: `backend/uploads/quotes/`
- Quotes attached to requisition ID, not deletable after adjudication submitted
- Finance and MD cannot upload quotes, only view

- Adjudication can be created only once per requisition
- System tracks who uploaded each quote and when