TASK 6: Create a Strong Password and Evaluate its Strength

Objective: Understand what makes a password strong and test it against password strength tools
Tools: Online free password strength checkers (e.g, passwordmeter.com).
.

Prepared by: P. SUSMITHA,
Date: 28/10/2025.

Passwords

- **PASSWORD:** A password is a secret combination of characters used to verify a user's identity and protect data from unauthorized access. The strength of a password depends on its length, complexity, and unpredictability. Password strength is generally categorized into four levels — Weak, Good, Strong, and Very Strong

❖ **Weak Password:**

- ★ Usually short (less than 8 characters).
- ★ Contains only letters or only numbers.
- ★ Often includes personal details like names, birthdates, or simple sequences
- ★ Easily cracked through brute-force or dictionary attacks.
- ★ Examples: 12345, password, Test4.

❖ **Very weak Password:**

- ★ Usually short (less than 6characters).
- ★ Contains only one type of characters- either all letters or all numbers.
- ★ Lack of uppercase letters, symbols, or numbers combination.
- ★ Instantly cracked through brute-force or dictionary attacks.
- ★ Examples: 12345, abcde, Test.

❖ **Good Password:**

- ★ Meets minimum requirements (around 8–10 characters).
- ★ Contains a mix of letters, numbers, or symbols, but may still have predictable patterns.
- ★ Offers basic protection, but can be improved with more variety or length.
- ★ Example: Test123, 1234gh.

- **Strong Password:**
  - ★ Has 12 or more characters.
  - ★ Includes uppercase and lowercase letters, numbers, and special symbols.
  - ★ Avoids common words and personal data.
  - ★ Hard to guess manually and takes long to crack by automated tools.
  - ★ Example: Ramya123, Test@12, User#345

- **Very Strong Password**
  - ★ Long (12–16+ characters) and highly complex.
  - ★ Combines random sequences of uppercase, lowercase, numbers, and multiple symbols.
  - ★ Contains no dictionary words or predictable sequences.
  - ★ Extremely resistant to brute-force and dictionary attacks.
  - ★ Example: R54ya@123#6, &58Hk$104GFT#*6

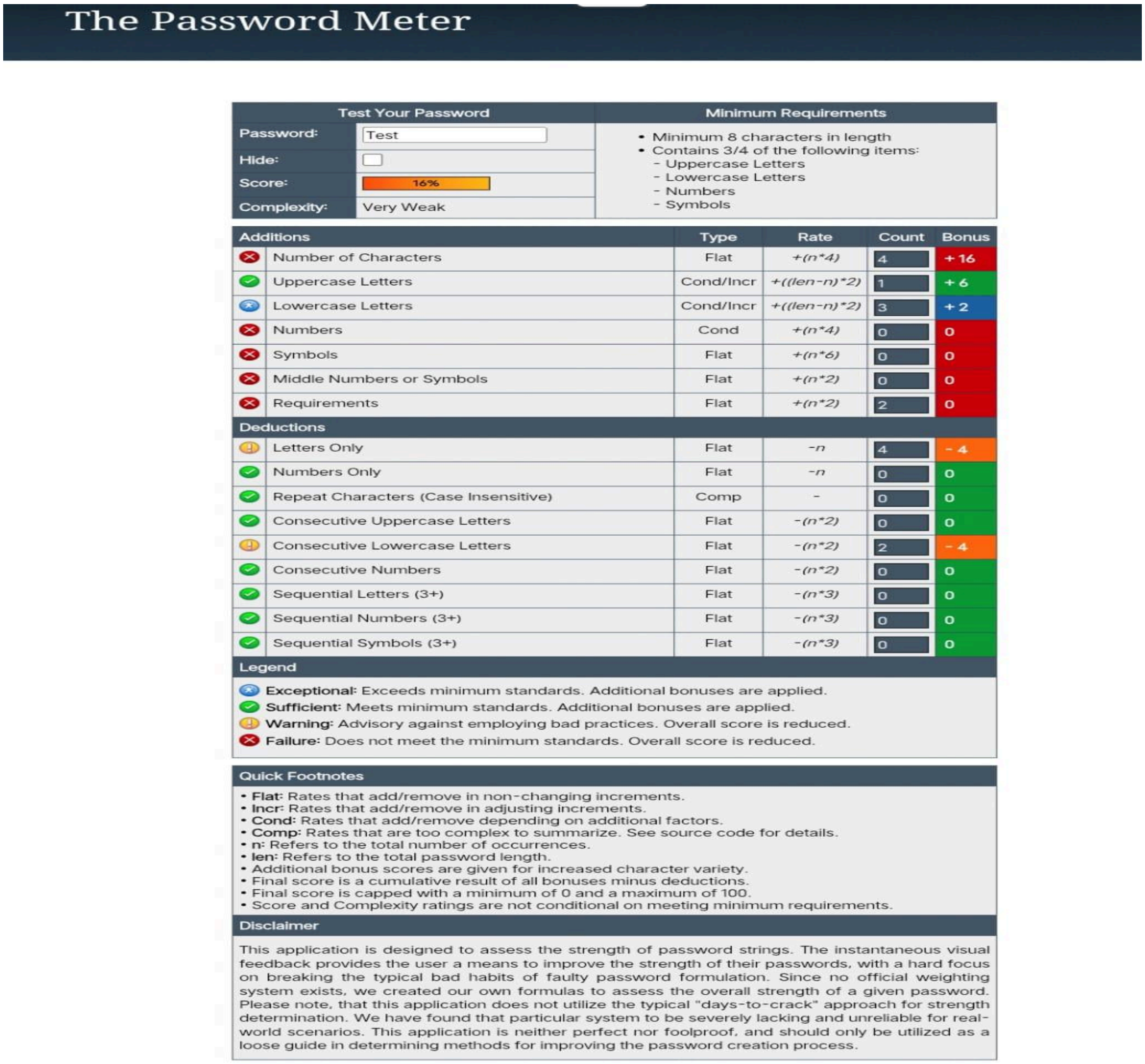
COMMON PASSWORD ATTACKS:

