

## SKALLA – SISTEMA DE GESTÃO DE ESCALAS DE TRABALHO

Leonardo Queiroga Ramirez<sup>1</sup>, Maickel Pirchiner da Costa<sup>2</sup>

Instituto Federal do Paraná

**Resumo:** *O presente artigo tem como objetivo abordar o desenvolvimento do sistema de gerenciamento de escalas de trabalho, SkAlla. Tendo em vista que muitos processos trabalhistas têm como tema a parte de horas extras, muitas horas ininterruptas, cansando assim o colaborador e afetando a empresa como um todo, o SkAlla vem apresentar uma maneira fácil de cálculo de horas e gerenciamento de escalas de trabalho. Foi desenvolvido visando primariamente o trabalho no desktop e na internet, utiliza a linguagem Python com a ferramenta Django Framework, persistência através do PostgreSQL e interface com Bootstrap e VueJS. O sistema possui gerenciamento de clientes, colaboradores, locais de trabalho, turnos e criação de escalas com notificações e solicitações de alteração ou cancelamento por parte dos colaboradores, assim como também relatórios de escalas. Os resultados alcançados até a conclusão do presente artigo são satisfatórios, tendo como referência a expectativa inicial, viabilizando a implementação total e comercialização do sistema.*

**Palavras-chave:** Sistema web. Gestão de Escalas Trabalhistas. Horas Trabalhadas. Saúde do Trabalhador.

**Abstract:** *This article aims to address the development of the SkAlla workspace management system. Given that many labor processes are about overtime, uninterrupted hours, tiring the employee and affecting the company as a whole, SkAlla provides an easy way to calculate hours and work schedule management. It was developed primarily for desktop or internet work, uses Python language with the Django Framework tool, persistence through PostgreSQL and interface with Bootstrap and VueJS. It has clients, employees, workplaces, shifts management, and supports requests for changes or cancellation by employees as well as reports on schedules. The results achieved at this article's conclusion are satisfactory based on the initial expectations, enabling the implementation and commercialization of the system.*

**Keywords:** Web system. Labor Schedule Management. Worked hours. Worker's health.

---

1 mandaproleo@gmail.com

2 maickelpc@gmail.com

## 1 Introdução

Um grande número de empresas adotam o chamado horário comercial de atendimento, com jornadas de trabalho que giram em torno de oito horas diárias, e não ultrapassando quarenta e quatro horas semanais, atendendo a CLT [BRASIL, 1943, s. II].

No entanto, existem empresas de distintos ramos, tais como: fábricas, suporte a sistemas, atendimento de chamados de sistemas, energia elétrica, telecomunicações, empresas do ramo de segurança, limpeza, pronto socorro, cuja jornada de trabalho não se adequa à jornada de trabalho padrão, de forma que precisam manter turnos diferenciados, funcionando até mesmo 24 horas.

Visando atender esses cenários, a CLT apresenta alguns outros formatos de jornada de trabalho e descanso, tais como a jornada de 4x2, 5x1, 6x1, e 5x2, que relaciona os dias trabalhados com os dias de folga, e as escalas 12x36 e 24x48 que relaciona as horas trabalhadas com as horas de descanso [REDE JORNAL CONTÁBIL, 2019].

Entretanto, a gestão das escalas acaba por se tornar complexa para essas organizações, em função dos cálculos para computar a quantidade de horas trabalhadas por colaborador, gestão de recursos humanos e etc., de forma que manter uma escala de trabalho, sem a utilização de uma ferramenta adequada, torna-se uma tarefa não trivial. Isso fica evidente ao observarmos o tema "horas extras", figurando em primeiro lugar no *ranking* dos assuntos nos processos no TST (Tribunal Superior do Trabalho), ao ponto que o intervalo intrajornada seja o terceiro neste mesmo *ranking*. [TST.JUS.BR, 2019].

Diante do exposto, este trabalho tem como objetivo principal a implementação de um sistema em uma plataforma *web*, que possibilite às empresas a gestão simplificada de suas escalas de trabalho, permitindo consultar a quantidade de horas trabalhadas pelos colaboradores, e alertar quando houver alguma sobrecarga de algum colaborador, para ajudar a evitar problemas relacionados a essas questões.

