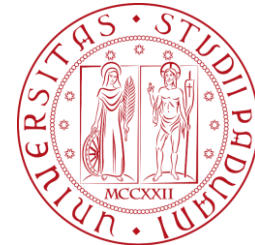


# QuantumCLEF: The Infrastructure

A computing infrastructure to develop algorithms and send submissions  
to real quantum annealers

Maurizio Ferrari Dacrema, **Andrea Pasin**, Paolo Cremonesi, and Nicola Ferro



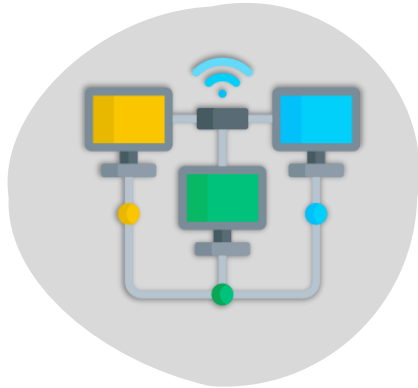
# Presentation Outline

- The importance of having a Custom Infrastructure
  - An overview of the QuantumCLEF Lab
  - Understand the role played by our infrastructure
  - What are the main objectives of our infrastructure
- How does the Infrastructure work?
  - View the components and analyse their roles
  - The interaction with quantum annealers
- Accessing the Web Application
  - Tour of the Web Application
  - Form working groups for the Hands-On part



# The importance of having a Custom Infrastructure

Realize the value of custom infrastructure to provide easy accessibility, computing power, and interaction with quantum annealers



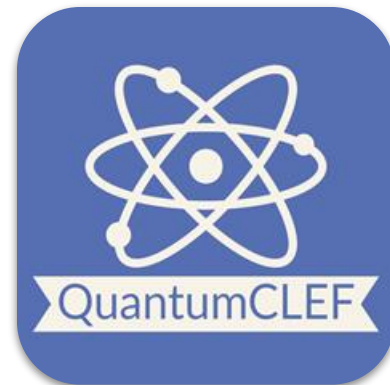
# The QuantumCLEF Lab

The QuantumCLEF Lab focuses on the use of Quantum Computing to solve problems related to the Information Access field.

In particular, in this year's edition, it focuses on solving 2 tasks: **Feature Selection** and **Clustering**.

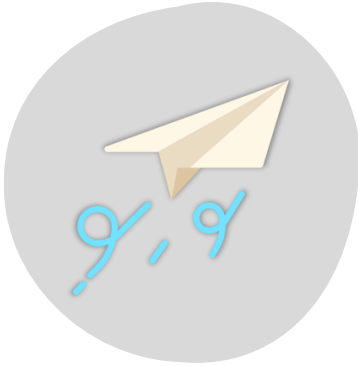
In the Feature Selection task we aim at retrieving the best subset of features that can be used to train a ranking model and a recommendation one.

On the other hand, in the Clustering task we aim at clustering documents in the form of embeddings to ease the browsing process.



# Why do we need an Infrastructure?

**Problem Submissions**



**Reproducibility / Fairness**



**Easy Team Accessibility**



# 1: Problem Submissions

D-Wave quantum annealers are hosted in datacentres scattered around the world. To interact with them, we need to call some APIs through which it is possible to send our problems over the internet and retrieve the corresponding solutions.

However, there are monthly quotas that enforce fairness among the utilizers of quantum annealers. In practice, this means that each user has a given **amount of time per month** to use quantum annealers.

Each user is given an **API-key** that must be sent along with the problems to identify the user itself and to check whether the user has still time left for the submission.

We cannot disclose our API-key and our system helps managing all the participants' submissions by assigning them custom API-keys and monitoring their quotas.



# 2: Reproducibility and Fairness

Most of the time, the measure that we only consider when developing systems is effectiveness (i.e., how good is our system in the given task).

In our case, we are also interested in **efficiency** since one of the promises of Quantum Computing is to boost efficiency while achieving similar or better effectiveness' values.

To ensure that all the participants will have the **same computing environment**, we make use of Kubernetes to provide them with the same computational resources.



# 3: Easy Team Accessibility

Participants do not need anything to develop their own algorithms apart from an **internet connection**.

