Resumo

**Resumo**

A cada dia é mais difícil conciliar os compromissos do dia a dia com escasso tempo disponível. As muitas responsabilidades – seja na área pessoal, seja na área profissional – exigem um planejamento a fim de evitar que saiam de controle.

Isso fez com que a organização se tornasse uma ferramenta imprescindível para aperfeiçoar a capacidade individual para o cumprimento dos objetivos.

David Allen, autor de uma das metodologias mais famosas e eficazes de gerenciamento pessoal, o GTD, defende a ideia de planejar o futuro e realizar o agora, abstendo-se ao máximo do stress do dia a dia devido à prévio planejamento e organização.

O objetivo desse projeto é auxiliar a administração e a produtividade pessoal através de uma agenda inteligente que gerencie desde os afazeres do dia a dia até as metas de longo prazo.

Índice

**Índice**

1. Introdução.................................................................................................3
2. Arquitetura.................................................................................................4
3. Tecnologia.................................................................................................5
4. Infraestrutura.............................................................................................6
5. Controle de Versão...................................................................................7
6. Cadastro....................................................................................................8
7. Protótipo....................................................................................................9
8. Requisitos Funcionais.............................................................................10
9. Requisitos Regulatórios..........................................................................11
10. Tarefas....................................................................................................13
    1. Tarefas Agendadas......................................................................14
    2. Tarefas a Agendar........................................................................15
    3. Split...............................................................................................16
    4. Remoção, Edição e Inplit..............................................................17
11. Alertas.....................................................................................................18
12. Metas......................................................................................................19
13. Estrutura de objetos relevantes..............................................................20
14. Bibliografias.............................................................................................21

Introdução

**Introdução**

O sistema basicamente consiste numa agenda inteligente que gere as tarefas, organizando-as, se cabível, de forma coerente a fim de contribuir para a performance e pormenorizar o stress causado por elas.

Também, alertas podem ser ativados, com ou sem vinculação a uma tarefa, para despreocupar o usuário, deixando-o assim livre para realizar as tarefas do dia a dia com dedicação e consequentemente melhor desempenho, como defende David Allen.

Além das tarefas e alertas, o projeto abrange as metas de longo prazo. O usuário pode, portanto, refletir sobre o seu dia a dia, analisar sua performance ou, se preciso for, planejar novamente suas metas a fim de manter seu foco.

O estabelecimento de perfis, os quais podem ser criados pelo próprio usuário, contribui e busca refletir as mais diversas faces da vida (ex. pessoal, profissional...), firmando, ao mesmo tempo, a segregação entre estas e o equilíbrio.

Também o APU (Análise de Perfil do Usuário) será outra ferramenta que colaborará com a inteligência do programa. Trata-se de um questionário em que o sistema se fundamentará para gerar a agenda do dia a dia.

A utilização do programa é viável tanto para o uso pessoal quanto corporativo.

Futuramente, o software deve suportar plataforma para dispositivos móveis e plug-ins como, por exemplo, finanças, saúde, trânsito etc, interagindo os afazeres do usuário com fatores externos que podem afetá-los, exibindo alertas ou dicas úteis.

O intuito é desenvolver um software com interface simples e com mensagens e alertas que serão exibidos de forma amigável e direta.

Arquitetura

**Arquitetura**

O software seguirá o padrão de projeto MVC Web porque, além do projeto de ser de caráter web, este modelo oferece uma estrutura de divisão em N camadas, facilitando o desacoplamento das classes.

A princípio, o projeto está estruturado em três camadas: interface, lógica e persistência; podendo futuramente ser dividido em mais camadas para diminuir o acoplamento e, por consequência, diminuir a complexidade de implementação.



