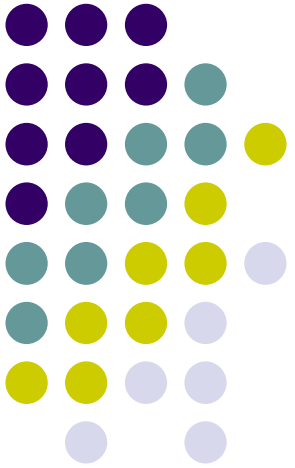
2G1523
Programming Web Services
Project

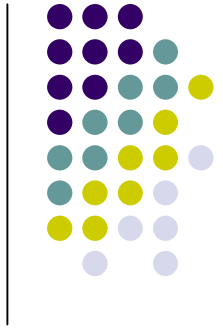


Course leader:
Professor Mihhail Matskin
misha@imit.kth.se

Teaching assistant:
Hamid Reza Mizani
hrmizani@imit.kth.se

KTH/ICT/ECS , VT07

Project - Deadline

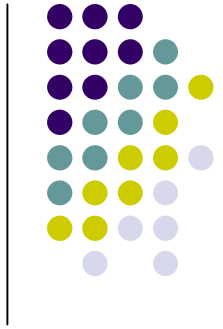


- Due date: 2007-03-11

- ***6*** bonus points (If approved before due date!)

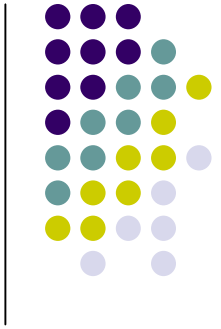
- Deliverable: Report (See course web)

Project - Aims



- Aims:
  - To apply knowledge obtained during the course to design and implementation of web services
  - To learn how to control the transactional interaction among Web services.
  - To learn how to compose Web services.

Project - Task

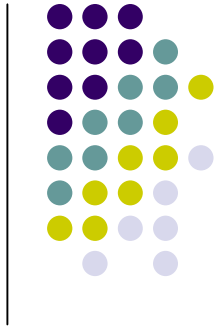


- Design and implement flight ticket reservation services (2 parts)

or

- Your own project proposal (***<u>If approves before 9/2/2006</u>***)

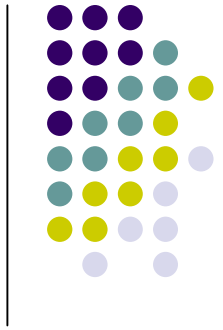
Project – Part 1



Implement the following services:

- Authorization of customers. Only the user with the correct username and password can have access to other services. This service should accept customer's Id and Password, check them with the list of registered customers and if found then give a reply that allows system access.

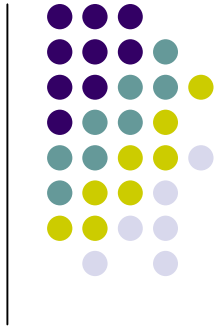
Project – Part 1



- Checking possible itinerary for flights given a departure city and a destination city. In case there is no direct flight from the departure to the destination, the service should find a route that combine several flights.

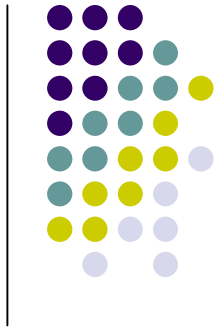
- Checking availability of tickets and finding their price for a given itinerary and given date.

Project – Part 1



- Output the price of available itineraries.
- Booking tickets for requested itinerary. Credit card number is required to book the tickets.
- Issue tickets. The ticket has to be booked before issuing.

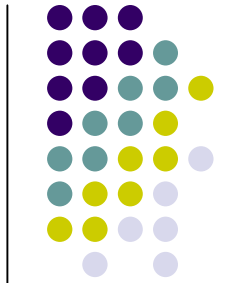
Project – Part 1



- The above-mentioned services must be described in WSDL, implemented in Java and deployed in the Axis.

- Construct both incoming and outgoing SOAP messages for invoking/replying all implemented services.

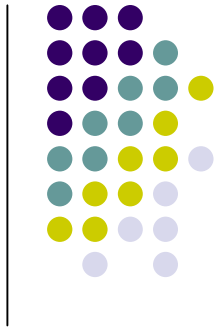
Project – Part 2



Implement the following functionalities:

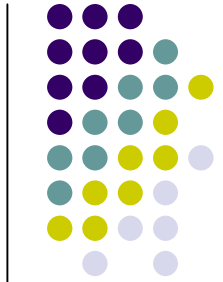
- Using the services implemented on the previous step implement the following composite service in BPEL4WS:
  - The composite service takes the username, password, departure, destination and credit card number as input, and issues the cheapest ticket. If no ticket for the require route available, the service outputs an error message.

Project – Part 2



- You have to consider the transaction control in you reservation services. For example, checking ticket availability and booking ticket is a transaction. Use WS-Coordination and WS-Transaction.
- You have to make at least one of the above services to be state full (for example Checking availability service - however, you can choose any other service). Use WS-Resources.
- Construct both incoming and outgoing SOAP messages for invoking/replying all implemented services.

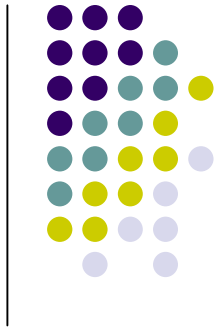
Project - Hints



- You implement functionality of all required services (***except for the composite service***) in Java.

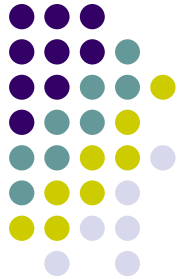
- You can use a database or simply a text file to store the flight tickets information.

Project – Report



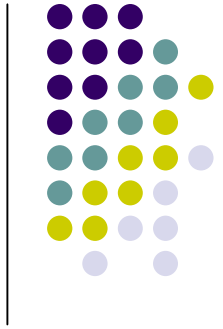
- Description of the problem and design solutions
- WSDL descriptions of all implemented services
- Java source code (or BPEL text) for all implemented services
- Protocols of services deployment

Project – Report



- Texts of all constructed incoming and outgoing SOAP messages for invoking/replying all implemented services.
- Explanation of your solution for implementing transactional control and state full resources in the project.
- Analysis of advantages and drawbacks of your solution, any remarks about implementing and usage of Web services.

Project - Delivery



- Send your report by e-mail to both misha@imit.kth.se and hrmizani@imit.kth.se.
- Deadline: 2007-03-11
- See course web for more information.

GOOD LUCK!