**1 Introdução**

Doador de sangue é o individuo que se voluntaria para que seja feita a extração de seu sangue para que este seja utilizado no futuro em indivíduo necessitado por esse ou quando surgir necessidade.

Como sangue não se fabrica artificialmente, a colaboração da população sempre será necessária, pois a única fonte para esse material é um indivíduo doador. No Brasil, a coleta, armazenamento e gerenciamento do sangue é feito por Hemocentros, Hospitais, Postos de Saúde, sejam estas de âmbito federal, estadual ou local.

Para se doar sangue é necessário fazer um prévio cadastro, exames para anemia e uma série de aflições que se atestado positivos resultariam na desclassificação do voluntário como doador, seja em caráter temporário ou permanente.

Atualmente, o controle destes voluntários é feito por sistemas de escopo local da instituição onde o cadastro do mesmo foi feito, seja por sistema informatizado ou registro em documentação física. [1]

Consequentemente, não existe nada que integre essas “ilhas” de doadores cadastrados, onde o banco de doadores entre diversos hemocentros e hospitais não compartilhados entre estes. É essencial agilizar e facilitar esse processo que, consequentemente, torna menos suscetível a erros e isso se torna possível com a utilização de técnicas modernas de desenvolvimento de software.

A ideia e o objetivo principal é mostrar através de ferramentas ágeis que se pode obter sangue em um processo bem mais rápido que o normal, deixando claro que é necessário sempre a colaboração e conscientização da população em geral, tratando os casos com urgência com mais prioridade e obtendo agilidade através da ferramenta API do Google Maps, introduzindo uma busca feita pelo próprio hemocentro em um raio aproximado, busca-se o maior número de pessoas com o tipo sanguíneo especificado para entrar em contato e agendar a doação o mais rápido possível.

Desenvolver software pode ser comumente comparado com a confecção de um trabalho de arte, pois este requer criatividade em seu projeto e grande habilidade para atingir a sua compleição. O software é frequentemente maleável, porém, incoerente e se mantém incompleto permanentemente. O desenvolvimento ágil de software é uma metodologia baseada em mudanças fundamentais no processo de desenvolvimento de software visionadas a muito.

Uma mudança fundamental em como gerenciar projetos, pois, se um software totalmente funcional é o resultado em busca de qual se trabalha, então se deve monitorar o progresso baseado no que se tem pronto e funcional no momento.

Para produzir um resultado que tenha a capacidade de tirar parte do ineficiente trabalho burocrático existente nessas instituições de saúde, porém mantendo a capacidade para a inclusão facilitada de modificações futuras, torna o desenvolvimento ágil de software à metodologia a ser escolhida nesse tipo de projeto, ou em qualquer outro onde o software funcional, satisfação e adequação aos requisitos do usuário são mais importantes do que a extensiva documentação de um software imutável e insatisfatório.

O Doando.se será a ferramenta que irá criar essa ponte inexistente entre doadores e hemocentros, se tornando necessário não somente para a organização interna dos hemocentros, mas como também para atrair futuros doadores.

**2 Objetivo do projeto**

O objetivo é usar a tecnologia em prol de uma causa social, facilitando o voluntariado na doação de sangue, criando uma ferramenta para os hemocentros reporem seus estoques de sangue através de uma metodologia de desenvolvimento ágil, para sanar todos estes problemas, tais como o estoque baixo, a falta de sangue, a urgência por bolsas de sangue, que muitas vezes levam a morte pela falta do mesmo.

Com esse intuito, adquire-se um foco para criar um site chamado “doando-se” permitindo o cadastro de hemocentros e doadores de sangue e que se mantenha uma conexão entre ambos.

**3 Justificativa do projeto**

A escassez de bolsas de sangue pode custar uma vida em situações de emergência e tendo em vista que atualmente a única forma de se conseguir mais doadores em situação de necessidades é por contato pessoal e aleatório. É um fator agravante não apenas a busca de doadores em si, mas sim, doadores com o tipo sanguíneo em questão.

A emergência ou doença que provoca a necessidade de transfusão de sangue é de fato um problema individual, mas para a comunidade não é, pois de antemão todos sabemos que diariamente ocorrerão acidentes de trânsito, de trabalho, crimes, violências e cirurgias.

Existem campanhas para doação de sangue em rádios, televisão, jornais, mas a grande verdade é que a atenção voltada para esses meios de comunicação, em contrapartida, o tempo gasto por cada pessoa na internet hoje em dia, vem crescendo muito, principalmente o uso de redes sociais.

Estudos mostram, que por volta do ano de 2007, as pessoas gastavam de 12 minutos online, 1 em redes sociais, porém esse número mudou, nos dias de hoje, as pessoas gastam 1 minuto em redes sociais a cada 6 minutos online. Além de que somente o Facebook em maio deste ano apenas nos Estados Unidos, atingiu a marca de 157.2 milhões de visitas registradas. [2]

Não é novidade que o Brasil sofre por uma grande defasagem de doadores de sangue, todo mês cirurgias são canceladas, pois hospitais não têm sangue suficiente em seus bancos de sangue para realizar as cirurgias.

