

ProGest: Sistema Web de Controle de Materiais nos Institutos Federais

Alexandre S. S. Soares¹, Danilo L. Santos², Pablo F. Matos³, Mark Rener S. Teixeira⁴

¹Discente de graduação em Sistemas de Informação - IFBA. e-mail: alexandre.soares@ifba.edu.br; ²Discente de graduação em Sistemas de Informação - IFBA. e-mail: danilo.lopes@ifba.edu.br; ³Professor do Curso de Bacharelado em Sistemas de Informação - IFBA. e-mail: pablofmatos@ifba.edu.br; ⁴Chefe de Patrimônio - IFBA. e-mail: markrener@ifba.edu.br

1 **RESUMO:** Soluções computacionais têm sido amplamente utilizadas na resolução de
2 diversos problemas. Assim, para que houvesse um melhor gerenciamento de itens no setor de
3 almoxarifado do Instituto Federal da Bahia *campus* Vitória da Conquista, foi desenvolvida
4 uma aplicação *web* para tal finalidade. Como resultado, o *software* desenvolvido pode
5 controlar entradas e saídas de itens em estoque, prover aos servidores uma interface para
6 solicitação de itens e gerar diversos relatórios gerenciais. A aplicação em questão utilizou-se
7 de *frameworks* para a agilização de seu desenvolvimento e foi implantada em regime de testes
8 no instituto, de forma que foi possível verificar adequações e melhorias a serem feitas.

9 **Palavras-chave:** almoxarifado, estoque, *software*

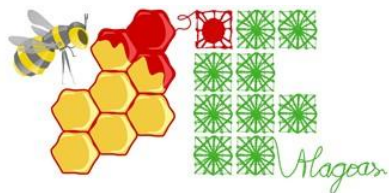
ProGest: Material Control Web System in Federal Institutes

13 **ABSTRACT:** Computing solutions has been widely used in solving various problems.
14 Therefore, so that there was better management of items in the warehouse of Federal Institute
15 of Bahia *campus* of Vitória da Conquista, a web application for this purpose was developed.
16 As a result, the developed software may control inputs and outputs of items in stock, give an
17 interface where the employees may request items and generate various management reports.
18 The application in question used frameworks to accelerate its development and it was
19 implemented in testing regime at the institute, so that it was possible to verify adjustments and
20 improvements to be made.

21 **KEYWORDS:** warehouse, stock, software

INTRODUÇÃO

24 A utilização de soluções computacionais para a resolução de problemas ou para o
25 controle de processos é uma prática comum. Pressman (2011, p. 55) evidencia a importância
26 dessas soluções afirmando que “o software distribui o produto mais importante da nossa era -
27 a informação. Ele transforma dados [...] de modo que possam ser mais úteis num determinado
28 contexto”. No entanto, a generalização destes *softwares* para uma determinada finalidade
29 pode comprometer os resultados alcançados com a sua adoção, de forma que a utilização de
30 uma solução específica e direcionada para o processo em questão pode trazer maiores
31 benefícios aos seus usuários, bem como ao contexto no qual o *software* atua.



32 Como exemplo, a utilização de um *software* comum para o gerenciamento de estoque e
33 suas movimentações pode funcionar perfeitamente para o ambiente comercial, mas não se
34 mostra igualmente adequada quando se trata de estoques de instituições públicas, onde a
35 entrada e saída de itens é regida diferentemente do modelo comercial.

36 Alguns trabalhos com a mesma abordagem podem ser observados na literatura, com
37 foco na melhoria de processos de ambientes específicos. Sobre este tema, Canzian (2010)
38 demonstra o processo de informatização das rotinas de pedidos e compras realizadas pelos
39 servidores da universidade onde o projeto foi desenvolvido, objetivando a otimização do
40 processo de compras. Já Paschoal e Castilho (2010) relatam o processo de implementação de
41 um sistema informatizado para gestão de materiais em um hospital universitário,
42 demonstrando quatro fases da implementação. O estudo demonstrou que, a aplicação do
43 sistema específico àquele ambiente contribuiu para a diminuição do consumo e do custo dos
44 materiais mantido em estoque.

