

Super Agenda

Trabalho final da disciplina Desenvolvimento de Software

Eduardo Schuindt Santos
Faculdade do Gama
Universidade de Brasília
Brasília, Brasil
202042927@aluno.unb.br

Fernando Vargas T. de Oliveira
Faculdade do Gama
Universidade de Brasília
Brasília, Brasil
180016491@aluno.unb.br

Hoton Geraldo do Carmo Junior
Faculdade do Gama
Universidade de Brasília
Brasília, Brasil
200038389@aluno.unb.br

João Lucas Pinto Vasconcelos
Faculdade do Gama
Universidade de Brasília
Brasília, Brasil
190089601@aluno.unb.br

Pedro Henrique Nogueira Bragança
Faculdade do Gama
Universidade de Brasília
Brasília, Brasil
190094478@aluno.unb.br

Raquel T. Eucaria Pereira da Costa
Faculdade do Gama
Universidade de Brasília
Brasília, Brasil
202045268@aluno.unb.br

Abstract—Este artigo tem como objeto relatar o processo de aprendizado em relação ao desenvolvimento de software para que os alunos tenham as primeiras experiências com programação. E o tema escolhido pelo grupo para essa experiência foi a de desenvolver uma agenda que auxilie o estudante a se organizar com seus estudos, entregas de trabalhos, horários de estudos, dias de provas, entre outras atividades. Foi utilizado também metodologias de desenvolvimento de software como o Scrum. A Partir disso abordaremos alguns itens importantes para o curso de Engenharia de Software.

Palavras-chaves—Scrum; desenvolvimento; protótipo.

I. INTRODUÇÃO

O presente artigo foi elaborado com o propósito de destacar a agenda como ferramenta para auxiliar os estudantes durante o seu aprendizado. É de grande importância o uso da agenda para o crescimento do estudante, entretanto, pode-se afirmar que a falta desta, afeta de forma significativa o mesmo, dificultando o seu aprimoramento.

Perante o atual estado em que se encontra o País devido a pandemia do coronavírus, é perceptível que os estudantes demonstraram ainda mais dificuldade com a organização da sua rotina e suas atividades, principalmente quando se trata sobre estudos. É normal o esquecimento para entregar um trabalho, de estudar para prova ou de fazer atividades enviadas para casa. Acaba-se esquecendo dessas tarefas devido a falta de organização e os contratempos do dia a dia.

O problema fica mais evidente quando os alunos tentam elaborar seu plano de estudos. Um plano de estudos bem organizado além de facilitar nos estudos pode facilitar no momento de recuperar algum conteúdo perdido.

A utilização de tecnologias nos estudos permite que os estudantes consigam realizar atividades com mais praticidade e rapidez, fazendo com que desperdicem menos tempo com outras atividades.

II. ASPECTOS TÉCNICOS

A. Método scrum

O método Scrum é um framework utilizado para gestão dinâmica de projetos, sendo muitas vezes aplicado para o desenvolvimento ágil de um software [1]. Este método tem como objetivo alcançar 3 fundamentos:

- Transparência dos processos, dos requisitos de entrega e status. Todos os aspectos significativos do processo como um todo devem estar visíveis e alinhados com todos os responsáveis pelos resultados.
- Inspeção constante de tudo o que está sendo feito.
- Adaptação tanto do processo, quanto do produto, que podem sofrer mudanças que necessitam de adaptação. Também é importante adaptar o Scrum para a realidade e cultura da empresa. Nessa metodologia há a distribuição de papéis entre as equipes, que fica dividida entre Product Owner, Scrum Master e Scrum Team.
- O Product Owner, também conhecido como PO, é o líder das ideias do produto e possui a tarefa de gerenciar sua produção de acordo com o Product Backlog, definindo o grau de prioridade da produção.
- O Scrum Master é o líder da equipe e responsável por aplicar os fundamentos do método Scrum dentro dos integrantes do scrum team. Ele que impulsiona e que fica responsável por “pegar no pé” da equipe.
- O Scrum Team é formado pelos demais membros da equipe e são a parte mais fundamental do projeto. Eles são designados para fazerem todas as tarefas descritas pelo PO de acordo com o Product Backlog e para darem o relato de quais tarefas serão possíveis ou não de serem feitas.

