

@design.estate/dees-document

A comprehensive tool for dynamically generating and rendering business documents like invoices using modern web technologies.

- [readme.md for @design.estate/dees-document](#)
- [changelog.md for @design.estate/dees-document](#)

readme.md for @design.estate/dees- document

A powerful TypeScript framework for dynamically generating professional business documents like invoices with web components and PDF export capabilities. 📄

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.

Installation

```
pnpm install @design.estate/dees-document
```

Features

- 📄 **PDF Generation** - Server-side PDF creation from structured data using Puppeteer
- 📄 **Web Components** - Lit-based custom elements for document rendering
- 📄 **Multi-language Support** - Built-in translation system (EN, DE, ES)
- 📄 **Automatic Pagination** - Smart content overflow handling across pages
- 📄 **QR Payment Codes** - EPC QR codes for SEPA payments
- 📄 **Print Mode** - Optimized rendering for both screen and print
- 📄 **A4 Format** - Precise DIN A4 document dimensions
- 📄 **Theming** - Customizable colors, backgrounds, and branding

Usage

Server-Side PDF Generation

```
import { PdfService } from '@design.estate/dees-document';
import type { TInvoice } from '@tsclass/tsclass/finance';

// Initialize the PDF service
const pdfService = new PdfService({});
await pdfService.start();

// Create invoice data
const invoice: TInvoice = {
  type: 'invoice',
  invoiceType: 'debitnote',
  id: 'INV-2024-001',
  invoiceId: 'INV-2024-001',
  date: Date.now(),
  currency: 'EUR',
  dueInDays: 30,
  reverseCharge: false,
  from: {
    name: 'Your Company GmbH',
    type: 'company',
    status: 'active',
    address: {
      streetName: 'Business Street',
      houseNumber: '123',
      city: 'Berlin',
      country: 'Germany',
      postalCode: '10115',
    },
  },
  sepaConnection: {
    bic: 'DEUTDEFF',
    iban: 'DE89370400440532013000',
  },
  registrationDetails: {
```

```
    vatId: 'DE123456789',
    registrationName: 'Amtsgericht Berlin',
    registrationId: 'HRB 12345',
  },
},
to: {
  name: 'Customer Inc.',
  type: 'company',
  status: 'active',
  address: {
    streetName: 'Client Avenue',
    houseNumber: '456',
    city: 'Munich',
    country: 'Germany',
    postalCode: '80331',
  },
  registrationDetails: {
    vatId: 'DE987654321',
  },
},
items: [
  {
    name: 'Web Development Services',
    unitQuantity: 40,
    unitNetPrice: 95,
    unitType: 'hours',
    vatPercentage: 19,
    position: 1,
  },
  {
    name: 'Hosting (Annual)',
    unitQuantity: 1,
    unitNetPrice: 299,
    unitType: 'item',
    vatPercentage: 19,
    position: 2,
  },
],
subject: 'Invoice INV-2024-001',
```

```

versionInfo: {
  type: 'final',
  version: '1.0.0',
},
};

// Generate PDF
const pdfResult = await pdfService.createPdfFromLetterObject({
  letterData: invoice,
  documentSettings: {
    languageCode: 'EN',
    enableDefaultHeader: true,
    enableDefaultFooter: true,
    enableFoldMarks: true,
    dateStyle: 'long',
  },
});

// Save to file
import { SmartFs, SmartFsProviderNode } from '@push.rocks/smartfs';
const fs = new SmartFs(new SmartFsProviderNode());
await fs.file('./invoice.pdf').write(Buffer.from(pdfResult.buffer));

// Don't forget to stop the service when done
await pdfService.stop();

```

Document Settings

Customize document appearance with `IDocumentSettings`:

```

const documentSettings = {
  // Language for translations
  languageCode: 'DE', // 'EN' | 'DE' | 'ES'

  // Layout options
  enableDefaultHeader: true,
  enableDefaultFooter: true,
  enableFoldMarks: true,
  enableTopDraftText: true,

```

```
enableInvoiceContractRefSection: true,

// Display options
vatGroupPositions: true,
dateStyle: 'long', // 'short' | 'medium' | 'long' | 'full'

// Theming
theme: {
  colorPrimaryForeground: '#ffffff',
  colorPrimaryBackground: '#e4002b',
  colorAccentForeground: '#333333',
  colorAccentBackground: '#f5f5f5',
  pageBackground: 'url(your-watermark.png)',
  coverPageBackground: 'url(your-cover.png)',
},
};
```