Um exemplo está numa matéria publicada pelo jornal Estadão(2012) onde diz que pelo menos 15 cirurgias foram canceladas no diâmetro das cidades que atinge Araçatuba, Marília e Jaú no interior de São Paulo.[3]

Atualmente, no Brasil, a situação é drástica, pois, como exemplos, o estoque de sangue da Santa Casa de São Paulo está com a capacidade de 30%, em Belo Horizonte o número de doadores diminuiu 17%, em Curitiba 30%, no Rio de Janeiro caiu quase pela metade. Isso mostra que os doadores deixam de ir devido ao frio.

Houve um caso em que a cirurgia de um paciente para a remoção de um tumor acabou sendo cancelada por falta de estoque no hemocentro e toda a família teve que se unir e divulgar o problema na Internet, o que resultou na aparição de 50 doadores mobilizados pela divulgação e a cirurgia ocorreu no dia seguinte. [4]

A internet, nos dias de hoje, é o meio de comunicação mais rápido que existe, pois tudo que é colocado na internet atinge milhões de pessoas instantaneamente. Pensando no poder de mídia que a internet possui, como no caso da professora citado acima e na necessidade de divulgação sobre essas campanhas surgiu a ideia da construção de um aplicativo web que agrupe todos os possíveis doadores de sangue, mediante a um cadastro de uma forma simples, mas que contenham todas as informações que necessita ter sobre o doador. Seria uma solução na hora em que uma pessoa necessitasse do auxílio das pessoas, a aplicação necessita ser de fácil uso para a população e de fácil entendimento para os desenvolvedores na hora de uma possível alteração.

No desenvolvimento de software, os projetos são orientados a documentação que contém as especificações do software como suas funcionalidades, requisitos, restrições do software que são realizadas em uma fase onde o desenvolvedor conversa com o cliente para estabelecer essas regras, e a implementação do projeto, onde essas especificações são produzidas. Porém, muitas empresas não possuem recursos para processos de desenvolvimento pesados. Para essa questão, foram criadas metodologias de desenvolvimento ágeis, no final da década passada, para suprir as necessidades dos então atuais métodos de desenvolvimentos.

Metodologias de desenvolvimento ágil surgiu em 2001, um grupo de 17 pessoas que eram consultores de softwares e líderes no campo de ideias para desenvolvimento de softwares, se reuniram Snowbir, Utah e definiu o desenvolvimento Ágil [5]

Desenvolvimento ágil busca priorizar a comunicação entre o desenvolvedor e o cliente, preferindo conversas pessoais à outros meios de comunicação e têm os seguintes valores:

* Os indivíduos e a comunicação entre as partes acima de processos e ferramentas
* Preferência de um software funcionando ao invés de uma documentação compreensível.
* Colaboração do cliente ao invés de negociação de contrato.
* Agilidade para mudanças ao invés de seguir um plano. [6]

Nos dias atuais, tudo tem que ser produzido o mais rápido possível, em questão ao desenvolvimento ágil, existe hoje no mercado, ferramentas de desenvolvimento que trabalham com condensação de códigos to, por exemplo: um método que antes era escrito com trinta linhas de código, passa a ser escrito com quinze linhas, sendo uma alternativa rápida para o desenvolvimento, fazendo com o que o programador use um tempo menor digitando as linhas de comando e também na hora de fazer uma possível alteração no código fonte, o programador teria que lidar com menos linhas de códigos a serem analisadas, sendo assim, a manutenção se torna muito mais prática e rápida.

Por ser uma maneira atual e mais prática de se desenvolver software, a implementação ágil evita vários custos na implementação de um projeto e apresenta grandes vantagens na hora de se escrever um programa, vantagens como agilidade, o tempo gasto na construção da documentação é usados em fins mais produtivos foi usado a metodologia de desenvolvimento ágil na construção dessa aplicação.

**4 Fundamentação Teórica**

**4.1 Web 1.0 e o surgimento do manifesto ágil.**

A Web 1.0, era uma plataforma, onde usuários, no caso espectadores, podiam apenas navegar entre as páginas e visualizar seu conteúdo; não sendo permitido nenhum tipo de interação com o serviço, como por exemplo, reeditar ou alterar algum conteúdo exposto na página.

        Além disso, a grande maioria de serviços prestados na Internet, eram liberados mediante a pagamentos, ou seja, para usufruir de algum conteúdo, era necessário realizar um pagamento e uma licença única e restrita àquele usuário era gerada para que o mesmo tivesse acesso a tais serviços.

         Apesar do vasto conteúdo de informações transmitidas para o usuário, a Internet era aberta para todos, sem nenhum dono para controlar os dados acessados.

Eis que surge a segunda fase da Internet que foi o marco da evolução da Internet, onde Tim O’Reilly no ano de 2005 diz as seguintes considerações:

*A web 2.0 é a mudança para uma Internet como plataforma, e um entendimento das regras para obter sucesso nesta nova plataforma. Entre outras, a regra mais importante é desenvolver aplicativos que aproveitem os efeitos de rede para se tornarem melhores quanto mais são usados pelas pessoas, aproveitando a inteligência coletiva. (REILLY, 2005). [7]*

      Nesta mudança, obteve-se um enorme avanço em relação à interação do usuário, publicações de conteúdo, edições, criações de páginas online sem nenhuma dificuldade em nível de programação e também pode citar que o utilizador disponibilizava servidores gratuitamente.