45 Assim sendo, este trabalho busca demonstrar os resultados obtidos com o
46 desenvolvimento de um sistema web específico, denominado ProGest, para o gerenciamento
47 de estoque de Institutos Federais, bem como o seu processo de desenvolvimento e os
48 resultados iniciais alcançados com a sua utilização.

49

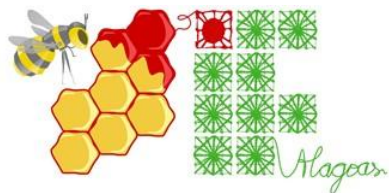
50 MATERIAL E MÉTODOS

51 Tendo como base as atividades genéricas do processo de desenvolvimento de *software*
52 definidas por Pressman (2011), pode-se dividir a construção do projeto descrito neste trabalho
53 nas seguintes etapas: planejamento, modelagem, desenvolvimento e implantação.

54 Planejamento

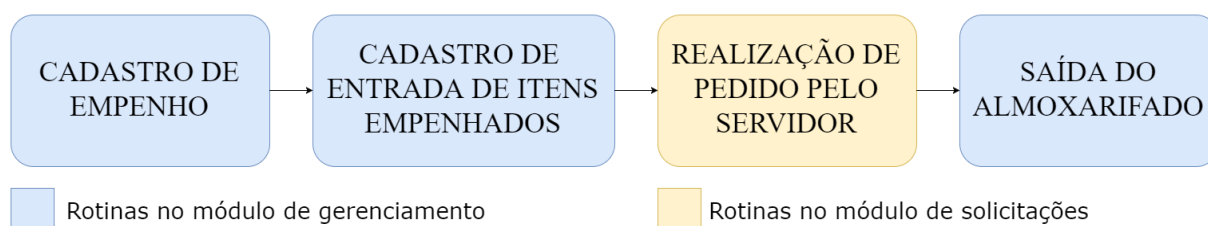
55 O objetivo do planejamento é a compreensão do problema a ser abordado, de modo a
56 planejar-se a sua melhor solução. Sendo assim, para a realização do projeto, foram realizadas
57 algumas tarefas para que os desenvolvedores pudessem enxergar o escopo e o modo de
58 funcionamento do meio onde o *software* deveria atuar. Para tanto, inicialmente foram
59 realizadas entrevistas com o cliente do projeto, coordenador do setor de almoxarifado do
60 Instituto Federal de Educação, Ciência e Tecnologia da Bahia *campus* Vitória da Conquista.

61 Esta etapa se mostra uma das mais importantes, uma vez que a elucidação dos requisitos
62 é tida como uma das maiores dificuldades no desenvolvimento de um projeto. Estas



63 dificuldades foram particularmente potencializadas, uma vez que o ambiente em questão se
64 mostrou mais complexo do que o esperado, já que o setor se relaciona com outros setores da
65 instituição. O almoxarifado é responsável pelo recebimento de itens no instituto, pelo seu
66 armazenamento e liberação para uso. Assim sendo, deve-se relacionar com os setores
67 responsáveis por licitações, empenho de itens, bem como todos os demais setores e servidores
68 que podem receber tais itens.

69 A partir da realização das entrevistas, foi possível compreender o processo de aquisição
70 e liberação de itens. No entanto, inicialmente, o *software* em questão somente deveria atuar
71 no setor de almoxarifado, entendendo-se que os processos de aquisição de itens realizados
72 anteriormente, em outros setores, e que são a entrada para dados no sistema seriam realizados
73 como de costume. Por outro lado, deveria haver uma interface para solicitação de itens pelos
74 demais servidores do instituto. Na Figura 1 é demonstrado o fluxo principal de operações a
75 serem realizadas pelo ProGest.



76 **Figura 1.** Fluxo principal de utilização do ProGest. IFBA, 2016.

