Study required for the Midterm Test

The midterm test will be a 70-minute (1 hour and 10 minutes) closed-book exam. No supporting material is allowed. The test will take place on Wednesday, April 13, 2005 from 5:30 pm in the regular classroom PE-208. Its weight is 20% of the course grade.

For this test you are required to study the two textbooks, and to know your own project (Specification and Design parts).

PART I The chapters required from the [Arlow & Neustadt, 2002] textbook are the following:

Chapter 1 What is UML?
Chapter 2 What is UP? (sections 2.2., 2.6, 2.7, and 2.8 only)
Chapter 3 The requirements workflow (sections 3.5 and 3.6 only)
Chapter 4 Use case modeling
Chapter 7 Classes and objects
Chapter 8 Finding analysis classes
Chapter 9 Relationships
Chapter 10 Inheritance and polymorphism
Chapter 13 Activity diagrams
Chapter 19 Basic statecharts

From the above, the following sections or sub-sections are not required:

Section 1.3 The birth of UML
Section 1.8 UML common mechanisms
Section 2.1 Chapter 2 roadmap
Section 2.3 The birth of UP
Section 2.4 UP and the RUP
Section 2.5 Instantiating UP for your project
Section 2.9 UP phases
Section 3.1 Chapter 3 roadmap
Section 3.2 The requirements workflow
Section 3.3 Software requirements- metamodel
Section 3.4 Requirements workflow detail
Section 7.7 Object construction and destruction
Subsection 9.4.2.3 Hierarchies and networks
Subsection 9.4.5 Association classes
Subsection 9.4.6 Qualified associations
Subsection 9.5.2 Abstraction dependencies
Subsection 9.5.3 Permission dependencies
Section 13.10 Signals
PART II: The laws required from the [Endres & Rombach, 2003] textbook are as follows:

From Chapter 2: Glass’ law (L1)
From Chapter 2: Boehm’s First law (L2)
From Chapter 3: Curtis’s law (L5)
From Chapter 3: Constantine’s law (L7)
From Chapter 3: Fitts-Shneiderman law (L10)
From Chapter 4: DeRemer’s law (L11)
From Chapter 4: Lanergan’s law (L14)
From Chapter 4: McIllroy’s law (L15)
From Chapter 9: Sackman’s second law (L31)
From Chapter 9: Brook’s law (L36)
From Chapter 10: Kupfmuller’s law (L38)
From Chapter 10: Gestalt laws (L39)

Three of the above laws will be given in the test. You will be required to choose two (out of the three given) and comment on them in terms of: applicability, one significant evidence, main theory, and your own personal opinion.

Sample questions for Part I are as follows:

1. Briefly describe your project in CS790z (10 to 15 lines description) and give 5 examples of functional requirements and 2 examples of non-functional requirements from your team’s project. In the description, indicate the goals of project and its most distinguishing characteristics. Also, indicate why you think yours is an interesting and useful project.

2. Explain what is meant by the “4+1 views” architecture in UML. Briefly describe the five views of this architecture.

3. Explain what is meant by a use case and what is meant by a scenario. Indicate what is a primary scenario and what are secondary scenarios. Give an example of a use case using the template used in the textbook for use case specification. The use case should be related to your project and should have at least 5 steps.

4. Indicate what is a project glossary and why is it this useful. Give five examples of terms that are (or could be) included in your team project glossary. Explain the meaning of the terms.

5. Explain what is the difference between an analysis class and a design class.

6. Indicate the UML notations for activity diagrams (start and stop states, action states, sub-activity states, transitions, decisions, forks, joins, and swinlanes). Give an example of an activity diagram that includes at least 6 action states (not counting start and stop states), a fork, a join and two swinlanes. Provide some additional (textual) description to your activity diagram to explain its meaning.

7. Indicate the UML syntax for statechart diagrams, (detailed syntax for states and detailed syntax for transitions) and enumerate the four types of event that trigger transitions (note that you need not explain each type of event). Give an example of a statechart diagram that has at least 6 states (not counting start and stop states). Provide some additional (textual) description to your activity diagram to explain its meaning.

8. Explain why UML is considered a unified modeling language for software systems.

9. Outline the CRC (Class-Responsibilities-Collaborators) analysis technique for finding analysis classes in object-oriented modeling.

10. Explain the meaning of the relationship UML building block and indicate what is meant by the navigability of a relationship. Give an example of relationship between two classes and indicate whether this relationship is navigable or not. Explain why it is or it is not navigable.