ESCOLA SENAI

“PROF. DR. EURYCLIDES DE JESUS ZERBINI”

Ana Luiza Cardoso Podavi

Felipe Macedo Dos Santos

Matheus Lena Barros

Sérgio Borges Dos Santos Filho

**Yourself**

Campinas, SP

2024

Ana Luiza Cardoso Podavi

Felipe Macedo Dos Santos

Matheus Lena Barros

Sérgio Borges Dos Santos Filho

**Yourself**

Projeto apresentado à Escola SENAI “Prof. Dr. Euryclides de Jesus Zerbini” para obtenção do certificado de conclusão do Curso Técnico de Desenvolvimento de Sistemas.

Orientadores: Paulo Henrique Pansani e Renato de Mattos Onofre

Campinas, SP

2024

Ana Luiza Cardoso Podavi

Felipe Macedo Dos Santos

Matheus Lena Barros

Sérgio Borges Dos Santos Filho

**Yourself**

Trabalho de conclusão de curso aprovado como requisito parcial para obtenção do grau de técnico, do curso Técnico de Desenvolvimento de Sistemas da Escola SENAI “Prof. Dr. Euryclides de Jesus Zerbini”.

**BANCA EXAMINADORA**

1º Examinador

2º Examinador

3º Examinador

Campinas, SP

2024

**DEDICATÓRIA**

Dedicamos esse projeto a todos os nossos conhecidos que possuem dificuldades em organizar suas vidas acadêmicas, tais quais nos inspiraram a criar uma solução adequada que, além de auxiliar nesse quesito, aprimora a documentação do tempo de concentração e descanso.

**AGRADECIMENTOS**

Agradecemos imensamente aos nossos orientadores, Paulo e Renato, pela confiança depositada em nós, pela orientação paciente e pela dedicação em compartilhar seus conhecimentos. Sua expertise foi fundamental para o sucesso deste trabalho.

**EPÍGRAFE**

**“Se você fizer aquilo que a sua intuição manda,**

**usando bom senso e deixando de lado a vaidade, você**

**terá todas as possibilidades de alcançar seu objetivo”.**  
**Silvio Santos**

SUMÁRIO

[1 INTRODUÇÃO 8](#_Toc184308115)

[2 JUSTIFICATIVA 10](#_Toc184308116)

[3 OBJETIVOS 11](#_Toc184308117)

[3.1. Objetivos Gerais 11](#_Toc184308118)

[3.2. Objetivos Específicos 11](#_Toc184308119)

[4 PRODUCT BACKLOG 12](#_Toc184308120)

[5 REQUISITOS NÃO FUNCIONAIS 13](#_Toc184308121)

[6 PREMISSAS 14](#_Toc184308122)

[7 RESTRIÇÕES 15](#_Toc184308123)

[7.1. RESTRIÇÕES PARA FUNCIONAMENTO 15](#_Toc184308124)

[8 ANÁLISE DE RISCOS DE UM PROJETO 16](#_Toc184308125)

[8.1. Nível e Planos de Ação para os Riscos 16](#_Toc184308126)

[8.2. Planos de ação 16](#_Toc184308127)

[9 SPRINTS 18](#_Toc184308128)

[9.1. Primeiro Sprint 18](#_Toc184308129)

[9.1.1. Product Backlog 19](#_Toc184308130)

[9.1.2. Sprint Backlog 19](#_Toc184308131)

[9.1.3. Burn Down Chart 20](#_Toc184308132)

[9.1.4. Diagramas 20](#_Toc184308133)

[9.1.5. Kanban e Retrospectiva 20](#_Toc184308134)

[9.2. Segundo Sprint 26](#_Toc184308135)

[9.2.1 Product Backlog 27](#_Toc184308136)

[9.2.2 Sprint Backlog 27](#_Toc184308137)

[9.2.3 Burn Down Chart: 27](#_Toc184308138)

[9.2.4 Diagramas 28](#_Toc184308139)

[9.2.5 Kanban e Retrospectiva 28](#_Toc184308140)

[9.3 Terceiro Sprint 35](#_Toc184308141)

[9.3.1 Product Backlog 36](#_Toc184308142)

[9.3.2 Sprint Backlog 36](#_Toc184308143)

[9.3.3 Burn Down Chart: 37](#_Toc184308144)

[9.3.4 Diagramas 37](#_Toc184308145)

[9.3.5 Plano de Testes 37](#_Toc184308146)

[9.3.6 Kanban e Retrospectiva 40](#_Toc184308147)

[9.4 Quarto Sprint 46](#_Toc184308148)

[9.4.1 Product Backlog 46](#_Toc184308149)

[9.4.2 Sprint Backlog 46](#_Toc184308150)

[9.4.3 Burn Down Chart 46](#_Toc184308151)

[9.4.4 Plano de testes 47](#_Toc184308152)

[9.4.5 Kanban e Retrospectiva 47](#_Toc184308153)

[10 Modelo de Dados 49](#_Toc184308154)

[10.2 Modelo de Dados do Banco 49](#_Toc184308155)

[10.3 Dicionário de dados 51](#_Toc184308156)

[11 PRINCIPAIS TELAS DO SISTEMA 53](#_Toc184308157)

[12 CONCLUSÃO 62](#_Toc184308158)

[12.2 Escreva os resultados obtidos 62](#_Toc184308159)

[12.3 Constatações 62](#_Toc184308160)

[12.4 Sugestões de possíveis aperfeiçoamentos técnicos 62](#_Toc184308161)

[13 REFERÊNCIAS 63](#_Toc184308162)

[14 GLOSSÁRIO 64](#_Toc184308163)

# INTRODUÇÃO

De acordo com Francesco Cirillo, criador do método Pomodoro, ressalta ser essencial que, mesmo que você não consiga terminar a tarefa em 25 minutos, tire o tempo de descanso — durante esse período, não faça nada que requeira muito esforço mental. Você pode comer ou beber um copo d'água, ou assistir à TV.

A crescente complexidade da vida moderna exige cada vez mais organização e produtividade. Diante desse cenário, a busca por ferramentas e técnicas que auxiliem na gestão do tempo e das tarefas tornou-se uma necessidade.

A gamificação, que consiste na aplicação de elementos de jogos em contextos não lúdicos, emerge como uma promissora alternativa para tornar as atividades cotidianas mais engajadoras e motivacionais. Este trabalho tem como objetivo investigar a potencialidade da gamificação como ferramenta para auxiliar na organização das tarefas diárias, explorando o desenvolvimento e a avaliação de um sistema gamificado que incentive a criação de hábitos mais eficientes e produtivos.

Portanto, a Yourself tem como foco solucionar as problemáticas apresentadas acima, integrando diversas funcionalidades necessárias e não exploradas por outros aplicativos de estudo. Dessa maneira, os estudantes poderão aprender mais facilmente.

A gestão eficaz do tempo e a otimização da produtividade são desafios cada vez mais presentes no cotidiano das pessoas. A técnica Pomodoro, por exemplo, popularizada por Francesco Cirillo, demonstra a importância de intervalos regulares para manter a concentração e evitar a fadiga mental. No entanto, a aplicação de técnicas como essa pode ser desafiadora, especialmente em um mundo repleto de distrações.

A gamificação, ao incorporar elementos lúdicos em contextos não-lúdicos, emerge como uma estratégia promissora para aumentar a motivação e o engajamento dos usuários. Ao transformar tarefas cotidianas em desafios e jogos, a gamificação pode tornar a organização pessoal mais divertida e eficaz.

Este trabalho tem como objetivo desenvolver e avaliar um sistema gamificado, denominado Yourself, que visa auxiliar os usuários a gerenciar suas tarefas diárias de forma mais eficiente. Diferentemente de outros aplicativos, o Yourself se destaca pela sua capacidade de se adaptar às necessidades individuais de cada usuário, oferecendo um plano de estudos personalizado e recursos interativos que tornam o processo de aprendizagem mais prazeroso. Ao integrar elementos da técnica Pomodoro com mecânicas de gamificação, o Yourself busca proporcionar uma experiência mais completa e personalizada, auxiliando os usuários a alcançar seus objetivos de forma mais consistente e satisfatória.