79 Modelagem

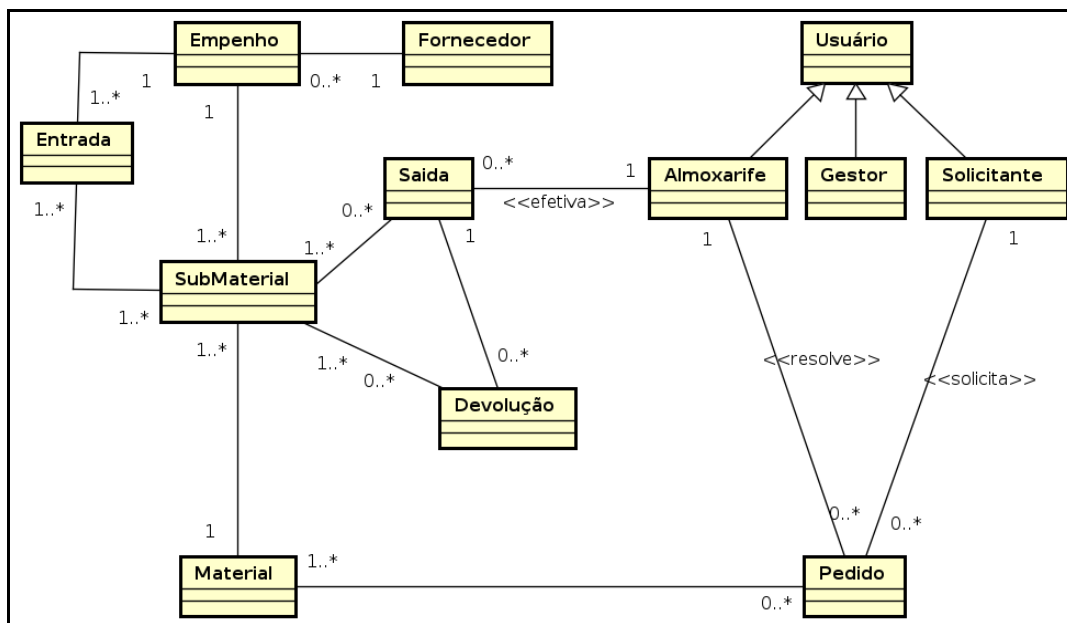
80 A modelagem de uma solução computacional objetiva gerar um projeto ou modelo de
81 como deverão ser desenvolvidas diversas características relativas ao funcionamento, estética e
82 utilização de componentes de um *software*, tendo como entrada as informações colhidas na
83 análise dos requisitos.

84 Especificamente, a modelagem de classes busca identificar as classes que deverão
85 compor o *software* e suas relações, as funções que cada uma deverá executar e os objetos que
86 serão manipulados. Na Figura 2 é possível visualizar um diagrama simplificado das classes
87 que compõem a aplicação ProGest, criado utilizando-se o *software* Astah Professional¹, em

¹ <http://astah.net/editions/professional>

88 sua versão 7.0.0, licenciada gratuitamente para uso estudantil. Neste diagrama, as informações
89 relativas aos atributos e métodos de cada classe foram suprimidas para permitir uma melhor
90 visualização das classes.

91 As classes visualizadas no diagrama da Figura 2 compreendem todos os objetos e
92 relações utilizados no ProGest. A classe Usuário e suas generalizações (Almoxarife, Gestor e
93 Solicitante) são utilizadas para a identificação do usuário no acesso ao sistema e para a
94 solicitação de itens, além de garantir a restrição de operações para o usuário de acordo com o
95 seu nível de acesso permitido. A classe Empenho é utilizada para o cadastro de empenhos, se
96 relacionando com as classes Entrada, SubMaterial e Fornecedor. A classe Material e sua
97 especialização SubMaterial são utilizadas para o cadastro de itens no sistema, bem como no
98 controle de entradas e saídas desses. A classe Fornecedor é utilizada para o cadastro dos
99 fornecedores. As demais classes representadas são utilizadas na realização de operações de
100 estoque, sendo elas Entrada, Saída, Devolução, responsáveis, respectivamente, pelo controle
101 de entrada, saída e devolução de itens ao estoque e Pedido, classe responsável pela realização
102 de pedidos pelos servidores do instituto.