Blogs, PodCast, Hi5, De.li.cio.us são alguns termos que podem ser citados como exemplo da grande variedades de serviços que estão hoje disponíveis graças ao avanço que a Web 2.0 propôs. [8]

       Outras características importantes, são as informações que se atualizam constantemente a fim de diminuir e obter correções para maior comodidade do utilizador, interações com pessoas do mundo inteiro, nisso, incluímos aplicações como Messenger, Googletalk, Skype e Voip, algumas ferramentas de escrita online como Google Docs e Podcast e o avanço do acesso a vídeos que cresceu muito nos últimos anos com o surgimento do Youtube, GoogleVideos e YahooVideos.

Sendo assim, com o número de usuários na web vem crescendo a todo instante, a demanda de programação para a web foi crescendo cada vez mais, e mesmo com a evolução dos computadores e tecnologias, problemas como, prazos de entrega, construção de softwares confiáveis, custos. [9] Até mesmo o não entendimento por parte dos programadores que ficavam um pouco confusos na hora de ser feita a interpretação dos textos da documentação, surgiu um novo modo de ser feita a organização dessas documentações. O Manifesto Ágil não ignora os processos e ferramentas, documentação, ou a negociação e contratos, somente mostra que tais têm importância secundária num projeto, algo que tornaria a prática na implementação do software em algo muito mais eficiente e com bastante clareza.

Um exemplo de metodologia tradicional é o modelo em cascata, que segue as regras de desenvolvimento sequencial. Ou seja, levantamento de requisitos, análise, projeto, implantação e manutenção [10] e teve como dominação na forma de desenvolver software até o início da década de 90 apesar dos pesquisadores da área de Engenharia de Software e dos desenvolvedores já tivessem notado os problemas gerados ao adotar o método sequencial. Um exemplo disto está no artigo de Fred Brooks(1987) “No silver Bullet: Essence and Accidents of Software Engineering”, que diz que é impossível especificar um projeto por completo antes do início de sua implementação. [11]

Antes do manifesto, vários fatores agravavam o não cumprimento de metas e prazos de entrega dos softwares aos clientes, tais como poucas reuniões com os clientes, onde ficavam faltando informações sobre o projeto, advogados que brigavam por causa de contratos e documentação muito extensa que acarretava o desentendimento dos desenvolvedores na hora de analisar a documentação.

O termo, desenvolver com agilidade ou “Metodologias Ágeis”, teve início no ano de 2001, e para este método foram estabelecidos princípios e características comuns destes métodos. Assim foi criada a “Aliança Ágil” e efetuou-se o estabelecimento do “Manifesto Ágil”.

Extreme Programming (XP) é uma metodologia voltada para pequenas empresas que desenvolvem projetos que contém requisitos básicos, mas que se modificam com certa frequência. Entre suas diferenças com as metodologias de desenvolvimento clássica, estão o feedback que têm que ser constante, a abordagem incremental e a boa comunicação entre cliente e desenvolvedor. Tem como seus principais objetivos a agilidade no processo de desenvolvimento e buscando sempre a satisfação do cliente. Suas práticas são conduzidas por quatro princípios básicos, sendo eles:

1. **Princípio de comunicação** – busca manter o relacionamento entre cliente e desenvolvedor sempre da melhor forma possível, com reuniões práticas e esclarecedoras, prezando sempre pelo contato pessoal.
2. **Princípio da simplicidade** - busca-se alcançar o objetivo do projeto de software com o menor número possível de classes e métodos, também está em sua característica a procura da implementação de requisitos atuais, implementando algo simples do que realizar a implementação de requisitos complicados que talvez nunca venham a ser usados.
3. **Princípio do feedback** – o desenvolvedor terá constante feedback do código que está sendo construído se informando num geral sobre todo o projeto e recebendo também feedback do cliente para qual o projeto é destinado. A informação do código é conseguida através de constantes testes em que indicam os erros que o código possui, tanto erros pessoais quanto erros de código integrado.
4. **Princípio da coragem** – Nem todas as pessoas têm facilidade de se expressar e se comunicar, sendo assim, o método é prezado pela simplicidade, os desenvolvedores podem experimentar e buscar novas soluções para os problemas, além do que, é preciso coragem para cobrar o cliente por constantes feedback. [12]

As práticas do modo XP começam pelo planejamento, sendo definido o que é, e o que não é necessário ser feito no projeto, baseando-se em requisitos atuais, deixando que requisitos futuros sejam implementados somente quando haja a necessidade deles no projeto. A entrega do software acontece conforme os requisitos vão sendo implementados, a cada requisito há uma atualização do software, cada versão entregue ao cliente deve conter o menor tamanho possível, contendo os requisitos de maior valor para o negócio.

A descrição do projeto fica por conta de metáforas, sendo assim, a descrição não conta com termos técnicos, o que possibilita um maior entendimento para o cliente, sendo muito importante a participação do cliente em todo o processo de desenvolvimento para sanar dúvidas de requisitos, evitando atrasos ou até mesmo construções erradas.