Web Component Usage

For browser-based document viewing:

```
import '@design.estate/dees-document/web';

// In your HTML/Lit template
html`
  <dedocument-viewer
    .letterData=${invoiceData}
    .documentSettings=${documentSettings}
  ></dedocument-viewer>
`;

// Or render the document directly
html`
  <dedocument-dedocument
    .letterData=${invoiceData}
    .documentSettings=${documentSettings}
  ></dedocument-dedocument>
`;
```

Translation System

The package includes a translation system for multi-language document generation:

```
import { translate } from '@design.estate/dees-document/shared';

// Translate document labels
translate('DE', 'invoice@@totalGross'); // "Gesamtbetrag (Brutto)"
translate('EN', 'invoice@@totalGross'); // "Total (Gross)"
translate('ES', 'invoice@@totalGross'); // "Total (Bruto)"
```

Module Exports

The package provides multiple entry points:

```
// Node.js server-side (PDF generation)
import { PdfService } from '@design.estate/dees-document';
import { PdfService } from '@design.estate/dees-document/node';

// Browser web components
import '@design.estate/dees-document/web';

// Shared utilities and types
import { translate, IDocumentSettings } from '@design.estate/dees-document/shared';

// TypeScript interfaces
import type { IDocumentSettings } from '@design.estate/dees-document/interfaces';
```

Document Components

The framework includes these web components:

Component	Description
<code><dedocument-dedocument></code>	Main document container with pagination
<code><dedocument-viewer></code>	Document viewer with controls

Component	Description
<dedocument-page>	Single A4 page with scaling support
<dedocument-pageheader>	Company logo and branding header
<dedocument-letterheader>	Sender/recipient addresses
<dedocument-pagecontent>	Main content area
<dedocument-pagefooter>	Company info and page numbers
<dedocument-contentinvoice>	Invoice-specific content layout
<dedocument-paymentcode>	EPC QR code for SEPA payments

Draft vs Final Documents

Control document version display:

```
const invoice = {
  // ... other fields
  versionInfo: {
    type: 'draft', // Shows watermark "DRAFT" across pages
    version: '0.1.0',
  },
};

// For final documents
const invoice = {
  // ... other fields
  versionInfo: {
    type: 'final', // Clean document without watermark
    version: '1.0.0',
  },
};
```

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 @design.estate/dees- document

2025-12-11 - 2.2.3 - fix(contentinvoice)

Use shared translation.TranslationKey type in DeContentInvoice.translateKey and remove local import

- Replaced local TranslationKey import with plugins.shared.translation.TranslationKey to ensure type consistency
- Updated DeContentInvoice.translateKey parameter type to reference the shared translation type
- Removed an unused type import from ts_web/elements/contentinvoice.ts

2025-12-11 - 2.2.0 - feat(viewer)

Add pagination metadata and thumbnail offset rendering for viewer thumbnails

- Expose per-page pagination metadata (IPagePaginationInfo) from DeDocument during rendering (pageNumber, startOffset, contentLength).
- DeDocument now collects pagination info while paginating and dispatches a 'pagination-complete' CustomEvent with pageCount and paginationInfo (bubbled & composed).
- DeContentInvoice: add renderStartOffset and renderContentLength properties and applyOffsetTrimming to support rendering trimmed content (used for thumbnails).
- DeDocumentViewer: listen for 'pagination-complete', store paginationInfo, and render thumbnail previews by mounting DeContentInvoice instances with precomputed offsets to produce accurate thumbnails.
- Viewer: wire thumbnail generation to use per-page offsets so thumbnails show exactly the content slice for each page.

- Demo data: extended demo invoice items in `ts_shared/demoletter.ts` to better exercise pagination and thumbnail rendering.
- Remove registry override from `.npmrc` (previously pointed to `registry.npmjs.org`).

2025-12-11 - 2.1.1 - fix(viewer)

Improve sidebar resizing UX and update deps

- Add smooth width transition for the thumbnail sidebar to improve resize visuals
- Disable viewport CSS transitions while user is actively resizing the sidebar to avoid janky animations
- Mark viewport as resizing in classes so layout updates skip transitions during drag
- Make sidebar resize mousemove listener passive to improve scrolling/interaction performance
- Bump `@design.estate/dees-wcctools` to `^2.0.0` and `@git.zone/tswatch` to `^2.3.11`

2025-12-11 - 2.1.0 - feat(viewer)

Add smooth zoom and scroll animations to viewer

- Introduce smooth zoom animation: new private `animateZoomTo(targetZoom, duration)` method and `zoomAnimationId` to track/cancel running animations
- Animate transitions for numeric zoom presets and zoom step changes (`handleZoomStep` now uses `animateZoomTo`)
- Cancel ongoing zoom animation when user manipulates the zoom slider
- Make fit-to-width and fit-to-page calculations optionally animate (`calculateFitWidth/calculateFitPage` accept `animate` flag)
- Add animated scrolling helper `animateScrollTo` for smooth page scrolling when requested (used when `smooth=true`)

2025-12-11 - 2.0.6 - fix(DeDocumentViewer)

Account for toolbar and padding when calculating Fit Page zoom in viewer

- `calculateFitPage` now subtracts explicit top (toolbar + padding), bottom, and side paddings from viewport dimensions before computing scale.

