

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA
CELSO SUCKOW DA FONSECA – CEFET/RJ**

Sistema de Gestão de Escala de Trabalho

Maria Camila Leite Pereira
Priscila Patrício de Lima

Prof. Orientador: Fábio Paschoal Júnior, D.Sc.

**Rio de Janeiro
Agosto de 2017**

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA
CELSO SUCKOW DA FONSECA – CEFET/RJ**

Sistema de Gestão de Escala de Trabalho

Maria Camila Leite Pereira
Priscila Patrício de Lima

Projeto Final apresentado em cumprimento
às normas do Departamento de Educação
Superior do Campus Maracanã de CEFET/RJ,
como parte dos requisitos para obtenção do
título de Tecnólogo de Sistemas para Internet

Prof. Orientador: Fábio Paschoal Júnior, D.Sc.

**Rio de Janeiro
Agosto de 2017**

Ficha catalográfica

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

P436 Pereira, Maria Camila Leite
Sistema de gestão de escala de trabalho / Maria Camila Leite
Pereira, Priscila Patrício de Lima.—2017.
xi, 43f. : il.color. , grafs. , tabs. ; enc.

Projeto Final (Tecnólogo) Centro Federal de Educação
Tecnológica Celso Suckow da Fonseca , 2017.
Bibliografia : f. 42-43
Orientador : Fábio Paschoal Júnior

1. Internet – Programas. 2. Sistemas de suporte de decisão. 3.
Horário de trabalho flexível. I. Lima, Priscila Patrício de II. Paschoal
Júnior, Fábio (Orient.). III. Título.

CDD 004.678

Agradecimentos

Agradecemos a todos os professores do CEFET-RJ pelos conhecimentos passados, pela dedicação a cada aula e pela paciência a cada erro nosso, sem cada um esse projeto não seria possível. Agradecemos os nossos familiares, em especial a mãe da Maria Camila, Cristina Leite, que sempre lhe incentivou e, mesmo quando parecia impossível, não lhe deixou desistir e lhe fez seguir em frente até o fim. Agradecemos aos amigos que contribuíram de alguma forma nessa caminhada, cada incentivo ou conhecimento compartilhado foi de extrema importância. Agradecemos ao nosso orientador Fábio Paschoal Júnior, que sempre esteve à disposição nos apontando os caminhos a seguir e fez desse projeto uma realidade.

Resumo

Esse projeto tem como principal finalidade a elaboração de escalas de trabalho de forma mais ágil e dinâmica, sem que haja outras funcionalidades acopladas, encurtando o tempo de preparação da mesma e diminuindo o trabalho ao fazê-las, como ocorria ao serem feitas manualmente. O objetivo foi alcançado com a elaboração de um sistema que gera, a partir de dados fornecidos por um gestor, a escala com feriados, turnos e individual para cada funcionário, exibindo-as *on-line*. O *software* foi idealizado para ser um sistema de apoio a decisão, onde um administrador cadastra o funcionário, a equipe a qual o funcionário pertence, seu turno e para qual mês deseja, gerando, assim, a escala.

Palavras-chave: Escala de Trabalho, Revezamento, Apoio a Decisão.

Abstract

This Project has as main purpose the elaboration of work scales in a more agile and dynamic way, without having other functionalities coupled, shortening the preparation time and reducing the work in doing them, as it happened when done manually. The goal was achieved with the elaboration of a system that generates, from data provided by a manager, the scale with holidays, shifts and individual for each employee, displaying them on-line. Software designed to be a decision support system, where an administrator registers employee, team to which the employee belongs, his shift and for which month he wishes, thus generating the scale.

Keywords: work scales, Rotation, decision support system.

SUMÁRIO

1	Introdução	1
2	Fundamentação Teórica.....	4
2.1	Escala de Horário.....	4
2.2	Sistemas de Apoio à Decisão.....	4
2.3	Programação Linear.....	5
2.4	PHP: <i>Hypertext Preprocessor</i>	6
2.5	<i>Framework</i> Laravel	6
2.6	MySQL.....	7
2.7	Trabalhos Relacionados.....	7
2.7.1	Escala Fácil.....	7
2.7.2	SDMPE.....	9
2.7.3	PontoVit.....	10
2.7.4	Revex	11
2.8	SGET	11
3	Especificação do Sistema	13
3.1	Mini-mundo	13
3.2	Regras de Negócio.....	14
3.3	Modelo de Caso de Uso.....	15
3.3.1	Diagrama de Caso de Uso	15
3.3.2	Descrições Textuais dos Casos de Uso.....	16
3.4	Modelo de Classes	21
3.4.1	Diagrama de Classe	21
3.5	Comparativo	23
3.6	Banco de Dados.....	23
3.7	Interface	26

4 Avaliação Experimental	30
4.1 SUS – <i>System Usability Scale</i>	30
4.2 Resultados.....	33
5 Conclusão	39
REFERÊNCIAS	42

LISTA DE FIGURAS

Figura 1 - Exemplo do <i>Software</i> Escala Fácil	8
Figura 2 - Exemplo do <i>Software</i> SDMPE	9
Figura 3 - Exemplo do <i>Software</i> PontoVit	10
Figura 4 - Comparativo entre os <i>Softwares</i>	12
Figura 5 - Diagrama de Casos de Uso	15
Figura 6 - Diagrama de Classes.....	22
Figura 7 - Modelo Lógico do Banco de Dados	25
Figura 8 - Interface da Pagina Inicial	27
Figura 9 - Interface Lista de Feriados.....	27
Figura 10 - Interface de Cadastro de Novo Usuário.....	28
Figura 11 - Interface de Motivos	28
Figura 12 - Interface de Lista de Usuários	29
Figura 13 - Lista de escalas já criadas por mês	29
Figura 14 - Interface da Escala Pronta	29
Figura 15 - Exemplo da Implementação do SUS	32
Figura 16 - Resultado em gráfico da Avaliação Experimental	34
Figura 17 - Resultado em gráfico da Avaliação Experimental	35
Figura 18 - Resultado em gráfico da Avaliação Experimental	36
Figura 19 - Resultado em gráfico da Avaliação Experimental	37
Figura 20 - Resultado da Avaliação Experimental.....	38

LISTA DE TABELAS

Tabela 1 - Regras de Negócio.....	14
Tabela 2 - Comparativo de Tempo	23

LISTA DE ABREVIATURAS E SIGLAS

BH	Banco de Horas
CLT	Consolidação das Leis Trabalhistas
ERP	<i>Enterprise Resource Planning</i>
F	Folga
FE	Férias
HE	Hora Extra
HTML	<i>HyperText Markup Language</i>
LM	Licença Médica
MCU	Modelo de Caso de Uso
MVC	<i>Model View Controller</i>
PHP	<i>PHP: Hypertext Preprocessor</i>
PO	Pesquisa Operacional
PRL	Programação Linear
SAD	Sistema de Apoio à Decisão
SDMPE	Sistema de Distribuição Mensal de Pessoal de Enfermagem
SGET	Sistema de Gestão de Escala de Trabalho
SQL	<i>Structured English Query Language</i> – Linguagem de Consulta Estruturada
SREP	<i>Supervisory Review and Evaluation Process</i>
T	Trabalho
UML	Linguagem de Modelagem Unificada
URL	<i>Uniform Resource Locator</i>

1 Introdução

De acordo com a ANATEL, todos os usuários de serviços de telecomunicações, serviços esses considerados essenciais, têm o direito de acesso com padrões de qualidade e regularidade, adequados à sua natureza (ANATEL, 1997. Art.3). Para que isso seja possível, as operadoras de telefonia dispõem de equipes de operação e de manutenção, que trabalham ininterruptamente, visando assegurar a confiabilidade e disponibilidade dos serviços oferecidos.

Empresas que fornecem serviços contínuos precisam ter a garantia que a qualquer momento haja recursos disponíveis para atender as demandas dos seus clientes. Esta disponibilidade de recursos normalmente é do tipo 24/7, ou seja, 24 horas por dia e 7 dias na semana. Assim, as empresas são obrigadas a organizar seus colaboradores em equipes com turno de trabalho bem definido, prevalecendo o revezamento entre os empregados que exercem a mesma função, independente do horário de trabalho na escala (CLT, 1943. Art.230).

A escala de trabalho normalmente é elaborada por uma pessoa que conhece a dinâmica da equipe e as leis trabalhistas vigentes, pois a tomada de decisão sobre as folgas dos escalados depende de regras preestabelecidas a partir dos direitos dos trabalhadores e da necessidade da empresa. É comum que as empresas montem suas escalas de trabalho de forma manual, através de planilhas, o que requer dedicação e tempo para alcançar um resultado que garanta um mínimo de disponibilidade de pessoal trabalhando em cada turno com a maior produtividade. Porém, quando se trata de uma empresa de médio ou grande porte o problema se torna muito mais complexo, pois o número de colaboradores em escala pode aumentar.

Quanto maior o número de colaboradores mais complexa se torna a elaboração de uma escala e maior o tempo dedicado para isso, pois aumenta-se também a ocorrência de férias, licenças, atestados e faltas, sendo que para essas duas últimas ocorrências não há como prever o dia que elas ocorrerá. Por causa disso, a pessoa responsável pela elaboração deve ter sempre uma margem de segurança para garantir o mínimo necessário de colaboradores por turno de trabalho, sem privar os funcionários de seus direitos previstos na CLT.

Nesse contexto, busca-se desenvolver um Sistema de Apoio a Decisão (SAD) capaz de auxiliar a elaboração dessas escalas de revezamento para equipes organizadas em um regime 24x7 para otimização do tempo gasto com essa atividade. Porém, identificar as necessidades do usuário e traduzi-las em linguagens computacionais requer sensibilidade, empenho e experiência. Isso torna a interação entre usuários e analista uma das maiores dificuldades no

desenvolvimento de *software*.

No desenvolvimento de um SAD a dificuldade aumenta, uma vez que, conforme BARBOSA, ALMEIDA e COSTA(2006):

- O usuário pode incluir fatores e analisar cenários variados, se tornando raro comportamentos padrões entre o usuário e o aplicativo.
- A frequência com que se repete uma decisão é baixa.
- Geralmente, tratam de decisões não-estruturadas sobre as quais vai-se adquirindo conhecimento aos poucos. Isso permite que um aplicativo esteja em constante evolução.
- Necessitam de dados informais, como por exemplo, intuição, experiências e heurística em adição à lógica.

Após acompanhamento da composição e manutenção manual de algumas escalas de trabalho, observou-se que algumas empresas possuem pessoas alocadas exclusivamente para tais atividades devido a sua complexidade. O processo de elaboração pode durar alguns dias, pois possui algumas etapas:

1. Levantamento das preferências de folgas;
2. Contagem da quantidade de finais de semana e feriados no mês;
3. Cálculo da quantidade de folgas que o colaborador terá direito no mês;
4. Verificação da existência de Férias agendadas;
5. Levantamento da quantidade de pessoas necessárias por turno/dia;
6. Cálculo.

Sendo essas horas distribuídas nas seguintes etapas: consolidação das solicitações de folgas, organização dos funcionários por turno de trabalho, cálculo da quantidade de folgas que serão permitidas (baseada na quantidade de sábados e domingos no mês), definição do número de pessoas que serão necessárias por dia e por horário, distribuição das folgas, verificação da aplicação das regras da Consolidação das Leis Trabalhistas (CLT), verificação se as solicitações foram atendidas, validação geral para garantir que todas as regras e critérios foram atendidos, bem como a divulgação da escala finalizada.

Ao longo do mês, já pensando no mês subsequente, são dedicadas algumas horas do dia para que se possa coletar os dados de cada turma (dos que se encontram de férias, dos possíveis feriados, mínimo de pessoal por dia exigido por cada turno e setor, e etc.), seguindo para a elaboração da escala, respeitando os dados já coletados, depois é feita uma revisão na

escala já pronta, para se ter certeza que nenhuma regra foi esquecida e, então, é finalizada a escala. Com todo esse processo, somando-se as horas gastas por dia, estima-se que são necessárias 20 horas mensais apenas para a elaboração em seu todo.

Em uma empresa, é indispensável se buscar a eficácia com eficiência. Partindo desse princípio, não é aceitável que se aloque recursos apenas para dedicar-se a elaboração de escalas de trabalho. Além disso, o nível de falhas é elevado, podendo se dizer que é proporcional à quantidade de pessoas envolvidas nas escalas.

Estima-se que todo esse processo tenha uma redução no tempo dedicado com a utilização do Sistema de Gestão de Escala de Trabalho (SGET) para até 15 minutos, uma vez que o *software* que foi desenvolvido por este trabalho decidirá se a solicitação poderá ser atendida sem que haja impacto para as regras e critérios predefinidos pelo responsável pela equipe. A este caberá a confirmação dos dados dos funcionários e a manutenção dessas regras, quando for necessário, também caberá as alterações futuras na escala pronta se houver necessidade de troca de horário/folga entre funcionários.

A solução do problema da elaboração de uma escala de trabalho recai em um problema de otimização. A otimização é a área da Pesquisa Operacional que utiliza de métodos científicos para apoiar a tomada de decisões, procurando determinar como melhor projetar e operar um sistema, usualmente sob condições que requerem a alocação de recursos escassos. Dentre as técnicas de otimização existentes, a melhor que se aplicou ao problema abordado por este trabalho foi a Programação Linear, que é explicada mais adiante.

Este trabalho está organizado em 5 capítulos. Este primeiro capítulo apresentou a introdução do trabalho, indicando a motivação, o contexto e os objetivos a serem alcançados no desenvolvimento deste trabalho.

O Capítulo 2 apresenta a fundamentação teórica, destacando os conceitos e os trabalhos relacionados a este trabalho.

O Capítulo 3 apresenta o desenvolvimento da solução realizada por este trabalho, incluindo a modelagem do sistema, além de um comparativo de tempo entre escalas sendo feitas manualmente e na ferramenta SGET.

O Capítulo 4 apresenta a avaliação experimental deste trabalho e o Capítulo 5 apresenta a conclusão desse trabalho, incluindo a sugestão de trabalhos futuros.

2 Fundamentação Teórica

Este capítulo descreve melhor as principais tecnologias e conceitos computacionais que deram base para o desenvolvimento do trabalho, incluindo alguns problemas em aberto, além dos trabalhos relacionados. Para tanto, este capítulo foi dividido em seções que tratam os SAD, a Programação Linear (PRL) e os trabalhos relacionados ao sistema desenvolvido.

2.1 Escala de Horário

Para uma empresa trabalhar no regime 24x7 (que consiste em 24 horas do dia e 7 dias da semana) é necessário que os funcionários sigam uma escala de trabalho que organiza os dias, horários e folgas. A cada turno é alocado uma equipe, cada equipe tem seu horário de entrada e saída predefinidos. Além disso, cada membro dessa equipe dispõe de um dia na semana, a qual lhe será alocado, para folgar.