A Extreme Programming (XP) prioriza a validação do projeto durante todo o processo de desenvolvimento. Os desenvolvedores implementam o software criando primeiramente os testes, sendo a implementação de código feito em duplas, ou seja, dois desenvolvedores trabalham em um único computador, procurando identificar erros sintáticos e semânticos, pensando em conjunto em como melhorar o código que está sendo implementado.

Muito importante também é a refatoração do código, tornando um código lapidado, limpo, para que não haja problema na hora que for feita a leitura deste código que pertence a todos os membros da equipe. Isso significa que qualquer pessoa pode perceber alguma necessidade no código, pode fazê-lo mesmo sem ter sido a pessoa propriamente dita quem desenvolveu aquele código, mas desde que faça os testes necessários e não prejudique as funcionalidades atuais. A grande vantagem de todos terem acesso a todo o código, é que se algum membro abandona o projeto, outro membro não terá grandes dificuldades ao se encarregar das tarefas que foram deixadas pelo outro membro do projeto, pois todos conhecem as partes do software mesmo que não seja de forma detalhada.

A XP também assume trabalhar muitas horas por dia, mas que pode acabar acarretando um mal desempenho, pois o programador acaba ficando cansado, que por consequência acaba cometendo mais erros e pra corrigi-los, leva ainda mais tempo. Então diz-se que não se deve trabalhar mais que 40 horas semanais. Caso seja necessário trabalhar mais que 40 horas, significa que existe um problema sério no projeto que deve ser resolvido não com o aumento de horas trabalhadas, mas sim como melhorar o planejamento. [13]

Outro modelo de desenvolvimento ágil é o Scrum [14]. Seu objetivo é manter um processo de desenvolvimento favorável para o projeto de desenvolvimento orientado a objeto. O objetivo desse modelo é manter um ambiente de desenvolvimento flexível em um ambiente de trabalho onde o projeto passa por várias mudanças.

No desenvolvimento de software ágil muda-se o processo de gerenciamento do projeto para no quais obtêm software funcional feito um pouco de cada vez. O desenvolvimento ágil derruba diversos paradigmas comparados com o modo de desenvolvimento que era utilizado até então, seguindo seus princípios fundamentais.

Ao desenvolver um software é muito comum que os requisitos mudem com certa frequência, enquanto ainda a implementação está sendo realizada, mudanças em toda documentação do projeto, levariam muito tempo e tornaria o projeto imprevisível e complexo, pondo em risco a data de entrega ao cliente. Sendo assim, a ideia é a entrega de um produto que é realmente útil para o cliente. [15]

É necessário e essencial nos dias de hoje utilizar o desenvolvimento ágil de software, pois este traz mudanças na forma como o software é desenvolvido, mudando paradigmas do processo de desenvolvimento que vem acompanhando por décadas, onde o desenvolvimento tradicional têm fases e checkpoints bem definidos durante seu processo de desenvolvimento, pois no final de cada fase algum objetivo é cumprido e algo é entregue.

O desenvolvimento ágil possui iterações ao invés de fases, onde no final de cada iteração temos código funcional que pode ser usado e avaliado, podendo responder aos requisitos do usuário que estão sempre mudando e evoluindo.

As chamadas metodologias de desenvolvimento tradicionais, devem ser aplicadas apenas para projetos em que não haverá muitas mudanças no decorrer da implementação, alterações onde a necessidade de refazer o código não for muito grande e não apresente um alto custo.

Além da evolução em como a internet era vista pelos navegadores, após a evolução da web 1.0 para web 2.0 como citada acima, o desenvolvimento também sofreu uma evolução, a maioria dos sites antigos eram todos escritos em HTML.

HTML foi projetado para definir a estrutura de um documento da web, onde o código <p> é um elemento estrutural que se refere à "parágrafo", <LI> também é um elemento estrutural que se refere à "listar item" e enquanto o HTML era expandido, mais elementos eram adicionados. Estes elementos definem como o texto que será exibido, será formatado.

Em outras palavras, forma e conteúdo se tornaram inseparáveis em HTML, porém o XML foi projetado para fazer exatamente isso onde <title> não define a forma e sim o conteúdo, o mesmo com <link> e <description>, assim os dados podem ser exportados, livres de restrições de formatação.

Livres de restrições e com a popularização da internet, usuários não precisavam mais saber programas para criar um espaço próprio na Internet, nascem então os blogs, onde a programação já era toda definida pelo servidor em que hospedava os blogs, o usuário a partir de sua conta, somente tinha que enviar informações em forma de textos simples, que seria exibida em sua página.

Mesmo sendo um avanço tecnológico grande, para se construir um site, o programador dedicava uma grande parte do seu tempo para digitar inúmeras linhas de comando, às vezes para se construir pouca coisa, necessitava de um grande número de linhas.

Com o aprimoramento das tecnologias, novas linguagens de programação para web foram surgindo, e que satisfazem essa questão da necessidade do programador digitar várias linhas de comando. Exemplo dessas tecnologias de desenvolvimento ágil é a linguagem Ruby que foi escolhida por ser uma linguagem dinâmica e atual, com constante crescimento no mercado de trabalho, segue abaixo suas definições e ferramentas que podem ser usadas juntamente com a linguagem.