- This produces a more accurate zoom level for the Fit Page preset by considering toolbar height and page margins.

2025-12-11 - 2.0.2 - fix(page)

Use theme variables for page styling, make pagecontainer background theme-aware, and use accent background for footer separator

- Extract documentSettings.theme to a local variable in dedocument-page and use safe optional checks when emitting CSS custom properties
- Use theme pageBackground / coverPageBackground values consistently for page background-image fallback logic
- Make pagecontainer background follow the --text-bg-color CSS variable instead of a hard #ffffff color
- Switch footer separator color variable to --color-accent-bg so the footer stripe uses the theme accent background

2025-12-10 - 2.0.1 - fix(core)

Migrate file I/O to @push.rocks/smartfs, adopt TC39 decorators v3 accessor in web components, and update docs/tests

- Added dependency @push.rocks/smartfs and replaced usages of @push.rocks/smartfile.fs with the SmartFs class-based API
- Updated test helpers and test code to use SmartFs (SmartFsProviderNode) and new file read/write methods
- Replaced legacy property declarations with TC39 decorators v3 accessor keyword across web components (ts_web/elements/*)
- Replaced synchronous smartfile directory operations with async SmartFs directory.exists/create flow for .nogit
- Updated README and added readme.hints.md with SmartFs usage examples, decorator guidance, and other developer hints

2025-12-10 - 2.0.0 - BREAKING CHANGE(build)

Upgrade dependencies and dev tooling; adjust TypeScript config

- Bumped runtime dependencies: @design.estate/, @push.rocks/, @git.zone/tsrun (to ^2.0.0), @tsclass/tsclass, puppeteer, @types/node, qrcode types, and others.
- Updated devDependencies: @git.zone/tsbuild, tsbundle, tstest, tswatch to newer releases.
- Removed experimentalDecorators and useDefineForClassFields from tsconfig.json — this may change decorator handling and affect compilation.
- These dependency and config changes may introduce breaking behavior; recommend a major version bump.

2025-01-01 - 1.6.11 - fix(license)

Update copyright notice in license to reflect new ownership

- Updated copyright from Lossless GmbH to Task Venture Capital GmbH.

2024-12-08 - 1.6.10 - fix(core)

Improve stability and performance of document generation

2024-12-08 - 1.6.9 - fix(contentinvoice.ts)

Improve invoice item layout and fix alignment issues.

- Fixed missing border on last invoice item.
- Added highlighted style for invoice lines.
- Corrected alignment and display of invoice item details.
- Included VAT percentage display adjustments.

2024-12-07 - 1.6.8 - fix(rendering and logging)

Removed debug logging from document rendering process.

- Removed console logs from content invoice, document rendering, page scaling, and overflow checking.

2024-12-07 - 1.6.7 - fix(document rendering)

Fixed overflow issues in document and page elements

- Ensured content overflow handling in document.ts
- Adjusted page element overflow settings in page.ts

2024-12-07 - 1.6.6 - fix(page-render)

Fix layout scaling adjustment for page component

- Ensure `font-family` is no longer explicitly set to inherit in `DePage` component.
- Adjust scaling logic to properly set width and overflow based on calculated scale.

2024-12-05 - 1.6.5 - fix(contentinvoice)

Fix VAT group item number formatting and remove custom font style in invoice sums.

- Removed custom font-family style from the invoice sums.
- Corrected VAT group item numbers display by properly formatting and removing trailing comma.

2024-12-05 - 1.6.4 - fix(styling)

Consolidated shared styles for consistent font applied across various components.

- Added a shared style file for consistent font family across components
- Applied shared style to contentinvoice, letterheader, pagecontent, pagefooter, and pageheader components

2024-12-05 - 1.6.3 - fix(ui)

Corrects font family in contentinvoice element.

- Updated the font-family for line items in contentinvoice.ts

2024-12-05 - 1.6.2 - fix(translation)

Corrected missing translation keys for VAT short form across multiple languages.

- Fixed missing 'vatShort' translation for English, German, Spanish, French, and Italian in translation.ts file.