O Product Backlog é a lista de itens que devem ser desenvolvidos pela Scrum Team e se divide entre épicos, features e histórias de usuário. Um épico representa uma coleção de funcionalidades da aplicação que estão logicamente agrupadas.

Se trata de uma área específica de uma grande funcionalidade (épico).

Também há, na metodologia Scrum, um artifício denominado por Sprint que são semanas que duram de 1 a 4 semanas e que devem ter duração constante, onde se é trabalhado funcionalidades do Product Backlog. A Sprint se inicia com um Sprint Backlog que é uma reunião que define a funcionalidade a ser trabalhada. Há também as Daily Sprints que são reuniões diárias para dar um feedback do que foi trabalhado no dia e onde surgem as críticas de possíveis melhorias.

Nossa equipe optou por fazer revezamentos entre as funções de PO, Scrum Master e Scrum Team para que todos tivessem a oportunidade de exercer as diferentes funções e pudessem adquirir uma melhor experiência com o projeto. Ficou definido o seguinte esquema de revezamento:

Sprint	Scrum Master	PO	Scrum Team
Sprint 0	Fernando	Hoton	Demais membros
Sprint 1	Raquel	Fernando	Demais membros
Sprint 2	Hoton	Raquel	Demais membros
Sprint 3	Eduardo	Pedro	Demais membros
Sprint 4	João Lucas	Eduardo	Demais membros
Sprint 5	Pedro	João Lucas	Demais membros
Sprint 6	Fernando	Raquel	Demais membros
Sprint 7	Eduardo	João	Demais membros
Sprint 8	Pedro	Hoton	Demais membros

Fig. 1. Revezamento Scrum.

O período escolhido de duração das sprints foi de 1 semana, que se iniciava às quartas-feira com a Sprint Backlog e se encerrava somente após o início da próxima Sprint Backlog. As Daily Scrum não foram diárias, mas somente segunda-feira e sexta-feira.

B. Backlog

Originalmente, o Backlog ficou dividido em 2 épicos. O primeiro épico se refere às características mais essenciais do produto e o segundo a possíveis adições para trazer mais funcionalidades.

O primeiro épico, definido como Agenda, possuía 4 features, com cada feature possuindo 4 histórias de usuário, exceto a última que possuía 8 histórias de usuário. Este épico continha o fundamental do aplicativo, contendo as seguintes features:

1) Horários: A visualização de uma tela de horários com os 7 dias da semana e as respectivas horas, com espaçamento de 1 hora entre elas. Também possuiria uma tela de cadastro, exclusão e/ou edição de horários, podendo escolher o dia da semana, a disciplina que seria atribuída, o horário de início e término e, por fim, um lembrete para notificações.

2) Disciplinas: A visualização de uma tela à disposição de todas as disciplinas cadastradas. Também possuiria uma tela

de cadastro, exclusão e/ou edição de disciplinas, contendo o nome da disciplina, a cor e o nome do professor.

3) Notas: A visualização de uma tela de notas anexas às disciplinas, contendo o valor, o peso, o título e a descrição da nota. Ainda, possuiria o cadastro, exclusão e/ou edição de notas e um cálculo da média ponderada, média aritmética ou, somente, soma.

4) Eventos: A visualização da tela de eventos cadastrados, podendo ser uma prova, um trabalho ou uma tarefa, a devolução de um livro, dentre outros. Conteria, ainda, uma tela para cadastro, exclusão e/ou edição de um evento que teria como conteúdo o nome, a data, o horário, a disciplina (opcional) e uma breve descrição. Também possuiria a funcionalidade de configurar um lembrete para o evento, podendo ser um alarme ou uma notificação, a visualização do evento na tela de calendários e uma listagem dos eventos a partir de um filtro mostraria os não concluídos, a data mais próxima, o tipo ou a disciplina.

O segundo épico, definido como Outros, possuiria 3 features com 8 histórias de usuário no total. Este épico abrangia as demais funcionalidades propostas, contendo as seguintes features:

1) Cores: A visualização de uma tela onde se poderia alterar a paleta de cores do aplicativo e a escolha de um alarme para o aplicativo.

