



**UNIVERSIDADE PAULISTA – UNIP**

**CURSO: 4º ANO - CIÊNCIA DA COMPUTAÇÃO - NOTURNO**

**APS - ATIVIDADES PRÁTICAS SUPERVISIONADAS**

**“Aplicação da Engenharia de Requisitos em um projeto software”**

**(7º semestre)**

**ALUNOS**

**JONAS ROMBALDO DENARDI – RA: C455881**

**MARCELO HENRIQUE RAMOS – RA: C612CG0**

**RIBEIRÃO PRETO**

## **1 - Sumário**

Objetivo do trabalho .....	03
Introdução .....	04
Conceitos gerais em um projeto software .....	05
Descrição das atividades .....	07
Conclusão .....	13
Bibliografia .....	14
Anexos (Templates do RUP) .....	15
Ficha de Atividades Práticas Supervisionadas .....	48

## **2 - Objetivo do trabalho**

O trabalho apresentado aqui visa elaborar um sistema para a ONG “Jovens Ambientalistas”, a qual recolhe, educa e oferece formação profissionalizante para jovens sem lar que depois de receberem cursos gratuitos por professores que são ex-alunos, prestam serviços remunerados, fabricando brinquedos “ambientalmente corretos” que são vendidos para o Brasil e o exterior. Ela deseja uma solução informatizada no qual será desenvolvido um sistema para efetuar um melhor controle sobre as informações da referida instituição, esse trabalho está constituído de uma pesquisa sobre os conceitos de requisitos, engenharia de requisitos e seus processos, elicitação, análise e negociação, especificação, modelagem, validação e gestão, posteriormente foi desenvolvido uma solução no qual foram explorados todos os processos da engenharia de requisitos, as atividades foram detalhadas e as nossas conclusões foram colocadas nesse trabalho.

O objetivo principal desse trabalho é nos familiarizarmos com um ambiente de desenvolvimento de software, seus processos, suas dificuldades e nos aperfeiçoarmos e capacitarmos com as atividades feitas durante o desenvolvimento do trabalho as situações recorrentes nesse meio.

Sistemas são importantes para controlar desde um simples celular até os processos mais importantes de um governo, por exemplo, então explorar essa área é fundamental para a formação de um aluno ciente das tecnologias e metodologias empregadas em um sistema, desenvolvimento de capacidades relacionadas à análise e solução de problemas, construção e prevenção dentro de um sistema, aprimorando assim suas qualidades e competências para o mercado de trabalho atual que exige mais qualidade e experiência dos candidatos.

Então esse trabalho serve para demonstrar como o conhecimento aliado à prática pode ajudar a moldar um aluno capaz de produzir um sistema de acordo com as exigências de um cliente seguindo os processos necessários para tal finalidade.

### 3 - Introdução

Engenharia é a ciência que aplica vários conhecimentos, matemáticos, técnicos e científicos para a criação, construção, análise, desenvolvimento e manutenção de máquinas, construções, processos, sistemas e outros. Ou seja, engenharia são conceitos utilizados em várias áreas, para criar, analisar, desenvolver, dar manutenção, á vários projetos, e infinitas coisas, seguindo padrões e normas par que tudo saia da melhor maneira possível.

Software é uma sequência de instruções executadas por um computador para realizar diversas tarefas. Ele é a parte lógica do computador e leva instruções para o hardware. O software é constituído por todos os programas que existem para um requerido sistema.

Engenharia de software é uma área da computação voltada á especificação, desenvolvimento e manutenção de sistemas de software, com aplicação de tecnologias e práticas de gerência de projetos e outras disciplinas, visando organização, produtividade e qualidade. A Engenharia de software abrange ferramentas de apoio para atividades, métodos para orientar a realização das atividades, processos para definir as atividades e os produtos, a qualidade de processo e de produtos de software. "A Engenharia de Software é o estabelecimento e uso de sólidos princípios de engenharia a fim de obter um software que seja confiável e que funciona de forma econômica e eficiente em máquinas reais" - Fritz Bauer.

Este conceito foi criado em 1969 por Fritz Bauer em uma conferência, no momento em que existia a crise do software que precisava de uma solução para a demanda cada vez mais crescente por software. Para ele a engenharia de um software utiliza sólidos princípios com o objetivo de aumentas a qualidade dos softwares produzidos, diminuindo custos e riscos dos processos na hora de elaborar os mesmos.

O projeto em si abordara o conceito de implementação de um software voltado a uma Usina sucroalcooleira para que a mesma consiga controlar informações referente sua frota de veículos e seu controle financeiro, portanto serão desenvolvidos dois módulos, uma de gestão de frota e um de gestão financeira, ambos atendendo todos os requisitos no qual o cliente solicitou. Esta Usina tem como principal renda a venda de açúcar e álcool que são vendidos tanto para o Brasil quanto para o exterior.

## **4 – Conceitos gerais em um projeto software**

### **4.1 – Engenharia de Requisitos**

Entendemos engenharia de software como sendo, de acordo com o IEEE, a aplicação de uma abordagem sistemática, disciplinada e quantificável no desenvolvimento, operação e manutenção de software. Sistemática por que parte do princípio de que existe um processo de desenvolvimento definindo as atividades que deverão ser executadas. Disciplinada por que parte do princípio de que os processos definidos serão seguidos. Quantificável por que se deve definir um conjunto de medidas a serem extraídas do processo durante o desenvolvimento de forma que as tomadas de decisão relacionadas ao desenvolvimento do software (por exemplo, melhoria de processo) sejam embasadas em dados reais, e não em “achismos”. Alguns de seus principais objetivos são:

- Qualidade de software;
- Produtividade no desenvolvimento, operação e manutenção de software;
- Permitir que profissionais tenham controle sobre o desenvolvimento de software dentro de custos, prazos e níveis de qualidade desejados.

Entretanto, o cenário de desenvolvimento de software atual e o cenário idealizado junto à engenharia de software ainda estão distantes. Vários fatores contribuem para isso, podemos citar dois:

- O não uso dos fundamentos da engenharia de software para apoiar as atividades do desenvolvimento;
- O mau uso dos fundamentos da engenharia de software para apoiar as atividades do desenvolvimento.

Isso tem diversas consequências. Gostaríamos de destacar neste artigo o crescente custo com manutenção dos sistemas. Consideramos como manutenção neste artigo como sendo qualquer retrabalho (em nível de requisitos, projeto, codificação, teste) causado por uma definição do domínio do problema mal elaborada nas fases iniciais do desenvolvimento.