**4.2 Ruby**

Ruby é uma linguagem de programação interpretada multiparadigma, de tipagem dinâmica e forte, com gerenciamento de memória automático, originalmente planejada e desenvolvida no Japão em 1995 por “Yukihiro Matz Matsumoto”, para ser usada como linguagem de script. Matz queria uma linguagem de script que fosse mais poderosa do que Perl e mais orientada a objetos do que Python.

A linguagem também suporta programação funcional, orientada a objetos, imperativa e reflexiva. Foi inspirada principalmente por Python, Perl, Smalltalk, Eiffel, Ada e Lisp, sendo muito similar em vários aspectos a Python. Atualmente, Ruby é a 11ª (décima primeira) linguagem de programação mais popular do mundo, de acordo com o Índice Tiobe [16]

**4.3 Rails**

Rails é seu framework, foi escrito por “David Heinemeier Hansson” em Ruby, organizado em MVC (Model-View-Controller). Lançado para o público em 2004, sua característica é a facilidade na hora de programar, tornando códigos menores o que leva o programador programar mais em menos tempo.

Este framework surgiu para trazer uma forma mais dinâmica para o desenvolvimento de sites orientado a banco de dados, focando na agilidade para aumentar a produtividade em uma quantidade de linhas menores.

Esta é uma plataforma com opinião, ela impõe que tem um modo certo de se fazer as coisas e foi desenvolvido para encorajar o programador a seguir este modo.

Rails segue algumas filosofias e uma delas é denominada DRY (Don't Repeat Yourself) ela diz que não é bom escrever o mesmo código mais de uma vez.

Em uma plataforma de arquitetura MVC (Model-View-Controller) o ROR utiliza um recurso chamado Scaffolding que gera automaticamente controllers e views a partir de um certo modelo mostrado para o programador. RoR também provê abstração de SGBDs (Sistema de Gerenciamento de Banco de Dados), o usuário usa apenas uma sintaxe para MySQL, PostgreSQL, SQL Server, Oracle, Firebird, entre outros

O Rails faz suposições sobre o que você quer fazer e como você estará fazendo isto, em vez de deixá-lo mudar cada minúscula coisa através de intermináveis arquivos de configuração. [17]

Esta ferramenta poderosa de programação para Web pode ser rodado em Windows, Linux e Mac OS e a permissão para o uso do software é gratuitamente, podendo modificar sem restrição, com direitos a usar, copiar, distribuir e até vender.

**4.3 CoffeScript**

A linguagem disponibiliza o uso do coffescript que é uma pequena linguagem de script que compila para JavaScript para desenvolvimento de websites, é apenas JavaScript, compilando de um-para-um para o js (JavaScript) equivalente. CoffeScript tenta mostrar as boas partes do JavaScript em um modo simplificado. Esta linguagem converte os scripts .coffe em scripts .js (JavaScript). [18] Segue abaixo exemplos de sintaxes escrita em JavaScript e a mesma sintaxe escrita em CoffeScript.

Imagem 1 – Exemplo Coffescript



**4.4 HAML**

Além do CoffeScript, o uso do HAML também é utilizado. A ideia do HAML é somente a estética da sintaxe, não funcionalidades. É uma sintaxe simplificada do HTML, renderizado para HTML que não necessita o fechamento de tags. Segue abaixo exemplo de sintaxe escrita em HTML e a mesma escrita em HAML.

Imagem 2 – Exemplo HAML



**4.5 SASS**

Quando o assunto diz respeito a CSS, a tecnologia utilizada chama-se SASS, que é uma extensão para o CSS3, desenvolvida Nathan Weizenbaum, adicionando regras aninhadas, variáveis, mixins, herança de seletor. É traduzida para o formato CSS padrão. [19]

Imagem 3 – Exemplo SASS



**5 Estudo de caso**

Atualmente, existem dois aplicativos que fazem campanhas e divulgações para doação de sangue, o primeiro é o Social Blood [20], criado por Karthik Naralasetty, é um aplicativo integrado ao Facebook para unir o máximo de pessoas para contribuírem a doação de sangue e respectivamente divididas por tipos sanguíneos no próprio aplicativo. No entanto, esse sistema só é disponibilizado para usuários do Facebook, sendo limitado para usuários que não possuam a rede social, o que é um problema, pois nem todos se interagem nesses ambientes.

Outro aplicativo é o Pró Sangue [21], imposto pelo governo do estado de São Paulo, é um hemocentro que tem com o intuito de divulgar os locais para doação de sangue, mostrar a posição de estoque atual de cada tipo sanguíneo, consequentemente informando qual está em alerta, além de informar e prestar todo o tipo de apoio aos hemocentros de todo o Brasil e todas as informações necessárias de como doar sangue. Porém, este aplicativo é focado mais no Estado de São Paulo, apesar de que há a doação de sangue para outros estados, não havendo interação total com os possíveis doadores, isso desfavorece e diminui a quantidade de pessoas para a visualização do site, mesmo havendo uma conta do Pró Sangue no Facebook há limitações em relação ao foco principal do Doando.se que será a total ajuda para os casos emergenciais e o empenho em aumentar a quantidade de usuários do aplicativo.

