Project/ scenario name:	Universal Identification for Student in University using RFID
Custom scenario description:	The object of any RFID system is to carry data in suitable transponders, generally known as tags, and to retrieve data, by machine-readable means, at a suitable time and place to satisfy particular application needs. Data within a tag may provide identification for an item in manufacture, goods in transit, a location, the identity of a vehicle, an animal or individual. By including additional data the prospect is provided for supporting applications through item specific information or instructions immediately available on reading the tag. When the student enters the classroom the card reader would be placed in front of the teacher and the student has to swipe the card and his In-Time would be marked, and when the student leaves the class his Out-Time would be marked. The students whose both In-Time and Out-Time is marked will only be considered as present. In this project we implement Automated Attendance Marking, Unit Test Result Generation, Centralized Library Automation, Automated Computer Labs, SMS Module , Report Generation. An Auto Generated mail would be sent to the principal, teachers, parents, students of school regarding the new or events, attendance, mark sheet, yearly magazine. Every student, staff, alumni would be given a username, password and they can check information related to them. For e.g. parents would be able to see the detailed report of his child attendance, mark sheet etc.
Objective/ vision:	The aim of the project is to detect which student is entering and leaving the college. At the end of the month based on the attendance. This scenario though mentioned Universal Identification for Student in University, is basically Identification system for all the students/employees of the university through RFID, hence the term wherever used for employee refers to university staff and similarly student applications can be conceived within the ambit of university.
Users of the system:	1.Administrator 2.Student 3.Professor /Employee 4.Parent
Functional requirements (include at least 8):	i. Secure registration of employees in the office(including professors, administration people) ii. Search employees based on different criteria iii. Setup the basic details of employees in organization such as working hours, number of working days per month to calculate the salaries iv. Set the hierarchy of the organization v. Send automatic mails to employees regarding salaries, attendance, holidays etc vi. Send messages to mobiles if employee absent or issued salaries or general messages regarding holidays wishes etc. vii. Issue of salaries viii. Employee must be able to view and update his details, view his attendance reports ix. Visitor must be able to view the organization basic details
Non-functional requirements (include at least 4):	i. Secure access of confidential data (user details). SSL can be used. ii. 24 X 7 availability iii. Better component design to get better performance at peak time iv. Flexible service based architecture will be highly desirable for future extension
Optional features:	student chat
User interface priorities:	A. Professional look and feel B. Use of AJAX atleast with all registration forms C. Browser testing and support for IE, NN, Mozila, and Firefox.

	D. Use of Graphical tool like JASPER to show strategic data to admin E. Reports exportable in .XLS, .PDF or any other desirable format
Reports:	A. Admin must be able to take student attendance reports in excel sheets B. Admin must be able to take the marks given in excel sheets C.Student must be able to take is attendance reports.
Other important issues:	A. Everything must be customizable by the admin
Technologies to be used	J2EE, DB2, WAS, WS-MQ, TSM
Tools to be used:	ROSE/RSA / WebSphere Modeler Eclipse/ RAD / Lotus Forms Designer / Portlet Factory WebSphere Portal/ WAS/ WAS CE / WPS DB2 Express or DB2 UDB Tivoli CDP/TSM / Tivoli Directory Server
Final deliverable must include:	1. Online or offline help to above said users, Application development executive and developer 2. Application archive (.war/.ear) with source code 3. Database backup and DDL Script 4. Complete Source code