### **4.2 – Requisitos de Software**

Antigamente dizia-se que requisitos eram sinônimos de funções, ou seja, tudo que o software deveria fazer funcionalmente. No entanto, atualmente assumiu-se que requisitos de software é muito mais do que apenas funções. Requisitos são, além de funções, objetivos, propriedades, restrições que o sistema deve possuir para satisfazer contratos, padrões ou especificações de acordo com o(s) usuário(s). De forma mais geral um requisito é uma condição necessária para satisfazer um objetivo.

Portanto, um requisito é um aspecto que o sistema proposto deve fazer ou uma restrição no desenvolvimento do sistema. Vale ressaltar que em ambos os casos devemos sempre

contribuir para resolver os problemas do cliente e não o que o programador ou um arquiteto deseja. Dessa forma, o conjunto dos requisitos como um todo representa um acordo negociado entre todas as partes interessadas no sistema. Isso também não significa que o programador, arquiteto ou um analista bem entendido no assunto de tecnologia não possam contribuir com sugestões e propostas que levem em conta o desejo do cliente.

Além disso, ainda temos um documento de requisitos que é uma coleção dos requisitos.

Por fim, os requisitos possuem alguns objetivos centrais como estabelecer e manter uma concordância com os clientes e outros envolvidos sobre o que o sistema deve fazer, deve oferecer aos desenvolvedores, projetistas e testadores do sistema uma compreensão melhor dos requisitos do sistema, definir fronteiras do sistema definindo o que deve ser incluído e o que não deve fazer parte do sistema, fornecer uma base para estimar o custo e o tempo de desenvolvimento do sistema e por fim definir uma interface de usuário para o sistema.

## **5 – Descrição das atividades**

### **5.1 – Elicitação**

**Descrição:** A eliciação de requisitos é uma fase muito importante em qualquer projeto de desenvolvimento de software, pois se elaborada de maneira incorreta, todo o projeto estará comprometido. É importante, pois é nesse momento que identificamos e definimos a estratégia de desenvolvimento do produto/serviço.

**Como foi elaborado:** cliente nos procurou relatando o problema existente quanto a dificuldade na organização dos dados. Foi um problema de fácil percepção pois está ligada a rotina da ONG. Propomos desenvolver um sistema simples que acate todas as necessidades quanto a organização e automação dos dados. Para essa etapa foi elaborado o documento de Visão (Anexo A).

### **5.2 – Análise e Negociação**

**Descrição:**

#### **5.2.1 – Atividades envolvidas**

Algumas das atividades envolvidas na análise de requisitos incluem:

**Classificação:** agrupamento de requisitos em "módulos" para facilitar a visão global do funcionamento pretendido para o sistema;

**Resolução de conflitos:** dada a multiplicidade e diversidade de papéis das partes interessadas envolvidas na captura e análise de requisitos, é inevitável a existência de conflitos nos requisitos identificados; é importante resolver estes conflitos o mais breve possível;

**Priorização:** consiste na atribuição de uma "prioridade" a cada requisito (por exemplo elevada/média/baixa); obviamente, este pode ser um fator gerador de conflitos;

**Confirmação:** é confirmada com as partes interessadas a completude dos requisitos, sua consistência e validade (de acordo com o que se pretende do sistema).

Estas fases não são independentes entre si, pois uma informação obtida numa delas pode servir para as demais fases.

A identificação e análise de requisitos é um processo iterativo que se inicia com a familiarização do domínio do futuro sistema e termina na confirmação dos requisitos, aumentando o grau de compreensão do sistema a cada ciclo de trabalho.

### 5.2.2 – Dificuldades

As dificuldades encontradas na análise são de diversas naturezas:

Fatores externos (políticos) podem influenciar os requisitos (alguma parte interessada, com poder de decisão, pode exigir requisitos específicos que sirvam aos seus interesses e não aos da organização, ou forçar o seu ponto de vista em detrimento dos demais interessados que irão operar o sistema);

O ambiente (econômico e/ou organizacional) em que a análise é feita possui fatores dinâmicos, e como tal, os requisitos estão sujeitos a alterações em decorrência destes (por exemplo: novas partes interessadas são envolvidas no projeto, ou alterações em prazos e orçamentos disponíveis).

### 5.2.3 – Negociações

Na fase de negociação, tornam-se necessários alguns cuidados para que esta decorra sem problemas, chegando-se logo a consensos. Algumas sugestões são:

Saber lidar com ataques pessoais (evitando-os sempre que possível, remetendo a sua resolução para mais tarde, fora de reunião), de preferência nunca tomando partidos;

Fomentar a justificação das posições (negativas) tomadas pelos intervenientes na negociação;

Salientar (e procurar encontrar) os benefícios que uma solução apresenta para todos os envolvidos;

Relaxar restrições, quando se torna óbvio que as atuais não levarão a um consenso.

**Como foi elaborado:** Após termos a definição do problema, identificamos os requisitos e os priorizamos através de uma lista negociada juntamente com o cliente. Os resultados podem ser vistos no documento de Regras de Negócios (Anexo B).

## 5.3 – Especificação

**Descrição:** é nesta fase que se dá a produção propriamente dita do Documento de Especificação de Requisitos.

Em todos os tipos de especificação há 2 tipos de requisitos a considerar:

Requisitos funcionais: descrevem as funcionalidades que se espera que o sistema disponibilize, de uma forma completa e consistente. É aquilo que o utilizador espera que o sistema ofereça, atendendo aos propósitos para qual o sistema será desenvolvido.

Requisitos não-funcionais: referem-se a aspectos não-funcionais do sistema, como



restrições nas quais o sistema deve operar ou propriedades emergentes do sistema. Costumam ser divididos em Requisitos não-funcionais de: Utilidade, Confiança, Desempenho, Suporte e Escalabilidade.

Pode-se também considerar os requisitos do domínio, que tal como o nome indica derivam do domínio e não de necessidades específicas dos usuários, podendo depois ser classificados como funcionais ou não-funcionais.

A documentação produzida poderá ter diferentes destinatários e como tal diferentes objetivos. Podem-se distinguir três tipos de especificação:

Especificação de requisitos do usuário ou utilizador;

Especificação de requisitos do sistema;

Especificação do design da aplicação.

A vantagem de conceber mais do que uma especificação para um dado sistema é a de em cada especificação se comunicar apenas um determinado tipo de informação adequado ao leitor a que se destina (usando "linguagens" que o utilizador conheça). Por exemplo, enquanto que nos requisitos do utilizador apenas é feita uma abordagem de alto nível das funcionalidades do sistema e suas restrições, usando linguagem natural e eventualmente diagramas (esquemas), nos requisitos do sistema cada requisito é descrito com mais detalhe introduzindo já alguns conceitos relativos à arquitetura do sistema, fazendo-se uso de linguagens estruturadas (notações gráficas como diagramas de casos de uso).

