My minimum requirements for a password replacement system:

Memoryless	Users should not have to memorize any secrets
Scalable	Scalable to thousands of apps
Secure	At least as secure as passwords

Additional requirements if token-based:

Loss-Resistant	If token lost, user can regain access to services
Theft-Resistant	If token stolen, thief can't impersonate user

Benefits promised by Pico in addition to the above:

(Usability-related)

Works-For-All	Works for *all* credentials, not just web passwords
From-Anywhere	The user can authenticate from any client
No-Search	The user doesn't have to select the correct credentials
No-Typing	The user no longer has to *type* the damn password
Continuous	Authentication is continuous, not just at session start

(Security-related)

No-Weak	The user cannot choose a weak password
No-Reuse	The user cannot reuse credentials with different apps
No-Phishing	Phishing (app impersonation) is impossible
No-Eavesdropping	Network eavesdropping is impossible
No-Keylogging	Keylogging is impossible
No-Surfing	Shoulder surfing is impossible
No-Linkage	Different credentials from same user can't be linked

Additional desirable properties that are not goals for Pico:

No-Cost	As cheap to deploy as passwords
No-App-Changes	Deployable without changes to existing apps
No-Cli-Changes	Deployable without changes to existing clients

Also worth considering for a fair comparison:

Implemented	This system was built, rather than just described
Open	The code and design are available as open-source
Widely-Used	Has been used by over a million individuals
No-Carry	Does not require the user to carry anything
No-TTP	No reliance on a TTP who knows your credentials