2024-12-05 - 1.6.1 - fix(core)

Ensure consistent project structure and code organization without additional changes.

2024-12-05 - 1.6.0 - feat(documentSettings + translations + vat in itemLines)

Updated dependencies in package.json

- Updated @types/node to version ^22.10.1
- Updated qrcode package to version ^1.5.4

2024-12-02 - 1.5.3 - fix(elements)

Update viewer attributes and fix font integration

- Updated viewer.ts to add missing property decorators for letterData and documentSettings.
- Fixed font loading by correcting URLs in index.html.
- Set demoDocumentSettings default header and footer to false.

2024-12-02 - 1.5.2 - fix(workflow)

Corrected Docker image references and package scope in YAML workflows for compatibility.

- Updated Docker image reference from GitLab to code.foss.global for npmci UI workflows.
- Fixed npm package scope for `@shipzone/npmci` in YAML workflows.
- Minor formatting corrections for better code readability.

2024-12-02 - 1.5.1 - fix(package)

Fix duplicate node export in package.json

- Resolved a duplication issue with the './node' export in package.json exports field.

2024-12-02 - 1.5.0 - feat(core)

Refactor project structure for better modularity and code organization

- Moved shared components to a dedicated 'dist_ts_shared' directory for better separation of concerns.
- Updated import statements across the codebase to align with the new file structure.
- Introduced translations directly under 'ts_shared' for language-specific string management.

2024-12-02 - 1.4.0 - feat(translation)

Add French and Italian translations for document interface

- Expanded translation support by adding French and Italian language translations
- Fixed duplicated text in German final page statement
- Improved Spanish translation of reverse VAT note and final page statement

2024-12-02 - 1.3.2 - fix(core)

Minor updates and optimizations in the output PDF formatting process.

- Improved handling of page content overflow in PDF generation.
- Refined default document settings for consistent rendering.

2024-12-02 - 1.3.1 - fix(documentation)

Updated project description and enhanced documentation in package.json and README

- The package description in package.json was updated to include more features.
- Added more details about setting up and using the PdfService.
- Included comprehensive examples for initializing and generating PDFs.
- Expanded sections covering advanced features, modular components, interactive documents, and localization support.

2024-12-02 - 1.3.0 - feat(translation)

Add multi-language support for document translations.

- Implemented translation interface for document labels.
- Added Spanish translations alongside existing English and German.
- Updated content invoice, page header, and page footer elements to utilize language-specific translations.

2024-12-01 - 1.2.0 - feat(core)

Enhance document generation capabilities with improved modular structure and extended translation support.

- Enhanced document generation framework with a more modular structure.
- Integrated support for translations using JSON-based language maps.
- Optimized PDF generation process for dynamic document content.
- Added properties in 'IDocumentSettings' for language code management.
- Updated README documentation with more comprehensive examples and guidance.

2024-11-30 - 1.1.0 - feat(core)

Enhanced document generation features and added translation capabilities

- Updated document rendering with advanced modular architecture.
- Introduced translation mechanism for document content.
- Improved PDF generation with dynamic settings support.
- Enhanced test cases for PDF generation with complex document structures.

2024-11-27 - 1.0.103 - fix(DeDocument)

Fix rendering of documents by properly moving new pages back to the document container

- Resolved an issue in the DeDocument component where new pages were wrongly managed, causing errors in the rendering of documents.
- Added a documentBuildContainer for temporary storage during page setup, ensuring correct page ordering.

2024-11-27 - 1.0.102 - fix(contentinvoice)

Fixed currency display issues in invoice content rendering.

- Corrected the usage of currency property to consistently pull from letterData content for invoice items.

- Ensured currency is correctly displayed in the invoice PDF generation.

2024-11-27 - 1.0.101 - fix(shared)

Fixed the demo letter item details by replacing `currency` with `position` to correctly define the item position.

- Replaced incorrect 'currency' field with 'position' in `demoletter.ts` for item details.
- Ensured item positions are correctly assigned in `demoletter.ts`.

2024-11-27 - 1.0.100 - fix(core)

Fix issues with PDF service, invoice item positioning, and update package dependencies.

- Updated package dependencies to latest versions for improved stability and performance.
- Fixed invoice item positioning in demo data.
- Ensured PDF service handles rendering updates more efficiently.

2024-04-20 to 2023-10-16 - 1.0.99 to 1.0.90 - Core Fixes and Documentation Update

Routine updates and fixes with a focus on core improvements and documentation enhancements.

- Regular core updates to enhance stability and performance
- Updated project documentation for better clarity and user guidance