# JUSTIFICATIVA

O crescente desafio enfrentado por estudantes de diversas idades em manter a concentração e a produtividade durante os estudos têm sido amplamente reconhecidos. Com a expansão da educação digital e a multiplicidade de distrações oferecidas pela tecnologia, torna-se cada vez mais difícil para os alunos gerenciarem seu tempo de forma eficaz. Neste contexto, o método Pomodoro surge como uma técnica amplamente reconhecida e eficaz para melhorar a concentração e a eficiência nos estudos, ao dividir o tempo de estudo em intervalos curtos e controlados, seguidos de breves pausas.

O aplicativo "Yourself" se propõe a incorporar essa técnica dentro de uma plataforma acessível e intuitiva, oferecendo suporte aos alunos que buscam maximizar seu rendimento acadêmico. Ao adotar o método Pomodoro, o aplicativo visa auxiliar os usuários a desenvolverem melhores hábitos de estudo, reduzir a procrastinação e, consequentemente, melhorar o desempenho escolar.

Além disso, o "Yourself" visa criar uma comunidade de apoio, onde os alunos podem compartilhar suas rotinas, dicas e progressos, promovendo um senso de pertencimento e motivação entre os usuários. Essa comunidade pode servir como uma rede de suporte mútua, onde os estudantes se incentivam e aprendem uns com os outros, tornando o processo de aprendizado mais colaborativo.

Por fim, a gamificação, que recompensa os usuários à medida que atingem metas de estudo, torna o processo mais dinâmico e envolvente. Esse aspecto lúdico pode ser especialmente eficaz para alunos mais jovens, ajudando-os a manter o interesse e o comprometimento com seus estudos.

A implementação desse aplicativo é, portanto, justificada pela necessidade de ferramentas educacionais que combinem eficiência e simplicidade, promovendo um ambiente de estudo mais focado e produtivo.

# OBJETIVOS

Segue abaixo os objetivos do projeto:

# Objetivos Gerais

Criar o projeto o Yourself e desenvolvê-lo com o objetivo de auxiliar as pessoas a se concentrarem nos seus estudos, utilizando a técnica Pomodoro, permitindo que elas tenham mais objetividade e foco na hora de estudar, com a opção de criar uma tarefa e escolher o tempo em que irá realizá-la e o tempo de descanso.

# Objetivos Específicos

* Produzir a documentação em formato ABNT;
* Criar uma aplicação mobile utilizando React Native;
* Utilizar a metodologia Scrum durante o projeto;
* Desenvolver o método de estudo dos usuários;
* Criar o manual do usuário;
* Realizar testes unitários;
* Criar o sistema de armazenamento da aplicação utilizando o banco de dados não relacional Firebase;
* Desenvolver um sistema de gamificação para motivar os usuários a atingir suas metas;
* Avaliar a interface e a experiência do usuário durante o desenvolvimento para identificar e corrigir possíveis problemas;
* Elaborar esboços da interface do usuário para visualizar a estrutura e o layout da aplicação.

# PRODUCT BACKLOG

RF01 - O usuário deve poder se cadastrar, informando email, nome, apelido e senha.

RF02 - O usuário deve poder entrar na plataforma com seu email e senha cadastrados anteriormente.

RF03 - O sistema deve manter uma sessão de usuário ativa por um período de, no mínimo, 3 horas após a última interação do usuário.

RF04 - O usuário deve poder cadastrar uma atividade, informando título, nível de dificuldade e categoria.

RF05 - O usuário deve ser capaz de, a qualquer momento, a partir de iniciada, após passado o tempo de 1 minuto no temporizador, finalizar uma atividade.

RF06 - O usuário deve poder pausar e despausar o temporizador acoplado à atividade.

RF07 - O usuário deve poder acessar o histórico de suas atividades nos últimos 2 meses.

RF08 - O usuário deve poder editar o tempo de concentração e descanso do temporizador acoplado às atividades.

RF09 O usuário deve ser capaz de excluir uma atividade criada anteriormente.

RF10: Ao finalizar uma atividade, o usuário deve ganhar pontos, os quais serão calculados a partir dos minutos de concentração que foram levados para concluir a tarefa multiplicados pelo nível de dificuldade estabelecido na criação dela. A pontuação máxima por atividade deve ser de 450 pontos.

RF11: O usuário deve poder gastar seus pontos adquiridos por meio de atividades em itens colecionáveis na loja do aplicativo. A partir do momento em que um item é comprado, este ficará disponível na tela de perfil do usuário, sendo exibido no formato de grade.

RF12: O usuário deve ter acesso às suas informações cadastradas anteriormente e poder, a partir de verificação de identidade, editá-las.

RF13: O sistema deve permitir que o usuário visualize dados específicos sobre seu desempenho: o total de pontos acumulados e a maior quantidade de dias consecutivos em que pelo menos uma atividade foi realizada.

# REQUISITOS NÃO FUNCIONAIS

RNF01 - O sistema deve ser intuitivo.

RNF02 - O sistema deve possuir um design atrativo.

RNF03 - O sistema deve apresentar ao usuário um contador visualizando a quantidade de dias consecutivos em que pelo menos uma atividade foi realizada.

RNF04 - O sistema deve apresentar a quantidade atual de pontos do usuário.

RNF05 - O aplicativo deve possuir um guia para o usuário, incluindo um tutorial inicial do uso do sistema.

# PREMISSAS

PRE01 - O projeto será planejado, desenvolvido e testado por 4 alunos, utilizando a metodologia ágil Scrum.

PRE02 - Serão disponibilizados, no mínimo, 4 computadores para o desenvolvimento do projeto.

PRE03 - O desenvolvimento dos componentes e telas do aplicativo será realizado utilizando o framework React Native. Os dados serão armazenados no banco de dados não-relacional Firestore, da plataforma Firebase, e a criação da API será feita usando o framework Express.

PRE04 - O método Pomodoro será uma das principais técnicas implementadas para auxiliar os usuários a manterem o foco e a concentração durante as atividades.

PRE05 - A interface do aplicativo será avaliada durante o desenvolvimento para garantir uma experiência intuitiva e de fácil utilização.

PRE06 - O sistema de gamificação será desenvolvido para motivar os usuários a atingirem suas metas de forma eficiente e lúdica.

# RESTRIÇÕES

RES01 - O projeto deverá ser entregue em 4 meses.

RES02 - O aplicativo deve atender aos padrões de qualidade definidos pela equipe.

RES03 - O aplicativo será desenvolvido para dispositivos android.

RES04 - Apenas 4 integrantes trabalharão no projeto.

# RESTRIÇÕES PARA FUNCIONAMENTO

* Software do aplicativo: Android 7.0 ou superior;
* Armazenamento: 2 GB ou mais;
* Memória RAM: 2 GB ou mais.

# ANÁLISE DE RISCOS DE UM PROJETO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Ameaça | Impacto | Probabilidade | Risco |
| 1 | Baixo domínio da tecnologia | 5 | 2 | 10 |
| 2 | Falta de acesso à internet | 5 | 2 | 10 |
| 3 | Falta de energia elétrica | 4 | 2 | 8 |
| 4 | Participação em eventos do SENAI | 2 | 4 | 8 |
| 5 | Ausência de integrantes | 4 | 2 | 8 |
| 6 | Compatibilidade tecnológica | 3 | 2 | 6 |
| 7 | Desempenho inadequado do computador | 3 | 1 | 3 |

# Nível e Planos de Ação para os Riscos

|  |  |  |  |
| --- | --- | --- | --- |
| **Impacto** | | **Probabilidade** | |
| Nível | Nome | Nível | Nome |
| 5 | Crítico | 5 | Altamente provável |
| 4 | Sério | 4 | Muito provável |
| 3 | Moderado | 3 | Provável |
| 2 | Leve | 2 | Pouco provável |
| 1 | Muito leve | 1 | Improvável |

# Planos de ação

ID1: Baixo domínio da tecnologia: Promover estudos dirigidos sobre as tecnologias utilizadas no projeto, dividindo os tópicos entre os membros para aprofundamento coletivo.

ID2: Falta de acesso à internet: Planejar envios frequentes dos arquivos do sistema ao repositório para evitar perda de dados em momentos de desconexão.

ID3: Falta de energia elétrica: Planejar tarefas offline, como revisão de código, leitura de documentação ou planejamento do projeto, para momentos sem eletricidade.

ID4: Participação em eventos do Senai: Identificar previamente o calendário de eventos e ajustar o cronograma do projeto para evitar sobreposições.

