

## REPORT VIEW

**David Elias Souza Lopes**

Centro Universitário de Anápolis - Uni EVANGÉLICA. sousadias2000@gmail.com

**Fabio Rocha de Sousa Pires**

Centro Universitário de Anápolis – Uni EVANGÉLICA. fabiorochaes@gmail.com

**Gabriel Fernandes Gonçalves**

Centro Universitário de Anápolis - UniEVANGÉLICA.fernandesgabriel1931@gmail.com

**Torry Filho**

Centro Universitário de Anápolis – Uni EVANGÉLICA filhotorry@gmail.com

**Osnaldo Neto**

Centro Universitário de Anápolis – Uni EVANGÉLICA osnaldoxgames@gmail.com

## Resumo

*O projeto consiste na interdisciplinaridade entre as matérias de Banco de Dados II, Análise e Projeto de Software, Arquitetura de Software e Programação e Tecnologias Web. Com o objetivo de criar um sistema que traga soluções para um problema real do ambiente de trabalho, e que de alguma maneira ele contribua para a boa convivência e bem estar onde passamos grande parte de nosso tempo. Este relato não tem o intuito de apresentar o sistema por completo, mas a intenção de descrever a experiência vivida ao longo do projeto. A ideia escolhida “ReportView” foi criar um sistema que crie uma ligação entre um usuário e o administrador que administre uma empresa, cidade, faculdade ou um comercio que tenha grande área construída e com grande número de usuários no local. O usuário poderá relata problemas diretamente ao administrador, através disso, problemas estruturais poderiam ser resolvidos mais rápidos e assim poupando recursos ambientais e financeiros*

**Palavras-Chave:** usabilidade, gestão, software, ambiente de trabalho, agilidade;

## 1. Introdução

Quais os problemas em grandes empresas? Como solucioná-los? Como facilitar a manutenção e como preveni-los? Essas perguntas formam a base para esse projeto. Uma casa má administrada pode gerar desperdícios de muitos recursos como água e energia, como por exemplo torneiras pingando, lâmpadas e aparelhos ligados sem necessidade, problemas estruturais no encanamento ou fiação, resultando em prejuízos econômicos, ambientais, e uma qualidade de vida ruim no local de trabalho.

Um local de trabalho adequado é indispensável para a um bem estar, e melhora na produtividade dos colaboradores. Afinal, como esperar que façam um bom trabalho em locais com barulho, goteiras, e infraestrutura inadequada.

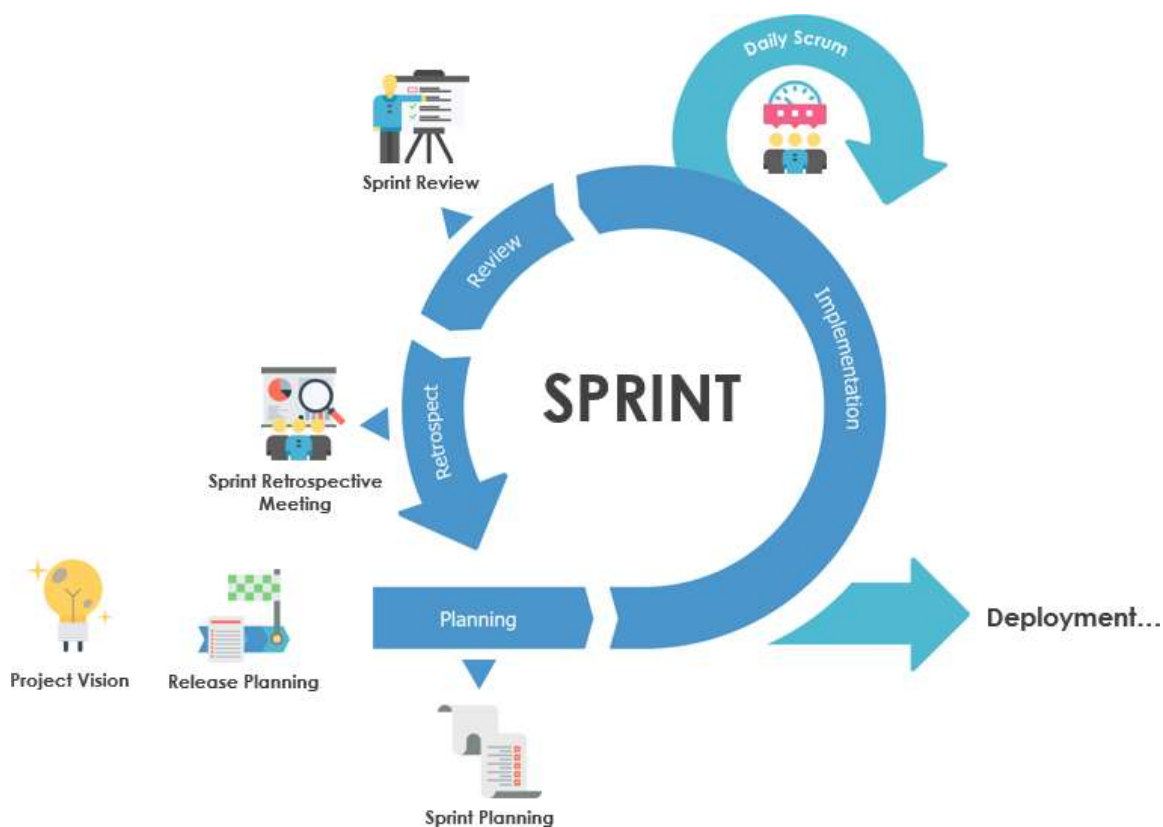
Mas é possível que esses problemas sejam mais fáceis de lidar, com a ajuda dos frequentadores do local, dando espaço e liberdade para que seja identificado e reportado com facilidade esses problemas em questão. Em seguida o setor responsável pelo local e tipo de problema, possa estar por dentro e organizando como será a solução.