Fazendo-se uma analogia entre a representação acima e o projeto se tem:

* View: corresponde à camada interface. Esta é a camada responsável por interagir com usuário e apresentar o conteúdo no browser.
* Controller: corresponde à camada lógica. Esta é a camada onde se encontram todas as classes responsáveis por receber as requisições enviadas pelo usuário e também toda a lógica da aplicação, divididas em camadas ou pacotes. Também, encontram-se as classes responsáveis por enviar o conteúdo a ser exibido para a interface.
* Model: corresponde à camada de persistência, onde se localizam as classes responsáveis por manipular todos os objetos a serem salvos no banco de dados. Esta camada também engloba as classes do Hibernate.

Tecnologia

**Tecnologia**

O software seguirá a tecnologia Java EE. Esta é uma plataforma voltada a desenvolvimento web que disponibiliza inúmeras ferramentas que facilitam a construção de aplicações. E juntamente a esta, as bibliotecas do Maven.

As bibliotecas do Maven trazem ferramentas para desenvolvimento de servelet, que tornam a aplicação muito mais ágil e eficiente, e proporciona melhor desempenho.

O software será desenvolvido segundo a IDE Eclipse. Além de termos certa familiaridade com ela, esta proporciona a possibilidade de serem integrados inúmeros plug-ins gráficos, que facilitam a construção da parte gráfica da aplicação. Esta IDE também facilita a integração com as demais ferramentas como, por exemplo, o Apache TomCat, que será usado para rodar a aplicação localmente a fim de teste.

Outra ferramenta importante será o Hibernate. Ele será responsável por receber os dados a serem inseridos na base de dados e também por captar esses dados e disponibilizá-los para a aplicação.

O banco de dados será o MySQL devido sua gratuidade e fácil manuseio.

Infraestrutura

**Infraestrutura**

Em relação à infraestrutura, pode-se dividir em duas partes. Uma seria o lado cliente, onde se têm as condições para que o cliente consiga utilizar aplicação. Outra, o lado servidor, onde se têm as condições para que aplicação seja hospedada.

* Servidor: A aplicação necessita que o servidor tenha instalado uma versão do Java igual ou superior à versão 6.0, juntamente Apache tomcat 6.0 ou superior. A aplicação também necessita de um servidor de banco de dados MySQL, que pode estar no mesmo servidor, porém se dá preferência à alocação em outro servidor devido a melhor performance e maior segurança.
* Cliente: Qualquer browser atualizado.

Controle de Versão

**Controle de Versão**

O controle de versão será feito através do software CVS, devido ao seu fácil manuseio e gratuidade, além de ser integrável a IDE usada para o desenvolvimento.

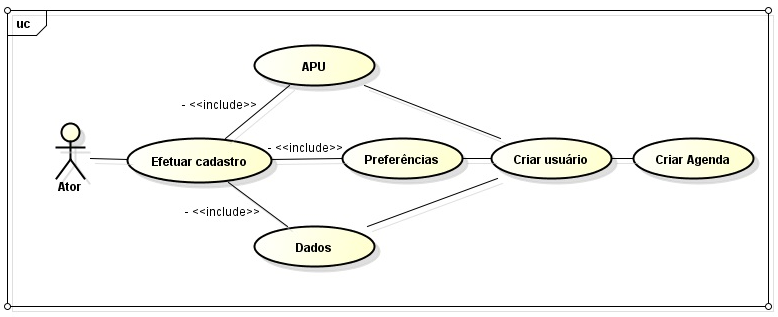
A motivação para usar o CVS é:

* Possibilitar que haja um histórico de códigos gerenciado pelo próprio software.
* Haver um controle de que e quem modificou o código.
* Facilidade em compartilhar os códigos.
* Possibilitar o trabalho paralelo e concorrente de vários desenvolvedores sobre o mesmo código, evitando que um sobrescreva o código de outro.

Cadastro

**Cadastro**

Para a utilização do sistema será preciso que o usuário realize previamente um cadastro. A responsabilidade desse cadastro é obter as informações de login e dados do cliente, obter a customização em relação ao uso das ferramentas disponibilizadas pelo sistema e, por fim, capturar informações relevantes para a composição da agenda inteligente.



