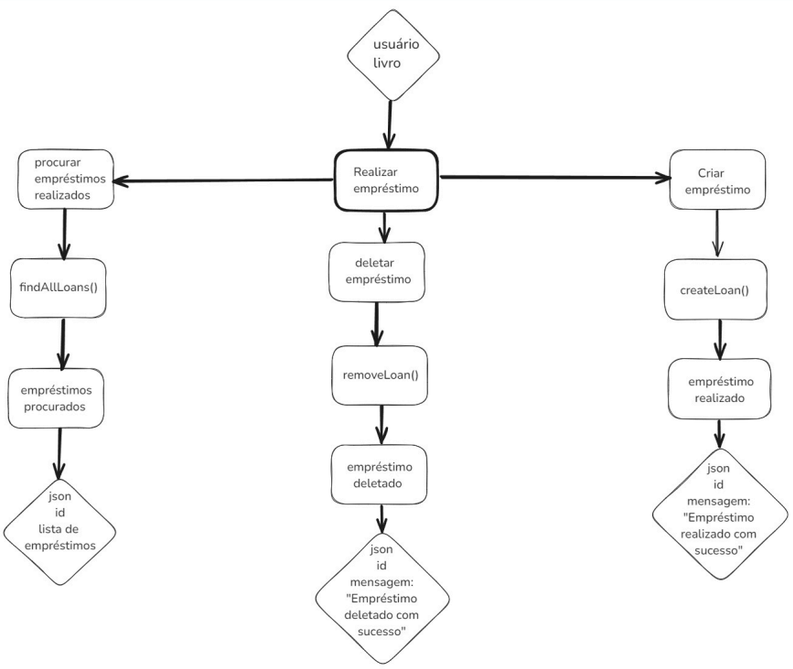
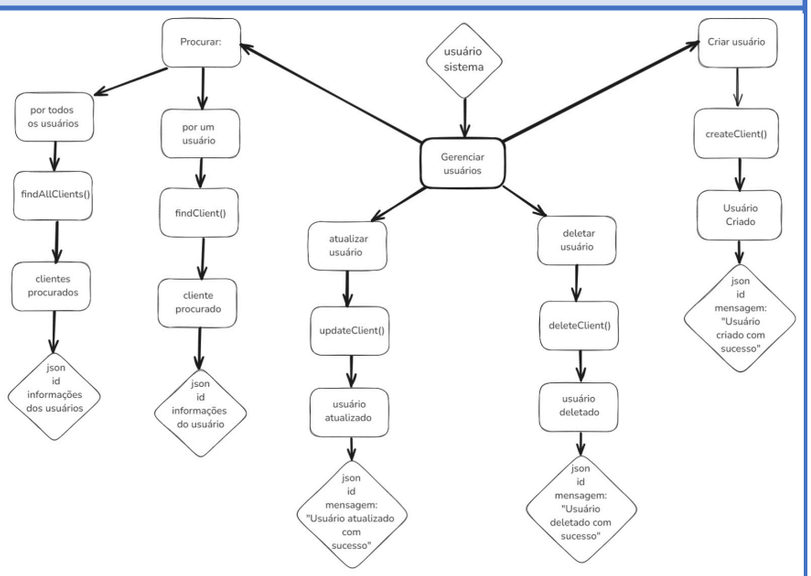
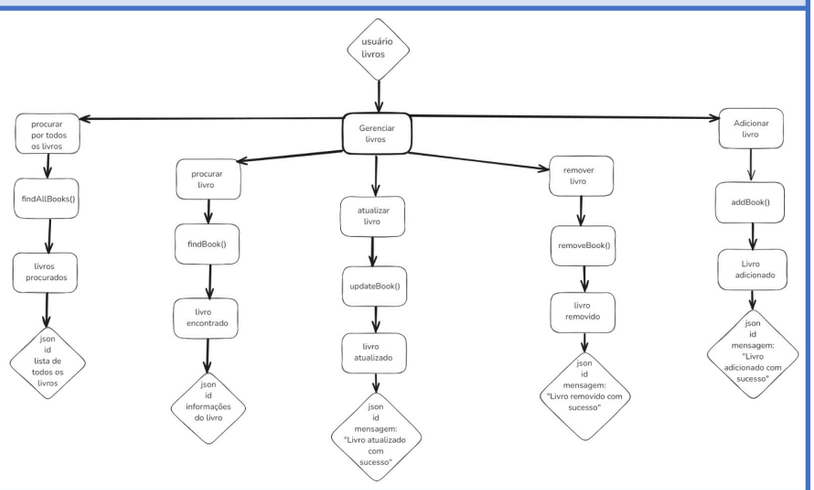
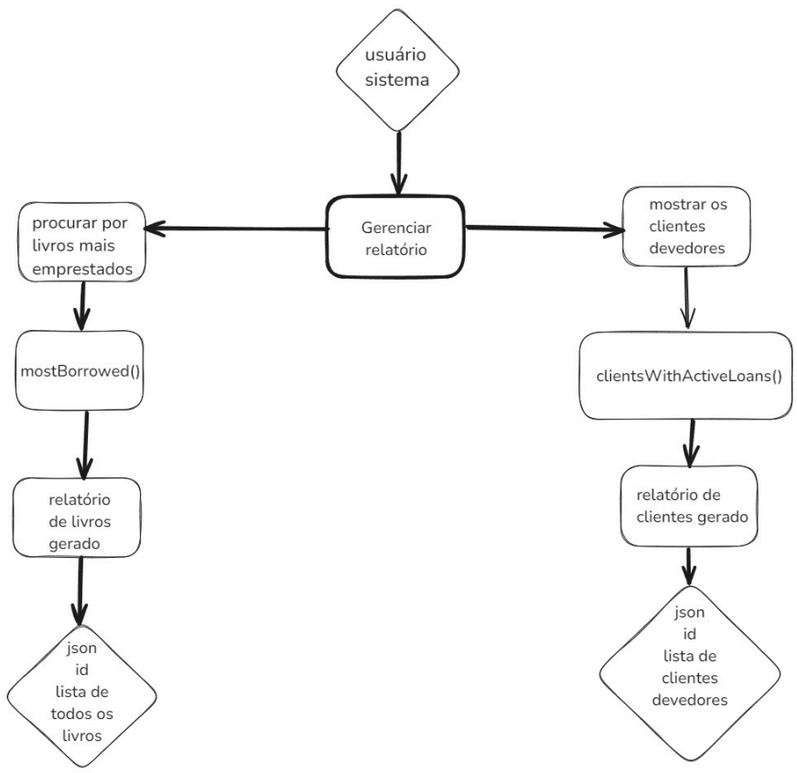
Casos de Uso