**Como foi elaborado:** após o entendimento do que deveria ser desenvolvido especificamos os requisitos obtidos. O resultado pode ser visto no documento de Especificação de Requisitos de Software (Anexo C).

## 5.4 – Modelagem

**Descrição:** é a atividade de construir modelos que expliquem as características ou o comportamento de um software ou de um sistema de software. Na construção do software os modelos podem ser usados na identificação das características e funcionalidades que o software deverá prover (análise de requisitos), e no planeamento de sua construção.

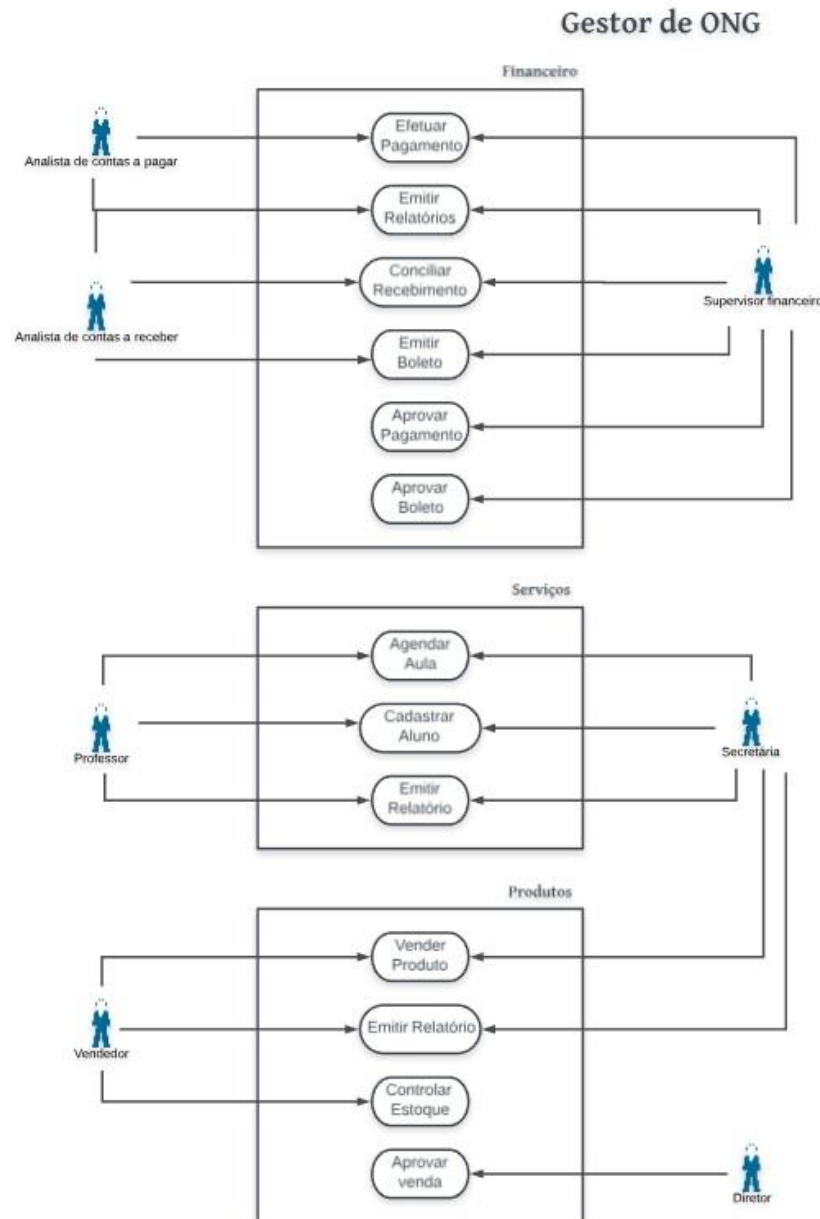
Frequentemente a modelagem de software usa algum tipo de notação gráfica e são apoiados pelo uso de Ferramentas CASE.

A modelagem de software normalmente implica a construção de modelos gráficos que simbolizam os artefatos dos componentes de software utilizados e os seus inter-relacionamentos. Uma forma comum de modelagem de programas procedurais (não orientados a objeto) é através de fluxogramas, enquanto que a modelagem de programas orientados a objeto normalmente usa a linguagem gráfica UML.

**Como foi elaborado:** com os requisitos descritos, elaboramos sua modelagem e criamos o diagrama de caso de uso que pode ser visto abaixo. Para criar o diagrama foi utilizado a plataforma web Lucidchart, o resultado pode ser visto abaixo ou através do link:

<https://www.lucidchart.com/invitations/accept/d5c34c33-7503-47d7-8914-7cd77a95bc9f>

.O detalhamento pode ser visto no documento de Especificação de Casos de Uso (Anexo D).



## 5.5 – Validação

**Descrição:** nesta fase pretende-se demonstrar que o documento de requisitos produzido corresponde, de fato, ao sistema que o cliente pretende.

À semelhança do que sucede na análise dos requisitos, pretende-se encontrar problemas/conflitos na especificação, porém ao contrário das fases anteriores esta fase lida com uma especificação completa dos requisitos.

A validação é especialmente importante em sistemas de grandes dimensões uma vez que erros encontrados demasiado tarde (durante o desenvolvimento ou já depois de o sistema estar a ser usado) no documento de requisitos têm repercussões proporcionais à dimensão do projeto. Uma vez que alterações em requisitos já consolidados têm um custo muito superior a alterações no código ou design, este tipo de erro traduz-se em elevados custos e necessidade de refazer muito do trabalho que se julgava já concluído.

Durante a fase de validação dos requisitos, devem ser verificados (através de checklists) os seguintes atributos dos requisitos:

**Validade:** a especificação resulta da análise dos requisitos identificados junto das diversas partes interessadas envolvidas. Como tal, requisitos identificados individualmente (isto é, junto de cada parte interessada) podem diferir da especificação final que se atinge após o cruzamento de informação e é necessário que cada cliente compreenda e aceite a especificação final obtida.

**Consistência:** não devem existir conflitos entre os requisitos identificados.

**Compreensibilidade / Ambiguidade:** os requisitos devem poder ser compreendidos de forma inequívoca pelas partes interessadas.

**Completude:** todas as funcionalidades pretendidas devem fazer parte da especificação do sistema.

**Realismo:** dadas as restrições do projeto (tecnológicas, financeiras e temporais) o sistema especificado tem de ser implementável.

**Verificabilidade:** de forma a evitar futuras discordâncias quanto à concretização dos requisitos especificados, estes devem ser descritos de modo a que seja possível verificar se foram ou não concretizados, isto é, se o sistema final corresponde à especificação inicial.

**Rastreabilidade:** a origem dos requisitos, em relação ao cliente, deve estar claramente identificada. Entre outros motivos, isto é importante para facilitar a gestão futura dos

requisitos.