ID5: Ausência de integrantes: Estabelecer comunicação ativa para realocar tarefas em caso de ausência.

ID6: Compatibilidade tecnológica: Realizar uma análise detalhada das especificações de hardware e software das máquinas utilizadas no projeto e nos testes.

ID7: Desempenho inadequado do computador: Identificar problemas técnicos específicos e solicitar suporte técnico ao Senai ou aos responsáveis pela manutenção.

# SPRINTS

Apresenta-se nesta seção a documentação referente ao desenvolvimento do software.

# Primeiro Sprint

A Arch-tech estruturou a lógica dos sprints com base nas horas diárias de trabalho disponíveis, que, neste caso, foram 3 horas por dia. Também foi levada em conta a quantidade de membros da equipe, que era composta por 4 pessoas. Ao multiplicar esses valores, obteve-se o tempo total de trabalho diário, somando 12 horas.

• 3 horas x 4 integrantes = 12 horas diárias

Com esse total, foi calculada a quantidade de dias que o sprint teria, desconsiderando feriados, finais de semana e as aulas de Programação de Dispositivos Móveis. Assim, foi determinado um total de 144 horas de trabalho disponíveis para o primeiro sprint.

• 20 dias úteis → 240 horas no Sprint 1

Com base nisso, estabeleceu-se uma meta de 120 horas de trabalho, reservando 24 horas (16% do sprint) como margem para eventuais contratempos e atrasos. Para atingir essa meta, seriam necessárias aproximadamente 10 horas de trabalho diárias.

**Metas do Primeiro Sprint**

Para o Sprint 1, o grupo definiu as seguintes metas:

1. **Design das Telas:**
   * As telas planejadas incluem:
     + Tela inicial
     + Tela de login/cadastro
     + Tela home (cadernos)
     + Tela Tarefas
     + Tela de Shop
     + Tela de Profile
2. **Desenvolvimento Android:**
   * Para o Android, o foco foi o desenvolvimento do front-end das telas e a implementação de validações.
3. **Banco de Dados:**
   * O planejamento do banco de dados inclui a criação do modelo conceitual e lógico.
   * Implementação dos dados por meio de scripts no MySQL Workbench.

.

# Product Backlog

Não houve alterações no Product Backlog.

# Sprint Backlog

RF01 - O usuário deve poder se cadastrar, informando e-mail, nome, nome de usuário e senha.

RF04 - O usuário deve poder cadastrar uma atividade, informando título, nível de dificuldade e categoria.

# Burn Down Chart

# Diagramas

Apresenta os principais diagramas realizados.



# Kanban e Retrospectiva

A Arch-Tech desenvolveu um cronograma e um método de organização focados nos objetivos e no tempo do projeto. Nesse contexto, as tarefas foram divididas em categorias específicas e gerais. O tempo e os critérios de cada unidade curricular também foram organizados por Sprint, considerando sua relevância.

Seguindo essa abordagem, o kanban foi estruturado com as prioridades definidas da seguinte forma: foram separados os requisitos para cada unidade curricular, especificando o que deve ser realizado em banco de dados, Android e na parte web.

Além disso, foi criada uma seção para acompanhar o andamento de cada tarefa, proporcionando uma visão clara do progresso da equipe. O grupo também incluiu uma área dedicada ao que entrará na fase de testes, e, por fim, após passar por todas as etapas, os requisitos serão movidos para a categoria de concluído.

**Dia 0:**



**Semana 1:**



Na primeira semana foi colocado todas as tarefas em andamento e algumas tarefas foram concluídas, como concluir a estrutura geral sendo o design, modelar o banco, começar a criar a logo do aplicativo e desenvolver os requisitos não-funcionais

**Semana 2:**



Tudo que foi colocado em andamento na primeira semana foi concluída. Com o design concluído, falta a documentação.

**Semana 3:**



Nessa semana concluímos diversas telas importantes e começamos a documentação.

**Semana 4:**



Foi feito tudo que foi colocado em andamento, assim concluindo o primeiro Sprint.

**Retrospectiva:**

O grupo analisou os acontecimentos do sprint e chegaram as seguintes conclusões:

**Coisas que deram certo:**

Conseguimos consolidar a ideia do projeto, estabelecer as metas e objetivos, e chegamos a conclusão das tecnologias que vão ser utilizadas no projeto.

**O que deu errado:**

Houve contratempos com a modelagem do banco de dados, o primeiro levantamento de dados e tivemos mudanças no design.

**O que pode melhorar:**

A organização, planejamento em relação a tarefas

# Segundo Sprint

A Arch-tech organizou a lógica dos sprints baseada na quantidade de horas de trabalho disponíveis por dia, que, no caso, foram 3 horas. Além disso, foi considerada a quantidade de integrantes presentes na equipe, que eram 4. Assim, multiplicando os valores, foi possível calcular o tempo aproximado de trabalho diário, que somava 12 horas.

* 3 horas X 4 pessoas = 12 horas por dia

Após a contagem de horas, foi realizado o cálculo de dias que o sprint teria, desconsiderando feriados, finais de semana e as aulas de Programação de Dispositivos Móveis. A partir disso, foi encontrado um total de 240 horas.

* 20 dias uteis → 240 horas no Sprint 2

Dessa forma, foi definida uma meta de trabalho de 230 horas, disponibilizando 20 horas (12,5% do sprint) para possíveis imprevistos e atrasos. Portanto, para uma boa execução do sprint seria necessárias 10 horas de trabalho diárias.

**Metas do Segundo Sprint**

Para esse Sprint o grupo separou as seguintes metas a serem cumpridas:

1. **Construir os Componentes das Telas Do projetos.**
2. **Fazer as Rotas Essenciais da Api**

# Product Backlog

Não houve alterações no Product Backlog.

# Sprint Backlog

RF01 - O usuário deve poder se cadastrar, informando e-mail, nome, nome de usuário e senha.

RF04 - O usuário deve poder cadastrar uma atividade, informando título, nível de dificuldade e categoria.

# Burn Down Chart:

# Diagramas

Apresenta os principais diagramas realizados.



|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSUO1 - Cadastro  **Objetivo:** Permitir o registro de um usuário no aplicativo.  **Frequência estimada:** Uma vez por usuário.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** Ao entrar no aplicativo, o usuário acessa a tela de login. 2. O aplicativo exibe a tela que comporta nome de usuário, e-mail, senha e confirmar senha. 3. O usuário preenche os campos com seus dados. 4. O aplicativo valida os dados para registrá-lo no banco de dados. |
| **CENÁRIO ALTERNATIVO** |
| 1. **Não preenchimento dos campos:** Caso o usuário não preencha todos os campos obrigatórios (e-mail, nome, apelido, senha e confirmar senha), o sistema exibirá uma mensagem informando que os campos obrigatórios não foram preenchidos. 2. **Campos “Senha” e “Confirmar senha” com valores diferentes:** Caso as senhas presentes no campo “Senha” e “Confirmar senha” difiram, o sistema exibirá uma mensagem de erro e solicitará correção antes de permitir o registro. 3. **Senha fraca:** Caso a senha não possua, pelo menos, um caractere especial, uma letra minúscula e uma letra maiúscula, o sistema exibirá uma mensagem informando que o usuário crie uma senha que atenda a esses requisitos. 4. **Apelido existente:** Se o usuário tentar se cadastrar com um apelido que já existe na plataforma, ele será informado que e o apelido já existe, não permitindo o cadastro. 5. **Email já cadastrado:** Caso o usuário tente se registrar com um e-mail já cadastrado na plataforma, ele será informado que o usuário já existe e não será permitido o cadastro. |
| **PRÉ-CONDIÇÕES** |
| 1. O usuário deve possuir uma conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. O usuário será redirecionado para a inicial. |

# Kanban e Retrospectiva

A Arch-Tech elaborou um cronograma e um método de organização alinhados aos objetivos do projeto. Dessa forma, as atividades foram categorizadas em tarefas específicas e gerais. O tempo e os critérios de cada unidade curricular foram organizados em Sprints, considerando sua importância.

Com base nessa abordagem, as prioridades no Kanban foram estruturadas da seguinte forma: foram identificadas as necessidades nas áreas de documentação, front-end. e back-end. Também foi criado um espaço para acompanhar o progresso de cada uma, proporcionando uma visão clara do avanço da equipe. Ao final de cada fase, os requisitos são movidos para a seção de concluídos.