Sendo assim, escala de horário é a forma de organizar toda a equipe em diferentes turnos, cada turno com funcionários folgando em diferentes dias, permitindo que uma empresa disponha sempre de um mínimo de funcionários trabalhando.

2.2 Sistemas de Apoio à Decisão

A informação vem auxiliando as organizações a alcançarem seus objetivos num ambiente de crescente concorrência, exigindo delas rapidez, flexibilidade e alta percepção. A abrangência cada vez maior do uso dos SAD como suporte na solução de problemas tem como proposta o processamento de informação para reduzir a incerteza e resolver conflitos (CLERICUZI, ALMEIDA e COSTA, 2006).

Segundo Reis e Löbler (2012), esses sistemas são instrumentos poderosos de incorporação de métodos científicos no auxílio de decisões complexas, com técnicas desenvolvidas na ciência da informação. Para eles, os SAD não substituem os seres humanos, mas aumentam a sua capacidade de lidar com problemas complexos. A interface do usuário determina se um SAD será utilizado na sua totalidade e se a qualidade final das decisões será maior do que quando o usuário não utiliza um SAD.

Pesquisas têm focado sobre a forma como as decisões são tomadas e o modo por que, teoricamente, deveriam ser tomadas; como consequência, o alcance e a diversidade da teoria têm sido vastas (DILLON, 1998). A partir disso, surgiu uma matriz de modelos de tomada de

decisões. Dependendo de sua base metodológica, esses modelos podem ser classificados como descritivos, prescritivos ou normativos.

A teoria da decisão Normativa ou Prescritiva é uma teoria sobre como as decisões devem ser tomadas. Ela orienta o decisor como agir para decidir em certas condições envolvendo escolha de ações ou alternativas. Já a teoria Descritiva foca no comportamento do decisor, é uma teoria sobre como decisões são realmente feitas (HANSSON, 1994). Propõe-se a descrever como um decisor comporta-se ou, durante o tempo, como irá se comportar em situações que, supostamente poderão ser descritas de maneira suficientemente precisas (REIS, LOBLER, 2012).

O sistema descritivo aborda o processo decisório como ele é, já o normativo, aborda o processo decisório como ele deveria ser. Já a análise prescritiva explica algumas das consequências lógicas da teoria normativa e os resultados empíricos dos estudos descritivos, de uma maneira geral (REIS, LOBLER, 2012).

2.3 Programação Linear

Frequentemente as empresas se deparam com situações onde existe a necessidade de uma escolha entre diversas situações. Cada gestor utiliza-se de um modelo de decisão, para uma escolha objetiva entre diversas linhas de ação. Desta forma, considerando os avanços tecnológicos, e a crescente competitividade surge então, a necessidade de tomada de decisão baseada em processos cada vez mais seguros e que forneça dados mais precisos (CORRAR, GARCIA, 2001).

A PRL é umas das técnicas utilizadas na Pesquisa Operacional (PO), que consiste num método matemático utilizado no processo de tomada de decisão. A PO surgiu na Segunda Guerra Mundial, com o objetivo de resolver problemas relacionados às operações estratégicas e táticas militares (CORRAR, GARCIA, 2001). Mas vem sendo aplicado nas análises empresariais, principalmente, com referência a melhor utilização dos recursos escassos, uma vez que oferece técnicas de resolução para uma vasta quantidade de problemas (SCHMIDT *et al.*, 2008).

Conforme Schmidt *et al.* (2008), os problemas de Programação Linear são caracterizados por equações e inequações lineares desconhecidas, representando o objetivo e as limitações do problema em estudo. Ainda segundo Schmidt *et al.* (2008)

“Para se chegar à solução de um problema pelo modelo de programação linear, é necessário considerar um modelo genérico que contempla, de forma geral:

- As variáveis de decisão cujo gestor tem poder de alterá-las a qualquer tempo;
- Os parâmetros, que são variáveis utilizadas no modelo, as quais o tomador de decisões não tem poder de alterá-las, já que não estão sob discussão;
- A função-objetivo, que define e mensura o principal objetivo;
- As restrições que combinam variáveis e parâmetros para estabelecer regras, relações e limites do modelo;
- Um modelo matemático que contemple parâmetros, variáveis de decisão, função-objetivo e restrições, e o qual representa o problema real em análise utilizando somente funções lineares.”

2.4 PHP: *Hypertext Preprocessor*

Esta seção apresenta um breve resumo sobre a tecnologia PHP ("PHP: *Hypertext Preprocessor*", originalmente *Personal Home Page*). Trata-se de uma linguagem criada por Rasmus Lerdorf, surgida por volta de 1994 (LOPES, 2007). É uma linguagem de código aberto, principalmente utilizada, mas não exclusivamente, para o desenvolvimento *web*. Podendo ser embutida dentro do *HyperText Markup Language* (HTML). Seu principal propósito é a implementação de soluções *web* que sejam eficientes para o desenvolvedor, ao mesmo tempo simples e veloz.

A linguagem PHP possibilita a interação com o usuário por meios de parâmetros da *Uniform Resource Locator* (URL), *links* e formulários. O código PHP é enviado e executado no servidor, e depois enviado para o usuário em forma de HTML.

2.5 *Framework Laravel*

Esta seção apresenta um resumo sobre o *Framework* Laravel. Trata-se de um *Framework* PHP usado em desenvolvimentos na plataforma *web*. Laravel utiliza a arquitetura *Model - View - Controller* (MVC).

A arquitetura MVC divide o desenvolvimento da aplicação em três camadas, sendo elas Modelo, Visão e Controlador. Na camada Modelo estão presentes as consultas ao banco de dados, as classes e as regras de negócio. Na camada Visão, como o nome já diz, é onde se localiza a interface, é a tela do sistema. A camada Controlador é responsável pelo controle do sistema. Nela são feitas as seguintes atividades: 1.) a validação dos dados na entrada de dados da *view*; 2.) a consulta a base de dados da camada Modelo; 3) a ligação dos dados com as

regras de negócio; 4.) o processamento de todas as requisições e; 5.) a geração de um resultado, com a criação de uma nova *view* (SATO, 2017).

O principal objetivo do Laravel é ajudar no desenvolvimento de aplicações seguras, que possuam códigos limpos e simples. Para a interface gráfica o Laravel trabalha com *template* chamado Blade. Para a comunicação com o banco de dados é usado uma implementação simples denominada Eloquent ORM, que possui diversas funcionalidades para a inserção, atualização, busca e exclusão (ADRIEL, 2017).

2.6 MySQL

Esta seção apresenta um resumo sobre o MySQL. O MySQL foi criado na Suécia por David Axmark, Allan Larsson e o finlandês Michael Widenius, no ano de 1980 (RICARDO, 2017). É um sistema de gerenciamento de banco de dados relacional de código aberto que está presente na maior parte das aplicações gratuitas que gerenciam bases de dados. *Structure Query Language* – Linguagem de Consulta Estruturada (SQL) é a linguagem utilizada como interface, essa linguagem é a mais usada em bancos de dados para a inserção, acesso e gerenciamento dos dados.

É um sistema de gerenciamento de dados confiável que fornece os mais variados recursos, um bom desempenho e uma diversidade para todos os tipos de clientes, atendendo a aplicativos *web*, ou armazenamento de dados locais ou até mesmo remotamente.

2.7 Trabalhos Relacionados

Há no mercado *softwares* com a finalidade de elaborar escalas de revezamento para várias áreas distintas. Na busca por sistemas que preencham essas necessidades, ressaltam-se alguns bons exemplos, tais como o Escala Fácil, o Sistema de Distribuição Mensal de Pessoal de Enfermagem (SDMPE), o PontoVit e o Revex.

2.7.1 Escala Fácil

O Escala Fácil foi desenvolvido atrelado a funcionalidades capazes de realizar controle de ponto eletrônico, de férias e de consumo de vale-transporte de cada funcionário. A

partir do armazenamento das informações dos funcionários, a geração da escala nos meses subsequentes ocorre, automaticamente, a partir do último dia de registro. Porém essa distribuição não é aleatória nem considera a quantidade mínima de funcionários necessários por dia, assim, ao término do ciclo daquela escala os dias de folgas voltam a se repetir (ESCALA FÁCIL, 2014).

Esse sistema, porém, vem acoplado a um mecanismo de ponto e vale transporte. A princípio estas funcionalidades serão vistas como vantagens. Contudo, quando se pensa em empresas de menor porte, ou empresas que não trabalham com batida de ponto eletrônica ou, simplesmente, empresas que apenas necessitam de um gerador de escalas, essas funções se tornam desvantagens e o sistema preterido.

A Figura 1 apresenta um exemplo de escala no *software* Escala Fácil. O gráfico a seguir apresenta uma estimativa, hora a hora, da quantidade de funcionários que a empresa precisa ter para manter o bom funcionamento. A primeira imagem representa a quantidade de funcionários por hora necessários de segunda a sábado, a segunda representa o domingo, onde o número de funcionários é reduzido, principalmente em função das folgas.

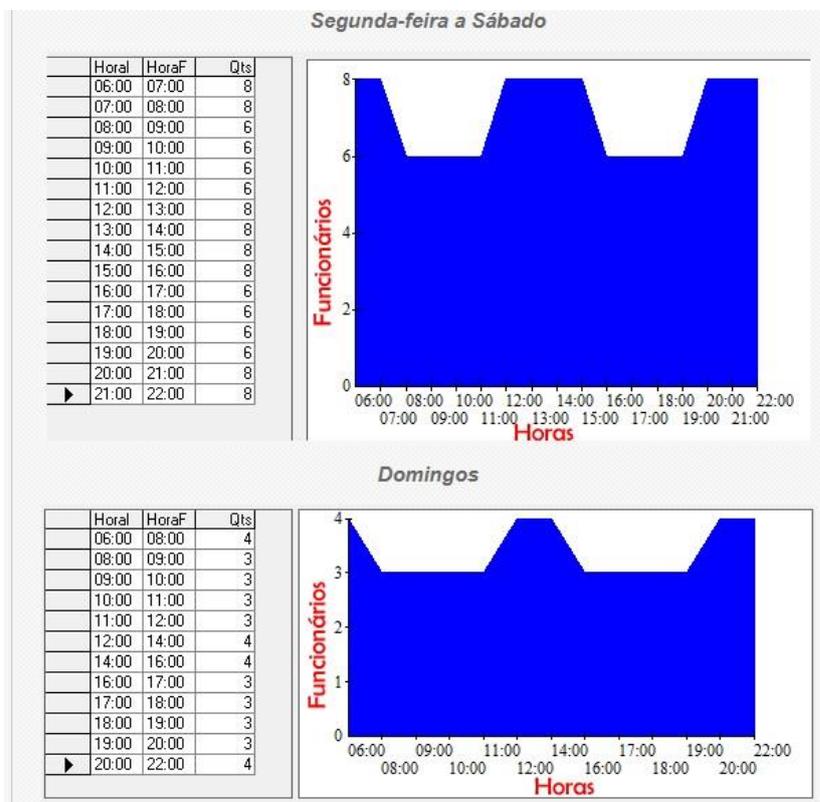


Figura 1 - Exemplo do *Software* Escala Fácil

2.7.2 SDMPE

O SDMPE foi baseado nos conceitos de um SAD e realiza a elaboração de escala periódica de trabalho a partir do armazenamento das informações trabalhistas de todos os funcionários de enfermagem de uma instituição. Nele, a escala de trabalho somente poderá ser gerada por local de trabalho e para cada um dos cargos existentes, ou seja, as escalas de enfermeiros e auxiliares de enfermagem devem ser separadas (RANGEL, ÉVORA, 2007). Tal sistema não considera o mínimo necessário diário de cada profissional, nem considera a possibilidade de várias folgas consecutivas.

Este é um sistema restrito a área de enfermagem. Sua restrição no uso se torna sua maior desvantagem, além das escassas funcionalidades oferecidas. É um sistema simples, apenas para hospitais, porém bem eficaz no que se propõe a fazer.

A Figura 2 apresenta um exemplo de escala no *software* SDMPE. Na imagem é possível ver uma escala já pronta do mês de janeiro de uma determinada função, no caso do exemplo é pediatria, com todos os funcionários já listados com suas respectivas folgas, dias de trabalho e turno.

Escala de Trabalho de Profissionais de Enfermagem - Mozilla Firefox

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

Http://127.0.0.1/enfermagem/principal.html

Cadastros Funcionários Escalas Logout

Domingo, 19 de novembro de 2006.

Exporta a Escala para o LINDO/LINGO

Período da Escala: JANEIRO/2006 Cargo: ENFERMEIRA Local de Trabalho: PEDIATRIA 7A

Opções: Pesquisador Exportar Importar Sair

Cód.	Funcionário	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	C.H.P.	C.H.T.
1	FLNC 01	F	F	M	F	T	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	132,00	136,00
2	FLNC 02	F	T	M	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	132,00	144,00
3	FLNC 03	M	N	F	F	M	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	132,00	150,00	
4	FLNC 04	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	132,00	144,00	
5	FLNC 05	M	T	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	132,00	144,00
6	FLNC 06	T	M	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	132,00	132,00	
7	FLNC 07	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	F	F	T	M	N	132,00	138,00	
8	FLNC 08	N	F	F	N	F	F	N	F	F	N	F	F	N	F	F	N	F	F	N	F	F	N	F	F	N	F	F	N	F	F	132,00	132,00	

Usuário: Alexandre Rangel

javascript:void(0);

Figura 2 - Exemplo do *Software* SDMPE

2.7.3 PontoVit

O módulo Escalas de Trabalho do Sistema PontoVit foi criado para eliminar definitivamente problemas com a elaboração e a gestão de escalas de todos os tipos. Uma ferramenta prática, 100% adequada a CLT que facilita a montagem das Escalas garantindo total conformidade na organização dos horários e das jornadas, tudo via *web* em alguns minutos (PONTOVIT, 2017).

Entre as vantagens que esse sistema oferece tem-se a geração de um documento logo depois de finalizado o processo de montagem da escala, conformidades com as regras da CLT, sistema via *web*, o que possibilita os funcionários, de qualquer lugar que haja rede de Internet, consultar sua escala, controle das horas trabalhadas de cada funcionário, graças ao sistema de ponto integrado.

A Figura 3 apresenta um exemplo de escala no *software* PontoVit. Na imagem é representada uma escala em preparação, especificando a empresa de atuação, o tipo de escala e o período em que a escala é válida. Nela, o gestor tem seus funcionários listados para designar, suas férias, licença e, caso haja, banco de horas extras.

