

UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Pedro Henrique Vestena Rossato

**DESENVOLVIMENTO DO SISTEMA DE AGENDAMENTO  
DE SESSÕES PARA O PLANETÁRIO DA UNIVERSIDADE  
FEDERAL DE SANTA MARIA (SASP-UFSM)**

Santa Maria, RS

2022

**Pedro Henrique Vestena Rossato**

**DESENVOLVIMENTO DO SISTEMA DE AGENDAMENTO DE SESSÕES PARA O PLANETÁRIO DA UNIVERSIDADE FEDERAL DE SANTA MARIA (SASP-UFSM)**

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Ciência da Computação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para a obtenção do grau de **Bacharel em Ciência da Computação**

Orientadora: Prof. Dra. Lisandra Manzoni Fontoura

Vestena Rossato, Pedro Henrique

Desenvolvimento do Sistema de Agendamento de Sessões para o Planetário da Universidade Federal de Santa Maria (SASP-UFSM) / por Pedro Henrique Vestena Rossato. – 2022.

68 f.: il.; 30 cm.

Orientadora: Lisandra Manzoni Fontoura

Trabalho de Conclusão de Curso - Universidade Federal de Santa Maria, Centro de Tecnologia, Bacharelado em Ciência da Computação, RS, 2022.

1. Desenvolvimento. 2. Incremental. 3. Iterativo. 4. Avaliação.  
I. Manzoni Fontoura, Lisandra. II. Desenvolvimento do Sistema de Agendamento de Sessões para o Planetário da Universidade Federal de Santa Maria (SASP-UFSM).

---

© 2022

Todos os direitos autorais reservados a Pedro Henrique Vestena Rossato. A reprodução de partes ou do todo deste trabalho só poderá ser feita mediante a citação da fonte.


E-mail: phrossato@inf.ufsm.br

**Pedro Henrique Vestena Rossato**

**DESENVOLVIMENTO DO SISTEMA DE AGENDAMENTO DE SESSÕES PARA O PLANETÁRIO DA UNIVERSIDADE FEDERAL DE SANTA MARIA (SASP-UFSM)**

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Ciência da Computação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para a obtenção do grau de **Bacharel em Ciência da Computação**

**Aprovado em 15 de Fevereiro de 2022:**



---

**Lisandra Manzoni Fontoura, Dra. (UFSM)**  
(Presidente/Orientadora)



---

**Luís Alvaro de Lima Silva, Dr. (UFSM)**



---

**Eduardo Kessler Piveta, Dr. (UFSM)**

Santa Maria, RS

2022

## AGRADECIMENTOS

À minha mãe, **Marivane Vestena Rossato**, e meu pai, **Alexandre Rossato**, pelos ensinamentos e incentivos de sempre.

À minha irmã, **Manuela Vestena Rossato**, pelos bons e divertidos momentos, dos quais sempre me lembrarei.

Ao meu amor, **Giovana Stefani**, pelo amor e carinho essenciais em minha vida, e por todo apoio e incentivo ao longo desta caminhada, para que eu pudesse realizar este trabalho da melhor forma possível.

À minha orientadora, **Dra. Lisandra Manzoni Fontoura**, por todos os conselhos, ensinamentos e tempo dedicado a mim e a minha formação.

Aos membros da banca, **Dr. Eduardo Kessler Piveta** e **Dr. Luís Alvaro de Lima**, pelo compromisso de fazer parte da banca deste trabalho, e pelas contribuições e sugestões para o melhoramento do mesmo.

À **equipe do Planetário da Universidade Federal de Santa Maria** pelo apoio e colaboração para que a realização deste trabalho se tornasse possível.

À **Universidade Federal de Santa Maria (UFSM)** pela estrutura e suporte ao longo de minha graduação, proporcionando meu crescimento pessoal e profissional.

## RESUMO

### DESENVOLVIMENTO DO SISTEMA DE AGENDAMENTO DE SESSÕES PARA O PLANETÁRIO DA UNIVERSIDADE FEDERAL DE SANTA MARIA (SASP-UFSM)

AUTOR: PEDRO HENRIQUE VESTENA ROSSATO  
ORIENTADORA: LISANDRA MANZONI FONTOURA

Ao tomar conhecimento, por meio de relações interpessoais, das necessidades específicas do Planetário da UFSM, no que diz respeito ao processo de agendamento de sessões, notou-se a inexistência de um sistema que atenda tais necessidades. Assim, o objetivo geral deste trabalho é desenvolver o Sistema de Agendamento de Sessões do Planetário da Universidade Federal de Santa Maria (SASP-UFSM). Especificamente, o trabalho busca: (a) desenvolver o sistema de maneira iterativa e incremental, para manter entregas frequentes aos *stakeholders*; (b) manter operacional o sistema, desde a primeira versão, sem despesas ao desenvolvedor e ao Planetário da UFSM; (c) avaliar a usabilidade do sistema desenvolvido com os *stakeholders*; e (d) avaliar a satisfação dos *stakeholders* em relação ao processo de desenvolvimento e à expectativa dos mesmos em relação ao sistema desenvolvido. O processo de desenvolvimento iterativo e incremental foi realizado em conjunto com práticas e artefatos de diferentes métodos ágeis, sendo que cada iteração envolveu o ciclo de desenvolvimento de duas semanas. A implementação resultou no acatamento da grande maioria das necessidades identificadas, como a geração de relatórios sobre os agendamentos, o agendamento de sessões presenciais e virtuais, e em horários diferentes dos pré-definidos pelos monitores, bem como a avaliação de uma sessão já realizada. Destaca-se que os métodos de hospedagem proporcionaram um sistema gratuito com alta disponibilidade e que a avaliação realizada pelos *stakeholders* mostrou que o sistema atendeu todas as características de usabilidade, bem como que houve grande satisfação com o processo de desenvolvimento e à expectativa do sistema desenvolvido.

**Palavras-chave:** Desenvolvimento. Incremental. Iterativo. Avaliação.

## ABSTRACT

### DEVELOPMENT OF THE SESSION SCHEDULING SYSTEM FOR THE PLANETARIUM OF THE FEDERAL UNIVERSITY OF SANTA MARIA (SASP-UFSM)

AUTHOR: PEDRO HENRIQUE VESTENA ROSSATO

ADVISOR: LISANDRA MANZONI FONTOURA

Upon becoming aware, through interpersonal relationships, of the specific needs of the UFSM Planetarium, with regard to the process of scheduling sessions, it was noted the inexistence of a system that meets these needs. Thus, the general objective of this work is to develop the Session Scheduling System of the Planetarium of the Federal University of Santa Maria (SASP-UFSM). Specifically, the work seeks to: (a) develop the system iteratively and incrementally, to maintain frequent deliveries to stakeholders; (b) keep the system operational, from the first version, free of charge to the developer and the UFSM Planetarium; (c) evaluate the usability of the system developed with stakeholders; and (d) assess stakeholder satisfaction in relation to the development process and their expectations in relation to the developed system. The iterative and incremental development process was carried out in conjunction with practices and artifacts from different agile methods, with each iteration involving a two-week development cycle. The implementation resulted in the fulfillment of the vast majority of the identified needs, such as the generation of reports on the appointments, the scheduling of in-person and virtual sessions, and at times different from those pre-defined by the monitors, as well as the evaluation of a session that has already been held. It is noteworthy that the hosting methods provided a free system with high availability and that the evaluation carried out by the stakeholders showed that the system met all the usability characteristics, as well as that there was great satisfaction with the development process and the expectation of the developed system.

**Keywords:** Development. Incremental. Iterative. Rating.

## LISTA DE FIGURAS

FIGURA 1	Estrutura de uma História de Usuário .....	16
FIGURA 2	Ciclo TDD .....	17
FIGURA 3	O Modelo Incremental .....	20
FIGURA 4	Ciclo completo da iteração .....	25
FIGURA 5	Arquitetura do Sistema .....	27
FIGURA 6	Botão Publicar do Visual Studio .....	28
FIGURA 7	Conteúdo da Página Inicial da Primeira Iteração .....	34
FIGURA 8	Componente Calendário .....	34
FIGURA 9	Código do Componente Calendário .....	35
FIGURA 10	Funções de Filtro de Horários e Datas Indisponíveis .....	36
FIGURA 11	E-mails Enviado ao Planetário .....	37
FIGURA 12	Pop-up de Cancelamento de Agendamento .....	38
FIGURA 13	Acordeões de Sessões no Formulário de Agendamento .....	40
FIGURA 14	Formulário de login .....	41
FIGURA 15	Botão de Logout da Área do Monitor .....	41
FIGURA 16	Tabela de Agendamentos da Área do Monitor .....	42
FIGURA 17	Ações na Tabela de Agendamentos da Área do Monitor .....	44
FIGURA 18	Tabela de Sessões da Área do Monitor .....	44
FIGURA 19	Tabela de Horários Disponíveis da Área do Monitor .....	44
FIGURA 20	Modal de Ação .....	45
FIGURA 21	Botões de Encaminhamento na Página Inicial do Sistema .....	46
FIGURA 22	Mensagem de Aviso de Sem Horarios Diponiveis .....	46
FIGURA 23	Caixa de Seleção e Seletor .....	48
FIGURA 24	Tabela de Agendamentos Pendentes de Aprovação .....	48
FIGURA 25	E-mails de Aprovação ou Recusa .....	49
FIGURA 26	Consultas de Informações Necessárias ao Arquivo CSV .....	51
FIGURA 27	Arquivo CSV Gerado .....	52
FIGURA 28	Modal de Avaliação de Sessão .....	52
FIGURA 29	Tabela de Avaliações da Área do Monitor .....	53
FIGURA 30	Diagrama Entidade-Relacionamento .....	53
FIGURA 31	Histórias de Usuário por Iteração .....	55
FIGURA 32	Tempo de Disponibilidade do Sistema .....	56
FIGURA 33	Avaliação de Usabilidade .....	57
FIGURA 34	Avaliação de Satisfação .....	58



## LISTA DE TABELAS

TABELA 1	Histórias de Usuários da Primeira Iteração .....	33
TABELA 2	Histórias de Usuários da Segunda Iteração .....	39
TABELA 3	Histórias de Usuários da Terceira Iteração .....	43
TABELA 4	Histórias de Usuários da Quarta Iteração.....	47
TABELA 5	Histórias de Usuários da Quinta Iteração.....	50
TABELA 6	Histórias de Usuários de Todas as Iterações .....	64

## LISTA DE ABREVIATURAS E SIGLAS

UFSM	Universidade Federal de Santa Maria
XP	Extreme Programming
TDD	Test-Driven Development
JS	JavaScript
GUI	Graphical User Interface
JSX	JavaScript XML
SSL	Secure Sockets Layer
API	Application Programming Interface
SQL	Structured Query Language
AWS	Amazon Web Services
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
XML	Extensible Markup Language
ACID	Atomicity, Consistency, Isolation, Durability
ISO	International Organization for Standardization
SMTP	Simple Mail Transfer Protocol