2) Ajustes: O cadastro de outros tipos de eventos, a troca do primeiro dia da semana e o envio de sugestões para os desenvolvedores.

3) Avaliação/Divulgação: A indicação do aplicativo para um amigo, a avaliação e a leitura de algumas informações sobre o aplicativo.

Por conta de adversidades enfrentadas pela equipe, onde se encontrou grande dificuldade para fazer algumas histórias mais prioritárias, caracterizadas como must, foi necessário reformular o backlog para se enquadrar à nova prioridade de produção do projeto. O Backlog final ficou definido com apenas um épico, 3 features e 12 histórias de usuário, sendo 4 para cada feature.

As novas features anexaram partes das features antigas e deixaram de lado outras partes. Ficaram definidas da seguinte forma:

Épicos	Features	Story de Usuário
Agenda	Eventos	Visualizar eventos
		Cadastrar evento
		Definir nome, data, hora, disciplina (opcional), descrição, cor
	Disciplinas	Visualizar disciplinas
		Adicionar disciplina
		Definir nome da disciplina
		Excluir disciplina
		Definir professor, cor e editar disciplina
	Tarefas	Visualizar a lista de tarefas
		Adicionar tarefa
		Definir texto da tarefa
		Excluir tarefa

Fig. 2. Backlog Final

C. Sprints

As sprints foram o modo como o grupo conseguiu ter controle sobre o desenvolvimento e andamento do projeto. Com as sprints sendo uma vez por semana e contendo o maior número de integrantes durante as realizações delas.

Sprint 0 - Discussões iniciais sobre o projeto.

- Tarefas concluídas: Criar o Product Backlog; Documentação inicial; Criar o Github Pages.
- Pontos positivos: Sucesso na interação; Assimilação da equipe do que será feito; Retorno positivo do professor; integração do time.
- Pontos de melhoria: Dependência do Scrum master; Atrasos nas reuniões; Participação e comunicação.

Sprint 1 - Divisão em duplas e trios para criação de propostas de protótipos.

- Tarefas concluídas: Capacitação Figma; Proposta de Logo (baixa fidelidade); Proposta de Protótipo (baixa fidelidade).
- Pontos positivos: Resultados das duplas; Todo mundo integrou; Retorno positivo do professor; Integração do time.
- Pontos de melhoria: Organização; Atrasos nas reuniões.

Sprint 2

- Tarefas concluídas: Criar o ambiente de desenvolvimento do Software; Proposta de Protótipo (média fidelidade)).
- Pontos positivos: Avaliação do professor em relação ao protótipo.
- Pontos de melhoria: Atrasos nas reuniões; Dispersão da equipe.

Sprint 3 - Primeira interação e avaliação do professor sobre nosso protótipo no Figma e gravação das reuniões para membros ausentes e para revisão.

- Tarefas concluídas: Criação do ambiente de desenvolvimento; Capacitar a equipe para o uso do React-Native.
- Pontos positivos: Entrosamento da equipe; Feedback do professor em relação ao protótipo; Gravação; Criação do ambiente e upload do mesmo.
- Pontos de melhoria: Acumulou trabalho das sprints; A participação do PO; Melhorar organização para continuar o trabalho; Divisão das tarefas.

Sprint 4 - Familiarização com o ambiente de desenvolvimento.

- Tarefas concluídas: Documentação do protótipo de baixa e média fidelidade.
- Pontos positivos: Todos estão ambientados.
- Pontos de melhoria: Pontualidade nos encontros em geral; Comunicação; Participação.

Sprint 5 - Familiarização com o ambiente de desenvolvimento.

- Tarefas incompletas: US-Tela de eventos; US-Tela de disciplina; US-Tela de horário.
- Pontos positivos: Boa comunicação entre a maioria dos membros; Muita coisa foi aprendida; Design do aplicativo.

- Pontos de melhoria: Acelerar o desenvolvimento do protótipo.

Sprint 6 - Dificuldades na implementação do Backlog devido a problemas com as ferramentas.

- Tarefas concluídas: Apresentação - 2º Ponto de controle.
- Tarefas incompletas: US-Tela de eventos; US-Tela de disciplina; US-Tela de horário. Pontos positivos: Descanso.
- Pontos de melhoria: Acelerar o desenvolvimento do protótipo; Dispersão da equipe.

