MCT620
Distributed Systems
Module Handbook

Master of Science in Software Engineering & Database Technologies (MScSED)
Diploma in Software Engineering
# Table of Contents

1 Module Details  
  1.1 Module Description 2  
  1.2 Prerequisites 2  
  1.3 Module Objectives 3  
  1.4 Required Text 3  
  1.5 Module Assignments 3  
  1.6 Module Grading 4  

2 Module Outline 5
1 Module Details

1.1 Module Description

This Distributed Systems module brings together many of the different strands of study in the MSc in Software Engineering & Database Technologies programme: database applications, software engineering, middleware, architecture, and Java programming. It challenges students to combine these into a powerful technical platform for complex modern problem solving. The course aims to balance a practical understanding of various programming concepts with a higher-level architectural facility for engineering Distributed Systems. Right from the start, this balance should be obvious: in Week 1, students will deploy a simple Web application to a cloud platform. They will follow with a critical analysis of that cloud platform thereby building their practical, critical and analytical muscle for designing and building these systems.

Students will become very familiar with the common challenges of building Distributed Systems - and over the weeks they will build an evaluative framework which they can apply to nearly any Distributed Systems scenario. This framework will be built on principles that have stood the test of time so it is unlikely to change much as new approaches and technologies evolve. This foundation framework should see students through a career and not just through the 'next big thing'.

Over the following weeks students will spend time on a number of important evolutions in Distributed Systems including relational & non-relational database systems, Web Services and Java EE. They will also get an appreciation of legacy systems like CORBA, RMI and RPC, right down to Sockets.

1.2 Prerequisites

The course will involve a good deal of network-based, distributed programming using the Java programming language and its related APIs and frameworks. It is essential, therefore, that course participants have a good working knowledge of Java and be familiar with good software design principles (i.e. completion of module MCT619 Object Oriented Programming, or equivalent). Some prior knowledge of Networks and Data Communications would also be useful, but is not essential.
1.3 Module Objectives

The module aims to:

• Provide students with a clear understanding of the evolution of Distributed Systems
• Introduce students to the newest distributed paradigms e.g. Cloud Computing and Massively Parallel Process AKA ‘Big Data’ and Web Services
• Bring together programming frameworks and database management in a cohesive, applied approach
• Give students a theoretical framework which they can use to evaluate any Distributed System
• Apply Distributed Systems theory to practical scenarios

1.4 Recommended Text

There is no required textbook for this module. Students will be directed to online open resources to support their work in the module.

The following textbooks are supplementary textbooks which students may find helpful:


1.5 Module Assignments

Details of module assignments and a sample final examination are detailed fully in the MCT620 Facilitator Guide. Student assessment will take the form of:

• Weekly Virtual Lab activities
• Weekly assignments
• Participation in the weekly forum discussions
• A final exam
### 1.6 Module Grading

<table>
<thead>
<tr>
<th>Assessment</th>
<th>Weight</th>
<th>Workshop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forum participation</td>
<td>20%</td>
<td>1 to 8</td>
</tr>
<tr>
<td>vLabs</td>
<td>30%</td>
<td>1, 2, 3, 4, 6</td>
</tr>
<tr>
<td>Weekly assignments</td>
<td>30%</td>
<td>1, 2, 3, 5, 6, 7</td>
</tr>
<tr>
<td>Final exam</td>
<td>20%</td>
<td>8</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td>100%</td>
<td></td>
</tr>
</tbody>
</table>
## 2 Module Outline

<table>
<thead>
<tr>
<th>Workshop</th>
<th>Sections</th>
<th>Assessment Summary</th>
</tr>
</thead>
</table>
| 1. Getting started in the Cloud, some Definitions and some History       | • Amazon Web Services  
• Definitions and challenges in the study of Distributed Systems  
• Evolution of Distributed Systems | • Deploy a simple application to AWS  
• Analyse network traffic using Wireshark |
| 2. Distributed Systems Frameworks - Old and New                          | • Sockets  
• From RPC to CORBA | • Deploy a simple application (using Java Socket API or CORBA)  
• Analyse scalability |
| 3. Persistence: from the RDBMS to NoSQL                                 | • Newer web frameworks  
• The evolving role of the database | • Create database/persistence layer on AWS server  
• Write a PHP web application to interact with database |
| 4. Web Services & SOA                                                   | • Web services and heterogeneity | • Create a Web Services version of app |
| 5. Cloud Computing                                                      | • Cloud computing  
• Delivery models | • Use AWS to improve last week’s app  
• Implement new AWS feature and assess its efficacy |
| 6. Massively Parallel Processing and Hadoop                             | • Hadoop  
• Possibilities of big data | • Implement application on locally installed Hadoop cluster |
<p>| 7. Applying technologies and concepts of                                | • Challenges of Distributed | • Research activity: Future of cloud |</p>
<table>
<thead>
<tr>
<th>Distributed Systems to real problems</th>
<th>Systems</th>
<th>computing</th>
</tr>
</thead>
<tbody>
<tr>
<td>8. Review and Final Exam</td>
<td>• Review</td>
<td>• Final Exam</td>
</tr>
</tbody>
</table>