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	11
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> .....	14
2.1	ENGENHARIA DE SOFTWARE .....	14
2.2	ENGENHARIA DE SOFTWARE ÁGIL .....	14
<b>2.2.1</b>	<b><i>Extreme Programming (XP)</i></b> .....	16
<b>2.2.2</b>	<b>Scrum</b> .....	18
<b>2.2.3</b>	<b>Kanban</b> .....	19
2.3	DESENVOLVIMENTO INCREMENTAL .....	19
2.4	FERRAMENTAS .....	20
<b>2.4.1</b>	<b>Desenvolvimento</b> .....	21
<b>2.4.2</b>	<b>Hospedagem</b> .....	21
2.5	ALTERNATIVAS .....	22
<b>3</b>	<b>DESENVOLVIMENTO DO SASP-UFSM</b> .....	24
3.1	PROCESSO DE DESENVOLVIMENTO .....	24
<b>3.1.1</b>	<b>Reunião de Revisão e Priorização do <i>Backlog</i> de Produto</b> .....	26
<b>3.1.2</b>	<b>Planejamento</b> .....	26
<b>3.1.3</b>	<b>Implementação e Verificação</b> .....	27
<b>3.1.4</b>	<b>Implantação</b> .....	28
<b>3.1.5</b>	<b>Validação</b> .....	29
3.2	HOSPEDAGEM DO SISTEMA .....	29
3.3	AVALIAÇÃO DO SISTEMA .....	30
<b>4</b>	<b>PROJETO E IMPLEMENTAÇÃO</b> .....	32
4.1	PRIMEIRA ITERAÇÃO .....	32
4.2	SEGUNDA ITERAÇÃO .....	38
4.3	TERCEIRA ITERAÇÃO .....	42
4.4	QUARTA ITERAÇÃO .....	47
4.5	QUINTA ITERAÇÃO .....	49
4.6	ESTADO FINAL DO BANCO DE DADOS .....	53
<b>5</b>	<b>RESULTADOS E AVALIAÇÃO</b> .....	55
<b>6</b>	<b>CONCLUSÃO</b> .....	59
	<b>REFERÊNCIAS</b> .....	62
	<b>APÊNDICE A - HISTÓRIAS DE USUÁRIO</b> .....	64

# 1 INTRODUÇÃO

O uso de um *software* pode trazer inúmeros benefícios para uma organização, tais como a solução de problemas e a automatização de processos, exigindo menor esforço por parte das pessoas. Neste sentido, os sistemas *web* vem ganhando cada vez mais espaço entre os *softwares* desenvolvidos, visto que atualmente cada vez mais serviços podem ser realizados de modo *online* em qualquer dispositivo com acesso à *Internet*.

Com base nisso, a proposta deste trabalho é desenvolver um sistema *web* que auxilie o processo de agendamento de sessões do Planetário da Universidade Federal da Santa Maria (UFSM). O Planetário da UFSM, fundado em 14 de dezembro de 1971, é um local desenvolvido para simular o céu em prol do conhecimento de seus visitantes (UNIVERSIDADE FEDERAL DE SANTA MARIA, 2021). São realizadas sessões de cúpula que propiciam ao público em geral uma imersão visual tendo em vista a melhor compreensão de diversos temas científicos e intelectuais com enfoque em astronomia.

Para ter acesso às sessões, as instituições interessadas realizam o agendamento prévio. Para isso, elas enviam ao Planetário da UFSM as informações definidas por este para que sua sessão seja efetivada. Além disso, é necessário que a sessão seja em um horário que respeite as regras definidas pelo Planetário da UFSM.

Inicialmente, o agendamento era efetuado via telefone, o qual acarretava eventuais divergências de informações como o nome da instituição interessada e o número de alunos que compareceriam à sessão. Buscando solucionar este problema, o agendamento passou a ser solicitado e gerenciado manualmente via e-mail, o que trouxe uma melhora significativa em relação às divergências tidas anteriormente. Porém, este meio não supria todas as necessidades vinculadas ao agendamento, como a possibilidade de encontrar um horário disponível para ambas as partes sem demandar comunicação direta entre o Planetário da UFSM e a instituição interessada.

Em virtude do agravamento da pandemia de Covid-19 nos anos de 2020 e 2021, o isolamento social se fez necessário (Decreto nº 55.240/2020). Diante deste cenário, o Planetário da UFSM teve que adaptar suas sessões de cúpula, as quais eram realizadas de maneira presencial, em sessões virtuais que se sucedem de maneira *online*. A adoção de sessões virtuais determinou um considerável aumento na procura por este tipo de sessão, visto que desta maneira as instituições interessadas não precisariam se deslocar ao Planetário da UFSM. O aumento da demanda

determinou que a equipe do Planetário tivesse que realizar uma série de mudanças como, a busca por uma plataforma digital que permitisse o agendamento de sessões sem a necessidade de uma comunicação extensiva entre o Planetário da UFSM e a instituição interessada. Esta plataforma (SimplyBook<sup>1</sup>) foi adotada de forma provisória, o que possibilitou aos monitores do Planetário identificar que a mesma não supria todas as necessidades relacionadas ao agendamento de sessões, como por exemplo, o envio de e-mails de confirmação no formato desejado, geração de relatório, entre outras necessidades, além de ser paga mensalmente. Com a ausência de um *software* que suprisse essas necessidades, o desenvolvimento de um sistema específico se tornou indispensável.

O objetivo geral deste trabalho é desenvolver o Sistema de Agendamento de Sessões para o Planetário da Universidade Federal de Santa Maria (SASP-UFSM). Este, deve atender requisitos funcionais, visando solucionar, com urgência, a necessidade de um *software* que auxilie o Planetário da UFSM no que diz respeito ao agendamento de sessões.

Os objetivos específicos deste trabalho são: (i) desenvolver o sistema de maneira iterativa e incremental a fim de manter entregas frequentes aos *stakeholders*; (ii) manter operacional o sistema, desde a primeira versão, sem despesas ao desenvolvedor ou ao Planetário da UFSM; (iii) avaliar a usabilidade do sistema desenvolvido com os *stakeholders*; e (iv) avaliar a satisfação dos *stakeholders* em relação ao processo de desenvolvimento e à expectativa dos mesmos em relação ao sistema desenvolvido.

Ao tomar conhecimento, por meio de relações interpessoais, das necessidades do Planetário da UFSM no que diz respeito ao processo de agendamento de sessões, notou-se a inexistência de um sistema que atenda tais necessidades. Portanto, o desenvolvimento do sistema proposto neste trabalho pode agregar valor nos serviços prestados pelo Planetário da UFSM.

Além disso, o desenvolvimento deste sistema, como um trabalho de conclusão de curso, permitirá ao autor compreender a teoria relacionada a um processo de desenvolvimento, às práticas e artefatos de diferentes metodologias ágeis, bem como a aplicação prática destes conceitos, em especial, a interação com *stakeholders* e ciclos de desenvolvimentos curtos, com entregas frequentes e avaliação por parte dos usuários finais (equipe do Planetário). Tais práticas e artefatos de diferentes metodologias ágeis têm sido utilizados pela indústria como um alternativa ao processo tradicional de desenvolvimento, visto que práticas ágeis auxiliam a trazer produtos e serviços para o mercado rapidamente, além de promover uma maior adaptabilidade no

---

<sup>1</sup> <https://simplybook.me/>

que se refere às mudanças constantes impostas por um cenário competitivo. Por conseguinte, o aprendizado teórico e prático deste processo, práticas e artefatos de métodos ágeis é de suma importância para o perfil profissional de um desenvolvedor de *software*.

Este texto está organizado em 6 capítulos, incluindo esta Introdução. O Capítulo 2 apresenta as definições e conceitos relacionados ao contexto do trabalho, assim como as ferramentas utilizadas no desenvolvimento do sistema. No Capítulo 3, expõe-se a metodologia utilizada para elaboração deste trabalho. Já o Capítulo 4 exhibe os incrementos de *software*, sendo que cada incremento possui seu ciclo de vida e funções desenvolvidas. O Capítulo 5 apresenta os resultados e avaliação do sistema desenvolvido. Por fim, o Capítulo 6 conclui o trabalho destacando as contribuições do sistema desenvolvido, assim como suas dificuldades e recomendações para a evolução posterior do *software*.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são introduzidos conceitos teóricos que são de suma importância ao trabalho desenvolvido, a fim de lograr os objetivos do mesmo. A primeira seção (2.1) aborda as definições a respeito da Engenharia de *Software*. Na segunda seção (2.2), disserta-se sobre os conceitos de Engenharia de *Software* Ágil. O desenvolvimento incremental de *Software* é o foco da terceira seção (2.3). A quarta seção (2.4) aborda as ferramentas utilizadas no desenvolvimento do sistema proposto. Finalmente, a quinta seção (2.5) discorre sobre as alternativas ao sistema proposto, bem como no que se difere este com as alternativas citadas.

### 2.1 ENGENHARIA DE SOFTWARE

A Engenharia de *Software* é uma área de engenharia a qual o interesse está em todos os conceitos da produção de *software*, desde as fases iniciais da especificação do sistema até sua manutenção, quando o sistema já está em uso. A engenharia de *software* é importante pelos motivos: (i) Há uma necessidade crescente de sistemas de *software* avançados. Logo, os engenheiros de *software* têm de ser aptos à produzir sistemas confiáveis de maneira econômica e veloz; (ii) A longo prazo, é mais barato usar métodos e técnicas de engenharia de *software* nos sistemas de *software*, ao contrário da implementação de maneira simplória (SOMMERVILLE, 2010).

Processo de *software* é a abordagem sistemática usada na engenharia de *software* e é uma sequência de atividades as quais resultam em um produto de *software*. Os processos de *software* compartilham quatro atividades essenciais, quais sejam: (i) especificação, a qual engenheiros de *software* e *stakeholders* (partes interessadas) definem o *software* a ser elaborado e suas restrições de operação; (ii) desenvolvimento, na qual o *software* é projetado e programado; (iii) validação, na qual o *software* desenvolvido é autenticado a fim de certificar que o mesmo está de acordo com o que os *stakeholders* querem; (iv) evolução, a qual o *software* é alterado a fim de retratar a mudança de requisitos dos *stakeholders* (SOMMERVILLE, 2010).

### 2.2 ENGENHARIA DE SOFTWARE ÁGIL

A Engenharia de *Software* Ágil é uma possível alternativa para a engenharia de *software* convencional voltada para determinadas classes de *software* e tipos de projetos. Essencial-

mente, métodos ágeis se esforçam para resolver fraquezas reais e perceptíveis na Engenharia de *Software* convencional e oferecem benefícios importantes. Entretanto, métodos ágeis não são indicados para todos os casos.

O Manifesto Ágil, escrito por Fowler, Highmish et al. (2001), é uma declaração que fundamenta o desenvolvimento ágil de *software*. Nele são descritos quatro valores e doze princípios de agilidade para que uma metodologia seja considerada ágil. Os quatro valores são: (i) indivíduos e interações são mais importantes que processos e ferramentas; (ii) *software* em funcionamento é mais importante que documentação extensiva; (iii) colaboração do cliente é mais importante que negociação contratual; e (iv) resposta a mudanças é mais importante que seguir um plano. Nele são descritos 12 princípios de agilidade, são eles: (i) prioridade na satisfação do cliente por meio da entrega contínua de um *software* com valor; (ii) mudanças são favoráveis à vantagem competitiva na relação com o cliente; (iii) entrega de *software* com frequência; (iv) trabalho em conjunto entre desenvolvedores e pessoal comercial; (v) construção de projetos com indivíduos motivados por meio de ambiente, confiança e apoio necessário; (vi) comunicação aberta e presencial entre membros da equipe; (vii) *software* em funcionamento como principal medida de progresso; (viii) desenvolvimento sustentável por meio de um ritmo constante indefinidamente entre proponentes, desenvolvedores e usuários; (ix) importância da excelência técnica dos projetos como forma de aumento da agilidade; (x) simplicidade é essencial, ou seja, deve-se minimizar a quantidade de trabalho que precisa ser feito; (xi) qualidade de requisitos, arquiteturas e projetos é maior em equipes que tem maior autonomia e se auto-organizam; e (xii) encontros regulares entre a equipe para manutenção do comportamento com o objetivo de se tornar mais eficaz (FOWLER; HIGHSMITH et al., 2001).

