

changelog.md for @ecobridge.xyz/catalog

2026-01-16 - 3.40.1 - fix(deps)

bump @design.estate/dees-catalog to ^3.37.0 and @types/node to ^25.0.9

- Upgrade @design.estate/dees-catalog from ^3.36.0 to ^3.37.0
- Upgrade @types/node from ^25.0.7 to ^25.0.9

2026-01-13 - 3.40.0 - feat(eco-view-containers)

add eco-view-containers demo and export; update remove button to destructive; bump devDependencies

- Add ts_web/views/eco-view-containers/eco-view-containers.demo.ts with sample container and log data and a demo template
- Export eco-view-containers from ts_web/views/eco-view-containers/index.ts and add export to ts_web/views/index.ts
- Change remove button in eco-view-containers to use .type='destructive' (removed .status='error')
- Bump devDependencies: @git.zone/tsbuild ^4.1.0 -> ^4.1.2 and @types/node ^25.0.6 -> ^25.0.7

2026-01-12 - 3.39.1 - fix(deps)

bump @design.estate/dees-catalog to ^3.36.0

- Updated dependency @design.estate/dees-catalog from ^3.35.0 to ^3.36.0 in package.json

2026-01-12 - 3.39.0 - feat(eco-view-system)

add memory usage history, process metrics, and top processes display with loading fallback

- Add memoryUsageHistory state and update it in setMetrics to track memory usage trend
- Introduce process state: processTotal, processRunning, processSleeping, processBlocked, and topProcesses
- Extend setProcesses signature to accept blocked and a typed process list and store values in state
- Use reactive state values in UI tiles (memory trend, process tiles) instead of hardcoded placeholders
- Replace 'Threads' tile with 'Blocked' tile and update its icon and color
- Render top processes from state and show a 'Loading...' fallback when the list is empty

2026-01-12 - 3.38.0 - feat(eco-view-system)

add extended system metrics and display formatted total network usage in eco system view

- Added numerous new observable properties for richer system telemetry: cpuPhysicalCores, cpuSpeedMax, memoryAvailable, memoryCached, memoryBuffers, swapTotal, swapUsed, diskTotal, diskUsed, diskFree, networkRxSec, networkTxSec, networkRxTotal, networkTxTotal, distro, and coreLoads.
- Reworked network/usage UI: removed several temporary network/latency cards and replaced them with Total Downloaded/Uploaded cards that use formatBytes(...) and show totals since boot.
- Bumped dependency @design.estate/dees-catalog from ^3.34.1 to ^3.35.0.

2026-01-12 - 3.37.0 - feat(elements)

add eco-provider-frame and dataprovider interfaces; improve virtual keyboard interactions; add demos, exports and bump dev dependencies

- Add new eco-provider-frame web component (iframe-based provider protocol) with messaging API (sendData, requestData, ping) and events for provider-ready, features-changed, provider-response, provider-error
- Introduce comprehensive dataprovider TypeScript interfaces (ts_web/elements/interfaces/dataprovider.ts) describing messages, payloads, provider state and store schema
- Improve virtual keyboard and launcher interactions: add tabIndex, preventDefault on mousedown, stopPropagation on pointer handlers, remove key focus outline to fix touch/mouse behavior
- Add demos and new exports for provider-frame, browser and scan views/components (demo files and index exports)
- Update build/dev config: add npmextra.json tsbundle config and bump dependencies: @design.estate/dees-catalog -> ^3.34.1, @git.zone/tsbuild -> ^4.1.0, @git.zone/tsbundle -> ^2.8.1, @types/node -> ^25.0.6

2026-01-12 - 3.36.1 - fix()

no changes

- No files changed in this commit
- No release required

2026-01-12 - 3.36.0 - feat(eco-view-peripherals)

encapsulate networkRanges as internal @state and add public getter/setter methods

- Replaced @property({ type: Array }) with @state() for networkRanges to treat it as internal reactive state.
- Added setNetworkRanges(ranges: INetworkRange[]) and getNetworkRanges(): INetworkRange[] to provide a controlled public API for updating and reading network ranges.
- Methods use array spread to return/copy arrays, preventing external mutation of internal state.
- Added JSDoc comments for the new methods.

2026-01-12 - 3.35.0 - feat(peripherals)

Add peripherals settings panel with network range management, network scanning, and manual device probe; update peripheral types and adjust UI/styling; overhaul README with expanded docs, quick start, and updated company/contact information

- Add 'settings' peripheral category and new INetworkRange type; change IPeripheralDevice.type to exclude 'settings' to separate settings metadata from actual devices
- Implement network range management UI: add/remove ranges, display network list, scan networks (with isScanning state) and related controls
- Add "Add Device by IP" probe workflow and input UI to allow manual device discovery by IP
- Introduce new styles and header/button UI (settings-section, icon-button, network-list/network-item, etc.) for eco-view-peripherals
- Adjust categoryLabels type to exclude 'settings' to keep device grouping consistent
- Major README rewrite: expanded features list, installation examples (pnpm/npm), quick start code samples, developer commands, and updated company information to Task Venture Capital GmbH with new contact email

2026-01-12 - 3.34.4 - fix(catalog)

no changes (empty diff) — no files modified

- No files were modified; git diff is empty
- No dependency or source changes detected
- No version bump required

2026-01-12 - 3.34.3 - fix(catalog)

no changes detected

- Git diff contains no changes; no files modified
- No version bump required

2026-01-06 - 3.34.2 - fix(applauncher)

throttle inactivity timer resets in menus, optimize sound slider updates, and adjust keyboard layout/keys

- Add `lastActivityTime` and throttle `resetInactivityTimer` to only reset if 5+ seconds have passed in battery, sound, and wifi menus to reduce frequent resets from continuous input.
- Remove `@mousemove` listener in menu containers and rely on `mousedown` + throttled resets to lower event noise.
- Debounce slider `mousemove` handling in sound menu using `requestAnimationFrame` and `pendingPercentage` to batch `setVolume` calls and cancel RAF on `mouseup`, preventing excessive volume updates.
- Add up/down arrow keys to virtual keyboard, reduce space key width from 4 to 3, and add `.key.wide-3` CSS class to support the new sizing.

2026-01-06 - 3.34.1 - fix(elements/applauncher)

add eco app launcher components, wifi/sound/battery menus, demos and new eco-screensaver; replace dees-screensaver (breaking API change)

- Add eco-applauncher group and subcomponents: `eco-applauncher`, `eco-applauncher-wifimenu`, `eco-applauncher-soundmenu`, `eco-applauncher-batterymenu`
- Add demos for applauncher, wifi menu, sound menu and battery menu; include mock data for `networks/devices/apps`
- Introduce eco-screensaver component and demo; remove legacy `dees-screensaver` implementation and export
- Update elements index and z-index key (replace `'dees-screensaver'` with `'eco-screensaver'`)
- Menus dispatch events and include interactive behavior: `wifi-toggle`, `network-select`, `volume-change`, `device-select`, `battery-saver-toggle`, `settings-click`, `menu-close`

2026-01-06 - 3.34.0 - feat(dees-screensaver)

improve screensaver activation, visuals, and dismissal animations

- Add a safe demo flow that ensures only one screensaver instance: append an inactive element to document.body, store static instance, and activate after a 2s delay.
- Introduce static instance management and destroy flow so multiple demo calls don't leak elements.
- Rework DOM/CSS structure: add .screensaver-content container and .vignette element, separate backdrop, and cache element references for animations.
- Adjust transitions and timings (overlay opacity, vignette transform, time opacity) for smoother entrance and staged reveal.
- Implement a masked circular reveal dismissal: on click/touch, compute coordinates, animate a radial mask expanding from the pointer, dispatch screensaver-click with {x,y}, then remove the screensaver.
- Reset mask on activation to support repeated activations without leftover masking styles.

2026-01-06 - 3.33.0 - feat(dees-statsgrid)

add multiPercentage tile type to stats grid

- Add new 'multiPercentage' type to IStatsTile (percentages: [{label, value, color?}])
- Implement renderMultiPercentage() to render up to 3 percentage items with label, value and colored progress bars
- Add CSS styles for multi-percentage layout, bars, labels and values
- Update demo to replace 'Error Rate' tile with a 'Resource Usage' multiPercentage example (CPU, Memory, Disk)
- Change is additive and backward-compatible with existing tile types

2026-01-04 - 3.32.0 - feat(demo)

add demoGroup metadata to components and update related dependencies

- Add public static demoGroup properties to many components to categorize demos (groups added: App UI, Button, Chart, Data View, Form, Input, PDF, Simple, Workspace).
- Bump package deps/devDeps: @design.estate/dees-domtools -> ^2.3.7, @design.estate/dees-element -> ^2.1.5, @design.estate/dees-wcctools -> ^3.7.1.
- Clean up package.json deps (removed duplicate entries and removed 'lit' dependency).
- Refactor dees-pdf-viewer to use consolidated directives import (directives.keyed and directives.repeat) instead of separate keyed/repeat imports.

2026-01-04 - 3.31.0 - feat(dees-input-list)

enhance drag-and-drop reordering for dees-input-list and migrate tests to chromium runner

- Add rich drag state to dees-input-list: dragStartY, dragCurrentY, targetIndex, itemHeight and originalItemRects for accurate hit detection.
- Introduce bound global drag handlers and centralized global drag end/cleanup logic (handleGlobalDragOver / handleGlobalDragEnd).
- Improve drag visuals and animations: 'dragging', 'move-up', 'move-down' transforms, box-shadow, and smoother transitions; prevent hover styling while dragging.
- Move reorder logic away from per-item drop to global drag end to avoid race/positioning issues and ensure consistent reflow and cleanup.
- Migrate many browser test files to chromium-specific variants (added *.chromium.ts) and remove duplicate browser test counterparts.

2026-01-04 - 3.30.1 - fix(dees-statsgrid)

refine spacing, sizing, and colors in dees-statsgrid for a tighter, more compact appearance

- Reduce global spacing and sizing variables (grid-gap 16→12, tile-padding 24→16, header-spacing 16→12, content-min-height 48→40, description-spacing 12→8, border-radius 8→6).
- Adjust typographic scale (value-font-size 30→26, unit-font-size 16→14, label-font-size 13→12, title-font-size 14→13).
- Switch color tokens to neutral hex values and tighten hover/box-shadow (tile border and backgrounds updated from HSL to hex, hover bg to #fafafa/#0d0d0d, border-color and shadow reduced).

- Downsize graphical elements: gauge and SVG dimensions (width 140→120, height 80→70), stroke-widths 8→6, radius 48→40.
- Slim down percentage bar and trend visuals (percentage bar height 8→6, border-radius 4→3, trend stroke-width 2→1.5, trend fill moved to RGBA).
- No functional or API changes — purely visual/CSS and SVG adjustments.

2026-01-03 - 3.30.0 - feat(appui)

add dees-appui-bottombar component with config, programmatic API, demo and docs

- Adds a new dees-appui-bottombar web component (ts_web/elements/00group-appui/dees-appui-bottombar/) implementing widget and action management (add/update/remove/get/clear).
- Introduces bottom bar types and API in ts_web/elements/interfaces/appconfig.ts (IBottomBarWidget, IBottomBarAction, IBottomBarConfig, IBottomBarAPI) and extends the app config/type to include bottomBar and bottomBar APIs.
- Integrates the bottom bar into dees-appui: imports and registers component, renders conditionally, exposes bottomBar proxy API, visibility controls (set/getBottomBarVisible), and wires initial config to populate widgets/actions.
- Updates layout/styles (reduces main grid height to account for 24px fixed bottom bar and adds bottombar-hidden attribute handling) and exports component from the appui index.
- Adds interactive demos (dees-appui-bottombar.demo.ts and integration demo) and documents usage and API in readme.hints.md.

