What will the final look like?

• General characteristics
  – Short answer questions worth about 50 points total
  – Essay questions worth about 30 points total
  – Problems/programming questions worth about 70 points total
What will the final look like?

• You may bring in one 8.5 x 11 sheet of notes, front and back, with anything on it you desire.
• You will have 2 hours (120 minutes) to complete 150 points worth of questions
  – That’s approximately 1.25 points per minute
  – Alternately, that’s a 10-point problem in 8 minutes.
How should I study?

• Examine the outcomes for the course and for each week since the midterm in the course. Ask yourself, “can I do these things?”

• Is there anything you can eliminate?
  – Yes. Nothing specific to Cake framework (but there could be questions about frameworks in general).
How should I study?

• Examine the outcomes for the course and for each week since the midterm in the course. Ask yourself, “can I do these things?”

• Is there anything you can eliminate?
  – Yes. Nothing specific to Cake framework (but there could be questions about frameworks in general).

Not comprehensive means that I won’t ask anything specific from earlier sections, but you can’t forget everything either!
What outcomes?

• Course outcomes
  – Design, code, test, and debug programs using a server-based scripting language.
  – Persist objects in a relational database.
  – Compare and contrast Model 1 and Model 2 web-architectures.
  – Implement object-oriented model, view, and controller components.
  – Implement basic security techniques for web information systems.
What outcomes?

- Course outcomes
  - Design, code, test, and debug programs using a server-based scripting language.
  - Persist objects in a relational database.
  - Compare and contrast Model 1 and Model 2 web-architectures.
  - Implement object-oriented model, view, and controller components.
  - Implement basic security techniques for web information systems.

These are *always* fair game, regardless of before or after the midterm.
What outcomes?

• Week 8
  – Write regular expressions that test, capture, and replace data within strings
  – Explain the purpose and use of exception handling for error detection and correction
  – Use the keywords try, throw, and catch to implement exception handling
  – Use regular expressions and exception handling to validate data.
What outcomes?

• Week 9
  – Design databases from real-world problem statements
  – Normalize databases
  – Employ SQL to create database tables and indices
  – Employ PHP scripting to create and load tables with initial data
What outcomes?

• Week 10
  – Use PDO to interact with databases.
  – Employ SQL to query, update, and delete entities from databases.
  – Employ multi-table SQL joins to solve problems.
What outcomes?

• Week 11
  – Discuss reasons to avoid and alternatives to user-entered HTML markup in web-applications.
  – Explain how a browser uses a certificate to establish an encrypted connection to a server.
  – Compare and contrast ACL and RBAC approaches to authorization.
  – Implement authentication/authorization.
What outcomes?

• Week 12
  – Explore the security implications of file uploads
  – Write code that receives, stores, processes, and transmits files uploaded via the web browser.
What outcomes?

• Week 13
  – Send e-mail from web applications.
  – Invoke web services from remote sites using cURL.
  – Examine JSON as a data-interchange format.
What outcomes?

• Week 14
  – List the advantages of using a web-framework for application development
  – List and explain the typical features of a web-framework
  – Examine some of the features of the Cake PHP framework.