Conformidade com normas: para além dos aspectos funcionais dos requisitos, a sua especificação deve obedecer às normas usadas ao longo de todo o documento.

**Como foi elaborado:** ao final foi elaborado um documento de validação com um checklist de questionário o qual o cliente preencheu e assinou validando os dados levantados. Tal validação pode ser vista no documento de aceitação do cliente (Anexo E).

## 5.6 – Gestão

Matriz de rastreabilidade valor da célula preenchido com 1 significa que o requisito da linha interfere no requisito da coluna correspondente caso esteja preenchido com 0 não interfere.

Requisitos Funcionais	Efetuar Pagamento	Emitir Relatórios Financeiros	Conciliar Recebimento	Emitir Boleto	Aprovar Pagamento	Aprovar Boleto	Agendar Aula	Cadastrar Aluno	Emitir Relatórios de Serviços	Vender Produtos	Emitir Relatório de Produtos	Controlar Estoque	Aprovar Venda
Efetuar Pagamento		1	0	0	1	0	0	0	0	0	0	0	0
Emitir Relatórios Financeiros	0		0	0	0	0	0	0	0	0	0	0	0
Conciliar Recebimento	0	1		0	0	0	0	0	0	0	0	0	0
Emitir Boleto	0	1	0		0	1	0	0	0	0	0	0	0
Aprovar Pagamento	1	1	0	0		0	0	0	0	0	0	0	0
Aprovar Boleto	0	1	0	1	0		0	0	0	0	0	0	0
Agendar Aula	0	0	0	0	0	0		0	1	0	0	0	0
Cadastrar Aluno	0	0	0	0	0	0	0		1	0	0	0	0
Emitir Relatórios de Serviços	0	0	0	0	0	0	0	0		0	0	0	0
Vender Produtos	0	0	0	0	0	0	0	0	0		1	1	1
Emitir Relatório de Produtos	0	0	0	0	0	0	0	0	0	0		0	0
Controlar Estoque	0	0	0	0	0	0	0	0	0	1	1		0
Aprovar Venda	0	0	0	0	0	0	0	0	0	1	1	1	

## **6 – Conclusão**

As principais dificuldades foram abstrair as necessidades do cliente de forma sucinta e elaborar uma documentação detalhada sobre os mesmos.

O trabalho é importante para o crescimento profissional pois nos dá o entendimento das principais dificuldades a serem enfrentadas na elaboração da documentação de análise de requisitos e ainda mostra o caminho a seguir para elaboração do mesmo.

## 7 – Bibliografia

<https://www.devmedia.com.br/introducao-a-requisitos-de-software/29580>

acesso em: 21/04/2018

<https://www.passeidireto.com/disciplina/engenharia-de-software-2>

acesso em: 24/04/2018

[https://pt.wikipedia.org/wiki/Engenharia\\_de\\_requisitos#An%C3%A1lise\\_e\\_negocia%C3%A7%C3%A3o\\_dos\\_requisitos](https://pt.wikipedia.org/wiki/Engenharia_de_requisitos#An%C3%A1lise_e_negocia%C3%A7%C3%A3o_dos_requisitos)

acesso em: 28/04/2018

[https://pt.wikipedia.org/wiki/Modelagem\\_de\\_software](https://pt.wikipedia.org/wiki/Modelagem_de_software)

acesso em: 29/04/2018

<https://www.lucidchart.com>

acesso em: 05/05/2018

[http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup\\_wd\\_tmpl/bm/rup\\_brul.dot](http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup_wd_tmpl/bm/rup_brul.dot)

acesso em: 06/05/2018

[http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup\\_wd\\_tmpl/req/rup\\_vision\\_sp.dot](http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup_wd_tmpl/req/rup_vision_sp.dot)

acesso em: 10/05/2018

[http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup\\_wd\\_tmpl/bm/rup\\_brul.dot](http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup_wd_tmpl/bm/rup_brul.dot)

acesso em: 11/05/2018

[http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup\\_wd\\_tmpl/req/rup\\_srsuc.dot](http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup_wd_tmpl/req/rup_srsuc.dot)

## **8 – Anexos (Templates do RUP)**

### **8.1 – Anexo A – Elicitação**

**Gestor de ONG**  
**Visão (Projeto Pequeno)**

**Versão 1.0**



### Histórico da Revisão

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
01/05/2018	0.1	Obtenção dos dados	Jonas R. Denardi
05/05/2018	0.5	Ajuste	Marcelo H. Ramos
08/05/2018	0.8	Ajuste	Jonas R. Denardi
10/05/2018	1.0	Final	Marcelo H. Ramos

## **Índice Analítico**

1.	Introdução	19
1.1	Referências	19
2.	Posicionamento	19
2.1	Descrição do Problema	19
2.2	Sentença de Posição do Produto	19
3.	Descrições dos Envolvidos e Usuários	20
3.1	Resumo dos Envolvidos	5
3.2	Resumo dos Usuários	5
3.3	Ambiente do Usuário	6
3.4	Resumo das Principais Necessidades dos Envolvidos ou Usuários	6
3.5	Alternativas e Concorrência	6
4.	Visão Geral do Produto	6
4.1	Perspectiva do Produto	6
4.2	Suposições e Dependências	6
5.	Recursos do Produto	6
5.1	Financeiro	6
5.2	Produtos	7
5.3	Serviços	7
6.	Outros Requisitos do Produto	7

## **Visão (Projeto Pequeno)**

### **1. Introdução**

ONG Jovens Ambientalistas trata-se de uma instituição que recolhe, educa e oferece formação profissionalizante para jovens sem lar que depois de receberem cursos gratuitos por professores que são ex-alunos, prestam serviços remunerados, fabricando brinquedos “ambientalmente corretos” que são vendidos para o Brasil e o exterior. A instituição vem enfrentando problemas em processos nas áreas de serviços, produtos e controle financeiro.

No módulo de serviços necessita-se de um controle em relação as aulas dadas pelos professores aos alunos recolhidos, controlando horários de aulas, alunos, matérias dadas e carga horária de cada professor pois com o aumento da instituição houve a perda do controle desses dados que são feitos por meio de anotações.

No módulo de produtos necessita-se de um controle de estoque e qual aluno montou o produto pois com o aumento da instituição houve a perda do controle desses dados que também são feitos por meio de anotações.

No módulo financeiro necessita-se de um controle de contas a pagar, contas a receber e emissão de boletos por DDA. O controle contas a pagar e contas a receber é feito por meio de anotações em caderno no formato de fluxo de dados o que se torna inviável e improdutivo. Os recebimentos são feitos apenas em dinheiro, a implementação do DDA possibilitará a emissão e envio de boletos aos clientes.

