

KNOWLEDGE TEST

SOLAR PANEL CLEANING SERVICE | 2024



COPYRIGHT / © 2024 SUNNY APP S. A. S – Sunnybotics INC®
Main Building: Km 2 route Neiva – Palermo.
Warehouse G-08 Palermo Industrial Park
Palermo – Huila

Phone: (+57) 3173679453

<https://sunnybotics.com>



Software engineer hiring test

Objective:

The objective is to demonstrate knowledge of programming and general web development by developing a comprehensive project using the syntax and programming languages essential to web programming, as well as demonstrating good programming practices and problem-solving skills.

Problem Development:

1. Realize a login for the entrance of the users and the form to be able to create a user. The interface must have 3 views, one for the user login already created, another with the form to add the user to the site, the form to add the user to the site, the form to add the user to the database, and one to show that the user has successfully logged in and the session can be successfully logged in and can close the session; for this you must be able to design the visual interface using html, css and javascript. The visual interface using html, css and javascript, use a language of your choice for backend management (python is recommended), and use a database of your choice (it can be SQL or noSQL). (it can be SQL or noSQL), the information display, the forms, the handling of fields, the forms, the handling of fields, the validation of data, the security of fields, and the use of good programming practices. and the use of good programming practices.

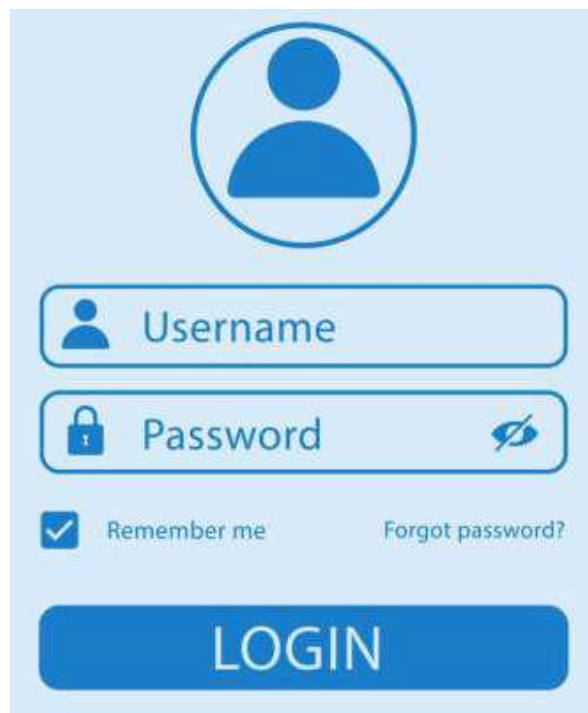
A user login form with a light blue background. At the top is a circular icon containing a blue silhouette of a person. Below this are two input fields: the first is labeled 'Username' with a small person icon on the left; the second is labeled 'Password' with a lock icon on the left and an eye icon on the right. Below the password field are two links: 'Remember me' with a checked checkbox and 'Forgot password?'. At the bottom is a large blue button with the word 'LOGIN' in white capital letters.

Figure 1. Representation of a user login.

- To create a graph that retrieves data from a table in a database to display its information. To display the information, the table content must have the following initial data.

DATE	VALUE
01/01/2024	29.1
02/01/2024	28.5
03/01/2024	28.1
04/01/2024	25.6
05/01/2024	24.8
06/01/2024	27.10
07/01/2024	29.9

In addition, there must be a button that allows you to create new information, which must be created as follows as follows:

- The date data must be an automatic data that generates a sequential date, for example, in this case, when pressing the button. For example, in this case, when pressing the button, the date would be "08/01/2024".
- The number data must be a random decimal value between 24.0 and 29.9, with one digit of decimal precision.

For example, pressing the Generate button 3 times may display data similar to the following this:

08/01/2024	24.7
09/01/2024	29.2
10/01/2024	26.8

The generated data should be automatically stored in the database and displayed in the graph either automatically or by pressing the graph either automatically or reloaded.

NOTE: Preferably, this graph should be within the view that is generated when the user logs in.





3. The code must be deployed and saved on GitHub, please attach the repository link at the end to present the test, comment the commits in a structured way, and attach a short video showing the result of the two tests, maximum 5 minutes.

Extra: You can deploy the application through a PaaS such as heroku, rail, render, or the one of your preference, it is not mandatory to deploy, but showing knowledge in this area can improve the test result.

Scoring Criteria: The ability to find the solution will be evaluated, and it will be checked for mainly functionality and correct handling of programming languages (Python, php, javascript) and syntax (HTML, CSS), rather than the resulting visual interface, the use of frameworks and structured methodologies will also be taken into account.