2026-01-03 - 3.29.3 - fix(elements/appui)

prevent scroll chaining on app UI components by adding overscroll-behavior: contain

- Added CSS overscroll-behavior: contain to activity log, main menu, secondary menu, profile dropdown, and tabs components to prevent scroll chaining and unintended body scrolling on touch/trackpad.
- Styling-only change; no public API or behavioral changes beyond scroll handling.
- Bump patch version from 3.29.2 to 3.29.3.

2026-01-03 - 3.29.2 - fix(dees-appui)

set min-height: 0 on .maingrid > dees-appui-maincontent to prevent layout overflow in flex container

- Added min-height: 0 to .maingrid > dees-appui-maincontent in ts_web/elements/00group-appui/dees-appui/dees-appui.ts to prevent unwanted growth/overflow when used inside a flex container.
- Pure CSS/layout fix — no API or behavior changes to components.

2026-01-03 - 3.29.1 - fix(dees-appui)

prevent main grid overflow by adding overflow:hidden; and add Playwright scroll containment screenshots

- Add overflow: hidden to .maingrid in ts_web/elements/00group-appui/dees-appui/dees-appui.ts to prevent content from escaping during grid-template-columns transitions.
- Add Playwright artifacts: .playwright-mcp/after-scroll.png and .playwright-mcp/scroll-containment-check.png (screenshots for scroll containment testing).

2026-01-03 - 3.29.0 - feat(docs)

add documentation for new input components, activity log features, theming, and expand DeesAppui docs

- Updated top-level README to reflect component count increase (75+ → 80+) and added many new component docs
- Added documentation and examples for DeesInputList (sortable list input) and DeesInputProfilepicture (cropping, upload, processing)
- Introduced DeesTheme documentation with usage examples and CSS custom property overrides
- Expanded DeesAppui readme with architecture overview, activity log panel docs, activity entry types, and navigation/secondary menu guidance

- Documented activity log APIs and controls (activityLog.add, addMany, clear, getEntries, filter, search) and new control API helpers (setActivityLogVisible, toggleActivityLog, getActivityLogVisible)
- Updated AppBar examples to include activity log toggle properties (.showActivityLogToggle, .activityLogCount, .activityLogActive) and @activity-toggle event
- Added interface docs (IViewDefinition, IActivityEntry) and updated menu/secondary menu type references
- Changes are documentation-focused (README/element readmes); no source code logic changes shown in this diff

2026-01-03 - 3.28.1 - fix(appui)

adjust layout and spacing in app UI components: fix activity log overflow, contain main content overscroll, and refine secondary menu padding/transition

- ts_web/elements/00group-appui/dees-appui-activitylog: removed host max-width, added overflow:hidden and set .maincontainer width to 280px to prevent horizontal overflow
- ts_web/elements/00group-appui/dees-appui-maincontent: added overscroll-behavior: contain to .content-area to prevent scroll chaining/overscroll
- ts_web/elements/00group-appui/dees-appui-secondarymenu: updated .groupHeader padding and hover border behavior, increased group icon size from 14px to 16px, and added margin + transition tweaks to .groupItems for smoother collapse/expand

2026-01-02 - 3.28.0 - feat(dees-appui)

Rename DeesAppuiBase to DeesAppui and migrate related API, exports, demos and docs

- Renamed public component/tag and TypeScript types: DeesAppuiBase -> DeesAppui and TDeesAppuiBase -> TDeesAppui; updated IViewActivationContext.appui type accordingly
- Moved/rewired view registry implementation from dees-appui-base to dees-appui and updated module exports
- Updated README and demo files to reference DeesAppui and new readme paths (removed dees-appui-base docs/demo)
- Replaced dependency/imports of '@webcontainer/api' with '@tempfix/webcontainer__api' (package.json and source imports)
- Changed tsconfig.json: skipLibCheck set from true to false

2026-01-01 - 3.27.1 - fix(dees-actionbar)

always render actionbar wrapper and delay adding visible class to ensure grid/opacity animations run reliably

- Always render the actionbar wrapper (.actionbar-item and .actionbar-content) instead of returning early so grid-template-rows and opacity transitions can animate.
- Use optional chaining for current bar access (bar?.type, bar?.timeout) to avoid runtime errors when no bar is present.
- Adjust styles and structure: set :host display:block; move background/border to .actionbar-item; add .actionbar-content with min-height/opacity and transitions.
- Make processQueue asynchronous and await updateComplete, then add the 'visible' class inside requestAnimationFrame so the CSS transition is triggered after render.

2026-01-01 - 3.27.0 - feat(services)

introduce DeesServiceLibLoader to lazy-load heavy client libraries from CDN and update components to use it

- Add DeesServiceLibLoader singleton (ts_web/services/DeesServiceLibLoader.ts) to lazily load and cache libraries via jsDelivr ESM: xterm, xterm-addon-fit, highlight.js, ApexCharts, and Tiptap.
- Inject xterm CSS dynamically to avoid shipping xterm styles in the initial bundle.
- Expose helper methods preloadAll() and isLoaded(), and typed bundle interfaces (IXtermBundle, IXtermFitAddonBundle, ITiptapBundle).
- Update components to use runtime-loaded modules: dees-chart-area, dees-dataview-codebox, dees-input-richtext, wysiwyg code block, dees-workspace-terminal, terminal-tab-manager, dees-workspace-terminal-preview.
- TerminalTabManager now requires setXtermModules(...) before creating tabs and will throw if not initialized; workspace terminal now initializes and passes the loaded modules.
- Replace direct runtime imports of heavy libs with typed imports and runtime-loaded bundles to reduce initial bundle size and improve load performance.

2026-01-01 - 3.26.1 - fix(dees-actionbar)

animate actionbar hide using grid-template-rows and wait for animation before clearing state

- Switch host layout from block/max-height to grid using grid-template-rows for open/close transitions
- Add min-height: 0 to .actionbar-item to prevent flex children overflow and collapsing
- Introduce async hideCurrentBar() that removes 'visible', sets isVisible=false, waits 220ms then clears currentBar and currentResolve
- processQueue() now calls hideCurrentBar() asynchronously instead of clearing state immediately

2026-01-01 - 3.26.0 - feat(workspace)

add external file change detection, conflict resolution UI, and diff editor

- Watch open files for external changes with debounced file watchers (startWatchingFile/stopWatchingFile/stopAllFileWatchers).
- Prompt the user when disk changes conflict with unsaved local edits via dees-actionbar (actions: Load from Disk, Save Local, Compare).
- Introduce dees-workspace-diff-editor component and export it; support comparing and resolving diffs (diff-resolved / diff-closed events).
- Add setContentExternal in dees-workspace-monaco to update editor content from external sources while optionally preserving cursor, selections and scroll position.
- Start/stop file watchers when files are opened/closed and integrate diff view and actionbar into the workspace UI for seamless conflict handling.

2026-01-01 - 3.25.0 - feat(dees-actionbar)

add action bar component and improve workspace package update handling

- Introduce dees-actionbar component (dees-actionbar.ts) with interfaces, queueing, timed auto-trigger and demo usage
- Add actionbar.interfaces.ts and index export; export dees-actionbar from elements index
- Enhance workspace bottombar: add pendingPackageUpdate flag, process-complete handler, and connected/disconnected listeners to auto-refresh package status after updates
- Make pnpm outdated checking robust by streaming output via a reader and adding a 10s timeout to avoid hanging; handle timeout and stream cancellation
- Update package update commands to include '--latest' for updatePackage and updateAllPackages, and show 'Checking...' label during checks
- Add '@types/node' (^22.0.0) to devDependencies in the workspace package config

2026-01-01 - 3.24.0 - feat(workspace)

add workspace bottom bar, terminal tab manager, and run-process integration

- Add dees-workspace-bottombar component and export; bottom bar emits run-process events to launch processes.
- Introduce terminal interfaces (IRunProcessEventDetail, ITerminalTab, ICreateTerminalTabOptions, etc.) and a TerminalTabManager to manage multiple terminal tabs and lifecycle.
- Integrate bottombar into dees-workspace: layout refactor (workspace-outer), move/adjust resize handles and panels, and handle @run-process to create terminal tabs and focus terminal panel.
- Enhance dees-workspace-terminal: tabbed terminal UI, new public APIs (createProcessTab, writeToTab, sendInputToTab), theme updates, and improved disposal/cleanup behavior.
- Update module exports to include bottombar and additional terminal sub-exports (interfaces, terminal-tab-manager).

2025-12-31 - 3.23.0 - feat(workspace)

add resizable file tree and terminal panes with draggable handles and public layout APIs

- Introduce reactive state for `currentFileTreeWidth`, `currentTerminalHeight`, `isDraggingFileTree` and `isDraggingTerminal`
- Add mouse event handlers (`mousedown/move/up`) to drag-resize the file tree and terminal with min/max clamping
- Dispatch window resize event after resizing to notify Monaco/editor of layout changes
- Clean up resize event listeners in `disconnectedCallback`
- Initialize current sizes from component properties in `firstUpdated`
- Expose public layout methods: `setFileTreeWidth`, `setTerminalHeight`, `resetLayout`

2025-12-31 - 3.22.0 - feat(workspace)

add resizable markdown editor/preview split with draggable handle and markdown outlet styling/demo

- Introduce a flexible split layout for the workspace markdown editor with a draggable resize handle (`splitRatio`, `minPanelSize`, dragging state, mouse handlers and cleanup).
- Enhance `dees-workspace-markdown`: switch from grid to flex, add resize handle UI, prevent pointer selection while dragging, notify Monaco on resize.
- Add comprehensive styling and demo content for `dees-workspace-markdownoutlet` (dark/light themed markdown styles, syntax highlighting classes, and demo scenarios).
- Fix `typescript-intellisense monaco` model update: only set model value when content actually changed to avoid cursor resets.
- Add markdown outlet demo helper and numerous screenshot/image assets (`.playwright-mcp`) for demos/documentation.

2025-12-31 - 3.21.0 - feat(terminal)

add dynamic bright/dark theming for terminal components and terminal preview

- Add bright/dark theme PNG assets under `.playwright-mcp` for previews.
- Replace hardcoded terminal background/colors with `cssManager.bdTheme` in workspace terminal and preview styles.
- Introduce `getTerminalTheme` helper to compute `xterm` theme for bright/dark modes.
- Subscribe to `themeManager.themeObservable` and apply updates to `xterm` (`terminal.options.theme`) so terminals update live on theme change.

- Remove hardcoded background property/CSS var and unused background accessor from workspace terminal.
- Ensure proper cleanup: unsubscribe theme subscriptions and dispose terminals in `disconnectedCallback`.

2025-12-31 - 3.20.1 - fix(dees-workspace)

fix demo wrapper and workspace layout; always render terminal preview

- Import `@design.estate/dees-wcctools/demotools` and wrap demo content in
- Create an absolute-positioned container so the `dees-workspace` fills 100% height/width
- Always render `dees-workspace-terminal-preview` (use empty command when none) to avoid conditional rendering issues
- Set a fixed height (200px) for the terminal preview in the initializing state
- Add Playwright demo asset `.playwright-mcp/dees-workspace-demo-4k.png`

2025-12-31 - 3.20.0 - feat(workspace)

rename editor components to workspace group and move terminal & TypeScript intellisense into workspace

- Renamed components and modules from `00group-editor/dees-editor-*` to `00group-workspace/dees-workspace-*` (e.g. `dees-editor-monaco` -> `dees-workspace-monaco`).
- Moved terminal implementation from `dees-terminal` to `dees-workspace-terminal` and updated related imports/usages.
- Moved TypeScript intellisense manager into `00group-workspace` and updated paths.
- Updated `ts_web` elements index to export `00group-workspace` instead of `00group-editor` and adjusted internal imports accordingly.
- Updated `scripts/update-monaco-version.cjs` to write `MONACO_VERSION` into the new workspace path and updated log tags.