O colaborador em geral que tem maior chance de identificar os problemas, pelo fato de nem sempre ter alguém da manutenção andando vagamente pelo local, entretanto esse sistema resolve também o problema de comunicação entre o usuário e o gerenciador do local, evitando o desgaste de se locomover do local dependendo da distância, somente para reportar um problema.

## 2. Metodologia

O modelo de ciclo de vida de software escolhido foi o XP, O modelo oferece análise prévia dos acontecimentos dentro do projeto, que oferece qualidade e confiança. E com reuniões semanais é mais fácil encontrar erros.

Figura 1: Modelo Xp.(Fonte: Visual paradigm. **Extreme Programming XP vs Scrum**)



### 2.1 Protótipo

O protótipo a seguir foi feito no site 'Canva':

Figura 3: Protótipo administrador



O protótipo foi feito pensando para ser ágil e fácil de usar, para incentivar os usuários do sistema a acessá-lo sempre que vem um problema no local, um sistema com muitas funções apenas deixaria mais confuso e desmotivando as pessoas a usá-lo.

## 2.2 Funções:

**Empresa Logo:** como o nome deixa a entender está área permite a adição de uma imagem com o logo da empresa, é um detalhe simples que não está entre os principais requisitos do sistema porém esses detalhes podem fazer diferença na escolha do cliente, o programa lyceum é um exemplo que disponibiliza essa opção de personalização ao usuário.

**Notificações:** Todos as novas mensagens para o administrador serão enviadas para este local, separando os problemas antigos dos novos.

**Usuário:** área para login, garantindo a segurança do usuário em usar o site. E qualquer mensagem enviada para o administrador fica salvo o usuário que enviou a mensagem, para evitar que pessoas de dentro ou de fora do sistema os usem de forma maliciosa, como enviar trotes de falsos problemas.

**Problemas a tratar:** Problemas que ainda esperam uma decisão a ser tomada.

Relatório: função que mostra para o administrador os resultados obtidos usando o sistema como por exemplo: o total de problemas recebidos, problemas resolvidos e não resolvidos, e a interação dos usuários no sistema.

Histórico: problemas passados que já tiveram uma solução ou não.

Figura 3: Protótipo map view



Map View: essa função mostra ao administrador onde os problemas aconteceram através de um mapa do local que deverá ser adicionado previamente pelo administrador.

### 2.3 Diagramas:

Para a projeção dos diagramas criados durante o projeto foi utilizado as ferramentas Draw.io, BrModelo, dentre outras. Com o intuito de descrever o que o sistema irá fazer, foram construídos diversos diagramas, a seguir alguns deles.

