## Syllabus

### Course Description

<table>
<thead>
<tr>
<th><strong>Course Title</strong></th>
<th>Maker Lab</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Course Code</strong></td>
<td>76205</td>
</tr>
<tr>
<td><strong>Scientific Sector</strong></td>
<td>ING-INF/05</td>
</tr>
<tr>
<td><strong>Degree</strong></td>
<td>Bachelor in Computer Science</td>
</tr>
<tr>
<td><strong>Semester</strong></td>
<td>2nd</td>
</tr>
<tr>
<td><strong>Year</strong></td>
<td>1st</td>
</tr>
<tr>
<td><strong>Credits</strong></td>
<td>3</td>
</tr>
<tr>
<td><strong>Total Lecturing Hours</strong></td>
<td>30</td>
</tr>
<tr>
<td><strong>Total Lab Hours</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Prerequisites</strong></td>
<td>basic knowledge of programming languages and computer architectures</td>
</tr>
<tr>
<td><strong>Course Page</strong></td>
<td><a href="http://www.inf.unibz.it/~gennari/makerlab.html">http://www.inf.unibz.it/~gennari/makerlab.html</a></td>
</tr>
</tbody>
</table>

### Specific Educational Objectives

Type of course: "caratterizzanti" for L-31
Scientific area: "Discipline informatiche" for L-31

The course is designed for acquiring professional skills and knowledge. It gives general practical knowledge and skills necessary for developing basic interactive IoT solutions.

<table>
<thead>
<tr>
<th><strong>Lecturer</strong></th>
<th>TBA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Scientific Sector of the Lecturer</strong></td>
<td>INF/01</td>
</tr>
<tr>
<td><strong>Teaching Language</strong></td>
<td>English</td>
</tr>
<tr>
<td><strong>Office Hours</strong></td>
<td>TBA</td>
</tr>
<tr>
<td><strong>Teaching Assistant</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Office Hours</strong></td>
<td>-</td>
</tr>
</tbody>
</table>
### List of Topics Covered

- Principles of electronics and programming of microcomputers (Arduino)
- Planning, design and production of artefacts
- Use of machines for the constructions of artefacts
- Principles of electronics, interaction design and programming for IoT:
  - Introduction to Basics of physical computing with Raspberry Pi 3 (RPi) and other controllers for IoT RPi hats
  - Introduction to the basics of Interaction Design for IoT
  - Introduction to Python, MicroPython and high-level programming languages for IoT
  - Plan, design and develop prototypes of IoT interactive solutions
  - Use of machines for the constructions of IoT artifacts

### Teaching Format

- Lectures
- Seminars
- Workshop

### Learning Outcomes

**Knowledge and understanding:**
- Know the main methods for the design of interactive IoT products.

**Applying knowledge and understanding:**
- Be able to apply interactive design principles and patterns.
- Be able to develop IoT solutions.

**Making judgments:**
- Be able to plan and re-plan a technical project activity aimed at building an interactive IoT solution and to bring it to completion by meeting the defined deadlines and objectives.

**Communication skills:**
- Be able to coordinate the work of a project team and to interact positively with members of the group.
- Be able to interact and collaborate with peers and experts in the realization of a project or research.

**Learning skills:**
- Be able to autonomously extend the knowledge acquired during the study course by reading and understanding scientific and technical documentation in Italian, German and English.
- Be able, in the context of a problem-solving activity, to extend even incomplete knowledge taking into account the objective of the project.

### Assessment

The assessment consists of tasks: two parts. Tasks are related to the following learning outcomes:

1. Knowledge of Python;
2. The ability to write a Python program for a given problem;
3. The ability to correct a Python program for a given problem;
4. The ability to optimise a Python program for a given problem;
5. Knowledge of RPi;
6. The ability to design a physical interactive IoT solution with RPi and Python;
7. The ability to fix a physical interactive IoT solution with RPi and Python;
8. The ability to develop a physical interactive IoT solution with RPi and Python.

The assessment can be:
either during the course, in class: students tackle progressive tasks 
(a.k.a. challenges) and teachers give in-person formative feedback;
or after the course, in a final written exam: students tackle tasks without computers and others with RPi.

Class attendance is mandatory for tackling tasks during the course and using the course material (e.g., RPi, micro-electronics packages, RPi hats), which is made available for free only during class hours.

Project, consisting of small progressive tasks

The project part of the exam will assess the learning outcomes related to: (1) the ability to build basic circuits with controllers, sensors and actuators; (2) the ability to design IoT interactive prototype solutions; (3) the ability to develop small programs for the designed solutions, mainly in Python and/or MicroPython. Projects require group work.

For students attending the course, formative feedback and assessment on projects will be delivered in person when requested by the students during the course hours. Being a making course, regular attendance is highly recommended so as to exploit this type of in-person interactions besides the course material, which is made available for free to students during course hours only.

The oral exam will consist of verification questions related to the developed projects.

ASSESSMENT LANGUAGE

English

EVALUATION CRITERIA AND CRITERIA FOR AWARDING MARKS

The assessment is positive if more than 60% of tasks are correctly resolved.
Criteria for evaluating resolutions of tasks will be: their clarity, correctness and completeness, in case of Python programs; their usability and user experience in case of RPi solutions.
During labs, students work on small-size projects, which count for 50% of the mark.

The final oral exam has a presentation with questions and counts for 50% of the mark.

The oral exam will be evaluated in terms of the ability to explain the developed projects—autonomously, clearly, correctly and completely.

The projects will be evaluated in terms of the quality of the designed interactive solutions and programs, and specifically: the usability of interactive solutions; the correctness, completeness and clarity of programs.

REQUIRED READINGS

TBA
<table>
<thead>
<tr>
<th>SUPPLEMENTARY READINGS</th>
<th>-</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOFTWARE USED</td>
<td>-</td>
</tr>
</tbody>
</table>