- **Brute Force Attack:**
  - ★ The attacker uses automated tools or software to try every possible combination of letters, numbers, and symbols until the correct password is found.
  - ★ This method is time-consuming but effective against short or simple passwords.
  - ★ Example: Trying “a”, “aa”, “aaa”, … until the correct one like “abc123” is found.
  - ★ Protection Tip: Use long, complex passwords (12+ characters) and multi-factor authentication (MFA) to prevent brute-force success.

- **Dictionary Attack:**
  - The attacker uses a predefined list of common passwords and words (like “password”, “welcome123”, “qwerty”) to guess the correct one.
  - It doesn’t try all combinations, only words that real users commonly choose.
  - Faster than brute force but relies on users using weak, predictable passwords.
  - Protection Tip: Avoid using common words or simple variations of them; include symbols, numbers, and uppercase letters.

- **Phishing Attack:**
  - The attacker tricks users into revealing their passwords by pretending to be a trusted source (like a bank or company).
  - This often happens through fake emails, websites, or messages that ask users to “verify” or “reset” their account.
  - Example: A fake email saying “Your account is locked, click here to log in.”
  - Protection Tip: Always check the sender’s email, URL, and never share passwords through links or emails.

- **Credential Stuffing Attack:**
  - Attackers use previously stolen username-password pairs from data breaches on one website and try them on other websites.
  - Since many people reuse the same password across accounts, this attack often succeeds.
  - Example: If your Netflix password was leaked, the attacker may try the same credentials on your Gmail or Facebook.
  - Protection Tip: Use unique passwords for each account and enable multi-factor authentication (MFA).



The Password Meter

Test Your Password		Minimum Requirements
Password:	Test	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	16%	
Complexity:	Very Weak	

	Additions	Type	Rate	Count	Bonus
✗	Number of Characters	Flat	+(n*4)	4	+ 16
✓	Uppercase Letters	Cond/Incr	+((len-n)*2)	1	+ 6
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	3	+ 2
✗	Numbers	Cond	+(n*4)	0	0
✗	Symbols	Flat	+(n*6)	0	0
✗	Middle Numbers or Symbols	Flat	+(n*2)	0	0
✗	Requirements	Flat	+(n*2)	2	0
	Deductions				
⚠	Letters Only	Flat	-n	4	- 4
✓	Numbers Only	Flat	-n	0	0
✓	Repeat Characters (Case Insensitive)	Comp	-	0	0
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
⚠	Consecutive Lowercase Letters	Flat	-(n*2)	2	- 4
✓	Consecutive Numbers	Flat	-(n*2)	0	0
✓	Sequential Letters (3+)	Flat	-(n*3)	0	0
✓	Sequential Numbers (3+)	Flat	-(n*3)	0	0
✓	Sequential Symbols (3+)	Flat	-(n*3)	0	0

Legend

★ **Exceptional:** Exceeds minimum standards. Additional bonuses are applied.
✓ **Sufficient:** Meets minimum standards. Additional bonuses are applied.
⚠ **Warning:** Advisory against employing bad practices. Overall score is reduced.
✗ **Failure:** Does not meet the minimum standards. Overall score is reduced.