#### **1.1 Referências**

Orientações para a disciplina de Atividades Práticas Supervisionadas 6º/7º Ciência da Computação (CC), 16/04/2018.

### **2. Posicionamento**

#### **2.1 Descrição do Problema**

O problema	Grande volume de informações para se controlar
afeta	Setor financeiro, professores, administração em geral
cujo impacto é	Perda de informações devido ao controle manual das mesmas
uma boa solução seria	Automatizar o controle de informações

#### **2.2 Sentença de Posição do Produto**

Para	ONGs
Quem	Controle de informações
O (nome do produto)	Gestor de ONGs
Que	Gestão de informações financeiras, produtos, serviços.
Diferente de	Sistemas burocráticos de difícil interação
Nosso produto	Simplicidade, praticidade e geração de boletos DDA

### 3. Descrições dos Envolvidos e Usuários

-Analista de contas a pagar: efetua os pagamentos aos fornecedores.

-Analista de contas a receber: controle os recebimentos dos clientes.

-Supervisor financeiro: controla todas as partes financeiras (contas a pagar, receber e fluxo).

-Professores: agendamentos de aulas.

-Alunos: fabricam os brinquedos.

-Secretária: ajuda com os agendamentos de aulas e vendas.

-Vendedor: faz as vendas e controle de estoque.

-Administrador: faz o controle geral das operações.

-Diretor: gerencia todos os setores, toma as decisões finais

Todos os relacionados acima faziam seus controles por meio de anotações em cadernos.

#### 3.1 Resumo dos Envolvidos

Nome	Descrição	Responsabilidades
Aluno	Jovens sem lar que recebem formação profissionalizante da ONG	Após o término do curso prestam serviços remunerados, fabricando brinquedos “ambientalmente corretos” que são vendidos para o Brasil e o exterior
Administrador	Controla todas as operações da empresa	Verifica necessidades dos setores e acompanha o desenvolvimento dos jovens resgatados

#### 3.2 Resumo dos Usuários

Nome	Descrição	Responsabilidades	Envolvido
------	-----------	-------------------	-----------

Anal. de contas a pagar	Efetua operações gerais de pagamentos	Pagamentos, impressão de relatórios, conferências	
Anal. de contas a receber	Efetua operações gerais de recebimentos	Conciliação recebimentos, emissão de relatórios e boletos, controle de inadimplentes	
Supervisor financeiro	Controle geral da área financeira	Aprovação de pagamentos, impressão de relatórios, conferências, controle de fluxo de caixa	
Secretária	Agendamento de aulas	Agendamento de aulas e matérias dadas pelos os professores, vendas dos produtos, cadastro de alunos	
Vendedor	Vendas dos brinquedos e controle de estoque	Vendas dos produtos, controle de estoque, emissão de relatórios	
Professor	Agenda aulas	Agendamento de aulas e matérias dadas e cadastro de alunos	
Diretor	Tomas das decisões finais das operações importantes	Aprova orçamentos, vendas, projetos, etc	

### 3.3 Ambiente do Usuário

O ambiente é formado por 11 pessoas compostas por 1 analista de contas a pagar, 1 analista de contas a receber, 1 supervisor financeiro, 1 secretária, 1 administrador, 1 diretor e 5 professores.

É possível que este número aumente devido crescimento da ONG.

Atualmente o controle das operações são feitas em anotações em cadernos levando muito tempo para se concretizarem.

No ambiente não há nenhum sistema sendo utilizado que necessitará de integração com o Gestor de ONGs.

### 3.4 Resumo das Principais Necessidades dos Envolvidos ou Usuários

Necessidades dos usuários da área financeira são: sistema simples que organize automaticamente as informações gerando relatórios para análise dos dados e opção de geração de boletos.

Necessidades do vendedor são: sistema de controle de estoque e vendas.

Necessidades dos professores: agenda práticas com informações claras sobre as aulas e cadastro de alunos.

Necessidades da secretária: como suas tarefas envolvem ajudar o vendedor e professores as necessidades são as mesmas deles.

### **3.5 Alternativas e Concorrência**

-Pontos fortes: controle da área fiscal e contábil.

-Pontos fracos: sistema burocrático e de difícil aprendizado.

## **4. Visão Geral do Produto**

### **4.1 Perspectiva do Produto**

Sistema completo moldado para as necessidades específicas da ONG Jovens Ambientalistas com o intuito de satisfazer as necessidades das áreas financeira, vendas e controle de serviços.

O sistema tem o intuito de gerar o menos telas possíveis para o usuário facilitando e simplificando seu trabalho.

### **4.2 Suposições e Dependências**

Alteração no sistema de DDA dos bancos poderá afetar a emissão dos boletos por DDA no sistema.

## **5. Recursos do Produto**

### **5.1 Financeiro**

-Seleção de contas a pagar: onde o usuário consultar, alterar, pagar, emitir relatórios das despesas da empresa.

-Seleção de contas a receber: onde o usuário consultar, alterar, conciliar, emitir relatórios dos recebimentos da empresa e emitir boletos para envio por e-mail ou DDA.

-Fluxo de caixa: onde é possível ver as despesas pagas, não pagas, recebimentos pendentes e já recebido, previsões de pagamentos, etc.

Cadastro de clientes e fornecedores: onde se cadastra, edita e exclui os clientes e/ou fornecedores.

### **5.2 Serviços**

-Cadastro de professores e alunos: onde se cadastra, edita e exclui os alunos e/ou professores.

-Agenda de aulas dos professores.

### **5.3 Produtos**

-Cadastro de produtos: onde se cadastra, edita e exclui os produtos

-Estoque: controle do estoque. Por exemplo não possibilitar venda de um produto esgotado ou quantidade maior que a existente no estoque.

## **6. Outros Requisitos do Produto**

Sistema deve rodar sob sistema operacional Windows 7 ou superior.

Arquitetura do sistema operacional e do processador de 32 ou 64bits.

Pelo menos 2 gigas de memória ram.

Processador Intel Core 2 Duo ou superior.

Sistema deve ser simples e prático com telas diretas nos pontos de interesse sem precisar passar por várias telas para processar uma tarefa.

Manual do usuário contendo as instruções ilustradas de uso.