Nesta Sprint, o grupo optou por uma organização diferente em relação à primeira. Assim, as tarefas foram divididas por semanas, o que impactou a forma como o Kanban será apresentado.

**Dia 0:**

****

****

**Semana 01:**

****

Na primeira semana nós já havíamos finalizado as premissas e restrições do projeto, elaboramos os componentes de tela, começamos a estilizar as telas de cadastro e login, processo de criação para o cadastro do usuário e o diagrama de classes.

**Semana 02:**

****

Em nossa segunda semana de projeto no segundo sprint, foi finalizada a tela de login, a tela de cadastro estava em processo, iniciamos as telas de atividade ativa e histórico de atividade.

**Semana 03:**



Na terceira semana foi iniciado o carregamento do histórico do app, componente de tela do perfil e a documentação dos Sprints

**Semana 04:**



Semana quatro, sprint dois já finalizado.

**Retrospectiva**

O grupo analisou os acontecimentos do sprint e chegaram as seguintes conclusões:

**Coisas que deram certo:**

* Organizaçãodo Tempo**:** A equipe calculou corretamente o total de horas disponíveis, somando 240 horas, o que permitiu uma boa gestão do tempo.
* Divisãode Tarefas**:** As metas foram claramente definidas, como a construção dos componentes das telas e a configuração das rotas, o que facilitou o acompanhamento do progresso.
* Documentação**:** O diagrama de classes e a documentação dos Sprints foram finalizados, proporcionando uma base sólida para o desenvolvimento.

**Coisas que deram errado:**

* ImprevistosTécnicos**:** Durante o Sprint, a equipe enfrentou problemas de compatibilidade entre os sistemas operacionais utilizados nas máquinas, o que atrasou algumas atividades.
* Faltade Conectividade: Algumas sessões de trabalho foram impactadas pela instabilidade da internet, dificultando o acesso a recursos online e a comunicação em tempo real.
* Atrasosnas Entregas: Algumas tarefas, como a finalização da tela de cadastro, não foram concluídas no prazo estabelecido, o que causou um efeito cascata nas entregas subsequentes.

**O que pode melhorar:**

* Preparaçãopara Imprevistos**:** Criar um plano de contingência mais robusto para lidar com problemas de conexão e energia, como ter materiais offline e opções de trabalho em equipe que não dependam da internet.
* Treinamentoe Suporte**:** Oferecer sessões de capacitação sobre as ferramentas e tecnologias que serão utilizadas, para minimizar o baixo domínio da tecnologia.
* Revisãodas Metas**:** Revisar as metas do Sprint para torná-las mais realistas, considerando a carga de trabalho e o tempo disponível. Isso pode ajudar a evitar atrasos e frustrações.
* FeedbackContínuo**:** Implementar sessões de feedback mais frequentes durante o Sprint, permitindo ajustes imediatos e uma melhor adaptação às dificuldades encontradas.

# Terceiro Sprint

A Arch-tech organizou a lógica dos sprints com base nas horas de trabalho disponíveis por dia, que foram estipuladas em 3 horas. Além disso, foi considerada a quantidade de integrantes da equipe, totalizando 4 pessoas. Com isso, ao multiplicar os valores, foi possível estimar o tempo total de trabalho diário em 12 horas.

3 horas x 4 pessoas = 12 horas por dia (NO SENAI).

Após o levantamento das horas, foi feito o cálculo dos dias de trabalho no sprint, desconsiderando feriados, finais de semana e as aulas de Programação de Dispositivos Móveis. O resultado foi um total de 240 horas.

20 dias úteis → 240 horas no Sprint 3.

Com base nisso, foi estabelecida uma meta de trabalho de 150 horas, reservando 90 horas (37,5% do sprint) para possíveis imprevistos e atrasos. Portanto, para uma boa execução do sprint, seriam necessárias cerca de 10 horas de trabalho diárias.

Para este sprint, o grupo definiu as seguintes metas: desenvolver as telas, como a loja, perfil e atividades já prontas. Isso inclui desde a parte de design, codificação, responsividade, até a integração com o modelo MVC e conexão das funções ao banco de dados. Além disso, foi planejado finalizar toda a parte de back-end (lógica) das classes flashcard, arquivo e pomodoro.

**Metas do Terceiro Sprint**

Para esse Sprint o grupo separou as seguintes metas a serem cumpridas:

* **Começar a fazer os testes**
* **E consumir as rotas**

# Product Backlog

* Não houve alterações no Product Backlog.

# Sprint Backlog

* RF11: O usuário deve poder gastar seus pontos adquiridos por meio de atividades em itens colecionáveis na loja do aplicativo. A partir do momento em que um item é comprado, este ficará disponível na tela de perfil do usuário, sendo exibido no formato de grade.
* RF12: O usuário deve ter acesso às suas informações cadastradas anteriormente e poder, a partir de verificação de identidade, editá-las.
* RF13: O sistema deve permitir que o usuário visualize dados específicos sobre seu desempenho: o total de pontos acumulados e a maior quantidade de dias consecutivos em que pelo menos uma atividade foi realizada.

# Burn Down Chart:

# Diagramas

Apresenta os principais diagramas realizados.

|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSU02 - Login**.**  **Objetivo:** Permitir que um usuário entre no aplicativo.  **Frequência estimada:** Todas as vezes que for necessário acessar o sistema.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** O caso de uso inicia quando o usuário abre o aplicativo. 2. **Tela de Login:** O sistema exibe a tela de login, contendo campos para inserção de e-mail e senha, além de um botão para login. 3. **Inserção das Credenciais:** O usuário insere o e-mail e a senha cadastrados no aplicativo. 4. **Submissão:** O usuário clica no botão de "Login". 5. **Validação:** O sistema de autenticação verifica se as credenciais inseridas são válidas. 6. **Acesso Autorizado:** Com as credenciais validadas, o sistema autoriza o login e carrega o ambiente da tela inicial. 7. **Redirecionamento:** O usuário é redirecionado para a tela principal do aplicativo, onde ele pode conferir suas atividades recentes. |
| **CENÁRIO EXCESSÃO** |
| 1. **Inserção de Credenciais:** O usuário insere um e-mail e/ou senha que não correspondem aos dados cadastrados. 2. **Tentativa de Autenticação:** O usuário clica no botão de login, e o sistema tenta validar as credenciais. 3. **Erro de Autenticação:** O sistema detecta que o e-mail ou senha não são válidos. Então o sistema exibe uma mensagem de erro: "E-mail ou senha incorretos. Tente novamente." 4. **Novo Login:** O usuário retorna à tela de login, podendo tentar novamente com credenciais corretas. |
| **PRÉ-CONDIÇÕES** |
| 1. O usuário já criou uma conta no aplicativo. 2. O usuário deve possuir conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. O usuário está autenticado no sistema e pode utilizar o aplicativo para gerenciar suas atividades. |