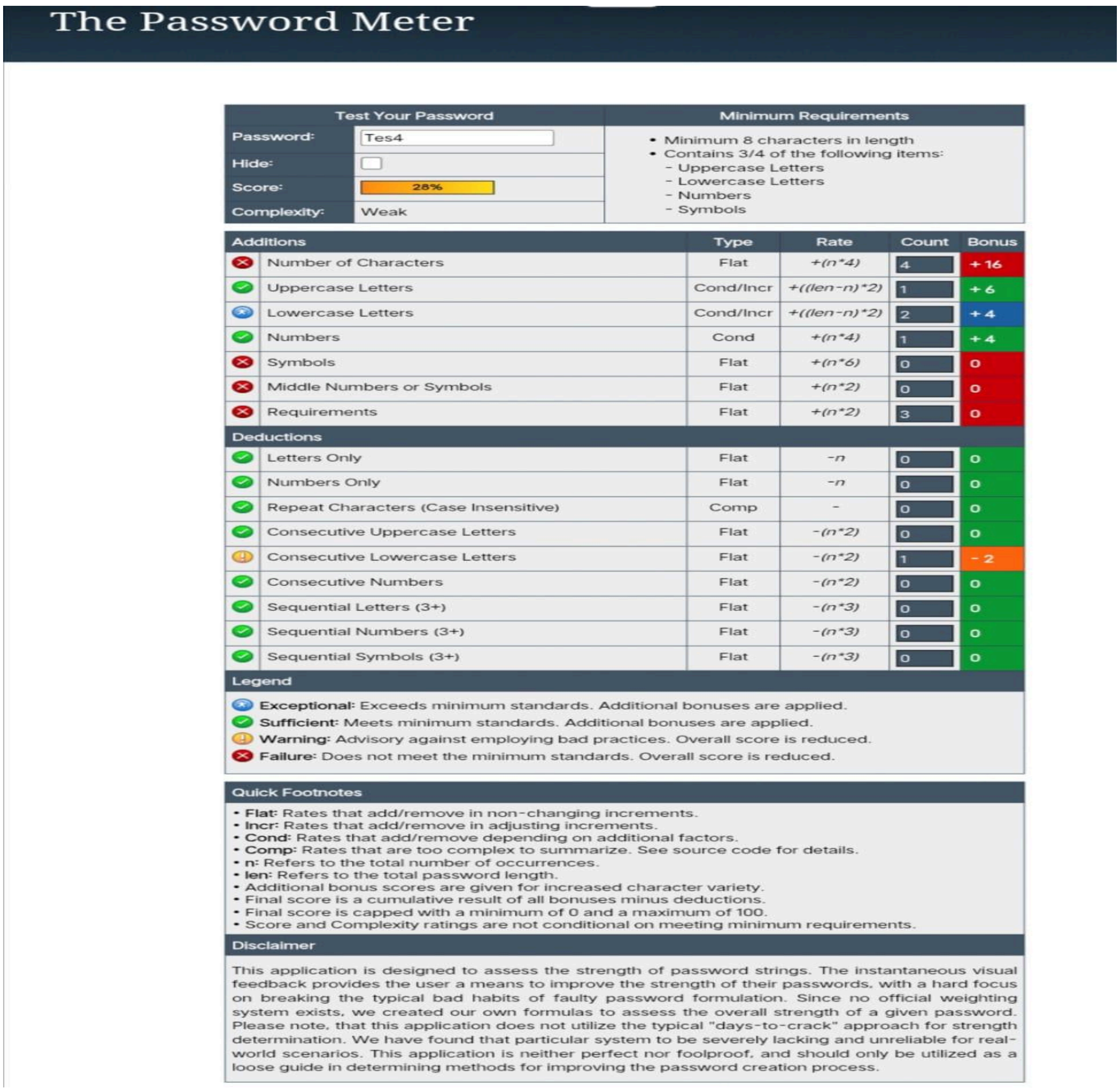
Quick Footnotes

- **Flat:** Rates that add/remove in non-changing increments.
- **Incr:** Rates that add/remove in adjusting increments.
- **Cond:** Rates that add/remove depending on additional factors.
- **Comp:** Rates that are too complex to summarize. See source code for details.
- **n:** Refers to the total number of occurrences.
- **len:** Refers to the total password length.
- Additional bonus scores are given for increased character variety.
- Final score is a cumulative result of all bonuses minus deductions.
- Final score is capped with a minimum of 0 and a maximum of 100.
- Score and Complexity ratings are not conditional on meeting minimum requirements.