The screenshot displays the 'Gerenciar Escalas' interface. At the top, there are navigation tabs: Escalas, Administração, Configurações, and Relatórios. Below this, the current view is 'Montagem' (Setup), with sub-tabs for Análise, Autorização, and Impressão. The main content area shows the company 'Empresa Teste' and the production unit 'Produção 6x1' for the period '01/09/2016 a 30/09/2016'. A legend at the bottom identifies interruption types: Banco de Horas (B), DSR (D), Férias (X), and Licença (L). A 'Turnos' dropdown menu is also visible.

Dias semana	Q	S	S	D	S	T	Q	Q	S	S	D	S	T	Q	Q	S	S	D	S	T	Q	Q	S	S	D	S	T	Q	Q	S	
Funcionários / Dias	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
Jor_5 - 07:30 até 17:30 - 120 minutos																															
Colaborador 1 - (Operador)				D			D				D			D				D			D									D	
Colaborador 2 - (Operador)	D							D			D			D				D			D									D	
Colaborador 3 - (Operador)		D							D					D				D			D										D
Colaborador 4 - (Operador)			D							D								D							X	X	X	X	X	X	X
Colaborador 5 - (Operador)	X	X	X	X	X	X	X	X	X	X	X	X	X						D												
Colaborador 6 - (Operador)						D					D		D					D			D									D	
Colaborador 7 - (Operador)							D							D				D			D										D

Figura 3 - Exemplo do *Software* PontoVit

2.7.4 Revex

O Revex constrói uma escala de trabalho em 5 minutos, permite o gerenciamento total das escalas e ainda auxilia na análise de desempenho (Professor Dorirley Rodrigo Alves, sobre Revex, 2012). Atendem as regras trabalhistas, permite gerar, gerenciar e visualizar as escalas de folga rapidamente, integração com *softwares* complementares (*Enterprise Resource Planning* (ERP), *Supervisory Review and Evaluation Process* (SREP) e etc.), o gerenciamento permite operar todas as questões ligadas às ausências dos colaboradores, possui vários níveis de acesso (Administrador, Gerente e Usuários simples) e o colaborador acessa sua escala de folga a qualquer momento por meio da *web* e celular (REVEX, 2012).

2.8 SGET

Tendo em vista os *softwares* citados, a falta de automação na distribuição das folgas e a existência de funções adicionais, tais como o controle de frequência e controle de vale-transporte, são fatores limitantes. A aquisição de ferramentas com essas características se tornam inviáveis para empresas onde apenas uma minoria trabalha em regime de escala de revezamento, como por exemplo, empresas de telefonia.

Nesse contexto, idealizou-se o SGET, *software* independente de outros sistemas, que tem como diferencial a capacidade de distribuição automática de folgas de acordo com a necessidade do cliente, sem detrimento às leis da CLT. Além disso, o SGET propõe a possibilidade de gestão das alterações de folgas, o armazenamento de escalas em histórico e o controle de perfis de usuários, importante para definição do controle de visualização *on-line* das escalas disponíveis no sistema.

A Figura 4 resume algumas funções comparativas entre o SGET e os trabalhos relacionados já relatados.

Softwares Funções	SGET	Escala Fácil	SDMPE	PontoVit	Revex
Formato 24/7	✓	✗	✓	✗	✗
Sistema de Ponto Obrigatório	✗	✓	✗	✓	✗
Exclusivo para Área de Saúde	✗	✗	✓	✗	✗
Sistema Online	✓	✗	✗	✓	✓
Sistema de Vale Transporte Obrigatório	✗	✓	✗	✗	✗
Gera Escala	✓	✗	✓	✓	✓
Armazena Histórico de Escalas	✓	✗	✗	✓	✓
Regras da CLT	✓	✓	✓	✓	✓

Figura 4 - Comparativo entre os *Softwares*

3 Especificação do Sistema

Este capítulo descreve a modelagem deste trabalho. Nele, será apresentada a arquitetura do trabalho.

3.1 Mini-mundo

O sistema desenvolvido permite a elaboração, o controle e a consulta de escalas de trabalho em regime 24x7. Cada funcionário que compõe a escala deverá cumprir uma carga horária de 40 horas semanais. O sistema possui o conceito de restrição por perfil, isto é, cada usuário será classificado com um determinado perfil, podendo ser: um Administrador, um Gestor ou Funcionário, e será capaz apenas de acessar as funcionalidades referentes ao seu perfil. Todos os funcionários envolvidos serão usuários do sistema

O perfil de Administrador do sistema poderá cadastrar um usuário, incluindo o nome, matrícula, equipe, cargo, data de admissão, telefone, *e-mail*, perfil, turno e *status*. Além dos parâmetros necessários para a criação das tabelas como o calendário de feriados. Outros perfis deverão existir como o Funcionário e o de Gestor. Ao perfil de Funcionário é permitido consultar as escalas disponíveis de sua equipe. Na área restrita ao Gestor pode-se definir o turno de um funcionário e manter as escalas de suas equipes.

O Gestor poderá também visualizar os funcionários associados à sua equipe em ordem alfabética, alterar seus dados como: turno de trabalho (Manhã: 6:00 às 15:00, Tarde: 13:00 às 22:00, Noite: 22:00 às 6:00 e Comercial: 08:00 às 17:00), *status* (ativo ou inativo), cargo e incluir o período de férias que será considerado na escala como Férias (FE).

Por padrão, todo funcionário é cadastrado com turno Comercial e *status* ativo. Uma vez inativo, o funcionário não poderá ser incluído em escalas futuras.

O sistema possuirá uma área para definir as características da escala a ser elaborada, tais como: mínimo de pessoas necessárias por dia (dia útil e final de semana) de acordo com o turno e máximo de folgas consecutivas permitidas. Na distribuição das folgas, o sistema avaliará a quantidade de folgas disponíveis no mês (somatório de sábados, domingos e feriados), as características definidas e considera as normas estabelecidas pela CLT para conclusão da escala.

Uma vez a escala concluída, o gestor poderá alterar a folga ou o turno de um funcionário se desejar, desde que essas alterações não violem as normas da CLT. Também será possível definir na escala de um funcionário as condições: Folga (F), Trabalho (T), Hora

Extra (HE), Banco de Horas (BH) e Licença Médica (LM).

3.2 Regras de Negócio

A Tabela 1 abaixo relata as políticas, condições e restrições que foram aplicadas ao *software* que foi produzido por este trabalho, definindo as suas Regras de Negócio.

Tabela 1 - Regras de Negócio

RN01	Nome: Participação em equipes
	Descrição: Um funcionário poderá fazer parte de uma única equipe.
RN02	Nome: Quantidade mínima de funcionários na escala
	Descrição: A escala somente poderá ser elaborada se houver no mínimo 1 funcionário no turno comercial ou pelo menos 2 em cada um dos demais turnos.
RN03	Nome: Folgas em feriados
	Descrição: Nos dias considerados feriados todos os funcionários terão na escala a marcação de folga.
RN04	Nome: Período de descanso
	Descrição: Entre duas jornadas de trabalho haverá um período mínimo de 11 horas de descanso.
RN05	Nome: Jornada consecutiva
	Descrição: O máximo de dias consecutivos trabalhados é de 7 dias.
RN06	Nome: Folga aos domingos
	Descrição: Todo funcionário deverá folgar pelo menos 1 domingo por mês.
RN07	Nome: Mês válido para criação de escala
	Descrição: A escala somente será criada ou alterada em mês corrente ou subsequente ao mês corrente.
RN08	Nome: Funcionário inativo
	Descrição: O funcionário não participará de escalas futuras se estiver inativo.
RN09	Nome: Período de férias
	Descrição: Durante o período de férias, o funcionário não poderá trabalhar.
RN10	Nome: Remoção de usuários
	Descrição: O usuário somente poderá ser removido se estiver inativo há mais de 2 meses.
RN11	Nome: Alteração de escala de usuário removido
	Descrição: A escala de um usuário removido não poderá ser alterada.

3.3 Modelo de Caso de Uso

A construção de um Modelo de Caso de Uso (MCU) envolve a construção de duas perspectivas, a gráfica e a textual. A primeira corresponde ao Diagrama de Caso de Uso e a segunda corresponde à Descrição dos Casos de Uso. Assim, essa sessão apresenta a modelagem do sistema feita pelo Diagrama de Caso de Uso e as suas descrições textuais.

3.3.1 Diagrama de Caso de Uso

A Figura 5 abaixo demonstra o Diagrama de Caso de Uso do *software* produzido por este trabalho. Nele estão presentes os atores do sistema, assim como suas ações.

No diagrama abaixo se observa a interação dos atores com o sistema e suas respectivas funções. O Administrador é responsável pela inclusão e a manutenção dos usuários, como inclusão, remoção, e alteração nos dados. O Funcionário apenas pode visualizar a escala depois de concluída. O Gestor tem a função de elaborar a escala, alterar a escala após pronta e disponibilizar a escala, assim como alterar dados do funcionário.

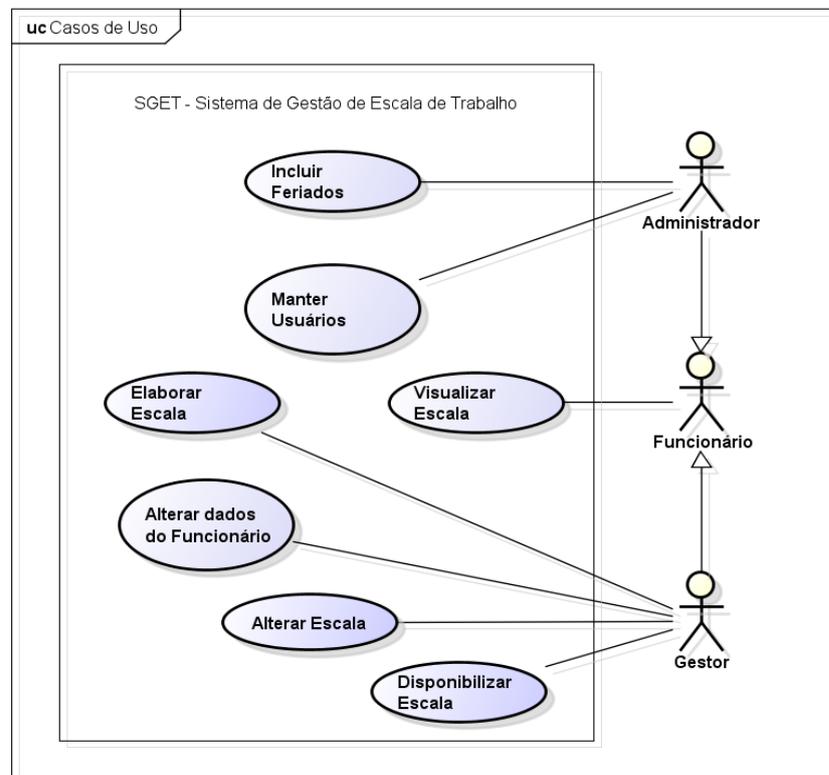


Figura 5 - Diagrama de Casos de Uso

3.3.2 Descrições Textuais dos Casos de Uso

A seguir, são descritos de forma textual os Casos de Uso, incluindo suas principais características de execução das tarefas e onde se enquadram as Regras de Negócio já definidas na Tabela 1 deste trabalho.

Manter Usuários (CSU01)

Sumário: O Administrador realiza o cadastro (inclusão, remoção, alteração e consulta) dos dados dos usuários.

Ator Primário: Administrador

Pré-condições: O Administrador está identificado pelo sistema.

Fluxo Principal:

1. O Administrador solicita a manutenção dos usuários;
2. O sistema apresenta as operações que podem ser realizadas: a inclusão de um novo usuário, a alteração dos dados de um usuário, a exclusão de um usuário e a consulta de usuários;
3. O Administrador indica a opção a realizar ou opta por finalizar o caso de uso;
4. O Administrador seleciona a operação desejada: Inclusão, Exclusão, Alteração ou Consulta;
5. Se o Administrador deseja continuar com a manutenção, o caso de uso retorna ao passo 2; caso contrário, o caso de uso termina.

Fluxo Alternativo (4): Inclusão

- a) O Administrador requisita a inclusão de um usuário;
- b) O sistema apresenta um formulário em branco para que os detalhes do funcionário (nome, matrícula, equipe, cargo, data de admissão, telefone, *e-mail*, perfil, turno e *status*) sejam incluídos (conforme RN01);
- c) O Administrador fornece os detalhes do novo usuário;
- d) O sistema verifica a validade dos dados. Se os dados forem válidos, inclui o novo

usuário; caso contrário, o sistema reporta o fato, solicita novos dados e repete a verificação.

Fluxo Alternativo (4): Remoção

- a) O Administrador seleciona um usuário e requisita ao sistema que o remova (conforme RN10);
- b) O sistema realiza a remoção.

Fluxo Alternativo (4): Alteração

- a) O Administrador altera um ou mais dos detalhes sobre um funcionário e requisita a sua atualização;
- b) O sistema verifica a validade dos dados e, se eles forem válidos, altera os dados do funcionário.

Fluxo Alternativo (4): Consulta

- a) O Administrador solicita a realização de uma consulta sobre a lista de usuários;
- b) O sistema apresenta uma lista com o nome, matrícula, equipe, cargo, data de admissão, telefone, *e-mail*, perfil, turno e *status* de todos os usuários, permitindo que o Administrador selecione o usuário desejado;
- c) O Administrador seleciona um usuário;
- d) O sistema apresenta os detalhes do usuário.

Pós-condições: Um usuário foi inserido ou removido, ou seus detalhes foram alterados.

Regras de Negócio: RN01 e RN10.

Incluir Feriados (CSU02)

Sumário: Serão adicionados todos os feriados existentes que interfiram no funcionamento da escala.

Ator Primário: Administrador

Pré-condições: Haver feriados a serem cadastrados.

Fluxo Principal:

1. O Administrador requisita o calendário;
2. O sistema apresenta o calendário para o Administrador;
3. O Administrador sinaliza os dias que há feriado;
4. O sistema registra as alterações;
5. O sistema atualiza o calendário com os feriados.

Fluxo de Exceção (3): Feriado já cadastrado

- a) No dia sinalizado pelo Administrador já havia cadastro de feriado: o sistema reporta o fato e o caso de uso retorna ao passo 3.

Pós-condições: O sistema gera um novo calendário com a inclusão dos feriados.

Elaborar Escala (CSU03)

Sumário: O Gestor realiza a elaboração da escala (modificação das características e criação da escala).

Ator Primário: Gestor

Pré-condições: O Gestor está identificado no sistema.