Figura 4: Diagrama de maquina de estado

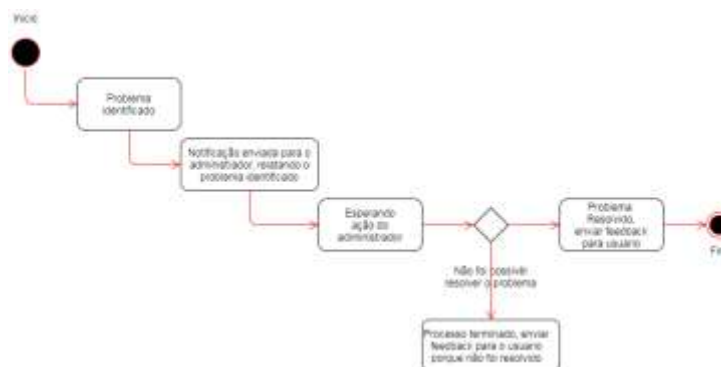
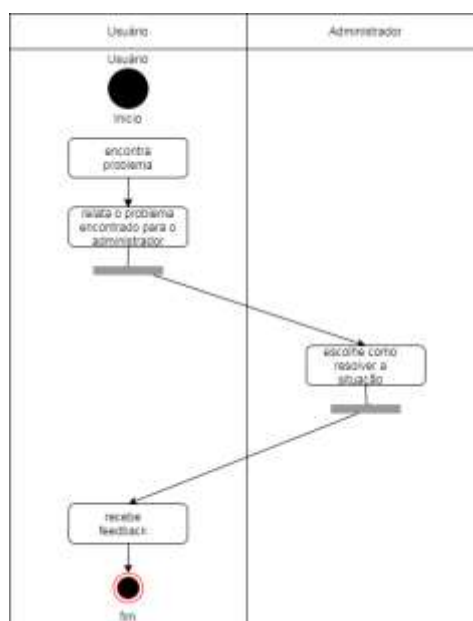


Figura 5: Diagrama Comportamental de atividades



O diagrama de maquina de estado mostra ações em cada fase do projeto, e o diagrama comportamental de atividades mostras as atividades dos usuários é do administrador, tem outros diversos diagramas esses dois foram escolhidos por retratar de maneiras mais simples o processo da resolução do problema.

### 3. Resultados e discussão

Concluído

Document Management System (DMS)

Escopo

Protótipos

Diagramas

Incompleto

React (em andamento)

Arquitetura (em andamento)

EAP

O DMS foi desenvolvido desde o início do projeto, a equipe fez adições ao documento periodicamente. No escopo desenvolvido no início do projeto, os requisitos ao longo do tempo ficaram mais desenvolvidos e sólidos, no restante não houve mudanças drásticas no desenvolvimento do projeto.

Não houve alterações no protótipo original e nos diagramas. Mesmo com as alterações de requisitos o protótipo se manteve estável e sem necessidade de alteração, a grande quantidade de diagramas faz com que seja inviável alterar todos os diagramas para cada alteração de requisito.

O distanciamento da pandemia atrasou o desenvolvimento do front-end e back-end, além do problema de comunicação entre os membros teve falta de recursos para desenvolver o projeto,

### 4. Conclusão

A partir desse trabalho tivemos a oportunidade de observar melhor a criação de um programa, não apenas nas áreas práticas como documentação e programação, mas também nas áreas de pesquisa sobre o tema, pensar como uma empresa funciona e do que precisa, esses conhecimentos vão além do que pode ser ensinado em aulas. Outro aspecto que vale ressaltar foi trabalho em equipe, a interdisciplinaridade das disciplinas do curso, e a experiência em utilizar ferramentas de criação de diagramas, construção de modelos científicos, protótipos e outros. Apesar de dificuldades e alterações da faculdade e tempo reduzido, conseguimos tirar boas experiências ao longo do projeto.

### 5. Referências

[Figura 1] Visual paradigm. **Extreme Programming XP vs Scrum**. Disponível em: <https://www.visual-paradigm.com/scrum/extreme->

[1] NILDES R. PITOMBO LEITE e LINDOLFO GALVÃO DE ALBURQUERQUE. **O desafio de gerir pessoas em uma estrutura organizacional remota**. Revista Usp, 2009.

[2] HABIL COWORKING. **A relação entre ambiente de trabalho e produtividade**, Belo Horizonte. Disponível em <<https://habilcoworking.com.br/a-relacao-entre-ambiente-de-trabalho-e-produtividade/>>

[3] MARINA RIBEIRO CESAR. **Tecnologia da informação como ferramenta para transparência e eficácia na administração pública**. Disponível em: <<https://app.uff.br/riuff/handle/1/16744>>