1. **Technologies and Why**

Languages: Java, xml

IDE: Netbeans 8.0.2

Web Service Server: Java Web Services deployed on GlassFish

Client Side Application: Java Web Applications

Messaging: JMS with GlassFish is JMS provider

The reasons we choose above technologies are varies. Firstly GlasshFish and Netbean are both free and go together. As AUT students we are familiar with them. Secondly, Netbean can provide part of generated xsd file for our objects so we do not have to 100% manually code wdsl and xsd files for SOAP web services. Finally ,with respect to JMS service, it is easy to configure in GlassFish . We can create Queue and Topic without having to create Maven Dependencies and downloading other Jar files from other JMS providers. It make the project less heavy.

1. **System Design**
   1. **Overall Design**

The system present an Announcement System for Students. We have 1 Server (StudentServer which provide SOAP Web Services to 2 Applications . 2 Applications are StudentClient and StudentApp which consume the Web Services functions to operate.

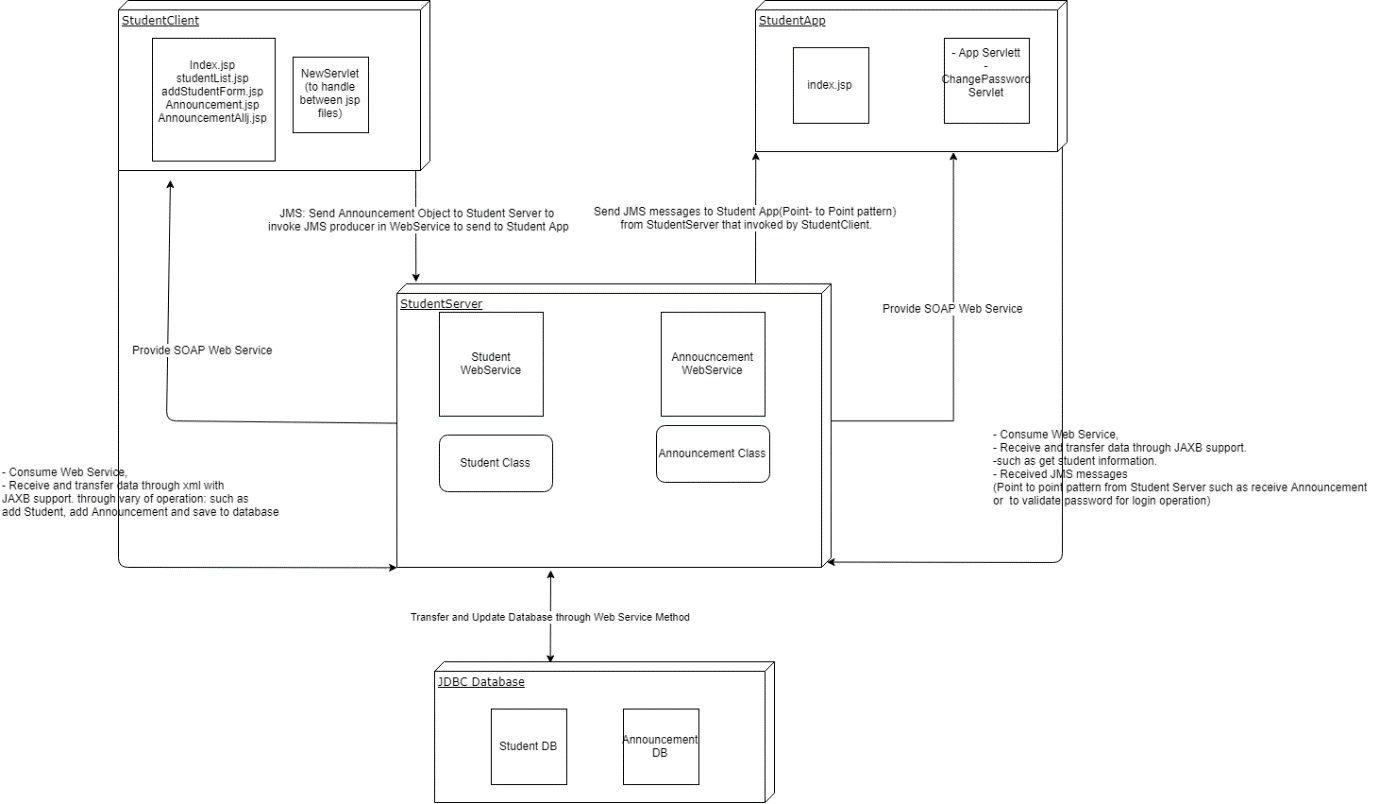
Objects are serialised passed between Sever and Clients using SOAP with JAXB support.

StudentClient act like an Admin System for Student. It can add Student, view Student, and provide Announcement to Students. When StudentClient provide Announcement, it invoke Web Service method to produce JMS message that send Announcement object to StudentClient.

StudentClient provide login page for Student, receive JMS message from server to password validation and receive JMS Announcement object from WebServer

JMS messages between StudentServer and StudentApp are Point-to-Point message pattern. The messages are sent to the queue and can be received by specific client.

* 1. **System components and used (please zoom in if you do not see the picture clear)**



1. **User Instruction**
   1. Download and Configuration
   2. Instruction