**Documento de Casos de Uso**

***iRural***

**Versão *1.0***

Douglas Canto

Flaviano Dias

Isabella Albuquerque

Jean Karlos

Rafael Albuquerque

**Histórico de Alterações**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 04/11/2013 | 0.01 | Criação do documento de casos de uso | Flaviano |
| 04/11/2013 | 0.02 | Edição | Isabella |
| 04/11/2013 | 0.03 | Adição de Casos de Uso | Jean |
| 04/11/2013 | 0.04 | Adição de Casos de Uso | Rafael |
| 10/01/2013 | 0.05 | Correção e Verificação/Revisão | Douglas |
|  |  |  |  |
|  |  |  |  |

**Conteúdo**

***1.*** ***Introdução***

**1.1** **Visão geral do documento**

**1.2** **Convenções, termos e abreviações**

*1.2.1* Identificação dos requisitos/casos de uso

*1.2.2* Prioridades dos requisitos/casos de uso

***2.*** ***Visão geral do sistema***

**2.1** **Abrangência e sistemas relacionados**

**2.2** **Descrição dos Usuários**

***3.*** ***Casos de Uso***

[UC01] Realizar Busca de Serviços

[UC02] Cadastrar Alunos

[UC03] Cadastrar Admins

[UC04] Verificar existência de usuário

[UC05] Cadastrar Cardápio RU

[UC06] Verificar existência do cardápio

[UC07] Visualizar Cardápio do RU

[UC08] Cadastrar Linha de Ônibus

[UC09] Verificar existência da linha

[UC10] Visualizar Linha de Ônibus

[UC11] Visualizar Agenda

[UC12] Visualizar Evento

[UC13] Incluir Evento

[UC14] Verificar existência de evento

[UC15] Visualizar Departamento

[UC16] Incluir Departamento

[UC17] Verificar existência de Departamento

[UC18] Visualizar Estação de Ônibus

[UC19] Incluir Estação de Ônibus

[UC20] Verificar existência de Estação

[UC21] Cadastrar Restaurante

[UC22] Verificar existência de Restaurante

***4.*** ***Requisitos não-funcionais***

**4.1** **Usabilidade**

**4.2** **Confiabilidade**

**4.3** **Desempenho**

**4.4** **Segurança**

**4.5** **Hardware e software**

**4.6** **Adequação a padrões**

***Apêndices***

***A*** ***Descrição da interface com o usuário***

# Introdução

Este documento especifica os casos de uso do sistema iRural, fornecendo aos desenvolvedores as informações necessárias para o projeto e implementação, assim como para a realização dos testes e homologação do sistema.

**Problema**

***Locomoção dentro do campus, acesso a informações sobre departamentos e serviços.***

**Pessoas Atingidas**

***Alunos, Funcionários e visitantes.***

**Impacto**

***Dificuldade em obter documentos, localizar departamentos, falta de informação sobre serviços ofertados.***

**Uma solução bem sucedida traria**

***Um sistema para facilitar e agilizar o acesso a informação e orientação aos usuários dentro da Universidade.***

## Visão geral do documento

Além desta seção introdutória, as seções seguintes estão organizadas como descrito abaixo.

* **Seção 2 – Descrição geral do sistema**: apresenta uma visão geral do sistema, caracterizando qual é o seu escopo e descrevendo seus usuários.
* **Seção 3 – Requisitos funcionais**: especifica as funcionalidades do sistema.
* **Seção 4 – Requisitos não-funcionais**: especifica todos os requisitos não funcionais do sistema, divididos em requisitos de usabilidade, confiabilidade, desempenho, segurança, distribuição, adequação a padrões e requisitos de hardware e software.
* **Seção 5 – Referências:** apresenta referências para outros documentos utilizados para a confecção deste documento.
* **Apêndices:**

**- Modelo de Funcionalidades**: apresenta uma ilustração do modelo de casos de uso do sistema.

## Convenções, termos e abreviações

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

### Identificação dos requisitos/casos de uso

Por convenção, a referência a requisitos é feita através do identificador do requisito, de acordo com a especificação a seguir:

[*identificador do requisito/caso de uso*]

Os requisitos e casos de uso devem ser identificados com um identificador único. A numeração inicia com o identificador [UC001] ou [NF001] e prossegue sendo incrementada à medida que forem surgindo novos requisitos.