Embora a ideia de envolvimento dos clientes no processo de desenvolvimento seja interessante, seu sucesso depende de clientes motivados e capazes de passar um bom tempo com a equipe de desenvolvimento, além de que os clientes possam representar todos os *stakeholders* do sistema. Além disso, certos membros da equipe podem não ter personalidade adequada para o intenso envolvimento típico de métodos ágeis. Outra questão é que priorizar mudanças pode ser difícil, ainda mais em sistemas com vários *stakeholders*, visto que cada *stakeholder* dá prioridades diferentes à mudanças diferentes (SOMMERVILLE, 2010).

As abordagens de engenharia de *software* ágil mais utilizada é o XP (PRESSMAN, 2009). Mas também há o Scrum, o Kanban, o Lean, entre outros.



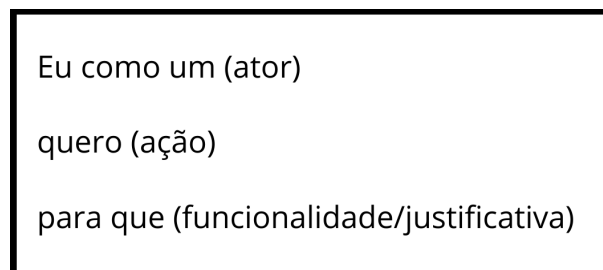
### 2.2.1 *Extreme Programming (XP)*

O trabalho seminal sobre o tema, “Extreme Programming Explained: Embrace Change”, foi escrito por Kent Beck em 2004. Beck definiu quatro valores que servem de base para a XP, são eles: (i) comunicação efetiva e estreita, embora informal; (ii) simplicidade, projetando somente as necessidades imediatas; (iii) *feedback* do cliente, de outros membros da equipe e do próprio *software*; e (iv) coragem, necessária a adoção estrita de certas práticas da XP (BECK; ANDRES, 2004). Estes valores são usados como direcionador das atividades da XP, e quando seguidos em conjunto, a equipe ágil impõe respeito entre seus membros e outros envolvidos.

O paradigma de desenvolvimento preferido do processo XP é a abordagem orientada a objetos e abrange uma coleção de regras e práticas constantes, são elas: (i) o *planning game*; (ii) pequenas versões; (iii) uso de metáforas; (iv) simplicidade de projeto; (v) testes; (vi) refatoração; (vii) programação em pares; (viii) propriedade coletiva do código; (ix) integração contínua; (x) quarenta horas semanais; (xi) cliente sempre presente; e (xii) padrões de codificação (BECK; ANDRES, 2004).

Em XP, os requisitos são definidos como histórias de usuário, os quais são implementadas a partir de uma série de tarefas (SOMMERVILLE, 2010). As histórias de usuário descrevem funcionalidades as quais serão valiosas aos *stakeholders* do sistema e essas devem ser curtas, simples e claras. Nelas devem ser listadas: (i) o ator, que é o interessado na funcionalidade; (ii) a ação, que é o que é o ator quer fazer; e (iii) a funcionalidade, que é o que o ator espera que aconteça ao realizar a ação e pode ser entendida como a justificativa para tal ação (COHN, 2004). O *template* para escrever uma história de usuário é mostrada na Figura 1.

Figura 1 – Estrutura de uma História de Usuário



Fonte: Produção do autor.

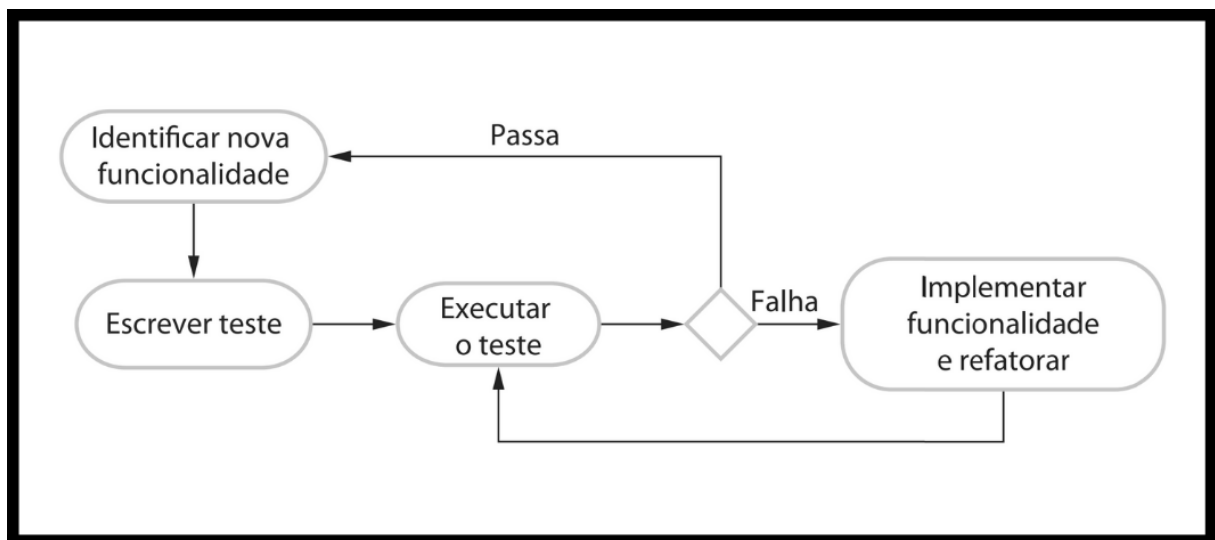
Os motivos para utilizar histórias de usuário como meio de expressar requisitos são: (i) enfatizam a comunicação verbal em vez de escrita; (ii) são compreensíveis para os *stakehol-*

ders e desenvolvedores; (iii) são do tamanho certo para o planejamento; (iv) trabalham para o desenvolvimento iterativo; e (v) incentivam a adiar os detalhes até que os *stakeholders* tenham um melhor entendimento sobre o que eles realmente necessitam.

Os críticos do uso de história de usuário argumentam que é necessário um modelo ou especificação mais formal para assegurar que omissões, inconsistências e erros sejam descobertos antes que o sistema seja produzido. Os defensores do uso rebatem dizendo que o fato de os requisitos possuírem uma natureza de alteração torna tais modelos e especificações obsoletos praticamente logo depois de terem sido desenvolvidos (PRESSMAN, 2009).

O desenvolvimento dirigido a testes (TDD, do inglês *Test-Driven Development*), uma das práticas do XP, é uma abordagem que modifica o paradigma tradicional de desenvolvimento de *software*. No TDD, intercalam-se testes e desenvolvimento de código (BECK, 2002). Basicamente, desenvolve-se um código de maneira incremental, em conjunto com um teste para esse incremento, e não prossegue para um próximo incremento enquanto o código desenvolvido não passe no teste (SOMMERVILLE, 2010). A Figura 2 ilustra o ciclo descrito anteriormente.

Figura 2 – Ciclo TDD



Fonte: (SOMMERVILLE, 2010).

O uso do TDD resulta em melhora na qualidade de código, qualidade de testes, qualidade da aplicação se comparado com técnicas tradicionais de testes. Os críticos do uso do TDD argumentam que as mudanças frequentes a testes com o uso do TDD são mais propensos a causar quebras no sistema, do que a abordagem de testar o software no final do ciclo de desenvolvimento, acarretando assim em retrabalho e perda de produtividade (BUCHAN; LI;

MACDONELL, 2011).

Em XP, a validação é determinada por testes de aceitação, denominados testes de cliente, os quais são especificados pelo cliente e mantêm o foco nas características e na funcionalidade do sistema que são visíveis e que podem ser revistas pelo cliente. Os testes de aceitação são obtidos de histórias de usuários implementadas como parte de uma versão do sistema. (PRESSMAN, 2009).

### 2.2.2 Scrum

Scrum é um processo simples para gerenciar projetos complexos os quais é impossível prever o que irá acontecer. Deste modo, os praticantes do Scrum sabem exatamente o que está acontecendo e são capazes de fazer ajustes para manter o projeto em direção ao objetivo (SCHWABER, 2004).

No Scrum há três papéis principais: (i) o *Product Owner*, o qual é responsável por representar os interesses das partes interessadas no produto; (ii) o ScrumMaster, o qual é responsável por ensinar e implementar o processo Scrum da maneira adequada, além de garantir que toda a equipe está seguindo suas regras e práticas; e (iii) o time de desenvolvimento, o qual desenvolverá o produto. Também há a Sprint, uma iteração de tamanho fixo, geralmente de duas a quatro semanas, na qual um incremento de *software* é desenvolvido. Além disso, o Scrum possui os seguintes artefatos: (i) o *Backlog* de Produto, uma lista de requisitos funcionais e não-funcionais que se tornarão funcionalidades do sistema; e (ii) o *Backlog* da Sprint, uma lista de requisitos advindos do *Backlog* de produto os quais serão implementados no próximo incremento do sistema.

O processo do Scrum começa com uma visão do sistema a ser desenvolvido. A visão pode ser vaga no início mas ficará mais clara conforme o projeto avança. O *Product Owner* obtém esta visão e registra os requisitos no *Backlog* de Produto. Mudanças no *Backlog* de Produto refletem em mudanças nos requisitos de negócio e na velocidade que o time de desenvolvimento pode transformar o *Backlog* de Produto em funcionalidade. Todo o trabalho é realizado em *Sprints*. Cada *Sprint* inicia com uma reunião de planejamento, onde o *Product Owner* e o time de desenvolvimento colaboram para selecionar o que será realizado na próxima *Sprint*. Selecionando as prioridades, o *Product Owner* comunica ao time de desenvolvimento o que é desejado, e o time comunica ao *Product Owner* o quanto daquilo que é desejado pode se tornar funcionalidade através da próxima *Sprint*. Ainda durante o planejamento, o time de

desenvolvimento, o qual é responsável por gerenciar seu próprio trabalho, divide o trabalho em tarefas as quais compõem o *Backlog* de *Sprint*. Ao decorrer da *Sprint*, são realizadas reuniões diárias de 15 minutos a fim de dar visibilidade a cada membro da equipe a respeito das atividades realizadas por outros membros da equipe, além de resolver possíveis problemas que algum membro da equipe esteja enfrentando. Durante a *Sprint* não pode ocorrer mudanças advindas de membros externos ao time. Ao final da *Sprint* uma reunião de revisão da *Sprint* é realizada com finalidade de proporcionar uma discussão em conjunto a respeito do que o time deve fazer a seguir. Nesta reunião o time de desenvolvimento apresenta o que foi desenvolvido durante a *Sprint* ao *Product Owner* e a qualquer *stakeholder* que quiser participar. Após a revisão de *Sprint* e antes da próxima reunião de planejamento, o ScrumMaster realiza uma retrospectiva de *Sprint* com o time a fim de encorajar o time a revisar seu processo de desenvolvimento com o objetivo de torná-lo mais eficaz e agradável para a próxima *Sprint* (SCHWABER, 2004).