In fact, our infrastructure is easily accessible from the internet. Each participating team has its own credentials that allow them to view their own personal data, monitor their quotas and access to their own workspace.

In this way, participants can use their own laptops exploiting the computational resources that are provided by our infrastructure.

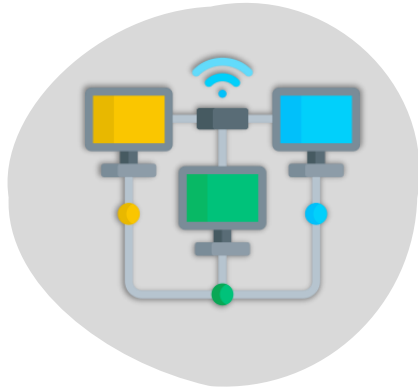




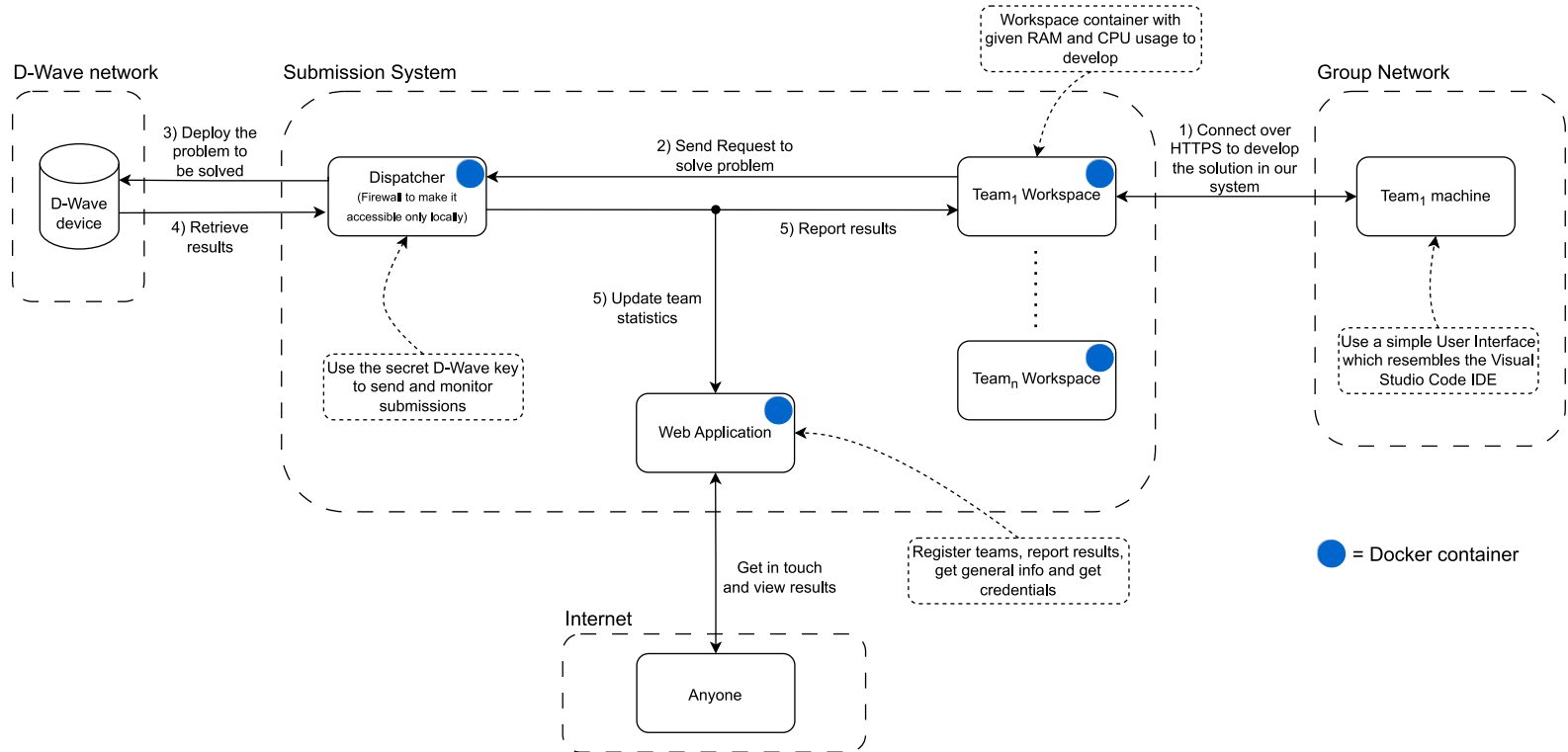


# How does the Infrastructure work?

Learn about the primary components of our infrastructure and discover how to utilize the workspaces to develop your own quantum algorithms



