### BANNARI AMMAN INSTITUTE OF TECHNOLOGY

An Autonomous Institution Affiliated to Anna University - Chennai • Approved by AICTE • Accredited by NAAC with 'A+' Grade SATHYAMANGALAM - 638401 ERODE DISTRICT TAMILNADU INDIA Ph: 04295-226000/221289 Fax: 04295-226666 Email: stayahead@bitsathy.ac.in Web: www.bitsathy.ac.in

TECHNICAL APPROVAL COMMITTEE

### GUIDE APPROVAL FORM

Date: \_\_ / \_\_ / 2023

Starting Date of Work				
Sl. No.	Student Name	Reg. No.	Role	Signature
1	SUNMATHI R	7376222IT267	Team Leader	
Applying for the work:		Paper/Project/Product (Product must be of commercialized quality)		
Title of Work		PASSWORD STRENGTH CHECKER		

(To be Filled by Faculty In charge)

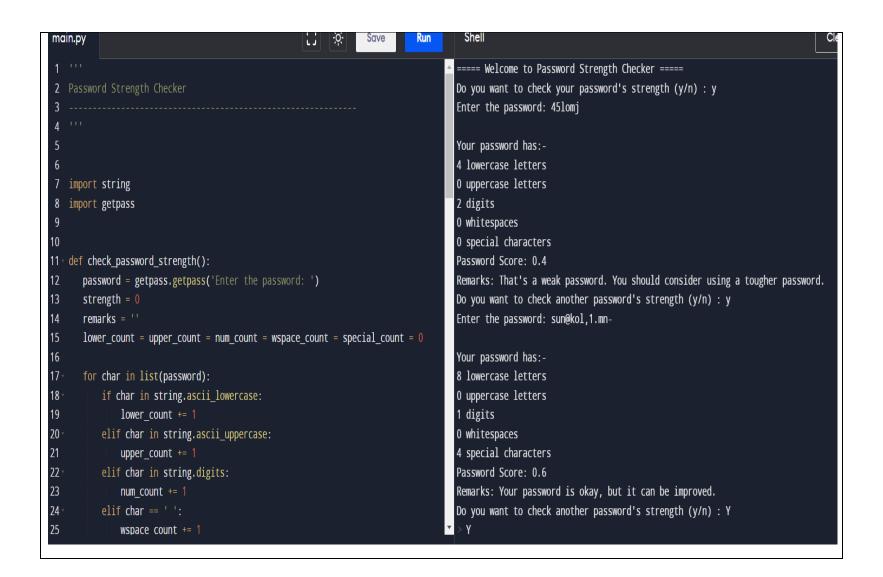
#### No. of students:

I acknowledge that I will act as a faculty in charge for the aforementioned students and guide them to complete the work by adopting the guidelines provided.

Lab Name:

(In case of Faculty belonging to any special lab)

Name and signature of the Faculty In charge with the date



### **Idea/Approach Details**

#### (Problem Statement), Proposed Solution for Password Strength Checker

The problem statement for a password strength checker is to determine whether a given password is strong or weak based on certain criteria. A strong password should satisfy the following conditions:

It should be at least 8 characters long.

It should contain at least one lowercase English character.

It should contain at least one uppercase English character.

*It should contain at least one special character from the set:* !@#\$%^&\*()-+

It should contain at least one digit.

A moderate password is one that satisfies the first three conditions and has a length of at least 6. Otherwise, the password is considered weak. The goal of a password strength checker is to evaluate a given password and determine whether it is strong, moderate, or weak based on these criteria. There are different approaches to creating a password strength checker, including using a machine learning model, regular expressions, or a simple algorithm

#### **METHODOLOGY:**

A password strength checker is a tool that evaluates the strength of a password based on certain criteria. The goal is to determine whether a given password is strong, moderate, or weak based on these criteria. There are different approaches to creating a password strength checker, including using a machine learning model, regular expressions, or a simple algorithm. The work involved in creating a password strength

checker includes defining the criteria for a strong password, creating a dataset or dictionary of known weak and strong passwords, writing a validation function, and ensuring the security of the tool. The output of this tool can be used to provide feedback to the user on whether the password is strong enough or needs improvement. Overall, a password strength checker is an important tool for ensuring the security of online accounts and protecting sensitive information from unauthorized access.

# The features/functions of the concern work of password strength checker:

To create a password strength checker in Python, the following steps can be taken:

Define the criteria for a strong password, such as length, complexity, and uniqueness.

Write a Python function to validate the password against the criteria for a strong password. This can be done using regular expressions to check if the password contains at least one uppercase letter, one lowercase letter, one digit, and one special character. The function should also check the length of the password.

Use the password validation function to check the strength of a given password and provide feedback to the user on whether the password is strong, moderate, or weak.

Optionally, use a machine learning model to train on a dataset of known weak and strong passwords to improve the accuracy of the password strength checker.

Ensure that the password strength checker is secure and does not transmit the password over the network.

Overall, the work involved in creating a password strength checker in Python includes defining the criteria for a strong password, writing a validation function, using the function to check the strength of a given password, and ensuring the security of the tool.

## TO SUCCESSFULLY COMPLETE THE PASSWORD STRENGTH CHECHER FOLLOWING TECHNOLOGIES AND FACILITIES ARE REQUIRED:

Regular expressions can be used to define a search pattern for validating a password. They can check if a password contains at least one uppercase letter, one lowercase letter, one digit, and one special character plate the prescribe work here.

Import the necessary Python libraries and the dataset needed for creating a password strength checker. The dataset should contain different combinations of letters and special symbols people use in passwords

JavaScript: Although not directly related to Python, JavaScript can be used to create a password strength checker for web applications. JavaScript libraries like the Password Strength Checker can be used to provide real-time feedback on the strength of a password as the user types

Signature of Faculty In Charge