|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSU03 – Criar Atividade  **Objetivo:** O objetivo principal é oferecer uma funcionalidade que permita ao usuário organizar seu tempo, facilitando a criação de tarefas que serão acompanhadas por ciclos de concentração e descanso, ajudando a evitar distrações e melhorando a gestão de tempo.  **Frequência estimada:**  A qualquer momento.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** O usuário se autentica e navega até a tela de criação de atividades. 2. **Tela de criação de atividade:** O aplicativo exibe um formulário para o usuário inserir os detalhes da nova atividade. Campos disponíveis:    * + Título da atividade      + Descrição da atividade (opcional)      + Nível de dificuldade da atividade      + Categoria da atividade 3. **Preenchimento dos dados:** O usuário preenche o título da atividade, nível de dificuldade e categoria (obrigatório) e, se desejar, uma breve descrição. 4. **Validação de dados:** O sistema valida se os campos obrigatórios estão preenchidos corretamente. Caso haja algum campo obrigatório não preenchido, o sistema exibe uma mensagem de alerta solicitando o preenchimento. 5. **Confirmação:** O usuário pressiona o botão “Iniciar” para confirmar o início da atividade. 6. **Gravação da atividade:** O usuário, ao se passar 1 minuto no temporizador, será capacitado de finalizar a nova atividade clicando no botão “Finalizar”. 7. **Fim do caso de uso:** Ao clicar em “Finalizar”, o usuário é direcionado à tela inicial, a qual listará suas atividades recentes, incluindo a que ele acabou de finalizar. |
| **CENÁRIO ALTERNATIVO** |
| 1. **Fluxo Alternativo 1 – Dados incompletos:** Se o usuário tentar salvar a atividade sem preencher os campos obrigatórios, o sistema exibe uma mensagem de erro e solicita o preenchimento correto dos campos. 2. **Fluxo Alternativo 2 – Cancelamento da criação:** O usuário pode optar por cancelar a criação da atividade a qualquer momento antes de finalizar, retornando à tela anterior sem que os dados de tempo sejam gravados**.** |
| **CENÁRIO EXCESSÃO** |
| 1. **Falha na validação de dados obrigatórios:** O sistema detecta que campos obrigatórios, como "Título", “Nível de dificuldade” e “Categoria”, estão vazios ou preenchidos incorretamente. Então o sistema exibe uma mensagem de erro, solicitando o preenchimento correto. O usuário corrige os dados e tenta novamente. 2. **Falha na conexão com o banco de dados:** O sistema não consegue acessar o banco de dados para salvar a atividade. Então o sistema exibe uma mensagem de erro e tenta reconectar automaticamente ou solicita ao usuário tentar mais tarde. 3. **Sessão expirada:** O sistema detecta que a sessão do usuário expirou. Então exibe uma mensagem pedindo que o usuário faça login novamente. Após o login, o usuário pode retornar à criação da atividade. 4. **Cancelamento pelo usuário:** O usuário decide cancelar a realização da atividade. Então ele pressiona o botão de voltar e é direcionado à tela anterior. |
| **PRÉ-CONDIÇÕES** |
| 1. O usuário deve estar autenticado no aplicativo. 2. O usuário deve possuir conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. Uma nova atividade é criada e listada na tela de atividades recentes. |

|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSU04 – Consultar Estatísticas  **Objetivo:**  Permitir que o usuário acesse seus dados de maior sequência de dias realizando uma atividade e total de pontos adquiridos.  **Frequência estimada:**  A qualquer momento.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** O usuário abre o aplicativo e navega até a tela “Perfil”. 2. **Visualização da ofensiva:** Na tela “Perfil”, há uma sessão que possibilita a visualização da maior ofensiva realizada pelo usuário. 3. **Exibição dos pontos totais:** Na tela “Perfil”, há uma outra sessão que possibilita a visualização do total de pontos conquistados pelo usuário. 4. **Finalização:** Após visualizar as estatísticas, o usuário pode retornar à tela inicial ou criar uma atividade. |
| **CENÁRIO EXCESSÃO** |
| 1. **Detecção do Erro:** O usuário identifica a falha em suas estatísticas, notando que as suas recentes atividades não foram contabilizadas. Isso pode ocorrer devido como:  * Problemas de conexão com o banco de dados. * Usuário não autenticado. * Falha no cálculo automático de pontuações.  1. **Ação Alternativa do Usuário:** O usuário pode tentar sair da sua conta e entrar novamente. 2. **Continuação:** O usuário pode retornar à tela de “Perfil” e acessar as suas informações novamente. |
| **PRÉ-CONDIÇÕES** |
| 1. O usuário deve estar autenticado no aplicativo. 2. O usuário deve possuir conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. O usuário obtém uma visão clara de seu progresso e desempenho no aplicativo, incentivando-o a continuar aumentando sua quantidade de pontos e manter sua ofensiva. |

|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSUO5 - Consultar Atividades.  **Objetivo:** Permitir que o usuário consulte suas atividades.  **Frequência estimada:** A qualquer momento.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** O caso de uso inicia quando o usuário se loga no aplicativo. 2. **Visualização das atividades recentes:** Ao se logar, o usuário é direcionado à tela de atividades, onde ele consegue acessar suas 5 atividades recentes. Essas atividades são organizadas em cartões contendo o título, o nível de dificuldade e a quantidade de pontos adquiridos com a atividade. 3. **Visualização do histórico**: Abaixo das atividades recentes, há um texto “Ver mais” que, ao ser clicado, exibe todas as atividades realizadas nos últimos 5 meses. |
| **CENÁRIO EXCESSÃO** |
| 1. **Nenhuma atividade cadastrada:** Caso o usuário não tenha nenhuma atividade realizada, aparecerá uma mensagem informando-o que não há atividades a serem mostradas. |
| **PRÉ-CONDIÇÕES** |
| 1. O usuário deve estar autenticado no aplicativo. 2. O usuário deve possuir conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. As atividades permanecem inalteradas, sem impacto em suas propriedades ou status. |

|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSU06 – Excluir Atividade  **Objetivo:**  Permitir que o usuário exclua a atividade.  **Frequência estimada:**  Conforme a necessidade.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** O usuário se autentica e acessa a tela inicial, onde visualiza uma lista de suas tarefas. 2. **Seleção da Atividade:** O usuário identifica uma atividade que deseja excluir. 3. **Ação de Exclusão:** O usuário toca na atividade. 4. **Confirmação de Exclusão:** O aplicativo exibe uma janela de confirmação perguntando se o usuário tem certeza de que deseja excluir a atividade. O usuário pode optar por "Excluir" ou "Cancelar". Se o usuário escolher "Excluir", a lista de atividades é atualizada automaticamente para refletir a mudança. Caso o usuário escolha "Cancelar", a janela de confirmação é fechada e o usuário retorna à lista de atividades sem alterações. |
| **CENÁRIO ALTERNATIVO** |
| 1. **Atividade Não Encontrada:** Se o usuário tenta excluir uma atividade que já foi removida ou não existe, o aplicativo exibe uma mensagem de erro: "Atividade não encontrada." |
| **PRÉ-CONDIÇÕES** |
| 1. O usuário deve estar autenticado no aplicativo. 2. O usuário deve possuir conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. A atividade é removida da lista do usuário e o aplicativo reflete a nova lista de tarefas. |

|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSU07 – Editar Perfil  **Objetivo:**  Permitir que o usuário modifique suas informações.  **Frequência estimada:** Conforme a necessidade.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** O usuário se autentica e navega até a tela de perfil. 2. **Acessando as configurações:** O usuário clica no botão “Configurações”. Ao fazer isso, ele é direcionado à tela de confirmação de identidade. 3. **Confirmação de identidade:** O usuário coloca a sua senha atual e, caso correta, ao clicar em “Avançar”, é direcionado à tela de configurações. 4. **Atualizando dados:** O usuário pode alterar as informações referentes ao nome, apelido e senha, inserindo novos valores. Caso a senha seja alterada, o usuário precisará se logar novamente. 5. **Usuário envia as alterações:** O usuário clica no botão “Salvar” para confirmar as mudanças. 6. **Sistema salva as alterações:** Se todas as informações enviadas forem válidas, o sistema atualiza o perfil do usuário. 7. **Feedback:** O sistema mostra uma mensagem de sucesso indicando que as alterações foram realizadas com sucesso. |
| **CENÁRIO ALTERNATIVO** |
| 1. **Campos “Senha” e “Confirmar senha” com valores diferentes:** Se o usuário optar por atualizar a senha, ele precisará confirmar a nova senha no campo “Confirmar senha”. Caso as senhas difiram, o sistema exibirá uma mensagem de erro e solicitará correção antes de permitir a atualização. 2. **Senha fraca:** Caso a nova senha não possua, pelo menos, um caractere especial, uma letra minúscula e uma letra maiúscula, o sistema exibirá uma mensagem informando que o usuário crie uma senha que atenda a esses requisitos. 3. **Apelido existente:** Se o usuário tentar atualizar seu apelido para um que já existe na plataforma, ele será informado que e o apelido já existe, não permitindo a atualização. |
| **PRÉ-CONDIÇÕES** |
| 1. O usuário deve estar autenticado no sistema. 2. O usuário deve possuir conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. O perfil do usuário é atualizado com o novo nome, apelido e/ou senha. 2. O usuário pode fazer login com a nova senha (se alterada). |