* Dados: parte do cadastro responsável pela obtenção dos dados de login (user name e password), bem como informações gerais e informações para contato (e-mail, celular...).
* Preferências: basicamente consiste na customização por parte do usuário em relação às ferramentas oferecidas, entre elas criação de perfis e formas de alertas.
* APU: análise de perfil do usuário. A APU é um questionário que tem por finalidade a coleta de dados de metodologia de organização e horários reservados diariamente (por exemplo, refeições), ou seja, informações que contribuam para a inteligência da agenda.

Efetuado o cadastro, o sistema conceberá um novo usuário e atribuirá a ele uma nova agenda.

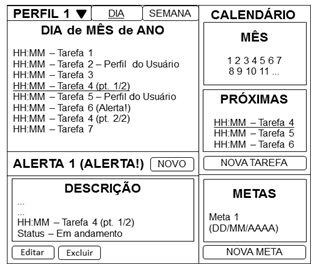
Protótipo

**Protótipo**

Na tela principal, o usuário poderá visualizar as suas tarefas já agendadas com seus respectivos horários. Se uma tarefa for selecionada, seja na visualização diária ou semanal, seja na visualização das próximas tarefas, a mesma terá sua descrição exibida no campo de tela “Descrição”. Alertas e metas são tratados da mesma forma.

A exibição das tarefas, alertas e metas está restrita ao perfil sobre a qual elas foram criadas, que agirá como um filtro exibição.

Através do calendário, o usuário poderá identificar os dias, de forma destacada, em que possui tarefa num determinado mês.



Basicamente essa tela consiste na exibição de tarefas por dia ou semana, próximas tarefas, alertas para o dia ou semana, metas, descrição e perfil selecionado.

Requisitos Funcionais

**Requisitos Funcionais**

1. Agendar tarefa.

2. Organizar e agendar tarefa.

3. Exibir tarefas agendadas.

4. Exibir descrição da tarefa.

5. Editar tarefa.

6. Excluir tarefa.

7. Cadastrar alerta.

8. Exibir alerta.

9. Exibir descrição de alerta.

10. Editar alerta.

11. Excluir alerta.

12. Cadastrar meta.

13. Exibir metas.

14. Exibir descrição de metas

15. Editar meta.

16. Excluir meta.

17. Cadastrar perfil.

18. Exibir Perfil.

19. Exibir agenda de perfil.

20. Excluir perfil.

21. Exibir calendário.

22. Cadastrar conta.

23. Editar conta.

24. Excluir conta.

25. Efetuar login.

Requisitos Regulatórios

**Requisitos Regulatórios**

1.1. A tarefa conterá os campos “Nome”, “Perfil”, “Prioridade”, “Complexidade”, “Motivação”, “Data”, “Duração”, “Divisível” e “Descrição”.

1.2. Os campos “Nome”, “Perfil”, “Prioridade”, “Complexidade”, “Motivação”, “Data”, “Duração” e “Divisível” são obrigatórios.

1.3. Os campos “Perfil”, “Prioridade”, “Complexidade” e “Motivação” serão exibidos como Combo Box, com um valor padrão predefinido.

1.4. Uma tarefa poderá ser vinculada a um alarme.

1.5. A tarefa somente será cadastrada mediante a disponibilidade.

2.1. A tarefa será organizada a partir do intervalo entre “De” e “Até”.

2.2. A tarefa será organizada mediante disponibilidade da agenda no intervalo dado.

2.3. O campo “Data” será atribuído ao fim da organização.

3.1. As tarefas serão exibidas por dia ou semana.

3.2. As próximas três tarefas, se existirem, serão exibidas.

5.1. A edição de uma tarefa remete aos itens “1” e “2”, se cabível, dos requisitos funcionais.

6.1. A exclusão de uma tarefa remete ao item “2”, se cabível, dos requisitos funcionais.

7.1. O alerta conterá os campos “Nome”, “Perfil”, “Data”, “Repetir” e “Descrição”.

7.2. Os campos “Nome”, “Perfil” e “Data” são obrigatórios.

7.3. Os campos “Perfil” e “Repetir” serão exibidos como Combo Box; “Repetir” conterá “Não” como padrão, além de semanal, mensal e anual.