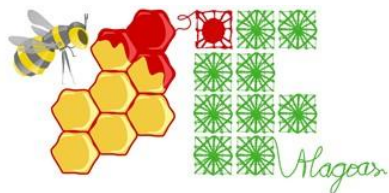
103

104

105

Figura 2. Diagrama de classes da aplicação ProGest. IFBA, 2016.

106 Segundo Pressman (2011, p. 313), o elemento mais importante de um sistema
107 computacional é a sua interface, uma vez que, sendo mal projetada, há um impacto na
108 capacidade do aproveitamento de todas as funcionalidades do *software* pelo usuário.



109 A partir do entendimento do problema em questão, procedeu-se o desenvolvimento do
110 projeto de interfaces. Esta etapa se mostra importante por permitir que o cliente e o usuário
111 final verifiquem, a partir da simulação do *software*, se esse contempla todas as necessidades
112 apontadas, bem como a facilidade e fluidez no seu uso, podendo sugerir correções ou
113 alterações.

114 Para a construção do projeto de interfaces, utilizou-se a ferramenta Axure², em sua
115 versão RP Pro 7.0, licenciada também gratuitamente para uso estudantil. A ferramenta em
116 questão permitiu o desenvolvimento das telas, bem como a criação do fluxo entre telas a partir
117 dos cliques em seus botões. Essa funcionalidade se mostrou importante por permitir que o
118 usuário final pudesse testar a fluidez e a facilidade na operação do *software* antes mesmo
119 deste ter seu desenvolvimento iniciado.

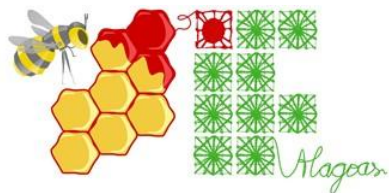
120 **Desenvolvimento**

121 A etapa de desenvolvimento contempla o processo de codificação da aplicação
122 proposta, a partir dos requisitos levantados e validados. Inicialmente, foram avaliadas as
123 opções para o desenvolvimento, desde a linguagem de programação e banco de dados a serem
124 utilizados, de acordo com a compatibilidade do servidor disponibilizado pelo instituto. Em
125 seguida, decidiu-se pela utilização de alguns *frameworks* para a agilização do processo de
126 desenvolvimento, visto o pouco tempo disponível para este fim.

127 A utilização de *frameworks* é uma prática comum no mercado de trabalho. Eles
128 permitem que, por meio do reuso de código e de outras funcionalidades, o tempo e esforço de
129 desenvolvimento seja reduzido. Ainda, pode-se perceber a padronização do desenvolvimento
130 quando este é guiado pelo uso de um *framework*, o que facilita o trabalho de manutenção do
131 *software*.

132 Para o desenvolvimento, optou-se pela utilização da linguagem PHP, entendendo-se que
133 a melhor abordagem para solucionar a demanda seria por meio de uma aplicação web. As
134 aplicações web têm como benefícios a sua capacidade de utilização por vários usuários, sua
135 alta disponibilidade e escalabilidade e o seu desempenho. Assim, todo o processamento dos
136 dados e serviços da aplicação devem ocorrer no servidor, retornando para o cliente páginas
137 web dinâmicas. O banco de dados escolhido para implementar a aplicação foi o MySQL, por

² Axure: <http://www.axure.com/>



138 ser compatível com a programação em PHP e com os *frameworks* utilizados e, assim como o
139 PHP, é suportado pelo servidor disponibilizado pelo instituto.

140 A justificativa pela escolha da linguagem PHP em conjunto com o banco de dados
141 MySQL é dada por alguns fatores apontados por Bento (2013): são tecnologias livres; há uma
142 facilidade e redução de custo de hospedagem de serviços que as utilizam; e pelo PHP ter uma
143 curva de aprendizado suave, em comparação com outras linguagens.

