Frontend [Container: HTML, Javascript, CSS] **API REST** [Container: Node, Express, Mongoose] Base de datos [Container: MongoDB] información de las colecciones de datos parametrizadas Transportar solicitudes desde y hacia el backend Gestiona registros en **Backend Web** la base de datos [Container: Node, Express, Mongoose] a través de la API Lee y escribe Gestiona usuarios. Core configura colecciones dinámicas [Container: Javascript] y gestiona sus registros

CITDEV Express [Software System]

CollectionsController

FieldsConfigController

PanelAdminController

AutoCrudController

Expone las operaciones básicas sobre los egistros de una colección de datos configurada

UserController

Router

ApiAutoCrudController

[Component: Node Express]

Genera formularios dinámicamente a partir de la configuración de atributos de una colección

AuthController

AuthMiddleware

[Component: Node]

Valida que la sesión del usuario o el token se encuentre vigente para acceder a las funcionalidades del sistema

FormBuilder

[Component: Node Library]

Genera formularios dinámicamente a partir de la configuración de atributos de una colección

CoreUsersModel

[Component: Node Mongoose Model]

Permite la lectura y escritura en base de datos de la información de los usuarios

CoreCollectionsModel

[Component: Node Mongoose Model]

Permite la lectura y escritura en base de datos de la información de las colecciones configuradas por el usuario

CoreAuditLogModel

[Component: Node Mongoose Model]

Permite la lectura y escritura en base de datos de las acciones de escritura sobre los registros de la dase de datos

DynamicResources

[Component: Node]

Crea y elimina dinámicamente el modelo de una colección de datos configurada por el usuario

AuditLogController