1. **Validation Layer** (**course.validation.js**):
   * Validation schemas for all API endpoints
   * Input validation for create, update, get, delete, and query operations
   * Uses Joi for schema validation
   * Includes validation for query parameters and pagination
2. **Service Layer** (**course.service.js**):
   * Business logic implementation
   * CRUD operations for courses
   * Title uniqueness checking
   * Multi-tenant support
   * Pagination and filtering support
   * Error handling with custom ApiError
3. **Controller Layer** (**course.controller.js**):
   * Request handling
   * Response formatting
   * Uses catchAsync for error handling
   * Integrates with service layer
   * Handles user context and permissions
4. **Routes** (**course.routes.js**):
   * API endpoint definitions
   * Authentication middleware
   * Validation middleware
   * Role-based access control

The API endpoints are:

Code

CopyInsert

POST /api/courses - Create new course

GET /api/courses - List/search courses

GET /api/courses/:courseId - Get course details

PATCH /api/courses/:courseId - Update course

DELETE /api/courses/:courseId - Delete course

Features implemented:

* Complete CRUD operations
* Input validation
* Authentication & Authorization
* Multi-tenant support
* Pagination and filtering
* Error handling
* Data consistency checks