# A High Level Overview

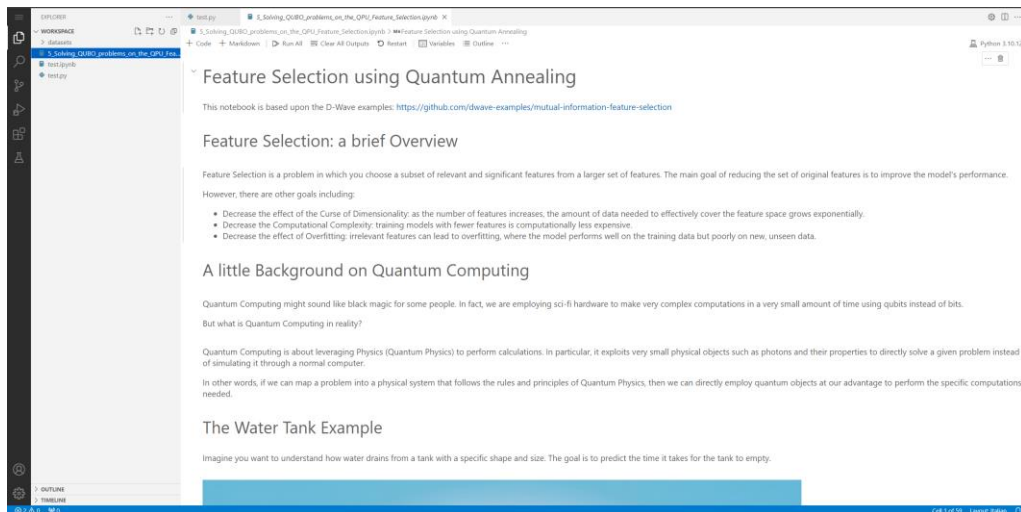


# The Participant's Perspective

Participants do not need to know how all the components of the infrastructure work. They are provided with a **user-friendly interface** that allows them to view their filesystem and to code directly from the browser. The interface is also customizable according to their own needs.

The interface resembles the one that is offered by the Visual Studio Code IDE, a free IDE that is commonly used because of its good properties such as lightweight and flexibility.

In the workspaces it is also possible to code using Jupyter Notebooks.

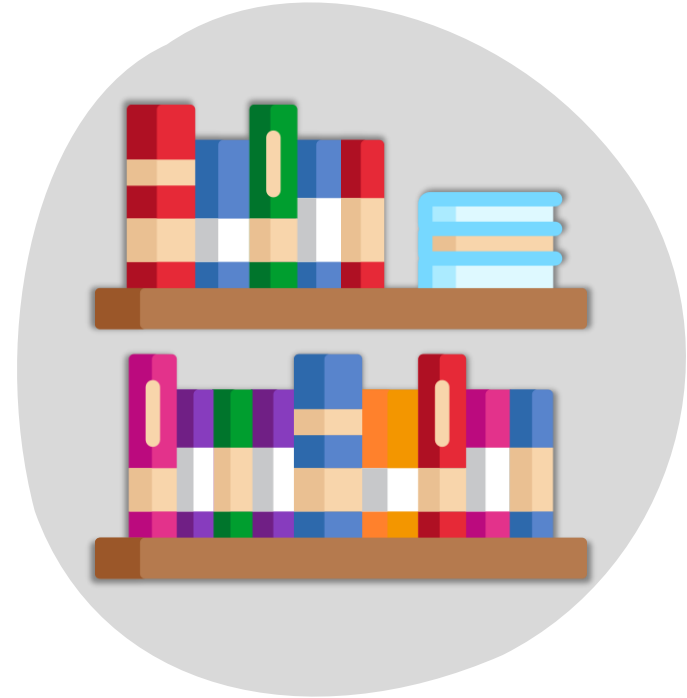


# D-Wave Official Libraries

The interaction with D-Wave quantum annealers happens by means of opportune **APIs**. D-Wave developed ad-hoc Python packages for the purpose of simplifying and standardizing the access to their quantum machines.