|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| **Sigla/Nome:**  CSUO8 - Editar Temporizador Pomodoro**.**  **Objetivo:** Permitir que o usuário ajuste as durações das sessões de concentração e descanso.  **Frequência estimada:** Periodicamente.  **Ator Principal:** Usuário. |
| **CENARIO PRINCIPAL** |
| 1. **Início:** Ao acessar a tela “Configurações”, abaixo do botão “Atualizar”, há a chamada “Editar temporizador pomodoro” que, caso pressionada, leva o usuário para a tela de edição do temporizador. 2. **Usuário modifica as informações:** O usuáriopreenche os campos que ele deseja atualizar, os quais são referentes ao tempo associado à concentração e ao descanso nos ciclos pomodoros ao iniciar uma tarefa. 3. **Usuário confirma as alterações:** Após modificar os tempos desejados, o usuário clica no botão "Salvar". 4. **Sistema salva as alterações:** O sistema atualiza as configurações de temporizador do usuário e armazena os novos valores. 5. **Feedback:** O sistema mostra uma mensagem de sucesso indicando que as alterações foram salvas. |
| **CENÁRIO ALTERNATIVO** |
| 1. **Erro na validação dos tempos:** Se o usuário inserir valores fora do intervalo permitido, o sistema exibe uma mensagem de erro. Ou seja, caso ele insira valores abaixo de 1 minuto, será lançado um alerta pedindo a correção dos valores. |
| **PRÉ-CONDIÇÕES** |
| * + - 1. O usuário deve estar autenticado no sistema.       2. O usuário deve possuir conexão com a internet. |
| **PÓS-CONDIÇÕES** |
| 1. O temporizador Pomodoro é atualizado com os novos tempos configurados pelo usuário. 2. As próximas atividades utilizarão os temporizadores atualizados. |

* + 1. Plano de Testes

O plano de teste é uma maneira de encontrar defeitos e bugs no sistema para serem futuramente corrigidos. Mesmo que durante a programação sempre foram realizados alguns, ainda sim no plano são realizados testes finais passo a passo.

|  |  |  |
| --- | --- | --- |
| Plano de Testes de Login | | |
| PASSOS | RESULTADO ESPERADO | RESULTADO OBTIDO |
| digitar um login cadastrado no banco, mas com a senha errada. | o sistema deverá mostrar a mensagem de erro ao logar, pois o nome e a senha devem estar iguais ao efetuar o cadastro. | O sistema mostra a mensagem de erro “senha incorreta”. |
| Digitar login e senha que não foram cadastrados no banco. | o sistema deverá mostrar a mensagem de erro ao logar, pois o nome e a senha devem estar iguais ao efetuar o cadastro. | O sistema mostra a mensagem de erro “usuário não cadastrado”. |
| Deixar o campo de senha vazio. | o sistema deverá mostrar a mensagem de erro ao logar, pois o nome e a senha devem estar preenchidos. | O sistema mostra que é necessário preencher o campo de senha. |
| Deixar o campo de login vazio. | o sistema deverá mostrar a mensagem de erro ao logar, pois o nome e a senha devem estar preenchidos. | O sistema mostra que é necessário preencher o campo de login. |
| digitar uma senha cadastrada no banco, mas com o login errado. | o sistema deverá mostrar a mensagem de erro ao logar, pois o nome e a senha devem estar iguais ao efetuar o cadastro | O sistema mostra a mensagem de erro “usuário não cadastrado”. |
| digitar um login e senha correspondentes cadastrados no banco. | o sistema deverá entrar na tela inicial do usuário. | O sistema entra na página inicial do usuário. |

|  |  |  |
| --- | --- | --- |
| Plano de Testes de Cadastro | | |
| PASSOS | RESULTADO ESPERADO | RESULTADO OBTIDO |
| Deixar os campos Email, nome, apelido e senha vazios. | O sistema deve retornar status 400 e mensagem de erro: "Todos os campos são obrigatórios". | O sistema retorna status 400 e mensagem de erro ("Todos os campos são obrigatórios"). |
| Digitar uma senha com menos de 6 caracteres. | O sistema deve retornar status 400 e mensagem de erro: "A senha deve ter pelo menos 6 caracteres". | O sistema retorna status 400 e mensagem de erro (A senha deve ter pelo menos 6 caracteres). |
| Usar um apelido já existente. | O sistema deve retornar status 400 e mensagem de erro: "Apelido já existe". | O sistema retorna status 400 e mensagem de erro (Apelido já existe). |
| Usar um Email já cadastrado. | O sistema deve retornar status 400 e mensagem de erro: "Usuário já cadastrado". | O sistema retorna status 400 e mensagem de erro (Usuário já cadastrado). |
| Preencher todos os campos corretamente. | O sistema deve chamar o próximo middleware, permitindo o cadastro do usuário. | O sistema chama o próximo middleware. |
| Deixar o campo apelido vazio, mas preencher os outros campos. | O sistema deve retornar status 400 e mensagem de erro: "Todos os campos são obrigatórios". | O sistema retorna status 400 e mensagem de erro (Todos os campos são obrigatórios). |
| Deixar o campo senha vazio, mas preencher os outros campos. | O sistema deve retornar status 400 e mensagem de erro: "Todos os campos são obrigatórios". | O sistema retorna status 400 e mensagem de erro (Todos os campos são obrigatórios.). |

|  |  |  |
| --- | --- | --- |
| Plano de Testes de Criação de Atividade | | |
| PASSOS | RESULTADO ESPERADO | RESULTADO OBTIDO |
| Deixar os campos **título da lição**, **dificuldade**, **categoria** e **descrição** vazios. | O sistema deve retornar a mensagem de erro: "Por favor, preencha o campo título". | O sistema retornou a mensagem de erro: "Por favor, preencha o campo título". |
| Preencher apenas o campo **Título da atividade** e deixar os outros campos vazios. | O sistema deve retornar a mensagem de erro: "Por favor, preencha o campo categoria". | O sistema retorna a mensagem de erro (A senha deve ter pelo menos 6 caracteres). |
| Selecionar uma **dificuldade** válida (por exemplo, clicando no botão para selecionar "Fácil", "Normal" ou "Difícil") e deixar os outros campos vazios. | O sistema deve retornar a mensagem de erro: "Por favor, preencha o campo título". | O sistema retornou a mensagem de erro: "Por favor, preencha o campo título". |
| Preencher os campos **título da atividade categoria**, **descrição**, mas não clicar no botão de **dificuldade**. | O sistema deve retornar a mensagem de erro: "Por favor, preencha o campo dificuldade". | O sistema retornou a mensagem de erro: "Por favor, preencha o campo dificuldade". |
| Preencher os campos **nome da lição**, **dificuldade**, **categoria**, mas deixar o campo **descrição** vazio. | O sistema deve permitir o cadastro, mas com descrição em branco. | O sistema chamou a tela de pomodoro. |
| Preencher todos os campos corretamente: **nome da lição**, **dificuldade**, **categoria**, **descrição**. | O sistema deve cadastrar a atividade e, passar para a tela de pomodoro. | O sistema cadastrou a atividade e passou para a tela de pomodoro. |

|  |  |  |
| --- | --- | --- |
| Plano de Testes de Consulta de Atividade | | |
| PASSOS | RESULTADO ESPERADO | RESULTADO OBTIDO |
| Consultar as atividades com o usuário logado. | O sistema exibe apenas as atividades pertencentes ao usuário logado. | O sistema exibiu corretamente as atividades do usuário logado. |
| Consultar as atividades sem estar logado no sistema. | O sistema não permite a consulta, exibindo uma mensagem de erro indicando que é necessário fazer login. | O sistema exibiu a mensagem: "É necessário fazer login para consultar as atividades." |
| Consultar atividades de mais de 2 meses atrás. | O sistema exibe apenas atividades dos últimos 2 meses, ignorando as mais antigas. | O sistema exibiu corretamente apenas atividades criadas nos últimos 2 meses.. |
| Tentar consultar as atividades de outro usuário. | O sistema não permite a consulta, exibindo apenas atividades do usuário logado. | O sistema exibiu apenas as atividades pertencentes ao usuário logado, sem acesso às de outros usuários. |

* + 1. Kanban e Retrospectiva

A Arch-Tech desenvolveu um cronograma e um método de organização focados nos objetivos e no tempo do projeto. Assim, as tarefas foram divididas em categorias específicas e gerais. O tempo e os critérios de cada unidade curricular também foram organizados por Sprint, considerando sua relevância.