Disclaimer

This application is designed to assess the strength of password strings. The instantaneous visual feedback provides the user a means to improve the strength of their passwords, with a hard focus on breaking the typical bad habits of faulty password formulation. Since no official weighting system exists, we created our own formulas to assess the overall strength of a given password. Please note, that this application does not utilize the typical "days-to-crack" approach for strength determination. We have found that particular system to be severely lacking and unreliable for real-world scenarios. This application is neither perfect nor foolproof, and should only be utilized as a loose guide in determining methods for improving the password creation process.

fig(1):Very weak password



The Password Meter

Test Your Password		Minimum Requirements
Password:	Tes4	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	28%	
Complexity:	Weak	

	Additions	Type	Rate	Count	Bonus
✖	Number of Characters	Flat	+(n*4)	4	+ 16
✔	Uppercase Letters	Cond/Incr	+((len-n)*2)	1	+ 6
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	2	+ 4
✔	Numbers	Cond	+(n*4)	1	+ 4
✖	Symbols	Flat	+(n*6)	0	0
✖	Middle Numbers or Symbols	Flat	+(n*2)	0	0
✖	Requirements	Flat	+(n*2)	3	0
	Deductions				
✔	Letters Only	Flat	-n	0	0
✔	Numbers Only	Flat	-n	0	0
✔	Repeat Characters (Case Insensitive)	Comp	-	0	0
✔	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
⚠	Consecutive Lowercase Letters	Flat	-(n*2)	1	- 2
✔	Consecutive Numbers	Flat	-(n*2)	0	0
✔	Sequential Letters (3+)	Flat	-(n*3)	0	0
✔	Sequential Numbers (3+)	Flat	-(n*3)	0	0
✔	Sequential Symbols (3+)	Flat	-(n*3)	0	0

Legend

★ **Exceptional:** Exceeds minimum standards. Additional bonuses are applied.
✔ **Sufficient:** Meets minimum standards. Additional bonuses are applied.
⚠ **Warning:** Advisory against employing bad practices. Overall score is reduced.
✖ **Failure:** Does not meet the minimum standards. Overall score is reduced.