Participants will learn how to use the official [D-Wave packages](#), thus having the possibility to apply what they have learnt also for future projects.

There is only one constraint: we require participants to call a specific function when sending a submission. This allows us to monitor the submission and to store all its data in a database.



# An Example

## Using the original D-Wave packages

```
from dwave.samplers import SimulatedAnnealingSampler
from dimod import BinaryQuadraticModel

linear_terms = {'x1': -1, 'x2': -1}
quadratic_terms = {'x1', 'x2': 2}
bqm = BinaryQuadraticModel(linear_terms,
                           quadratic_terms,
                           0,
                           'BINARY')

sampler=SimulatedAnnealingSampler()

sampleset=sampler.sample(
    bqm,
    num_reads=100,
    label="1A - An example")
```

## Using the original D-Wave packages and our submit function

```
from qclef import qa_access as qa
from dwave.samplers import SimulatedAnnealingSampler
from dimod import BinaryQuadraticModel

linear_terms = {'x1' : -1, 'x2' : -1}
quadratic_terms = {'x1', 'x2' : 2}
bqm = BinariQuadraticModel(linear_terms,
                           quadratic_terms,
                           0,
                           'BINARY')

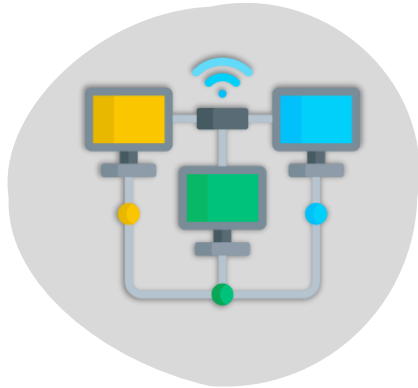
sampler = SimulatedAnnealingSampler()

sampleset = qa.submit(sampler,
                     SimulatedAnnealingSampler.sample,
                     bqm,
                     num_reads=100,
                     label="1A - An example")
```



# Accessing the Web Application

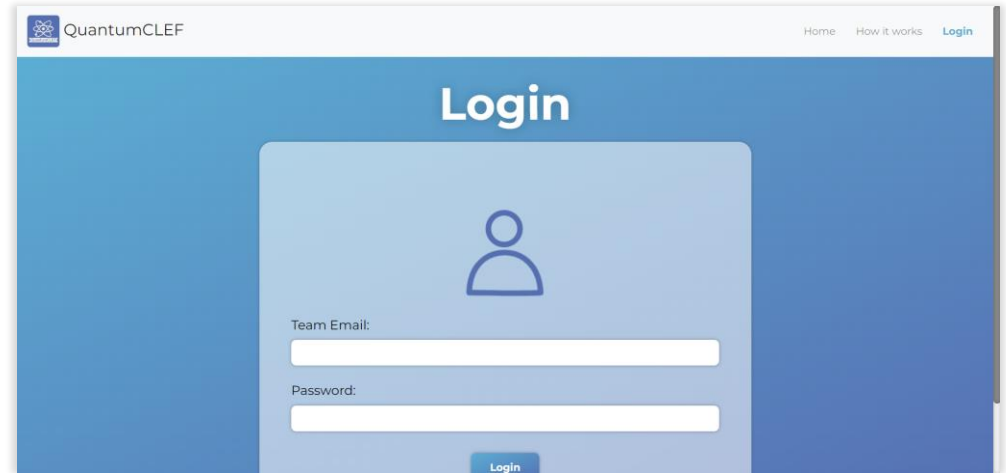
Learn how to access the Web Application and to the personal areas reserved for every participant



# Login

Each team participating in one of our tasks, will be given **credentials**. Those credentials allow to access a private area in which it is possible to view some statistics and personal data.