O sistema utilizará a linguagem Python com a ferramenta Django Framework, POSTGRESQL para a persistência e VueJS para responsividade da interface. O presente

trabalho apresenta as escalas de trabalho em relação às tecnologias de apoio, metodologias de desenvolvimento, escalas de trabalho e modelagem e considerações finais.

## 2 Escalas de Trabalho e Tecnologias de Apoio

O estado da saúde do colaborador pode também ser medido por meio da qualidade do seu sono. Nesse sentido, pesquisadores aplicaram questionários em colaboradores de empresas que atuam em ambos os regimes de trabalho. Tendo isso em vista, os resultados foram que os colaboradores que trabalham em turnos, especialmente noturnos, apresentam fadiga no trabalho e qualidade baixa de sono, afetando assim o desempenho no trabalho, e principalmente sua saúde. Foram encontrados problemas de ordem principalmente físicas no que tange a parte gastrointestinal e cardiovascular [ÅKERSTEDT, 1988, VENER ET AL. 1989].

Outro problema que o mal gerenciamento de escalas pode trazer é o jurídico. Horários estendidos de trabalho, assim como pouco descanso entre turnos figuram entre os principais tópicos dos tribunais do trabalho do país [TST.JUS.BR, 2019].

Segundo um estudo realizado com enfermeiros e enfermeiros docentes em um hospital universitário, um sistema de gerenciamento de escalas específico obteve resultado satisfatório em 87% dos enfermeiros e 98% dos enfermeiros docentes pesquisados [RANGEL, A. L. *et al*, 2012]. Tendo isso em vista, um sistema de gerenciamento de escalas que atenda grande parte do público pode ser uma maneira efetiva de diminuir os problemas descritos.

### 2.1 Tecnologia

Buscando tornar a ferramenta acessível e independente de plataforma, optou-se por desenvolver um sistema *web*. Foram avaliadas as principais tecnologias disponíveis no mercado, e optou-se pelo Django, um *framework* para criação de aplicações *web*, escrito na linguagem Python. A opção se dá pelo fato de o Python ser a linguagem que apresenta o maior crescimento do mercado nos últimos anos [ROBINSON, 2017], e o Django ser o maior *framework* que utiliza essa linguagem, focado em desenvolvimento rápido, limpo e com *design* pragmático, além de ser utilizado em grandes sistemas/portais como Globo.com, Pinterest, Instagram, entre outros [HIDEKI, 2013].

O banco de dados de modelo relacional utilizado no projeto é o Postgresql. A opção se dá por vários fatores, como a licença GNU [POSTGRESQL - LICENSE, POSTGRESQL, c1996-

2019], que oferece permissão total, para qualquer um copiar, utilizar e distribuir [GNU OPERATING SYSTEM, 2007]. Além disso, oferece integridade referencial, recurso de *schema*, *procedures* e *triggers*, que possibilita automatizar algumas funções, controle transacional, que auxilia em operações complexas que envolvem alterações em mais de um registro. Possui um controle MVCC (*Multiversion Concurrency Control* – Controle de Concorrência Multiversões) que trata os *locks* de forma inteligente, evitando conflito de operações entre usuários [POSTGRESQL – ABOUT, POSTGRESQL, c1996-2019].

Para uma tela mais amigável e intuitiva, o projeto utiliza o *framework* de *frontend* VueJS, que tem a característica de ser reativo, ou seja, modelos de dados são objetos javascript puros. Quando você os modifica, a camada visual se atualiza, reagindo às alterações do modelo. É voltado à criação de interfaces para o usuário. Por ter essa característica, fica mais intuitivo o desenvolvimento, e também aproxima a criação das *views* do paradigma de orientação a objetos [VUE.JS, c2019].

### **3 Metodologia**

A percepção da necessidade de um sistema que gerencie exclusivamente escalas de trabalho surgiu após percepção de problemas jurídicos de empresas com regimes diferenciados. Após a constatação inicial, iniciou-se uma revisão de literatura para que se conheça o estado da arte do tópico em questão.