2025-12-31 - 3.19.1 - fix(intellisense)

Debounce TypeScript/JavaScript IntelliSense processing and cache missing packages to reduce work and noisy logs

- Add 500ms debounce in editor workspace to avoid parsing on every keystroke
- Introduce notFoundPackages cache to skip repeated filesystem checks for packages without types
- Clear not-found cache when scanning node_modules so newly installed packages are re-detected
- Remove noisy console logs and make file/directory read errors non-fatal (ignored)
- Simplify processContentChange signature (removed optional filePath) and remove manual diagnostic refresh logic

2025-12-31 - 3.19.0 - feat(dees- editor-workspace)

improve TypeScript IntelliSense, auto-run workspace init commands, and watch node_modules for new packages

- Execute an onInit command from /npmextra.json on workspace initialization (e.g., run pnpm install).
- Add npmextra.json and an import test file (importtest.ts) plus a sample dependency in the scaffold to test package imports.
- Add node_modules watcher with debounce to auto-scan and load package types after installs.
- Enhance TypeScript IntelliSense: recursively load all .d.ts files from packages and @types packages, add package.json as extra lib, and log progress/errors for debugging.
- Change processContentChange signature to accept optional filePath and trigger a diagnostic refresh when new types are loaded.
- Expose scanAndLoadNewPackageTypes to scan top-level and scoped packages and load their types.
- Add start/stop logic for the node_modules watcher in workspace lifecycle to avoid leaks and handle cleanup.

2025-12-31 - 3.18.0 - feat(filetree)

add filesystem watch support to WebContainer environment and auto-refresh file tree; improve icon handling and context menu behavior

- Add IFileWatcher interface and watch(...) signature to IExecutionEnvironment.
- Implement watch(...) in WebContainerEnvironment using WebContainer's fs.watch and return a stop() handle.
- dees-editor-filetree: start/stop file watcher, debounce auto-refresh on FS changes, cleanup on disconnect, and track last execution environment.
- Add clipboard state (copy/cut) and related UI/menu enhancements for file operations (new file/folder, rename, delete, copy/paste).
- dees-icon: default to Lucide icons when no prefix is provided.
- dees-contextmenu: remove 'lucide:' prefix usage in templates and avoid awaiting windowLayer.destroy() to provide instant visual feedback.
- Menu item shape adjusted (use { divider: true } for dividers) and various menu icon name updates.

2025-12-31 - 3.17.0 - feat(editor)

add file explorer toolbar, empty-space context menu, editor auto-save, save-all, and keyboard save shortcuts

- Added filetree toolbar with New File / New Folder actions and toolbar styling
- Added right-click context menu for empty filetree space to create files/folders
- Implemented editor menu button with context menu (Auto Save toggle, Save, Save All)
- Added auto-save toggle with 2s interval and cleanup on disconnect
- Implemented Save and Save All APIs that persist files and update IntelliSense manager
- Added keyboard shortcuts: Cmd/Ctrl+S to save active file and Cmd/Ctrl+Shift+S to save all
- Made tabs scrollable with a tabs container and added an editor menu button

2025-12-30 - 3.16.0 - feat(editor)

improve TypeScript IntelliSense and module resolution for Monaco editor

- Add file cache (fileCache) and getFileContent() for synchronous access to project files
- Track and dispose Monaco extra libs (addedExtraLibs) and register project files via addExtraLib to enable TypeScript module resolution

- Add `addFileAsExtraLib` logic to register `.ts/.tsx` files also under `.js/.jsx` paths so ESM imports resolve to TypeScript sources
- Use `ModuleResolutionKind.Bundler` fallback to NodeJS and set `compilerOptions` (`baseUrl` `'/'`, `allowImportingTsExtensions`, `resolveJsonModule`) to improve resolution
- Adapt `executionEnvironment` API usage to `readDir/readFile` and check `entry.type` (`'directory'|'file'`) instead of `isDirectory/isFile`
- Add a debugging/screenshot asset: `.playwright-mcp/module-resolution-fixed.png`

2025-12-30 - 3.15.0 - feat(editor)

enable file-backed Monaco models and add Problems panel; lazy-init project TypeScript IntelliSense

- `dees-editor-monaco`: add `filePath` property and `create/get` Monaco models with `file://` URIs so editors are backed by real models; sync content into models and handle model switching when `filePath` changes; enable hover config and improved lifecycle handling.
- `dees-editor-workspace`: add bottom 'Problems' panel and panel tabs (`terminal/problems`), `diagnosticMarkers` state, marker listener, UI for problem list, and navigation to file/position when a problem is clicked; initialize IntelliSense lazily when a file is opened.
- `typescript-intellisense`: index project `.ts/.js` files from the virtual filesystem into Monaco models for cross-file resolution, enable `allowNonTsExtensions` and set eager model sync so TypeScript processes models eagerly.
- General: improved handling for language changes, model language switching, and deferred initialization of the IntelliSense manager.
- Add Playwright test images (workspace screenshots) used by CI/tests.

2025-12-30 - 3.14.2 - fix(editor)

bump monaco-editor to 0.55.1 and adapt TypeScript intellisense integration to the updated Monaco API

- Bumped dependency `monaco-editor` from 0.52.2 to 0.55.1 in `package.json`.
- Generated `MONACO_VERSION` module updated to 0.55.1 and moved target to `ts_web/elements/00group-editor/dees-editor-monaco/version.ts`.
- Refactored TypeScript IntelliSense code to use a typed Monaco TS API (added `IMonacoTypeScriptAPI`, `tsApi` getter, and replaced direct `monaco.languages.typescript.*` calls).
- Added `test/workspace` screenshot `.playwright-mcp/workspace-test.png` (binary asset).

2025-12-30 - 3.14.1 - fix(build)

bump @webcontainer/api and enable skipLibCheck to avoid type-check conflicts

- Updated @webcontainer/api from 1.2.0 to 1.6.1
- Added "skipLibCheck": true to tsconfig.json compilerOptions to suppress external library type errors
- No breaking changes expected; this is a build/dev fix

2025-12-30 - 3.14.0 - feat(editor)

add modal prompts for file/folder creation, improve Monaco editor reactivity and add TypeScript IntelliSense support

- Replace window.prompt for new file/folder with DeesModal + DeesInputText (showInputModal) to provide a focused modal input UX.
- Monaco editor: add language property, handle external content updates without emitting change events (isUpdatingFromExternal), dispatch 'content-change' events, and apply language changes at runtime.
- Add TypeScriptIntelliSenseManager to load .d.ts/type packages from the virtual filesystem (/node_modules), parse imports, load @types fallbacks, and add file models to Monaco for cross-file IntelliSense.
- Workspace demo now mounts an initial TypeScript project and exposes initializationPromise to wait for external setup; workspace initializes IntelliSense and processes content changes to keep types up to date.
- Export typescript-intellisense from workspace index so the manager is available to consumers.

2025-12-30 - 3.13.1 - fix(webcontainer)

prevent double initialization and race conditions when booting WebContainer and loading editor workspace/file tree

- Add loadTreeStarted flag in dees-editor-filetree to avoid double-loading the file tree and reset it on refresh or on error to allow retries.
- Add initializationStarted flag in dees-editor-workspace to prevent duplicate workspace initialization and reset it on initialization failure to allow retry.
- Make WebContainerEnvironment use a shared singleton container and a bootPromise so only one WebContainer boot runs per page; instances wait for an ongoing boot instead of booting again.

- Reset bootPromise/sharedContainer on boot failure and clear them on teardown so subsequent attempts can retry cleanly.

2025-12-30 - 3.13.0 - feat(editor/runtime)

Replace bare editor with Monaco-based editor and add runtime + workspace/filetree integration

- Removed dees-editor-bare and replaced usages with dees-editor-monaco (includes MONACO_VERSION file).
- Added IExecutionEnvironment interface and WebContainerEnvironment implementation (uses @webcontainer/api) to provide a browser Node/runtime API.
- Added new components: dees-editor-filetree and dees-editor-workspace to support file tree, multiple open files, and workspace actions wired to the execution environment.
- dees-terminal updated to accept an executionEnvironment (IExecutionEnvironment), renamed environment -> environmentVariables, provides environmentPromise (deprecated note), and now initializes/uses the provided environment to spawn shell processes and write /source.env.
- Updated imports/usages across components (dees-input-code, dees-editor-markdown, group index exports) to use the new Monaco editor and runtime modules.
- Behavioral breaking changes: consumers must supply an IExecutionEnvironment to components that now depend on it (e.g. dees-terminal, workspace, filetree); dees-editor-bare removal is a breaking API change.

2025-12-30 - 3.12.2 - fix(dees- editor-bare)

make Monaco editor follow domtools theme and clean up theme subscription on disconnect

- Set initial Monaco theme from domtools.themeManager.goBrightBoolean instead of hardcoded 'vs-dark'
- Subscribe to domtools.themeManager.themeObservable to update editor theme dynamically
- Add monacoThemeSubscription property and unsubscribe in disconnectedCallback to avoid memory leaks

2025-12-30 - 3.12.1 - fix(modal)

fix modal editor layout to prevent overlap by adding relative positioning and reducing height

- Added Playwright screenshots: `.playwright-mcp/dees-input-code-demo.png` and `.playwright-mcp/modal-overlap-issue.png`
- Updated `ts_web/elements/00group-input/dees-input-code/dees-input-code.ts`: `modal-editor-wrapper` set position: relative and changed height from `calc(100vh - 160px)` to `calc(100vh - 175px)` to avoid overlap

2025-12-30 - 3.12.0 - feat(editor)

add code input component and editor-bare, replace dees-editor usage, and add modal `contentPadding`

- Add new `dees-input-code` component (full-featured code editor input with modal, toolbar, language selector, copy and wrap toggles).
- Introduce `dees-editor-bare` component and remove the legacy `dees-editor` implementation; update editor markdown component to use `dees-editor-bare`.
- Export and include `DeesInputCode` in input index and include it in the unified form input types and `dees-form` usage.
- Add `contentPadding` property to `DeesModal` and apply it to the modal content area (configurable modal inner padding).
- Update element exports to point to `dees-editor-bare` and adjust related imports/usages.
- Bump devDependency `@design.estate/dees-wcctools` from `^3.3.0` to `^3.4.0` in `package.json`

2025-12-30 - 3.11.2 - fix(tests)

make WYSIWYG tests more robust and deterministic by initializing and attaching elements consistently, awaiting `customElements/firstUpdated`, adjusting selectors and assertions, and cleaning up DOM after tests

- Create WYSIWYG elements with `document.createElement` and set properties before attaching to DOM to ensure `firstUpdated` sees data
- Await `customElements.whenDefined` and add small delays (`setTimeout`) so nested components finish rendering in test environments
- Replace outdated selectors (`.block.code`) with `.code-editor` and update expectations for code block selection and language controls

- Adjust divider expectations to check for

and data-block-id instead of a divider icon; change toBeDefined -> toBeTruthy for assertions where appropriate

- Add cleanup (document.body.removeChild) after tests to avoid leaking elements between tests
- Relax computed font-family assertion to be platform-agnostic and verify that a fontFamily exists rather than matching 'monospace'
- Add notes/guards around synthetic DragEvent/KeyboardEvent behavior: verify handlers/state existence and dispatch events but avoid relying on native focus/drag internals in CI
- Update BlockRegistry render tests to assert template structure (data-block-id, data-block-type, class names) rather than final content which is populated later

2025-12-30 - 3.11.1 - fix(tests)

migrate tests to @git.zone/tstest tapbundle and export tap.start() in browser tests

- Replaced imports from @push.rocks/tapbundle to @git.zone/tstest/tapbundle across test files
- Replaced bare tap.start() calls with export default tap.start() in browser test files so the runner can be imported
- Bumped devDependency @git.zone/tstest from ^3.1.3 to ^3.1.4 and removed @push.rocks/tapbundle from devDependencies
- Changes include package.json and updates to multiple test files (11 test files)

2025-12-30 - 3.11.0 - feat(dees-appui-tabs)

improve horizontal tabs UX with scroll fades, hover scrollbar, and smooth scroll-to-selected

- Add reactive scroll state (canScrollLeft / canScrollRight) and updateScrollState to track horizontal overflow.
- Introduce scroll-fade gradient elements and CSS to indicate overflow on left/right edges.
- Show a thin, styled scrollbar on hover (webkit + Firefox styling) instead of hiding it completely.
- Auto-scroll selected tab into view using scrollTabIntoView and smooth scroll when selecting a tab.