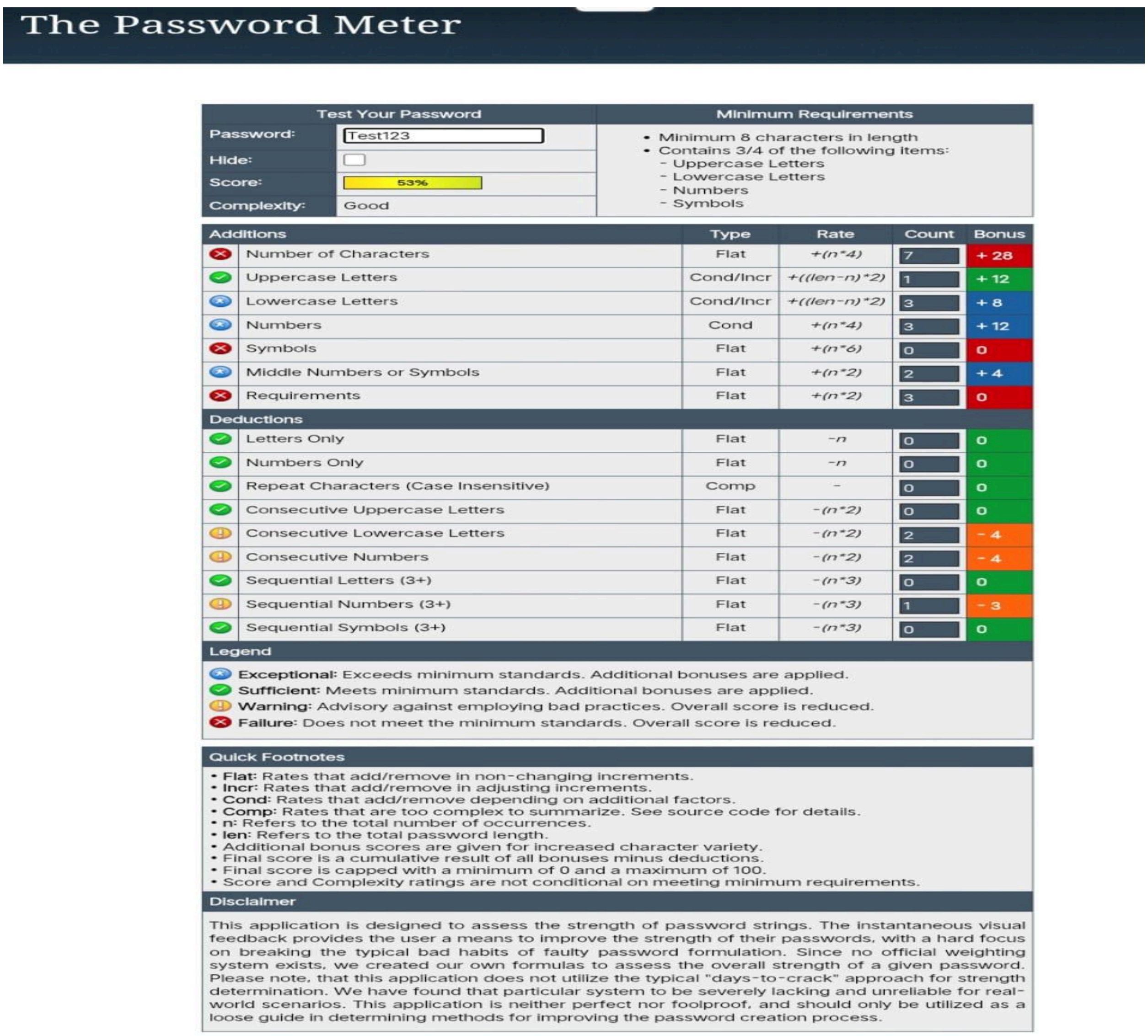
Quick Footnotes

- **Flat:** Rates that add/remove in non-changing increments.
- **Incr:** Rates that add/remove in adjusting increments.
- **Cond:** Rates that add/remove depending on additional factors.
- **Comp:** Rates that are too complex to summarize. See source code for details.
- **n:** Refers to the total number of occurrences.
- **len:** Refers to the total password length.
- Additional bonus scores are given for increased character variety.
- Final score is a cumulative result of all bonuses minus deductions.
- Final score is capped with a minimum of 0 and a maximum of 100.
- Score and Complexity ratings are not conditional on meeting minimum requirements.

Disclaimer

This application is designed to assess the strength of password strings. The instantaneous visual feedback provides the user a means to improve the strength of their passwords, with a hard focus on breaking the typical bad habits of faulty password formulation. Since no official weighting system exists, we created our own formulas to assess the overall strength of a given password. Please note, that this application does not utilize the typical "days-to-crack" approach for strength determination. We have found that particular system to be severely lacking and unreliable for real-world scenarios. This application is neither perfect nor foolproof, and should only be utilized as a loose guide in determining methods for improving the password creation process.

fig(2):Weak Password



The Password Meter

Test Your Password		Minimum Requirements
Password:	Test123	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	53%	
Complexity:	Good	

	Additions	Type	Rate	Count	Bonus
✗	Number of Characters	Flat	+(n*4)	7	+ 28
✓	Uppercase Letters	Cond/Incr	+((len-n)*2)	1	+ 12
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	3	+ 8
★	Numbers	Cond	+(n*4)	3	+ 12
✗	Symbols	Flat	+(n*6)	0	0
★	Middle Numbers or Symbols	Flat	+(n*2)	2	+ 4
✗	Requirements	Flat	+(n*2)	3	0
	Deductions				
✓	Letters Only	Flat	-n	0	0
✓	Numbers Only	Flat	-n	0	0
✓	Repeat Characters (Case Insensitive)	Comp	-	0	0
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
!	Consecutive Lowercase Letters	Flat	-(n*2)	2	- 4
!	Consecutive Numbers	Flat	-(n*2)	2	- 4
✓	Sequential Letters (3+)	Flat	-(n*3)	0	0
!	Sequential Numbers (3+)	Flat	-(n*3)	1	- 3
✓	Sequential Symbols (3+)	Flat	-(n*3)	0	0

Legend

★ **Exceptional:** Exceeds minimum standards. Additional bonuses are applied.
✓ **Sufficient:** Meets minimum standards. Additional bonuses are applied.
! **Warning:** Advisory against employing bad practices. Overall score is reduced.
✗ **Failure:** Does not meet the minimum standards. Overall score is reduced.