Fluxo Principal:

1. O Gestor seleciona o mês para o qual deseja elaborar a escala;
2. O sistema apresenta as operações que podem ser realizadas: a criação de uma nova escala e alteração das características da escala;
3. O Gestor indica a opção a realizar ou opta por finalizar o caso de uso;
4. O Gestor seleciona a operação desejada: Criar Escala ou Editar Características;
5. Se o Gestor deseja continuar com a elaboração, o caso de uso retorna ao passo 4; caso contrário, o caso de uso termina.

Fluxo Alternativo (4): Criar Escala

- a) O sistema apresenta a quantidade de folgas permitidas para o mês selecionado;
- b) O sistema lista os funcionários que pertencem a equipe do Gestor (conforme RN02 e RN08);

- c) O sistema calcula as condições para distribuição das folgas (conforme RN03, RN04, RN05, RN06 e RN09);
- d) O sistema apresenta a escala pronta.

Fluxo Alternativo (4): Editar Características

- a) O sistema apresenta quais as características passíveis de edição (mínimo de funcionários necessários em dias úteis, mínimo de funcionários necessários nos finais de semana, mínimo de funcionários necessários no feriado e máximo de folgas consecutivas);
- b) O Gestor realiza as alterações que considerar necessárias;
- c) O sistema solicita confirmação das alterações realizadas;
- d) Se o Gestor confirmar as alterações, o caso de uso prossegue do passo 2.

Fluxo de Exceção (1): Violação da RN07

- a. Se o mês escolhido for anterior ao mês corrente (conforme RN07), o sistema informa ao Gestor que o mês informado não é válido.

Pós-condições: A escala poderá ser divulgada ou alterada.

Regras de Negócio: RN02, RN03, RN04, RN05, RN06, RN07, RN08 e RN09.

Disponibilizar Escala (CSU4)

Sumário: Gestor disponibiliza a escala para visualização.

Ator Primário: Gestor

Pré-condições: O Gestor estar identificado no sistema.

Fluxo Principal:

1. O Gestor solicita a disponibilização da escala;
2. O sistema apresenta as operações que podem ser realizadas: disponibilizar a visualização apenas para os funcionários da equipe ou disponibilizar a visualização para todos os funcionários da empresa;
3. O Gestor indica a opção a realizar ou opta por finalizar o caso de uso;
4. O Gestor seleciona a operação desejada: Para Todos, Para a Equipe;

5. Se o Gestor deseja continuar com a disponibilização, o caso de uso retorna ao passo 2; caso contrário, o caso de uso termina.

Fluxo Alternativo (4):

a) O sistema habilita a visualização da escala para todos os funcionários.

Fluxo Alternativo (4):

a) O sistema habilita a visualização da escala para os funcionários pertencentes à equipe do Gestor.

Alterar Escala (CSU05)

Sumário: O Gestor realiza alterações na escala dos funcionários.

Ator Primário: Gestor

Pré-condições: O Gestor está identificado pelo sistema e a escala deve estar pronta.

Fluxo Principal:

1. O Gestor solicita a alteração da escala;
2. O sistema apresenta as escalas disponíveis para alteração (conforme RN07);
3. O Gestor seleciona uma escala;
4. O sistema apresenta os funcionários com suas respectivas folgas distribuídas;
5. O Gestor seleciona um funcionário;
6. O sistema apresenta os dias referentes ao mês da escala selecionada;
7. O Gestor seleciona um dia;
6. O sistema apresenta as condições que podem ser atribuídas ao dia selecionado: Folga, Trabalho, Férias, Hora Extra, Banco de Horas ou Licença Médica;
7. O Gestor seleciona a opção desejada: F, T, FE, HE, BH, LM;
8. Se o Gestor deseja continuar com a alteração, o caso de uso retorna ao passo 4; caso contrário, o caso de uso termina.

Pós-condições: Alteração na escala é realizada.

Regras de Negócio: RN07

Alterar Dados do Funcionário (CSU06)

Sumário: O Gestor altera os dados do funcionário.

Ator Primário: Gestor

Pré-condições: O Gestor está identificado pelo sistema.

Fluxo Principal:

1. O Gestor solicita a alteração de dados dos funcionários;
2. O sistema apresenta a lista de funcionários da equipe;
3. O Gestor seleciona um funcionário (conforme RN11);
4. O sistema apresenta os dados passíveis de alteração (Turno e Cargo);
5. O Gestor realiza as alterações necessárias;
6. O sistema solicita a confirmação dos dados;
7. Se o Gestor deseja continuar com a alteração, o caso de uso retorna ao passo 2; caso contrário, o caso de uso termina.

Pós-condições: O turno ou o cargo do funcionário foi alterado.

Regras de Negócio: RN11.

3.4 Modelo de Classes

Dentre os modelos da Linguagem de Modelagem Unificada (UML), o diagrama de classes é o mais rico em termos de notação. Nessa seção são apresentados os elementos do diagrama de classes utilizado para a construção do modelo de classes do *software* desenvolvido.

3.4.1 Diagrama de Classe

Na Figura 6, são apresentados os elementos do diagrama de classes, com seus atributos e operações.

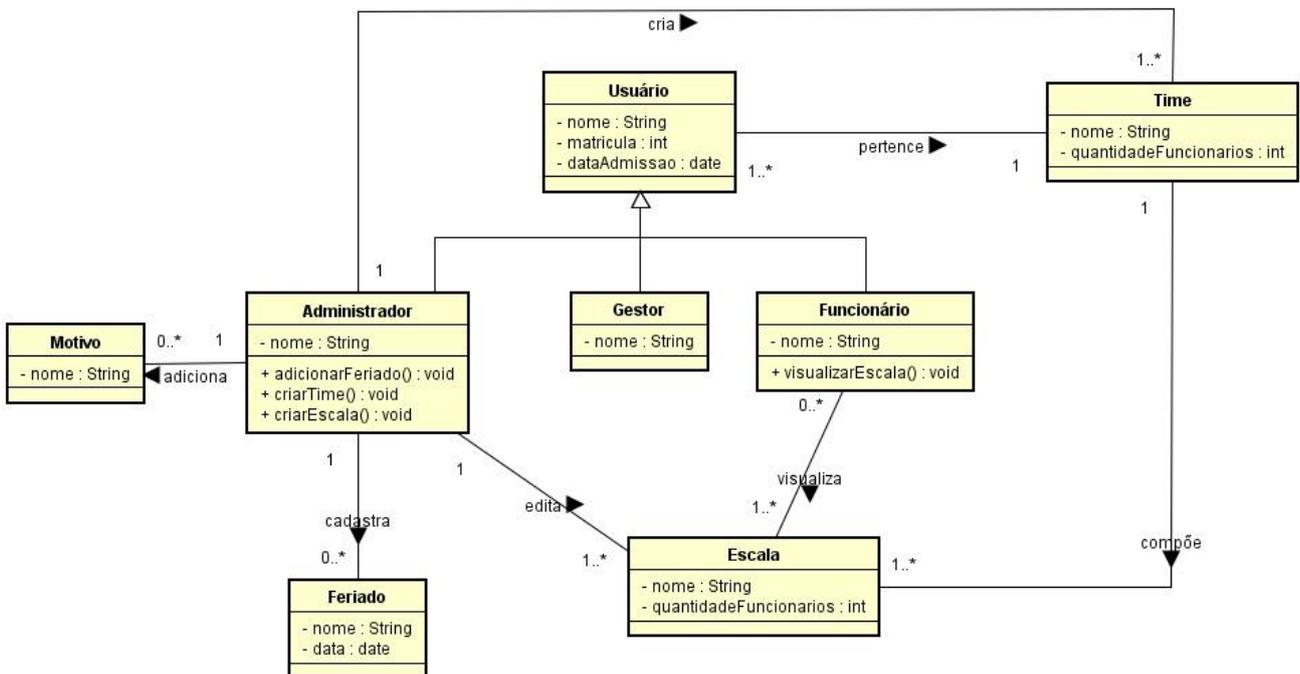


Figura 6 - Diagrama de Classes

A classe *Escala* é uma importante classe do sistema, é ela que distribui as folgas dos funcionários. Essa classe é utilizada para compor a classe *Equipe*.

A classe *Time* apresenta informações sobre a equipe e a quantidade de funcionários.

A classe *Funcionario* apresenta informações relevantes sobre o funcionário, além de herdar as características da classe *Usuario*. Ela se relaciona com uma das principais classes do sistema: *Escala*. O método *visualizarEscala* permite a visualização da escala após sua divulgação.

A classe *Gestor* é uma das principais do sistema, ela herda as características da classe *Usuario*.

A classe *Administrador* é uma das mais importantes, ela herda as informações da classe *Usuário*. Contém os seguintes métodos:

- *adicionarFeriado* - permite que sejam incluídos os feriados de cada mês na escala;
- *criarTime* – divide os funcionários por equipe de trabalho;
- *criarEscala* - é o principal método desta classe, uma vez que o sistema calcula as condições para distribuição das folgas e apresenta a escala pronta.

A superclasse *Usuário* apresenta as características herdadas pelas classes *Funcionario*, *Gestor* e *Administrador*.

A classe *Feriado* possui informações do nome e data dos feriados.

A classe *Motivo* apresenta os motivos pelos quais, eventualmente, o funcionário virá a não comparecer ao trabalho.

3.5 Comparativo

Nesta seção será feito um comparativo para se analisar o desempenho de uma mesma escala sendo feita manualmente e com o SGET. Foi analisado o tempo gasto, assim como todos os passos que eram utilizados para a confecção sem uma ferramenta de tecnologia.

Os dados são referentes a um quadro de vinte funcionários. Quanto maior o quadro de funcionários, menos viável se torna o uso manual de montagem de escalas. Além disso, deslocar um funcionário e retirá-lo de suas funções apenas para se dedicar a construção da escala se torna um desperdício de tempo e produtividade.

Conforme representado na Tabela 2, o ganho expressivo de tempo com a ferramenta SGET foi a principal qualidade.

Tabela 2 - Comparativo de Tempo

Atividade	Manualmente	SGET
Levantamento da Quantidade de pessoas por dia	10 minutos	200 segundos
Verificação de Férias de cada colaborador	60 minutos	60 segundos
Verificação da Quantidade de Folgas por mês	100 minutos	120 segundos
Verificação de Feriados	5 minutos	30 segundos
Caso haja Feriado, alocação dos feriados	140 minutos	160 segundos
Cálculo	800 minutos	30 segundos
Total	1115 minutos	10 minutos

3.6 Banco de Dados

Nesta seção é apresentado o banco de dados utilizado neste trabalho. Optou-se pela linguagem de programação *PHP: Hypertext Preprocessor (PHP)*, linguagem orientada a objeto que torna um *website* dinâmico, também foi aplicado o *Framework Laravel* e o banco de dados utilizado foi o *MySQL*.

A princípio montou-se o projeto lógico de banco de dados, relatado a seguir, com o objetivo de se estudar o esquema conceitual e se ter uma ideia geral do banco, facilitando a implementação mais a frente.

escala_equipe(**id**, escalas_id, equipes_id, turnos_id, funcionários_id, data, motivos_id)

RI01: escalas_id não pode ser nulo

RI02: equipes_id não pode ser nulo

RI03: turnos_id não pode ser nulo

RI04: funcionários_id não pode ser nulo

RI05: data não pode ser nulo

escalas(**id**, mês, ano, min_func, max_folgas, escalas_id)

RI01: mês não pode ser nulo

RI02: ano não pode ser nulo

RI03: min_func não pode ser nulo

RI04: max_folgas não pode ser nulo

RI05: escalas_id não pode ser nulo

turnos(**id**, nome, descrição)

RI01: nome não pode ser nulo

motivos(**id**, nome)

funcionário(**id**, primeiro_nome, sobrenome)

RI01: primeiro_nome não pode ser nulo

RI02: sobrenome não pode ser nulo

funcao(**id**, descrição)

funcao_funcionario(funcionário_id, funcao_id)

RI01: funcionário_id não pode ser nulo

RI02: funcao_id não pode ser nulo

equipes(**id**, status, funcionários_id)

RI01: funcionários_id não pode ser nulo

feriados(**id**, descricao, data)

RI01: data não pode ser nulo

A partir do modelo lógico, montou-se o modelo físico do banco de dados. Concluído o modelo lógico, o próximo passo foi a construção do banco de dados, conforme ilustrado na Figura 7, a qual representa o modelo lógico do banco de dados representado na forma de um diagrama Entidade Relacionamento.

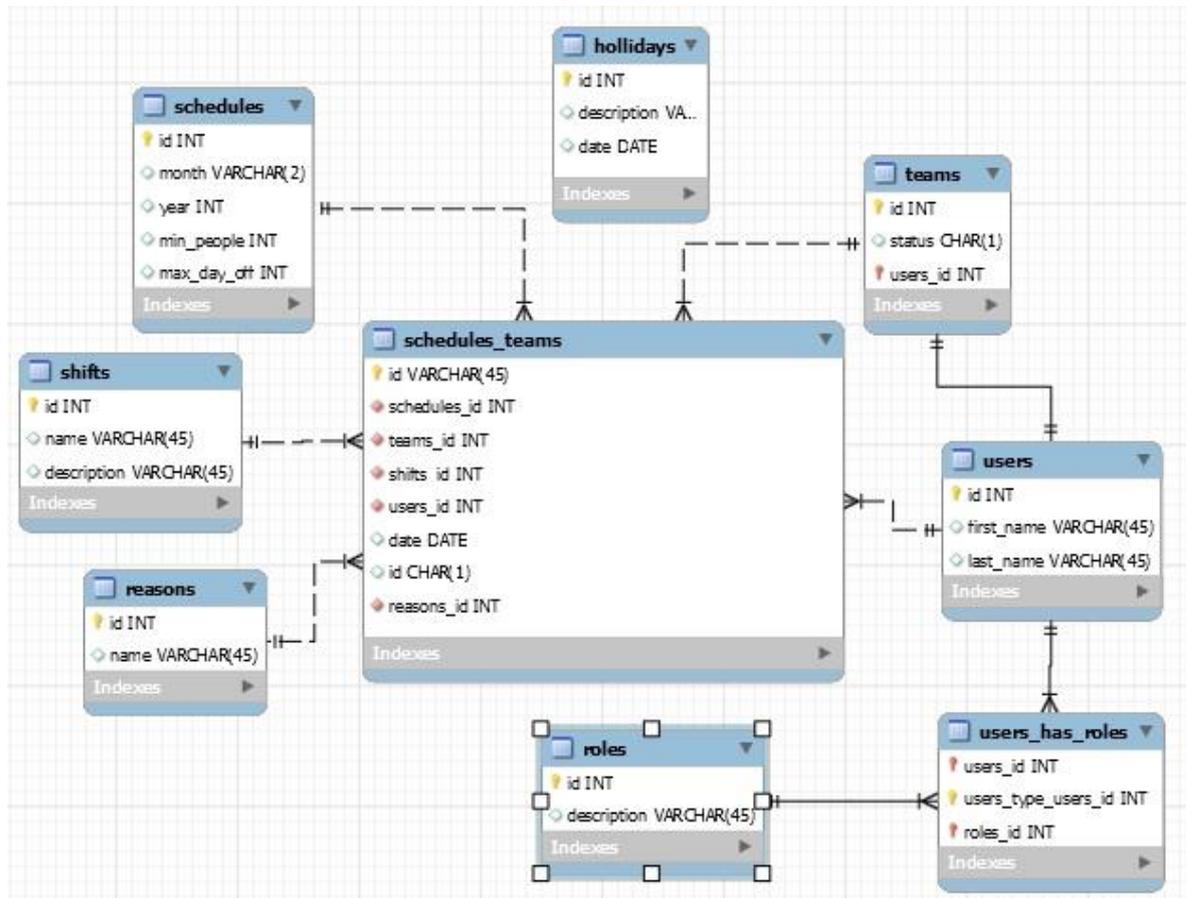


Figura 7 - Modelo Lógico do Banco de Dados

A tabela *schedules_teams* é a central, é nela que serão armazenados os dados relevantes sobre a escala como as escalas, funcionários, turnos, equipe dos funcionários e motivos para o não comparecimento ao trabalho, seja ele por folga, falta, licença ou férias.