- Set up a ResizeObserver to recompute scroll state on container size changes and clean it up on disconnect.
- Ensure lifecycle hooks call updateScrollState (firstUpdated/updated) so indicators stay in sync after render/fonts ready.

2025-12-29 - 3.10.0 - feat(appui-tabs)

add closeable tabs and auto-hide behavior for content tabs, plus API and events to control them

- Add closeable tab support: `IMenuItem.closeable` & `IMenuItem.onClose`; `dees-appui-tabs` renders a close button, invokes `onClose`, and emits a 'tab-close' event.
- Add auto-hide feature: `dees-appui-tabs` (`autoHide`, `autoHideThreshold`) and corresponding properties in `dees-appui-maincontent/dees-appui-base` to hide tabs when count \leq threshold.
- Expose new API: `dees-appui-base.setContentTabsAutoHide(enabled, threshold)` and update `appconfig` interface to include `setContentTabsAutoHide`.
- Re-emit 'tab-close' events from `dees-appui-maincontent` and `dees-appui-base` so parent components can react to tab closures.
- Add interactive demos (`demo-closeable-tabs`, `demo-autohide-tabs`) demonstrating the new closeable and auto-hide behaviors and controls.

2025-12-29 - 3.9.0 - feat(dees-appui-mainmenu)

add status badges to main menu items with theme-aware styling

- Introduce `.badge` element and layout (`min-width`, `height`, `padding`, `font-size`, `weight`, `border-radius`) to display counts/status on menu items.
- Add four badge variants: `default`, `success`, `warning`, `error`, using `cssManager.bdTheme` for light/dark color pairs.
- Render the badge element conditionally in the menu item template when `tabArg.badge` is provided; hide badges when host has `[collapsed]` attribute.

2025-12-29 - 3.8.0 - feat(dees-appui-base)

add interactive demo controls to manipulate appui via view activation context

- Store IViewActivationContext on the demo element (this.ctx) during onActivate
- Add a new "Context Actions" UI section with buttons that call ctx.appui methods (toggle main/secondary menus, content tabs, collapse/expand main menu, set breadcrumbs, navigate to views, add activity entry, set/clear menu badges)
- Include styles for ctx-actions and button variants (success, danger, hover states)
- Change is limited to the demo file (dees-appui-base.demo.ts) and is non-breaking

2025-12-29 - 3.7.1 - fix(dees-appui-maincontent)

migrate main content layout to CSS Grid and enable topbar collapse transitions

- Replace absolute positioning with CSS Grid on :host and .maincontainer to enable natural document flow
- Make .topbar a grid and animate collapse via grid-template-rows; switch :host([notabs]) to grid-template-rows: 0fr instead of display:none to allow transitions
- Set .maincontainer to display:contents and add min-height:0 on content areas and topbar children to fix overflow/scrolling and flex/grid height issues
- Remove positional styles (position:absolute/top/left/right/bottom) so content scrolls correctly and layout is more robust

2025-12-29 - 3.7.0 - feat(dees-contextmenu,dees-appui-tabs,test)

Prevent double-destruction of context menus, await window layer teardown, update destroyAll behavior, remove tabs content slot, and adjust tests

- Add isDestroying guard in DeesContextmenu.destroy to avoid double-destruction.

- Await `windowLayer.destroy()` to ensure the window layer is fully torn down before continuing.
- Ensure submenu timeouts are cleared and `submenu.destroy()` is awaited during teardown.
- Change `destroyAll` to locate the root/top-level menu and destroy from the root to cascade teardown reliably.
- Remove the `.content` wrapper and the output from `dees-appui-tabs` (demo updated to render content outside the component) — this is a breaking change to the tabs API (slotted content no longer rendered).
- Increase test timeout in `test.contextmenu-nested-close.browser.ts` to 600ms to account for ~300ms `windowLayer` destruction + ~100ms context menu delay.

2025-12-29 - 3.6.1 - fix(readme)

document new App UI APIs to control main/secondary menu and content tabs visibility

- Added docs for `setMainMenuCollapsed()`, `setMainMenuVisible()`, `setSecondaryMenuCollapsed()`, `setSecondaryMenuVisible()`, and `setContentTabsVisible()` programmatic APIs.
- Included a TypeScript example showing how to hide secondary menu, hide content tabs, and collapse the main menu in a view's `onActivate` hook.

2025-12-29 - 3.6.0 - feat(dees-appui)

add visibility toggles for main/secondary menus and ability to show/hide content tabs; expose corresponding setters and appconfig entries

- `ts_web/elements/00group-appui/dees-appui-base`: added boolean properties `mainmenuVisible`, `secondarymenuVisible`, `maincontentTabsVisible`; render main and secondary menus conditionally; pass `showTabs` to `dees-appui-maincontent`; added setter methods: `setMainMenuVisible`, `setSecondaryMenuCollapsed`, `setSecondaryMenuVisible`, `setContentTabsVisible`.
- `ts_web/elements/00group-appui/dees-appui-maincontent`: added `showTabs` property, support for a `notabs` attribute via `styles`, `updated()` and `firstUpdated()` to apply `notabs` state so tabs can be hidden/shown dynamically.
- `ts_web/elements/interfaces/appconfig.ts`: expanded `appconfig` interface to include `setMainMenuVisible`, `setSecondaryMenuCollapsed`, `setSecondaryMenuVisible`, `setContentTabsVisible` so host app can control visibility programmatically.

- No breaking changes: defaults preserve existing behavior (menus and tabs remain visible by default).

2025-12-29 - 3.5.1 - fix(dees-appui-view)

remove DeesAppuiView component, its demo, documentation snippet, and related exports

- Deleted component implementation: `ts_web/elements/00group-appui/dees-appui-view/dees-appui-view.ts`
- Deleted demo file: `ts_web/elements/00group-appui/dees-appui-view/dees-appui-view.demo.ts`
- Removed index re-export: `ts_web/elements/00group-appui/dees-appui-view/index.ts` (deleted) and removed export from `ts_web/elements/00group-appui/index.ts`
- Removed documentation section for DeesAppuiView from `readme.md`
- Breaking change: any consumers using the component or its public API (`selectTab`, `getMenuItems`, `getTabs`) must be migrated to alternate components/approach

2025-12-29 - 3.5.0 - feat(theme,interfaces)

Introduce a global theming system and unify menu/tab interfaces; migrate components to use `themeDefaultStyles` and update APIs accordingly

- Add a new theme module and component (`00theme.ts` + `dees-theme`) that provides CSS tokens and `themeDefaultStyles` to import into components
- Migrate many components to include `themeDefaultStyles` in their static styles and add TODOs to replace hardcoded values with CSS variables
- Rename `ITab` -> `IMenuItem` and replace `group.tabs` with `group.items` across interfaces and components (`IMenuGroup` shape changed) — this is a breaking API change
- Remove legacy interfaces (`ISecondaryMenuGroup`, `ISelectionOption`) and update method and property types in `DeesAppui*` components and app config to use `IMenuItem/IMenuGroup`
- Move `@design.estate/dees-wcctools` from dependencies to devDependencies and bump its version to `^3.3.0`
- Add numerous demo files and expand README with usage, examples and theme documentation

2025-12-19 - 3.4.0 - feat(dees-appui-base)

overhaul AppUI core: replace simple view rendering with a full-featured ViewRegistry (caching, hide/show lifecycle, async lazy-loading), introduce view lifecycle hooks and activation context, add activity log API/component, remove built-in router and state manager, and update configuration interfaces and demos

- Removed files: app.router.ts and state.manager.ts — routing and state-persistence internals were removed (breaking).
- ViewRegistry rewritten: supports cached instances, activate/deactivate lifecycle, canDeactivate checks, async content loading, parameterized routes, and legacy renderView kept as deprecated compatibility.
- New interfaces added/changed: IViewActivationContext, IViewLifecycle, IActivityEntry, IActivityLogAPI, IViewLifecycleEvent; IViewDefinition.content now accepts async loaders and a cache flag; IMainMenuConfig and ITab expanded (logo, groups, badges).
- Activity log: dees-appui-activitylog now implements IActivityLogAPI and exposes reactive entries; demo and readme updated with usage and examples.
- App config changed: routing and statePersistence config entries removed/adjusted; defaultView moved into IAppConfig; view change and lifecycle event shapes changed (breaking).
- Demos and documentation: dees-appui-base demo and readme added/updated to showcase new lifecycle hooks, secondary menu behavior, activity log and new APIs.

2025-12-19 - 3.3.3 - fix(tests)

update test imports to new dees-input-wysiwyg paths

- Updated imports in test/test.wysiwyg-registry.both.ts to point to ts_web/elements/00group-input/dees-input-wysiwyg/*
- Aligns test references with relocated WYSIWYG block handlers and block registration module; no behavior changes to implementation

2025-12-19 - 3.3.2 - fix(build)

update build config, bump dependencies, and adjust test import paths after element reorganization

- npmextra.json: renamed gitzone entry to @git.zone/cli, moved tsdoc key to @git.zone/tsdoc, added @ship.zone/szci entry and added release registries + accessLevel
- package.json: bumped @design.estate/dees-wcctools ^2.0.1 -> ^3.1.0, lucide ^0.560.0 -> ^0.562.0, @git.zone/tsbuild ^3.1.2 -> ^4.0.2, @types/node ^25.0.0 -> ^25.0.3
- tests: updated import paths to follow reorganized source layout (wysiwyg files moved under elements/00group-input/dees-input-wysiwyg and dees-contextmenu moved to elements/dees-contextmenu/dees-contextmenu.js); updated BlockRegistry and blockregistration import paths
- Purpose: align tests and build metadata with refactored element file locations and updated tool/dependency versions

2025-12-11 - 3.3.1 - fix(dees-pdf-viewer)

Scroll active PDF thumbnail into view after rendering and on page changes; update dependency versions

- Ensure the active thumbnail is scrolled into view after thumbnails are rendered (improves sidebar navigation for dees-pdf-viewer).
- Scroll the thumbnail into view when navigating pages if the sidebar is visible (prevents the active page from being off-screen).
- Retain re-setup of the intersection observer for lazy-loading pages after thumbnail re-render.
- Bumped dependencies in package.json: @design.estate/dees-wcctools -> ^2.0.1, lucide -> ^0.560.0, @git.zone/tswatch -> ^2.3.13, @types/node -> ^25.0.0.

2025-12-09 - 3.3.0 - feat(dees-appui-base)

Add unified App UI API to dees-appui-base with ViewRegistry, AppRouter and StateManager

- Introduce ViewRegistry for declarative view registration and rendering (supports tag names, element classes and template functions).
- Add AppRouter with hash/history/external/none modes, URL synchronization, navigate/back/forward and onRouteChange listener support.
- Add StateManager to persist UI state (localStorage, sessionStorage or in-memory) with save/load/update/clear APIs.

- Extend interfaces (interfaces/appconfig.ts) with IAppConfig, IViewDefinition, IRoutingConfig, IStatePersistenceConfig and IAppState.
- Expose new public DeesAppuiBase methods: configure, navigateToView, getCurrentView, getUIState, restoreUIState, saveState, loadState, getViewRegistry, getRouter.
- Maintain backward compatibility with existing property-based API and slot usage.
- Export new modules (view.registry, app.router, state.manager) from dees-appui-base index and update element exports.

