### Levantamento de Requisitos

### Stakeholders:

- Orientadora do Projeto
- Project Owner (Membro da ItCenter)
- Equipa de Desenvolvimento
- Cliente (Empresa que necessita de um software de controlo de inventário)

# Levantamento de Requisitos Funcionais O visitante pode fazer o registo no site O visitante pode fazer login no site O visitante pode recuperar a password O utilizador pode ver os artigos do inventário O utilizador pode pesquisar os artigos do inventário (categoria, nome) O utilizador pode adicionar um artigo ao inventário O utilizador pode atualizar um artigo do inventário O utilizador pode apagar um artigo do inventário O utilizador pode ver as categorias do inventário O utilizador pode atualizar uma categoria ao inventário O utilizador pode atualizar uma categoria do inventário O utilizador pode atualizar uma categoria do inventário O utilizador pode apagar uma categoria do inventário O utilizador pode apagar uma categoria do inventário O utilizador pode ver um resumo do inventário por categorias O utilizador mudar o tema de cores da página

### Levantamento de Requisitos Não Funcionais

A base de dados deve ser desenvolvida em MySQL

O utilizador pode voltar ao topo da página

O utilizador pode fazer logout

A API de acesso à base de dados deve ser desenvolvida em Express (Node JS)

A página web deve ser desenvolvida em HTML/CSS/Bootstrap/JavaScript

A aplicação acede à base dados por intermédio da API para efetuar as operações solicitadas (CRUD)

O visitante para fazer registo deve introduzir o email, o nome, a password, e a password repetida

O visitante para fazer login deve introduzir o email e a password

O vistante para recuperar os dados de acesso deve introduzir o seu email

O utilizador após visualizar/pesquisar os artigos do inventário, pode atualizar ou apagar qualquer artigo

O utilizador após visualizar as categorias do inventário, pode atualizar ou apagar qualquer categoria

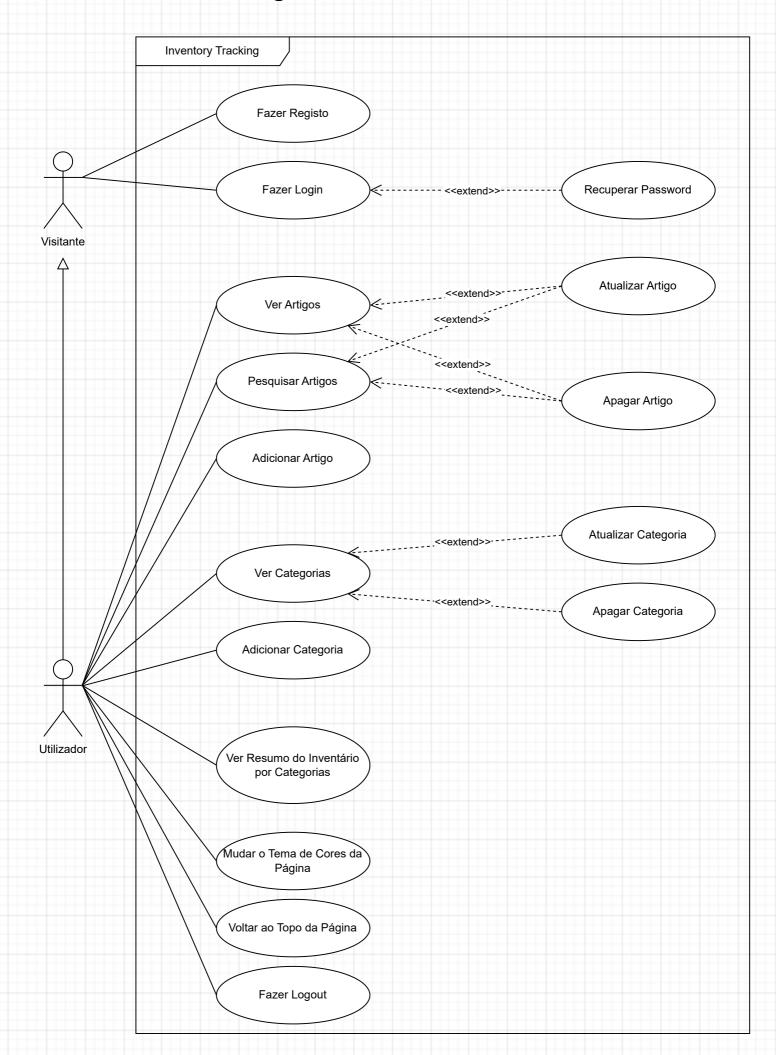
Desempenho/responsividade/fluidez do site

