Lab assignment 2: Application architecture

# Exercise 1

Design the architecture with all the necessary classes for a webshop with the following requirements:

The webshop contains products that we can add or remove from a shoppingcart We can also add multiple copies of the same product in the shoppingcart We can place an order based on our shoppingcart.

Orders can only be paid by creditcard

For every order we need to know

* The shipping address and the billing address
* The customer
* The creditcard information
* The shipping method

The webshop administrator should be able to add, remove and update products.

For every product we need to know the following information:

* ProductNumber, name , price, description.
* Reviews from customers that have purchased this product.
* Stock information: number in stock, and location code in the warehouse  The supplier for this product

Customers can create an account, so they can login when they want to place on order  For every customer we want to know the following information:

* CustomerNumber, first name, last name, email, phone and address information
* Creditcards from the customer
* Orders that the customer has placed

When an order is placed, the webshop should send an email to the customer.

The webshop administrator should be able to add, remove and update suppliers.

Draw a class diagram showing the different layers and their corresponding classes and relationships.

Draw a sequence diagram showing diagram showing how we place an order.

# Exercise 2

Describe the different places where we can store state. Give the advantages and disadvantages of these different options.

* We can store data in Relational Databases (MySQL, Oracle DB) or Non-Relational Databases (Eg: Mongo DB, Graph DB)

RDBMS

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Relational Schema | Object/Relational Impedance Mismatch |
| ACID properties | Schema evolution creates problems |
| Scalable Reads | Non-scalable writes |

NoSQL

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Schema Free | Newer technology |
| Scalable reads and writes | Data redundancy |
| High availability with clustering | Eventual consistency |

# Exercise 3

For each of the following integration possibilities, describe its advantages, its disadvantages and when you would apply it:

1. RMI

Advantages:

* Simplicity: Method call looks like local even though they are remote to another JVM
* Two applications running on different servers can communicate

Disadvantages:

* Overhead of object serialization
* Synchronous

When to use: Want to have distributed Java applications without using Http

1. Messaging (JMS)

Advantages:

* Asynchronous
* Can to broadcasting

Disadvantages:

When to use: Communicate between two or many (publisher-subscriber) applications in asynchronous fashion.

1. SOAP

Advantages:

* Loosely coupled applications

Disadvantages:

* Has to use XML
* SOAP standards are complex and steeper learning curve compared to REST

When to use: Want to make use of SOAP features of security, transactions

1. REST

Advantages:

* Loosely coupled applications
* Simpler in implementation compared to SOAP
* Can be made asynchronous
* Can use JSON instead of XML making it lightweight and simpler

Disadvantages:

* Not many strict standards

When to use: To communicate different applications via Http

1. Serialized objects over HTTP

Advantages:

* No conversion between Object and XML or JSON required after and/or before application, making it more efficient

Disadvantages:

* Both applications need to use same technology so that they can use serialized objects

When to use: When both applications are implemented using Java language

1. Database integration

Advantages:

* Can see the physical records of integration in the database
* Asynchronous

Disadvantages:

* Tighter coupling compared to other integrations
* Database properties needs to satisfy

When to use: If we cannot use other integration options

1. File based integration

Advantages:

* Asynchronous
* Easier to create files
* Allows data to be passed between different databases, or operating systems, without necessarily connecting directly to them

Disadvantages:

* Both applications need to know the exact structure of data in the files to parse the data and understand it properly
* Difficult to sync data in real-time.

When to use: Communication between legacy systems since not all software systems have available APIs for integration.

# Exercise 4

Read the following article:

http://www.drdobbs.com/errant-architectures/184414966

Explain the advantages and disadvantages of distributed systems.

* Advantages:

1. Can scale up the application by having multiple copies of certain part of application if required.
2. Distributed parts are independent of one another so change in one distributed part won’t affect another.
3. Different technologies can be use at different distributed components as required since they are independent.
4. Redundancy and resiliency.
5. Distributed systems implemented using clustering improves speed and benefits of parallel processing can be obtained.

* Disadvantages:

1. Since different components run in different process in different machines, latency increases when communication between the components happens.
2. The interface to communicate between the distributed parts of system needs to be coarse-grained to minimize the number of calls between them as they are slower. Having a course-grained interface is less flexible and extendable.
3. Problems can arise if the parts are stateful when clustering.
4. It’s more challenging to properly implement distributed systems.