

Enterprise Angular Migration Guide: PrimeNG v9 → v15 and Angular v15 → v20

Section A: PrimeNG v9 to v15 Migration

1. Class Prefix Changes from ui- to p- (and Custom Prefix in v13+) PrimeNG 10 renamed CSS class prefixes from ui- to p-. Developers must update custom styles accordingly. PrimeNG v13+ allows customizing CSS variable prefixes (for design tokens), not structural classes. Migration requires audit of selectors and a comprehensive find/replace of ui- to p-.
2. DOM Class Changes Mapping - Button: ui-button → p-button, ui-state-disabled → p-disabled - DataTable: ui-datatable → p-datatable, ui-state-highlight → p-highlight - Dialog: ui-dialog → p-dialog, ui-widget-overlay → p-component-overlay - Panel: ui-panel → p-panel - Fieldset: ui-fieldset → p-fieldset - Form Inputs: ui-inputtext → p-inputtext, ui-dropdown → p-dropdown - Checkbox: ui-chkbox → p-checkbox - Radiobutton: ui-radiobutton → p-radiobutton - Menu: ui-menu → p-menu ...and more for all major components.
3. Handling Legacy Styles and Custom Themes Legacy SASS themes (Nova, Luna) vs new CSS variables themes. Update _variables.scss overrides to use CSS variables (--p-*) or compile SASS themes from v15. Ensure PrimeFlex CSS is included for layout utilities.
4. Non-Prefixed Class Changes & Pitfalls State classes: ui-state-highlight → p-highlight, ui-helper-clearfix → p-clearfix. Content projection: p-header → . Deprecated components: removed in v12.
5. Sandbox Testing customPrefix PrimeNG's customPrefix for CSS variables tested in sandbox. Structural class prefix customization not officially supported. Migration via find/replace recommended.
6. Tooling for Class Renaming Regex scripts, VSCode search/replace, custom codemods (jscodeshift), Angular schematics (if available), lint rules to catch ui- classes, DOM validation via console scripts.
7. Test Impact & QA Strategy Update unit/E2E selectors (ui- → p-), add data-testid attributes, visual regression testing, cross-browser/responsive testing, accessibility audits, incremental verification phase.
8. PrimeNG v10–15 Highlights - v10 major overhaul (class prefix, new themes) - v11 minor breaking (Calendar locale removal, FilterService) - v12 DataTable scroll rewrite - v13 Ivy only, drop IE11, default Lara theme - v14 minor compatibility - v15 stabilization, PrimeFlex integration
9. Blockers & Edge Cases Premium templates, custom themes, integration with other libraries, legacy grid classes, service worker config, SSR/hydration, performance.
10. Step-by-Step Plan Pre-checks: audit usage, choose theme, align dependencies. Execution: branch, update package.json, install PrimeFlex, update imports, rename styles, manual fixes, review, tests, QA, deploy.

Section B: Angular v15 to v20 Upgrade Path

General Strategy: sequential upgrades (15→16→17→18→19→20) using ng update, test after each, update TS/Node, use CLI schematics.

B1. 15→16: remove ngcc, TS4.9+, Node16/18, opt-in Signals, update tsconfig(useDefineForClassFields), ng update. B2. 16→17: TS5.2+, Node18, new control flow syntax (@if/@for), esbuild builder default (ng update use-application-builder). B3. 17→18: TS5.4+, Node18/20, signals stable, zoneless preview, migrate templates to new syntax, drop deprecated APIs. B4. 18→19: TS5.6+, Node18/20, signals & standalone focus, SSR incremental hydration, prepare for future deprecations. B5. 19→20: TS5.7+, Node20, no major breakings, deprecate *ngIf/*ngFor, signals & resources stable, devtools enhancements.

Appendices: change logs, code diffs, mapping tables, sample scripts.

Section C: Appendices and Linked Documentation will be appended here in subsequent iterations.