Pre-Course Meeting

Practical Course
Software Engineering in Robotics

Prof. Dr.-Ing. habil. Alois Knoll
Florian Walter, M.Sc.
Mahmoud Akl, M.Sc.

Summer Semester 2020
Agenda

• **What** is this course about?
• **What** is the goal of this course?
• **What** do we expect from you?
• **How** to apply for this course?
• **What** is the schedule?
What is this course about?
Bringing Robotics to the Cloud

Many robotics applications can considerably benefit from cloud services:

• Scalability and support to a wide range of applications from simple data storage to machine learning
• Standardized service-oriented interfaces
• Connection to other cloud services
• Virtually unlimited compute power and storage
• Robotics simulations for development of machine learning and artificial intelligence applications
What is the goal of this course?
Develop Cloud Services for (Simulated) Robots

- Use cloud infrastructure to implement on-demand services for robotics
- Design and implement deployment strategies for machine learning with robot simulation
- Create APIs for cloud-based robotics systems
- Create new applications for robotics from existing cloud services
Possible Topics (Tentative)

• A database storage for machine learnings with cloud robotics
• Automatic scaling of parallel simulations with NRP and Kubernetes
• Highly parallelized robot simulations with DeepMind Impala and NRP
• Dynamic game-like environments and environment randomization in NRP experiments
• Add support for cloud-based services to the NRP (TensorFlow, PyTorch, Intel Loihi etc.)
• Robot modeling ...
<Insert Project Title>

Propose your own project!
What do we expect from you?
Requirements

• Practical programming skills in C/C++ and Python
• Good knowledge of Linux and networking
• Experience with Docker, web programming and software engineering is of advantage
• Basic knowledge of robotics is beneficial but not required
• Work in teams of two students
• Self-reliant and self-organized style of working

... and: you are really interested in the topic!
How to apply for this course?
Application Process

Please submit ...  

... a brief description of your skills and practical programming experience  

... a short paragraph outlining your motivation to join the course  

... a transcript of records

Send your applications to florian.walter@tum.de by February 13
What is the schedule?
## Application

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Today</td>
<td>Pre-Course Meeting</td>
</tr>
<tr>
<td>07.02. – 12.02.</td>
<td>Enter course preferences in the matching system and submit your application for this course</td>
</tr>
<tr>
<td>20.02.</td>
<td>Course assignment is available in the matching system</td>
</tr>
<tr>
<td>27.03.</td>
<td>Latest date for cancelling the course registration</td>
</tr>
<tr>
<td>10.04.</td>
<td>Final project descriptions are published</td>
</tr>
<tr>
<td>20.04.</td>
<td>Preliminary assignment of groups and topics</td>
</tr>
</tbody>
</table>
During the Semester

• Regular status meetings (every three weeks)
• Kick-off meeting with project presentations at the beginning of the semester (second week of the lecture period)
• Final presentation in the last week of the lecture period
• Communication, support and submissions through Moodle
• Code is organized in shared repositories
• Documentation will be submitted as a final project report (IEEE two-column format, 8 pages)

The final schedule will be published on Moodle by the beginning of the semester
Grading

- Kick-Off Presentation: 35%
- Midterm Presentation: 10%
- Final Presentation: 10%
- Project Work: 35%
- Final Report: 10%