Seguindo essa abordagem, o kanban foi estruturado com as prioridades definidas da seguinte forma: foram separados os requisitos para cada unidade curricular, especificando o que deve ser realizado em banco de dados, Android e na parte web.

Além disso, foi criada uma seção para acompanhar o andamento de cada tarefa, proporcionando uma visão clara do progresso da equipe. O grupo também incluiu uma área dedicada ao que entrará na fase de testes, e, por fim, após passar por todas as etapas, os requisitos serão movidos para a categoria de concluído.

**Semana 1**

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Na primeira semana focamos no Node e em criar as narrativas de caso de uso.

**Semana 2**

Imagem de texto no celular

Descrição gerada automaticamente com confiança média

Na segunda semana finalizamos o Node e a loja.

**Semana 3**

Tela de celular com publicação numa rede social

Descrição gerada automaticamente

Na terceira terminou as narrativas e começou a consumir as rotas.

**Semana 4**

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Na quarta foi iniciado os planos de testes.

**Semana 5**

Interface gráfica do usuário, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Na quinta foi finalizado os planos de testes e de consumir as rotas.

**Retrospectiva**

A Arch-Tech elaborou um cronograma e um método de organização alinhados aos objetivos do projeto. Nesse contexto, as atividades foram categorizadas em tarefas específicas e gerais. O tempo e os critérios de cada unidade curricular foram organizados em Sprints, considerando sua importância.

Com base nessa abordagem, as prioridades no Kanban foram estruturadas da seguinte forma: foram identificadas as necessidades nas áreas de documentação, front-end e back-end. Também foi criado um espaço para acompanhar o progresso de cada uma, proporcionando uma visão clara do avanço da equipe. Ao final de cada fase, os requisitos são movidos para a seção de concluídos.

Nesta terceira Sprint, o foco foi na integração e consumo das rotas das telas, com o objetivo de garantir que a comunicação entre o front-end e o back-end fosse eficiente e sem falhas. O grupo manteve a divisão das tarefas por semanas, impactando a forma como o Kanban é apresentado, de modo a refletir claramente o progresso das implementações**.**

# Quarto Sprint

No quarto sprint, a Arch-tech estruturou a organização das atividades levando em consideração as horas de trabalho disponíveis por dia, definidas como 3 horas. Também foi levado em conta o número de integrantes da equipe, totalizando 4 pessoas. Dessa forma, ao multiplicar as horas diárias pelo número de membros, obteve-se um total de 12 horas de trabalho diário.

3 horas x 4 pessoas = 12 horas por dia (no SENAI).

Com base nisso, foi calculado o total de horas do sprint, desconsiderando feriados, finais de semana e as aulas de Programação de Dispositivos Móveis. O total resultante foi de 240 horas de trabalho.

20 dias úteis → 240 horas no Sprint 4.

Para otimizar a execução do sprint, foi estabelecida uma meta de 150 horas de trabalho, deixando 90 horas (37,5% do sprint) para possíveis imprevistos e atrasos. Assim, para garantir um bom andamento das atividades, seria necessário dedicar cerca de 10 horas de trabalho por dia.

Nesse último sprint o grupo decidiu focar na conclusão do app e na parte do final do back, apenas finalizando o trabalho para entrega-lo. Portanto o grupo colocou como prioridade finalizar o back, o editar do perfil criar um timer funcional.

# Product Backlog

# Sprint Backlog

# Burn Down Chart

# Plano de testes

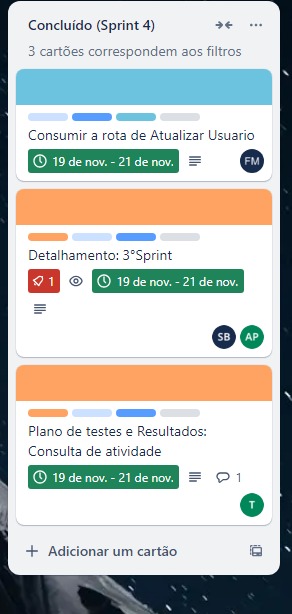
|  |  |  |
| --- | --- | --- |
| Plano de Testes de Editar Perfil | | |
| PASSOS | RESULTADO ESPERADO | RESULTADO OBTIDO |
| 1. Deixar os campos Nome, Apelido, Nova senha e confirmar senha vazios. | O sistema deve retornar mensagem de erro: " Preencha, pelo menos, um campo!". | O sistema retorna ("Preencha, pelo menos, um campo!"). |
| 2. Preencher o campo "Nova senha" sem as especificações ditas | O sistema deve retornar mensagem de erro: " Sua senha deve conter: • pelo menos 8 caracteres • pelo menos uma letra maiúscula • pelo menos um número • pelo menos um caractere especial (!@#$%^&\*(),.?":{}|<>)” | O sistema retorna mensagem de erro " Sua senha deve conter: • pelo menos 8 caracteres • pelo menos uma letra maiúscula • pelo menos um número • pelo menos um caractere especial (!@#$%^&\*(),.?":{}|<>)” |
| 4. Preencher "Confirmar senha" com valor diferente de "Nova senha". | O sistema deve retornar mensagem de erro: "As senhas não coincidem". | O sistema retorna status 400 e mensagem de erro (As senhas não coincidem). |
| 5. Preencher todos os campos corretamente. | O sistema deve chamar a próxima tela, atualizando os dados do usuário e mandando ele ir fazer login de novo. | O sistema chama a próxima tela. atualizando os dados do usuário e mandando ele ir fazer login de novo. |
| 7. Deixar o campo "Confirmar senha" vazio, mas preencher os outros campos. | O sistema deve retornar mensagem de erro: "senha não coincidem". | O sistema retornou mensagem de erro: "senha não coincidem". |

# Kanban e Retrospectiva

No quarto sprint, a Arch-tech seguiu uma abordagem similar à do sprint anterior, utilizando um método de organização baseado no tempo disponível e na distribuição de tarefas entre os membros da equipe. A equipe levou em consideração que cada integrante trabalharia 3 horas por dia, totalizando 12 horas diárias de trabalho. Considerando o total de 240 horas de trabalho no sprint, o grupo estabeleceu uma meta de 150 horas, reservando 90 horas para imprevistos e atrasos. O foco principal do sprint foi a conclusão do aplicativo, com ênfase na finalização do back-end e na implementação de funcionalidades como o editor de perfil e a criação de um timer funcional.

Com base nisso, o grupo organizou o kanban para o quarto sprint da seguinte maneira:

**Semana 1**



# Modelo de Dados

Esta parte do planejamento traz informações necessárias para a construção de um banco de dados para o sistema Yourself.

# Modelo de Dados do Banco

Usuários: Array

[

{

anoRegistro: Int

apelido: String

maiorOfensiva: Int

nome: String

ofesiva: Int

preferenciaConcentração: Int

preferenciaDescanso: Int

TotalPontos: Int

email: String

}

]

Itens: Array

[

{

Id: Int

Nome:string

Preco: Int

Icone: string

}

]

Atividade: Array

[

{

Categoria: String

Titulo: String

Data: timestamp

Descrição: String

Dificuldade: Int

Email: String

Id: String

Pontos: Int

tempoConcentração: Int

}

]

# Dicionário de dados

|  |  |  |
| --- | --- | --- |
| **Coleção: usuarios** | | |
| **Descrição: Armazena as informações dos usuários. Cada documento é identificado pelo email do usuário.** | | |
| Campo | Tipo de Dados | Descrição |
| anoRegistro | number | Ano em que o usuário foi registrado. |
| apelido | string | Apelido do usuário. |
| atividades | array (string) | IDs das atividades associadas ao usuário. |
| email | string | Email do usuário (ID do documento). |
| itens | array (string) | IDs dos itens possuídos pelo usuário. |
| maiorOfensiva | number | Maior ofensiva alcançada pelo usuário. |
| nome | string | Nome completo do usuário. |
| ofensiva | number | Ofensiva atual do usuário. |
| pontos | number | Pontuação atual do usuário. |
| preferenciaConcentracao | number | Tempo de concentração preferido (em minutos). |
| preferenciaDescanso | number | Tempo de descanso preferido (em minutos). |
| senhaHash | string | Hash da senha do usuário. |
| totalPontos | number | Pontuação total acumulada pelo usuário. |
| ultimaAtividade | string | ID da última atividade realizada pelo usuário. |

|  |  |  |
| --- | --- | --- |
| **Coleção: itens** | | |
| **Descrição: Armazena os itens disponíveis no sistema. Cada documento é identificado por um ID aleatório gerado pelo Firestore.** | | |
| Campo | Tipo de Dados | Descrição |
| icone | string | URL ou referência ao ícone do item. |
| id | string | ID único do item. |
| nome | string | Nome do item. |
| preco | number | Preço do item em pontos. |

|  |  |  |
| --- | --- | --- |
| **Coleção: atividades** | | |
| **Descrição: Armazena as atividades realizadas ou disponíveis no sistema. Cada documento é identificado por um ID aleatório gerado pelo Firestore.** | | |
| Campo | Tipo de Dados | Descrição |
| categoria | string | Categoria da atividade. |
| data | timestamp | Data e hora da realização ou criação da atividade. |
| descricao | string | Descrição da atividade. |
| dificuldade | number | Nível de dificuldade da atividade. |
| email | string | Email do usuário responsável pela atividade. |
| id | string | ID único da atividade. |
| pontos | number | Pontos atribuídos pela realização da atividade. |
| tempoConcentracao | number | Tempo de concentração necessário (em minutos). |
| titulo | string | Título da atividade. |