Sprint 7 - Avanço no funcionamento do aplicativo e alteração do backlog, substituindo a tela de eventos por uma tela de tarefas.

- Tarefas concluídas: US- Tela de eventos; US- Tela de disciplina.
- Pontos positivos: Adaptações.
- Pontos de melhoria: Participação.

Sprint 8 - Implementação do Async Storage e simplificação do projeto para a entrega.

- Tarefas concluídas: US - Tela de tarefas; Formulários; Melhorias e design.
- Pontos positivos: Conseguindo progredir com mais simplicidade.
- Pontos de melhoria: Atrasos nas reuniões; Dispersão da equipe.

D. Identidade Visual

O conjunto dos aspectos visuais deste trabalho é composto por:

- 1 Paleta de cores: Usando como critério a estimulação visual das cores quentes no ser humano, a cor vermelha foi escolhida com o objetivo de chamar a atenção do usuário, para que ele fique atento às suas atividades. Definindo a cor vermelha como principal, para não sobrecarregar o aplicativo, já que este já possui uma cor chamativa como base, cores neutras como preto, branco e cinza acompanham a composição;

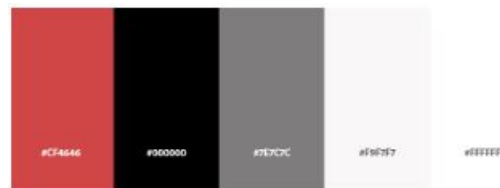


Fig. 3. Paleta de cor

- 2 Logomarca: A logo foi feita baseada na proposta do aplicativo de aproximação do usuário, fazendo uso de objetos escolares comumente utilizados, como cadernos, marcadores e lembretes, além seguir a paleta de cores definida anteriormente;

E. Protótipo

Com o objetivo de criar um modelo inicial do projeto foram feitos alguns protótipos de baixa fidelidade pelo grupo,



Fig. 4. Logomarca

seguindo o backlog. Por ser um rascunho do projeto, não foi necessário um desenvolvimento muito rebuscado dessa parte.

Após as deliberações, foi produzido um protótipo de média fidelidade para guiar na realização do aplicativo. Para a produção deste, foi usada a ferramenta Figma que permite uma criação visual mais fidedigna do aplicativo, além de permitir a interação entre telas.

Entretanto, devido às complicações na construção do software o protótipo de média fidelidade foi recriado com base nas novas features do backlog estabelecidas.

- 1 Protótipos de baixa fidelidade: Dividido o grupo em 3 duplas e apresentado 3 propostas diferentes, foi escolhida uma para basear os processos seguintes, no que caso foi a dupla João-Raquel:

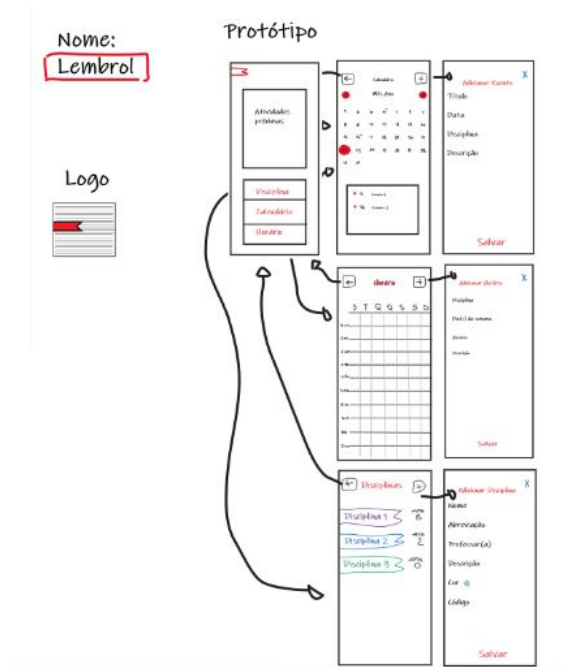


Fig. 5. Protótipo de baixa fidelidade - Joao Raquel

- 2 Protótipos de média fidelidade: A realização do protótipo de média fidelidade foi estabelecida como uma tarefa para quatro integrantes. Após criado o protótipo, este foi levado ao professor para análise, considerando as sugestões feitas, e com ambiente de desenvolvimento estabelecido a produção da aplicação foi iniciada. Porém, durante a

