

CPSC-5200 Software Arch. and Design Final

Please answer the following 10 questions and **return your exam to me by 1200 PT on Tuesday 17 March**. This exam is worth 30 percent of your quarter grade. That is, each question is worth 3 points of your total 100. And yes, one of these (4) entails more work on your part, but shouldn't be considered more difficult than the rest.

Remember, in many cases there is no necessarily correct or incorrect answer. Instead I'm looking to see how you think about the problem and approach it. The questions are taken from the course and lecture materials. If we talked about it in class, it was assigned reading, or was in one of the slide decks it's fair game as a question.

The questions

1. Why do we introduce a *knowledge level* separate from an *operational level* when doing system analysis and design?
2. Explain, in your own words, the second system effect as stated by Brooks. How can you avoid problems associated with the second system effect?
3. Describe the possible customers of a domain specific language. When would we want to define and implement a DSL? Give a few examples of a DSL.
4. Assume we have the following API which lets a developer issue SQL INSERT and UPDATE statements in such a way that their execution is delayed until the developer calls `Commit()`. No statements are issued until `Commit()` is executed, and the execution order is guaranteed.

```
public enum BinaryOperation
{
    Equals,
    NotEquals,
    GreaterThan,
    GreaterThanOrEquals,
    LessThan,
    LessThanOrEquals,
    In,
    Like,
    IsNull
}

public interface IPredicate
{
    public string ColumnName { get; }

    public BinaryOperation Operation { get; }

    public object Value { get; }
}
```

```

public interface IQueryManager
{
    // issues a SQL INSERT operation for tableName setting the provided
    // column values
    //
    // an IDictionary<string, object> is an interface that maps string keys
    // onto object values, for java developers think Map<String, Object> and
    // for javascript/typescript developers think 'hash'
    IQuery Insert(string tableName,
                  IDictionary<string, object> columnValues);

    // issues a SQL UPDATE operation for tableName setting the provided column
    // values and creating a WHERE clause from the predicates
    //
    // an IList<IPredicate> is an order list of predicate objects.
    //
    // java developers can think of IList much like ArrayList if that helps
    IQuery Update(string tableName,
                  IDictionary<string, object> columnValues,
                  IList<IPredicate> predicates);

    // issues a SQL DELETE operation for tableName with a WHERE clause
    // from the predicates
    IQuery Delete(string tableName,
                  IList<IPredicate> predicates);

    // causes the requested database operations to execute (returns true if at
    // least one row is effected), check the individual IQuery returns
    // from the above methods to determine the actual impact of each
    bool Commit();

    // abandons the requested database operations (returns false if there are
    // no operations)
    bool Rollback();
}

```

- a. Which design pattern (or patterns) would be applicable to the implementation of the API?
 - b. Show how the pattern you've decided upon implements the above api (*with code*). You are free to change the interface definition into a format that works for your chosen language.
5. Assume you are the architect for an application with several hundred individual services. Some services have markedly different scalability, availability, and runtime characteristics. Describe a method for load balancing your services assuming that you have a Facade or Front Controller governing access to them.
 6. Provide a UML static model for the following problem statement including domain objects and, as appropriate, internal implementation details:

The software we're building is a point-of-sale (POS) system that supports cash registers in stores. These registers process sales (for now). Sales are comprised of a sales header describing the customer and the transaction date. Each sale has a number of line items describing the product, a quantity, and a flag indicating if the product is taxable or not. Depending on the customer, the product, or some other arbitrary Sales and Marketing whim, a discount might be applied to a line item or the entire sale. The customer pays for the purchase based on the total sales amount of all line items and / or the sale as a whole.

- We need a way to get access to the current / correct tax calculator. This calculator is provided by a number of third party services and we don't know until runtime which processor to use
- We need to determine the correct price to charge the customer based on any discounts, the product types, and any applicable taxes
- We want to support the notion of customer discounts. Some examples include:
 - During one period all sales get 10% off, later it might be \$10 for sales > \$200
 - Seniors might enjoy a 15% off discount at any time
 - Buy one, got another at 1/2 price, 75% off, free...
 - Preferred customers may receive a discount of sales over \$400
 - ...any other discount strategies that Sales and Marketing dreams up

7. Provide the UML models (static and sequence) for the following GoF patterns:

- *Chain of Responsibility*
- *Command*
- *Strategy*
- *Factory*

8. In what ways would the image processor homework use a pipe-and-filter architecture? Compare this with a microservice architecture where the various transforms are individual services. What might this architecture look like? Use an appropriate communications strategy.

9. API design revolves around several general principles. What are those principles and what do they mean? In particular, why is your API a little language and what do we mean by that?

10. A number of criticisms have been leveled against design patterns as described in the *Gang of Four* book. One of these relates to the applicability of design patterns to application domain problems. Discuss.