### 2.2.3 Kanban

David Anderson e Andy Carmichael definiram o Kanban, em 2015, como sendo um método para definir, gerenciar, e aperfeiçoar serviços que entregam trabalho, como projetos de *software*. O Kanban pode ser caracterizado como um método “comece com o que você faz agora”, servindo como catalisador para mudanças rápidas e focadas dentro da organização (ANDERSON; CARMICHAEL, 2015).

Os valores do Kanban são: (i) transparência; (ii) equilíbrio; (iii) colaboração; (iv) foco no cliente; (v) fluxo; (vi) liderança; (vii) entendimento; (viii) acordo; e (ix) respeito (ANDERSON; CARMICHAEL, 2015).

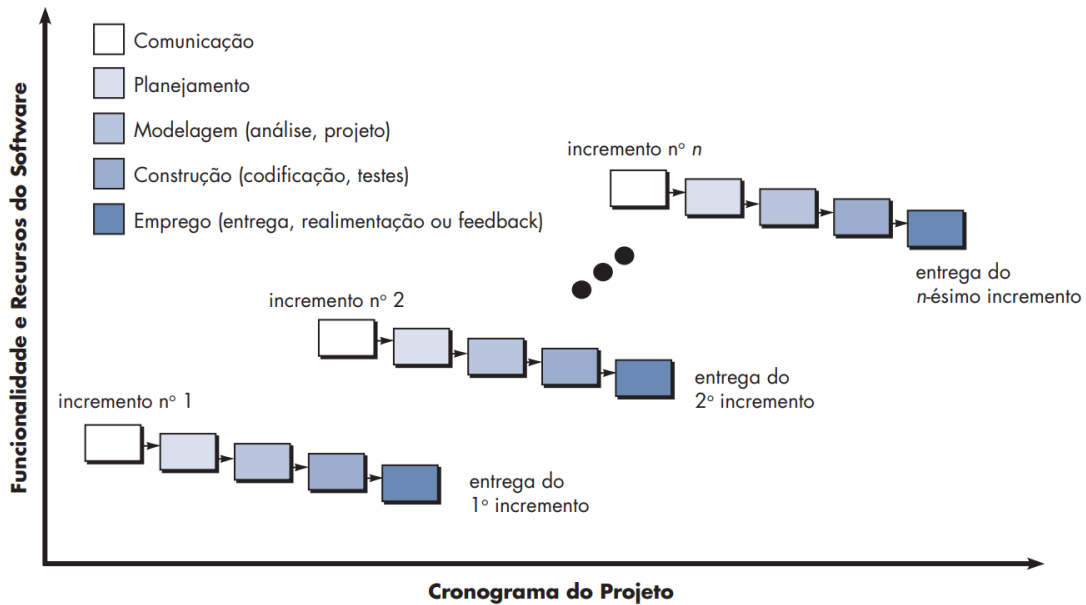
Kanban faz uso de cartões e um quadro para controlar o trabalho em progresso em um processo de produção. O objetivo é visualizar o fluxo de maneira visual, mostrando todos os passos do processo, tornando a identificação de falhas mais fácil (ANDERSON; CARMICHAEL, 2015).

## 2.3 DESENVOLVIMENTO INCREMENTAL

Desenvolvimento incremental é um dos modelos de ciclo de vida de um *software*. Sua ideia é desenvolver e entregar várias versões, que são partes de um sistema, e que se incrementam até que o sistema adequado esteja desenvolvido por completo. Desta maneira, as atividades

de especificação, desenvolvimento e validação ocorrem de maneira intercalada (SOMMERVILLE, 2010), assim como mostrado na Figura 3.

Figura 3 – O Modelo Incremental



Fonte: (PRESSMAN, 2009).

Frequentemente, quando se utiliza essa abordagem, o primeiro incremento é um produto essencial, no qual os requisitos básicos são atendidos e requisitos complementares ainda não são entregues. Esse produto essencial é utilizado e avaliado pelos clientes. Para conseguir uma adaptação incremental, a equipe ágil precisa de *feedback* dos mesmos. A partir do resultado dessa avaliação e do *feedback* recebido, um novo incremento é planejado (PRESSMAN, 2009). O desenvolvimento incremental é uma parte fundamental das abordagens ágeis e é uma das melhores para a maioria dos sistemas de negócios, *e-commerce* e sistemas pessoais.

## 2.4 FERRAMENTAS

Nesta seção são descritas as definições das principais ferramentas que contribuíram para o desenvolvimento, descritas na Subseção 2.4.1, e hospedagem do sistema, descritas na Subseção 2.4.2 proposto neste trabalho.

### 2.4.1 Desenvolvimento

Para o desenvolvimento de um sistema, além das linguagens de programação, há também bibliotecas e *frameworks* que auxiliam o desenvolvimento, não apenas oferecendo funcionalidades, mas também uma arquitetura para o trabalho de desenvolvimento.

Dito isso, o React JS é uma biblioteca de código aberto da linguagem de programação JavaScript (JS). Ela é usada para construir interfaces gráficas de usuário (GUI, do inglês *Graphical User Interface*). Atualmente, é a biblioteca mais popular do JavaScript e é usada por grandes companhias como Netflix, Airbnb, Facebook, entre outras (LONGEN, 2021). Seu funcionamento se dá por meio de componentes, que são diferentes partes de uma página conectados entre si, de modo a serem utilizados para reaproveitamento de código e padronização de interface. O React JS utiliza a extensão JSX para descrever seus componentes. Um arquivo JSX é uma extensão desenvolvida como uma mistura de HTML e JS e seu objetivo é ser transposta para JS da maneira mais simples possível (ROVEDA, 2019).

Para o desenvolvimento de uma API, o ASP.NET Core Web API é um *framework* para construção de serviços HTTP que podem ser acessados por qualquer cliente incluindo navegadores e aplicativos de celular. O ASP.NET Core Web API suporta diferentes tipos de resposta, incluindo JSON, XML, entre outros (TUTORIALSTEACHER, 2022).

O PostgreSQL é um sistema gerenciamento de banco de dados relacional *open source* o qual ganhou uma forte reputação de confiabilidade, robustez e performance. Sua origem foi em 1986 como projeto da Universidade da Califórnia, passou a ser compatível com propriedades ACID em 2001 e é compatível com a maioria dos sistemas operacionais. Ele vem com muitos recursos a fim de ajudar desenvolvedores a construir aplicações, administrar a integridade dos dados e construir ambientes tolerantes à falha. Além de ser grátis e *open source*, ele é extremamente extensível, por permitir ao desenvolvedor criar seus próprios tipos de dados, construir funções customizadas, entre outros (POSTGRESQL, 1996-2022).

### 2.4.2 Hospedagem

No que diz respeito à hospedagem de um sistema, ou seja, o local onde os dados e o sistema estão localizado, alguns serviços de hospedagem foram selecionados para a realização deste trabalho.

A Netlify é um serviço de que permite a hospedagem de sites de maneira gratuita. Além

disso, a Netlify proporciona: (i) SSL grátis; (ii) tempo de resposta baixo; e (iii) integração contínua com o Github, GitLab e BitBucket (SOUZA, 2021).

A Azure é uma plataforma da Microsoft que permite que empresas e desenvolvedores adquiram as capacidades de processamento e armazenamento dos *datacenters* da Microsoft para diversos serviços diferentes. São serviços ofertados pela Azure: (i) banco de dados SQL; (ii) máquinas virtuais na nuvem; (iii) análise de dados e com IA e *machine learning*; (iv) gerenciamento de rede; (v) serviços na nuvem; (vi) armazenamento e *backup*; (vii) streaming; e (viii) contêineres na nuvem (GARRETT, 2020).

A Amazon Web Services (AWS) é a plataforma de nuvem mais adotada no mundo. Ela oferece mais de 200 serviços completos de *datacenters* em todo o mundo. São serviços ofertados pela AWS: (i) computação; (ii) armazenamento e banco de dados; (iii) *machine learning*; (iv) inteligência artificial; (v) *data lakes*; (vi) análises; e (vii) internet das coisas (SYOZI, 2021).

## 2.5 ALTERNATIVAS

Atualmente, existem inúmeras alternativas, pagas e gratuitas, ao sistema proposto neste trabalho. A plataforma SimplyBook<sup>2</sup>, a qual o Planetário da UFSM adotou provisoriamente após uma breve pesquisa por parte de sua equipe, possui quatro planos incluindo o plano gratuito. O plano adotado pelo Planetário da UFSM foi o plano básico, o qual custa cerca de 40 reais por mês e seus principais recursos são: (i) 100 agendamentos por mês; (ii) limite de 15 usuários; (iii) página de reserva responsiva; e (iv) e-mails automáticos de confirmação de reserva. Apesar dos interessantes recursos, esta plataforma não têm: (i) boa personalização visual da página de reserva; e (ii) criação automática de encontro virtual no Google Meet. A plataforma encontrada com melhor plano gratuito foi o Calendly<sup>3</sup>, o qual contém: (i) número máximo de um tipo de evento; (ii) conexão com calendários; (iii) notificações automatizadas de eventos; e (iv) aplicativo para celular. Porém, em seu plano gratuito, o Calendly não possui: (i) notificações por e-mail customizadas; (ii) personalização da página de reservas; e (iii) visualização de métricas.

Outra alternativa muito popular é a plataforma Reservio<sup>4</sup>. Este por sua vez, não possui plano gratuito. Sua opção de melhor custo benefício é o plano Standart, o qual custa 49 reais por mês e seus principais recursos são: (i) 500 agendamentos por mês; (ii) domínio personalizado;

---

<sup>2</sup> <https://simplybook.me/pt/>

<sup>3</sup> <https://calendly.com/pt/>

<sup>4</sup> <https://app.reservio.com/>

(iii) visualização de estatísticas; e (iv) possibilidade de importar e exportar dados. Em contraste com estes recursos disponíveis, o Reservio, no caso do plano Standart possui propagandas na página de reserva, bem como nas notificações de confirmação de agendamento. Outro *software* encontrado publicamente, foi o SuperSaaS<sup>5</sup>. Este, em seu plano gratuito oferece até 50 agendamento mensais, porém com anúncios e sem sincronização com o Google Calendário<sup>6</sup>. O plano pago de menor custo oferece até 100 agendamentos, sem anúncios e com sincronização a partir de oito dólares mensais. Observa-se então que os sistemas disponíveis no mercado atualmente oferecem diferentes planos mensais, gratuitos ou pagos, os quais fornecem mais recursos conforme o preço do plano mensal.

O sistema proposto neste trabalho se difere dos sistemas citados nesta seção pois permite a operação de todos os recursos sem a necessidade de pagar por maior disponibilidade dos mesmos. No que diz respeito à personalização visual, o sistema proposto tem como base o manual de marca do Planetário, gerando um sistema adequado à tipografia, grafismo e paleta de cores definidas pelo Planetário da UFSM. Além disso, o sistema atende as necessidades específicas do Planetário da UFSM, tornando-o melhor no quesito usabilidade, visto que não há funcionalidades desnecessárias que podem prejudicar a operação do mesmo.