## **8.2 – Anexo B - Análise e Negociação**



**Gestor de ONG**  
**Regras de Negócios**

**Versão <1.0>**

#### Histórico da Revisão

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
01/05/2018	0.1	Obtenção dos dados	Jonas R. Denardi
05/05/2018	0.5	Ajuste	Marcelo H. Ramos
08/05/2018	0.8	Ajuste	Jonas R. Denardi
10/05/2018	1.0	Final	Marcelo H. Ramos

## Índice Analítico

1.	Introdução	28
1.1	Finalidade	28
1.2	Escopo	28
1.3	Referências	28
1.4	Visão Geral	28
2.	Definições	28
2.1	Financeiro	4
2.1.1	FIN01	4
2.1.2	FIN02	4
2.1.3	FIN03	4
2.1.4	FIN04	4
2.2	Serviços	<b>Erro! Indicador não definido.</b>
2.2.1	SER01	<b>Erro! Indicador não definido.</b>
2.3	Produtos	<b>Erro! Indicador não definido.</b>
2.4.1	PRO01	<b>Erro! Indicador não definido.</b>
2.4.2	PRO02	<b>Erro! Indicador não definido.</b>

# Regras de Negócios

## 1. Introdução

Aqui encontra-se toda regra de negócio da empresa, particularidades nos módulos e restrições.

### 1.1 Finalidade

Auxiliar no entendimento do funcionamento do sistema e suas restrições.

### 1.2 Escopo

Este documento está ligado ao sistema Gestor de ONG versão 1.0 e todos os seus módulos.

### 1.3 Referências

7ºsemestreCC.docx

### 1.4 Visão Geral

As regras de negócios estão divididas em 3 grupos: financeiro, serviços e produtos.

## 2. Definições

### 2.1 Financeiro

#### 2.1.1 FIN01

Toda despesa deve ter um identificador único.

#### 2.1.2 FIN02

Todo processo de pagamento deve ser aprovado pelo supervisor financeiro.

#### 2.1.3 FIN03

Toda receita deve ter um identificador único.

#### 2.1.4 FIN04

Toda emissão de boletos deve ser aprovada pelo supervisor financeiro.

### 2.2 Serviços

#### 2.2.1 SER01

Todo agendamento de aulas deve ser aprovado pelo professor correspondente.

### 2.3 Produtos

#### 2.3.1 PRO01

Toda venda e orçamento deve ser aprovado pelo diretor da empresa.

#### 2.3.2 PRO02

Não se pode vender um produto sem estoque.

### **8.3 – Anexo C – Especificação**

# **Gestor de ONG**

## **Especificação dos Requisitos de Software**

**Versão 1.0**

## Histórico da Revisão

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
01/05/2018	0.1	Obtenção dos dados	Jonas R. Denardi
05/05/2018	0.5	Ajuste	Marcelo H. Ramos
08/05/2018	0.8	Ajuste	Jonas R. Denardi
10/05/2018	1.0	Final	Marcelo H. Ramos

# Índice Analítico

1.	Introdução	33
1.1	Finalidade	33
1.2	Escopo	33
1.3	Definições, Acrônimos e Abreviações	33
1.4	Referências	33
1.5	Visão Geral	33
2.	Descrição Geral	33
2.1	Relatório Sintético de Modelo de Casos de Uso	33
2.2	Suposições e Dependências]	33
3.	Requisitos Específicos	5
3.1	Relatórios de Caso de Uso	5
3.2	Requisitos Suplementares	6
4.	Informações de suporte	6
4.1	Índice	6



# Especificação dos Requisitos de Software

## 3. Introdução

### 3.1 Finalidade

Este documento contém todas as informações sobre o levantamento de requisitos com o cliente, abrangendo os detalhes da necessidade de cada setor.

O sistema irá automatizar os processos das áreas citadas. Devendo ter visual simples e intuitivo, segurança nas operações e velocidade.

### 3.2 Escopo

O Gestor de ONGs é de fácil entendimento e muito intuitivo, levando pouco tempo para o aprendizado.

É seguro em suas operações e organiza as informações de forma eficiente.

### 3.3 Definições, Acrônimos e Abreviações

DDA: = “Débito Direto Autorizado”, sistema que permite que todos os compromissos de pagamentos sejam recebidos eletronicamente, por meio dos bancos que atendem pessoas físicas e jurídicas. Os valores das transações, a exemplo do que já ocorre hoje, serão automaticamente creditados na conta corrente dos clientes credores.

### 3.4 Referências

[http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup\\_wd\\_tmpl/req/rup\\_srsuc.dot](http://www.funpar.ufpr.br:8080/rup/wordtmpl/rup_wd_tmpl/req/rup_srsuc.dot) e 7ºsemestreCC.docx.

### 3.5 Visão Geral

Documento contém a relação de casos de uso seguindo a obtenção dos requisitos funcionais e não funcionais.

## 4. Descrição Geral

### 4.1 Relatório Sintético de Modelo de Casos de Uso

Analista de contas a pagar: efetuar pagamento e emitir relatórios financeiros.

Analista de contas a receber: conciliar recebimento, emitir boletos e relatórios financeiros.

Supervisor financeiro: efetuar pagamento, emitir relatórios financeiros, conciliar recebimento, emitir boletos, aprovar pagamentos e aprovar boletos.

Professor: agendar de aula, cadastrar aluno, emitir relatório de serviços.

-Secretária: agendar de aula, cadastrar aluno, emitir relatório de serviços, vender produto e emitir relatório de produtos.

-Vendedor: vender produto, emitir relatório de produtos, controlar estoque.

-Diretor: aprovar venda.

### 4.2 Suposições e Dependências

Todos os módulos estão disponíveis para uso.

A dependência existente fica por conta da conexão com o sistema de DDA dos bancos escolhidos para emissão dos boletos.

## 5. Requisitos Específicos

### 5.1 Relatórios de Caso de Uso

[RF01] Efetuar Pagamento

Possibilita a emissão e alteração de pagamentos a fornecedores contendo as informações do título a pagar e fornecedor.

[RF02] Emitir Relatórios Financeiros

Possibilita a emissão de relatórios dos títulos pagos, a pagar e vencidos. Contém filtros de data, fornecedor e status (pagos, a pagar e vencidos).

Possibilita a emissão de relatórios dos títulos recebidos, a pagar e vencidos. Contém filtros de data, cliente e status (recebidos, a receber e vencidos).

[RF03] Conciliar Recebimento

Possibilita a baixa e alteração de recebimentos de clientes contendo as informações dos títulos recebidos, a receber e recebidos e do cliente.

[RF04] Emitir Boleto

Possibilita a emissão de boletos a partir dos títulos gerados a cliente podendo escolher conta de emissão do boleto e vencimento.

[RF05] Aprovar Pagamento

Aprova a emissão de pagamento referente ao [RF01].

[RF06] Aprovar Boleto

Aprova a emissão de boleto referente ao [RF04].

[RF07] Agendar Aula

Possibilita o agendamento de aulas dos professores contendo nome do professor, matéria a ser aplicada, data e horário.

[RF08] Cadastrar Aluno

Possibilita o cadastro de alunos contendo nome do aluno, CPF, ano de nascimento e endereço.