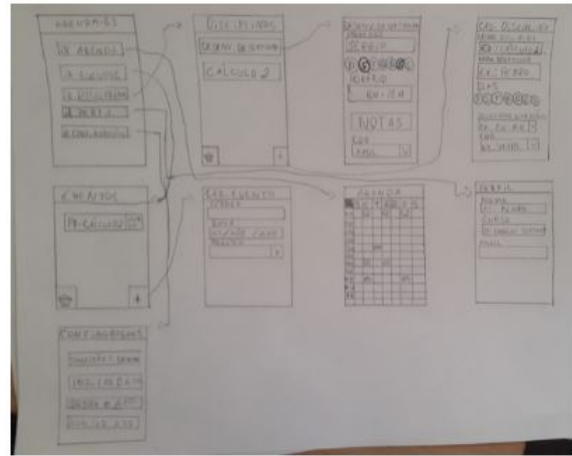


Fig. 6. Protótipo de baixa fidelidade - Hoton e Pedro



Fig. 7. Protótipo de baixa fidelidade - Raul e Eduardo

programação, por dificuldades e modificações no backlog, o protótipo de média foi modificado novamente.

F. Cronograma

O cronograma foi elaborado conforme o tempo disponível para a entrega dos pontos de controle realizado com o professor. Após a análise dos dias disponíveis para a construção do protótipo, dividimos as semanas para entregar as etapas da construção do projeto.

Da primeira até a quarta semana cada integrante da equipe ficou responsável por estudar a metodologia de Scrum, criação da documentação, mostrar a disponibilidade de horário para que houvesse o maior número de participantes na reunião semanal, trazer ideias de qual protótipo seria desenvolvido e qual ferramenta seria utilizada no desenvolvimento.

Da quinta até a nona semana foi elaborado pela equipe o desenvolvimento da identidade visual, nome do aplicativo e logo, também foi o período de capacitação dos membros por meio de cursos online do framework, React Native, juntamente com o desenvolvimento do software.

Da décima até a décima quarta semana foi realizada a revisão da documentação, manutenção no software com as modificações necessárias para que funcionasse corretamente, orientações de ajuste feitas pelo professor e ajuste dos últimos detalhes. Na décima quinta semana foi realizada a apresentação do protótipo e foi dado o feedback do que foi desenvolvido pela equipe.

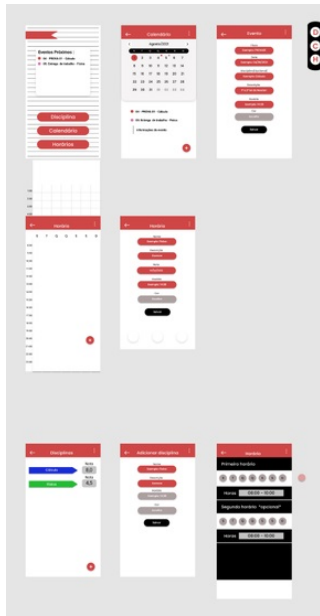


Fig. 8. Protótipo de Média Fidelidade

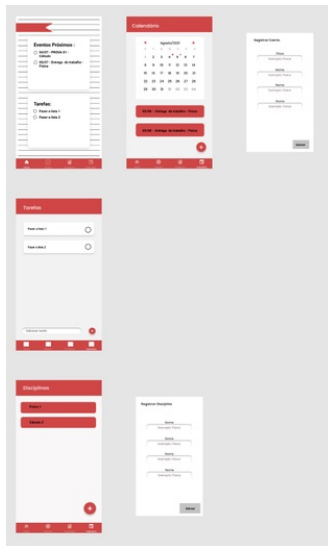


Fig. 9. Protótipo de Média Fidelidade Modificado (clique-me)

III. LIÇÕES APRENDIDAS

A maioria dos integrantes da equipe, são estudantes do 2 semestre, ou seja, iniciantes no ramo de desenvolvimento de software. Baseado nas experiências de integrantes mais avançados, foram escolhidas as ferramentas a serem utilizadas, uma decisão ousada do grupo visando um grande aprendizado.