---

<sup>5</sup> <https://www.supersaas.com.br/>

<sup>6</sup> <https://calendar.google.com/>



### 3 DESENVOLVIMENTO DO SASP-UFSM

Neste capítulo são abordados os métodos de desenvolvimento (Seção 3.1), hospedagem (Seção 3.2) e avaliação do sistema (Seção 3.3) proposto neste trabalho. Tais métodos de desenvolvimento são advindos a partir de diferentes metodologias ágeis a fim de propiciar a entrega de um *software* apto a operar de maneira satisfatória desde o primeiro uso. Já os métodos de hospedagem são advindos da escolha de plataformas que possibilitam a hospedagem gratuita, necessária para o cumprimento do objetivo de manter o sistema em operação sem gerar custos aos envolvidos. Por fim, o métodos de avaliação advém da gratuidade de uma plataforma a qual permite uma pesquisa com a equipe do Planetário para que a mesma avalie a usabilidade e a satisfação com o processo de desenvolvimento do sistema.

#### 3.1 PROCESSO DE DESENVOLVIMENTO

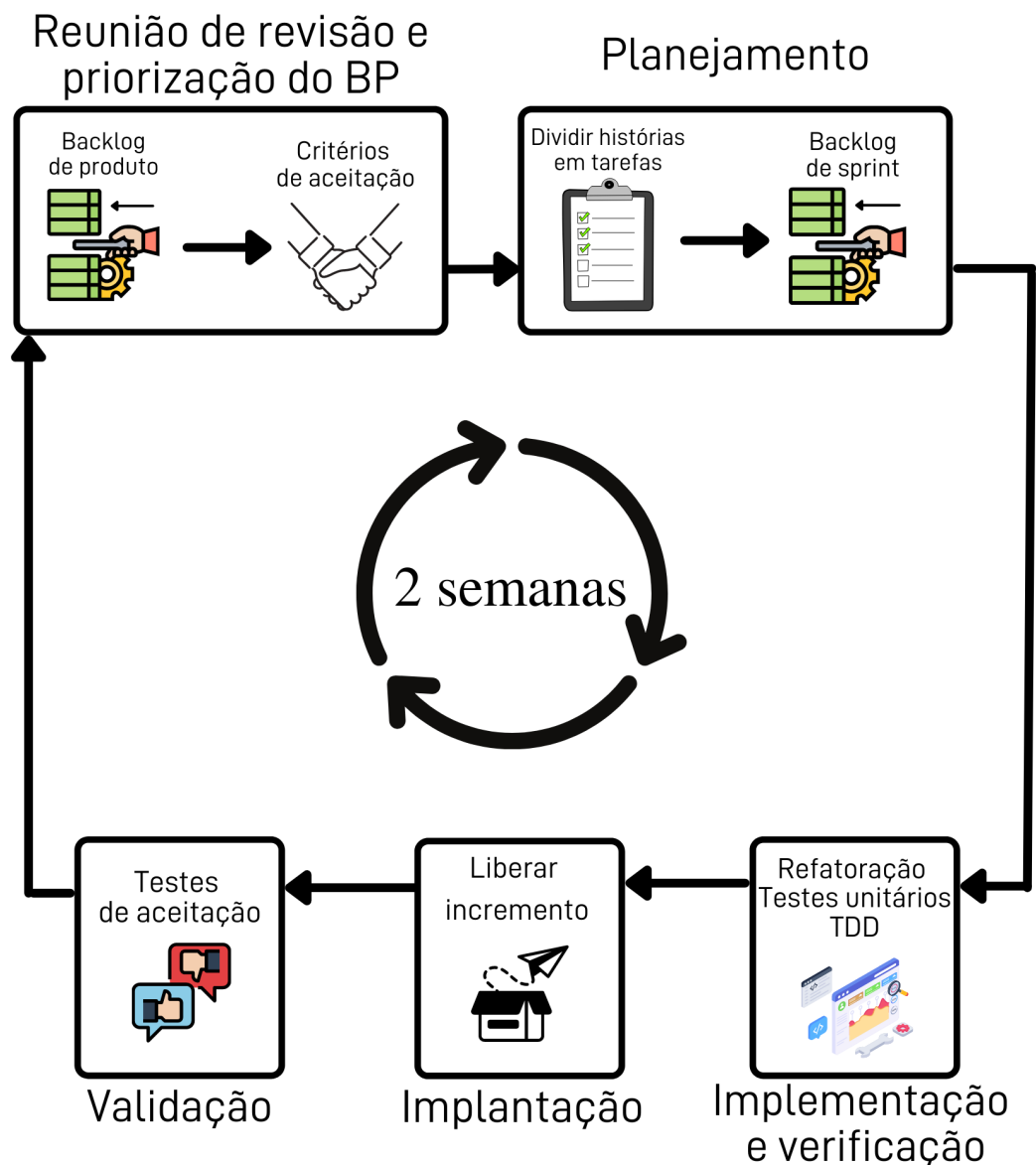
Neste trabalho, o desenvolvimento iterativo e incremental é utilizado em conjunto com certas práticas e artefatos advindos de diferentes metodologias ágeis. O uso desta abordagem de desenvolvimento, das práticas e dos artefatos se deve: (i) à necessidade de uma comunicação clara e frequente entre o desenvolvedor e a equipe (monitores e diretor) do Planetário, para que o *software* seja desenvolvido com base nas prioridades definidas pela equipe em diferentes momentos e para que se obtenha retorno constante sobre as funcionalidades já desenvolvidas; (ii) à necessidade de um auxílio na especificação de requisitos ao longo de todo o processo de desenvolvimento, visto que, no momento inicial do processo, a equipe do Planetário não têm uma visão clara de todos os requisitos que o sistema deve possuir; (iii) à finalidade de melhorar a qualidade do código produzido pelo desenvolvedor, para que o sistema seja eficiente, confiável e testável; e (iv) à finalidade de dar resiliência ao sistema produzido, tornando-o adaptável às possíveis mudanças e alterações, mantendo a integridade do mesmo quando estas forem executadas.

A frequência na comunicação com a equipe do Planetário deve gerar maior aproximação do produto final de acordo com os requisitos, aumentando a qualidade do sistema em si. A possível falta de envolvimento com o retorno ao desenvolvedor pode gerar dúvida se o desenvolvimento está gerando o produto adequado. Portanto, também é necessário o comprometimento por parte da equipe do Planetário em estar presente nas reuniões de iteração. Dado isso, novos

incrementos são entregues ao final de cada iteração, a qual tem um comprimento fixo de duas semanas. Estas iterações de curta duração acabam com o risco de não resolver problemas rapidamente, encontrando e solucionando defeitos mais rapidamente se comparado à um processo de desenvolvimento tradicional, visto que a comunicação e *feedback* da equipe do Planetário da UFSM se tornam frequentes.

O ciclo completo da iteração pode ser visualizado na Figura 4 e é descrito nesta seção.

Figura 4 – Ciclo completo da iteração



Fonte: Produção do autor.

### 3.1.1 Reunião de Revisão e Priorização do *Backlog* de Produto

Na fase de especificação da iteração, ocorre a Reunião de Revisão e Priorização do *Backlog* de Produto. Esta reunião é a união da reunião de revisão de *Sprint* com a reunião de planejamento de *Sprint*, duas práticas do Scrum. A escolha por realizar as duas reuniões no mesmo momento foi baseada no fato de a equipe do Planetário da UFSM não apresentar tempo disponível para a realizações de duas reuniões por iteração.

Para que houvesse maior qualidade da discussão, foram utilizadas perguntas em conjunto com a discussão informal, tornando a Reunião de Revisão e Priorização do *Backlog* de Produto em uma espécie de entrevista informal. O conjunto de perguntas é:

- O que já está suficiente na versão entregue até o momento?
- O que precisa melhorar da atual versão para a próxima?
- Quais os novos recursos que o sistema deve ter?
- Quais os novos recursos precisam ser entregues na próxima versão?

Dependendo do momento da reunião, como no início da primeira iteração, as primeiras duas perguntas não serão utilizadas, tornando a Reunião de Revisão e Priorização do *Backlog* de Produto uma reunião de planejamento de *Sprint*.

Por meio desta reunião, obtêm-se um *feedback* das funcionalidades já entregues (exceto na primeira reunião), além da identificação de novas funcionalidades ou correções para o sistema, gerando assim novos requisitos os quais são descritos em histórias de usuário, uma prática advinda do XP. Estas histórias de usuários são adicionadas ao *Backlog* de Produto, artefato do Scrum.

Após essa identificação, define-se os critérios de aceitação das histórias de usuários, ou seja, condições que devem ser atendidas para que a história de usuário seja aceita ao final da iteração. Estes critérios de aceitação são utilizados como base para os testes de aceitação que são realizados na fase de validação da iteração.

### 3.1.2 Planejamento

Em um momento posterior à Reunião de Revisão e Priorização do *Backlog* de Produto, divide-se as histórias de usuários em tarefas, uma prática utilizada no planejamento de *Sprint* do

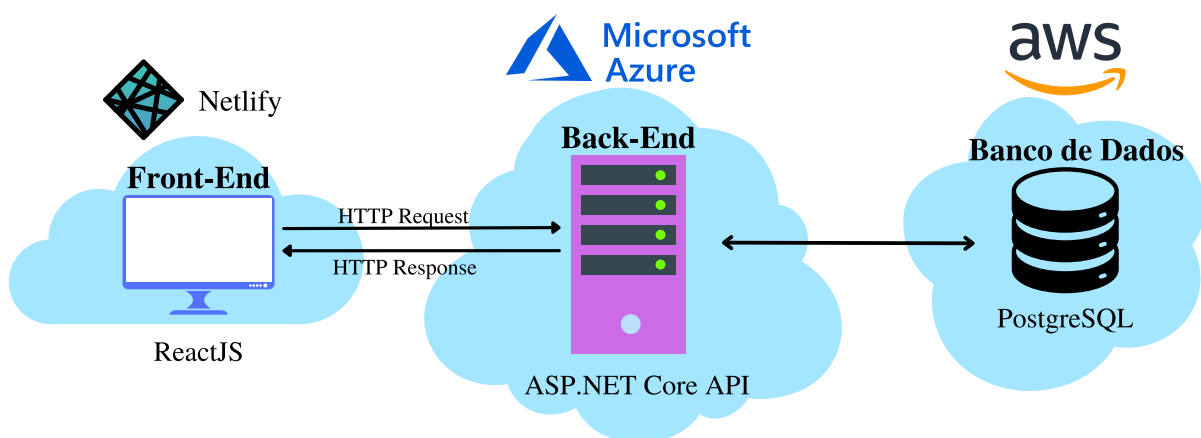
Scrum, facilitando assim a estimativa por parte do autor do presente trabalho de quanto tempo cada história de usuário irá demorar para ser implementada.

Com base nas prioridades dos *stakeholders* e nessas estimativas, algumas histórias de usuário e tarefas contidas no *backlog* de produto são selecionadas para a próxima iteração, movendo-as para o *Backlog* de *Sprint*.