Os estoques de sangue podem ser mantidos, no máximo, por cinco dias no caso das plaquetas, embora a medula óssea do doador faça a reposição das plaquetas por 72 horas. Os glóbulos duram um pouco mais, mas sofrem uma demanda maior, já o plasma pode ser armazenado, a menos de 18°C por tempo próximo há 12 meses [22]; ou seja, o estoque não é feito somente em caso de necessidade, mas também levando-se em conta o período de tempo em que esse material pode ser armazenado.

É essencial para os hemocentros terem sempre um estoque de sangue, pois o sangue que irá salvar vidas é o sangue já preparado e classificado. No caso de uma emergência um doador faz a diferença, porém muitas vezes o tempo necessário para preparar e analisar o sangue é longo demais fazendo com que o esforço do doador para atender a emergência seja fútil.

A comodidade das pessoas quando se trata de doação de sangue, no caso da cirurgia de um familiar ao serem informados da necessidade da doação de sangue, e são questionados se é possível trazer pessoas para doar, optam por transferir essa responsabilidade para os hemocentros raciocinando que é obrigação do mesmo fornecer o material necessário.

Portanto, a conscientização de cada um, é inevitável, pois todos os dias surgem emergências e a ajuda do próximo para manter uma corrente infinita é necessária para suprir todas as necessidades, além do amplo estoque para garantir o uso constante.

**Descrição do sistema**

O aplicativo tem como seu objetivo criar um banco de doadores de sangue, visando escopo nacional, havendo um cadastro centralizado de doadores de sangue em todo território nacional removendo uma barreira no caminho de hemocentros e instituições de saúde na hora em que se necessita de sangue, recursos como a API do Google Maps serão usados para facilitar a visualização de doadores em sua região.

Dados cadastrados nesse banco de dados serão de acesso de instituições da área da saúde, e também informações limitadas com relação aos doadores cadastrados, poderão ser visualizadas na área de acesso público, ou seja, a Homepage por qualquer pessoa, independente deste ser cadastrado ou não.

Instituições da área da saúde não só terão acesso irrestrito a central de dados de doadores, mas também terão meios de notificar doadores de tipos de sangue especifico ou de regiões especificas da necessidade de certos tipos de sangue, ou de campanhas em pró da doação de sangue que se encontra.

O Doando.se cria uma ponte entre o doador de sangue e os que precisarão deste sangue, também fazendo que seja fácil e prático o ato de se voluntariar como um doador bastando um simples cadastro.

**Página Inicial**

A página inicial é a pagina que todo e qualquer individuo irá ver quando se dirigir a URL http://doando.se a partir de um navegador web, ou seja esta é a porta de entrada para o doando.se, a primeira coisa que qualquer usuário irá ver será esta pagina, levando-se isto em conta esta pagina é de acesso totalmente público e sem nenhuma restrição, visando justamente a partir dela o individuo se torne um doador cadastrado. Os diferentes elementos desta página serão detalhados na próxima página.



Ao acessar a página inicial do Doando.se alguns ítems da página imediatamente se destacam ao visitante, logo no topo da página links para compartilhamento do Doando.se nas principais redes sociais incentivam ao visitante espalhar a ideia do Doando.se ao seu circulo de amigos, em seguida vem a barra de navegação principal do Doando.se com links para as suas diferentes seções, abaixo do menu principal da página encontra-se uma frase de apelo ao visitante para se tornar um doador, logo ao lado da frase em questão encontra-se um botão de grandes dimensões, em que a frase "Seja um doador", é acompanhada de menor destaque da frase "faça seu cadastro", sendo um dos ítens de maior destaque logo ao entrar na paágina, evidenciando o foco do Doando-se que é o cadastro de doadores.

Logo abaixo se encontra um mapa de grandes dimensões, tão largo quanto a página, sendo um ítem de grande destaque é impossível fazer a rolagem para o resto da pagina sem notá-lo, mapa este que terá uma representação gráfica bastante intuitiva de doadores e hemocentros que se encontram na região especificada pelos filtros, que se encontram na parte inferior do mapa.



Logo em seguida do mapa e seus filtros, vem a segunda parte da pagina inicial, onde algumas vantagens ou fatos motivadores são enumerados de forma elegante, para incentivar o visitante a se cadastrar e tornar-se um doador, logo em seguida se encontra três áreas paralelas, a primeira da uma breve explicação sobre o que é o Doando.se tendo opção para expandir o texto se assim o visitante desejar.

A segunda área exibe pequenos depoimentos feitos por pessoas que foram ajudadas pelo Doando.se ou que conhecem alguém que foi ajudado, essa área é dinâmica contendo dois botões de navegação onde pode se avançar pra um outro depoimento ou voltar a um anterior sem a necessidade de carregar a pagina novamente.

