

GitHub username

GitHub username — alphanumeric + single hyphen, no leading/trailing hyphen.

```
^[A-Za-z0-9](?:[A-Za-z0-9]|-(?=[A-Za-z0-9])){0,38}$
```

Token-by-token breakdown

Token	Meaning
^	start of string (or line in multiline mode)
[A-Za-z0-9]	any of: uppercase letters, lowercase letters, digits
(?:	start of a non-capturing group
[A-Za-z0-9]	any of: uppercase letters, lowercase letters, digits
	OR — try the alternative
-	literal text “-”
(?=[A-Za-z0-9]	start of a positive lookahead — followed by
[A-Za-z0-9]	any of: uppercase letters, lowercase letters, digits
)	end of group
{0,38}	end of group, between 0 and 38 times
\$	end of string (or line in multiline mode)

Quick usage in 7 languages

JavaScript	<pre>const re = /^[A-Za-z0-9](?:[A-Za-z0-9] -(?=[A-Za-z0-9])){0,38}\$/; re.test(value);</pre>
Python	<pre>import re re.match(r"^[A-Za-z0-9](?:[A-Za-z0-9] -(?=[A-Za-z0-9])){0,38}\$", value)</pre>
Java	<pre>Pattern.compile("^[A-Za-z0-9](?:[A-Za-z0-9] -(?=[A-Za-z0-9])){0,38}\$").matcher(value).m atches();</pre>
C# / .NET	<pre>Regex.IsMatch(value, @"^[A-Za-z0-9](?:[A-Za-z0-9] -(?=[A-Za-z0-9])){0,38}\$");</pre>
Go	<pre>regexp.MustCompile(`^[A-Za-z0-9](?:[A-Za-z0-9] -(?=[A-Za-z0-9])){0,38}\$`).MatchString(v alue)</pre>
Ruby	<pre>/^[A-Za-z0-9](?:[A-Za-z0-9] -(?=[A-Za-z0-9])){0,38}\$/ .match?(value)</pre>
PHP	<pre>preg_match('~^[A-Za-z0-9](?:[A-Za-z0-9] -(?=[A-Za-z0-9])){0,38}\$~', \$value);</pre>

Common pitfalls

- **Anchored to the whole string.** This pattern uses ^ and \$, so it requires the entire input to match. To find it inside a longer text, drop the anchors and use the global (g) flag.
- **ASCII letters only.** [a-zA-Z] excludes accented and non-Latin letters (é, ü, ß, ñ, and non-Latin scripts). For international input use Unicode properties like \p{L} with the u flag.
- **Validate beyond format.** Matching the format doesn't guarantee the value is real. Confirm the github username against a source of truth (database, API, or checksum) where it matters.