### 3.1.3 Implementação e Verificação

O sistema é implementado utilizando dois códigos-fonte separados, o *front end* e o *back end*, que se integram por meio de requisições HTTP e são versionados por meio de um repositório Git. A biblioteca React JS é utilizada no *front end*, enquanto uma API ASP.NET é empregada no *back end*. Esta API consultará em um banco de dados relacional PostgreSQL quando necessário. Estas tecnologias foram escolhidas de acordo com a preferência e experiência do autor. A arquitetura geral do sistema pode ser visualizada na Figura 5.

Figura 5 – Arquitetura do Sistema



Fonte: Produção do autor.

A fim de manter o sistema eficiente e sem duplicação de código, a prática de refatoração foi empregada. A técnica de refatoração altera o código-fonte do sistema sem alterar seu comportamento a fim de melhorar a qualidade do mesmo, implicando na melhora da manutibilidade do código, reduzindo a possibilidade de erros no desenvolvimento e proporcionando maior eficiência, confiabilidade e testabilidade.

Um manual de marca é um guia de boas práticas para a utilização do logotipo e/ou comunicação construída por uma organização (MOURA, 2020). O manual de marca do Planetário

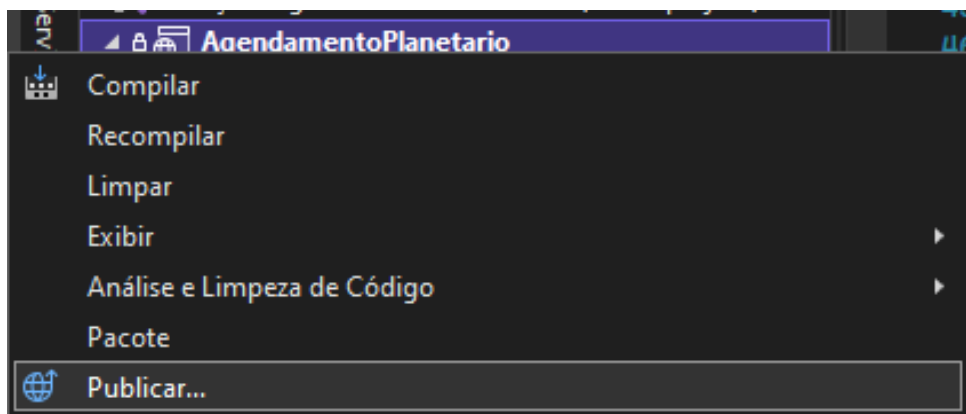
da UFSM contém boas práticas de uso do logotipo, da tipografia, do grafismo e da paleta de cores. Este documento está em posse do Planetário da UFSM e foi fornecido ao autor deste trabalho para auxílio na elaboração dos componentes gráficos desenvolvidos no *front end*.

A verificação do sistema desenvolvido é feita por meio de testes unitários automatizados aplicados ao *back-end* do sistema, a fim de manter sua integridade, ou seja, que o mesmo não contenha falhas na codificação nem deixe de funcionar adequadamente quando uma nova funcionalidade for implementada. Estes testes unitários são implementados em um ciclo de TDD, uma das técnicas do XP, proporcionando maior confiabilidade do código desenvolvido e mantendo-o mais limpo possível.

### 3.1.4 Implantação

Para a implantação de uma nova versão, são realizados dois procedimentos, um para cada código fonte do sistema. Para o *back end*, a integração entre o Visual Studio, que é a interface de desenvolvimento, e a Azure foi utilizada. Essa integração se dá por um simples botão “Publicar” que após a primeira configuração já pode ser usado sem qualquer processo adicional. Este botão pode ser visto na Figura 6. Com isto, com apenas um clique, a nova versão do *back end* já estava disponível.

Figura 6 – Botão Publicar do Visual Studio



Fonte: Produção do autor.

Para o *front end*, a cada *commit* na *branch* principal, o Netlify executa o *build* automaticamente e já implanta e disponibiliza a nova versão em questão de segundos. Após a implantação de todas as partes do sistema, o mesmo estará disponível para utilização pelos usuários finais.

### 3.1.5 Validação

A validação do sistema é realizada pela equipe do Planetário, após a implantação, realizando testes de aceitação com base nos critérios de aceitação de cada história de usuário. Essa validação é realizada por meio de testes funcionais a fim de verificar se as funcionalidades do incremento produzido estão de acordo com os critérios de aceitação que foram criados no início da iteração. Uma história de usuário aprovada implica que as funcionalidades foram desenvolvidas da maneira que a equipe do Planetário desejava.

Caso uma história de usuário não estiver implementada totalmente de acordo com os critérios de aceitação, o desenvolvedor e a equipe do Planetário discutirão se a mesma pode continuar em uso pelos usuários finais. Se a resposta for afirmativa, a funcionalidade é mantida. Se a resposta for negativa, a funcionalidade é retirada do incremento e uma nova versão, sem esta funcionalidade, é implantada na mesma iteração. Se a equipe do Planetário desejar, a mesma pode escolher manter a história de usuário na próxima iteração.

## 3.2 HOSPEDAGEM DO SISTEMA

Todas as soluções utilizadas para a hospedagem do sistema são de uso gratuito a fim de cumprir com o objetivo de manter o sistema operacional de maneira gratuita. Para o caso do *front-end*, a Netlify permite mensalmente até 100 gigabytes de tráfego de rede e 300 minutos de *build* com uma disponibilidade de serviço em 100% do tempo. Para o caso do *back-end*, a Azure disponibiliza um servidor 32 bits que automaticamente se desliga após 20 minutos sem requisições. Quando uma nova requisição ao *back-end* é feita, o servidor “acorda” e responde a requisição. O principal empecilho é que este “acordar” demora cerca de vinte e cinco segundos para finalizar, acarretando em uma espera muito grande na primeira requisição ao servidor quando um usuário necessita consultar ou enviar dados. Para contornar isto, a utilização de um *bot* da plataforma Uptime<sup>7</sup>, o qual realiza uma requisição ao servidor a cada cinco minutos, foi a solução encontrada. Com isso, todas as requisições são efetuadas e respondidas rapidamente. Este bot também propicia a porcentagem do tempo de disponibilidade do sistema, ou seja, a taxa de requisições respondidas com sucesso durante certo período de tempo. Portanto, no Capítulo 5 é descrita uma análise do tempo de disponibilidade do sistema utilizando as ferramentas e métodos descritos. Para o caso do banco de dados, a AWS disponibiliza uma instância de 20

---

<sup>7</sup> <https://uptimerobot.com/>

gigabytes da versão 12 do PostgreSQL em seu plano gratuito, além de fornecer os parâmetros de conexão com este banco.

### 3.3 AVALIAÇÃO DO SISTEMA

A avaliação do sistema com a equipe do Planetário da UFSM (totalizando cinco monitores) é realizada por meio da plataforma Google Forms<sup>8</sup>. A primeira parte da avaliação tem por finalidade avaliar a usabilidade do produto produzido, uma das características de qualidade de produto presente no modelo de qualidade de *software* da ISO 25010<sup>9</sup>. Esta ISO serviu de base para a construção de afirmações as quais serão respondidas de forma quantitativa utilizando a escala de Likert, que permite indicar o grau de concordância ou discordância em relação à afirmação. As perguntas desta parte são: (i) “o sistema é apropriado para suas necessidades”, refere-se à sub-característica “apropriação reconhecível”; (ii) “é fácil aprender a usar o sistema”, refere-se à sub-característica “apreensibilidade”; (iii) “é fácil de operar e controlar o sistema”, refere-se à sub-característica “operabilidade”; (iv) “o sistema te protege de errar”, refere-se à sub-característica “proteção de erros do usuário”; (v) “a interface é agradável”, refere-se à sub-característica “estética da interface do usuário”; e (vi) “o sistema permite a utilização do software por pessoas com características e capacidades diferenciadas”, refere-se à sub-característica “acessibilidade”.

A outra parte da avaliação tem por objetivo avaliar a satisfação da equipe do Planetário em relação ao uso do processo de desenvolvimento iterativo e incremental na visão de *stakeholder*, ou seja, ao uso de histórias de usuário e seus critérios de aceitação, bem como o nível de envolvimento com a Reunião de Revisão e Priorização do *Backlog* de Produto, além da satisfação em relação ao tempo de entrega. Também é avaliado, nesta parte da avaliação, o quanto a expectativa da equipe do Planetário foi cumprida em relação ao sistema desenvolvido. Nesta parte, todas as afirmações também são respondidas utilizando a escala de Likert. As afirmações desta parte são: (i) “o uso de histórias de usuário e seus critérios de aceitação facilitou a especificação e testagem dos requisitos”, a qual tem fim de avaliar o quanto o uso de histórias de usuário conseguiu expressar o que a equipe do Planetário queria e auxiliou no processo de validação; (ii) “você se envolveu e se comprometeu com a Reunião de Revisão e Priorização do *Backlog* de Produto”, a qual tem fim de avaliar o quanto a equipe do Planetário se envolveu e

<sup>8</sup> <https://docs.google.com/forms>

<sup>9</sup> <https://iso25000.com/index.php/en/iso-25000-standards/iso-25010>

comprometeu nesta atividade que gera *feedback* sobre o incremento produzido e identificação de novas funcionalidades; (iii) “você ficou satisfeito com tempo de entrega de duas semana”, a qual tem finalidade de avaliar o quanto a equipe do Planetário se sentiu satisfeita com o tempo de entrega de incremento de duas semanas; (iv) “sua expectativa foi cumprida em relação ao sistema desenvolvido”, a qual tem finalidade de avaliar se o sistema produzido atendeu às necessidades e expectativas da equipe do Planetário.



## 4 PROJETO E IMPLEMENTAÇÃO

Este capítulo aborda a parte prática do presente trabalho realizada de acordo com a metodologia proposta no Capítulo 3. Cada seção aborda uma iteração e descreve abstratamente como as funcionalidades foram implementadas. Esta parte prática teve início na metade de novembro de 2021 e conclusão no final de janeiro de 2022, totalizando dois meses e duas semanas. Como cada iteração tem a duração de duas semanas, todo o desenvolvimento prático do sistema será realizado em cinco iterações. Cada tabela apresentada neste capítulo correspondente a uma iteração e possui as histórias de usuário implementadas na mesma. Cada história de usuário é numerada como forma de identificação da mesma. Uma tabela contendo todas as histórias de usuário e seus critérios de aprovação está presente no Apêndice 6.

### 4.1 PRIMEIRA ITERAÇÃO

A primeira Reunião de Revisão e Priorização do *Backlog* de Produto, que nesta iteração se refere à reunião de planejamento, proporcionou a especificação e adição de três histórias de usuário ao *backlog* de produto: (i) as histórias de número 1 e 2 da Tabela 1; e (ii) a história de número 6 da Tabela 2 (presente na Seção 4.2).