10.1. Um alerta vinculado a uma tarefa, que foi editada, também será editado.

12.1. A meta conterá os campos “Nome”, “Perfil”, “Até” e “Descrição”.

12.2. “Nome” e “Perfil” são campos obrigatórios.

12.3. O campo perfil será exibido como Combo Box.

17.1. O perfil conterá somente o campo “Nome” e este deverá ser único.

17.2. Haverá o perfil padrão “Geral”.

19.1. A exibição da agenda ocorrerá segundo o perfil selecionado.

20.1. Ao excluir um perfil todas as tarefas pertencentes ao mesmo também serão excluídas.

20.2. O perfil “Geral” não poderá ser excluído.

21.1. O calendário exibirá mês a mês.

21.2. Os dias que possuem tarefa terão exibição destacada.

Requisitos Regulatórios

22.1. O cadastramento da conta incluirá preenchimento de dados pessoais, preferências e APU (análise do perfil do usuário).

23.1. Todos os dados de cadastramento poderão ser editados, exceto o login.

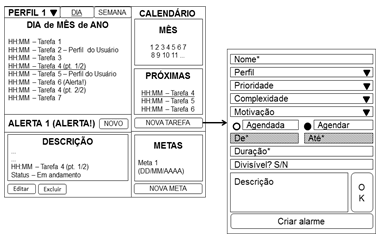
24.1. A exclusão de conta resultará na perda de todos os dados de conta e informações cadastradas.

25.1. O login será efetuado com o e-mail do usuário e senha.

Tarefas

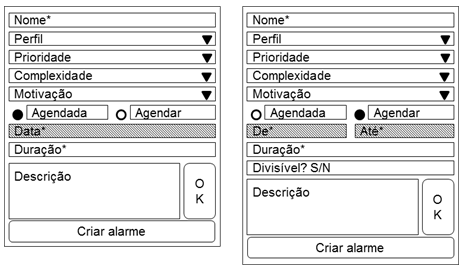
**Tarefas**

Ao clicar em nova tarefa, uma janela de preenchimento de campos será aberta.



Existem dois tipos de tarefa, as vinculadas a um compromisso com hora marcada e as não vinculadas a compromisso com hora marcada.

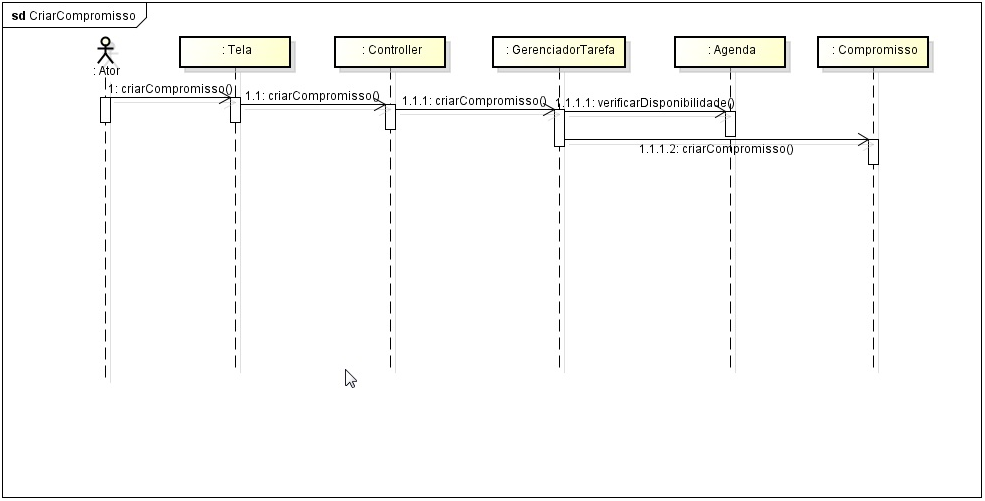
Esteticamente a diferença entre ambas estará nos campos de preenchimento. Quando selecionado o campo “Agendada”, trata-se de uma tarefa em que o usuário dará a data e hora de execução, através do preenchimento do campo “Data”. Quando selecionado o campo “Agendar”, trata-se de uma tarefa em que o usuário dará um prazo para sua execução, através do preenchimento do campo “De” e “Até”.