Após a revisão de literatura iniciou-se o processo geral, com um EAP<sup>3</sup> baseado em fases do RUP<sup>4</sup>. As fases utilizadas foram divididas em: Planejamento, Análise (Levantamento de Requisitos), Modelagem, Implementação, Testes, Entrega e Aceitação.

#### **3.1 Requisitos**

Segundo [SOMMERVILLE, 2007, p. 73], os requisitos do sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento. Eles devem refletir a necessidade do cliente para um sistema que serve a uma finalidade determinada.

##### **3.1.1 Requisitos funcionais**

Sommerville classifica os requisitos funcionais como declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve

---

3 Estrutura Analítica do Processo

4 *Rational Unified Process* (Processo Unificado da Rational)

se comportar em determinadas situações. Em alguns casos, os requisitos funcionais também podem explicitar o que o sistema não deve fazer. [SOMMERVILLE, 2007, p. 75]

<b>Referência</b>	<b>Requisito</b>
RF 01	O sistema deve manter País com um nome e uma sigla, assim como Estado com nome, sigla e país, e Cidade com um nome, código do IBGE e um estado.
RF 02	O sistema deve manter Empresas com um nome, nome fantasia, CNPJ, IE, uma imagem de logotipo, um endereço e a data de abertura.
RF 03	O sistema deve manter Clientes com nome, nome fantasia, cnpj, rg, endereço, um logotipo em imagem, um contato, um contato financeiro, telefone, e-mail e uma inscrição municipal.
RF 04	O sistema deve manter Usuários com um nome, sobrenome, e-mail, senha e status se está ativo ou não.
RF 05	O sistema deve manter Colaboradores que além dos dados que herdam de Usuário, possuem data de admissão, de nascimento, número de telefone, número de celular, um contato de emergência e seu telefone, número do PIS, se está ativo, horas limite de trabalho no mês e limite semanal de horas.
RF 06	O sistema deve manter Áreas das Empresas possuindo uma empresa, uma descrição e um colaborador responsável.
RF 07	O sistema deve manter Departamentos das Áreas com uma área, um nome e um responsável.
RF 08	O sistema deve manter Pontos de Alocação possuindo nome, descrição e um endereço.
RF 09	O sistema deve manter informações de Turnos possuindo descrição, período, horas trabalhadas e horas de descanso, hora de início e fim do turno.
RF 10	O sistema deve manter Perfis de Jornada possuindo

<b>Referência</b>	<b>Requisito</b>
	descrição, tipo e horas de antecedência antes de permitir duplicar.
RF 11	O sistema deve manter Escalas de Trabalho que possui um perfil de jornada, uma data de início e fim, um status e a data de duplicação, se houver.
RF 12	O sistema deve permitir a visualização da minha escala (do colaborador).
RF 13	O sistema deve permitir solicitações de alteração e cancelamento de escalas.
RF 14	O sistema deve imprimir uma escala geral da empresa.
RF 15	O sistema deve imprimir uma escala individual do colaborador.

### 3.1.2 Regras de Negócio

Segundo Dextra [2013], regra de negócio é o que define a forma de fazer o negócio, refletindo a política interna, o processo definido e/ou as regras básicas de conduta.

<b>Referência</b>	<b>RF</b>	<b>Regra</b>
RN 01	RF 11	A escala pode ser salva em um estado provisório para ser terminada depois, não necessitando ter todos os colaboradores para ser persistida.
RN 02	RF 11	Nenhuma escala de colaborador poderá repetir o colaborador em uma mesma data/hora.
RN 03	RF 11	Cancelamento ou inserção de colaborador em escalas fora do prazo máximo configurado na empresa do colaborador em questão, só poderão ser executadas por um usuário com perfil Administrador
RN 04	RF 12	O sistema apresentará somente as informações relacionadas ao usuário logado.
RN 05	RF 13	Somente poderão ser solicitadas alterações em escalas futuras, e que estejam dentro do prazo mínimo registrado na empresa a qual o colaborador faz parte.