ID:	**UC001**
Título:	Registrar Empréstimo;
Descrição:	Permitir que o usuário registre empréstimos feitos pelos clientes;
Ator Primário:	Usuário;
Pré-condição:	● O cliente deve estar registrado no sistema; ● O livro deve estar no estado “disponível”;
Pós-condição:	● O empréstimo ficará registrado no sistema;
Diagrama:	
Fluxo Principal:	O usuário solicita o registro do empréstimo através do id do livro, chamando a createLoan() e finalizando a solicitação;
Fluxo Opcional:	
Fluxo Alternativo:	O id do usuário não existe, o id do livro não existe, erro de exceção e exceder o limite de empréstimos;
Tratamento de exceções:	Retorna a mensagem de erro “Ocorreu um erro ao registrar empréstimo” com código de status 500, e uma mensagem é mostrada no console, informando mais detalhes do erro.

ID:	**UC002**
Título:	Gerenciar Usuários;
Descrição:	Permitir que o administrador gerencie os clientes;
Ator Primário:	Administrador;
Pré-condição:	● O cliente não deve haver registro prévio;
Pós-condição:	● O usuário deverá ser registrado; ● O usuário deverá haver a opção de ter seus dados consultados;
Diagrama:	
Fluxo Principal:	O usuário é adicionado no banco de dados, chamando a createClient() e finalizando a solicitação
Fluxo Opcional:	
Fluxo Alternativo:	O id do usuário não existe;
Tratamento de exceções:	Retorna com uma mensagem de erro “Usuário não identificado” com código: console.log(error.message)

ID:	**UC003**
Título:	Gerenciar Livros;
Descrição:	Permitir que o usuário acompanhe e modifique o fluxo de processos dos livros;
Ator Primário:	Usuário;
Pré-condição:	• O livro novo não pode estar registrado no sistema; • O livro previamente existente não pode estar atualizado no sistema;
Pós-condição:	• O livro novo ficará registrado no sistema; • O livro novo deverá haver a opção de ser modificado; • O livro previamente existente deverá ficar registrado no sistema; • O livro previamente existente deverá haver a opção de ser modificado;
Diagrama:	
Fluxo Principal:	O livro é adicionado ao banco de dados, podendo ser deletado, atualizado e consultado; caso o livro já esteja incluso no banco de dados e precise ser atualizado, adicionado ou deletado, seu id é utilizado para tal evento;
Fluxo Opcional:	
Fluxo Alternativo:	O livro não existe, o livro não conseguiu ser adicionado, o livro não conseguiu ser deletado, o livro não conseguiu ser consultado e o livro está indisponível;

Tratamento de exceções:	Retorna a mensagem de erro “Erro ao buscar livro” com o status 500, caso haja fluxo alternativo;
	`catch(err){`
	`res.status(500).json({ error: 'Erro ao buscar livro.' });`
	`console.log(error.message);`
	`}`

ID:	**UC004**
Título:	Gerenciar Relatórios;
Descrição:	Permitir que o usuário registre relatórios feitos pelo banco de dados do sistema;
Ator Primário:	Usuário;
Pré-condição:	● O relatório deve apresentar as atividades recentes feitas no sistema;
Pós-condição:	● O relatório ficará registrado no sistema; ● O relatório deverá gerar uma lista dos empréstimos ocorridos; ● O relatório deverá gerar uma lista com os livros onde houveram maior número de empréstimos;
Diagrama:	
Fluxo Principal:	O usuário solicita o registro do empréstimo através do id do livro, chamando a registrarEmprestimo() e finalizando a solicitação;
Fluxo Opcional:	
Fluxo Alternativo:	● O id do usuário não existe; ● o id do livro não existe, ● erro de exceção ● exceder o limite de empréstimos;

Tratamento de exceções:	Manda o status 500 “Ocorreu um erro ao registrar relatório” com o código: