**ESPECIFICAÇÃO E PROJETO DE DESENVOLVIMENTO DE SOFTWARE**

1. **DADOS DE IDENTIFICAÇÃO** 
   1. **Título do Projeto**

Sistema para Controle de tarefas internas de um setor imobiliário.

* 1. **Responsável pelo Projeto**

*Igor Camargo Moiano*

*Igor.moiano@hotmail.com*

1. **OBJETIVOS**

O sistema tem como principal objetivo ser capaz de manter o controle de fluxo de atividades individual de usuários, também disponibilizando acesso restrito a funcionalidades para determinados níveis de usuário.

1. **ESPECIFICAÇÃO INICIAL DOS REQUISITOS**
   1. **Método de Levantamento de Requisitos**

Os requisitos serão coletados através da perspectiva de um(a) gerente de uma imobiliária, pois este cargo tende a definir tarefas e realizar tarefas, sendo assim será realizados perguntas quanto a tipos de atividades o(a) entrevistado(a) tende a realizar e a definir para que seus subordinados tenham de realizar.

* 1. **Lista de funcionalidades**

*Obter a lista de funcionalidades principais que irão compor o sistema em desenvolvimento. Na tabela a seguir existem dois exemplos de funcionalidades para um sistema de apoio informatizado ao controle de vendas, de estoque, de compra e de fornecedores de uma mercearia.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Número de ordem** | **Funcionalidade** | **Necessidades** | **Benefícios** |
| 1 | Gestão de Tarefas | Controle de tarefas, saber quais tarefas devem ser realizadas. | Agilidade na definição de tarefas, diminuindo a necessidade da pessoa física definir as tarefas. |
| 2 | Gestão de pessoas | Fornece informações de cada usuário do sistema. | Identifica o que cada usuário esta fazendo e qual atividade deve realizar, tornando o processo de resolução das atividades de forma organizada. |
| 3 | Gestão de acesso ao sistema | Fornecer tipos de informações baseadas no tipo de usuário. | Cada usuário terá disponível apenas funcionalidades baseadas para seu nível de usuário. |

1. **ESTUDO DE VIABILIDADE**

O sistema pode ser implementado utilizando a linguagem de programação Php, pois grande parte dos sistemas imobiliários como Jetimob em Santa Maria tende a utilizar essa linguagem para desenvolvimento Web, pois como se trata de um sistema independente, se o mesmo apresentar um bom resultado final pode ser adicionado como um módulo em um sistema já existente.

Devido o contato próximo a um(a) gerente a coleta de requisitos será o mais próximo possível de situações reais tornando a ferramenta algo que possa ser utilizado no contexto de uma determinada imobiliária.

1. **PLANO DE ATIVIDADES E CRONOGRAMA**

*Descrever todas as atividades necessárias para desenvolver o projeto e elaborar um cronograma de execução das atividades.*

1. **ESPECIFICAÇÃO DOS REQUISITOS**
   1. **Requisitos Funcionais**

*Especificar todos os requisitos funcionais do projeto a ser desenvolvido e os requisitos não funcionais diretamente ligados a cada requisito funcional*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **F1 <Manter Funcionário>** | | | **Oculto ( )** | |
| O sistema devera fazer a inclusão, exclusão, alteração e listar os funcionários cadastrados. Mantendo dados como nome, telefone, endereço, empregado, verificar o que mais precisa | | | | |
| **Requisitos não-funcionais associados** | | | | |
| **Nome** | **Restrição** | **Categoria** | | **Desejável** |
| NF1.1 Controle de acesso | A função só poderá ser acessada por um gerente ou um administrador do sistema. | Segurança | | ( ) |
| NF1.2 Exclusão de funcionários. | Esta funcionalidade não remove totalmente o funcionário do banco, pois é necessário manter o registro sendo assim penas definido a variável empregado para false. | Usabilidade | | ( ) |
| NF1.3 Inserção de funcionário | Durante a inserção de um novo funcionário o mesmo é definido com a variável empregado true | usabilidade | | ( ) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **F2 <Manter Tarefa>** | | | **Oculto ( )** | |
| O sistema devera fazer a inclusão, exclusão, alteração e listar as tarefas cadastradas. Mantendo dados como descrição, funcionário(s) responsável(is), data limite de conclusão, local onde deve ser realizado, verificar o que mais precisa | | | | |
| **Requisitos não-funcionais associados** | | | | |
| **Nome** | **Restrição** | **Categoria** | | **Desejável** |
| NF1.1 Controle de acesso | A função só poderá ser acessada por um gerente ou um administrador do sistema. | Segurança | | ( ) |
| NF1.2 Exclusão de tarefas. | Esta funcionalidade não remove totalmente a tarefa do banco, pois é necessário manter o registro. | Usabilidade | | ( ) |

*ATENÇÃO: repetir esta tabela para cada requisito funcional existente.*

* 1. **Requisitos Suplementares**

*Corresponde aos requisitos não funcionais pertinentes ao sistema como um todo. Os requisitos suplementares são todo tipo de restrição tecnológica ou lógica que se aplica ao sistema como um todo e não apenas a funções individuais.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Restrição** | **Categoria** | **Desejável** |
| S1 <Nome Requisito não-funcional geral> |  |  | ( ) |
| S2 <Nome Requisito não-funcional geral> |  |  | ( ) |
| S3 <Nome Requisito não-funcional geral> |  |  | ( ) |

* 1. **Casos de uso relacionados aos requisitos funcionais**

*A tabela a seguir mostra a relação entre os casos de uso do sistema e os requisitos que os especificam.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome caso de uso** | **Atores** | **Descrição** | **Referências Cruzadas*\**** |
| *<Caso de uso 1>* |  |  |  |
|  |  |  |  |

*\*Cada caso de uso será associado a um conjunto de requisitos funcionais do sistema. Indicar o identificador de cada requisito funcional associado ao caso de uso.\*Cada caso de uso será associado a um conjunto de requisitos funcionais do sistema. Indicar o identificador de cada requisito funcional associado ao caso de uso.*

1. **METODOLOGIA DE DESENVOLVIMENTO**

*Descrever a metodologia que será utilizada para o desenvolvimento do sistema. [para acompanhar as atividades, para implementar, para troca de informações com o cliente, etc.]*

1. **MODELAGEM DO SISTEMA** 
   1. **Diagrama de casos de uso**

*Elaborar o diagrama de casos de uso (notação UML) que represente o sistema.*

* 1. **Descrição dos casos de uso**

*Descrever textualmente os principais casos de uso do sistema. Sistemas pequenos devem apresentar todos os casos de uso, já sistemas maiores podem apresentar apenas os casos de uso das principais funcionalidades, de forma a não deixar o texto tão repetitivo.*

* + 1. **Caso de uso <nome 1>**

|  |  |
| --- | --- |
| Descrição: | |
| Ator(es) envolvido(s): | |
| Pré-condição: | |
| **Fluxo Normal:** | |
| Ator | Sistema |
|  |  |
|  |  |
|  |  |
|  |  |
| **Fluxo Alternativo:** | |
|  | |
| **Fluxo de exceção:** | |
|  | |

*ATENÇÃO: repetir esta tabela para cada caso de uso existente.*

* 1. **Modelo ER**

*Além do diagrama E-R, fazer* uma descrição das entidades no formato de uma tabela (no máximo um parágrafo para cada entidade).

* 1. **Diagrama de Classes**

*Apresentar o diagrama de classes do projeto do sistema (conversar com o orientador).*

* 1. **Interface do Sistema (Prototipação de baixa, média ou alta fidelidade)**

*Apresentar um esboço das interfaces do sistema (rascunho do design ou projeto das páginas HTML que irão compor a aplicação).*