LAB GUIDE

IN4391 Distributed Computing Systems

Alexandru Iosup
(course manager and lecturer)
A.Iosup@tudelft.nl
http://www.pds.ewi.tudelft.nl/~iosup/

February 10, 2015

This document is part of the set of materials for the IN4391 course. The Course Guide provides an overview of the course and thus complements this document.

1. Structure of the Lab

The Lab consists of one exercise, to be solved in groups of 2 students. Provisions for larger teams (up to 4 students) and, exceptionally, for students working alone, can be made after a discussion with the course manager. If you do not have a partner, we will try to resolve this at the end of the first lecture. You can also solve this by sending an email to the TA (see Section 3) with both names and student IDs (*studienummers*); the TA will then assign you to a group.

General requirement for the lab exercise: design a distributed system that demonstrates complex operations on realistic scenarios, create an implementation, perform real-world experiments, and write a report. In general, this means that we expect to see requirements refined from the lab exercise, designs that cover the requirements, and systems that work. Reports are expected to include a discussion about system design, implementation details, and experiments featuring realistic scenarios. Analysis of the experimental results is important for top grades. Exceeding the basic requirements expressed in the exercise description can lead to additional points. Extending the exercise in directions not specifically identified in the exercise description is not discouraged, but needs to be discussed and agreed with the TA.

No lab room has been scheduled, students are expected to work from their own PCs and use the machines provided by the DAS-4 multi-cluster system. Provisions for students without their own PC may be offered after a discussion with the course manager, depending also on availability of machines in the work room of the PDS group.

The exercise combines elements of Chapter 4 (communication), Chapter 6 (synchronization), Chapter 7 (consistency and replication), and Chapter 8 (fault tolerance) of the course text book (Andrew S. Tanenbaum and Maarten van Steen, Distributed Systems: Principles and Paradigms, second edition, Prentice Hall, 2007). **In general, you will need to study and use in the lab material of the course before it has been discussed during lectures (flipped classroom). We expect MSc students to use this opportunity to test their ability to understand and think critically about the material.**

2. Signing up

Students who want to participate in the Lab should enroll in Blackboard before the second lecture (February 18, 2015).

3. Assistance and Assignment Review Process

The Teaching Assistants (TA) for the lab exercises are Mihai Capotă, who can be reached at mihai@mihaic.ro, and Yong Guo, who can be reached at Yong.Guo@tudelft.nl. Their contact info can be also found on Blackboard, under **"Contact Info"**. For 2015, the contact hours are each Monday, from 2pm to 4pm; you can of course contact the TAs and find a meeting arrangement that suits both sides.

The task of the TA is to help with extreme cases and important problems, and not to debug small technical details.

The review process for the lab assignment starts with an email sent by you to the TA, followed by a confirmation sent by the TA to you; a possible request for improving the report; a possible demonstration; and email or face-to-face exchanges related to grading. We discuss each in the following, in turn.

When you have completed an assignment and want to have it approved, you have to send the code and the report to the TA via email. If your attachments exceed 5MB, place the files online on a file-sharing site, such as Dropbox, and send a link in the email. Expect to be notified in a couple of days that the assignment has been received; should this not happen, try again; should this fail as well, CC the course manager. (Regarding the use of Blackboard, since 2012 email has proven to be more available.)

From past experience, writing technical reports of good professional standard is a skill that needs improvement. Expect a request from the TA to improve your report, coupled with a review of strong and weak points, so you can also learn during the process.

Unless the TA agrees otherwise, the demonstration takes place in the office of the TA or in the MSc room at the 7th floor of EEMCS. The demonstration involves showing that the system works, exemplifying its main features, and discussing code and design elements.

The grading process takes into account your pre-defined and additional (bonus-worthy) contributions. Grading is explained in Section 4. You can expect the TA to ask details about the time you spent on the exercise or a record of usage of cloud infrastructure (if you use Amazon EC2).

Note: the operational infrastructure for this Lab has been designed for a workload of 15 MSc students. Should the actual number of students be much larger, expect delays in processing assignments.

4. Grading

In general, we expect students to focus more on learning about distributed computing systems and on thinking independently about the topics of this course, and less about how to pass the course. The grading philosophy of this course reflects this expectation: students receive points not only for regular (pre-defined) activities, but also for exceptional (bonus-worthy) work. **The final grade of the course is the sum of all the points received during the course, divided by 1,000. The bonuses are only valid for the current academic year.**

The pre-defined Lab activity that can lead to points is:

1. Lab exercise (**max 4,000 points**).

Among the exceptional activities, students should expect:

- Extra lab assignments (self-defined) or
- Extensions to existing lab assignments (open choice from a set of pre-defined options).

The exact bonuses for the exceptional activities are described in each of the Lab Assignments.

5. Deadlines

The course has the following deadlines:

- **(mandatory) February 18, 2014 (week 3.2): Enroll in Blackboard. Students who have not enrolled will not be graded and cannot receive credits for this course.**
- **(recommended)** February 25, 2015 (week 3.3): Discuss with TA your group's system requirements and proposed approach for replication and consistency, for the lab exercise.
- **(recommended)** March 11, 2015 (week 3.5): Discuss with TA your group's system requirements and proposed approach for fault-tolerance and scalability, for the second lab exercise.
- **(mandatory) anytime before March 25, 2015 (week 3.7): Demonstrate to TA the system implemented for the lab exercise.**
- **(mandatory) March 25, 2015 (week 3.7): Turn in to TA the lab-exercise report.**

We expect students to follow the mandatory deadlines strictly.

We strongly recommend students to follow the deadlines marked with “recommended” as if they were mandatory deadlines. However, we believe that setting a mandatory deadline for each milestone is micro-management, which defeats our expectation of students choosing their own paths. **From past years’ experience, we have observed that students who do not follow the recommended deadlines struggle to finish in time. Because this comes at the expense of your own education, we expect you to take the best decision for yourself.**

FAQs

1. **Q: How can students define own goals and receive points for them?**
   A: There are several opportunities we offer our students. For example, the lab exercises are purposely refined from real-world products, one from the academic world and one from the industry. Students can make extensions to the exercises even to the point where they can become real, commercial products.

2. **Q: Why are so many deadlines *recommended* rather than *mandatory* (strictly enforced)?**
   A: We believe that students at MSc-level should already be able to organize their schedule. Thus, the course has strict deadlines only for turning in the final result of the assignments. We strongly encourage all of our students to take the recommended deadlines as strict, but doing this remains the student’s responsibility.

3. **Q: How much help should students expect for lab work?**
   A: Similarly to question 4, we expect MSc students to be able to complete the exercises entirely without help. A simplistic reason is that if completing our simple exercises is troublesome, how can students be expected to have lab-related exceptional activities and, later on, how can they be expected to perform well when hired to work on distributed systems? In exceptional cases, such as unclarity in the definition of the assignment or exceptional hardship, students should of course contact the teaching assistant (Yong Guo) or even the course manager (Alexandru Iosup).