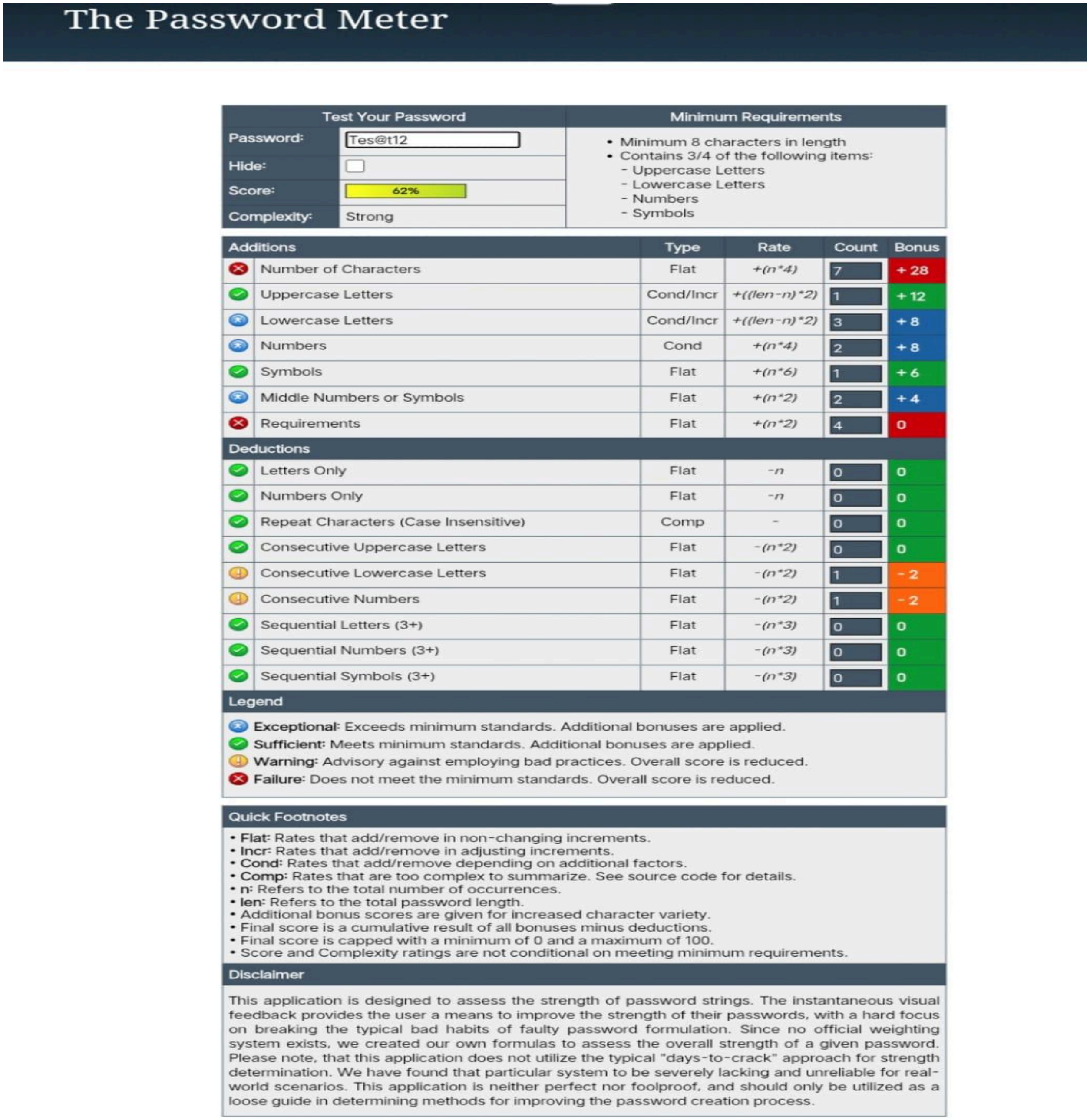
Quick Footnotes

- **Flat:** Rates that add/remove in non-changing increments.
- **Incr:** Rates that add/remove in adjusting increments.
- **Cond:** Rates that add/remove depending on additional factors.
- **Comp:** Rates that are too complex to summarize. See source code for details.
- **n:** Refers to the total number of occurrences.
- **len:** Refers to the total password length.
- Additional bonus scores are given for increased character variety.
- Final score is a cumulative result of all bonuses minus deductions.
- Final score is capped with a minimum of 0 and a maximum of 100.
- Score and Complexity ratings are not conditional on meeting minimum requirements.

Disclaimer

This application is designed to assess the strength of password strings. The instantaneous visual feedback provides the user a means to improve the strength of their passwords, with a hard focus on breaking the typical bad habits of faulty password formulation. Since no official weighting system exists, we created our own formulas to assess the overall strength of a given password. Please note, that this application does not utilize the typical "days-to-crack" approach for strength determination. We have found that particular system to be severely lacking and unreliable for real-world scenarios. This application is neither perfect nor foolproof, and should only be utilized as a loose guide in determining methods for improving the password creation process.

fig(3):Good Password



The Password Meter

Test Your Password	
Password:	Tes@t12
Hide:	□
Score:	62%
Complexity:	Strong

Minimum Requirements

- Minimum 8 characters in length
- Contains 3/4 of the following items:
  - Uppercase Letters
  - Lowercase Letters
  - Numbers
  - Symbols