Planejar e exercer o estudo das ferramentas e a cooperação de todos foi fundamental, principalmente para o projeto, e também para desenvolver habilidades necessárias para o mercado de trabalho na área de desenvolvimento.

O revezamento de funções dentro do método scrum mostrou ser outro grande aprendizado. Todos passaram pela função de Scrum-Master e Product Owner (PO) e tiveram as respons-

abilidades de um líder e dono do produto. Desse modo, cada participante teve que desenvolver as habilidades necessárias para exercer o cargo e tiveram uma perspectiva diferente do trabalho.

Quanto aos conhecimentos adquiridos, obtivemos um grande salto de conhecimento, lidamos com diversos problemas que foram essenciais para nosso progresso. Porém, da metade para o final do trabalho tais dificuldades foram limitantes e nos fizeram estagnar e, deste modo, para o cumprimento das datas nos adaptamos, precisamos realizamos a programação de modo mais simples que possibilitasse o alcance dos nossos objetivos.

Registrarmos nossos conhecimentos em tabelas e transformamos em gráficos para termos uma visão do andar da equipe.

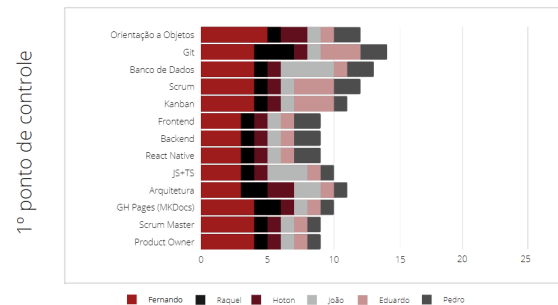


Fig. 10. Gráfico de conhecimentos - 1º ponto de controle

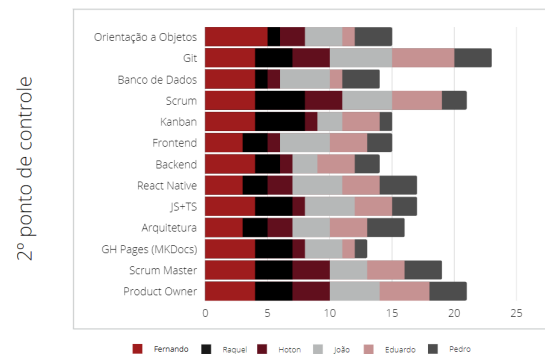


Fig. 11. Gráfico de conhecimentos - 2º ponto de controle e estagnação

IV. APLICATIVO

As telas do aplicativo ficaram da seguinte forma:

A. Tela de carregamento

Enquanto o aplicativo é carregado, a logo criada fica aparecendo até que esteja pronto para uso.

B. Tela Home

No topo existe uma figura que remete a um marcador de página. No centro da tela principal do aplicativo temos um bloco que apresenta as tarefas existentes, sendo possível "scrollar" as tarefas se passarem do espaço. E embaixo temos um painel de navegação que existe em todas as telas e possui ícones referente a tela que deseja ir.

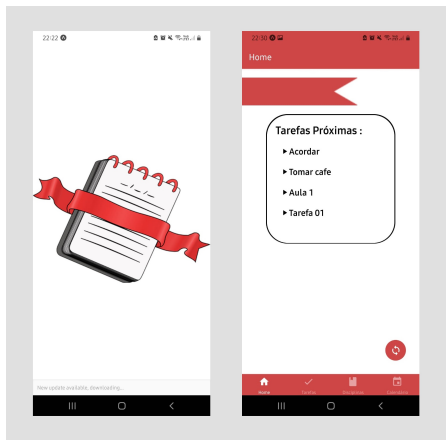


Fig. 12. Tela de carregamento e Home

C. Tela de Tarefas

Na parte inferior temos um campo de “input” que ao digitar a tarefa e clicar no botão inferior a direita, com um “+”, adicionamos as tarefas que ficam justapostas da parte superior até embaixo. Se existirem muitas tarefas é possível “scrollar” a tela. Tarefas como o mesmo texto não são possíveis de serem criadas, já com texto vazio é permitido.