[RF09] Emitir Relatórios de Serviços

Possibilita a emissão de relatórios de aulas dadas contendo nome dos professores, matérias aplicadas, datas e horários. Contém filtros de datas, professor e matéria aplicada.

[RF10] Vender Produtos

Possibilita a venda e alteração de venda a cliente contendo as informações do cliente e produto a ser vendido.

[RF11] Emitir Relatório de Produtos

Possibilita a emissão de relatórios de produtos cadastrados mostrando nome do produto e quantidade no estoque. Contém filtros de produto e quantidade.

Possibilita a emissão de relatórios de venda de produtos contendo dados do produto quantidade vendida e dados do cliente. Contém filtros de datas, produto e cliente.

[RF12] Controlar Estoque

Possibilita cadastrar e alterar produto no estoque e alterar sua quantidade.

[RF13] Aprovar Venda

Aprova a venda de produto referente ao [RF10].

[RNF01] Simplicidade

Sistema deve ser intuitivo para o usuário e de fácil aprendizado.

[RNF02] Segurança

Módulo financeiro deve ser seguro no processo das operações

## **5.2 Requisitos Suplementares**

[RS01] Ajuda nas Telas

Todas as telas devem ter campo de ajuda.

## **6. Informações de suporte**

### **6.1 Índice**

Definições, Acrônimos e Abreviações	33
Escopo	33
Finalidade	33
Referências	33
Relatórios de Caso de Uso	5
Relatório Sintético de Modelo de Casos de Uso	33
Requisitos Suplementares	6
Visão Geral	33
Suposições e Dependências	33

## **8.4 – Anexo D - Modelagem**

# **Gestor de ONG's**

## **Especificação de Caso de Uso**

**Versão 1.0**

## Histórico da Revisão

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
01/05/2018	0.1	Obtenção dos dados	Jonas R. Denardi
05/05/2018	0.5	Ajuste	Marcelo H. Ramos
08/05/2018	0.8	Ajuste	Jonas R. Denardi
10/05/2018	1.0	Final	Marcelo H. Ramos

# Índice

1.	Breve Descrição	40
	O objetivo de cada	
2.	Fluxo Básico de Eventos	40
3.	Fluxos Alternativos	41
3.1	Área de Funcionalidade	41
3.1.1	Executar pagamento	<b>Erro! Indicador não definido.</b>
3.1.2	Emitir relatório	<b>Erro! Indicador não definido.</b>
3.1.3	Agendar aula	<b>Erro! Indicador não definido.</b>
3.1.4	Emite relatório de produtos	<b>Erro! Indicador não definido.</b>
3.1.5	Cadastra aluno	6
3.1.6	Conciliação	6
4.	Subfluxos	6
4.1	Agendar aula	6
4.2	Executar pagamento	7
5.	Cenários Chave	7
6.	Condições Prévias	7
6.1	O sistema deve estar on-line	7
6.2	O sistema deve estar em um estado inicial de cada módulo	7
7.	Condições Posteriores	8
7.1	Em processamento	8
7.2	Disponível	8
8.	Pontos de Extensão	8
8.1	Site do Banco	8
8.2	Importação de Notas Fiscais Eletrônicas	8
9.	Requisitos Especiais	8
9.1	Ambiente do sistema	8
9.2	Designer do sistema	8
10.	Informações Adicionais	8

# Especificação de Caso de Uso

## 7. Breve Descrição

Os casos de uso têm objetivo de auxiliar a comunicação entre o desenvolvedor e o cliente. Ele descreve um cenário que mostra as funcionalidades do sistema pelo ponto de vista dos usuários.

## 8. Fluxo Básico de Eventos

Quando um ator emite um relatório o sistema pega os dados e completa de acordo com as informações atualizadas.

Quando o ator concilia um recebimento o mesmo verifica no extrato e faz baixa no sistema de acordo com o título em questão.

Quando o ator efetua pagamentos o sistema identifica o título em questão e automaticamente faz a baixa do título em sua base de dados.

Quando o ator emite um boleto o sistema preenche esse documento automaticamente de acordo com as informações desse cliente em sua base de dados e aguarda a aprovação da supervisão para a emissão do documento.

Quando o ator aprova um pagamento o sistema automaticamente faz a transação dos pagamentos em questão com os bancos já cadastrados no mesmo.

Quando o ator aprova a emissão de um boleto o sistema já tem as informações preenchidas no documento, executa o mesmo então somente a emissão do mesmo.

Quando um ator agenda uma aula o sistema verifica se o horário está disponível e se sim agenda a aula, caso contrário informa uma mensagem de que o horário está ocupado.

Quando um ator cadastra um aluno o sistema salva os dados em seu banco de dados de acordo com o que foi especificado pelo ator, completando os campos pré-definidos anteriormente.

Quando um ator vende um produto o sistema automaticamente verifica seu estoque e faz a baixa do mesmo em relação as quantidades vendidas.

Quando um ator controla o estoque o sistema apresenta para o mesmo todas as informações de cada produto que o usuário está pesquisando, quanto tem de cada produto no estoque e o quanto já foi vendido.

Quando um ator aprova uma venda o sistema automaticamente libera esse produto para o comprador após o pagamento e faz a baixa do mesmo no estoque.



## 9. Fluxos Alternativos

O ator sai da aplicação: O sistema cancela a operação automaticamente.

O ator cancela a operação corrente: O sistema acata o cancelamento.

O ator provê dados inválidos: O sistema não aceita os dados e emite uma informação de dados inválidos.

O ator provê dados incompletos: O sistema não aceita os dados e emite uma informação de dados incompletos.

O sistema falha: O sistema apresenta que correu um erro e pede para recomeçar.

O sistema está indisponível: O sistema apresenta uma mensagem que esta indisponível o serviço no momento.

### 9.1 Área de Funcionalidade

#### 9.1.1 *Executar pagamento*

A1: O usuário clicou executar pagamento.

1 - O sistema solicita aprovação.

2 - O usuário confirma.

3 - O sistema efetua o pagamento.

4 - O sistema imprime a mensagem "Pagamento efetuado com sucesso".

A2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

#### 9.1.2 *Emitir relatório*

A1: O usuário clicou emitir relatório.

1 - O sistema solicita aprovação.

2 - O usuário confirma.

3 - O sistema efetua a emissão.

4 - O sistema imprime a mensagem "Relatório emitido".

A2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

#### 9.1.3 *Agendar aula*

A1: O usuário clicou em agendar.

1 – O sistema verifica a disponibilidade.

2 - O sistema solicita aprovação.

3 - O usuário confirma.

4 - O sistema efetua o agendamento.

5 - O sistema imprime a mensagem "Aula agendada".

A2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

#### 9.1.4 *Emite relatório de produtos*

A1: O usuário clicou em emitir relatório de estoque.

1 - O sistema solicita aprovação.