2025-12-08 - 3.2.0 - feat(dees-simple-appdash, dees-simple-login, dees-terminal)

Revamp UI: dashboard & login styling, standardize icons to Lucide, and add terminal background/config

- Standardize icon usage to Lucide prefixes in dees-simple-appdash; add fallback handling for legacy icon names
- Revamped dees-simple-appdash sidebar: updated spacing, typography, header icon wrapper, scrollbar styling, section labels, hover/selected states, and visual indicators
- Change 'Logout' label to 'Sign out' in dees-simple-appdash and add explicit status classes for controlbar (connected, terminal)
- Improve terminal UX: smoother launch/close animations, updated shadow and sizing logic in dees-simple-appdash
- Add background property to dees-terminal, sync it to a CSS variable and apply it to xterm theme for configurable terminal background
- Redesign dees-simple-login: new header/subheader, card layout, spacing, and updated submit text to 'Sign in'
- Bump devDependency @git.zone/tswatch to ^2.3.5

2025-12-08 - 3.1.2 - fix(DeesAppuiMainmenu, DeesAppuiSecondarymenu)

Add position: relative to main and secondary app UI menus to fix positioning of overlays and tooltips

- `ts_web/elements/00group-appui/dees-appui-mainmenu/dees-appui-mainmenu.ts`: add `position: relative` to host styles
- `ts_web/elements/00group-appui/dees-appui-secondarymenu/dees-appui-secondarymenu.ts`: add `position: relative` to host styles
- Fixes incorrect positioning for absolutely positioned children (tooltips, overlays, badges) inside the main and secondary menus

2025-12-08 - 3.1.1 - fix(dees-appui)

Extract demos for main and secondary app menus, adjust collapsed styles and toggle placement, bump devDependency

- Extracted inline demo markup into separate demo files: `ts_web/elements/00group-appui/dees-appui-mainmenu/dees-appui-mainmenu.demo.ts` and `ts_web/elements/00group-appui/dees-appui-secondarymenu/dees-appui-secondarymenu.demo.ts` and wired them up via imported `demoFunc` to reduce component size.
- Moved collapse toggle button markup in both `dees-appui-mainmenu` and `dees-appui-secondarymenu` templates to after the main container to improve layout/stacking and focus behavior.
- Adjusted collapsed logo/heading styles: removed extra padding/gap and hide logo text using `display:none` for a cleaner collapsed state.
- Bumped devDependency `@git.zone/tswatch` from `^2.3.1` to `^2.3.2` in `package.json`.

2025-12-08 - 3.1.0 - feat(dees-appui)

Add collapsible/compact mode to AppUI sidebars (mainmenu & secondarymenu) with toggles, tooltips and improved z-index stacking

- Add collapsed property to `dees-appui-mainmenu` and `dees-appui-secondarymenu` (`reflect: true`) to enable compact horizontal mode.

- Add floating collapse toggle buttons and public toggleCollapse() methods on mainmenu and secondarymenu; these dispatch 'collapse-change' events (bubbles & composed).
- Expose and track collapse state in dees-appui-base via mainmenuCollapsed and secondarymenuCollapsed properties; bind states to child components and re-emit collapse-change events as mainmenu-collapse-change and secondarymenu-collapse-change.
- Implement collapsed styles and animations: reduced sidebar widths, hide/compact labels and headers, center icons, hide badges, and add smooth width/opacity transitions.
- Add tooltips that appear for tabs/items when sidebars are collapsed to preserve discoverability.
- Adjust layout grid in DeesAppuiBase (use auto columns) and add explicit z-index layering to ensure proper stacking order of mainmenu, secondarymenu, maincontent and activitylog.

2025-12-08 - 3.0.1 - fix(dees-appui)

Normalize header heights and box-sizing for App UI components

- Set topbar/header heights to 48px (was 40px) and adjusted dependent offsets (activity container top, topShadow position) in dees-appui-activitylog.
- Make logo and secondary menu headers fixed 48px tall and replace vertical padding with horizontal padding for consistent vertical alignment (dees-appui-mainmenu, dees-appui-secondarymenu).
- Ensure tabs wrapper uses explicit 48px height and tabsContainer fills height (height:100%) to keep tab items vertically centered (dees-appui-tabs).
- Add box-sizing: border-box to affected header/logo containers to prevent overflow and ensure correct sizing.
- Minor CSS alignment and overflow fixes to improve consistent layout and scrolling behavior across the app UI components.

2025-12-08 - 3.0.0 - BREAKING CHANGE(dees-appui-secondarymenu)

Add SecondaryMenu component and replace Mainselector with new SecondaryMenu in AppUI

- Add dees-appui-secondarymenu component: collapsible groups, badges, dynamic heading, context menu and legacy flat-options support
- Introduce interfaces ISecondaryMenuItem and ISecondaryMenuGroup under elements/interfaces
- Replace dees-appui-mainselector usage with dees-appui-secondarymenu in DeesAppuiBase (props/events updated: secondarymenuGroups, secondarymenuHeading, secondarymenuOptions, item-select / secondarymenu-item-select)
- Remove dees-appui-mainselector implementation and its index export; update group exports and imports to expose secondarymenu
- Update demos and pages to showcase the new SecondaryMenu and adjust import paths for grouped components
- Bump devDependency @git.zone/tswatch to ^2.3.1

2025-12-08 - 2.0.7 - fix(structure)

Add many new UI components, input controls, charts, editors, and demos

- Introduce App UI components: dees-appui-appbar, dees-appui-mainmenu, dees-appui-mainselector, dees-appui-maincontent, dees-appui-activitylog, dees-appui-profileddropdown, dees-appui-tabs, dees-appui-base, dees-appui-view (templates, styles and demos included).
- Add a comprehensive set of input components: dees-input-text, dees-input-checkbox, dees-input-dropdown, dees-input-fileupload, dees-input-datepicker, dees-input-phone, dees-input-iban, dees-input-quantityselector, dees-input-list, dees-input-typelist, dees-input-tags, dees-input-multitoggle, dees-input-radiogroup, dees-input-richtext and supporting demos/styles/templates.
- Add form primitives and integration: dees-form and dees-form-submit with validation, collection and demo pages showcasing usage.
- Add button family and utilities: dees-button (with updated variants, sizes, status handling and demo), dees-button-group and dees-button-exit.
- Add charting components: dees-chart-area (ApexCharts integration) and dees-chart-log (log viewer) plus rich demo scenarios and realtime features.
- Add data display components: dees-dataview-codebox (highlight.js integration) and dees-dataview-statusobject with copy/context behaviours and demos.
- Add editor tooling: dees-editor (Monaco loader/version management), dees-editor-markdown and dees-editor-markdownoutlet; also TipTap-based richtext input with toolbar and link handling.
- Add global utilities and infra: dees-toast (programmatic toast API and containers), z-index registry and theme/font helpers (fonts, color tokens), plus many styles and accessibility/keyboard improvements across components.
- Export and index updates: new group exports added to ts_web/elements index and many index.ts files to expose the new components and demos.

- Extensive demos and showcase pages added (input-showcase, component demos) to illustrate integration, keyboard navigation, theming and form flows.

2025-12-06 - 2.0.6 - fix(dees-input-richtext)

Initialize editor and link input element references in firstUpdated to ensure they exist before editor initialization.

- Assign editorElement from shadowRoot.querySelector('.editor-content') in firstUpdated.
- Assign linkInputElement from shadowRoot.querySelector('.link-input input') in firstUpdated.
- Call initializeEditor() after DOM references are set to avoid undefined-element runtime errors.

2025-12-06 - 2.0.5 - fix(build)

Bump devDependencies: update @git.zone/tsbundle and @git.zone/tswatch to patched versions

- Update @git.zone/tsbundle from ^2.6.2 to ^2.6.3
- Update @git.zone/tswatch from ^2.2.2 to ^2.2.3

2025-12-06 - 2.0.4 - fix(imports)

Normalize and fix relative import paths for web components to ensure correct module resolution

- Replaced numerous './.js' imports with explicit '../.js' paths across many elements and demos to fix module resolution.
- Updated imports for core shared components such as dees-icon, dees-panel, dees-contextmenu, dees-windowlayer, dees-windowcontrols and several app-ui components (appbar, maincontent, mainselector, activitylog, mobilenavigation, modal, pdf, profilepicture, statsgrid, etc.).
- No runtime behavior changes — this is a refactor to import paths to address build/bundling and resolution issues.

2025-12-03 - 2.0.3 - fix(dependencies)

Bump dependencies and developer tooling versions

- Upgrade lucide from ^0.553.0 to ^0.555.0
- Bump @git.zone/tsbuild from ^3.1.0 to ^3.1.2
- Bump @git.zone/tsbundle from ^2.5.2 to ^2.6.2
- Bump @git.zone/tstest from ^2.8.1 to ^3.1.3
- Bump @git.zone/tswatch from ^2.2.1 to ^2.2.2
- Upgrade @types/node from ^22.0.0 to ^24.10.1
- Patch release: increment package version to 2.0.3

2025-11-30 - 2.0.2 - fix(dees- stepper)

Make step validation abortable and cancel active step listeners when navigating

- Extend IStep.validationFunc signature to accept an optional AbortSignal so validation handlers can be cancelled.
- Store an AbortController on the selected step and pass its signal into validationFunc when invoked.
- Abort the step's AbortController when navigating to the previous or next step to cancel any active listeners or async operations.

2025-11-30 - 2.0.1 - fix(dees- stepper)

Improve dees-stepper visual style and transitions

- Smooth animation: extend .step transition duration and use a cubic-bezier curve for smoother motion.
- Add .step.entrance class with a shorter easing for entrance animations to keep entrance timing distinct.

- Visual tweaks: reduce border-radius from 18px to 12px and increase inner content padding to 32px.
- Color and border updates: adjust background and border colors for light/dark themes to more consistent values.
- Shadow simplification: replace theme-dependent heavy shadows with a single subtle shadow (0 8px 32px rgba(0,0,0,0.4)).
- Remove selected-state border/box-shadow overrides (selection visuals simplified).
- Remove background-clip: padding-box to simplify rendering.

2025-11-17 - 2.0.0 - BREAKING CHANGE(decorators)

Migrate to TC39 standard decorators (accessor) across components, update tsconfig and bump dependencies

- Replaced experimental decorator-backed class fields with the TC39-compatible "accessor" form across ~69 web component files (properties and state fields) to follow Lit 3.x recommendations.
- Updated tsconfig.json to remove experimentalDecorators and useDefineForClassFields, aligning compiler settings with the standard decorators migration.
- Fixed optional/nullable fields to explicit `Type | undefined = undefined` where necessary to preserve runtime behavior and typing.
- Adjusted/remove usages of some non-reactive decorators/@query patterns to be compatible with the new decorator model (notable changes in a few components).
- Bumped several dependencies and devDependencies (examples: @design.estate/dees-domtools, @design.estate/dees-element, @design.estate/dees-wcctools, @git.zone/tsbuild, @git.zone/tstest, apexcharts, lucide).
- Added migration notes and testing summary to readme.hints.md documenting the TC39 decorators migration and verification steps.

2025-10-23 - 1.12.6 - fix(dependencies)

Bump FontAwesome to ^7.1.0 and add local claude settings

- Updated @fortawesome packages (@fortawesome/fontawesome-svg-core, @fortawesome/free-brands-svg-icons, @fortawesome/free-regular-svg-icons,

@fortawesome/free-solid-svg-icons) to ^7.1.0 in package.json

- Added .claude/settings.local.json to configure local Claude/tooling permissions for repository operations

2025-09-23 - 1.12.5 - fix(ci)

Add local permissions settings for development

- Adds a new local settings file: .claude/settings.local.json
- Provides explicit permission entries for development tasks (allow running pnpm scripts, reading files, searching/replacing patterns, activating project, and helper tooling)
- Intended for local dev environment to enable tool automation without changing repository code

2025-09-20 - 1.12.4 - fix(ci)

Add local assistant settings to enable permitted dev tooling commands

- Add a local assistant settings file to configure allowed development tooling commands.
- Allows running pnpm scripts, file read/search/replace operations and other local project helper actions.
- Local configuration only — does not change library code or public API.