#### 4 Modelagem e as Escalas de Trabalho

O projeto de sistema em plataforma *web* é voltado à facilitação da criação e manutenção das escalas de trabalho, sem a pretensão de obrigar o usuário a atender todas as nuances da CLT, nem fazer encontro de informações com cartão ponto para verificação da fiel execução das escalas criadas.

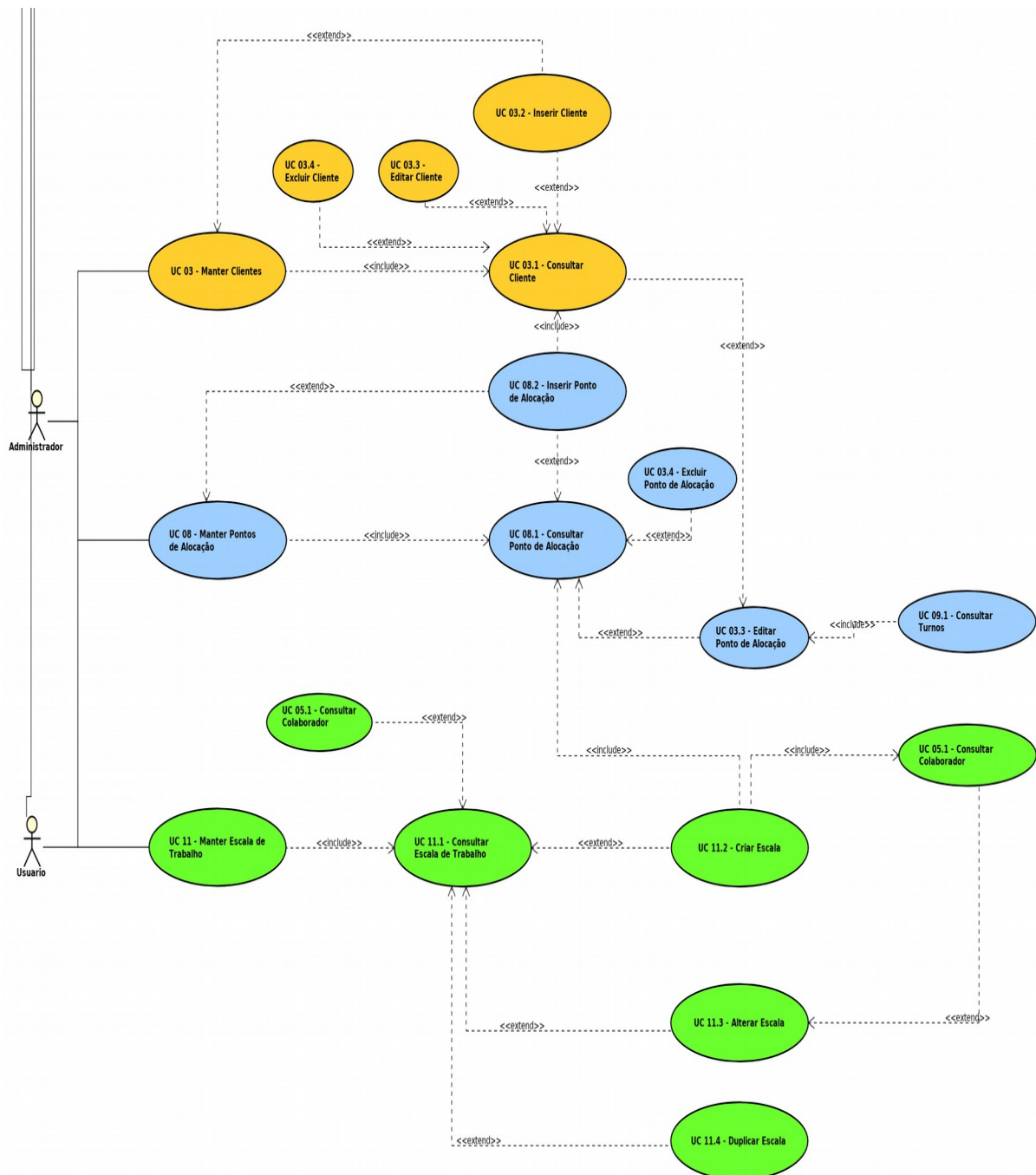


Figura 1: Casos de Uso Complexos

O diagrama de casos de uso complexo, conforme Figura 1, procura mostrar os casos de uso e seus atores, bem como as funções disponíveis em cada caso.