2 - O usuário confirma.

3 - O sistema efetua a emissão do relatório.

4 - O sistema imprime a mensagem "Relatório gerado".

A2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

#### 9.1.5 *Cadastra aluno*

A1: O usuário clicou em cadastrar aluno.

1 - O sistema solicita aprovação.

2 - O usuário confirma.

3 - O sistema efetua o cadastro do aluno.

4 - O sistema imprime a mensagem "Aluno cadastrado".

A2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

#### 9.1.6 *Conciliação*

A1: O usuário clicou Conciliar.

1 - O sistema solicita aprovação.

2 - O usuário confirma.

3 - O sistema efetua a conciliação.

4 - O sistema imprime a mensagem "Título conciliado"

A2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

## **10. Subfluxos**

### **10.1 Agendar aula**

S1: O usuário clicou em agendar.

1 – O sistema verifica a disponibilidade.

2 - O sistema solicita aprovação.

3 - O usuário confirma.

4 - O sistema efetua o agendamento.

5 - O sistema imprime a mensagem "Aula agendada".

S2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

S3: Sistema off-line.

1 - O sistema retorna uma mensagem avisando que o sistema está off-line.

S4: Horário indisponível.

1 - O sistema retorna uma mensagem avisando que o horário está indisponível, para que seja escolhido outro horário.

### **10.2 Executar pagamento**

S1: O usuário clicou executar pagamento.

1 - O sistema solicita aprovação.

2 - O usuário confirma.

3 - O sistema efetua o pagamento.

4 - O sistema imprime a mensagem "Pagamento efetuado com sucesso".

S2: O usuário clicou em Não.

1 - O sistema retorna ao fluxo principal.

S3: Sistema off-line.

1 - O sistema retorna uma mensagem avisando que o sistema está off-line.

S4: Dados de banco inválidos.

1 - O sistema retorna uma mensagem avisando para que seja revisada as informações de banco.

## **11. Cenários Chave**

Pagamentos, Conciliações bancárias, agendamento de aula, controle de estoque, venda de produtos.

## **12. Condições Prévias**

12.1 O sistema deve estar on-line

12.2 O sistema deve estar em um estado inicial de cada módulo.

## **13. Condições Posteriores**

13.1 Em processamento.

13.2 Disponível.

## **14. Pontos de Extensão**

### **14.1 Site do Banco**

Link direto para conexão ao banco selecionado para pagamento e emissão de boletos DDA.

### **14.2 Importação de Notas Fiscais Eletrônicas**

Para uma futura necessidade existe um link direto para acesso ao portal de nota fiscal eletrônica da Receita Federal para a importação das notas fiscais emitidas em nome da ONG.

## **15. Requisitos Especiais**

### **15.1 Ambiente do sistema**

O sistema é compatível com sistema operacional Windows, na versão 7 ou superior.

### **15.2 Designer do sistema**

O sistema deve conter as cores, verde como principal seguido das cores amarelo e branco.

## **16. Informações Adicionais**

O sistema de geração de boletos DDA é semelhante ao desenvolvido pela empresa TOTVS em seu ambiente Logix.

## 8.5 – Anexo E - Validação

Sistema: Gestor de ONGs

Cliente: ONG Jovens Ambientalistas

### Validação de Requisitos

No	Item para Verificação	SIM	NÃO
1	Cada requisito está descrito com clareza, concisão e sem ambigüidade?	X	
2	Existem requisitos implícitos?		X
3	Os requisitos exibem a distinção clara entre funções, dados e restrições?	X	
4	Existem requisitos conflitantes?		X
5	As restrições e dependências foram claramente descritas?	X	
6	Existem requisitos que contêm algum nível desnecessário de detalhe do projeto?		X
7	Os requisitos definem todos os usuários do sistema?	X	
8	Os requisitos definem todas as informações a serem apresentadas aos usuários?	X	
9	Os requisitos descrevem as respostas do sistema ao usuário devido às condições de erro?	X	
10	Existem situações não tratadas pelos requisitos que precisam ser consideradas?	X	
11	Os requisitos não funcionais (tais como tempo de resposta, armazenamento de dados, etc.) foram definidos?	X	
12	O requisito é testável? Se não, que informação está faltando para torná-lo testável?	X	
13	O documento contém realmente toda a informação prometida em sua introdução?	X	

Identificação do Inspetor: Paulo Ricardo da Silva

Cargo do Inspetor ONG: Diretor

Nome do Projeto: Gestor de ONGs

Data: 12/05/2018

Os descritos abaixo estão de acordo com a avaliação acima e conseqüentemente com o levantamento dos requisitos.

---

Inspetor

---

Responsável pela ONG

---

MIX Soluções em Sistemas



## **9 – Ficha de Atividades Práticas Supervisionadas**

**Advantages:** It allows us to make qualitative, subjective and subjective evaluations, because the results are subjective, subjective and subjective.

**NOTE**

**PM:**

**CURSED**

**CALIFORNIA**

**SEMESTRE** *1<sup>o</sup> Semestre*

**TURKISH**

DATA	ATIVIDADE	TOTAL DE HORAS	ALUNO	ASSINATURA PROFESSOR
21/04	Teoria sobre o currículo.	3		
22/04	Teoria sobre o currículo.	4		
23/04	Desenvolvimento do projeto.	4		
24/04	Desenvolvimento do projeto.	5		
01/05	Desenvolvimento do projeto.	4		
09/05	Desenvolvimento do projeto.	5		
05/05	Desenvolvimento do projeto.	4		
06/05	Desenvolvimento do projeto.	5		
10/05	Desenvolvimento do projeto.	4		
11/05	Desenvolvimento do projeto.	5		
12/05	Desenvolvimento do projeto.	3		
13/05	Desenvolvimento do projeto.	2		

**TOTAL DE HORAS**

James P. Dwyer

15528

**0159**

Ciência da Computação

Wages and Living Prox

## SEMI-ESTER

~~10~~

# TURBO

Next

DATA	ATIVIDADE	TOTAL DE HORAS	ALUNO	ASSINATURA	PROFESSOR
02/04	Revisão sobre contabilidade	3			
03/04	Revisão sobre contabilidade	4			
07/04	Desenvolvimento de projeto	4			
08/04	Desenvolvimento de projeto	4			
09/04	Desenvolvimento de projeto	5			
11/05	Desenvolvimento de projeto	4			
04/05	Desenvolvimento de projeto	5			
05/05	Desenvolvimento de projeto	4			
06/05	Desenvolvimento de projeto	5			
11/05	Desenvolvimento de projeto	4			
11/05	Desenvolvimento de projeto	5			
10/05	Desenvolvimento de projeto	3			
13/05	Condição de projeto	2			

**TOTAL DE HORAS**

52