Tabela 1 – Histórias de Usuários da Primeira Iteração

Nº	História de Usuário	Critérios de aceitação
1	Como professor/coordenador pedagógico, quero solicitar uma sessão, para obter um agendamento e desfrutar de uma sessão remota.	(i) A solicitação deve ser feita por um formulário; (ii) a escolha da data e hora de uma sessão deve ser um calendário, o qual: (a) os dias que já tiverem todos os horários com agendamento são desabilitados e marcados com vermelho; (b) os dias de final de semana são desabilitados e marcados com cinza; e (c) quando selecionado um dia, deve-se exibir apenas os horários disponíveis. (iii) exibir as sessões e tópicos abordados; e (iv) enviar um e-mail para quem agendou e um para o Planetário contendo as informações do agendamento.
2	Como professor/coordenador pedagógico, quero poder cancelar meu agendamento, para que caso ocorra algum imprevisto eu não precise enviar um e-mail ao Planetário.	(i) Deve-se inserir o identificador do agendamento para cancelar.

Fonte: Produção do Autor.

Das três histórias especificadas, as histórias 1 e 2 foram selecionadas e movidas ao *backlog* de *sprint* devido à urgência das mesmas. Elas correspondem ao processo de agendamento e cancelamento de agendamento de sessões virtuais e são prioritárias devido à necessidade de desde o primeiro momento permitir o agendamento por parte de instituições interessadas. A história 6, por sua vez, acabou não sendo selecionada para esta iteração devido à pouca urgência em conjunto com a estimativa de tempo de implementação das três histórias ser superior ao tempo da iteração.

Para que as histórias de usuário dessa iteração atendessem seus critérios de aceitação, duas páginas para o processo de agendamento foram criadas, além de um botão presente no cabeçalho de todas as páginas para o processo de cancelamento. A página inicial do sistema, contendo título e tópicos abordados em cada sessão virtual, aparece antes da página contendo o formulário de agendamento. A sinopse e recomendação não foram incluídas nesta página para não sobrecarregar a mesma. Nesta mesma página se encontra o botão para prosseguir ao formulário de agendamento, o qual contém todas as informações requeridas para a realização do agendamento. O conteúdo principal desta página pode ser visto na Figura 7.

Figura 7 – Conteúdo da Página Inicial da Primeira Iteração

**Agendamento de Sessões Virtuais**

**O QUE CADA SESSÃO ABORDA?**  
 Leia e encontre a sessão que inclua os tópicos que você mais deseja que sejam abordados. No momento do agendamento você deverá informar a sessão desejada.

- **Sessão 01. Stellarium – Simulação do Céu em Tempo Real**  
 Tópicos abordados: Reconhecimento do Céu; Pontos Cardeais; Efeito da Atmosfera; Constelações; Planetas Visíveis; Lua.
- **Sessão 02. Do Big Bang ao Sistema Solar**  
 Tópicos abordados: Big Bang; Galáxias; Estrelas; Sol; Planetas (detalhadamente); Asteroides; Meteoros; Tamanho do Sistema Solar.
- **Sessão 03. História da Astronomia e Instrumentos Observacionais**  
 Tópicos abordados: Telescópios (tipos e funcionamento); Ano-Luz; Geocentrismo; Heliocentrismo.
- **Sessão 04. Do Ponto de Vista da Terra**  
 Tópicos abordados: Rotação; Translação; Equinócios; Eclipses; Fases da Lua; Estações do Ano.
- **Sessão 05. Astronomia Moderna**  
 Tópicos abordados: Big Bang; Galáxias; Ciclo de Vida Estelar (Nebulosas, Supernovas, Buracos Negros); Fusão Nuclear; Elementos Químicos.
- **Sessão 06. Exploração Espacial**  
 Tópicos abordados: Missões Apollo; Corrida Espacial; Satélites; Foguetes; Estações Espaciais; Space Shuttle.
- **Sessão 07. Astrobiologia**  
 Tópicos abordados: Big Bang; Planetas gasosos e rochosos; Surgimento da Lua; Surgimento da água na Terra; Surgimento da vida na terra; Atmosfera primitiva; Características gerais dos seres vivos.

**CLIQUE AQUI PARA AGENDAR**

Fonte: Produção do autor.

Na página que contém o formulário de agendamento, a maioria das informações requeridas foram requisitadas através de “inputs” ou “selects” os quais são simples de implementar. A informação necessária que mais exigiu esforço de implementação foi a seleção de data e hora do agendamento. Para tal, foi utilizado como base o componente de seleção de data e hora da biblioteca Material-UI<sup>10</sup> disponível para React JS, simulando um calendário. A parte visual deste componente pode ser visualizado na Figura 8 e o código pode ser visualizado na Figura 9.

Figura 8 – Componente Calendário

**Escolha o dia e horário da sessão**


Dia e horário selecionado:

novembro 2021							Horário	
<	dom	seg	ter	qua	qui	sex	sab	
	31	1	2	3	4	5	6	
	7	8	9	10	11	12	13	
	14	15	16	17	18	19	20	
	21	22	23	24	25	26	27	
	28	29	30	1	2	3	4	

Fonte: Produção do autor.

<sup>10</sup> <https://mui.com/pt/>

Figura 9 – Código do Componente Calendário



```

<ReactDatePicker
  inline
  selected={dataHoraSessao}
  onChange={({dataSessao : Date} => setDataHoraSessao(dataSessao))
  onSelect={onSelectDate}
  locale="pt-BR"
  showTimeSelect
  timeCaption="Horário"
  filterDate={filterDate}
  filterTime={filterTime}
  excludeDates={diasSemAgendamentoDisponive }
  dayClassName={({dia})=>
    dia.getTime()=== dataHoraSessao?.getTime() ?
    "react-datepicker__day--selecte   :
    <{diasSemAgendamentoDisponive }.find(diaIndisponivel => diaIndisponivel.getTime() === dia.getTime())
    ||| (dia.getDay() !== 0 && dia.getDay()!==6) ?
    "unavailableDay" :
    (filterDate(dia) ? "availableDay": null)
  )
}
excludeTimes={horariosIndisponivei }
dateFormat="dd/MM/yyyy
required HH:mm"
/>

```

Fonte: Produção do autor.

A partir desse componente base, foram implementadas várias funções com o propósito de filtrar datas e horários que não estavam disponíveis para um novo agendamento. A função “filterDate” tem a função de não permitir um agendamento para sessões em dias de finais de semana nem para sessões com pelo menos dois dias a partir do dia atual, deixando os mesmos em cinza no componente calendário. A função “filterTime” tem a função de passar ao componente apenas os horários definidos pela equipe do Planetário como possíveis para um novo agendamento. A função “getUnavailableAgendamentosDates” tem a função de requisitar à API as datas e horários dos futuros agendamentos para que as mesmas não se tornassem disponíveis para agendamento no componente. Como resultado disto, os dias que não contém mais horários livres para agendamento se tornam vermelhos e não-clicáveis, enquanto os dias verdes e clicáveis, contém pelo menos um horário livre, mostrando o(s) horário(s) livre(s) na aba de horários do componente. Estas três funções podem ser visualizadas na Figura 10.

Figura 10 – Funções de Filtro de Horários e Datas Indisponíveis

(a) Função “filterDate”

```

const filterDate = (date: Date) => {
  const day = getDay(date);
  return day !== 0 && day !== 6 && date > addDays(new Date(),2);
}

```

(b) Função “filterTime”

```

const filterTime = (time: Date): boolean => {
  const sameHorario = horariosDisponiveis.find(horario=>horario.toLocaleTimeString()===time.toLocaleTimeString());
  return dataHoraSessao !== null && dataHoraSessao !== undefined && sameHorario !== undefined;
};

```

(c) Função “getUnavailableAgendamentosDates”

```

const getUnavailableAgendamentosDates = async () => {
  try {
    const { data } = await api.get<string[]>(`/agendamentos/GetUnavailableHorariosVirtuais`);
    const futureAgendamentos : Date[] = data.map((d: string)> new Date(d));
    setFuturosAgendamentos(futureAgendamentos);
    const diasSemAgendamentoComMaisDeUmHorarioString = futureAgendamentos
      .map(function (date) { return format(date,"dd/MM/yyyy") })
      .filter(function (date, i, array) {
        return array.indexOf(date) !== i;
      });
    const diasSemAgendamentoComMaisDeUmHorarioDate = diasSemAgendamentoComMaisDeUmHorarioString.map(dia=> parse
      (dia, 'dd/MM/yyyy', new Date()));
    setDiasSemAgendamentoDisponivel(diasSemAgendamentoComMaisDeUmHorarioDate);
    setLoaded(true);
  } catch (error:any) {
    console.error(error);
    return Swal.fire({
      title: 'Erro!',
      text: `${error}`,
      icon: 'error'
    });
  }
}

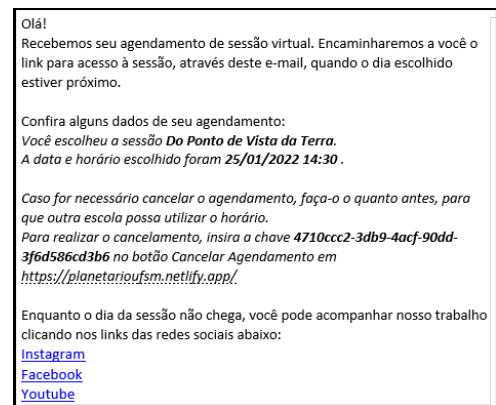
```

Após o preenchimento das informações necessárias para o agendamento, pode-se clicar no botão de agendar, também presente na página, a qual envia a solicitação à API, que cria este novo agendamento e encaminha dois e-mails utilizando o servidor SMTP do Google<sup>11</sup>, ilustrados na Figura 11. Um dos e-mails vai para o Planetário da UFSM, avisando sobre o novo agendamento e contendo as informações preenchidas por quem agendou, bem como a chave de cancelamento do agendamento. Esta chave de cancelamento serve para caso quem agendou deseje cancelar o agendamento de imediato. Logo, esta chave também é incluída no e-mail enviado à quem agendou. Neste e-mail, também estão contidas duas informações: (i) a data e hora do agendamento; e (ii) a sessão escolhida.

Figura 11 – E-mails Enviado ao Planetário



(a) E-mail de Aprovação



(b) E-mail Enviado à Quem Agendou

Fonte: Produção do autor.

Para o processo de cancelamento de agendamento, o botão criado no cabeçalho de todas as páginas do sistema para cancelamento de um agendamento tem função de abrir na mesma página um pequeno *pop-up* que exige a chave de cancelamento. Este *pop-up* pode ser visualizado na Figura 12.

<sup>11</sup> <https://www.google.com.br/>

Figura 12 – Pop-up de Cancelamento de Agendamento

Fonte: Produção do autor.

Ao clicar em “Realizar cancelamento”, uma requisição é enviada à API fazendo com que o agendamento seja excluído logicamente, ou seja, ele permanece no banco de dados porém com a propriedade booleana “Excluído” marcada como verdadeira. Além disso, a API também envia dois e-mails para o Planetário da UFSM e para quem agendou, ambos e-mails informando do cancelamento do agendamento.

Após a implantação das funcionalidades, as mesmas foram validadas pela equipe do Planetário, tendo como resultado o aceite de todas as histórias de usuário.

## 4.2 SEGUNDA ITERAÇÃO

Após a validação da primeira iteração, a qual possibilitou o uso e testes de aceitação das funcionalidades desenvolvidas, a equipe do Planetário estava apta a responder todas as perguntas pré-definidas para a Reunião de Revisão e Priorização do *Backlog* de Produto, ou seja, o primeiro *feedback* do sistema já em uso foi recebido. Logo, na Reunião de Revisão e Priorização do *Backlog* de Produto identificou-se: (i) uma história de usuário que alterava uma funcionalidade já desenvolvida (história 5 na Tabela 2); e (ii) quatro histórias de usuário para novas funcionalidades, correspondentes às histórias 3, 4, 5 da Tabela 2 e história 7 da Tabela 3 da Seção 4.3..