A tabela *users* é responsável por armazenar o nome e sobrenome de cada funcionário. Cada funcionário terá uma função dentro de uma empresa, para isso foi criada a tabela *users_has_holes* que possui o tipo de usuário e sua função. Para cada função foi criada a tabela *roles* que armazena a descrição de cada função desempenhada na empresa.

Um funcionário deverá pertence a uma equipe, seja ela de funcionários padrão, gestores ou os administradores, para isso, foi criado uma tabela *teams* que armazena os funcionários e seu *status*.

A tabela *shift* representa o turno no qual o funcionário irá trabalhar podendo ser ele manhã, tarde ou noite, para isso, os dados da tabela são nome e descrição. Durante o mês o funcionário terá folgas, poderá ter férias, faltas e até licença médica, por isso, foi criada a tabela *reasons*, contendo o nome do motivo ao não comparecimento ao trabalho. Em caso de feriado no mês, foi criada a tabela *hollidays* que aloca os feriados de cada mês com sua descrição e data. E por último a tabela *schedules* responsável pela escala em si, essa tabela armazena os dados de mês e ano a qual a escala pertence, mínimo de pessoas trabalhando e o máximo de dias de folga que o funcionário terá direito.

3.7 Interface

Nesta seção é apresentada a interface gráfica do sistema SGET. A interface é responsável pela interação entre o usuário e o sistema. Uma interface limpa e de fácil compreensão se torna um ponto positivo, já que os usuários não apresentarão grandes dúvidas e descarta uma possível necessidade de treinamento para o manuseio. Buscou-se a implementação de uma página autoexplicativa, com um menu bem claro sobre cada função.

O sistema possui uma página inicial que conta, em caso de usuário administrador, as opções de cadastro de usuários, cadastro de feriado, cadastro de times, cadastro de motivos e, enfim, cadastro da escala em si.

Assim, serão apresentadas as telas de início, cadastro e exibição de novo feriado, cadastro de novo funcionário e um exemplo de escala pronta. A Figura 8 representa a tela inicial de um usuário Administrador. Como se observa, é uma interface simples e de fácil compreensão pelo usuário. Nela, o administrador poderá fazer cadastro de novos usuários, feriados, motivos, alocar novos times e gerar a escala.

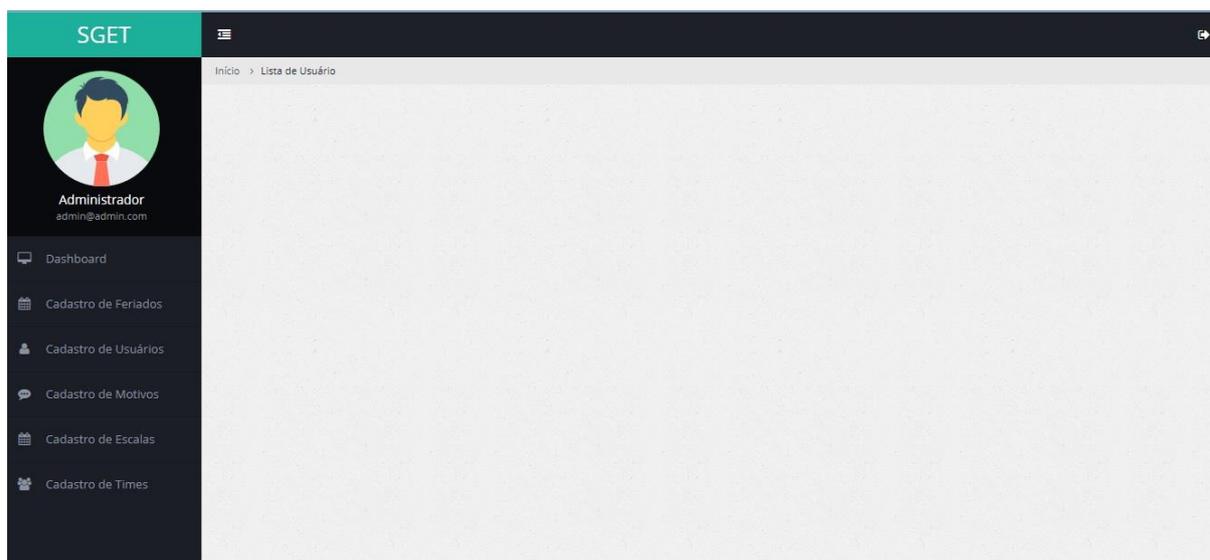


Figura 8 - Interface da Pagina Inicial

A Figura 9 representa a interface de lista de feriados já cadastrados do mês. Além da relação dos feriados, nessa tela há a opção “Novo”, que permite ao Administrador a inclusão de novos feriados, se houverem.

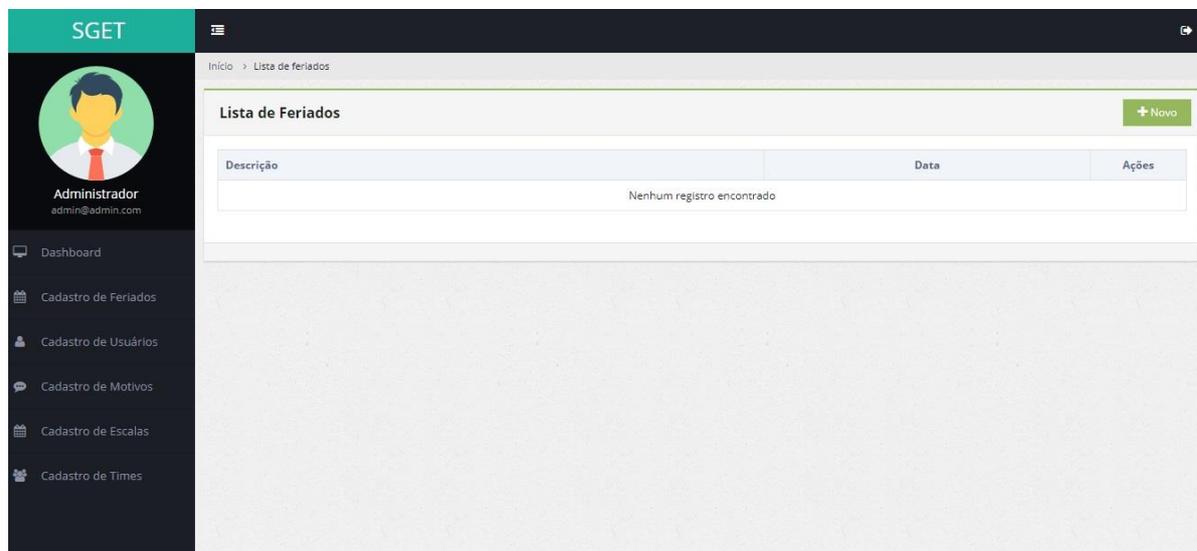


Figura 9 - Interface Lista de Feriados

A Figura 10 representa a interface de cadastro de novos funcionários. Para esta ação, o Administrador deverá informar o nome e sobrenome, obrigatoriamente, o *e-mail* do funcionário, em qual nível ele pertence na empresa, seu número de matrícula, cadastrar e confirmar uma senha e fornecer sua data de admissão na empresa. Depois de concluído o preenchimento e salvo, o novo funcionário estará salvo no sistema e à disposição para a próxima criação de escala.

The screenshot shows the 'Novo Usuário' (New User) form in the SGET system. The interface includes a sidebar with navigation options: Dashboard, Cadastro de Feriados, Cadastro de Usuários, Cadastro de Motivos, Cadastro de Escalas, and Cadastro de Times. The main form fields are:

- Primeiro Nome: Ex. Jose
- Último Nome: Ex. Silva
- E-mail: Ex. josesilva@exemplo.com
- Nível: Administrador (dropdown menu)
- Matricula: Ex. 0000001
- Senha: [input field]
- Confirme a senha: [input field]
- Data de admissão: [calendar icon]

Buttons for 'Cancelar' and 'Salvar' are located at the bottom of the form.

Figura 10 - Interface de Cadastro de Novo Usuário

A Figura 11 representa a interface de Motivos. Em casos de falta, férias, atestado ou a solicitação de alteração da folga já estipulada, o administrador acrescenta na escala o motivo do não comparecimento do funcionário, depois da escala já criada. É possível a criação de mais motivos posteriormente.

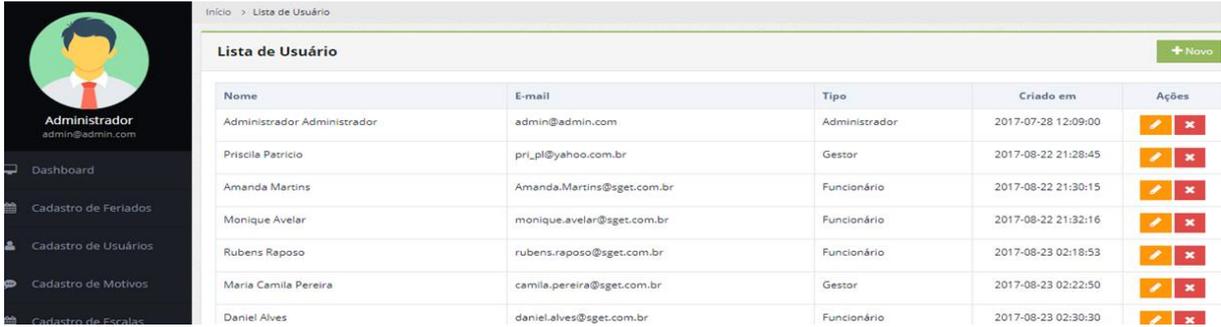
The screenshot shows the 'Lista de Motivos' (List of Reasons) page in the SGET system. The page includes a sidebar with navigation options: Dashboard, Cadastro de Feriados, Cadastro de Usuários, Cadastro de Motivos, Cadastro de Escalas, and Cadastro de Times. The main content area shows a table with the following data:

Motivo	Ações
folga	[edit icon] [delete icon]
férias	[edit icon] [delete icon]
falta	[edit icon] [delete icon]
atestado	[edit icon] [delete icon]

A '+ Novo' button is located in the top right corner of the table area.

Figura 11 - Interface de Motivos

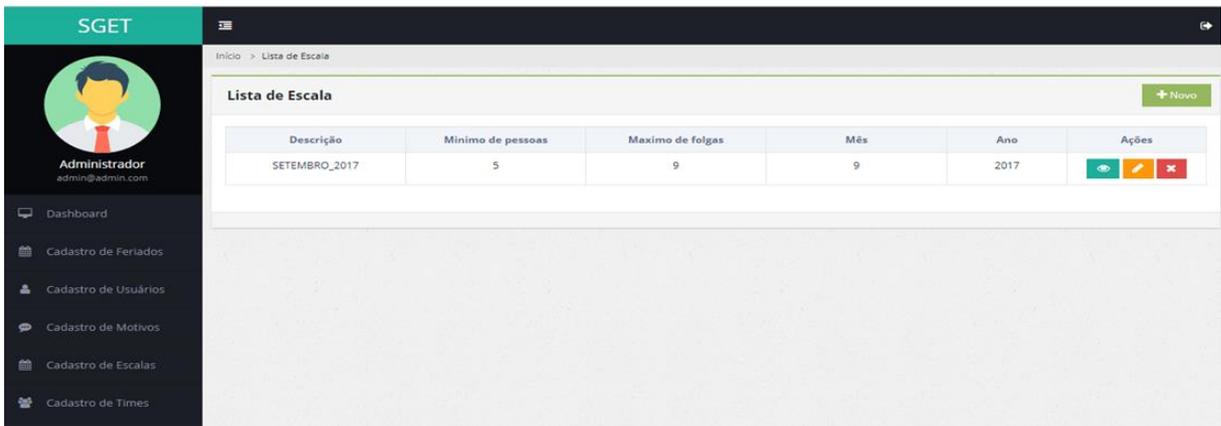
A Figura 12 representa a interface referente a lista de todos os usuários cadastrados no sistema. É possível a edição e exclusão de membros já cadastrados e adição de novo membros.