144 O principal *framework* utilizado no desenvolvimento foi o Laravel³. O Laravel permite
145 que os dados da aplicação sejam tratados como objetos, o que possibilita a programação em
146 alto nível, por meio da programação orientada a objetos. Outros fatores que permitiram a
147 escolha deste *framework* foram a sua capacidade de criação do banco de dados da aplicação a
148 partir dos objetos definidos, retirando dos programadores a responsabilidade de desenvolver
149 todo o projeto de banco de dados exigido para um projeto, sua completa compatibilidade com
150 as demais tecnologias utilizadas, sua documentação sucinta e sua comunidade de
151 desenvolvedores ativa e prestativa na resposta às dúvidas dos desenvolvedores.

152 Skvorc (2015) demonstra que, em uma pesquisa realizada com dezenas de países e com
153 cerca de 7.823 contribuidores, o Laravel foi escolhido como o melhor e mais popular
154 *framework* para desenvolvimento PHP. O Laravel se mostrou o *framework* mais utilizado em
155 20 países, dentre os 34 que tiveram um mínimo de 50 contribuidores, além de ter sido o
156 favorito em 4 das 5 faixas etárias definidas pela pesquisa, para projetos pessoais e comerciais.

157 Além do Laravel, foram utilizados os *frameworks* JQuery⁴, para validação de campos de
158 preenchimento do usuário, e Bootstrap⁵, para a estilização das páginas da aplicação. Sobre a
159 estilização, optou-se por utilizar um *template* gratuito de visualização de páginas em um
160 padrão de gerência, o que deu maior fluidez e padronização à aplicação, além de garantir a
161 sua responsividade. O *template* escolhido foi o AdminLTE⁶.

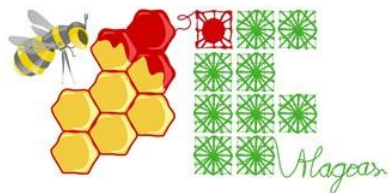
162 A responsividade é um conceito citado inicialmente por Marcotte (2010), quando sugere
163 um padrão de codificação único, que se adapte ao formato das diferentes telas de visualização
164 do usuário. Assim, não seria necessário o desenvolvimento de um código diferente para cada
165 tamanho ou resolução de tela.

³ Laravel: <https://laravel.com/>

⁴ JQuery: <https://jquery.com/>

⁵ Bootstrap: <http://getbootstrap.com/>

⁶ AdminLTE: <https://almsaeedstudio.com/preview>



166 **Implantação**

167 Segundo Souza e Zwicker (2000), a implementação de um sistema com características
168 com foco em uma situação problema específico “pode ser definida como o processo pelo qual
169 os módulos do sistema são colocados em funcionamento em uma empresa”. Esta etapa é de
170 fundamental importância, uma vez que a partir da utilização da solução em ambiente real,
171 podem-se verificar muitos pontos de melhoria no *software*, adequações e correções. Sendo
172 assim, nesta etapa, as diversas funcionalidades desenvolvidas foram colocadas para uso pelo
173 administrador do sistema e, posteriormente, para os demais usuários.

