

@signature.digital/c atalog

the catalog for signature.digital

- [readme.md for @signature.digital/catalog](#)
- [changelog.md for @signature.digital/catalog](#)

readme.md for @signature.digital/catalog

A comprehensive catalog of customizable web components designed for building and managing e-signature applications. Built with modern web technologies using LitElement and TypeScript.

Issue Reporting and Security

For reporting bugs, issues, or security vulnerabilities, please visit community.foss.global/. This is the central community hub for all issue reporting. Developers who sign and comply with our contribution agreement and go through identification can also get a code.foss.global/ account to submit Pull Requests directly.

Install

```
npm install @signature.digital/catalog  
# or  
pnpm install @signature.digital/catalog
```

📦 Overview

This package provides three main components for e-signature workflows:

Component	Tag	Description
SignPad	<code><sdig-signpad></code>	Canvas-based signature capture pad
SignBox	<code><sdig-signbox></code>	Complete signing interface with controls
ContractEditor	<code><sdig-contracteditor></code>	Contract document management component

Usage

Basic Import

```
import '@signature.digital/catalog';
```

This registers all custom elements and makes them available for use in your HTML.

SignPad Component

The `<sdig-signpad>` is a canvas-based signature capture component that allows users to draw their signatures directly in the browser.

```
<sdig-signpad></sdig-signpad>
```

API Methods:

```
const signpad = document.querySelector('sdig-signpad');

// Get signature data as point arrays
const data = await signpad.toData();

// Load signature from data
await signpad.fromData(data);

// Export signature as SVG string
const svg = await signpad.toSVG();

// Undo last stroke
await signpad.undo();

// Clear the signature pad
await signpad.clear();
```

SignBox Component

The `<sdig-signbox>` wraps `SignPad` with a complete UI including Clear, Undo, and Submit buttons.

```
<sdig-signbox></sdig-signbox>
```

Events:

```
const signbox = document.querySelector('sdig-signbox');

signbox.addEventListener('signature', (event) => {
  const signatureData = event.detail.signature;
  console.log('Signature captured:', signatureData);
});
```

ContractEditor Component

The `<sdig-contracteditor>` provides contract viewing and editing capabilities using the `IPortableContract` interface from `@signature.digital/tools`.

```
import '@signature.digital/catalog';
import { IPortableContract } from '@signature.digital/tools/interfaces';
import { demoContract } from '@signature.digital/tools/demodata';

const editor = document.querySelector('sdig-contracteditor');
editor.contract = demoContract;
```

Integration Example

Here's a complete example showing all components working together:

```
import '@signature.digital/catalog';
import { LitElement, html, css } from 'lit';
import { customElement } from 'lit/decorators.js';

@customElement('my-signature-app')
class MySignatureApp extends LitElement {
  static styles = css`
    :host {
      display: block;
      max-width: 800px;
    }
  `;
}
```

```

    margin: 0 auto;
  }
`;

render() {
  return html`
    <h2>Please sign below</h2>
    <sdig-signbox @signature=${this.handleSignature}></sdig-signbox>
  `;
}

private handleSignature(e: CustomEvent) {
  console.log('Signature submitted:', e.detail.signature);
  // Process or store the signature
}
}

```

☐☐ Theming

Components support automatic light/dark theme detection and can be customized using CSS custom properties:

```

sdig-signpad, sdig-signbox {
  --main-background-color: #ffffff;
  --line-color: #000000;
  --button-color: #007bff;
}

```

The components use `cssManager.bdTheme()` for automatic theme switching based on system preferences.

☐☐ Requirements

- Modern browser with Custom Elements V1 support
- ECMAScript Modules (ESM) compatible environment
- TypeScript 5.0+ (for development)

📦 Dependencies

- `@design.estate/dees-element` - LitElement-based component framework
- `@signature.digital/tools` - Contract interfaces and demo data
- `signature_pad` - Canvas signature capture library

License and Legal Information

This repository contains open-source code licensed under the MIT License. A copy of the license can be found in the [LICENSE](#) file.

Please note: The MIT License does not grant permission to use the trade names, trademarks, service marks, or product names of the project, except as required for reasonable and customary use in describing the origin of the work and reproducing the content of the NOTICE file.

Trademarks

This project is owned and maintained by Task Venture Capital GmbH. The names and logos associated with Task Venture Capital GmbH and any related products or services are trademarks of Task Venture Capital GmbH or third parties, and are not included within the scope of the MIT license granted herein.

Use of these trademarks must comply with Task Venture Capital GmbH's Trademark Guidelines or the guidelines of the respective third-party owners, and any usage must be approved in writing. Third-party trademarks used herein are the property of their respective owners and used only in a descriptive manner, e.g. for an implementation of an API or similar.

Company Information

Task Venture Capital GmbH Registered at District Court Bremen HRB 35230 HB, Germany

For any legal inquiries or further information, please contact us via email at hello@task.vc.

By using this repository, you acknowledge that you have read this section, agree to comply with its terms, and understand that the licensing of the code does not imply endorsement by Task Venture Capital GmbH of any derivative works.

changelog.md for @signature.digital/catalog

2025-12-18 - 1.2.0 - feat(Icons)

migrate icon usage to the new dees-icon API and integrate collaboration sidebar into the editor

- Replaced deprecated `.iconFA` with `.icon` across multiple components
- Updated lucide icon identifiers to PascalCase/camelCase to match new dees-icon format
- Added `sdig-collaboration-sidebar` component and exported it from `elements/index`
- Integrated a toggleable editor sidebar (`PanelRight`) and wired comment/suggestion navigation & add-comment events in `sdig-contracteditor`
- Added development hints (`readme.hints.md`) documenting dees-icon usage and icon name formats
- Minor UI/styling tweak: `.btn-ghost.active` appearance

2025-12-18 - 1.1.0 - feat(Catalog)

add `ContractEditor` and many editor subcomponents; implement `SignPad` and `SignBox`; update README and bump dependencies

- New contract editor module with many subcomponents added (header, metadata, parties, content, terms, signatures, attachments, collaboration, audit)
- Implemented signature UI components: `sdig-signpad` and `sdig-signbox` (canvas-based signature capture, `undo/clear/export` APIs)
- Reorganized exports (`ts_web/elements/index.ts`) to expose new submodules and barrel files
- Added editor types and smartstate store (`ts_web/elements/sdig-contracteditor/types.ts` and `state.ts`) for editor state management, `undo/redo` and events
- Large README overhaul: improved usage examples, API docs, security/issue reporting section, theming and integration examples
- Dependencies bumped in `package.json` (notable bumps: `@design.estate/*` packages, `@git.zone` tooling, `signature_pad`) and `pnpm` overrides added
- `tsconfig.json` compiler options modified (removed `experimentalDecorators` and `useDefineForClassFields`) — may affect local build configurations

2024-12-19 - 1.0.59 - fix(dependencies)

Update package dependencies and project metadata

- Updated package dependencies to the latest versions in package.json
- Synchronized project description and keywords in npmextra.json with package.json

2023-11-28 - 1.0.55 to 1.0.58 - core updates

Main changes include fixing and updating core functionalities.

- Fixed core issues and updated core functionalities in versions 1.0.55, 1.0.56, and 1.0.57.
- Further updates and improvements were carried out in version 1.0.58.