Tarefas

**Tarefas Agendadas**

As tarefas agendadas são tarefas com horário de compromisso já definido pelo usuário, portanto não é necessária a organização destas.

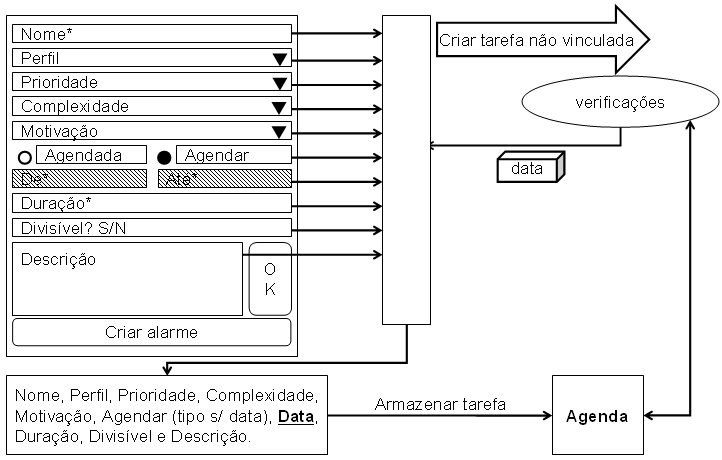


Como o esquema acima descreve, o ator emite uma ordem de criar uma tarefa vinculada a um compromisso com horário predefinido através da Tela. A Tela passa a requisição para a Controller e a Controller, por sua vez, passa a requisição para o Gerenciador de Tarefas. O Gerenciador de Tarefas tem como objetivo verificar as precondições para criação de uma tarefa. Se a agenda possuir disponibilidade de tempo para aquela determinada tarefa, então o gerenciador repassará a ordem para criá-la. Se não, uma mensagem será enviada ao usuário notificando-o que é impossível a criação daquela tarefa para a data e horário selecionado.

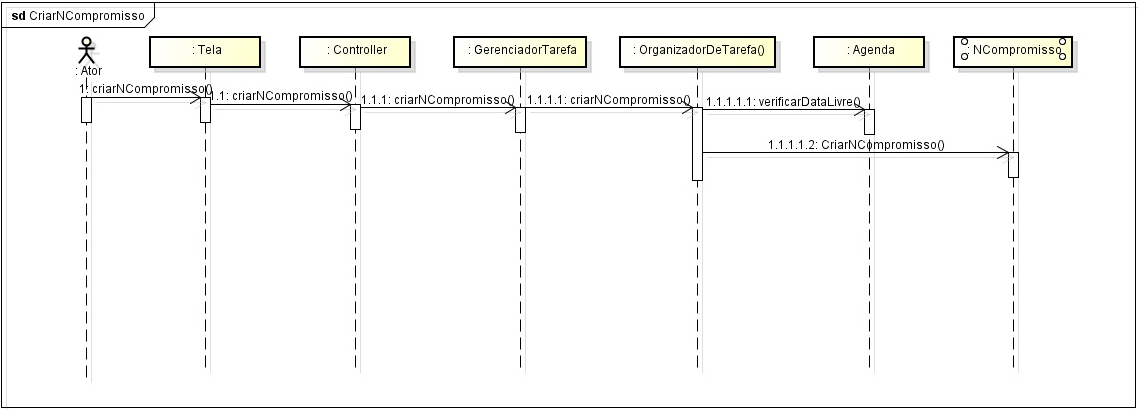
Tarefas

**Tarefas a agendar**

As tarefas a agendar são tarefas sem horário de compromisso definido pelo usuário, no entanto é exigido um intervalo de tempo para que o sistema possa encaixá-la na agenda.



