Effective Course Projects for Teaching Distributed-application Development
A tutorial in course-project design, coaching, and evaluation

Distributed systems, and more specifically, distributed applications are pervasive in today's highly connected world. Even seemingly simple mobile apps, for example, are often sophisticated distributed applications under the hood. Consequently, most students entering a software-engineering profession need to be familiar with distributed-application development, including its design principles, best practices for implementation, and testing techniques. Their skill sets need to include network communications, inter- and intra-process concurrency, proper handling of partial failures, the ability to manage multiple concurrent communication channels, task synchronization, and efficient communication protocol design. They also need to be familiar with challenges and solutions related to reliability, security, scalability, extensibility, maintainability, and more.

Teaching these concepts effectively and providing meaningful opportunities for students to develop the necessary skill set can be very challenging for instructors. This tutorial will cover goals, challenges, and techniques for

1. designing course projects, so they are engaging and cover as many of the core concepts as feasible,
2. coaching students to successful completion of substantial, resume-building course projects, and
3. evaluating the student performance in constructive and fair ways.

Participants in the tutorial will have the opportunity to share their own insights and apply what they learn to design a sample course project that makes use of an indoor drone as one of its components. The drone has a simple API based on text commands transmitted through UDP datagrams. Participants are encouraged to bring laptops or tablets, but having one is not essential.

The learning objectives for this tutorial include:

1. Gaining a better understanding of the core concepts and skills that students need in order to be effective distributed-application developers.
2. Gaining a better understanding of how distributed-application development concepts and skills can be taught in conjunction with good software engineering principles and practices.
3. Gaining new ideas about how to make a course project more engaging.
4. Gaining new insights into how to better coach students to successful completion of a substantial project.
5. Gain new insights into how to evaluate student performance in a constructive and fair way.