

e516 - Engineering Cloud Computing

for Engineering, Data Science, and Computer Science Students

Gregor von Laszewski

Editor

laszewski@gmail.com

<https://cloudmesh-community.github.io/book/vonlaszewski-cloud.epub>

July 03, 2019 - 03:56 PM

Created by Cloudmesh & Cyberaide Bookmanager, <https://github.com/cyberaide/bookmanager>

Gregor von Laszewski

(c) Gregor von Laszewski, 2018

[1 CLASS](#)

[1.1 Course Syllabus Tables](#)

[1.1.1 Proposed Lecture Timeline](#)

[1.1.2 Assignments Timeline](#)

[1.1.3 Group Breakdown Checkpoint](#)

1 CLASS

1.1 COURSE SYLLABUS TABLES

Please note that the the dates may change throughout the class. It is your responsibility to check weekly and download the lecture notes on a weekly basis to see the updates.

- All assignments are due 2 weeks before semester end.

1.1.1 Proposed Lecture Timeline

The times indicate when to start a particular technology documentation or lecture. It may take you some weeks to complete some of the sections. If you know the topic well, it may take you less time so you should move ahead. For example while it may take some time for some to learn python, others have the knowledge already and should move on. The book can be used as reference material in that case.

Legend markings

- Class released +
- Class under development - or

Week	Unit	Titl	e Desc	ription
+	1	1	Introduction	Gregor von Laszewski
+	1			Class summary
+	1			Definition of Cloud Computing
+	1	2	Tools	Tools and Services
+	1			- Virtual Box
+	1			- Vagrant
+	1			- Github
+	1			- Linux

+	2	3	Python	Python
+	2			- Introduction
+	2			- Installation
+	2			- Interactive Python
+	2			- Editors
+	2			- Basic Language Features
+	2			- Modules
+	2			- Data Management
+	2			- Matplotlib
+	2			- Cloudmesh Commandshell CMD5
+	2			- OpenCV
+	2			- Secchi Disk
+	3			Data Center
+	3	4	Architectures	- NIST Big Data Reference Architecture
+	3			- Cloud Architectures
+	3			REST
+	3			- OpenAPI and Swagger
+	3			- OpenAPI Specification
+	3			- OpenAPI Service
+	3			- Github as Rest Service
+	4	5	Virtualization	Virtualization, Qemu, KVM, Virtual machines
+	4	5	Virtualization I	- Qemu
+	4	6	Infrastructure	Infrastructure as a Service
+	4			- Azure
+	4			- AWS
+	4			- OpenStack

Chameleon

Cloud

+	5			- Resources
+	5			- Hardware
+	5			- Charge
+	5			- Quick start
+	5			- KVM user guide
+	5			- CLI
+	5			- Horizon
+	5			- Heat
+	5			- Baremetal
+	5			- FAQ
+	5	8	Programming	Python for Cloud Computing,
+	5			- Libcloud
+	6		Virtualization II	Containers, Docker, Kubernetes
+	7	9	Map/Reduce	Map/Reduce , [Hadoop] (#sec:hadoop-introduction), Spark
+	8	10	Messaging	Messaging
+	8	11	Messaging	- MQTT
+	8			- Graphql
+	13		Go	Go Introduction
+	13			- Go Links
+	13			- Go Install
+	13			- Go Editors
+	13			- Go Language
+	13			- Go Libraries
+	13			- Go cmd
+	13			- Go Cloud
+	13			- Go REST

1.1.2 Assignments Timeline

Students must conduct all assignments listed here. They must conduct one project of type A, B, C, or E. The project is selected in the first 3 weeks of the semester and conducted throughout the rest of the semester.

	Week	Unit	Title	Description
+	1	A0	Survey	Fill out the Survey before Friday in the first week
+	2	A0	Bio	Post your formal Bio into Piazza
+		A1	Sections	Contribute significant sections. Do not develop redundant or duplicated content.
+		A2	Chapter	Contribute a significant chapter that may use your section to the class documentation. Do not develop redundant or duplicated content.
+		A3	Project Report draft due	Develop a draft for the project. This is a hard deadline as we integrate your draft into a proceedings over the break.
+		A3	Project Type A	Build a cloud cluster out of Raspberry Pis
+	13		Project Type B	Build a Significant OpenAPI REST Service
+	13		Project Type C	Contribute to the new Cloudmesh code
+	13		Project Type D	Your own Project Type A, B, C, D (upon approval)

- The project is a long term assignment (and are ideally worked on weekly by residential students). It is the major part of the course grade.

(*) Sections and chapters prepare you for documenting a technical aspect related to cloud computing. It is a preparation for a document that explains how to execute your project in a reproducible manner to others.

- all times are in EST

Additional lectures will be added that allow easy management of the project. These lectures can be taken any time when needed.

Date	Unit	Title	Description
+ anytime	1	Scientific Writing with markdown	
+ anytime		Plagiarism	How to avoid plagiarism and cheating
+ anytime		Markdown	How to use markdown
+ anytime	1	Scientific Writing II	
+ anytime	1	only relevant for the bibtex section fro this class, we will not use LaTeX	
+ anytime		Writing a Project Report	How to write a high quality Project report following our template
+ anytime		Bibliography Management	How to easily manage bibliographies for your Project Report

1.1.3 Group Breakdown Checkpoint

Please note that all checkins in case of group projects are visible in github. If

we detect that a group member has disproportionally fewer contributions to the project than other project team members we will invite the team for a special section in which each team member needs to explain what has been done. This is to avoid that a team member unfairly relies on other team members and does not contribute to the project. For example, a project containing three members should contribute the work of three team members and not fewer. This also means that if you want to work in a project you need to vet your team members. Do choose them based on capability and make sure they are a good fit for your project.