 created the following structure:

1. **Learning Management Integration**:
   * Created **learning-management.service.js** for integration with LMS
   * Handles program, calendar, and activity synchronization
   * Includes error handling and logging
2. **Academic Program API**:
   * Created routes in **academic-program.routes.js**
   * Created controller in **academic-program.controller.js**
   * Includes CRUD operations and statistics
   * Proper authentication and authorization
3. **Academic Calendar API**:
   * Created routes in **academic-calendar.routes.js**
   * Created controller in **academic-calendar.controller.js**
   * Includes calendar management and LMS synchronization
   * Event and term management

Key Features:

1. **Role-based Access Control**:
   * Admin and academic\_admin for full access
   * program\_manager for limited management
   * instructor and student for read access
2. **Integration with LMS**:
   * Program synchronization
   * Calendar event synchronization
   * Learning activity fetching
   * Term schedule management
3. **Audit Logging**:
   * All important operations are logged
   * User tracking for accountability
   * Detailed audit trail