Design Responsivo adaptando-se a dispositivos móveis

UI/UX - A usabilidade deve ser intuitiva

Compatibilidade com browsers distintos

# Diagrama de Casos de Uso

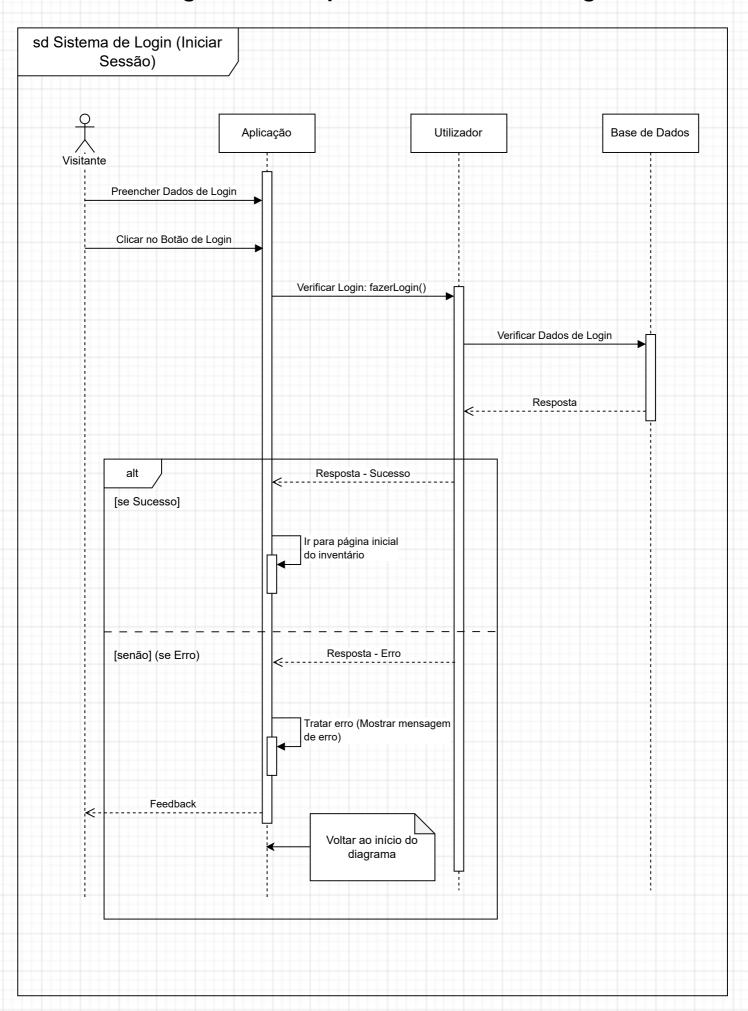


# Diagrama de Classes

Visitante	Utilizador
+ fazerRegisto(): String	- id_utilizador: int
+ fazerLogin(): void	- nome: String
+ recuperarPass(): void	- email: String
	- password: String
	+ get_id_utilizador(): String
	+ get_nome(): String
	+ get_email(): String
	+ set_nome(): String
	+ set_email(): String
	+ set_password(): String
	+ fazerLogout(): String

Artigo	0*	tem	01	Categoria
· id_artigo: int	0	tem	01	- id_categoria: int
- nome: VarChar(50)				- tipo: String
- quantidade: int				+ get_id_categoria(): int
- id_categoria: int				+ get_tipo(): String
- datainsercao: date				+ set_tipo(): String
+ get_id_artigo(): int				+ adicionarCategoria(): String
+ get_nome(): String				+ atualizarCategoria(): String
+ get_quantidade(): int				+ apagarCategoria(): String
+ get_id_categoria(): int				
+ get_datainsercao(): date				
+ set_nome(): String				
+ set_id_categoria(): int				
+ set_datainsercao(): date				
+ adicionarArtigo(): String				
+ set_quantidade(): int				
+ atualizarArtigo(id_artigo:int): String				
+ apagarArtigo(id_artigo:int): String				

# Diagrama de Sequência - Sistema de Login



# Diagrama de Sequência - Sistema de Adicionar Categoria