Ainda conforme a Figura 1, o usuário poderá registrar os clientes que serão atendidos, bem como seus vários pontos de alocação com seus turnos e quantidade mínima de colaboradores para cada turno. O escopo prevê também a gestão de várias empresas, suas áreas, departamentos e colaboradores.

A criação de escalas é o ponto onde o usuário vai criar as escalas e ter uma fácil visualização das informações dos pontos de alocação e turno, com o perfil da jornada, mínimo de colaboradores, bem como as horas trabalhadas/previstas para o colaborador na semana e no mês, a fim de evitar sobrecarga. Caso o usuário tente inserir um usuário com horas em sobrecarga na semana e/ou no mês, o sistema vai exibir um alerta, chamando atenção do usuário para tal ação.

Após a criação das escalas, as mesmas ficam disponíveis para a gestão que pode cancelar escalas inteiras antes da mesma se iniciar, e também pode inserir/remover colaboradores, sempre em escalas futuras. O colaborador também poderá solicitar alteração em suas escalas futuras, respeitando o prazo máximo de antecedência de cada empresa. As solicitações, bem como o retorno dessas ficam registrados, para consultas futuras.

No diagrama de classes conceitual, conforme Figura 2 são mostradas as classes e os dados referentes a ela, descrevendo toda a estrutura do sistema.



