We're building a banking channel app for corporate clients. Based on the user stories, companies need to manage their finances and handle employee payroll through our system. Think of it as their main interface for business banking operations.

**Core functionality from user stories:**

* Account balance management and transaction history
* Payroll payment creation (single and bulk)
* Multi-level approval workflows
* Reporting and statement generation

**Feature Requirements**

From what I read in the stories, these are the must-haves:

**Authentication & Security**

* User login with role-based access
* Account lockout after failed attempts
* Session management

**Transaction Processing**

* Manual transaction entry forms
* Bulk Excel upload functionality
* Transaction editing and management
* Print/export capabilities

**Approval System**

* Hierarchical approval based on user roles
* Transaction queues for approvers
* Approve/reject/request changes functionality

**Account Management**

* Real-time balance checking
* Transaction history views
* Report generation

**User Role Structure**

Based on the user stories, I'm seeing this hierarchy:

Super Admin - Full system access, can approve anything

│

├── Department Head - Department-level approvals

│ │

│ ├── Manager - Can create transactions, approve subordinates

│ │ │

│ │ └── Employee - Can create transactions only

│ │

│ └── Viewer - Read-only access

**Transaction Workflow**

From the user stories, here's how transactions should flow:

**Creation Process:**

1. User logs in, system checks role
2. Options presented based on permissions:
   * Single transaction form
   * Bulk Excel upload
   * Template download
3. Data validation against business rules
4. Save as draft or submit for approval

**Approval Process:**

1. Transaction enters approval queue
2. Appropriate approver receives notification
3. Approver reviews transaction details
4. Decision: Approve/Reject/Request Changes
5. Transaction status updated
6. Process continues or completes

**Business Rules (from user stories)**

**Approval Restrictions:**

* Users cannot approve their own transactions
* Only higher-level roles can approve transactions
* Different approval levels based on transaction amount

**Amount Thresholds:**

* Under $1,000: Manager approval
* $1,000-$10,000: Department Head approval
* Over $10,000: Super Admin approval

**Audit Requirements:**

* All actions must be logged
* Track user ID, timestamp, and changes made
* Maintain before/after values for modifications

**Data Requirements**

**User Management:**

* User profiles with role assignments
* Role hierarchy definitions
* Access control matrices

**Transaction Data:**

* Individual transaction records
* Batch grouping for bulk uploads
* Status tracking through workflow
* Approval history and comments

**Audit Trail:**

* Complete change history
* User action logs
* System event tracking

**Technical Considerations**

**Database Design:**

* Users table with role relationships
* Transactions table with status fields
* Approval workflow tracking
* Comprehensive audit logging

**File Processing:**

* Excel template generation
* Bulk upload validation
* Error handling and reporting

**Security:**

* Role-based access control
* Data encryption for sensitive information
* Secure session management

**Integration Points**

From the user stories, we need to consider:

* Employee database integration for validation
* Banking system integration for processing
* Notification system for approvals
* Reporting system for statements