D. Aba de exclusão

Quando o usuário tentar excluir uma tarefa o aplicativo pede para confirmar o exclusao para garantir que o usuário tem certeza sobre excluir a tarefa.

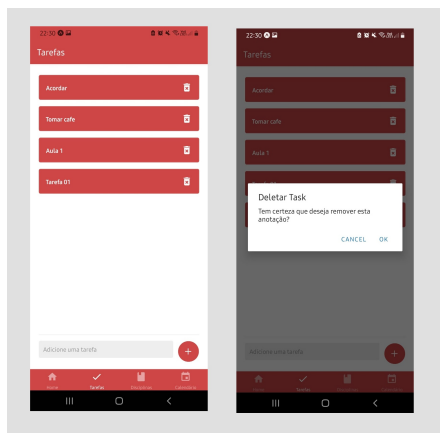


Fig. 13. Tela de Tarefa e aba de exclusão

E. Tela de disciplina

Ao selecionar a aba de Disciplinas é apresentada uma tela contendo as disciplinas adicionadas pelo usuário e um botão que abre uma janela de registro de disciplina. Para excluir determinada disciplina é preciso apenas clicar e segurar o item desejado.

F. Formulário de disciplina

Com apenas um campo de texto, o usuário pode inserir o nome da disciplina e para registrá-la na tela de disciplina é preciso apertar o botão inferior à direita escrito “salvar”.

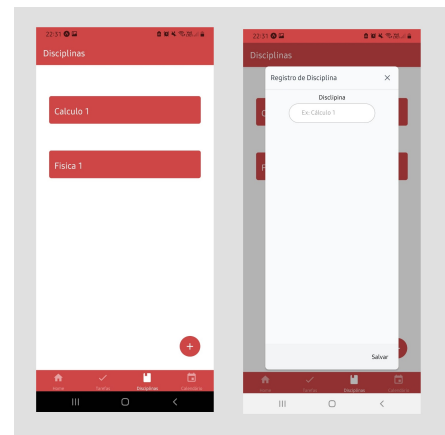


Fig. 14. Tela de disciplina e sua janela de registro

G. Tela de Evento

Ao selecionar a aba de calendário é apresentado um calendário com informando o dia atual, na parte superior, e dentro do calendário o usuário pode navegar pelos meses. Já do meio para baixo os eventos registrados ficam apresentados pelo usuário, e na parte inferior a direita é exibido um botão onde o usuário adiciona um novo evento. Para excluir determinado evento é preciso apenas clicar e segurar o item desejado.

H. Formulário de Evento

Abre uma janela de registro de evento com um espaço para ser adicionado uma data e um nome para o evento. E para registrá-lo na tela de evento é preciso apertar o botão inferior à direita escrito “salvar”.

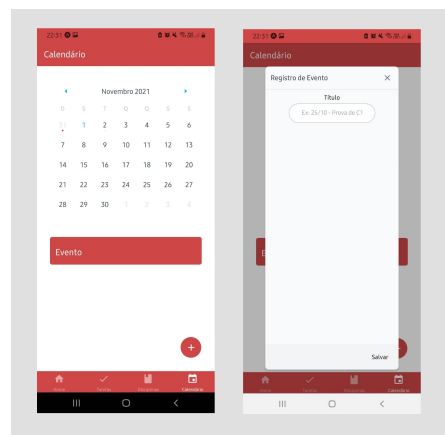


Fig. 15. Tela de evento e sua janela de registro

V. REFERÊNCIAS

- 1 F. Dessoldi, “Método SCRUM — Um resumo de tudo o que você precisa saber”, reprogramabr, <https://medium.com/reprogramabr/scrum-um-breve-resumo-f051e1bc06d9> (Current Oct. 30, 2021)
- 2 <https://github.com/SFernandoS/Agenda-G3>
- 3 <https://github.com/HacKairos/SuperAgenda>
- 4 <https://reactnative.dev/docs/getting-started> Roger Cruz, “Parte do curso de React Native + Expo que apresentei aos meus colegas”,
- 5 <https://rogercruz.medium.com/react-native-com-expo-um-exemplo-40e5574c6904> (Current Aug. 14, 2017)