O ator emite uma ordem de criar uma tarefa não vinculada a um compromisso através da Tela. A Tela passa a requisição para a Controller e a Controller, por sua vez, passa a requisição para o Gerenciador de Tarefas. O Gerenciador de Tarefas ao identificar que esta é tarefa não vinculada, envia a ordem para o Organizador de Tarefas. O papel do Organizador de Tarefas, fundamentalmente, é verificar se existe um horário livre no intervalo dado, sendo assim, dá autorização para criação da Tarefa.

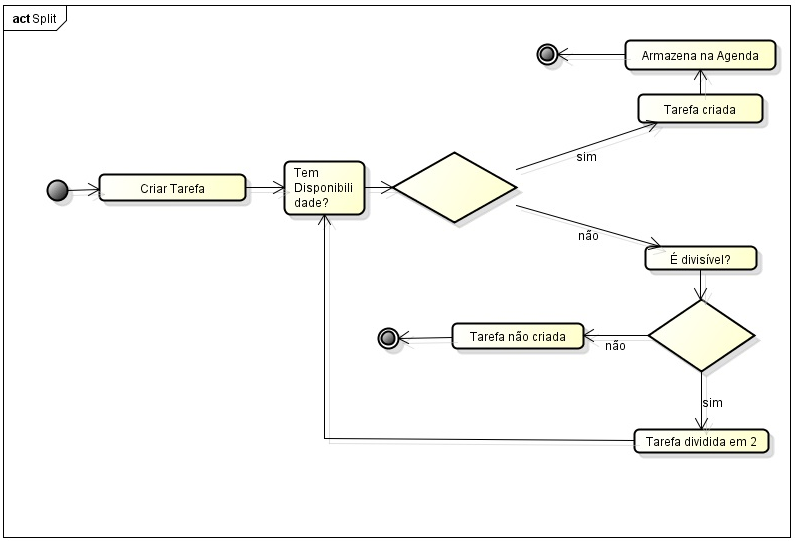
****

Tarefas

**Split**

Quando não há disponibilidade integral para uma tarefa não vinculada, no entanto há disponibilidade “em janelas”, ou seja, em momentos dispersos num prazo, a função split pode ser útil.

Desde que o campo “Divisível” da tarefa não vinculada esteja habilitado, o split quebra a tarefa em duas partes com o intuito de encaixá-la nos intervalos encontrados no dia.



O esquema acima mostra como funciona o fluxo do split. O sistema, ao verificar que não há disponibilidade integral na agenda, verifica se a tarefa é divisível ou não. Se sim, o split será executado e o sistema dividirá a tarefa em duas, ou seja, ajustará o campo “Duração” da original para metade e criará uma cópia desta com um código de partição adicionado. Se não, uma notificação será destinada ao usuário informando que não foi possível constituir a tarefa.

As tarefas que sofrerem split ficarão numa fila e até que essa fila esteja vazia, o sistema continuará o processo de organização de tarefa. Isso acontece porque a função split pode ser recursiva, ou seja, ser executada várias vezes para uma mesma tarefa.

Tarefas

**Remoção, Edição e Inplit**

A cada remoção ou edição (dos campos “Data” ou “Duração”) de uma tarefa, o Organizador de Tarefas será acionado com o intuito de reorganizar as tarefas e aperfeiçoar a agenda.