2025-09-19 - 1.12.3 - fix(dees-input-fileupload)

Show selected files inside dropzone and improve file upload UX

- Render the selected file list inside the dropzone container so files are displayed inline with the drop area
- Add dropzone--has-files class and styles to visually indicate when files are present
- Avoid opening the file selector when clicking on the browse button or inside the file list (prevents accidental re-opening)
- Refine file list and file-row styles (sizes, paddings, border radius, hover/background behavior and thumbnail/icon sizes) for a more compact and consistent appearance
- Simplify empty-state handling by returning an empty template when no files are present (file list is only rendered when files exist)

2025-09-18 - 1.12.2 - fix(dees-input-wysiwyg)

Integrate output format preview into WYSIWYG demo; update plan and add local dev settings

- Wire output format preview into the WYSIWYG demo (`ts_web/elements/dees-input-wysiwyg.demo.ts`) by calling `setupOutputFormatDemo(editors.meeting, editors.recipe)` so HTML/Markdown preview controls are initialized.
- Update `readme.plan.md`: mark the Output Formats review tasks as completed and document that preview controls were added.
- Add a local settings file to allow running local tooling tasks (grants permission for `pnpm run` scripts and related local commands).
- No library API or runtime component behavior changed — this is a demo/documentation and local-settings update.

2025-09-18 - 1.12.1 - fix(ci)

Add local settings to allow running `pnpm` scripts and enable dev chat permission

- Add a repository-local settings file granting permission to run `pnpm` scripts (`Bash(pnpm run:*)`) for development tooling.
- Enable the `mcp_zen_chat` permission for local dev workflows.

2025-09-18 - 1.12.0 - feat(dees-stepper)

Revamp `dees-stepper`: modern styling, new steps and improved navigation/validation

- Visual refresh for `dees-stepper`: updated card shapes, shadows, refined borders and stronger selected-state visuals for a modern shadcn-inspired look
- Improved transitions and animations (`transform`, `box-shadow`, `filter`) for smoother step selection and show/hide behavior
- Expanded default/demo steps: replaced small sample with a richer multi-step flow (Account Setup, Profile Details, Contact Information, Team Size, Goals, Brand Preferences, Integrations, Review & Launch)

- Enhanced step interactions: safer goNext/goBack handling with boundary checks and reset of validation flags to avoid stale validation state
- Better toolbar/controls placement for stepper demo (spacing, counters, accessible back control) and improved keyboard/UX affordances
- Minor documentation and meta updates: readme.plan.md extended with dees-stepper plan items and added .claude/settings.local.json

2025-09-18 - 1.11.8 - fix(ci)

Add local tool permissions config to allow running pnpm scripts and enable mcp_zen_chat

- Add local settings file to grant permission to run pnpm scripts (Bash(pnpm run:*))
- Enable mcp_zen_chat permission in local tool settings

2025-09-16 - 1.11.7 - fix(readme)

Expand README with comprehensive component documentation, examples and developer guide; add local Claude settings

- Expanded README substantially: installation, component overview, detailed component docs, usage examples, demos and developer guidance
- Updated many example snippets and API usage examples (icons, inputs, editor, forms, overlays, charts, etc.) to be more explicit and consistent
- Added .claude/settings.local.json to configure local Claude permissions for repository tooling
- No runtime or library code changes — documentation and demo content only

2025-09-16 - 1.11.6 - fix(dees-table)

Improve Lucene range comparisons, pin monaco-editor to 0.52.2, and add local dev metadata

- Fix lucene inRange behavior to correctly compare homogeneous types (strings, numbers, dates) and fall back to string comparison when needed (ts_web/elements/dees-table/lucene.ts).
- Pin monaco-editor to 0.52.2 in package.json to avoid a breaking upgrade regression observed with ^0.53.0.

- Add local development/tooling metadata and conveniences: `.claude/settings.local.json` (tool permissions) and `.serena/` memory files (`done_checklist`, `project_overview`, `style_and_conventions`, `suggested_commands`).
- Minor housekeeping: update project dev docs / memories to capture build/test/checklist guidance.

2025-09-16 - 1.11.5 - fix(ci)

Add local Claude agent settings for CI tooling

- Add `.claude/settings.local.json` to configure local Claude agent permissions
- Allow Bash commands matching `pnpm run:*` and the `mcp_zen_chat` permission for development tooling

2025-09-10 - 1.11.4 - fix(readme)

Rewrite and expand README with Quick Start, feature highlights, demos and usage examples; add local Claude settings file

- Completely rewritten and reorganized README: added Quick Start, component highlights, usage examples, demos, development workflow, troubleshooting and links.
- Added `.claude/settings.local.json` with local Claude permission configuration.

2025-09-08 - 1.11.3 - fix(dees-input-list)

Prevent list animations from affecting scroll bounds and fix content-visibility issues in `dees-input-list`; add local developer settings

- `dees-input-list`: add `overflow:hidden` to list items to prevent animations from altering scroll bounds and causing visual/scroll glitches
- `dees-input-list`: force `content-visibility/contain` to `visible/none` to avoid unexpected scrolling/layout issues when items animate
- Add `.claude/settings.local.json` with local developer permissions (allows running `pnpm` scripts via Claude-local tooling)

2025-09-07 - 1.11.2 - fix(DeesFormSubmit)

Make form submit robust by locating nearest dees-form via closest(); add local CLAUDE settings

- Fix: DeesFormSubmit.submit now walks up the DOM with closest('dees-form') to find and call gatherAndDispatch on the parent form. This fixes cases where the submit button is slotted or not a direct child of the form.
- Chore: Add .claude/settings.local.json to permit running pnpm scripts in the local CLAUDE environment (allows Bash(pnpm run:*)).

2025-09-06 - 1.11.1 - fix(dees- input-text)

Normalize Lucide icon names for password toggle

- Updated password visibility toggle icons in dees-input-text from 'lucide:eye'/'lucide:eye-off' to 'lucide:Eye'/'lucide:EyeOff' to match Lucide exports and avoid missing icon rendering.

2025-09-05 - 1.11.0 - feat(dees- icon)

Add full icon list and improve dees-icon demo with copy-all functionality and UI tweaks

- Added readme.icons.md containing 1900+ icon identifiers (FontAwesome + Lucide) for easy reference and tooling
- Enhanced ts_web/elements/dees-icon.demo.ts: added a 'Copy All Icon Names' button that copies prefixed icon names (fa:..., lucide:...) to the clipboard and shows temporary feedback
- Updated demo presentation: prefixed displayed icon names (fa: / lucide:), improved search-container spacing and added button styling for better UX
- Changes are documentation/demo only — no production runtime component logic changed

2025-09-05 - 1.10.12 - fix(dees-simple-appdash)

Fix icon rendering in dees-simple-appdash to respect provided icon strings

- dees-simple-appdash: stop forcing a 'lucide:' prefix when rendering view icons — use the icon string as provided.
- Prevents incorrect/missing icons when the iconName already includes a library prefix (e.g. 'fa:' or 'lucide:').

2025-09-05 - 1.10.11 - fix(dees-simple-appdash)

Bump deps and fix dees-simple-appdash icon binding and terminal sizing

- Updated runtime dependencies: @design.estate/dees-element -> ^2.1.2, @design.estate/dees-wcctools -> ^1.1.1, @fortawesome/* -> ^7.0.1, apexcharts -> ^5.3.4, lucide -> ^0.542.0 (compatibility/security/stability updates)
- Updated dev tooling: @git.zone/tsbuild -> ^2.6.8, @git.zone/tstest -> ^2.3.6, @git.zone/tswatch -> ^2.2.1
- Fix: dees-simple-appdash — use proper string interpolation for lucide icon properties (prevents incorrect icon rendering)
- Fix: dees-simple-appdash — enforce terminal maxWidth/maxHeight to avoid overflow and improve layout stability
- Cosmetic: small style/behavior tweaks to dees-simple-appdash (logout/terminal/wifi icon bindings corrected)

2025-06-29 - 1.10.10 - improve(dees-dashboardgrid, dees-input-wysiwyg)

Enhanced dashboard grid component with advanced spacing and layout features inspired by gridstack.js

Dashboard Grid improvements:

- Improved margin system supporting uniform or individual margins (top, right, bottom, left)
- Added collision detection to prevent widget overlap during drag operations
- Implemented auto-positioning for new widgets to find first available space
- Added compact() method to eliminate gaps and compress layout vertically or horizontally
- Enhanced resize constraints with minW, maxW, minH, maxH support
- Added optional grid lines visualization for better layout understanding
- Improved resize handles with better visibility and hover states
- Added RTL (right-to-left) layout support
- Implemented cellHeightUnit option supporting 'px', 'em', 'rem', or 'auto' (square cells)
- Added configurable animation with enableAnimation property
- Enhanced demo with interactive controls for testing all features
- Better calculation of widget positions accounting for margins between cells
- Added findAvailablePosition() for intelligent widget placement
- Improved drag and resize calculations for pixel-perfect positioning

WYSIWYG editor drag and drop fixes:

- Fixed drop indicator positioning to properly account for block margins
- Added defensive checks in drag event handlers to prevent potential crashes
- Improved updateBlockPositions with null checks and error handling
- Updated drop indicator calculation to use simplified margin approach
- Fixed drop indicator height to match the exact space occupied by dragged blocks
- Improved drop indicator positioning algorithm to accurately show where blocks will land
- Simplified visual block position calculations accounting for CSS transforms
- Enhanced margin calculation to use correct values based on block type (16px for paragraphs, 24px for headings, 20px for code/quotes)
- Fixed index calculation issue when dragging blocks downward by adjusting target index for excluded dragged block

2025-06-28 - 1.10.9 - feat(dees-dashboardgrid)

Add new dashboard grid component with drag-and-drop and resize capabilities

- Created dees-dashboardgrid component for building flexible dashboard layouts
- Features drag-and-drop functionality for rearranging widgets
- Includes resize handles for adjusting widget dimensions
- Supports configurable grid properties (columns, cell height, gap)
- Provides widget locking and editable mode controls

- Styled with shadcn design principles
- No external dependencies - built with native browser APIs
- Emits events for widget movements and resizes
- Includes comprehensive demo with sample dashboard widgets

2025-06-27 - 1.10.8 - feat(ui-components)

Update multiple components with shadcn-aligned styling and improved animations

- Updated dees-modal with shadcn colors, borders, and subtle shadows
- Updated dees-chips with shadcn styling and fixed selection logic bug
- Updated dees-dataview-codebox with shadcn syntax highlighting colors and responsive label layout
- Updated dees-input-multitoggle with transparent blue indicator and smooth animations
- Updated dees-appui-tabs with animated sliding indicator for both horizontal and vertical layouts
- Fixed indicator positioning to be perfectly centered on tab content
- Indicator width is content width + 8px for minimal visual padding
- Fixed tab content centering by using consistent padding (12px → 16px on all sides)
- Fixed icon rendering by correcting property name from `.iconName` to `.icon`
- Added visual separators between tabs for better distinction
- Added subtle hover backgrounds for improved interactivity
- Refactored tabs component code for better maintainability and elegance
- Updated dees-appui-activitylog with shadcn-aligned styling:
 - Updated background and text colors to match shadcn palette
 - Enhanced topbar with better spacing and typography
 - Improved activity entries with subtle hover states and better spacing
 - Added activity type icons with color-coded backgrounds (login, logout, view, create, update)
 - Added date separators ("Today", "Yesterday") for better temporal organization
 - Enhanced streaming indicators with animated pulse effect
 - Redesigned searchbox with modern input styling, search icon, and focus states
 - Added custom scrollbar styling for consistency
 - Updated timestamps to be more subtle with tabular number formatting
 - Refined shadow effects for better visual hierarchy
 - Added subtle box shadow to component for depth
 - Added fade-in animation for new activity entries
 - Improved user name highlighting with better typography
 - Updated context menu with more relevant actions
- Improved overall spacing and visual consistency across components

2025-06-27 - 1.10.1 - fix(modal)

Improve modal overscroll behavior by adding 'overscroll-behavior: contain' to content container

- Added 'overscroll-behavior: contain' to .modal .content to ensure proper scroll containment
- Applied overscroll-behavior in modal container for enhanced responsiveness on mobile and desktop

2025-06-26 - 1.10.0 - feat(dees-modal)

Add mobileFullscreen option to modals for full-screen mobile support

- Introduced a new boolean property 'mobileFullscreen' in ts_web/elements/dees-modal.ts
- Updated modal CSS under the media query to apply 'mobile-fullscreen' class, allowing full viewport modals on mobile devices
- Extended modal style rules to include adjustments for margin, border-radius, and maximum heights on smaller screens

2025-06-26 - 1.9.9 - fix(dees-input-multitoggle, dees-input-typelist)

Replace dynamic import with static import for demo functions in dees-input-multitoggle and dees-input-typelist

- Converted `await import('./dees-input-multitoggle.demo.js')` to a direct static import.
- Converted `await import('./dees-input-typelist.demo.js')` to a direct static import to improve build performance and clarity.