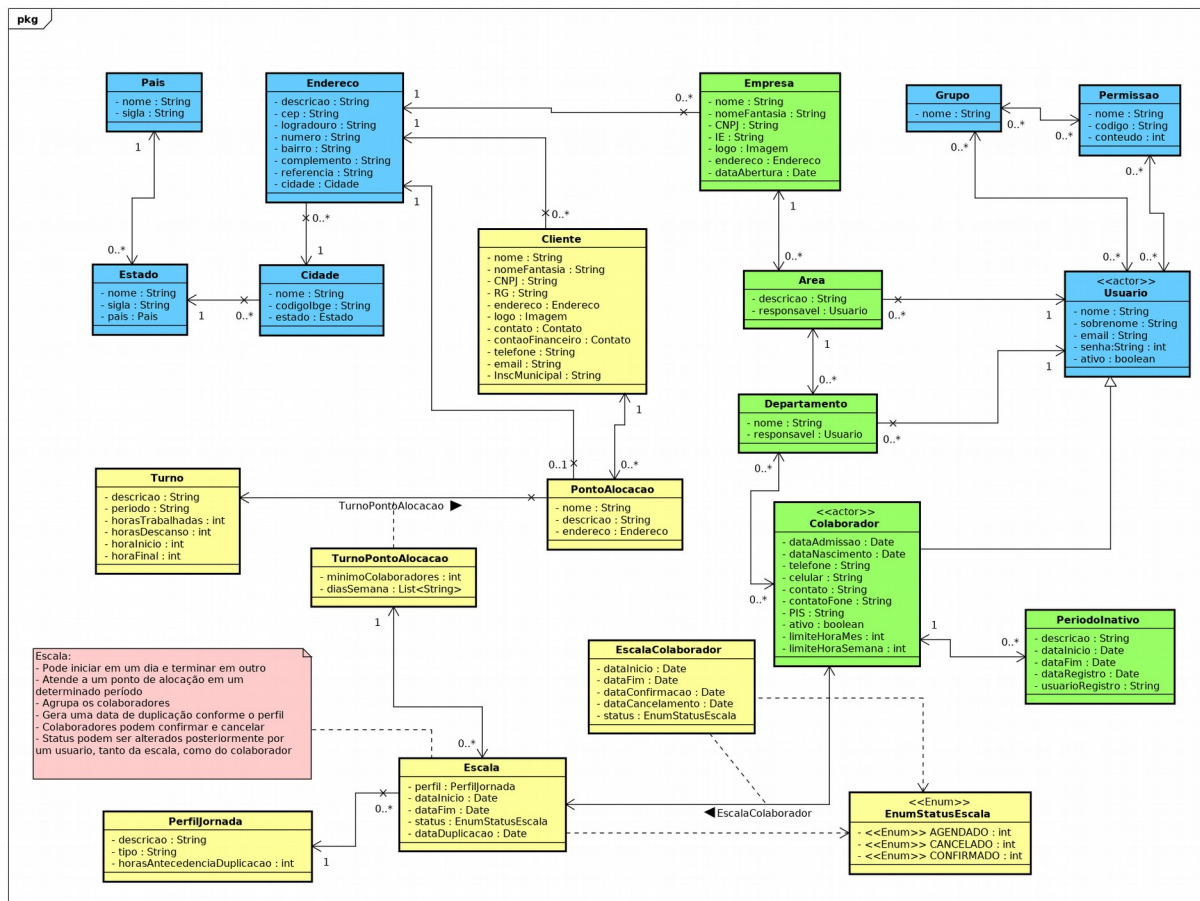


Figura 2: Diagrama de Classes Conceitual

Relatórios para operação como escala de um colaborador, escala de um ponto de alocação, e escalas de um cliente específico são recursos que o sistema oferece e como eles podem ser acessados pelos atores está conforme Figura 3.