### Prioridades dos requisitos/casos de uso

Para estabelecer a prioridade dos requisitos e casos de uso, nas seções 3 e 4, foram adotadas as denominações: “essencial”, “importante” e “desejável”.

* **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.
* **Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.
* **Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

# Visão geral do sistema

O sistema iRural tem como objetivo ajudar os membros da comunidade universitária (novatos e veteranos), no que diz respeito a disposição dos prédios da UFRPE e quais serviços são oferecidos. Também está incluída uma agenda para que o aluno possa saber das datas, como de matricula, modificação, trancamento e eventos tais como palestras, juntamente com o cardápio do RU e localização de outros pontos de alimentação. Para a informação sobre onde ficarão os prédios será utilizado um conjunto de mapas previamente mapeados em sistema desenvolvido em plataforma web. O sistema deverá ter uma área para login que irá habilitar edição de mapas, cardápio, entre outras funcionalidades internas, porém para visualização/utilização não será preciso efetua-lo.

## Abrangência e sistemas relacionados

O sistema iRural irá permitir aos seus usuários a execução e processamento das seguintes operações:

* Informar como e onde o aluno deverá ir para usufruir de algum serviço da comunidade acadêmica.
* Localização de pontos de ônibus.
* Localização de locais onde são vendidos alimentos.’
* Informação sobre o cardápio do RU;
* Informação sobre eventos;
* Informar também onde pode ser encontrado serviços gerais (como xérox, por exemplo);

## Descrição dos Usuários

O principal cliente do sistema é o aluno da UFRPE que poderá acessar através da internet identificado como [ ALUNO ] com o fim de interagir com o sistema, podendo ainda usar de forma passiva (só recebendo as informações) sem precisar se cadastrar.

Como alguns eventos e serviços são oferecidos pelos DA’s o membro do DA terá um acesso exclusivo ao sistema para adicionar as informações de serviços e eventos oferecidos pelo respectivo DA identificado como [ MEMBRO DA ]\*;

Existirá também um nível hierárquico e será criado também uma espécie de administrador do DA que será identificado como [ ADMIN DA ]\*;

Outros serviços serão oferecidos pelos setores da universidade existirão também o usuário responsável por cadastrar esses serviços identificado como [MEMBRO UFRPE ];

Existirá também um administrador para cada setor que será o administrador ele poderá adicionar editar e excluir eventos e/ou serviços do seu respectivo setor ele será identificado como [ ADMIN UFRPE];

Existirá também um administrador geral que será identificado como [ADMIN IRURAL]

E professores? E funcionários? Teriam alguma categoria especial ou seriam enquadrados em alguma das acima?

# Casos de Uso

Nesta seção detalharemos todos os cenários de utilização do sistema.

**[UC01] *Realizar Busca de Serviços***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o usuário cliente faça buscas por serviços disponíveis nos prédios e locais situados no campus da UFRPE.

**Pré-condição:** o cliente deve estar conectado à internet e a geolocalização ativa.

**Pós-condição**: resultado da busca com rota em mapa e descrição do serviço e horário de funcionamento.

**Fluxo de eventos principal**

1. O usuário navega para a tela de seleção de serviços.

2. O sistema mostra a tela de Serviços com um combobox com os nomes dos serviços:

Exemplo de serviços: Xerox, Caixas Eletrônicos

3. O usuário seleciona uma opção.

4. O sistema exibe um mapa destacando as localizações do serviço.

**Fluxos secundários**

1. Em qualquer momento o usuário pode voltar à tela inicial.

**[UC02] *Cadastrar Aluno***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o Aluno cadastre-se para utilização dos serviços.

**Pré-condição:** O Aluno deverá fornecer um CPF/Matrícula válido.

**Pós-condição**: Os dados do novo usuário validados e o usuário inserido no sistema.

**Fluxo de eventos principal**

1. O administrador informa os dados do novo usuário:

* CPF;
* Nome;
* RG;
* Tipo (DA, iRural ou UFRPE)
* Função
* Endereço;
* Logradouro
* Número
* Bairro
* Cidade
* Estado
* Complemento
* CEP

2. O sistema verifica as informações digitadas.

3. O cliente é inserido no sistema.

**[UC03] *Cadastrar Admins***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o administrador geral cadastre administradores específicos.

**Pré-condição:** o administrador geral deve estar conectado e autenticado no sistema (ter efetuado o login).

**Pós-condição**: os dados do novo usuário validados e o usuário inserido no sistema.

**Fluxo de eventos principal**

1. O administrador informa os dados do novo usuário:

* CPF;
* Nome;
* RG;
* Tipo (DA, iRural ou UFRPE)
* Função
* Endereço;
* Logradouro
* Número
* Bairro
* Cidade
* Estado
* Complemento
* CEP

2. O sistema verifica as informações digitadas.

3. O sistema inicia o UC03.

4. O cliente é inserido no sistema.

**Fluxos secundários**

1. No passo 2, se o CPF não for um CPF válido, o sistema informa que o CPF é inválido e retorna ao passo 1 do fluxo principal de eventos.

2. No passo 3, se o usuário já existe, o sistema exibe a mensagem “Usuário já cadastrado” e retorna ao passo 1 do fluxo principal de eventos.

3. Em qualquer momento o usuário pode cancelar a operação e voltar à tela inicial.

**[UC04] *Verificar existência de usuário***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve verificar se um determinado usuário existe no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1.Com o CPF é feito um acesso ao banco de dados que busca a existência do dado informado.

2.Se o resultado da pesquisa for nulo o cadastro é efetuado.

**Fluxos secundários**

1.Caso a pesquisa em banco retorne algum resultado, o cadastro é impedido.

**[UC05] *Cadastrar Cardápio RU***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o administrador responsável cadastre o Cardápio do RU.

**Pré-condição:** o administrador deve estar conectado e autenticado no sistema (ter efetuado o login).

**Pós-condição**: os dados do novo cardápio devem ser validados e inseridos no sistema.

**Fluxo de eventos principal**

1. O administrador informa os dados do novo cardápio:

* Dia em que será servido;
* Qual refeição se trata (Almoço ou Jantar);
* Itens do cardápio;

- Prato Principal;

- Opção;

- Guarnição;

- Feijão;

- Arroz;

- Salada crua;

- Salada cozida;

- Sobremesa;

- Suco1;

- Suco2.

2. O sistema inicia o UC05.

3. O cardápio é inserido no sistema.

**Fluxos secundários**

1. No passo 2 caso o UC05 retorne que já existe o cardápio a operação é cancelada.

2. Em qualquer momento o usuário pode cancelar a operação.

**[UC06] *Verificar existência do cardápio***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve verificar se um determinado cardápio existe no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1.Com o dia informado é feito um acesso ao banco de dados que busca a existência de algum cardápio informado para a refeição selecionada;

2.Se o resultado da pesquisa for nulo o cadastro é efetuado.

**Fluxos secundários**

1.Caso a pesquisa em banco retorne algum resultado, o cadastro é impedido.

**[UC07] *Visualizar Cardápio do RU***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir ao usuário cliente visualizar o cardápio do Restaurante Universitário (RU), para aquele dia.

**Pré-condição:** o cliente deve estar conectado à internet.

**Pós-condição**: Tabela com opções de cardápio do dia no Restaurante Universitário.

**Fluxo de eventos principal**

1. O cliente seleciona a opção alimentação na tela Home do aplicativo:

2. O sistema mostra a tela de Alimentação com as seguintes opções:

- Almoço;

- Jantar.

3. O usuário seleciona uma das opções.

4. O sistema retorna ao usuário uma tabela que deverá ser aberta uma nova guia do navegador com o cardápio do dia para a refeição selecionada previamente, contendo os campos: Prato Principal, Opção, Guarnição, Feijão, Arroz, Salada crua, Salada cozida, Sobremesa, Suco1 e Suco2.

**Fluxo Excepcional do Evento**

Caso o cliente selecione a opção errada no item 3, poderá selecionar o botão “ voltar” e voltará apenas para a tela de “Alimentação”, obtendo a chance de escolher a opção correta.

**[UC08] *Cadastrar Linha de Ônibus***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o administrador geral cadastre as linhas dos ônibus que circulam nas proximidades da universidade.

**Pré-condição:** o administrador deve estar conectado e autenticado no sistema (ter efetuado o login).

**Pós-condição**: os dados da nova linha devem ser validados e inserida no sistema.

**Fluxo de eventos principal**

1. O administrador informa os dados da nova linha:

* Nome;
* Código;
* Preço.

2. O sistema inicia o UC08.

3. A linha é inserida no sistema.

**Fluxos secundários**

1. No passo 2, se a linha já existe, o sistema exibe a mensagem “Linha já cadastrada” e retorna ao passo 1 do fluxo principal de eventos.

2. Em qualquer momento o usuário pode cancelar a operação.

**[UC09] *Verificar existência da linha***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve verificar se um determinado usuário existe no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1.Com o código da linha é feito um acesso ao banco de dados que busca a existência do dado informado.

2.Se o resultado da pesquisa for nulo o cadastro é efetuado.

**Fluxos secundários**

1.Caso a pesquisa em banco retorne algum resultado, o cadastro é impedido.

**[UC10] *Visualizar Linha de Ônibus***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir ao usuário cliente visualizar as linhas de ônibus que passam pela área da universidade.

**Pré-condição:** o cliente deve estar conectado à internet.

**Pós-condição**: Tabela com as linhas cadastradas.

**Fluxo de eventos principal**

1. O cliente clica o botão Ônibus na tela Home do sistema:

2. O sistema mostra a tela de com a listagem das linhas cadastradas.

3. O usuário seleciona uma das opções.

4. O sistema retorna ao usuário as informações sobre a linha:

- Número

- Nome/Rota.

**Fluxo Excepcional do Evento**

Caso o cliente selecione a opção errada no item 3, poderá selecionar o botão “ voltar” e voltará apenas para a tela de “Ônibus”, obtendo a chance de escolher a opção correta.

**[UC11] *Visualizar Agenda***

**Prioridade**: Importante.

**Descrição do requisito funcional:** O sistema deve permitir ao usuário cliente visualizar eventos cadastrados e dispô-los ordenadamente em um gridview de acordo com a data, o usuário pode selecionar um evento para visualizar [UC12].

**Pré-condição:** o cliente deve estar conectado à internet e Logado.

**Pós-condição**: Grid com as linhas cadastradas.

**Fluxo de eventos principal**

1. O cliente clica no botão Agenda na tela Home do aplicativo

2. O sistema abre a tela com um checkbox as seguintes opções:

- Eventos da Semana.

- Eventos do Mês.

3. O sistema filtra e mostra na tela um grid com a listagem dos eventos cadastrados.

**Fluxo Secundário**

1. Em qualquer momento o usuário pode cancelar a operação.

**[UC12] *Visualizar Evento***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir ao usuário cliente visualizar os eventos agendados para acontecer na UFRPE.

**Pré-condição:** o cliente deve estar conectado à internet, Logado e na tela de Agenda [UC11].

**Pós-condição**: Tela de visualização de eventos.

**Fluxo de eventos principal**

1. O cliente clica (como um link) no evento selecionado, na tela de Agenda.

4. O sistema retorna ao usuário os dados do evento previamente cadastrados: Data, Nome, Local, Horário e Preço.

**Fluxo Excepcional do Evento**

1. Caso o cliente selecione a opção errada no item 1, poderá selecionar o botão “ voltar” e voltará apenas para a tela de Agenda [UC12], obtendo a chance de escolher a opção correta.

**[UC12] *Incluir Evento***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o administrador geral cadastre eventos.

**Pré-condição:** o administrador deve estar conectado e autenticado no sistema (ter efetuado o login).

**Pós-condição**: os dados do novo evento validados e o evento inserido no sistema.

**Fluxo de eventos principal**

1. O administrador informa os dados do novo evento:

* Data;
* Nome;
* Local;
* Horário:
* Preço;

2. O sistema verifica as dados digitados.

3. O sistema inicia o UC13.

4. O evento é inserido no sistema.

**Fluxos secundários**

1. No passo 3, se o evento já existe, o sistema exibe a mensagem “Evento já cadastrado” e retorna ao passo 1 do fluxo principal de eventos.

2. Em qualquer momento o usuário pode cancelar a operação.

**[UC13] *Verificar existência de evento***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve verificar se um determinado evento existe no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1.Com a Data e ID é feito um acesso ao banco de dados que busca a existência do dado informado.

2.Se o resultado da pesquisa for nulo o cadastro é efetuado.