2025-06-26 - 1.9.8 - fix(deps, windowlayer)

Update dependency versions and adjust dees-windowlayer CSS to add pointer-events fix

- Bump @design.estate/dees-wcctools from ^1.0.98 to ^1.0.101
- Bump @tiptap packages from 2.22.3 to 2.23.0
- Bump lucide from ^0.522.0 to ^0.523.0
- Bump @git.zone/tsbundle from ^2.4.0 to ^2.5.1 and tswatch from ^2.0.37 to ^2.1.2
- Add 'pointer-events: none' to dees-windowlayer CSS to improve overlay behavior

2025-06-22 - 1.9.0 - feat(form-inputs)

Improve form input consistency and auto spacing across inputs and buttons

- Add an 'insideForm' property to dees-button for auto-detection and proper margin adjustment in forms.
- Update dees-input-radio to include a 'name' property so that radio buttons in the same group are mutually exclusive.
- Enhance dees-form to group radio inputs properly when collecting form data.
- Revise readme.hints.md and readme.plan.md to document changes and provide guidance for dees-input-radio.
- Update demos for dees-button and dees-form to showcase correct spacing in vertical and horizontal layouts.

2025-06-20 - 1.8.20 - fix(deps)

Update dependency versions: bump @design.estate/dees-domtools from ^2.1.1 to ^2.3.3, @design.estate/dees-element from ^2.0.42 to ^2.0.44, lucide from ^0.515.0 to ^0.518.0, and @git.zone/tsbundle from ^2.0.15 to ^2.4.0

- Upgrade @design.estate/dees-domtools from ^2.1.1 to ^2.3.3
- Upgrade @design.estate/dees-element from ^2.0.42 to ^2.0.44
- Upgrade lucide from ^0.515.0 to ^0.518.0
- Upgrade @git.zone/tsbundle from ^2.0.15 to ^2.4.0

2025-06-10 - 1.8.1 - fix(dees-statsgrid)

Adjust stats grid styling for better alignment and improved visualizations in gauge and trend tiles.

- Center-align tile header elements by setting align-items to center and ensuring full width.
- Increase tile content height to 90px and center its content.
- Update gauge visualization: reduce circle radius from 40 to 30, adjust stroke dasharray (from 251.2 to 188.5), and decrease gauge text font size.
- Refine trend chart layout: set trend-svg height to 40px, center trend value and adjust typography to larger, bolder text.
- Ensure overall grid responsiveness with adjusted gap and column sizing.

2025-04-25 - 1.8.0 - feat(dees-pagination)

Add new pagination component to the library along with its demo and integration in the main export.

- Introduced dees-pagination component with support for various page range scenarios.
- Created demo file to showcase pagination with both small and large sets of pages.
- Updated the module's index to export the new pagination component.

2025-04-22 - 1.7.0 - feat(dees-searchbar)

Add dees-searchbar component with live search and filter demo

- Introduces a new dees-searchbar element with an input field, a search button, and filters
- Wires up events for 'search-changed' and 'search-submit' to provide real-time feedback
- Adds a demo file to showcase usage and logging of search events

2025-04-22 - 1.6.0 - feat(documentation/dees-heading)

Add codex documentation overview and dees-heading component demo

- Introduce contributor overview doc (`codex.md`), now consolidated into (`readme.info.md`) to provide a high-level overview of project layout, component patterns, and build workflow
- Add and update dees-heading component with demo to support multiple heading levels and horizontal rule styles
- Update component export index to include dees-heading

2025-04-18 - 1.5.6 - fix(dependencies)

Bump dependency versions and update demo code references

- Upgrade `@design.estate/dees-element` from `^2.0.39` to `^2.0.41`
- Upgrade `@tsclass/tsclass` from `^4.4.0` to `^9.0.0`
- Upgrade `lucide` from `^0.488.0` to `^0.501.0`
- Update `@types/node` from `^22.10.7` to `^22.14.1`
- Update dees-icon demo: scope search to demo container and adjust hover scaling
- Replace `resolveExec` with `directives.resolveExec` in dees-table for proper rendering

2025-04-12 - 1.5.5 - fix(catalog)

No code or documentation changes were detected. This commit records an empty update in commit information and confirms that the current state remains stable.

- Verified that there are no modifications in source, documentation, or demos
- Commit metadata and build configuration remain unchanged

2025-04-11 - 1.5.4 - fix(readme)

Update readme with company and trademark guidelines, clarifying legal usage without exposing licensing details.

- Added sections detailing company information and trademark guidelines.
- Outlined legal disclaimers for trademark usage.

2025-04-11 - 1.5.3 - fix(readme)

Update readme.md: remove redundant usage section and refine component documentation with improved examples.

- Removed the standalone manual import and usage example for components.
- Added refined examples demonstrating both basic and option-based usage (e.g. for DeesButton).
- Improved markdown formatting and consistency across component documentation.

2025-04-11 - 1.5.3 - fix(readme)

Update readme.md for clearer documentation: removed redundant 'Usage' section and refined component examples (e.g., DeesButton's basic and options usage) for improved clarity and consistency.

- Removed standalone usage example showing manual import and creation of components
- Added refined examples demonstrating both basic and option-based usage of components
- Improved overall readme formatting and consistency across component documentation

2025-04-11 - 1.5.2 - fix(ci)

Remove obsolete GitLab CI configuration file

- Deleted .gitlab-ci.yml as the CI pipeline configuration is now managed elsewhere.
- Cleaned up CI stages for security, testing, release, and metadata.

2025-04-11 - 1.5.1 - fix(readme)

Update readme with comprehensive reference documentation: add a usage snippet for components like DeesButton, introduce a detailed overview table of all component categories, and enhance documentation sections for each component group.

- Added a code example showing how to import and use DeesButton.
- Introduced a components overview table that categorizes Core UI, Forms, Layout, Data Display, Visualization, Dialogs & Overlays, Navigation, and Development components.
- Expanded detailed documentation with usage examples for each component type.
- Reorganized content to improve clarity and ease of navigation for developers.

2025-04-11 - 1.5.0 - feat(badge)

Add dees-badge component with demo file and update packageManager field in package.json

- Introduce a new badge component allowing different types (default, primary, success, warning, error) with an optional rounded style
- Provide a demo for the badge component
- Export the badge component in the main elements index
- Update package.json to include an explicit packageManager field

2025-01-20 - 1.4.1 - fix(dependencies)

Update dependency versions for smartpromise, webcontainer/api, tapbundle, and @types/node

- Update @push.rocks/smartpromise to version ^4.2.0
- Downgrade @webcontainer/api to version 1.2.0
- Update @push.rocks/tapbundle to version ^5.5.6
- Update @types/node to version ^22.10.7

2025-01-20 - 1.4.0 - feat(dees-terminal)

Enhanced the dees-terminal component to support environment variable settings and improved setup command execution.

- Added environment property to pass custom environment variables.
- Introduced webcontainerDeferred to handle the promise for web container creation.
- Enhanced demo to illustrate environment variable usage.
- Improved async interaction with the terminal for setting environment variables and executing setup commands.

2025-01-15 - 1.3.4 - fix(chart)

Fix chart rendering and appearance issues in the DeesChartArea component.

- Resolved issues with chart dimensions calculation based on padding.
- Adjusted grid and axis lines appearance for better visibility.
- Updated tooltip and grid line styling for better accessibility.
- Improved series data representation as time-series for more accurate display.

2024-12-17 - 1.3.3 - fix(dees-input-multitoggle)

Add missing TypeScript declaration for dees-input-multitoggle

- Added a missing declaration to the HTMLTagNameMap for 'dees-input-multitoggle' element.

2024-12-09 - 1.3.2 - fix(metadata)

Updated package metadata and readme for better project description and structure.

- Updated package.json and npmextra.json with a detailed project description and list of keywords.
- Enhanced readme.md with installation instructions, component usage examples, and detailed component descriptions for clarity.

2024-11-07 - 1.3.1 - fix(DeesSimpleAppDash)

Fix: add border to controlbar in DeesSimpleAppDash

- Fixed the missing border at the top of the controlbar in DeesSimpleAppDash.

2024-11-07 - 1.3.0 - feat(dees-simple-appdash)

Enhance responsive styling and terminal setup command

- Added a new property `terminalSetupCommand` for configuring terminal setup commands.
- Improved responsive styling and positioning for components to achieve a fluid layout.
- Fixed layout shifts by switching positions to `absolute` for `appbar` and `appcontent`.

2024-10-07 - 1.2.0 - feat(index.ts)

Add export for colors module in index.ts

- The index.ts file now exports the colors module, making color utilities available for external use.

2024-10-06 - 1.1.13 - fix(dees-button)

Fix styling issue in button component.

- Moved the `.button.disabled` styling block to its correct position after the `.button.highlighted` block.

2024-10-06 - 1.1.12 - fix(dees-button)

Fix reflect attribute for disabled property on dees-button component

- Added `reflect: true` to the 'disabled' property ensuring changes reflect in the DOM attribute.

2024-10-05 - 1.1.11 - fix(DeesStepper)

Adjusted CSS properties in DeesStepper component

- Increased border-radius from 8px to 16px for step container elements
- Adjusted font-size and font-weight for the title in the step container to improve readability

2024-10-04 - 1.1.10 - fix(dependencies)

Reverted @webcontainer/api version

- Changed @webcontainer/api version from ^1.3.0 to 1.2.0 in package.json

2024-10-04 - 1.1.9 - fix(dependencies)

Update various dependencies for compatibility and stability.

- Update @design.estate/dees-domtools to version ^2.0.61
- Update @design.estate/dees-element to version ^2.0.39
- Update @webcontainer/api to version ^1.3.0
- Update apexcharts to version ^3.54.0
- Update monaco-editor to version ^0.52.0
- Update pdfjs-dist to version ^4.6.82
- Update @push.rocks/tapbundle to version ^5.3.0
- Update @types/node to version ^22.7.4

2024-09-02 - 1.1.8 - fix(dees-simple-appdash)

Corrected viewTab active background color

- Corrected the background color of the viewTab in active state for better visual consistency.

2024-09-02 - 1.1.7 - fix(dependencies)

Update dependencies to their latest versions

- Update dependency versions in package.json:
 - ◦ @design.estate/dees-element from ^2.0.34 to ^2.0.36
 - ◦ @fortawesome/fontawesome-svg-core from ^6.5.2 to ^6.6.0
 - ◦ @fortawesome/free-brands-svg-icons from ^6.5.2 to ^6.6.0
 - ◦ @fortawesome/free-regular-svg-icons from ^6.5.2 to ^6.6.0
 - ◦ @fortawesome/free-solid-svg-icons from ^6.5.2 to ^6.6.0
 - ◦ @tsclass/tsclass from ^4.0.63 to ^4.1.2
 - ◦ apexcharts from ^3.49.2 to ^3.53.0
 - ◦ highlight.js from 11.9.0 to 11.10.0
 - ◦ monaco-editor from ^0.50.0 to ^0.51.0
 - ◦ pdfjs-dist from ^4.3.136 to ^4.5.136
 - ◦ @git.zone/tsbuild from ^2.1.82 to ^2.1.84
 - ◦ @push.rocks/tapbundle from ^5.0.23 to ^5.0.24
 - ◦ @types/node from ^20.14.9 to ^22.5.2

2024-07-01 - 1.1.6 - fix(dees-dataview-codebox)

Corrected the font-family order for better font rendering.

