72939
Software Systems Engineering M

a.a. 2018-2019 Bologna

https://www.unibo.it/en/teaching/course-unit-catalogue/course-unit/2018/385373

72939 (ISS) Goal

Learn how to build in a concrete way …

… working individually or in a team …

… following agile (model-centered) methodologies

a software distributed system …

.. that must satisfy a set of pre-defined requirements …

… running on a heterogeneous set of nodes

for IOT (WOT) applications

72939 (ISS) Overview

- Software Systems
  - Components
  - Interaction
  - Behavior
- Design Patterns / (Software) Architectures / Platforms
- Programming Languages
- Software production Methodologies , Methods and Tools
- IOT / WOT Application Domains
- Using proper tools (Download / Install)

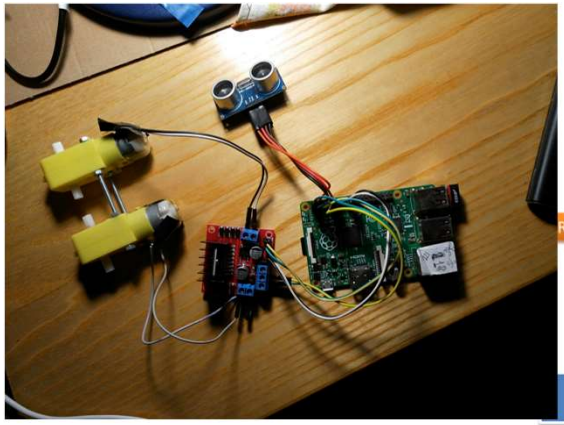
Components

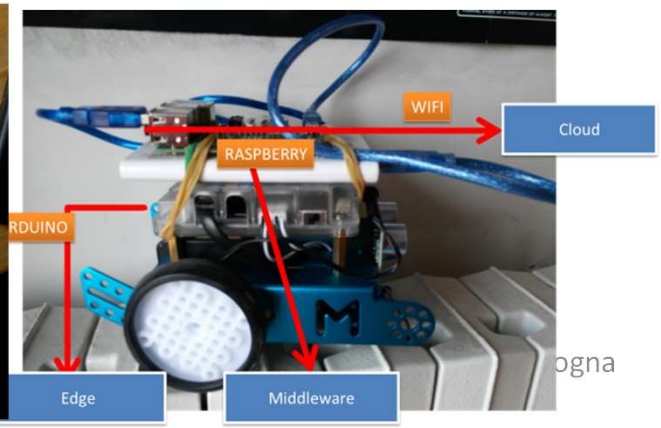
Hardware

- Processing unit
  - Arduino (Genuino Uno)
  - RaspberryPi3
  - Android
- Sensor
  - Sonar HC-SR04
  - …

Software (see Languages)

- Function
- Object
- Process
- Thread
- Actor
- Coroutine
- …





Interaction

Common memory

- Shared spaces

Distributed

- Message
  - Fire and forget
  - Request-response
- Event

Behavior

Control based

- Proactive
- Reactive
- Finite State Machine

Message / Event driven

- …

Languages / Styles

Languages

- C++
- Java
- Kotlin
- JavaScript / NodeJs
- Prolog

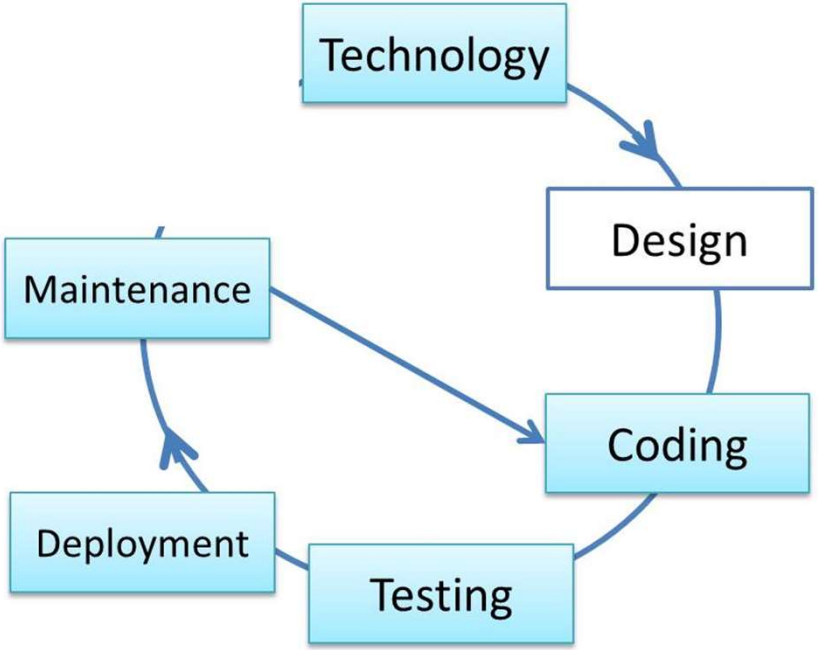
Styles

- Imperative, object
- Imperative, object
- Imperative, functional,object
- Functional
- Logical

Software production

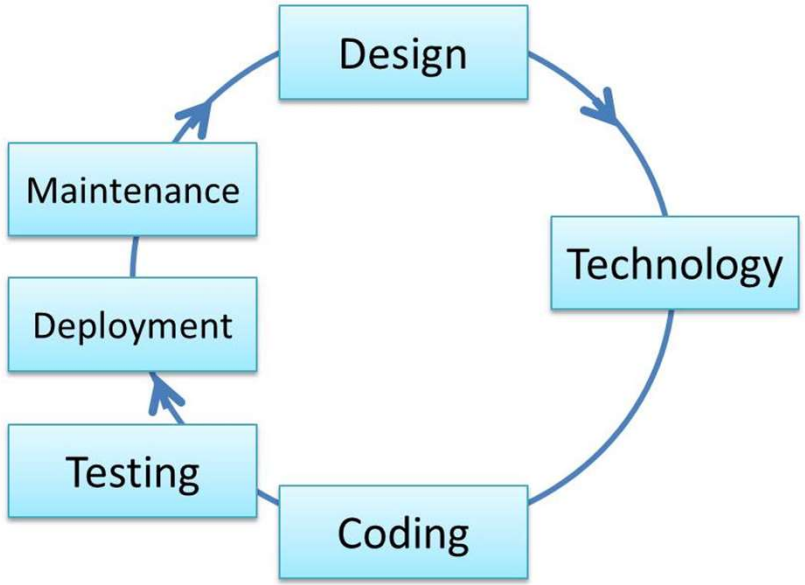
Bottom-Up

- Components/ Technology first



Top-Down

- Problem Analysis / Project first



Methodology

Model-based

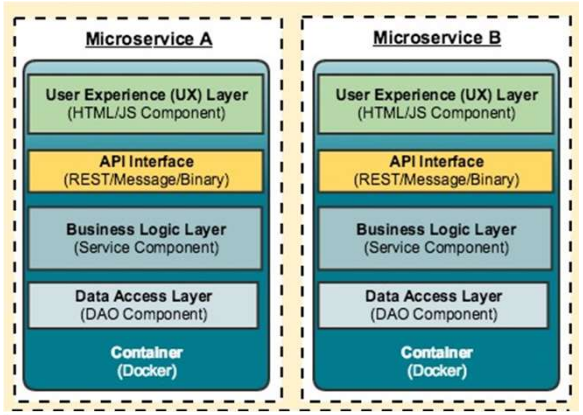
- Beyond UML
- Custom metamodel (based on Xtext)

Agile

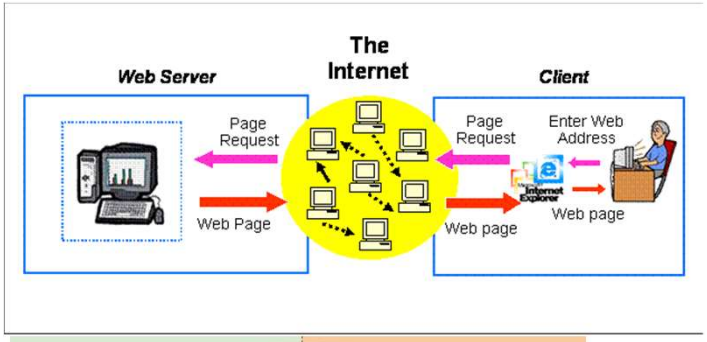
- KANBAN
- SCRUM

Architecture / Platform

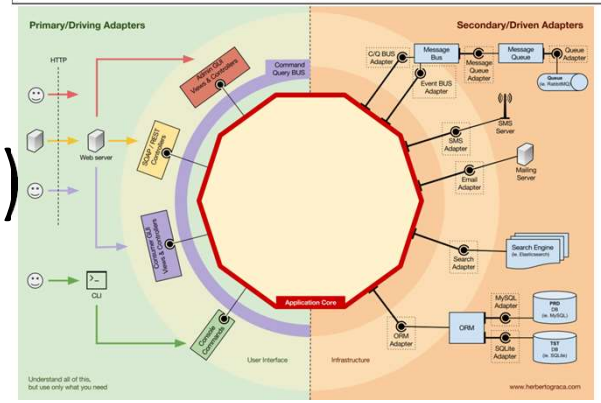
- Layered



- Web (RestFul)



- Hexagonal (Port/Adapter)

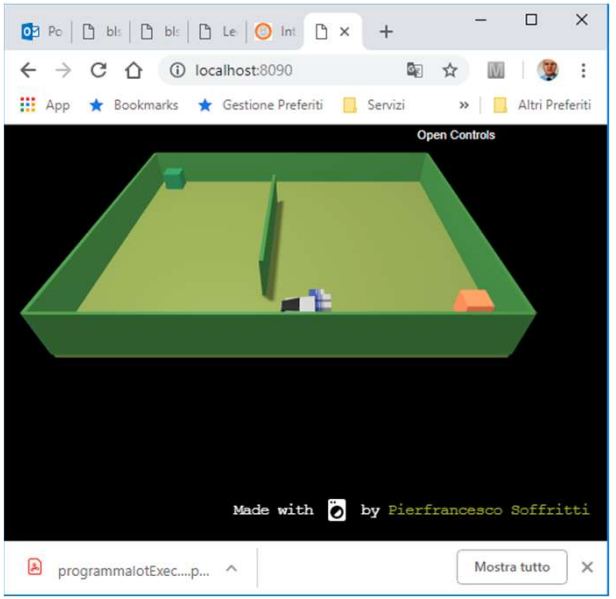


TOOLS

- GIT
- Gradle
- Intellij
- Eclipse
- XText

Application Domains

- IOT
- WOT



Dowload / Install

- Gradle
- NodeJs
- Eclipse DSL / IntelliJ IDEA
- Android Studio
- MQTT-Mosquitto
- OPTIONALLY
  - Docker

Material

- Lectures and Course site:
  http://infolab.ingce.unibo.it/iss2018/it.unibo.issMaterial/issdocs/Material/LectureBologna1819.html

- MANUALS and BOOKS (see the site)

- GIT HUB
  - Lab Code:
    - http://github.com/anatali/issLab2019.git

Assessment

- Criteria: see https://www.unibo.it/en/teaching/course-unit-catalogue/course-unit/2018/385373
- Final task examples
  - Bologna2017-2018
  - Cesena 2018-2019

FIRST APPLICATION: BLS system

- Design and build a *ButtonLed* software system (BLS) in which a **Led** starts / stops blinking each time a **Button** is pressed (by an human user).
- The system should run (at the moment) on a single computational support, e.g. a Conventional PC, a RaspberryPi or Arduino.
- The Led / Button devices can be real or virtual

WORK TO DO

- Prepare a personal PC to be used in the Lab
- Write a document not longer than two pages hat describes the architecture of the proposed solution to the BLS problem and includes
  - Name, badge number, and foto (card) of the author
- **Printed on a SINGLE PAGE**
- **To be delivered to the teacher on Feb 26st**