



**Intelligent Irrigation System for Low-cost Autonomous Water Control**

**in Small-scale Agriculture**

**The INTEL-IRRIS coding challenge with IIWA**

Prof. Congduc Pham



<http://www.univ-pau.fr/~cpham> Université de Pau, France

IoT – from idea to reality

* INTEL-IRRIS is a project to build an autonomous system to optimize irrigation targeting smallholder farms
* The INTEL-IRRIS starter-kit consists of a soil humidity sensor device and an IoT gateway to collect data
* The INTEL-IRRIS Irrigation WaziApp (IIWA) is an embedded application running on the INTEL-IRRIS IoT gateway itself
* IIWA is the core of the "intelligent Irrigation in-the-box" & "plug- &-sense" INTEL-IRRIS approach
* Its objective is to enhance the irrigation indication by applying sensor calibration models with soil/plant/weather parameters

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

* This INTEL-IRRIS coding challenge is open to developers to improve the IIWA application

Knowledge Database

Decision Support System

AI

Machine Learning

Irrigation technique

Soil type

Geographical area

Crop type

Weather

**Embedded Intelligence**

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

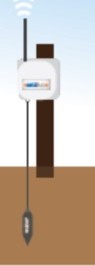
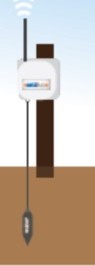
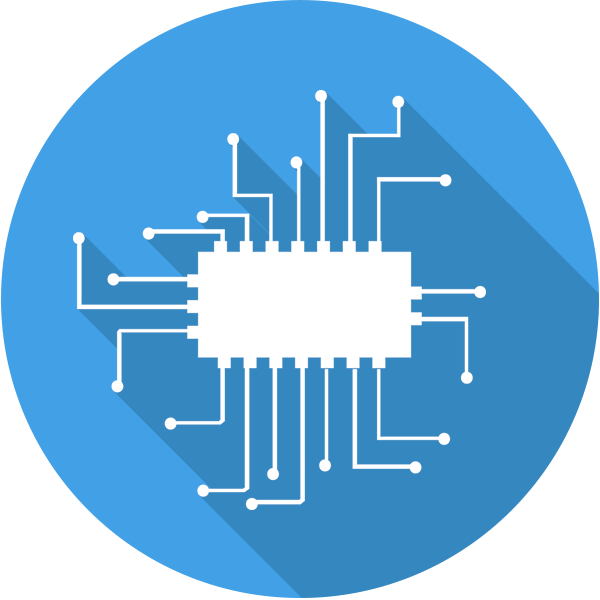
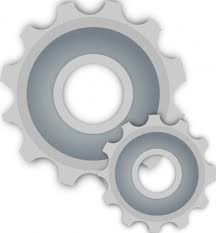
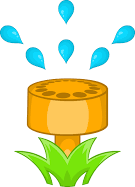
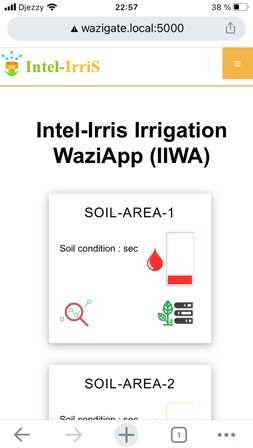
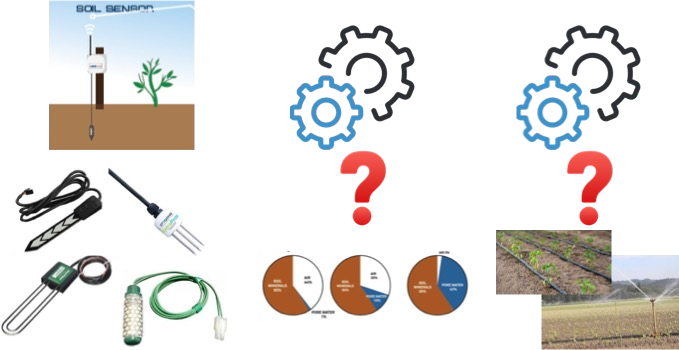
**Multiple kwnowledge streams**

