

Progetti

Corso di Sistemi Distribuiti e Cloud Computing A.A. 2019/20

Valeria Cardellini

Laurea Magistrale in Ingegneria Informatica

Project choice and deadline

- Send me an email by February 24th 2020 with the following info:
 - Group members (names and emails)
 - Chosen project (and description of the application, if applicable)
- A maximum number of available slots for each project (see course site for details)
 - Assignment with a FIFO discipline
- Communicate promptly and motivate any change to the group
- The project is valid only for A.A. 2019/20

Project delivery

- When to deliver
 - By **September 2020**
 - About one week before the project presentation
 - No prefixed dates for the presentation
- What to deliver
 - URL pointing to cloud storage or code repository containing: project code, report and (if pertaining) datasets of experimental results
 - Paper copy of the report (only the report, no code)
 - Write the report possibly as a scientific paper (**maximum 10 pages** using ACM or IEEE format)
 - ACM proceedings templates <http://bit.ly/1UbvNLJ>
 - IEEE proceedings templates <http://bit.ly/1lunzPr>

Project presentation

- All the group members discuss their project in the same date
- What to present
 - Prepare a presentation with some **slides**
 - Prepare a live **demo** of the project
 - Each group member discusses a part of the project (it is your choice which part)
 - Maximum **10 minutes** per member
 - I will check the time and interrupt you
 - Questions during and at the end of the presentation

Common requirements for all projects

- You can choose the programming language
- You can use support libraries and tools to develop your project (of course they should not overlap with the project goals!)
 - Be careful: their use must be properly mentioned in the project report
- System/service with configurable parameters (no hard-coded!)
 - Through a configuration file/service
- You must test all the functionalities of your developed system/service and present and discuss the testing results in the project report

Common requirements for all projects (2)

- System/service supports multiple, autonomous entities contending for shared resources
- System/service supports real-time updates to some form of shared state
- System/service state should be distributed
 - The only allowed centralized service can be one that supports users logging on, adding or removing clients or servers, and other housekeeping tasks
- System/service scalability and elasticity
- System/service fault tolerance
 - In particular, system/service continues operation even if one of the participant nodes crashes (optionally, it recovers the state of a crashed node so that it can resume operation)

Grant for cloud services

- You can use Amazon Web Services (AWS) through AWS Educate <https://aws.amazon.com/education/awseducate/>
 - 100 \$, renewable each year
 - Additional 10 \$ with the student developer pack from GitHub <https://education.github.com/pack>
 - Plus AWS Free Tier for 12 months (unless you have already registered for an AWS account)
 - Check the list of available services!
 - For instructions on how to request the AWS grant, see the email you received

Grant for cloud services

- Alternatively, you can use Google Cloud Platform (GCP)
- Two options
 1. 300 \$ for 12 months <https://cloud.google.com/free/>
 - Plus Always Free (unless you have not previously signed up for free trial)
 - Check the list of available services!
 2. Grant (approximately 50\$ per student) *(to be confirmed)*
 - Send me your Gmail address

Project types

- See first lesson of the course
- Type A
 - Final score:
 - 50% written exam (plus elective oral exam)
 - 50% project (2-4 students per team)
- Type B
 - Final score:
 - 75% written exam (plus elective oral exam)
 - 25% individual project
- Changing the project type (from A to B or viceversa) is **not allowed**

Projects A overview

- Project A1: An edge/fog application for tourism
 - 3/4 students per team
 - Mobile users at the network edge
 - Point of interests and related information
 - Offloading compute-intensive tasks
 - Caching at the network edges
 - Broadcast to users in a given area
 - Project for CINI Smart Cities University Challenge 2020
 - Deadline: July 3, 2020

Projects A overview

- Project A2: Your own fog application with IoT devices
 - 3 students per team
 - IoT devices at the the network edge
- Project A3: Context-aware message queue system in Go
 - 2/3 students per team

Projects B overview

- Project B1: Serverless computing application
 - 1 student per team
 - Requirements: use at least 2 Cloud services
- Project B2: Microservice application
 - 1 student per team
 - Requirements: use at least 2 patterns