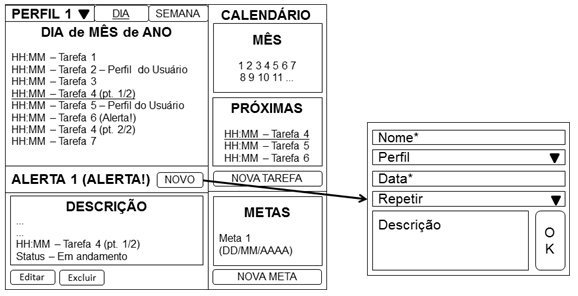
O mesmo pode acontecer caso uma tarefa com horário definido seja adicionada no horário de uma tarefa com o horário agendado pelo sistema. Se possível, o sistema armazenará em filas as tarefas não vinculadas para privilegiar o compromisso recém requisitado. Esse processo somente será efetivado se houver disponibilidade em relação aos prazos das tarefas não vinculadas.

A fim de aperfeiçoar o sistema, será necessária uma função contrária ao split, o inplit. O inplit somente será acionado caso haja uma remoção ou edição (dos campos “Data” ou “Duração”) de uma tarefa, permitindo que uma tarefa, antes quebrada, possa unificar-se novamente.

Alertas

**Alertas**

O alarme se trata de um dispositivo de notificação. Através dele o usuário será notificado de forma destacada sobre um determinado evento que cadastrar. Além disso, o alarme pode ser criado independente ou vinculado a uma tarefa, ou seja, o alarme será a notificação da tarefa e toda alteração feita na tarefa refletirá também no alarme.

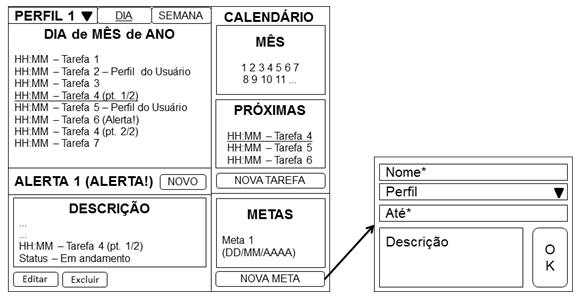


O campo “Repetir”, se habilitado, dá capacidade ao alarme de ser periódico, sendo um lembrete semanal, mensal ou anual.

Metas

**Metas**

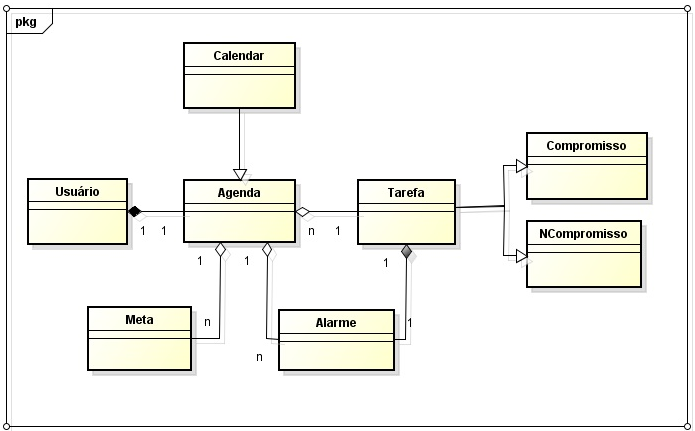
A meta são os objetivos em longo prazo. Consiste em lembretes que auxiliam a focar mais os afazeres do dia a dia. Outra função das metas é ajudar na análise do desempenho atual e do nível de crescimento obtido se, é claro, as mesmas forem realistas. Caso contrário, serve também como indicativo para reestruturação das metas para metas mais realistas.



Estrutura de objetos relevantes

**Estrutura de objetos relevantes**

A classe usuário armazenará todos os dados de login, preferências e APU, também possuirá uma instância do elemento central do projeto, a Agenda.



A agenda será uma extensão de calendário e conterá uma lista de tarefas, uma de alertas e outra de metas. Essa abstração espelha o verdadeiro sentido de agenda. *Um programa que permite a um usuário registrar ou anotar os compromissos diários* (Dicionário Michaelis).

Bibliografias

Bibliografias

Livro Getting Things Done por David Allen

Livro Zen To Done por Leo Babauto

**Livro Administração do Tempo por Marcelo de Almeida**

http://astah.change-vision.com/en/index.html