Here's a structured 10-phase development plan for your course management system:

**Phase 1: Database & Model Setup**

* Create migrations/models for:
  + Course Types (Long/Short/Industrial)
  + Programs (with course\_type relationship)
  + Sessions (for Long Courses)
  + Batches (for Short/Industrial)
  + Courses (with polymorphic session/batch relationships)
  + Student Profiles (linked to Users)
  + Registrations/Applications/Enrollments

**Phase 2: User Role System Enhancement**

* Extend Laravel Breeze with:
  + Student (Pending/Active) and Admin roles
  + Middleware for role-based access control
  + Student registration form template

**Phase 3: Admin Course Management**

* Build CRUD interfaces for:
  + Programs under each course type
  + Session/Batch management
  + Course creation with manual IDs
  + Online/Offline toggle system

**Phase 4: Student Registration System**

* Develop:
  + Student profile creation workflow
  + Admin approval/rejection system
  + Status transition logic (Pending → Active)
  + Notification system for status updates

**Phase 5: Course Application Engine**

* Implement:
  + Course selection interface (session/batch logic)
  + Application submission system
  + Pending→Approved→Enrolled state machine
  + Enrollment record generation

**Phase 6: Student Portal Development**

* Create:
  + Dashboard with enrolled courses
  + Course application interface
  + Payment status tracking
  + Course material access system

**Phase 7: Admin Approval Workflows**

* Build:
  + Student registration approval panel
  + Course application review system
  + Bulk enrollment processing
  + Rejection reason tracking

**Phase 8: Guest Course Browsing**

* Develop:
  + Public course catalog
  + Program/course filtering system
  + Session/batch availability display
  + Registration CTAs

**Phase 9: Notification System**

* Implement:
  + Email/SMS alerts for status changes
  + Enrollment confirmation notices
  + Payment reminders
  + Batch/session start notifications

**Phase 10: Final Polish & Deployment**

* Conduct:
  + Cross-browser responsiveness testing
  + User role validation checks
  + Data integrity verification
  + Performance optimization
  + Production deployment