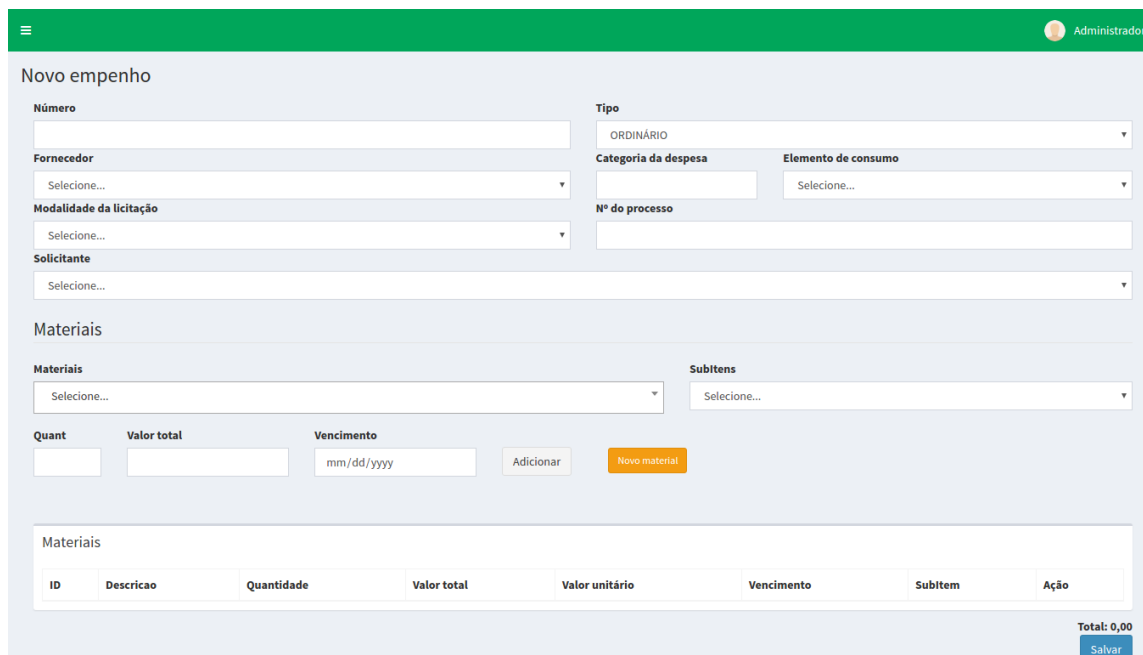
174 Inicialmente, o administrador utilizou o ProGest para verificar se todas as
175 funcionalidades previstas foram desenvolvidas e realizar testes de usabilidade. Após a
176 observação e recomendação de melhorias, procedeu-se a carga inicial de dados, quando foram
177 inseridos no banco de dados da aplicação os dados referentes aos produtos disponíveis no
178 almoxarifado. Para a realização desta carga de dados inicial, é preciso escolher entre uma
179 entrada genérica dos itens listados em estoque ou na simulação de várias entradas, o que
180 viabiliza a geração de relatórios de datas anteriores à implantação do *software*.

181 Após a carga de dados inicial e havendo produtos para requisição listados no ProGest, o
182 sistema poderia ser disponibilizado para acesso pelos usuários cadastrados, etapa realizada em
183 um terceiro momento, o que possibilitou a verificação de possíveis correções e melhorias.
184 Após a utilização por alguns usuários, esta funcionalidade deixou, momentaneamente, de
185 estar disponível para os mesmos, para que as correções e melhorias verificadas pudessem ser
186 desenvolvidas e, novamente disponibilizar a interface de solicitação de itens aos servidores.