Nome	E-mail	Tipo	Criado em	Ações
Administrador Administrador	admin@admin.com	Administrador	2017-07-28 12:09:00	
Priscila Patricio	pri_pl@yahoo.com.br	Gestor	2017-08-22 21:28:45	
Amanda Martins	Amanda.Martins@sget.com.br	Funcionário	2017-08-22 21:30:15	
Monique Avelar	monique.avelar@sget.com.br	Funcionário	2017-08-22 21:32:16	
Rubens Raposo	rubens.raposo@sget.com.br	Funcionário	2017-08-23 02:18:53	
Maria Camila Pereira	camila.pereira@sget.com.br	Gestor	2017-08-23 02:22:50	
Daniel Alves	daniel.alves@sget.com.br	Funcionário	2017-08-23 02:30:30	

Figura 12 - Interface de Lista de Usuários

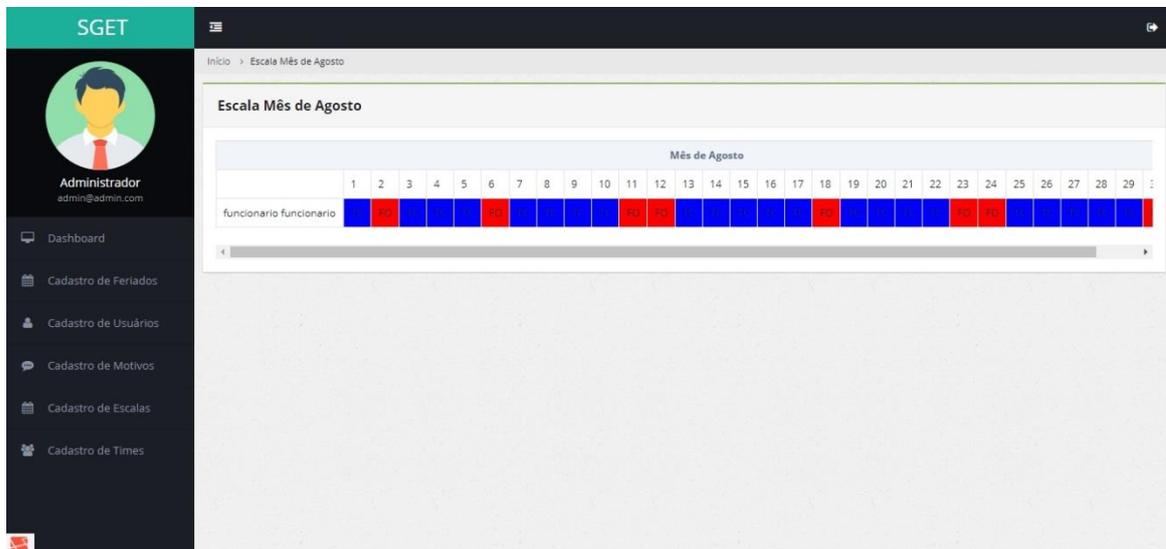
A Figura 13 representa a interface que lista as escalas que já foram criadas. Tendo a possibilidade de visualização, edição e exclusão ou ainda a criação de novas escalas.



Descrição	Mínimo de pessoas	Máximo de folgas	Mês	Ano	Ações
SETEMBRO_2017	5	9	9	2017	

Figura 13 - Lista de escalas já criadas por mês

A Figura 14 representa uma das interfaces mais importantes, a tela da escala criada. Nela é exibido o mês desejado, os funcionários alocados pelo Administrador já com suas folgas e dias de trabalho.



		Mês de Agosto																													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	:
funcionario	funcionario	blue	red	blue																											

Figura 14 - Interface da Escala Pronta

4 Avaliação Experimental

Nesta seção é apresentada a avaliação experimental feita para testar a eficácia do sistema implementado durante o projeto. A avaliação experimental contou com 36 voluntários que testaram o sistema durante dois dias. A avaliação contou com pessoas que trabalham com elaborações de escala, entendem o funcionamento básico de como uma escala é projetada, mas também com pessoas que nunca tiveram antes a experiência de montar uma escala, seja manualmente ou com o auxílio de um sistema.

4.1 SUS – *System Usability Scale*

Para essa avaliação, foi utilizada uma metodologia de escala numérica de usabilidade. Um teste de usabilidade com usuários reais do produto pode rapidamente apontar as tarefas que as pessoas têm mais dificuldade na utilização do produto, mas ainda assim não consegue indicar “o quão grande” o problema de usabilidade é, em uma escala numérica (TEIXEIRA, 2015).

Foi escolhido o *System Usability Scale* (SUS). O SUS foi criado por John Brooke em 1986, é um método muito conhecido, além de ser simples. Não é um teste longo ou cansativo para o pesquisado e conta com uma base científica para provar suas conclusões. Seu campo de avaliação se estende a qualquer tipo de sistema que possua uma interface de interação com o usuário.

Os critérios usados pelo SUS são a efetividade do sistema, eficiência e satisfação. Três perguntas são essenciais para cada um dos critérios. Se o usuário consegue completar seus objetivos, seu sistema é efetivo, se não é necessário muito esforço e recurso para o manuseio do sistema, então ele é eficiente. Se a experiência ao utilizar o sistema foi satisfatória, então ele completa o terceiro e último critério, a satisfação.

O mecanismo de avaliação do SUS consiste em dez perguntas que serão respondidas pelos testadores do sistema em uma escala de 1 a 5, onde a opção 1 significa que o avaliador discorda completamente e a opção 5 que significa que concorda completamente com a afirmação sugerida. A seguir serão listadas as dez afirmações base que são usadas para a avaliação, podendo ser modificadas até serem moldadas perfeitamente para que se encaixem em cada tipo de sistema.

1 - Eu acho que gostaria de usar esse sistema com frequência.

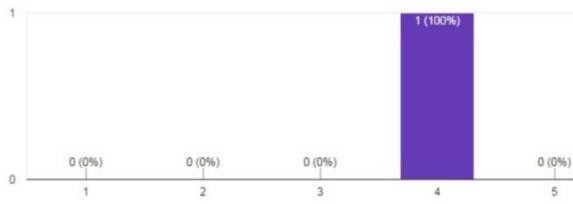
- 2 - Eu acho o sistema desnecessariamente complexo.
- 3 - Eu achei o sistema fácil de usar.
- 4 - Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema.
- 5 - Eu acho que as várias funções do sistema estão muito bem integradas.
- 6 - Eu acho que o sistema apresenta muita inconsistência.
- 7 - Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.
- 8 - Eu achei o sistema atrapalhado de usar.
- 9 - Eu me senti confiante ao usar o sistema.
- 10 - Eu precisei aprender várias coisas novas antes de conseguir usar o sistema.

A partir da avaliação, foram coletados os resultados obtidos dessas dez perguntas para calcular a pontuação final que o sistema recebeu no geral. O cálculo da pontuação consistiu em, para cada pergunta ímpar (pergunta de número 1, 3, 5, 7, 9) subtrair um ponto do número assinalado pelo usuário, ou seja, caso o usuário tenha dado nota 4 na questão 3, subtraíu um ponto do 4. Nos casos em que a pergunta foi par (perguntas de número 2, 4, 6, 8, 10), subtraíu a resposta dada do número 5, ou seja, se o usuário, na segunda questão, tenha respondido com o número 1, subtraíu 1 do número 5. Após o cálculo de cada valor individual das perguntas, foi somado o resultado das dez perguntas e, dessa soma final, foi multiplicado por 2.5. A pontuação final pode variar de 0 a 100. A média estabelecida pelo SUS são 68 pontos, se a pontuação final ficar abaixo dessa média, para o SUS, seu sistema está enfrentando problemas de usabilidade, é sinal que se deve dar uma atenção maior e fazer melhorias.

Na Figura 15 a seguir é possível observar um exemplo prático do cálculo implementado em uma das avaliações que foi feita no SGET. Se baseando nos resultados obtidos a seguir, tem-se na primeira pergunta uma resposta 4, lembrando a regra das questões de número ímpar deve-se reduzir 1 ponto do que foi obtido, portanto, na primeira questão tem-se a nota 3. Seguindo adiante, a regra aplicada deve ser a dos números pares, portanto, reduzir o número obtido de 5, como obteve-se uma avaliação 1, o resultado final é 4. Dando seguimento, tem-se outra questão ímpar (resultado final $4-1=3$), na quarta obteve-se um 5 (resultado 0), na quinta mais um 4 (resultado 3), na sexta questão um resultado 1 (totalizando 4), na sétima outro 4 (resultado 3), na oitava outro 1 (resultado 4), na nona questão 4 (resultado 3), na décima e última questão obteve-se uma pontuação 1 (resultado 4). Somando-se todos os resultados chega-se ao total de 31 pontos, deve-se multiplicar esse resultado por 2,5, o que resulta em 77,5. Com a primeira e única avaliação feita, o SGET estaria acima da média exigida pelo SUS.

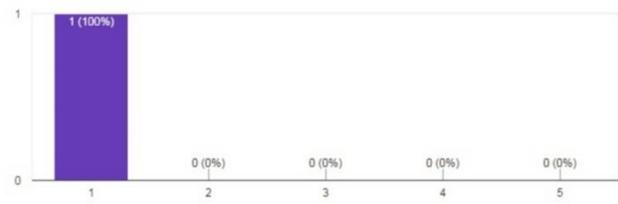
Eu acho que gostaria de usar esse sistema com frequência.

1 response



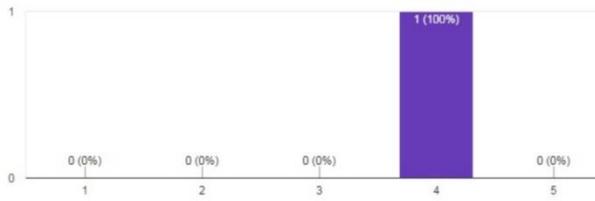
Eu acho o sistema desnecessariamente complexo.

1 response



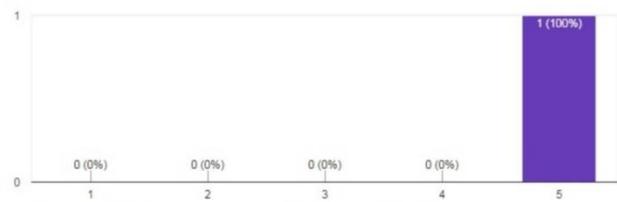
Eu achei o sistema fácil de usar.

1 response



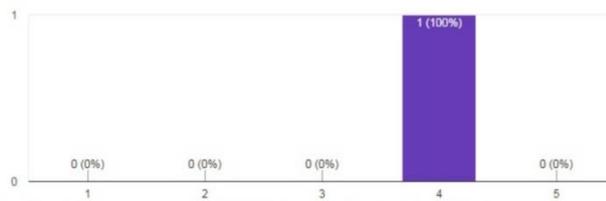
Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema.

1 response



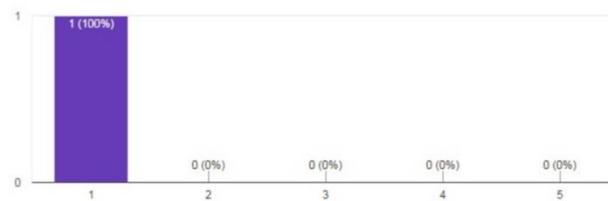
Eu acho que as várias funções do sistema estão muito bem integradas.

1 response



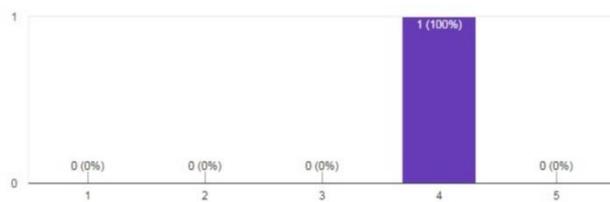
Eu acho que o sistema apresenta muita inconsistência.

1 response



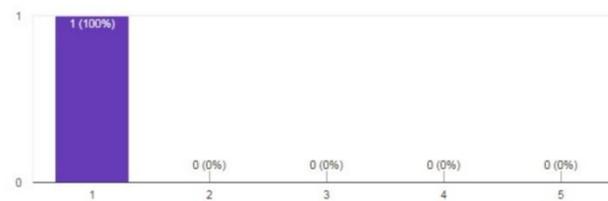
Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.

1 response



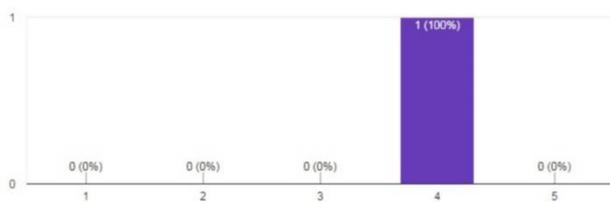
Eu achei o sistema atrapalhado de usar.

1 response



Eu me senti confiante ao usar o sistema.

1 response



Eu precisei aprender várias coisas novas antes de conseguir usar o sistema.

1 response

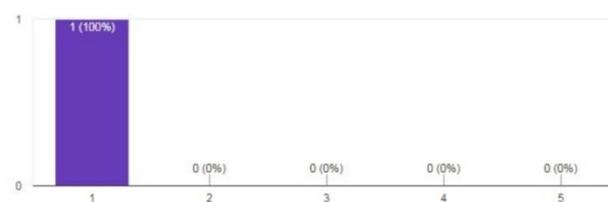


Figura 15 - Exemplo da Implementação do SUS

4.2 Resultados

A importância da Avaliação Experimental é poder ter uma resposta imediata de como o público qualificou o projeto. A partir dessas respostas pode-se ter um melhor direcionamento de quais aspectos devem ser feitos as melhorias, em quais setores, o que mais agradou e ter uma base se foi ou não bem aceito. Assim, o sistema passou por uma avaliação com usuários com conhecimento nulo, intermediário e avançado em escalas considerando com base o SUS.

O sistema foi avaliado por 36 pessoas. Sendo 1 experiente na elaboração de escalas de trabalho complexas para grandes equipes, 13 profissionais que trabalham em regime de escala e 22 usuários que não haviam tido qualquer contato com uma escala anteriormente. Os usuários foram agrupados conforme experiência com escalas. Assim, os usuários compreendidos entre as posições p1 a p22 são do grupo que não haviam tido qualquer tipo de experiência com escalas anteriormente. Aqueles usuários compreendidos entre as posições p23 e p35 são do grupo que trabalham em regime de escala de trabalho e que utilizariam o SGET para consulta da escala. Já a posição p36 representa o único usuário que tem experiência em elaborar escalas complexas. Antes do primeiro contato com o SGET, foi apresentado o objetivo e o que se esperava dos testes para todos os usuários, assim como as perguntas que deveriam ser respondidas ao final.

A seguir, na sequência de Figuras 16 a 19, é apresentado o resultado final do questionário respondido pelos voluntários. É possível observar as diferentes opiniões para algumas das afirmações, algumas com opiniões bem divididas e outras, como a décima, foi praticamente unânime a escolha.