**Build datasets**

Agronomist

Technology experts

Pilot fields



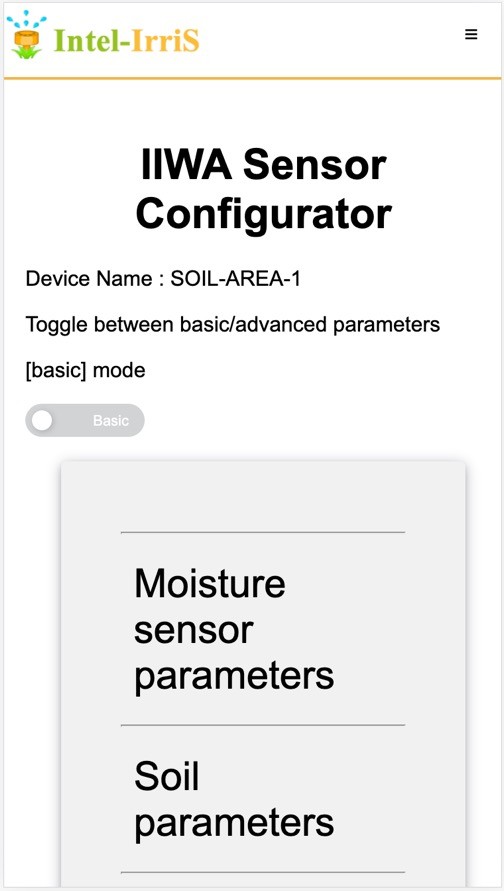
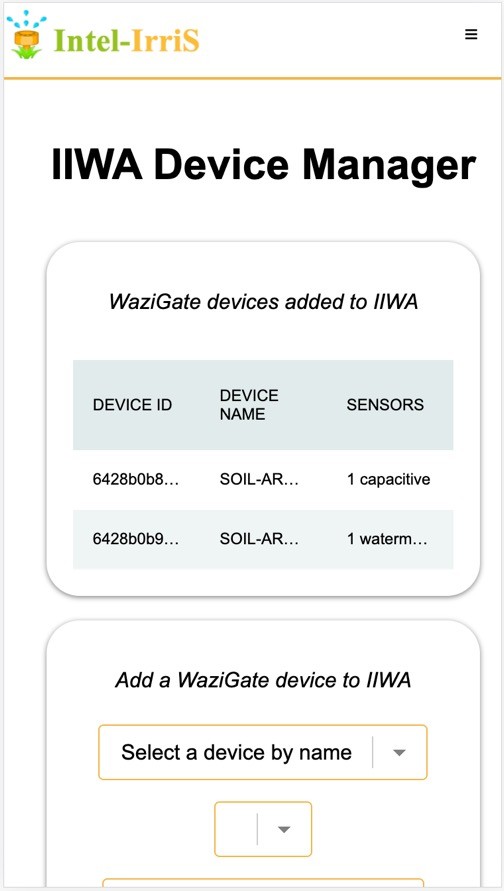
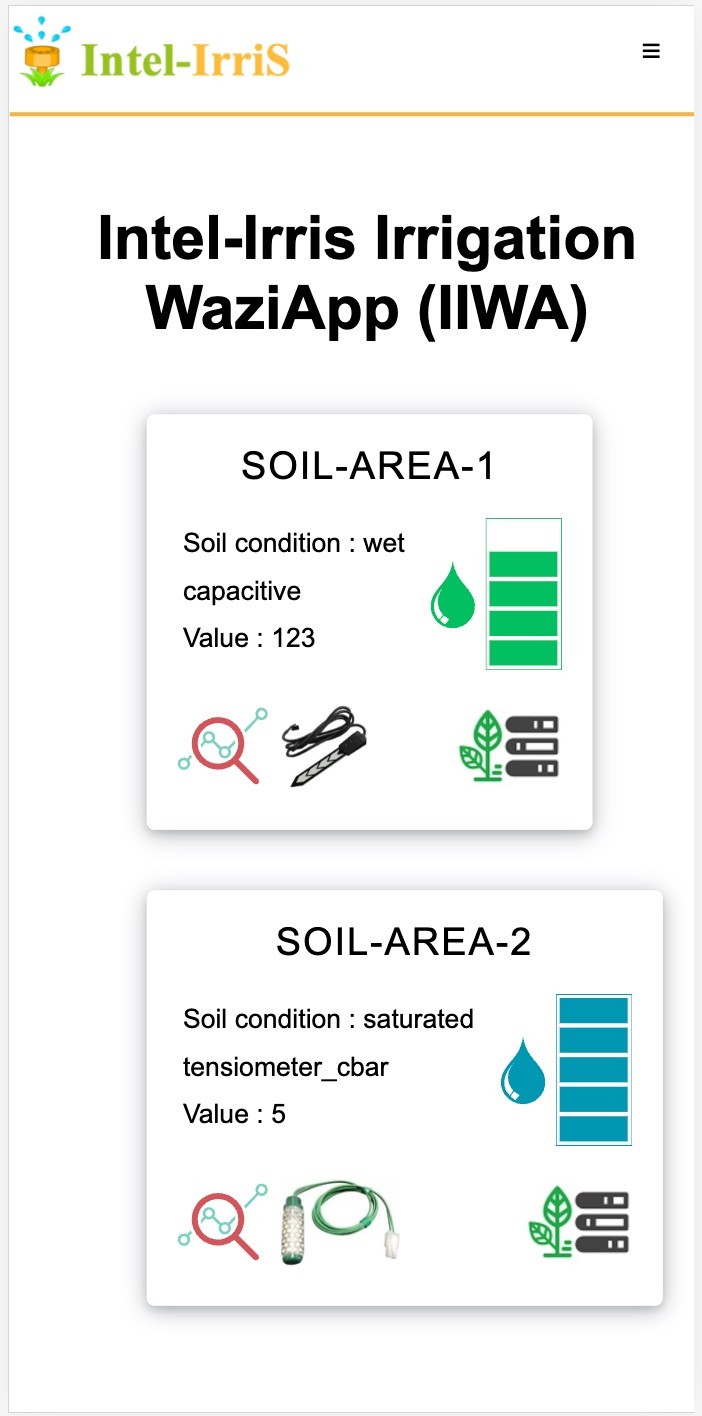
**AI/ML**

