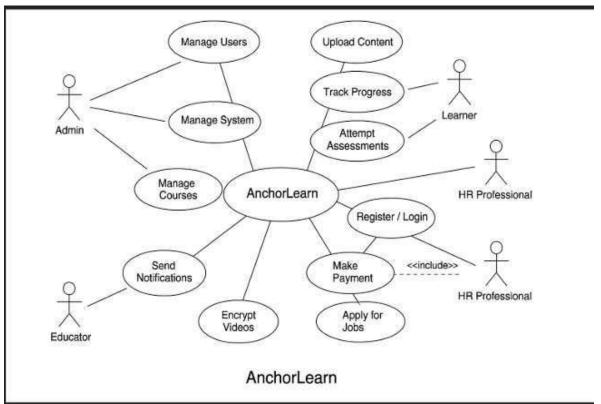
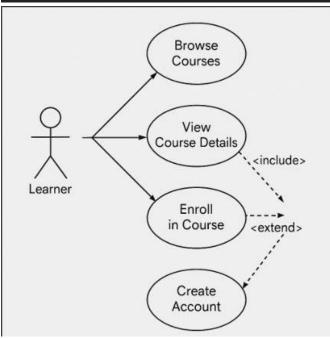
Fig 1.1: USE CASE DIAGRAM





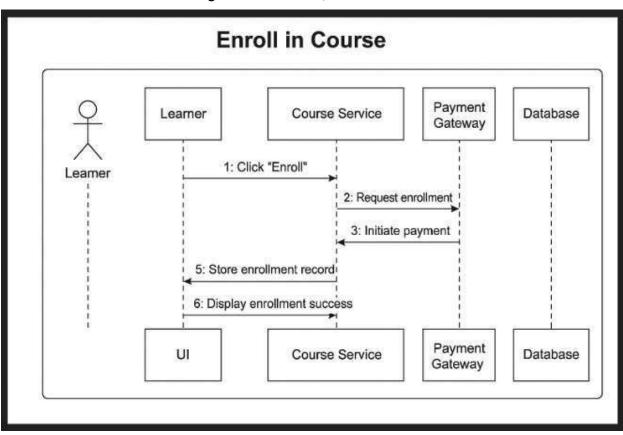


Fig 1.2: SYSTEM SEQUENCE DIAGRAM

**Enroll in Course** 1: enrollInCourse(course) controller: CourseController learner: Learner 1.1 1.2: enrollSuccess() anroll(couurse) 1.1: makePayment(learner, course) courseService paymentService: PaymentService 1.1.2: addEnrollIment (learner, course) 1.1.1 [paymentSuccess]courseCatalog: CourseCatalog

Fig 1.3: COMMUNICATION DIAGRAM (USING GRASP)

Learner Course Enrollment learnerld: String courseld: String enrollmentld: String enrolls in name: String title: Learner learner: Learner eourse: Course email: String course: Course handles enroll(course: Course) getEnrollmentInfo() getDetails():String handles saves payment CourseService **EnrollmentRepository PaymentService** enrollLearner processPayment save(enrollment: (learner: Leamer, (learner: Learner, Enrollment): void eourse: Course) course: Course) findByLearner (learnerId: String): Enrollment Boolean List<Enrollment>

Fig 1.4: CLASS DIAGRAM

Fig 1.5: Fully Dressed USE-CASE DIAGRAM

Use Case Name	Enroll in Course
Date Created	May 15, 2025
Primary Actor	Learner
Stakeholders and Interests	Educator, Admin     Learner must learn newe skills and earn certification: Enant students enrolled in their course.     Admin wants ensure course quality and valid transactions.
Preconditions	<ul> <li>Learner must be logged in.</li> <li>Course must be publisthed and approved by admin.</li> </ul>
Postconditions	<ul> <li>Learner is enrolled in the course.</li> <li>Payment (if required) is completied).</li> <li>Course progress tracking is initializazed.</li> </ul>
Success Guarantee	Learner successfully joins a course and sees it in ther dashboard.
Minimal Guarantee	Learner remains logged in; no enrolm happens if payment fails or course is full
Trigger	Learner selects a course from the catalog
Main Success Scenario	Learner logs into the platform.     Learner navigates to course catalog

Fig 1.6: MOCKUP