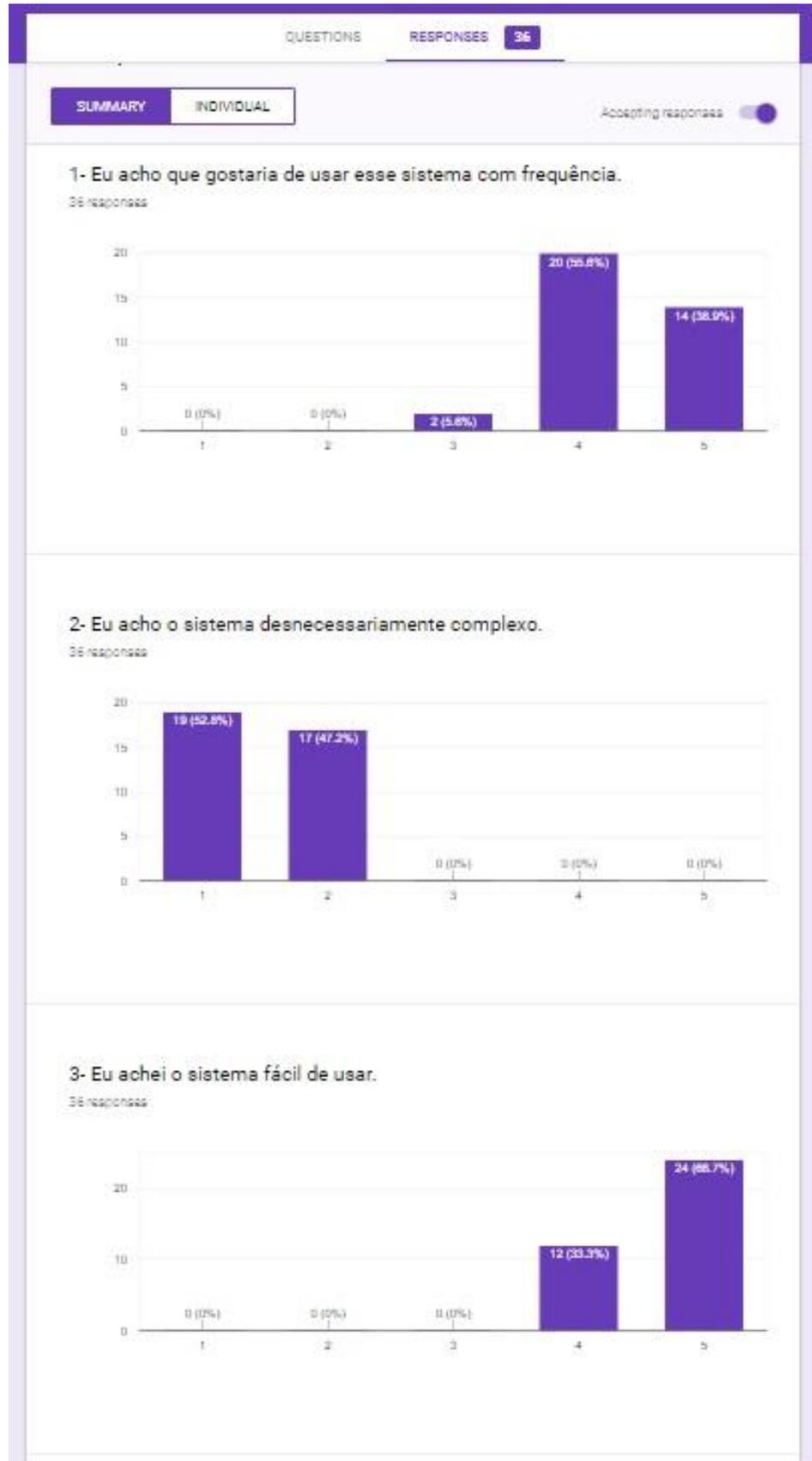


Figura 16 - Resultado em gráfico da Avaliação Experimental

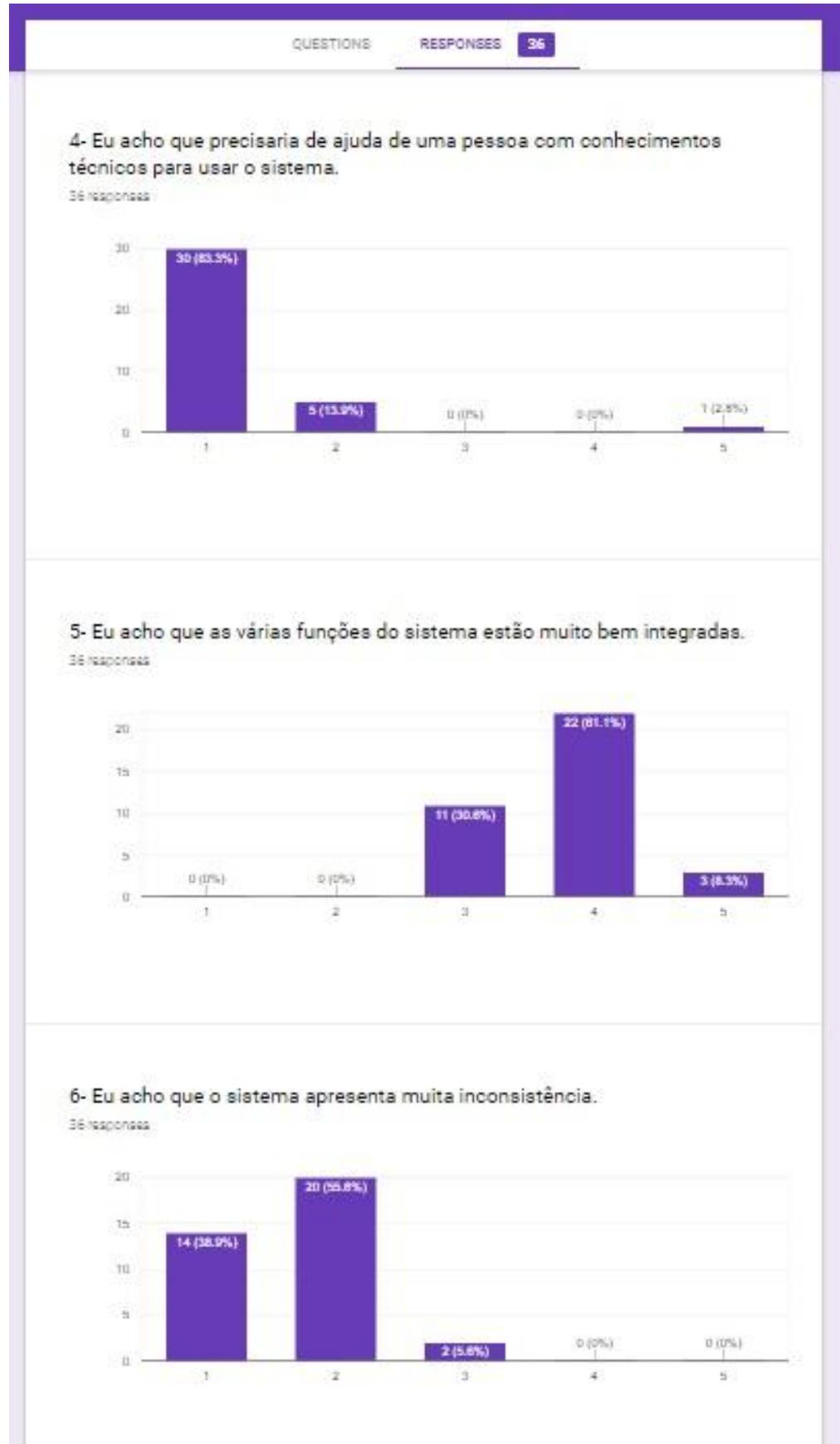


Figura 17 - Resultado em gráfico da Avaliação Experimental

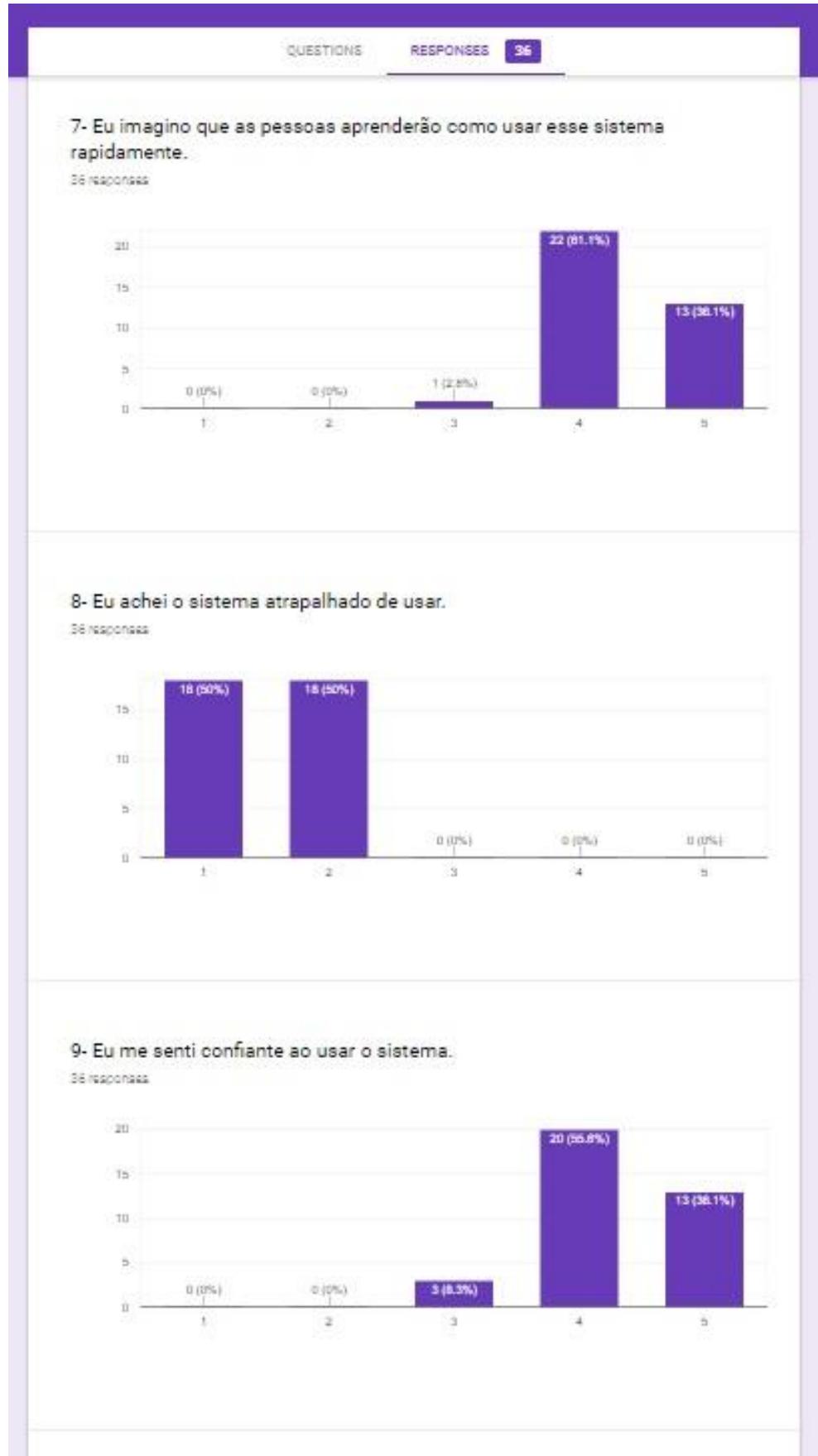


Figura 18 - Resultado em gráfico da Avaliação Experimental

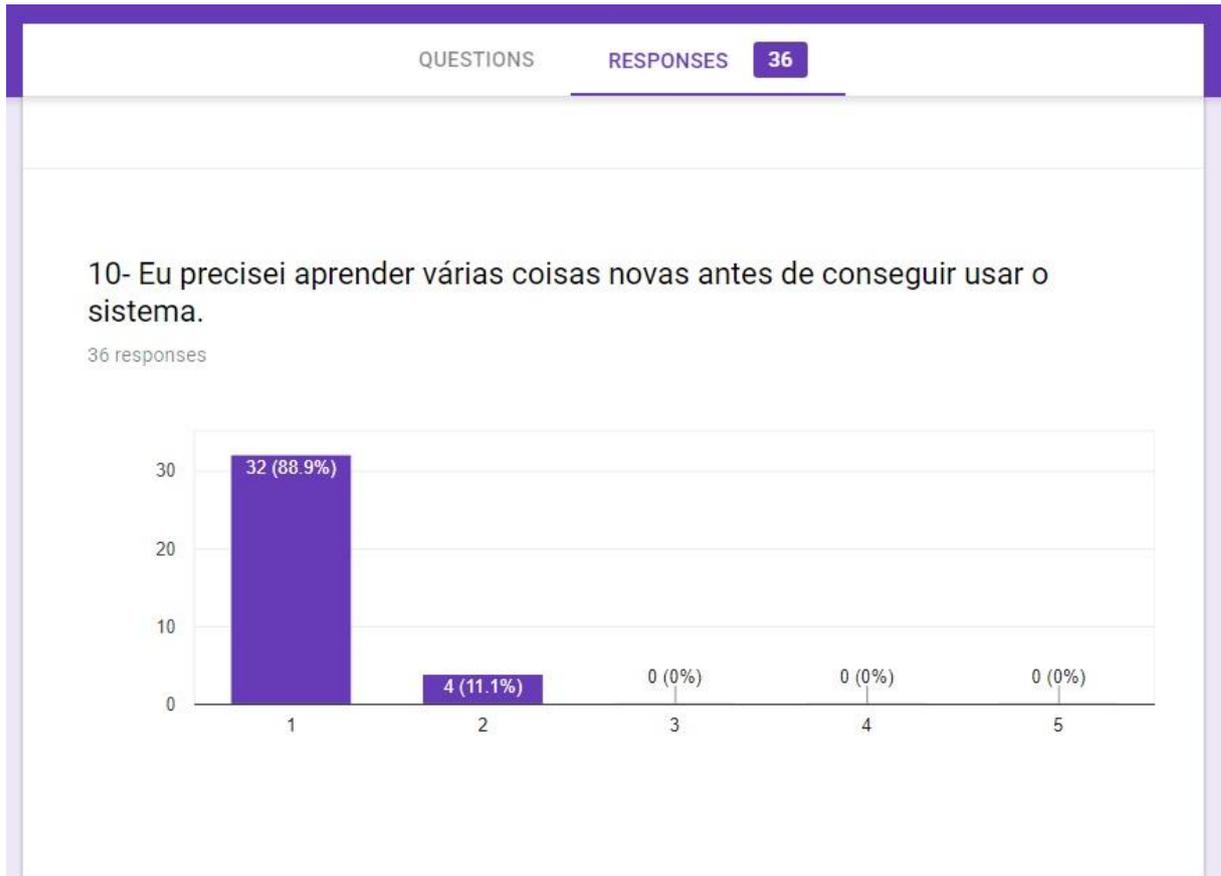


Figura 19 - Resultado em gráfico da Avaliação Experimental

Após finalização das avaliações, as notas foram distribuídas conforme a Figura 20 para melhor análise do resultado. Dessa maneira, foi possível observar que os usuários sem experiência em escalas deram notas melhores, uma vez que focaram a avaliação na interface do sistema, não exigindo das funcionalidades de distribuição de folgas. Já a avaliação do segundo grupo de usuários, além da avaliação da interface buscaram avaliar com mais atenção a distribuição das folgas, pois consideraram essa funcionalidade mais importante para eles. A avaliação mais rigorosa ficou a cargo da única pessoa com experiência em elaboração de escalas, tanto que foi a avaliação com a menor nota, pois além de avaliar a interface e a distribuição de folgas, considerou todas as outras funcionalidades propostas pelo SGET.