E finalizando a terceira área contem um pequeno formulário de contato, onde o visitante pode nos direcionar suas dúvidas, críticas ou opiniões com relação ao Doando.se, esse formulário também será dinâmico e o seu uso pra envio de alguma mensagem não resultara em outro carregamento da pagina, ele ira conter os campos E-mail, nome e o assunto a ser tratado



**Área de Hemocentros**

A área de hemocentros, assim como a pagina inicial é uma área de acesso público não sendo necessário ser cadastrado no Doando.se ou ter qualquer privilégio de acesso para visualizá-la, ela basicamente é uma listagem de hemocentros baseada em uma série de filtros assim definidos pelo usuário, facilitando a localização de hemocentros e locais disponíveis para doação, na parte superior da página se encontra uma série de filtros, é escolhido o estado e cidade, assim limitando esta listagem por local, também pode ser escolhida tipos de notificações expedidas, como por exemplo campanhas ou procura de doadores de tipos específicos de sangue, e por final é escolhido a natureza da instituição, seja esta um hemocentro ou hospital, logo abaixo há um campo de busca livre onde pode ser feita uma busca pelo nome do hemocentro ou instituição procurada.



Em seguida, se localiza a lista com os resultados dessa busca ou filtragem, essa lista apresenta as respectivas colunas, nome: nessa coluna é exibido o nome do hemocentro ou instituição de saúde que se enquadra nos critérios da busca, endereço: é exibido a localização exata do hemocentro ou instituição de saúde, tipo: a natureza da instituição, ou seja se é um hemocentro ou hospital. Detalhes: é exibido um link para uma página com informações detalhadas da instituição especificada, como por exemplo campanhas em andamento ou falta de tipos específicos de sangue.



**Área privada**

Esta é a principal pagina inicial do doando.se que será utilizada pelas instituições de saúde, será a página onde serão feita todas as consultas e ações por parte das instituições de saúde, esta é uma pagina restrita somente aos hemocentros e usuários permitidos por estes, não sendo uma página de acesso público, pelo contrario a partir desta pagina informações pessoais dos doadores podem ser exibidas sendo essencial o seu acesso restrito, basicamente toda a coleta de dados e registro de doadores do doando.se é com o objetivo de dar as instituições de saúde essa ferramenta.



Na lateral esquerda desta pagina se encontram informações detalhadas da empresa, informações como nome, CNPJ e razão social são exibidas na íntegra, assim como informações de contato e o endereço completo.



Logo ao lado se encontra uma serie de links com ações disponíveis a instituição, o link para paágina de usuários leva a empresa a página para o cadastro e edição de seus usuários que terão acesso ao sistema, em seguida vem o acesso a página de doadores, onde a instituição poderá procurar e filtrar doadores por diversos parâmetros como local de residência, sexo, idade, tipo sanguíneo entre outros, a partir dai disparando notificações para estes doadores ou os contatando de alguma outra forma. Logo vem a página de quem somos, onde poderá ser editadas informações sobre a identidade da instituição que será de acesso público aos usuários, logo em seguida vem a central de notificações onde poderão ser criadas notificações para grupo de doadores ou até mesmo campanhas que poderão ser visualizadas por todos, independentemente de ser um doador registrado ou não.



Abaixo disso se encontra uma lista atualizada de forma dinâmica, onde é exibida notificações disparadas recentemente e seu estado atual, ou seja se algum doador já respondeu a esta notificação ou não, em cada item da lista se encontra um link com o objetivo de exibir detalhes de uma notificação especifica.

**Área de usuários**

Na área de usuários é listado os usuários do sistema registrados pela instituição, como informações detalhadas sobre estes como carga na empresa e se ele é um empregado atual da empresa ou não, esta lista contém alguns filtros simples, se o usuário tem sua permissão de acesso ativa ou não, e também conta com um campo para busca de usuários específicos, logo em sua parte superior se encontra um botão para a adição de novos usuários.



**Área de administradores**

A área de administradores ou "admin" é similar a pagina das instituições, porém ela não será relevante a nenhum dos usuários do sistema sejam eles doadores ou instituições, pois seu acesso será habilitado somente aos administradores do doando.se, e pode fazer alterações de escopo global. Nela serão cadastradas novas instituições que utilizarão o doando.se, como também poderá ser feita a remoção de alguma instituição que não mais queira utilizar o doando.se, permite também o cadastro de usuários com privilégio administrativo, como também o cadastro de parceiros do doando.se e edição de informações exibida na pagina principal e nas áreas de acesso público.

**6 Apêndice A**

**6.1 Dicionário de Dados (glossário)**

**Open-source:** Open Source é um conceito de distribuição de software, que estabelece como fundamentais, os princípios de desenvolvimento compartilhado, distribuição na forma de código fonte e licenciamento gratuito.

**CSS:** CascadingStyle Sheets,é uma [linguagem de estilo](http://pt.wikipedia.org/w/index.php?title=Linguagem_de_estilo&action=edit&redlink=1) utilizada para definir a apresentação de documentos escritos em uma linguagem de marcação, como [HTML](http://pt.wikipedia.org/wiki/HTML) ou [XML](http://pt.wikipedia.org/wiki/XML).

**Framework:** Framework é um conjunto de classes que colaboram para realizar uma responsabilidade para um domínio de um subsistema da aplicação. -  Fayad e Schmidt

**Facebook:** Facebook é um [site](http://pt.wikipedia.org/wiki/Site) e serviço de [rede social](http://pt.wikipedia.org/wiki/Rede_social) que foi lançada em [4 de fevereiro](http://pt.wikipedia.org/wiki/4_de_fevereiro) de [2004](http://pt.wikipedia.org/wiki/2004), operado e de propriedade privada da Facebook Inc.

