

## Class 22: The OOA&D Lifecycle

**Held:** Thursday, April 22, 2010

**Summary:** We look at the complete object-oriented design process, as reflected in the Head-First text.

### Related Pages:

- EBoard.

### Notes:

- Extra credit for today's CS extra.
- Extra credit for tomorrow's CS table.
- Reading for Tuesday: Ch. 1 of *Design Patterns* (to be distributed).

### Overview:

- The OOA&D Lifecycle.
- Dijkstra's Shortest Path Algorithm.
- Subways.
- Reflecting on Head-First.
- Project.

## The OOA&D Lifecycle

- The OOAD "Lifecycle" as they describe it, is as follows
  - Feature List
  - Use Case Diagrams
  - Break Up The Problem
  - Requirements
  - Domain Analysis
  - Preliminary Design
  - Implementation
  - Delivery
- They also recommend iterating the "Requirements / Domain Analysis / Preliminary Design / Implementation" stages for each smaller problem
- They also insert an "Understand the Problem" part for each subproblem.
- What do you see as the big differences between the lifecycle diagram given in chapter 10, and the XP model?

## Dijkstra's Shortest Path Algorithm

- How many of you don't know Dijkstra's shortest path algorithm?
- How many of you know it well enough to teach it to someone else?
- If we have many people who don't know the algorithm, we'll go through it.
- If we have many people who are confused by the code, we'll go through that, too.

## Fun with Subways

We'll explore their subway code in some depth.

- What do you think about their decision not to use a Graph data structure?
- What hidden assumptions might be encoded in the design of a subway that we should explore?
- What do you see as flaws in the code on p. 524? How does that relate to what we've learned in the previous chapter.

## Reflecting on Head First Object-Oriented Analysis and Design

- Pedagogy vs. presentation
  - The "stop and think about this" pedagogy is important. It's certainly how I prefer to teach my classes.
  - But it gets lost in the silliness, which I think is geared toward students with different characteristics than yours.
- Big pictures vs. small picture issues.
  - We all seem to dislike a lot of their code.
  - But our dislike of the code does not obscure that they make some important points.

## Project

Questions sent to me

- What is the "main" class of our project?
  - To be written
  - There will probably be two main classes: One for the local version and one for the networked version.
- What owns everything?
  - The main class
- How does information flow from the board to the GUI?
  - Covered last class

---

Copyright © 2010 Samuel A. Rebelsky. This work is licensed under a Creative Commons Attribution-NonCommercial 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/2.5/> or send a letter to Creative

Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.