

■ MiniCycle Modularization Guide v3.0 — Cheat Sheet Appendix

■ Pattern Selection (Design-Time)

Module Type	Best Pattern	Key Indicators
DOM Utilities / Formatters	■ Static Utility	Pure functions, no state
Notifications / Simple Modals	■ Simple Instance	Self-contained, graceful degradation
Stats Panel / Widgets	■■ Resilient Constructor	Complex UI, can degrade gracefully
Data Processing / Core Logic	■ Strict Injection	Critical logic, must fail fast

■ Error Handling (Run-Time)

Pattern	Error Strategy	Logging	When to Throw
■ Static Utility	Return safe defaults	console.warn	Never
■ Simple Instance	Graceful fallback	console.warn + fallback	Never
■■ Resilient Constructor	Degrade gracefully	console.warn + user notification	Never
■ Strict Injection	Fail fast	throw + showNotification('error')	Missing deps

■ Lifecycle Standards for UI Modules

- Simple Instance: constructor(), destroy()
- Resilient Constructor: constructor(deps), init(), update(), destroy()
- Strict Injection: configure with setDependencies() before use
- Static Utility: no lifecycle — pure functions only

■ Initialization Order

- 1) Static Utilities → no config
- 2) Simple Instances → ready after import
- 3) Strict Injection modules → configure before use
- 4) Resilient UI Components → inject what's available, degrade gracefully