**Fluxos secundários**

1. Caso a pesquisa em banco retorne algum resultado, o cadastro é impedido.

**[UC14] *Visualizar Departamento***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir ao usuário cliente visualizar os departamentos existentes na UFRPE.

**Pré-condição:** o cliente deve estar conectado à internet.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1. O cliente clica o botão Departamentos na tela Home do aplicativo:

2. O sistema mostra a tela de Departamentos com as seguintes opções:

- Por prédio;

- Por curso.

3. O usuário seleciona uma das opções.

4. O sistema retorna ao usuário a localização em um mapa imediatamente abaixo, do departamento de acordo com a opção selecionada previamente.

**Fluxo Excepcional do Evento**

1. Caso o cliente selecione a opção errada no item 3, poderá selecionar o botão “ voltar” e voltará apenas para a tela de “Departamentos”, obtendo a chance de escolher a opção correta.

**[UC15] *Incluir Departamento***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o administrador geral cadastrar Departamento.

**Pré-condição:** o administrador deve estar conectado e autenticado no sistema (ter efetuado o login).

**Pós-condição**: os dados do novo departamento validados e o departamento inserido no sistema.

**Fluxo de eventos principal**

1. O administrador informa os dados do novo departamento:

* Nome;
* Localização (Latitude e Longitude);
* Horário de funcionamento;

2. O sistema verifica as informações digitadas.

3. O sistema inicia o UC015.

4. O departamento é inserido no sistema.

**Fluxos secundários**

1. No passo 3, se o evento já existe, o sistema exibe a mensagem “Departamento já cadastrado” e retorna ao passo 1 do fluxo principal de eventos.

2. Em qualquer momento o usuário pode cancelar a operação.

**[UC16] *Verificar existência de Departamento***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve verificar se um determinado Departamento existe no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1.Com a Localização e ID é feito um acesso ao banco de dados que busca a existência do dado informado.

2.Se o resultado da pesquisa for nulo o cadastro é efetuado.

**Fluxos secundários**

1. Caso a pesquisa em banco retorne algum resultado o cadastro é impedido.

**[UC17] *Visualizar Estação de Ônibus***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir ao usuário cliente visualizar as estações de ônibus existentes no campus da UFRPE.

**Pré-condição:** o cliente deve estar conectado à internet e Logado no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1. O cliente clica o botão Estações Ônibus na tela Home do aplicativo:

2. O sistema retorna ao usuário a localização das estações próximas em pontos fixos em um mapa que será exibido na tela.

3. O usuário seleciona a estação desejada e o sistema traça um caminho da sua atual localização até a estação.

**Fluxo Excepcional do Evento**

1. Caso o cliente selecione a estação errada no item 3 (mudou de ideia ou simplesmente confundiu a parada), poderá selecionar o botão “voltar” e voltará apenas para a tela de “Estação de Ônibus”, obtendo a chance de escolher a estação correta.

**[UC18] *Incluir Estação de Ônibus***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir que o administrador geral cadastre estações de ônibus.

**Pré-condição:** o administrador deve estar conectado e autenticado no sistema (ter efetuado o login).

**Pós-condição**: os dados da nova estação de ônibus validados e a estação inserida no sistema.

**Fluxo de eventos principal**

1. O administrador informa os dados da nova estação de ônibus:

● Numero;

● Localização (Latitude e Longitude);

2. O sistema verifica as informações digitadas.

3. O sistema inicia o UC18.

4. A estação é inserida no sistema.

**Fluxos secundários**

1. No passo 3, se a estação de ônibus já existe, o sistema exibe a mensagem “Estação já cadastrada” e retorna ao passo 1 do fluxo principal de eventos.

2. Em qualquer momento o usuário pode cancelar a operação.

**[UC19] *Verificar existência de Estação***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve verificar se uma determinada estação existe no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1.Com a Localização e ID é feito um acesso ao banco de dados que busca a existência do dado informado.

2.Se o resultado da pesquisa for nulo o cadastro é efetuado.

**Fluxos secundários**

1. Caso a pesquisa em banco retorne algum resultado o cadastro é impedido.

**[UC20] *Cadastrar Restaurante***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve permitir ao Administrador cadastrar estabelecimentos alimentícios caracterizados como restaurantes, com a finalidade de indicar no mapa sua devida localização.