187 188 **RESULTADOS E DISCUSSÃO**

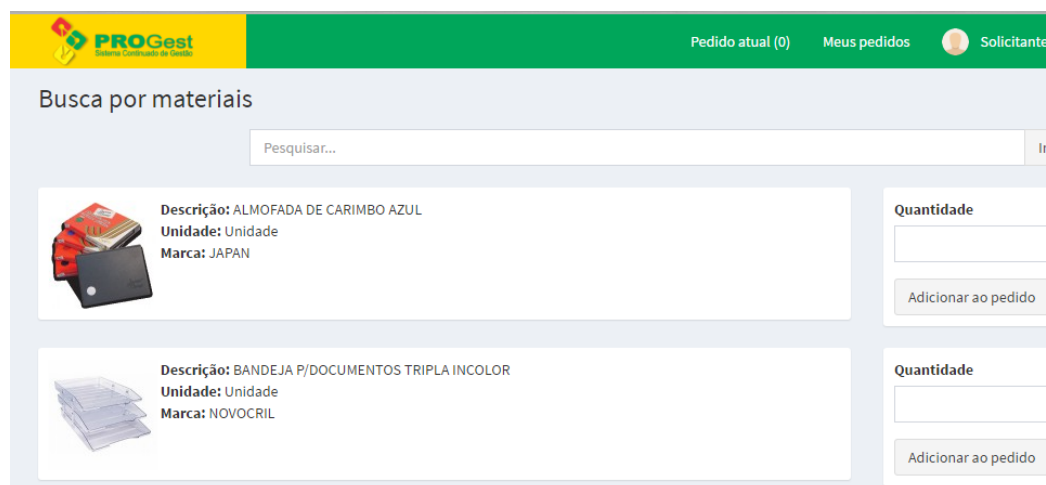
189 O *software* desenvolvido no instituto tem como propósito o gerenciamento de entradas
190 e saídas do almoxarifado. Assim, o seu fluxo de utilização inicia-se no cadastro de empenhos
191 (registro de produtos a serem recebidos no instituto), como visto na Figura 3, passando à
192 entrada destes produtos e registro de suas notas fiscais.

193 Além desta funcionalidade geral, foi desenvolvido outro módulo na aplicação
194 possibilitando o seu uso pelos demais servidores do instituto, para a realização de pedidos de
195 itens disponibilizados pelo administrador do sistema, de forma análoga às lojas virtuais, como
196 visto na Figura 4.



197
198
199

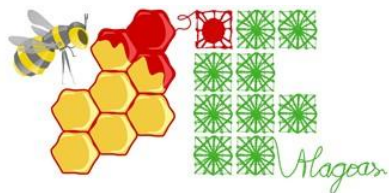
Figura 3. Tela de cadastro de empenhos - ProGest. IFBA, 2016.



200
201
202

Figura 4. Tela de pedidos de itens - ProGest. IFBA, 2016.

203 A partir da utilização do ProGest em seus dois módulos principais (de gerenciamento de
204 itens e de solicitações de usuários), diversos relatórios gerenciais podem ser gerados,
205 permitindo uma melhor visualização e controle de consumo de itens, seja por servidores,
206 setores ou eventos. Um exemplo de relatório gerado pode ser visualizado na Figura 5. Além
207 disso, há um maior controle sobre os produtos que deverão ser recebidos, bem como daqueles
208 que estão com seu vencimento próximo, permitindo um melhor gerenciamento do consumo.



209 O ProGest trata-se de um projeto *open source*, estando disponível sob a licença GNU
210 *General Public License*. O acesso ao seu código-fonte e projeto de interfaces pode ser feito
211 acessando o repositório⁷ online da aplicação. O estudo de caso descrito neste trabalho foi
212 realizado no Instituto Federal de Educação, Ciência e Tecnologia da Bahia – IFBA *campus*
213 Vitória da Conquista (<http://progest.conquista.ifba.edu.br/>), sendo iniciado em Novembro de
214 2015 e finalizado em Março de 2016. Atualmente, o desenvolvimento do ProGest foi
215 retomado, de forma que novas funcionalidades deverão ser implementadas.

Relatório - Entradas de NF's por período

Número empenho: Fornecedor: Data Inicial: Data Final: Ir

Entradas de Notas Fiscais - Período: 01/07/2016 a 19/09/2016

Nota Fiscal	Fornecedor	Empenho	Data de recebimento	Valor
988.721.938	Vazza LTDA	1231NE823123	16/08/2016	10,00
120.039.000	Vazza LTDA	0129NE838012	17/08/2016	300,03
019.098.910	Vazza LTDA	1203NE898100	17/08/2016	100,00
889.879.182	Vazza LTDA	9128NE837921	23/08/2016	230,10
123.213.123	Print Jet	1312NE831232	23/08/2016	300,00
120.937.198	Vazza LTDA	3312NE842352	29/08/2016	101,70
333.333.333	Vazza LTDA	3312NE842352	31/08/2016	203,40
456.576.879	Print Jet	4444NE844444	30/08/2016	100,00
123.123.123	Vazza LTDA	1231NE823123	01/09/2016	10,00
555.555.555	Print Jet	4444NE844444	01/09/2016	30,00
Total				1.385,23

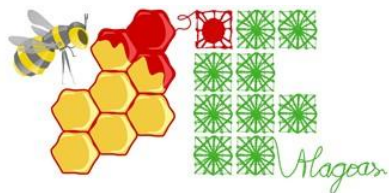
216 **Figura 5.** Relatório gerencial - ProGest. IFBA, 2016.