In this way, we provide teams a friendly interface that allow them to see their available time and to make some general statistics about the usage of the Quantum Computing Processors.



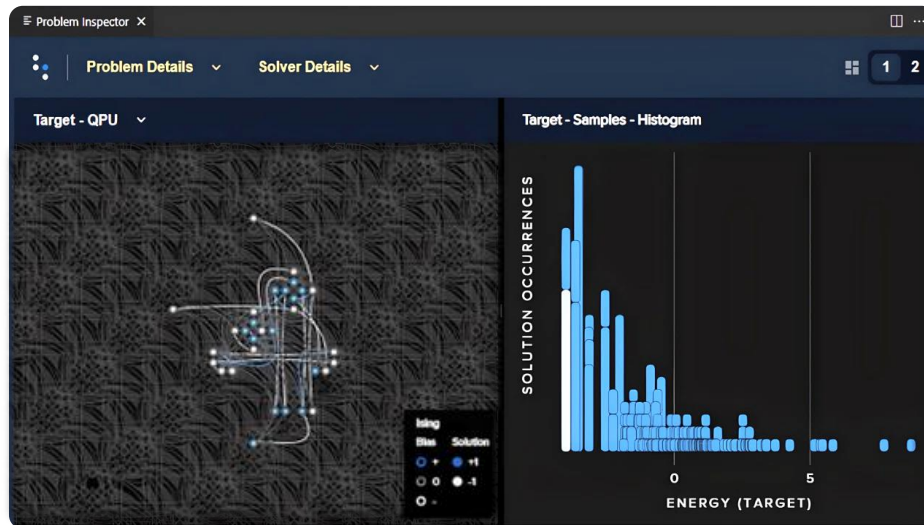
The image shows a screenshot of the QuantumCLEF login page. The page has a blue header with the QuantumCLEF logo on the left and navigation links for 'Home', 'How it works', and 'Login' on the right. The main content area is a light blue box with the word 'Login' in large white text at the top. Below the title is a blue icon of a person. Underneath the icon are two white input fields: the first is labeled 'Team Email:' and the second is labeled 'Password:'. At the bottom of the input fields is a blue button with the word 'Login' in white text.

# Inspect Problems

Teams will have access to the **Problem Inspector** tool provided by D-Wave.

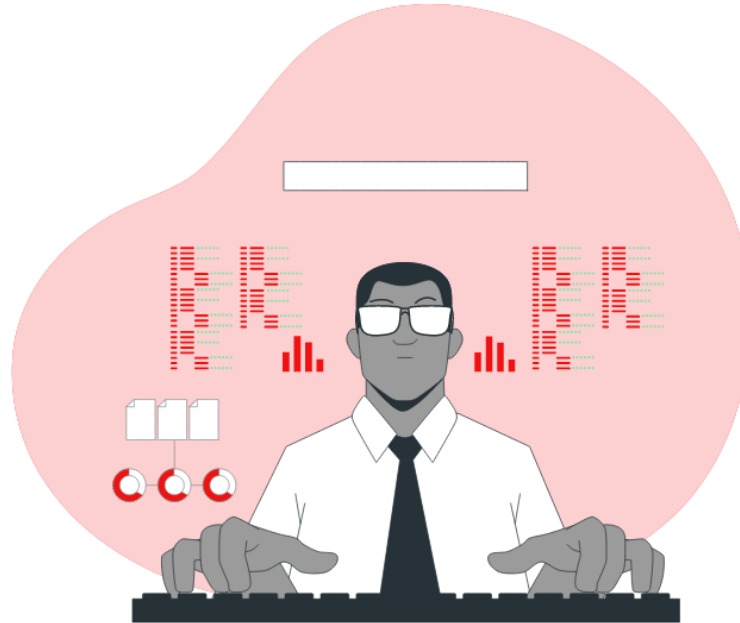
It allows to view how the problem has been physically embedded in the Quantum Processing Unit and to analyse how each of the qubits represent the variables of our problem.

We will see how it works more in details during the Hands-On part.





# A Tour of the Web Application



# References

All images/icons have been designed by Freepik and can be found in [Flaticon](#).

All other animations are provided by [Lottiefiles](#).

All resources are used only for educational purposes.