	Additions	Type	Rate	Count	Bonus
✖	Number of Characters	Flat	+(n*4)	7	+ 28
✔	Uppercase Letters	Cond/Incr	+((len-n)*2)	1	+ 12
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	3	+ 8
★	Numbers	Cond	+(n*4)	2	+ 8
✔	Symbols	Flat	+(n*6)	1	+ 6
★	Middle Numbers or Symbols	Flat	+(n*2)	2	+ 4
✖	Requirements	Flat	+(n*2)	4	0
	Deductions				
✔	Letters Only	Flat	-n	0	0
✔	Numbers Only	Flat	-n	0	0
✔	Repeat Characters (Case Insensitive)	Comp	-	0	0
✔	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
⚠	Consecutive Lowercase Letters	Flat	-(n*2)	1	- 2
⚠	Consecutive Numbers	Flat	-(n*2)	1	- 2
✔	Sequential Letters (3+)	Flat	-(n*3)	0	0
✔	Sequential Numbers (3+)	Flat	-(n*3)	0	0
✔	Sequential Symbols (3+)	Flat	-(n*3)	0	0

Legend

★ **Exceptional:** Exceeds minimum standards. Additional bonuses are applied.
✔ **Sufficient:** Meets minimum standards. Additional bonuses are applied.
⚠ **Warning:** Advisory against employing bad practices. Overall score is reduced.
✖ **Failure:** Does not meet the minimum standards. Overall score is reduced.