217 CONCLUSÕES

218
219 Analisando-se a proposta inicial de se desenvolver um *software* para o gerenciamento
220 de entradas e saídas do almoxarifado do instituto, pode-se dizer que o objetivo foi alcançado.
221 A solução desenvolvida conta com diferentes funcionalidades, o que permite o gerenciamento
222 de itens, de usuários, realização de entradas e saídas, realização de pedidos pelos servidores e
223 geração de múltiplos relatórios.

224 A partir do uso do ProGest, espera-se que haja um maior controle do consumo no
225 instituto, bem como uma melhor organização de itens em estoque. É importante citar que o

⁷ Repositório ProGest: <https://github.com/loppedan/progest>



227 sistema em questão deverá ser aprimorado e, para tanto, um estagiário foi disponibilizado
228 pelo instituto para continuar o desenvolvimento.

229 Como trabalhos futuros, pretende-se continuar o desenvolvimento da aplicação, de
230 forma que esta contemple todos os detalhes do seu ambiente de atuação. Após essa etapa,
231 pode-se pensar em estender a aplicação para outras áreas ligadas ao setor de almoxarifado,
232 como o registro e movimentação de bens duráveis ou o setor financeiro.

233 Ao passo em que novas funcionalidades ou melhorias forem implementadas, espera-se
234 que o ProGest entre em plena utilização por todos os servidores do instituto, de modo que
235 possa ser verificada sua completa adequação para o ambiente e destinação a que se propõe.

236

237 **AGRADECIMENTOS**

238 Agradecemos ao Instituto Federal de Educação, Ciência e Tecnologia da Bahia *campus*
239 Vitória da Conquista e sua Coordenação de Pesquisa e Extensão (COPEX) pelo custeio de
240 bolsas para a realização do projeto, por meio do edital COPEX IFBA/VC N° 01/2015.

241

242 **REFERÊNCIAS**

243 BENTO, E. J. **Desenvolvimento web com PHP e MySQL**. Casa do Código, 2013.

244

245 CANZIAN, C. *et. al.* Sistema de requisição eletrônica de compras – apresentação e proposta
246 de melhorias para os setores de suprimentos e almoxarifado da FOP – UNICAMP. **Revista**
247 **Eletrônica do SIMTEC**, Campinas - SP, n. 3, p. 53-53, 2010.

248

249 MARCOTTE, E. **Responsive Web Design**. A list apart. 2010. Disponível em:
250 <<http://alistapart.com/article/responsive-web-design>>. Acesso em: 23 set. 2016.

251

252 PASCHOAL, M. L. H.; CASTILHO, V. Implementação do sistema de gestão de materiais
253 informatizado do Hospital Universitário da Universidade de São Paulo. **Revista da Escola de**
254 **Enfermagem da USP**, v.44, n.4, p.984-988, 2010.

255

256 PRESSMAN, R. S. **Engenharia de Software: uma abordagem profissional**. 7. ed. Porto
257 Alegre: AMGH, 2011.

258

259 SKVORC, B. **The Best PHP Framework for 2015: SitePoint Survey Results**. 2015.
260 Disponível em: <[https://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-](https://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/)
261 [results/](https://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/)>. Acesso em: 27 set. 2016.

262

263 SOUZA, C. A.; ZWICKER, R. **Ciclo de vida de sistemas ERP**. Caderno de Pesquisas em
264 Administração, São Paulo, v. 1, n. 11, p. 46-57, 2000.