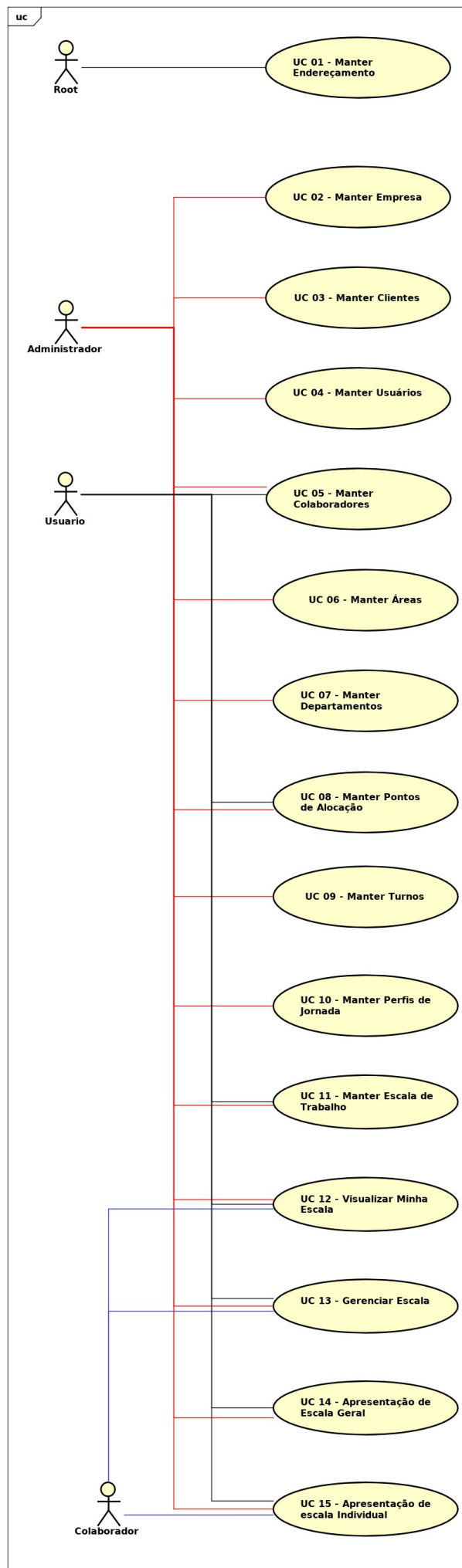


Figura 3: Casos de Uso Geral

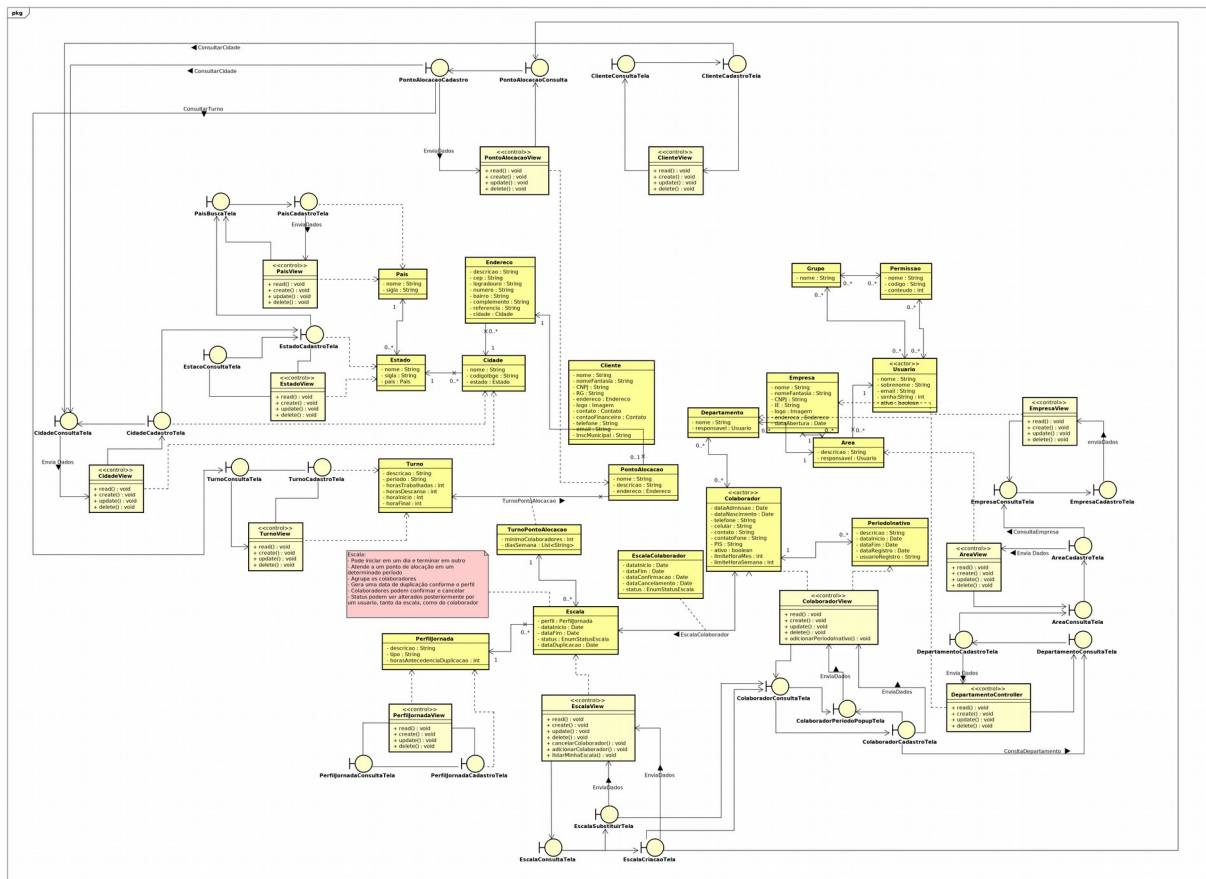


Figura 4: Diagrama de Classes - Implementação

Na Figura 4 está o diagrama de classes de implementação que possui além dos atributos, a navegabilidade e interfaces do sistema.

## 5 Considerações Finais

Conforme visto no início deste artigo, é custoso para uma empresa o erro na escala de trabalho, seja em horas excedentes, ou em poucas horas de descanso. Isso onera financeiramente a empresa e os clientes, perdendo tempo com processos legais.

Nos testes efetuados identificou-se que é possível melhorar o processo das escalas de trabalho nas empresas por meio do uso do sistema. As ferramentas de controle da escala ajudam a diminuir ocorrências de trabalho estendido e melhoram a gerência de horas dos colaboradores. Isso ajuda a diminuir ocorrências de processos contra a empresa com o assunto de quantidade de horas trabalhadas e horas intrajornada.

