

Clauses

Clause

Represents a reusable segment of text or conditions that form part of a document. The textual data of a clause is stored as a template, which can be rendered using the Scriban syntax. The template data can include variables, conditions, and loops to generate dynamic content. The variables within a clause are represented by clause fields.

Snippet

Represents a reusable snippet of template or code that can be used within clauses. Used for fields that are common across multiple clauses or for defining reusable templates.

ClauseField

Represents a specific field within a clause, detailing the data requirements that part of the clause. A field can have a condition under which it is applicable.

ClauseGroup

Represents a group of clauses, allowing for hierarchical organization of clauses within the system. Used by the front-end application to organize clauses into a library.

ContractModels

ContractModel

Represents the model for a contract, defining the structure and components that make up a contract type. Is what is versioned. It would be equivalent of the categories + versions of the old system.

ContractModelGroup

Represents a group of contract models, allowing for organizational structuring and categorization.

ContractModelPartyAdditionalField

Represents additional fields associated with a party in a contract model, allowing for the customization and extension of party data requirements.

ContractModelParty

Represents a party within a contract model, defining the roles and parties involved in the contract. Is possible to define a template for each type of party and what types are allowed. The template is

used when the meta field `_template` is used in a clause template data. If the template is not set the default template, defined in `PartyType`, is used.

Contracts

ContractDocumentRender

Represents a rendered version of a document associated with a contract, which might include the final formatted document ready for review or signatures.

Contract

Represents a specific instance of a contract. Is created from a contract model, and contains all the information and values specific to an individual contract.

ContractArtifact

Represents an artifact related to a contract, such as a generated document or file, providing a way to store and reference these items within the system.

ContractQuestionValue

Represents the answer or value provided for a specific question within a contract, capturing user input or selections related to the contract's configuration.

ContractPartyValue

Represents the actual assignment of a customer's party to a party required by a contract model.

ContractClauseFieldValue

Represents the value assigned to a specific field within a clause for a given contract, capturing the data input or selection made during contract customization.

Customers

User

Represents a user in the system, typically an individual who interacts with the application, such as a customer or an administrator.

CustomerParty

Represents a party related to a customer, detailing information about an individual or organization's role or identity in the context of contracts.

Customer

Represents a customer of an accountant. Is the owner of contracts.

CustomerPartyFieldValue

Represents a specific value or attribute associated with a customer party, detailing information such as contact details, identifiers, or preferences.

Documents

DocumentClause

Represents a clause within a document, with their conditions to be rendered and order in the document. These conditions are based on the questionnaire answers and party assignments. The conditions can also be described as part of the old system's modules.

Document

Represents a document within a model. The document is a collection of clauses and is what orchestrates the clauses into a coherent document. It's the equivalent of the old system's Google Docs templates.

Fields

Field

Represents a field within the system, which can be a part of a party, question, or clause.

FieldOption

Represents an option within a field, specifically for fields that allow selecting from predefined values, such as dropdowns or radio buttons.

FieldType

Represents a type of field within the system, defining how a field behaves and is interacted with, such as text, number, date, etc.

FieldGroup

Represents a group of fields, allowing for the logical grouping and organization of fields within the system.

FieldTypeFieldTypeUnit

Represents the association between a field type and a field type unit, allowing for the specification of units for fields that require them.

FieldTypeUnit

Represents a unit of measurement or categorization for field types that require or support specifying a unit, such as length, weight, currency, etc.

Jurisdictions

Jurisdiction

Represents a legal jurisdiction under which contracts, clauses, and other legal documents can be governed, such as a country, state, or specific legal area.

PartyTypes

PartyTypeField

Represents the association between a party type and a field, defining the fields or information that are required or applicable for a specific type of party.

PartyType

Represents a type of party that can be involved in contracts, such as an individual or company, defining the roles and information requirements for parties in contracts.

Questions

Question

Represents a question that may be posed within the context of a contract or document. Used to gather information or make decisions that affect the document's content or clauses. Are mainly used in the conditional clauses.

Translations

TranslationStatus

Represents the status of a translation within the system, indicating whether a translation is pending, approved, or requires review, thus managing the workflow and quality control of translations.

Translation

Represents a translation of content within the system, facilitating multilingual support for documents, questions, and other textual content.

Language

Represents a language supported by the system, used for translations and localization of content, documents, and user interfaces.

Namespace Instaclause.Accountant.Api. Controllers

Classes

[AnswersController](#)

[ContractsController](#)

[PartiesController](#)