**Pré-condição:** O Administrador deve estar logado.

**Pós-condição**: Estabelecimento cadastrado e exibido no mapa.

**Fluxo de eventos principal**

1. O cliente clica o botão restaurantes na tela Home do aplicativo:

2. O sistema mostra a tela de Restaurantes com os campos:

- Localização (Latitude e Longitude);

- Nome;

- Foto (Opcional).

3. O sistema inicia o UC20.

4. O restaurante é inserido no sistema.

**Fluxos secundários**

1. Em qualquer momento o usuário pode cancelar a operação.

**Fluxo Excepcional do Evento**

1. Caso haja resultado positivo na consulta ao banco de dados, o sistema deverá exibir um popup descrevendo o estabelecimento que já está cadastrado.

**[UC21] *Verificar existência de Restaurante***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve verificar se um determinado restaurante existe no sistema.

**Pós-condição**: o usuário é retornado ao caso de uso principal.

**Fluxo de eventos principal**

1.Com as coordenadas geográficas é feito um acesso ao banco de dados que busca a existência do dado informado.

2.Se o resultado da pesquisa for nulo o cadastro é efetuado.

**Fluxos secundários**

1.Caso a pesquisa em banco retorne algum resultado o cadastro é impedido.

**[UC22] *Calculo de rotas***

**Prioridade**: Essencial

**Descrição do requisito funcional:** O sistema deve calcular as rotas e traça-las no mapa utilizando a longitude e a latitude do ponto em questão e o GPS do dispositivo.

**Pré-condição:** o GPS do dispositivo, assim como a internet, devem estar habilitados.

**Pós-condição**: Rota traçada no mapa e atualizada constantemente até chegar no ponto de acordo com a locomoção do usuário.

**Fluxo de eventos principal**

1. Usuário acessa o sistema;

2. Escolhe a opção “Localização” na página HOME;

3. Escolhe o ponto com o qual deseja se locomover;

4. O servidor solicita ao dispositivo sua geolocalização;

5. Se a requisição trouxer dados, o servidor retorna a rota atualizada no mapa.

**Fluxos secundários**

1.Caso ocorra falha no retorno, pede para tentar novamente através de um lightbox solicitando realizar o procedimento novamente.

2. O usuário a qualquer momento pode sair da rota, caso isso ocorra, o servidor deverá atualizá-la até o ponto de destino.

3. A qualquer momento o usuário poderá apertar no botão “HOME” e o servidor cancela a rota traçada.

# Requisitos não-funcionais

## Usabilidade

1. Durante a fase de beta-testes, a interface deve ser considerada amigável por 80% dos usuários envolvidos nos testes.

2. O Sistema deve seguir um padrão de cores claras.

3. Todos os elementos (links, imagens) devem ter a propriedade alt para facilidade da leitura da página para deficientes.

## Confiabilidade

O sistema deve estar disponível 24 horas por dia, 7 dias por semana, com não mais que 2% do tempo com o sistema fora do ar.

## Desempenho

1. O servidor Web do sistema deve suportar o maior número possível conexões simultâneas.

2. O sistema não deve demorar mais de 100 ms para ser carregado.

3. Todos os ‘CSS’ e ‘Javascript’ devem ser comprimidos.

4. Usar o mínimo de imagens possível para o design.

## Segurança

Para ter acesso ao sistema na parte referente à adição e edição de informações os usuários devem estar logados.

O sistema ira permitir recuperação de senha caso seja necessário por parte do usuário.

O sistema ira permitir redefinição de senha caso seja necessário por parte do usuário.

## Hardware e software

1. O sistema deve integrar-se com os mapas OpenStreetMaps.

2. A interface do usuário deve ser feita de modo responsivo para que se adapte aos vários tipos de dispositivos.

## Adequação a padrões

1. As tecnologias utilizadas para a criação do sistema são: HTML5, CSS3, Javascript, PHP5.

2. Os frameworks utilizados para o auxílio ao desenvolvimento são ‘CakePHP’ para PHP , ‘Twiiter Bootstrap’ para CSS, ‘Jquery’ para Javascript.

3. O SGBD a ser utilizado é o MySQL.

4. O conjunto de mapas será o OpenStreetMaps

**Apêndices**

1. **Descrição da interface com o usuário**
   * + 1. No modelo ER tem a entidade Setor e aqui no doc de UC, não é mencionada.
       2. Item cardápio
       3. Tipo cardápio