**HTML:** abreviação para a [expressão inglesa](http://pt.wikipedia.org/wiki/L%C3%ADngua_inglesa) *HyperText Markup Language*, que significa *Linguagem de Marcação de Hipertexto*, é uma [linguagem de marcação](http://pt.wikipedia.org/wiki/Linguagem_de_marca%C3%A7%C3%A3o) utilizada para produzir [páginas na Web](http://pt.wikipedia.org/wiki/P%C3%A1gina_web).

**Ministério da Saúde:** O Ministério da Saúde, no [Brasil](http://pt.wikipedia.org/wiki/Brasil), corresponde ao setor governamental responsável pela administração e manutenção da [Saúde pública](http://pt.wikipedia.org/wiki/Sa%C3%BAde_p%C3%BAblica)do país.

**Referências**

1 Portaria Anvisa Aprova o Regulamento Técnico de Procedimentos Hemoterápicos (2011). Disponível em <http://portal.anvisa.gov.br/wps/wcm/connect/7a2915004b948667a9fabbaf8fded4db/Portaria\_MS\_1353\_13\_de\_junho\_de\_2011.pdf?MOD=AJPERES>. Acesso em 24 Junho 2012.

2 Lipsman, A. (2011) The Network Effect: Facebook, Linkedin, Twitter & Tumblr Reach New Heights in May. Disponível em: <http://blog.comscore.com/2011/06/facebook\_linkedin\_twitter\_tumblr.html>. Acesso em 07 abril 2012

3 Siqueira, C. (2012) Cirurgias são canceladas por baixo estoque de sangue. Disponível em: <http://www.estadao.com.br/noticias/impresso,cirurgias-sao-canceladas-por-baixo-estoque-de-sangue--,831192,0.htm>. Acesso em 07 de Julho 2012

4 Baixo estoque de sangue em hemocentros do Brasil adia cirurgias (2012). Disponível em <http://g1.globo.com/jornal-nacional/noticia/2012/05/baixo-estoque-de-sangue-em-hemocentros-do-brasil-adia-cirurgias.html> . Acesso em 24 junho 2012.

5 History: The Manifesto Agile (2001). Disponível em: <http://www.agilemanifesto.org/history.html> Acesso em 24 junho 2012.

6 Manifesto for Agile Software Development (2001) Disponível em <http://www.agilemanifesto.org/> Acesso em 24 junho 2012

7 O'Reilly, T. (2005) What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software Disponível em: <http://oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>. Acesso em 13 julho 2012

8 Richardson, W. (2006). Blogs, Wikis, Podcast and other powerful Web tools for classrooms. Thousand Oaks, CA: Corwin Press.

9 Standish Group, “CHAOS report”, 586 Olde Kings Highway, Dennis,

MA 02638, USA, (1995)

10 Pressman, R. “Engenharia de Software” McGraw-Hill, (2001)

11 Brooks, F., “No Silver Bullet: Essence and Accidents of Software Engineering” Proc. IFIP, IEEE CS Press, 1987, pp. 1069-1076; reprinted in IEEE Computer, Apr, (1987), pp.10-19

12 Conceitos básicos sobre Metodologias Ágeis para Desenvolvimento de Software (Metodologias Clássicas x Extreme Programming). Disponível em: <http://www.devmedia.com.br/conceitos-basicos-sobre-metodologias-ageis-para-desenvolvimento-de-software-metodologias-classicas-x-extreme-programming/10596> Acesso em 14 de julho 2012

13 BECK, Kent; FOWLER, Martin. (2000) Planning Extreme Programming

14 Schwaber, K. e Beedle, M. "Agile Software Development with SCRUM", Prentice-Hall, (2002)

15 Schwaber, K. "Scrum Development Process", OOPSLA'95 Workshop on Business Object Design and Implementation. Springer-Verlag. (1995)

16 Bruce, S. (2001). An Interview with the Creator of Ruby. Disponível em <http://linuxdevcenter.com/pub/a/linux/2001/11/29/ruby.html>. Acesso em 14 julho 2012

17 Ruby. O que é Ruby on Rails? (2012) Disponível em <http://ruby-br.org/?page\_id=89> Acesso em 26 de Maio 2012.

18 Jeremy, A. CoffeScript (2011). Disponível em <http://coffeescript.org/> Acesso em 16 de julho 2012.

19 Sass. Syntactically Awesome Stylesheets. (2011). Disponível em <http://sass-lang.com/> Acesso em 16 de Julho 2012.

20 Social Blood. (2012) Disponível em <http://socialblood.org/>. Acesso em 20 de Julho 2012.

21 Prosangue. (1984) Disponível em <http://prosangue.sp.gov.br/home/Default.aspx> . Acesso em 20 de Julho 2012.

22 O sangue que nos salva, tudo começa no hemocentro. (1998) Disponível em <http://boasaude.com.br/lib/ShowDoc.cfm?LibDocID=3915&ReturnCatID=1771>. Acesso em 22 de Julho 2012.