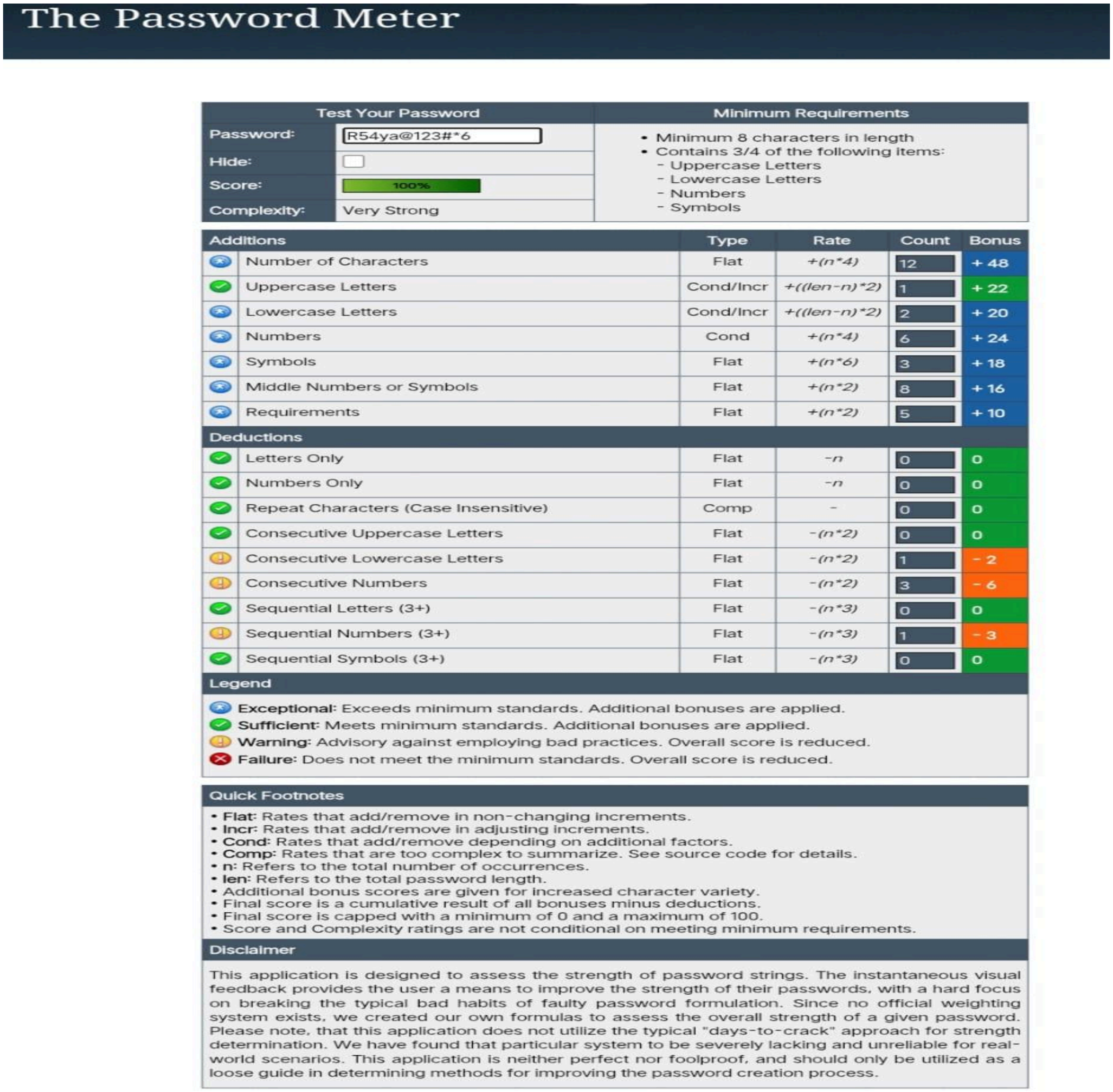
Quick Footnotes

- **Flat:** Rates that add/remove in non-changing increments.
- **Incr:** Rates that add/remove in adjusting increments.
- **Cond:** Rates that add/remove depending on additional factors.
- **Comp:** Rates that are too complex to summarize. See source code for details.
- **n:** Refers to the total number of occurrences.
- **len:** Refers to the total password length.
- Additional bonus scores are given for increased character variety.
- Final score is a cumulative result of all bonuses minus deductions.
- Final score is capped with a minimum of 0 and a maximum of 100.
- Score and Complexity ratings are not conditional on meeting minimum requirements.

Disclaimer

This application is designed to assess the strength of password strings. The instantaneous visual feedback provides the user a means to improve the strength of their passwords, with a hard focus on breaking the typical bad habits of faulty password formulation. Since no official weighting system exists, we created our own formulas to assess the overall strength of a given password. Please note, that this application does not utilize the typical "days-to-crack" approach for strength determination. We have found that particular system to be severely lacking and unreliable for real-world scenarios. This application is neither perfect nor foolproof, and should only be utilized as a loose guide in determining methods for improving the password creation process.

fig(4): Strong Password



The Password Meter

Test Your Password		Minimum Requirements
Password:	R54ya@123#*6	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	100%	
Complexity:	Very Strong	

	Additions	Type	Rate	Count	Bonus
★	Number of Characters	Flat	+(n*4)	12	+ 48
✓	Uppercase Letters	Cond/Incr	+((len-n)*2)	1	+ 22
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	2	+ 20
★	Numbers	Cond	+(n*4)	6	+ 24
★	Symbols	Flat	+(n*6)	3	+ 18
★	Middle Numbers or Symbols	Flat	+(n*2)	8	+ 16
★	Requirements	Flat	+(n*2)	5	+ 10
	Deductions				
✓	Letters Only	Flat	-n	0	0
✓	Numbers Only	Flat	-n	0	0
✓	Repeat Characters (Case Insensitive)	Comp	-	0	0
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
!	Consecutive Lowercase Letters	Flat	-(n*2)	1	- 2
!	Consecutive Numbers	Flat	-(n*2)	3	- 6
✓	Sequential Letters (3+)	Flat	-(n*3)	0	0
!	Sequential Numbers (3+)	Flat	-(n*3)	1	- 3
✓	Sequential Symbols (3+)	Flat	-(n*3)	0	0

Legend

★ **Exceptional:** Exceeds minimum standards. Additional bonuses are applied.
✓ **Sufficient:** Meets minimum standards. Additional bonuses are applied.
! **Warning:** Advisory against employing bad practices. Overall score is reduced.
✗ **Failure:** Does not meet the minimum standards. Overall score is reduced.

Quick Footnotes

- **Flat:** Rates that add/remove in non-changing increments.
- **Incr:** Rates that add/remove in adjusting increments.
- **Cond:** Rates that add/remove depending on additional factors.
- **Comp:** Rates that are too complex to summarize. See source code for details.
- **n:** Refers to the total number of occurrences.
- **len:** Refers to the total password length.
- Additional bonus scores are given for increased character variety.
- Final score is a cumulative result of all bonuses minus deductions.
- Final score is capped with a minimum of 0 and a maximum of 100.
- Score and Complexity ratings are not conditional on meeting minimum requirements.

Disclaimer

This application is designed to assess the strength of password strings. The instantaneous visual feedback provides the user a means to improve the strength of their passwords, with a hard focus on breaking the typical bad habits of faulty password formulation. Since no official weighting system exists, we created our own formulas to assess the overall strength of a given password. Please note, that this application does not utilize the typical "days-to-crack" approach for strength determination. We have found that particular system to be severely lacking and unreliable for real-world scenarios. This application is neither perfect nor foolproof, and should only be utilized as a loose guide in determining methods for improving the password creation process.

fig(5): Very Strong Password