# PRINCIPAIS TELAS DO SISTEMA

* Tela de cadastro



* Tela de login



* Tela inicial



* Tela de nova atividade



* Tela de atividade ativa



* Tela da loja



* Tela de perfil



* Tela de atualização do perfil



* Tela de edição do temporizador



# CONCLUSÃO

Pelo apresentado no projeto, podem-se concluir que, ao utilizar técnicas e métodos de estudo, é possível ser mais produtivo em tarefas de carga importante e cotidiana. Dando ênfase no método pomodoro, que se mostra, evidentemente, uma técnica de importante relevância para acadêmicos, otimizando o tempo dos estudos e os fazendo mais produtivos. Ao aplicar a técnica é inteligível sua eficácia, cronometrando o tempo e pausas, foi possível relativizar ciclos de tarefas pendentes, comprovando a técnica criada por Francesco Cirillo, em 1980.

Há estudos que demonstram a procrastinação acadêmica como um problema comum entre os jovens atuais, relacionado a questões na interação entre o indivíduo, a tarefa e o contexto em que ele está inserido, o que inclui o ambiente, fatores pessoais e a situação em que se encontra no momento. Esses elementos interagem de maneira dinâmica, criando uma rede de fatores que contribuem para o fator. Compreender essa interação é essencial para desenvolver estratégias eficazes de enfrentamento, que vão desde o aprimoramento da gestão do tempo e do autocontrole até intervenções no ambiente e no formato das tarefas propostas.

O aplicativo se mostra uma ferramenta eficaz para combater a procrastinação quando ajuda os estudantes a enfrentarem desafios como a falta de foco, a sobrecarga de tarefas e a dificuldade em administrar o tempo. Ao criar um ambiente estruturado e previsível para o trabalho, o aplicativo auxilia e leva o usuário a superar a procrastinação e ansiedade em relação ao volume de tarefas ou a distração com fatores externos. Além disso, ao permitir que o usuário ajuste o ritmo de trabalho às suas necessidades, o aplicativo oferece uma flexibilidade que torna o processo de estudar mais adaptável e menos intimidante, contribuindo para uma maior produtividade.

# Escreva os resultados obtidos

Ao longo da implementação de uso do método Pomodoro, foi possível observar resultados significativos que demonstram sua eficácia em aumentar a produtividade acadêmica e gerenciar a procrastinação. Nós observamos maior foco durante os ciclos de estudo e trabalho, com as pausas estruturadas ajudando a reduzir a fadiga mental e permitindo maior consistência na realização das tarefas.

Além disso, a aplicação da técnica de cronometragem com ciclos curtos e intervalos promoveu uma melhor organização do tempo, facilitando a divisão de tarefas complexas em partes menores e mais gerenciáveis. Isso resultou em uma percepção positiva sobre a capacidade de cumprir prazos e metas.

Outro ponto observado foi a redução da procrastinação. A estrutura do método Pomodoro funcionou como um estímulo para iniciar as atividades, ajudando os participantes a superarem a inércia inicial e manterem-se engajados. O uso do aplicativo, em conjunto com a técnica, contribuiu para criar um ambiente mais propício ao trabalho, minimizando distrações externas e aumentando a confiança para lidar com o volume de tarefas. Esses resultados comprovam que a aplicação de estratégias bem planejadas pode ser eficaz para melhorar o desempenho acadêmico e enfrentar desafios comuns como a procrastinação e a falta de foco.

# Constatações

Com base nos resultados obtidos, constata-se que o método Pomodoro se mostrou uma técnica eficaz para promover uma abordagem mais organizada e produtiva em atividades acadêmicas e profissionais. Sua estrutura simples e prática facilitou a gestão do tempo, proporcionando ciclos de trabalho mais focados e eficientes, além de períodos de descanso que ajudam a evitar a fadiga mental.

Outro aspecto relevante foi o impacto positivo na organização das tarefas, permitindo que os usuários dividissem atividades complexas em etapas menores e mais gerenciáveis. Esse método reforçou a criação de rotinas mais consistentes e alinhadas aos objetivos dos participantes.

Por fim, percebeu-se que a flexibilidade da técnica possibilitou sua aplicação em diferentes contextos e demandas individuais, demonstrando sua adaptabilidade e versatilidade como uma ferramenta prática para melhorar o desempenho e criar hábitos mais sustentáveis de estudo e trabalho.

# Sugestões de possíveis aperfeiçoamentos técnicos

* **Alarmes**

Poder colocar alarmes personalizados para tocar nos finais dos ciclos.

* **Relatórios Anuais**

O app disponibilizaria um relatório dos seus anos, com sua evolução, mostrando gráficos do tempo de sua concentração. O usuário poderá baixar o relatório

* **Conectar outros aplicativos**

Conseguir logar sua conta de outros aplicativos no seu perfil do app.

* **Estimativa de tempo de conclusão**
* Obtenha uma estimativa do tempo necessário para concluir suas tarefas diárias.

# REFERÊNCIAS

O que é o 'Método Pomodoro' de gestão de tempo que conquistou fãs ao redor do mundo por melhorar produtividade. **BBC News Brasil,** 2021. Disponível em: https://www.bbc.com/portuguese/geral-55563393. Acesso em: 06 jan. 2021.

CIRILLO, F. **A Técnica Pomodoro (O Pomodoro)**. Tradução: Elaine Cristina Rocha de Oliveira. PomodoroTechnique.pdf

TANCREDI, Silvia. "Técnica Pomodoro - o que é e como funciona"; *Brasil Escola*. Disponível em: https://brasilescola.uol.com.br/dicas-de-estudo/tecnica-pomodoro-que-e-e-como-funciona.htm. Acesso em 06 de dezembro de 2024.

KARINA, R.; CRISTINA, I. Procrastinação acadêmica: um estudo exploratório. Estudos Interdisciplinares em Psicologia, v. 2, n. 2, p. 242–262, 2024.

DA. Redes sociais e concentração: O desafio da atenção fragmentada | UNINASSAU. Disponível em: <https://www.uninassau.edu.br/noticias/redes-sociais-e-concentracao-o-desafio-da-atencao-fragmentada>.

# GLOSSÁRIO

**\*Pomodoro**: A Técnica Pomodoroé um método de gerenciamento de tempo desenvolvido por Francesco Cirillo no final dos anos 1980. A técnica consiste na utilização de um cronômetro para dividir o trabalho em períodos de 25 minutos, separados por breves intervalos.

\***Kanban:** Kanban é um sistema visual de gestão de trabalho que utiliza cartões ou sinais visuais para controlar e gerenciar o fluxo de trabalho. Ele representa o processo de produção e permite uma gestão organizada e compreensível para todos os envolvidos. O método visa otimizar os fluxos de trabalho e identificar possíveis gargalos no processo.

**Scrum**: Metodologia ágil de gerenciamento de projetos que utiliza iterações chamadas "sprints" para desenvolvimento incremental. Inclui papéis como Product Owner, Scrum Master e equipe de desenvolvimento, além de reuniões diárias e retrospectivas.

**Repositório**: Repositório é um local centralizado para armazenar e gerenciar arquivos, dados e recursos.