

**Hep Agenda - Software de gestão de atendimentos prestados no ambulatório de
Hepatites da UFMG**

Emerson Pereira Siqueira

Guilherme Gabriel Silva Pereira

Henrique Penna Forte Monteiro

Layla Raissa Silva Pereira

Lucas Ângelo Oliveira Martins Rocha

Maycon Henrique Souza Tavares

Victor Boaventura Góes Campos

¹Instituto de Informática e Ciências Exatas – Pontifícia Universidade Católica de Minas
Gerais (PUC MINAS)

Belo Horizonte – MG – Brasil

emerson.siqueira@sga.pucminas.br

ggspereira@sga.pucminas.br

henrique.forte@sga.pucminas.br

raissalayla16@gmail.com

laomrocha@sga.pucminas.br

maycon255@hotmail.com

bgcampos@sga.pucminas.br

**Hep Agenda - Software de gestão de atendimentos prestados no ambulatório de
Hepatites da UFMG**

Desenvolver um programa para registrar os atendimentos prestados no ambulatório com o intuito de facilitar e melhorar o atendimento prestado ao paciente.

Coordenador da extensão: Ivre Marjorie Ribeiro Machado

Belo Horizonte

2021

SUMÁRIO

1. Apresentação.....	4
1.1. Problema.....	4
1.2.1 Objetivo geral.....	5
1.2.2 Objetivo específico.....	5
1.3. Justificativa.....	5
1.4. Público alvo.....	6
2. Requisitos.....	6
3. Arquitetura da Aplicação.....	10
4. Metodologia de trabalho.....	11
5. Implementação da solução.....	11
Referências.....	13

1. Apresentação

1.1. Problema

O ambulatório da UFMG necessita de um sistema melhor de organização para registrar os atendimentos dos pacientes, pois o sistema anteriormente utilizado não contempla todos os requisitos necessários, faltando componentes essenciais para uma consulta eficiente e entendimento do paciente por completo.

1.2.1 Objetivo geral

O presente trabalho tem como objetivo geral melhorar a organização dos dados e atendimentos aos pacientes do Ambulatório da UFMG. O intuito é desenvolver uma aplicação capaz de agendar consultas e visualizar o histórico dos pacientes, de forma que o trabalho dos médicos no ambulatório possa ser facilitado e a experiência dos mesmos com a agenda seja superior ao sistema que possuem atualmente.

1.2.2 Objetivos específicos

Este trabalho objetiva especificamente desenvolver uma aplicação que possibilite ao usuário administrador marcar consultas de forma mais eficiente, visualizar a agenda de consultas, preencher dados específicos sobre os pacientes, anexar arquivos, bem como visualizar posteriormente o histórico dos pacientes. Nesse sentido, objetivando um melhor acompanhamento e entendimento da situação de cada um, além de facilitar a administração de médicos/usuários do sistema que trabalham no ambulatório.

1.3. Justificativa

1.4. Público alvo

2. Requisitos

1. O administrador deve cadastrar, ler, atualizar e deletar (soft) usuários no sistema (médicos, administradores e visualizadores).
2. O administrador deve ser capaz de editar e alterar a senha dos usuários.
3. O sistema deve orientar aos usuários sobre como solicitar a alteração de senha (Enviar e-mail ao adm ou entrar em contato diretamente).

4. O sistema deve permitir o login automatizado do usuário, após o primeiro acesso, caso ele queira.
5. O sistema deve permitir o logar e deslogar.
6. O administrador e médico devem cadastrar e atualizar pacientes no sistema.
7. O médico e o administrador devem ser capazes de visualizar os agendamentos por dia/mês/ano.
8. O administrador deve ser capaz de marcar/criar e atualizar um agendamento no sistema.
9. O médico deve ser capaz de atualizar o status do agendamento para aguardar atendimento realizado.
10. O sistema deve gerar um histórico de alterações dos agendamentos (Log).
11. O administrador deve ser capaz de classificar o agendamento em: aguardando confirmação, aguardando atendimento, atendimento realizado e atendimento cancelado.
12. O administrador deve ser capaz de ativar ou desativar datas semanais de agendamentos.
13. O administrador deve classificar o médico em caso de ausência por um período de folga/participação em congresso ou férias.
14. O sistema deve permitir o agendamento sem vínculo de médico.
15. O sistema deve permitir pesquisar usuários por nome e visualizar nome completo, data de nascimento e nome da mãe
16. O sistema deve informar se é o primeiro agendamento (se nunca concluiu nenhum agendamento antes) do paciente ou retorno.
17. Todos os usuários devem ser capazes de filtrar os pacientes por nome, tipo de hepatite, idade e gênero.
18. Todos os usuários podem visualizar os atendimentos.
19. Todos os usuários podem filtrar os agendamentos por: aguardando confirmação, aguardando atendimento, atendimento realizado e atendimento cancelado.
20. Todos os usuários podem visualizar os dados do paciente.
21. Todos os usuários podem visualizar a quantidade de agendamentos marcados.
22. O administrador e o médico podem anexar arquivos relacionados ao paciente no sistema.
23. O administrador e o médico podem adicionar observações referentes ao atendimento prestado.

24. O sistema deve permitir visualizar o relatório de todos os atendimentos realizados no sistema, categorizados por: ano, mês, tipo de hepatite e outros.
25. O administrador pode criar e desativar tratamentos de hepatite.

3. Arquitetura de Aplicação

4. Metodologia de trabalho

A metodologia para gerenciar o projeto que será utilizada pelo grupo será o Scrum. Um backlog com todas as funcionalidades gerais do projeto será criado para ser entregue com eficiência. Caso surjam bugs ou novas funcionalidades, eles serão priorizados no backlog. O desenvolvimento do projeto será feito gradativamente, sendo entregue valor ao cliente a cada sprint finalizado. Cada sprint terá a duração de uma semana e um board será criado no GitHub para controlar as tarefas da equipe.

Aplicativos e métodos que serão usados:

Scrum: metodologia ágil de gerenciamento;

WhatsApp: meio de comunicação;

GitHub: planejamento dos backlogs, definição de tarefas e prazos;

GitHub: compartilhamento de código;

Visual Studio Code: software para criação das páginas em HTML, CSS e JavaScript.

O planejamento inicial será levantar os requisitos, criar o wireframe de cada tela que será exposta para os clientes, criar a identidade visual da marca, implementar telas em HTML, CSS, Javascript e testar todas as páginas e funcionalidades, para garantir a qualidade da aplicação.

5. Implementação da solução

6. Avaliação da aplicação

REFERÊNCIAS