Nota-se também o potencial comercial do sistema, podendo ser oferecidos a empresas de várias áreas como segurança, saúde, indústrias, sendo necessárias poucas adaptações para a realidade de cada cliente.

Constatou-se que o período disponibilizado para o desenvolvimento do sistema não foi suficiente para o desenvolvimento de todas as funções almejadas ao início do projeto, tendo-se que deixar partes para o desenvolvimento futuro, como alguns relatórios, notificações e confirmações via aplicativo por exemplo.

Pretende-se também, aprimorar a interface e usabilidade em todo o sistema, de forma que melhore a experiência e a agilidade dos processos realizados.

## **6 Referências bibliográficas**

ÅKERSTEDT, Torbjörn, 1988. **Sleepiness as a consequence of shift work.** *Sleep*, 11:17-34.

BRASIL. Decreto-lei nº 5.452, de 1º de Maio de 1943. **CONSOLIDAÇÃO DAS LEIS DO TRABALHO.** Brasília, DF, Maio de 1943. Disponível em <[http://www.planalto.gov.br/ccivil\\_03/decreto-lei/del5452.htm](http://www.planalto.gov.br/ccivil_03/decreto-lei/del5452.htm)>. Acesso em 04 nov. de 2019.

REDE JORNAL CONTÁBIL. **CLT: Conheça todos os tipos de escalas de trabalho permitidas.** 2019. Disponível em <<https://www.jornalcontabil.com.br/clt-conheca-todos-os-tipos-de-escalas-de-trabalho-permitidas>>. Acesso em 04 Nov. 2019.

DEXTRA. **REQUISITO OU REGRA DE NEGÓCIO?** (05 de Nov. de 2013) <<https://dextra.com.br/pt/requisito-ou-regra-de-negocio/>> Acesso em 11 Nov. 2019.

GNU GENERAL PUBLIC LICENSE. **GNU OPERATING SYSTEM.** V3. 29 Jun. 2007. Disponível em <<https://www.gnu.org/licenses/gpl-3.0.en.html>>. Acesso em 04 Nov. 2019.

HIDEKI, Erik. Django: Fontes de estudo e referências. **APRENDENDO PYTHON.** 08 de Mai. de 2013. Disponível em <<https://ericstk.wordpress.com/2013/05/08/django-fontes-de-estudo-e-referencias>>. Acesso em 04 Nov. 2019.

O QUE É VUE.JS?. **VUE.JS.** C2019. Disponível em: <<https://br.vuejs.org/v2/guide/index.html>>. Acesso em 04 Nov. 2019.

POSTGRESQL – LICENSE. **POSTGRESQL.ORG.** c1996-2019. Disponível em <<https://www.postgresql.org/about/licence>>. Acesso em 04 Nov. 2019.

POSTGRESQL – ABOUT. **POSTGRESQL.ORG.** c1996-2019. Disponível em <<https://www.postgresql.org/about>>. Acesso em 04 Nov. 2019.

RANGEL, A. L., ÉVORA, Y. D. M., OLIVEIRA, M. M. B., 2012. **O processo de avaliação do software de geração automática de escala de trabalho da enfermagem e da escala por ele gerada.** Journal of Health Informatics. 04 Dez. 2012. 4:5

ROBINSON, David. The Incredible Growth of Python. **STACK OVERFLOW.** 06 de Set. de 2017. Disponível em <<https://stackoverflow.blog/2017/09/06/incredible-growth-python>>. Acesso em 04 Nov. 2019.

SOMMERVILLE, Ian. **Engenharia de Software.** 9ª ed. São Paulo: Pearson Addison-Wesley, 2007.

TRIBUNAL SUPERIOR DO TRABALHO. **TST.JUS.BR.** Ranking dos Assuntos mais Recorrentes. c2019. Disponível em <<http://www.tst.jus.br/web/estatistica/tst/assuntos-mais-recorrentes>>. Acesso em 04 Nov. 2019.

VENER, K. J.; SZABO, S. & MOORE, J. G., 1989. **The effect of shift work on gastrointestinal (GI) function: A review.** *Chronobiologia*, 18:421-439.