Dessa maneira, é possível observar na Figura 20 que as notas tendem a diminuir à medida que as posições avançam, visto a distribuição de posições atribuída aos usuários com mais experiência. Após conclusão da avaliação, as notas dos usuários foram agrupadas e calculadas conforme metodologia do SUS (Figura 20), sendo possível identificar que o SGET ficou acima da média do SUS de 68 pontos, atingindo 86 de pontuação. Esse resultado foi considerado satisfatório.

Participant	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	SUS Score
p1	4	1	4	5	4	1	4	1	4	1	77.5
p2	4	1	5	1	4	1	4	1	3	1	87.5
p3	3	2	4	1	3	1	4	1	4	1	80.0
p4	5	1	5	1	5	1	5	1	5	1	100.0
p5	4	1	5	2	4	1	4	2	5	1	87.5
p6	4	2	5	1	3	1	5	1	4	1	87.5
p7	4	2	5	1	3	2	5	2	4	1	82.5
p8	5	1	5	1	5	1	5	1	5	1	100.0
p9	4	2	4	2	4	2	4	2	4	1	77.5
p10	4	2	4	1	4	2	4	1	4	1	82.5
p11	3	1	5	1	3	1	5	1	3	1	85.0
p12	4	1	5	1	3	2	4	1	3	1	82.5
p13	5	1	5	1	4	2	4	2	4	1	87.5
p14	5	2	4	1	4	1	4	1	5	1	90.0
p15	5	2	5	2	4	1	5	2	4	1	87.5
p16	5	1	5	1	4	1	4	1	4	1	92.5
p17	4	1	4	1	3	2	4	1	4	1	82.5
p18	4	1	5	1	4	2	5	2	5	1	90.0
p19	5	1	5	1	4	2	5	2	4	1	90.0
p20	4	2	5	1	3	2	5	1	4	1	85.0
p21	4	2	5	1	4	2	5	2	4	1	85.0
p22	4	2	5	1	4	2	4	2	4	1	82.5
p23	5	2	5	1	3	2	5	1	5	1	90.0
p24	5	2	4	1	4	2	4	1	5	1	87.5
p25	5	1	5	1	4	1	4	1	5	1	95.0
p26	5	1	5	1	4	1	5	1	4	1	95.0
p27	5	2	5	1	4	1	3	2	5	2	85.0
p28	4	1	5	1	4	2	4	2	5	1	87.5
p29	4	1	5	1	4	2	4	2	5	1	87.5
p30	5	1	4	1	4	3	4	2	4	2	80.0
p31	5	1	4	1	5	2	5	2	5	1	92.5
p32	4	2	5	1	4	2	4	2	5	1	85.0
p33	4	1	4	2	3	2	4	2	4	2	75.0
p34	3	2	4	1	3	1	4	1	4	1	80.0
p35	4	2	4	1	3	2	4	1	4	1	80.0
p36	4	2	5	2	3	2	4	2	4	2	75.0
										Total	86.0

Figura 20 - Resultado da Avaliação Experimental

5 Conclusão

O SGET foi desenvolvido com a principal finalidade de elaborar escalas de trabalho para equipes com necessidades distintas. Para garantir isso, idealizou-se as seguintes funcionalidades para o sistema: Interface amigável e de fácil usabilidade permitindo consultas *on-line* à escala; distribuição e validação das folgas automaticamente, considerando feriados, leis trabalhistas e critérios pré-definidos; redução do tempo de interação humana para elaboração da escala, exigindo poucos passos para conclusão da escala. Porém, somente alguns objetivos foram totalmente atendidos.

Com a utilização da linguagem PHP, foi possível criar uma interface intuitiva e de fácil manuseio para o SGET. Dessa forma, o sistema não exige qualquer tipo de treinamento para utilização de seus recursos, garantindo rápida adaptação e maior agilidade na execução de tarefas. Além disso, o sistema é de fácil manutenção, pois foi utilizada a linguagem PHP.

Com a utilização de poucos perfis, e com características bem definidas, o sistema permite para cada um desses perfis execução das tarefas principais seguindo apenas alguns passos. Cabe ao Administrador realizar o cadastro de novos funcionários e designá-los aos seus respectivos times de atuação. Também cabe a ele o cadastro de feriados e período de férias, além das demais ações pertinentes aos demais usuários. Ao Gestor, cabe a definição do time que comporá a escala e a definição dos critérios de distribuição de folga que serão adotados. O sistema disponibiliza para o Gestor uma tela para cada uma dessas ações, que após concluídas realiza, automaticamente, a distribuição das folgas dos funcionários. Para estes usuários, é disponibilizada uma única tela, na qual é possível escolher a escala desejada para visualização de todos os funcionários envolvidos e suas respectivas folgas.

Para o SGET, não foi possível o desenvolvimento de algoritmo matemático capaz de interpretar as condições limitantes pela CLT, consolidar os critérios definidos pelo Gestor e distribuir de forma aleatória as folgas, garantindo que todos os critérios fossem atendidos. Devido à complexidade de desenvolvimento, optou-se por permitir apenas a definição da quantidade mínima de pessoas necessárias na escala e o limite de folgas para garantir que o SGET faça a distribuição das folgas de forma aleatória, porém sem garantia de atendimento às leis trabalhistas.

Ainda devido à complexidade, não foi possível aplicar ao SGET a funcionalidade que permite a separação de horário/turno em uma única escala. Assim, o sistema somente cria

escala considerando que os funcionários estão trabalhando no mesmo horário, independente do turno de atuação desses.

Mesmo com atendimento parcial dos objetivos propostos no início deste documento, o SGET executa suas funcionalidades sem depender de qualquer outro sistema externo, sendo projetado para abranger todos os seguimentos empresariais com necessidade de escalação de pessoal. Garantindo redução no tempo de elaboração e validação das folgas em detrimento às escalas elaboradas manualmente a partir de planilhas.

Como no mercado há poucos *softwares* para a elaboração de escalas voltados para empresas de telecomunicações, o SGET foi finalizado com a certeza de redução de tempo de execução, com a divulgação *on-line* da escala e garantindo que os critérios definidos pelo Gestor serão atendidos. Além disso, o sistema não exige treinamento para ser utilizado, garantindo assim, utilização imediata após aquisição.

Após conclusão da avaliação experimental, observou-se que ainda há muito a fazer para garantir que o SGET seja um sistema robusto e capaz de atender todos os objetivos propostos nesse documento. Portanto, vale garantir a explanação de todas essas ações, para que no futuro elas sejam executadas. São elas: integração com outros sistemas, adaptar a interface do sistema com recursos de acessibilidade, algoritmo matemático pra distribuição e validação de folga, separação dos horários e atividades dos funcionários em uma mesma escala, permitir que o Gestor tenha autonomia para alterar o horário, a atividade e/ou a folga do funcionário em dia específico após elaboração da escala, bem como permitir a extração de relatórios.

A integração com outros sistemas permitirá que o SGET seja utilizado não só para acompanhamento de assiduidade do funcionário, mas também garantir que o que foi planejado está sendo cumprido, se a integração ocorrer com um sistema de ponto, por exemplo.

Cada vez mais as empresas de telecomunicações se preocupam em contratar pessoas com deficiência, de maneira que em vários setores é possível encontrar pessoas com alguma limitação. Pensando na inclusão dessas pessoas, identificou-se a necessidade de implementação de recursos de acessibilidade, Assim, será importante a adaptação da interface do sistema com recursos de acessibilidade, garantindo assim maior abrangência do sistema nas empresas.

Como já mencionado anteriormente, é fundamental que seja desenvolvido um algoritmo matemático capaz de realizar a distribuição e validação de folgas de uma escala

considerando os critérios estabelecidos pelo Gestor e, prioritariamente, todas as regras estabelecidas pela CLT referente à escala de revezamento.

Com foco em atingir o maior número de equipes de uma empresa, o SGET deverá permitir que haja escalas com separação de horários e/ou atividades, e que os funcionários possam ser distribuídos considerando essas peculiaridades, de acordo com o planejamento do Gestor. Uma vez que o SGET esteja apto para separação de horário/atividade na mesma escala, para que ele garanta maior flexibilidade, o SGET deverá permitir que o Gestor tenha autonomia para alterar o horário, a atividade e/ou a folga do funcionário em dia específico após elaboração da escala. Mas que só permita que essas alterações sejam realizadas se os critérios definidos pelo Gestor e as leis trabalhistas sejam mantidos.

Por fim, para permitir maior controle sobre alterações de folgas, horários, atividades e férias, deverão ser inseridos campos apropriados para extração de relatórios e métricas. Com isso, o SGET também permitirá que seja avaliada se a distribuição de folgas adotada pelo SGET está atendendo as necessidades da equipe.

Com base em tudo que foi exposto nesse capítulo, fica claro que o SGET atente parcialmente as funcionalidades idealizadas, mas que mesmo assim é capaz de garantir maior visibilidade da escala e redução no tempo de elaboração de escala. Portanto, o SGET alcançou a satisfação desejada na avaliação experimental mostrando que, mesmo com suas limitações, é um sistema promissor com possibilidade de alcançar o mercado após implementações das melhorias apresentadas.

REFERÊNCIAS

- ADRIEL, Wendell. Introdução ao Laravel Framework PHP. 2017. Disponível em <<http://www.devmedia.com.br/introducao-ao-laravel-framework-php/33173>>. Acesso em: 7 ago. 2017.
- ANATEL. Lei Geral de Telecomunicações – Lei nº 9.472 Art.3º, de 16 de julho de 1997. Disponível em <<http://legislacao.anatel.gov.br/leis/2-lei-9472>>. Acesso em: 18 out. 2014.
- BARBOSA, Gilka Rocha; ALMEIDA, Adiel Teixeira de; COSTA, Ana Paula Cabral Seixas. SAD: Análise da percepção de usuários e desenvolvedores através de análise fatorial. Produção, v. 16, n. 2, p. 216-228, Maio/Ago. 2006. Disponível em <<http://www.scielo.br/pdf/prod/v16n2/03.pdf>>. Acessos em 28 set.2014.
- CLERICUZI, Adriana Zenaide; ALMEIDA, Adiel Teixeira de; COSTA, Ana Paula Cabral Seixas. Aspectos relevantes dos SAD nas organizações: um estudo exploratório. Prod., São Paulo, v. 16, n. 1, abr. 2006. Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-65132006000100002&lng=pt&nrm=iso>. Acesso em: 05 out. 2014.
- CLT. Decreto de lei Nº 5.452, Art. 230, de 01 de maio de 1943. Disponível em <http://www.planalto.gov.br/ccivil_03/decreto-lei/del5452.htm>. Acesso em: 18 out. 2014.
- CORRAR, Luis João; GARCIA, Editinete A. da Rocha. Programação linear: uma aplicação à contabilidade de custos no processo de tomada de decisão. In: CONGRESSO INTERNACIONAL DE CUSTOS, jul. 2001, León. Disponível em <<http://www.intercostos.org/documentos/Trabajo066.pdf>>. Acesso em: 09 nov.2014
- DILLON, S. M.(1998, agosto). Descriptive decision making: comparing theory with practice. *Anais do Annual Operational Research Society of New Zealand Conference*, Auckland, New Zealand, 33. Disponível em <<https://orsnz.org.nz/conf33/papers/p61.pdf>>. Acesso em: 05 out. 2014.
- ESCALA FÁCIL. Desenvolvido pela Mega Arts Ltda, 2012. Software Gerador de Escala de Trabalho e Apoio a Folha de Pagamento. Disponível em <<http://www.escalafacil.com.br/>>. Acesso em: 30 set.2014.
- HANSSON, Sven Ove. Decision Theory A Brief Introduction. Royal Institute of Technology, Stockholm, 1994. Disponpivel em <<http://people.kth.se/~soh/decisiontheory.pdf>>. Acesso em: 02mai.2017.
- LOPES, S. O que é PHP. 2007. Disponível em <https://www.oficinadanet.com.br/artigo/659/o_que_e_php>. Acesso em: 7 ago. 2017.
- PONTOVIT. Desenvolvido pela Vitória Humana Sistemas, 2004. Software Gerador de Escala de Trabalho. Disponível em <<http://www.pontovit.com.br/>>. Acesso em: 02 mai.2017.
- RANGEL, Alexandre Leite; ÉVORA, Yolanda Dora Martinez. Elaboração automática da

escala periódica de trabalho dos profissionais de enfermagem por meio de um software específico. Revista Eletrônica de Enfermagem [serial on line] 2007 Jan-Abr; 9(1): 17-30. Disponível em <<http://www.fen.ufg.br/revista/v9/n1/v9n1a02.htm>>. Acesso em: 08 nov. 2014.

REIS, Eliete dos; LOBLER, Mauri Leodir. O processo decisório descrito pelo indivíduo e representado nos sistemas de apoio à decisão. **Rev. adm. contemp.**, Curitiba , v. 16,n. 3,jun. 2012 . Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1415-65552012000300005&lng=pt&nrm=iso>. Acesso em: 05 out. 2014.

REVEX. Software Gerador de Escala de Revezamento. Desenvolvido pelo Professor Dorirley Rodrigo Alves, 2012. Software Gerador de Escala de Trabalho. Disponível em <<http://escaladerevezamento.com.br/>>. Acesso em: 04 mai.2017.

RICARDO, José. Introdução ao MySQL. 2017. Disponível em <<http://www.devmedia.com.br/introducao-ao-mysql/27799>>. Acesso em: 7 ago. 2017.

SATO, Kléber. A arquitetura MVC no desenvolvimento PHP. 2017. Disponível em <<http://www.devmedia.com.br/a-arquitetura-mvc-no-desenvolvimento-em-php/23121>>. Acesso em: 7 ago. 2017.

SCHMIDT, Alice Gerhardt; PAIM, Eliane Suely Everling; PIENIZ, Luciana Paim; MARETH,Taciana. Contribuição da programação linear na análise de viabilidade: O caso da indústria de confecções de uniformes. Revista de Negócios, n.5, mar.2008. Disponível em <<http://www.unifin.com.br/Content/arquivos/20111006160641.pdf>>. Acesso em: 09 nov.2014.

TEIXEIRA, Fabrício O que é o SUS (System Usability Scale) e como usá-lo em seu site. 2015. Disponível em <<https://brasil.uxdesign.cc/o-que-%C3%A9-o-sus-system-usability-scale-e-como-us%C3%A1-lo-em-seu-site-6d63224481c8f>>. Acesso em: 10ago.2017.