- Corrected the font-family order in dees-dataview-codebox.ts to ensure 'Intel One Mono' is prioritized over 'Geist Mono'.

2024-07-01 - 1.1.5 - fix(dees-dataview-codebox)

Adjusted line number font weight in codebox

- Changed the line number font weight from 400 to 200 in the codebox for better visual alignment.

2024-07-01 - 1.1.4 - fix(UI)

Fixed font-family order for code and value elements

- Updated font-family order in dees-dataview-codebox.ts
- Updated font-family order in dees-dataview-statusobject.ts

2024-07-01 - 1.1.3 - fix(dees-dataview-codebox)

Adjusted codebox font weight and font family.

- Changed font weight from 200 to 400 for better readability.
- Updated font-family to include 'Geist Mono' for uniform appearance.

2024-06-30 - 1.1.2 - fix(elements)

Fix various UI components and improve styles

- Updated styles for multiple components for better appearance in dark mode.
- Added missing event handling in `dees-input-fileupload` component.
- Implemented tooltip support in `dees-label` component.

- Fixed alignment issues in `dees-appui-*` components.
- Resolved various accessibility issues in form elements.

2024-06-30 - 1.1.1 - fix(build)

Fix build script by updating source folders.

- Updated build script to correctly reference 'tsfolders' folder.

2024-06-30 - 1.1.0 - feat(project dependencies)

Updated various dependencies and internal fonts

- Updated dependencies in package.json with the latest versions.
- Updated font-family references from 'Roboto' and 'Inter' to 'Geist Sans' across multiple components.

2024-04-20 - 1.0.289 - Documentation Update

Updated the project documentation.

2024-02-05 to 2024-02-13 - 1.0.277 to 1.0.288 - Core Fixes

Series of core updates and patches.

2024-01-22 to 2024-02-05 - 1.0.257 to 1.0.276 - Core Fixes

More core updates and patches.

2024-01-21 - 1.0.256 to 1.0.257 - Core Fixes

Minor bugfixes for core functionalities.

2024-01-18 - 1.0.246 - Core Fixes

Bugfixes for core functionalities.

2024-01-15 - 1.0.242 to 1.0.245 - Core Fixes

Various core bugfixes.

2024-01-11 - 1.0.241 to 1.0.242 - Core Fixes

Bugfixes for core functionalities.

2024-01-10 - 1.0.240 to 1.0.241 - Core Fixes

Minor core fixes and improvements.

2024-01-09 - 1.0.239 to 1.0.240 - Core Fixes

Core updates and patches.

2023-12-26 - 1.0.238 to 1.0.239 - Core Fixes

Updates to core functionalities.

2023-12-20 - 1.0.236 to 1.0.238 - Core Fixes

Bugfixes and updates for core functionalities.

2023-12-08 - 1.0.233 to 1.0.236 - Core Fixes

Multiple updates to core functionalities.

2023-11-29 - 1.0.231 to 1.0.232 - Core Fixes

Minor core patches.

2023-10-31 - 1.0.229 to 1.0.230 - Core Fixes

Various updates and fixes for core.

2023-10-24 - 1.0.228 to 1.0.229 - Core Fixes

More core updates and patches.

2023-10-23 - 1.0.226 to 1.0.228 - Core Fixes

Series of updates and fixes for core functionalities.

2023-10-20 - 1.0.223 to 1.0.225 - Core Fixes

Minor updates and fixes for core.

2023-10-18 - 1.0.222 to 1.0.223 - Core Fixes

Core patches and bugfixes.

2023-10-17 - 1.0.221 to 1.0.222 - Core Fixes

Further updates to core functionalities.

2023-10-12 - 1.0.220 to 1.0.221 - Core Fixes

Minor core updates.

2023-10-11 - 1.0.219 to 1.0.220 - Core Fixes

Bugfixes and updates for core.

2023-10-07 - 1.0.217 to 1.0.219 - Core Fixes

Series of core updates and patches.

2023-10-05 - 1.0.216 to 1.0.217 - Core Fixes

Minor updates for core functionalities.

2023-09-22 - 1.0.212 to 1.0.215 - Core Fixes

Various updates and fixes for core.

2023-09-20 - 1.0.210 to 1.0.211 - Core Fixes

Series of core patches.

2023-09-18 - 1.0.209 to 1.0.210 - Core Fixes

Bugfixes and improvements to core.

2023-09-17 - 1.0.207 to 1.0.209 - Core Fixes

Minor updates and fixes.

2023-09-15 - 1.0.203 to 1.0.206 - Core Fixes

Series of core improvements and fixes.

2023-09-14 - 1.0.199 to 1.0.202 - Core Fixes

Patches for core functionalities.

2023-09-13 - 1.0.194 to 1.0.198 - Core Fixes

Various bugfixes and updates for core.

2023-09-12 - 1.0.190 to 1.0.193 - Core Fixes

Minor core patches.

2023-09-09 - 1.0.188 to 1.0.189 - Core Fixes

Updates and bugfixes for core.

2023-09-07 - 1.0.186 to 1.0.187 - Core Fixes

Core patches and improvements.

2023-09-04 - 1.0.184 to 1.0.185 - Core Fixes

Series of updates and bugfixes for core.

2023-09-01 - 1.0.180 to 1.0.183 - Core Fixes

Various core updates and fixes.

2023-08-28 - 1.0.177 to 1.0.179 - Core Fixes

Minor bugfixes and updates.

2023-08-26 - 1.0.176 to 1.0.177 - Core Fixes

Bugfixes for core functionalities.

2023-08-20 - 1.0.173 to 1.0.175 - Core Fixes

Series of updates and fixes for core.

2023-08-08 - 1.0.172 to 1.0.173 - Core Fixes

Minor core updates.

2023-08-07 - 1.0.171 to 1.0.172 - Core Fixes

Various updates to core.

2023-04-12 - 1.0.169 to 1.0.170 - Core Fixes

Bugfixes for core functionalities.

2023-04-10 - 1.0.163 to 1.0.168 - Core Fixes

Series of core updates and patches.

2023-04-07 - 1.0.160 to 1.0.162 - Core Fixes

Various updates for core functionalities.

2023-04-06 - 1.0.155 to 1.0.159 - Core Fixes

Bugfixes and improvements to core.

2023-03-27 - 1.0.150 to 1.0.154 - Core Fixes

Minor updates and patches for core functionalities.

2023-03-09 - 1.0.149 to 1.0.150 - Core Fixes

Bugfixes for core functionalities.

2023-01-17 - 1.0.146 to 1.0.148 - Core Fixes

Updates and fixes for core.

2023-01-16 - 1.0.144 to 1.0.145 - Core Fixes

Series of updates and bugfixes for core.

2023-01-13 - 1.0.134 to 1.0.143 - Core Fixes

Various core patches and improvements.

2023-01-12 - 1.0.132 to 1.0.133 - Core Fixes

Minor fixes for core functionalities.

2023-01-11 - 1.0.126 to 1.0.131 - Core Fixes

Series of updates and fixes for core.

2023-01-09 - 1.0.122 to 1.0.123 - Core Fixes

Minor core patches and updates.

2023-01-07 - 1.0.113 to 1.0.121 - Core Fixes

Updates and bugfixes for core functionalities.

2023-01-03 - 1.0.107 to 1.0.110 - Core Fixes

Series of core updates and fixes.

2022-12-12 - 1.0.105 to 1.0.106 - Core Fixes

Various core patches.

2022-12-07 - 1.0.104 to 1.0.105 - Core Fixes

Bugfixes and improvements to core functionalities.

2022-12-06 - 1.0.103 to 1.0.104 - Core Fixes

Series of updates and patches for core.

2022-08-18 - 1.0.98 to 1.0.102 - Core Fixes

Updates and bugfixes for core functionalities.

2022-07-14 - 1.0.93 to 1.0.94 - Core Fixes

Minor core updates.

2022-06-26 - 1.0.92 to 1.0.93 - Core Fixes

Bugfixes for core functionalities.

2022-06-10 - 1.0.91 to 1.0.92 - Core Fixes

Updates and patches for core.

2022-05-30 - 1.0.90 to 1.0.91 - Core Fixes

Minor updates and bugfixes.

2022-05-26 - 1.0.87 to 1.0.89 - Core Fixes

Series of core updates and patches.

2022-05-24 to 2022-05-20 - 1.0.83 to 1.0.86 - Core Fixes

Various updates and bugfixes for core functionalities.

2022-05-03 - 1.0.79 to 1.0.80 - Core Fixes

Updates and patches for core functionalities.

2022-03-18 - 1.0.74 to 1.0.77 - Core Fixes

Bugfixes and updates for core functionalities.

2022-01-24 - 1.0.73 to 1.0.74 - Core Fixes

Minor core patches.

2021-12-10 - 1.0.71 to 1.0.72 - Core Fixes

Series of updates and fixes for core functionalities.

2021-11-27 - 1.0.70 to 1.0.71 - Core Fixes

Bugfixes and improvements to core.

2021-10-08 - 1.0.68 to 1.0.69 - Core Fixes

Various core updates and patches.

2021-10-07 - 1.0.67 to 1.0.68 - Core Fixes

Minor updates and improvements for core.

2021-09-15 - 1.0.64 to 1.0.66 - Core Fixes

Bugfixes and improvements to core functionalities.

2021-09-14 - 1.0.61 to 1.0.62 - Core Fixes

Minor patches and updates for core.

2021-09-10 - 1.0.57 to 1.0.60 - Core Fixes

Series of updates and fixes for core functionalities.

2021-09-09 - 1.0.54 to 1.0.56 - Core Fixes

Various core updates and patches.

2021-09-01 - 1.0.51 to 1.0.53 - Core Fixes

Minor updates for core functionalities.

2021-08-29 - 1.0.49 to 1.0.50 - Core Fixes

Bugfixes and patches for core.

2021-08-27 - 1.0.44 to 1.0.48 - Core Fixes

Series of updates and improvements for core functionalities.

2021-08-26 - 1.0.42 to 1.0.43 - Core Fixes

Minor updates and bugfixes for core.

2021-08-25 - 1.0.38 to 1.0.41 - Core Fixes

Extracted series of core updates and patches.

2021-08-24 - 1.0.37 - Core Fix

Bugfixes for core functionalities.

2021-08-20 - 1.0.34 to 1.0.36 - Core Fixes

Core functionalities updates and patches.

2021-05-05 - 1.0.32 to 1.0.33 - Core Fixes

Series of core bugfixes.

2021-03-06 - 1.0.29 to 1.0.31 - Core Fixes

Updates and improvements for core functionalities.

2021-02-13 - 1.0.27 to 1.0.28 - Core Fixes

Bugfixes and patches for core functionalities.

2020-12-10 - 1.0.25 to 1.0.26 - Core Fixes

Various core updates and improvements.

2020-12-02 - 1.0.23 to 1.0.24 - Core Fixes

Minor updates for core functionalities.

2020-12-01 - 1.0.21 to 1.0.22 - Core Fixes

Bugfixes and patches for core.

2020-09-13 - 1.0.19 to 1.0.20 - Core Fixes

Updates and bugfixes for core functionalities.

2020-05-11 - 1.0.17 to 1.0.18 - Core Fixes

Core functionalities updates.

Revision #2

Created 2026-03-28 13:06:49 UTC by foss.global Team

Updated 2026-03-29 16:49:16 UTC by foss.global Team