**Prediction models**

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

* Dashboard, Device Manager and Sensor Configuration



* Competitors will propose improvements on the IIWA application
* Several contribution coding categories are defined
  + Categoty 1: User Graphical Interface
  + Categoty 2: Calibration of sensor data
  + Categoty 3: Processing & transformation of sensor data
  + Categoty 4: Integration of agriculture data & knowledge
* The current IIWA code is available, competitors can use it as a skeleton to propose improvements
* It is written in Python+Flask with some HTML/JavaScript code

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

* The code can be run on desktop computers: Windows, Linux and MacOS operating system are supported
* Competitors can be organized in team and a team contact person must be designated
* Competitors must send an email to national contact persons listed at the end of this document
* The initial mail must indicate
  + The name and affiliation of all team members
  + The team contact person
  + The coding category the team will be working on
* The coding challenge will end on July 31st, 2023

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

* Team can register until July 1st, 2023
* The current IIWA code for the coding challenge can be downloaded from GitHub repository https://github.com/CongducPham/PRIMA-Intel-IrriS-IIWA-challenge
* Competitors can optionally fork the repository
* Read installation instructions in build-local folder
* Use a desktop/laptop computer to develop IIWA improvement, although IIWA will normally be run on a RaspberryPi
* Propose, develop, test and document the improvement
* Avoid as much as possible complex software dependencies that may limit the genericity of the contributions

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

* However, if the whole dependencies can be easily installed locally it can be accepted
* As INTEL-IRRIS mainly targets an autonomous system for smallholder farms, relying on Internet connectivity to get access to resources on Internet should be avoided
* However, for the coding challenge, contributions relying on Internet connectivity can be accepted provided that they do not represent the main improvements

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

* Team must submit the following items
  + The complete code of the new & improved IIWA application
  + It can be a .zip archive or a link to a GitHub repository
  + The code must be self-contained so that it can be run & tested by the jury members on their desktop/laptop computer
  + A report in .pdf explaining the technical improvements and what they bring to the IIWA application. The page limit is 15 pages, including figures and screenshots
  + A short video showing the main improvement on the IIWA application
  + A special attention must be made to clearly indicate what open-source code have been used and integrated into the main IIWA code if this is the

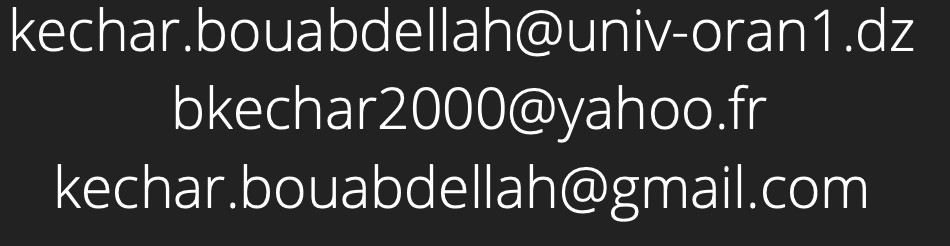
Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

case (it is not necessary to list well-known open-source libraries needed as dependencies for the contributions)

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

* Start: now
* Team can register until July 1st, 2023
* End: July 31st, 2023
  + contributions must be submitted by mail
* Algeria
  + Pr Bouabdellah Kechar, University of Oran 1
* Morocco
  + Pr Kamal Baraka, ENSA Safi
  + Pr Chahbouni Othman, ENSA Safi & Université Cadi Ayyad

Prof. Congduc Pham

<http://www.univ-pau.fr/~cpham>