A fim de todas as histórias de usuário caberem em uma iteração, quatro das cinco histórias identificadas foram selecionadas para esta iteração, como ilustrado na Tabela 2.

Tabela 2 – Histórias de Usuários da Segunda Iteração

Nº	História de Usuário	Critérios de aceitação
3	Eu como monitor do Planetário, quero efetuar login no sistema, para poder gerenciar recursos privados	(i) Deve ser feita com o e-mail e senha definidos pelos monitores do Planetário; e (ii) após o login, devo ser redirecionado à área do monitor.
4	Eu como monitor do Planetário, quero efetuar logout no sistema, para caso eu queira não deixar meu navegador logado no sistema	(i) Deve existir um botão de logout na área do monitor.
5	Eu como professor/coordenador pedagógico quero visualizar as informações de cada sessão no momento do preenchimento do formulário para que eu não precise lembrar das mesmas no momento do preenchimento do formulário	(i) Deve-se mostrar título, descrição, tópicos abordados e recomendação; (ii) cada tipo de sessão deve ser um radio button; e (iii) quando clicar em um tipo, deve-se mostrar as informações da mesma.
6	Eu como monitor do Planetário, quero visualizar uma tabela de agendamentos para obter conhecimento dos mesmos	(i) Deve-se mostrar a tabela na área do monitor; (ii) a tabela deve mostrar as seguintes informações: (a) data e hora de sessão; (b) sessão escolhida; (c) instituição; (d) série/ano; (e) número de participantes; e (f) professor; (iii) a tabela deve ser ordenada por data e hora de sessão em ordem decrescente, ou seja, a data mais tardia deve ser mostrada por primeiro; e (iv) a tabela apenas deve mostrar os agendamentos não excluídos.

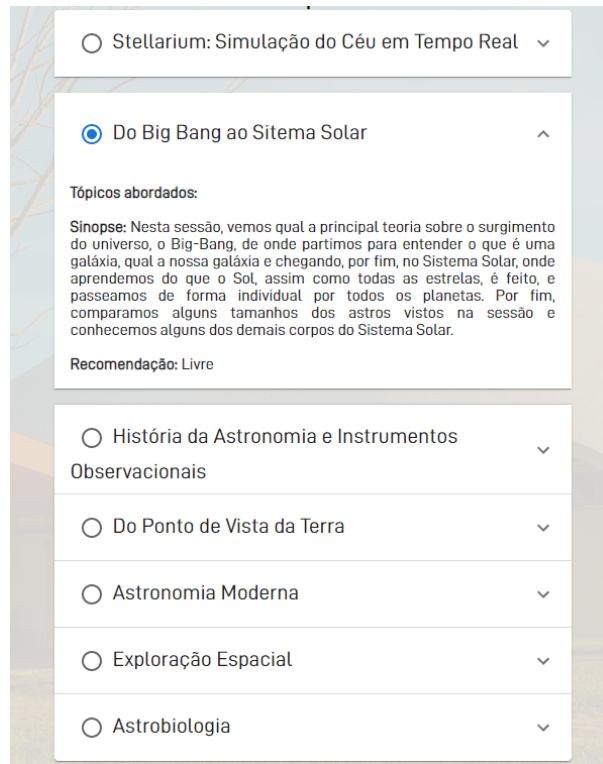
Fonte: Produção do Autor.

Com relação à história 5, que altera uma funcionalidade desenvolvida, a solução encontrada para mover a listagem de sessões presente na página inicial (Figura 7) para a página que contém o formulário de agendamento foi o uso de acordeões, deixando a página inicial somente com o botão de encaminhamento à página contendo formulário de agendamento. Esta solução resolve o problema de falta de espaço descrito anteriormente. Portanto, no formulário de agendamento, quando um acordeão está fechado, o mesmo mostra um *radio button*, bem como o título da sessão. Quando há um clique sobre o acordeão, o mesmo se abre mostrando as outras informações. Caso um acordeão esteja aberto ao abrir um outro, o mesmo se fecha automati-



camente. Desta forma, o formulário de agendamento não fica sobrecarregado de informações sobre as sessões. Os acordeões podem ser visualizados na Figura 13.

Figura 13 – Acordeões de Sessões no Formulário de Agendamento



Fonte: Produção do autor.

Para a história de usuário que envolve efetuar login no sistema (história 3), foi definido na Reunião de Revisão e Priorização do *Backlog* de Produto que deveria existir apenas um usuário, com e-mail e senha já definidos no momento da reunião. Este e-mail e senha foram armazenados como variáveis de ambiente no portal da Azure. A página de login construída pode ser acessada por um botão “Área do Monitor” cabeçalho presente em todas as páginas do sistema. Nesta página, quando o usuário clica em “Entrar”, as informações preenchidas (e-mail e senha) são enviadas ao *back end* a qual autentica de acordo com as variáveis de ambiente, gera um token JWT (Json Web Token) que expira após 24 horas e retorna este token ao *front-end*, o qual armazena-o e usa-o para realizar requisições aos métodos do *back end* que exigem autorização (geralmente os que retornam informações sensíveis). Quando o *front end* faz uma requisição ao *back end* e este retorna que a requisição não foi autorizada, o *front end* direciona o usuário à página de login. O formulário de login pode ser visualizado na Figura 14.

Figura 14 – Formulário de login



Login

Seu email \*

Sua senha \*

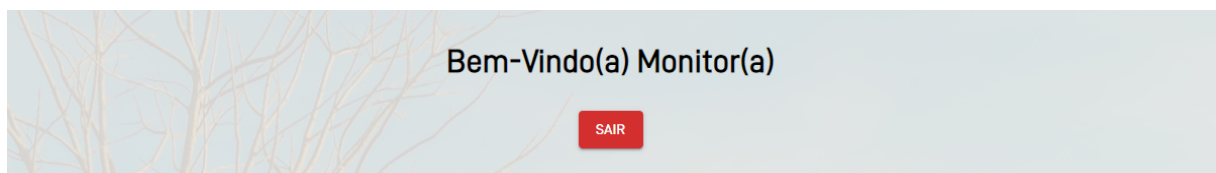
**ENTRAR**

**i** Esta área é destinada aos monitores do Planetário

Fonte: Produção do autor.

Para a história de usuário que diz respeito ao logout no sistema, foi criado o botão “Sair” na área do monitor. Este botão tem por função excluir o token armazenado no *front end* e direcionando o usuário para a página de login. Ele pode ser visualizado na Figura 15. Para a história de usuário que envolve a visualização de uma tabela de agendamentos, a implementação da mesma gerou como resultado a tabela que pode ser observada na Figura 16.

Figura 15 – Botão de Logout da Área do Monitor



Fonte: Produção do autor.

Figura 16 – Tabela de Agendamentos da Área do Monitor

Agendamentos					
Data e hora	Sessão Escolhida	Instituição	Série/Ano	Nº de participantes	Professor(a)
01/12/2021 10h00	Exploração Espacial	Polo tecnológico de Balneário Camboriú	6º e 7º ano do ensino fundamental	20	Katrine
25/11/2021 15h00	História da Astronomia e Instrumentos Observacionais	Escola Viva	5 ano do Ensino Fundamental	54	Samuel
24/11/2021 15h00	Do Big Bang ao Sistema Solar	Colégio Franciscano Sant'Anna	3º ano	95	Fabiana
24/11/2021 10h00	Do Big Bang ao Sistema Solar	Colégio Franciscano Sant'Anna	3º ano	25	Caroline

Linhas por página: 5 6-10 de 48 |< < > >|

Fonte: Produção do autor.

Após a implantação das funcionalidades desenvolvidas nesta iteração, todas as histórias de usuário foram aprovadas pela equipe do Planetário.

### 4.3 TERCEIRA ITERAÇÃO

A Reunião de Revisão e Priorização do *Backlog* de Produto desta iteração resultou na identificação das histórias de usuário 8, 9 e 10 presentes na Tabela 3 e na da história de usuário 13 presente na Tabela 4 (Seção 4.4). Foram selecionadas as histórias 7, 8, 9 e 10 para serem implementadas. Nesta iteração não há nenhuma história de usuário a qual altera alguma funcionalidade já desenvolvida.

Tabela 3 – Histórias de Usuários da Terceira Iteração

Nº	História de Usuário	Critérios de aceitação
7	Eu como monitor do Planetário quero poder visualizar, editar e excluir um agendamento para que eu possa manter um agendamento atualizado	(i) As ações de visualizar, editar e excluir devem estar localizadas na tabela de agendamentos da área do monitor, em uma coluna de ações; (ii) as ações de visualizar e editar um agendamento devem abrir um modal com as todas as informações do agendamento a serem visualizadas ou editadas; e (iii) a ação de excluir um agendamento deve exibir um alerta de confirmação da ação.
8	Eu como monitor do Planetário, quero poder manter (criar, visualizar, editar e excluir) as sessões para que haja agendamentos para novos tipos de sessões e para que as sessões já existentes fiquem atualizadas.	(i) As ações de visualizar, editar e excluir devem estar localizadas na tabela de sessões da área do monitor, em uma coluna de ações; (ii) a ação de criar uma sessão deve estar localizada ao final dos registros mostrados na tabela; (iii) as ações de visualizar e editar uma sessão devem abrir um modal com as todas as informações da sessão a serem visualizadas ou editadas; e (iv) a ação de excluir uma sessão deve exibir um alerta de confirmação da ação.
9	Eu como monitor do Planetário, quero poder manter (criar, visualizar, editar e excluir) os horários disponíveis para que haja novos agendamentos apenas para os horários especificados.	(i) As ações de visualizar, editar e excluir devem estar localizadas na tabela de horários disponíveis da área do monitor, em uma coluna de ações; (ii) a ação de criar um horário disponível deve estar localizada ao final dos registros mostrados na tabela; (iii) as ações de visualizar e editar um horário disponível devem abrir um modal com as todas as informações do horário a serem visualizadas ou editadas; e (iv) a ação de excluir um horário deve exibir um alerta de confirmação da ação.
10	Eu como professor/coordenador pedagógico quero poder solicitar tanto uma sessão virtual quanto uma sessão presencial para que dependendo da minha escolha, seja realizado um agendamento.	(i) A escolha deve ser feita na página inicial do sistema; (ii) deve haver uma página para gerenciamento de agendamentos, sessões e horários de sessões virtuais e outra para gerenciamento de agendamentos, sessões e horários de sessões presenciais, as duas devem poder ser acessadas pela área do monitor.

Fonte: Produção do Autor.

Para as três primeiras histórias de usuário da Tabela 3, as implementações resultaram em: (i) inclusão da coluna de ações na tabela de agendamentos como mostrado na Figura 17; (ii) criação da tabela de sessões com a coluna de ações e botão de criação como mostrado na Figura 18; e (iii) criação da tabela de horários disponíveis com a coluna de ações e botão de

criação como mostrado na Figura 19.

Figura 17 – Ações na Tabela de Agendamentos da Área do Monitor

Agendamentos Virtuais						
Data e hora	Sessão Escolhida	Instituição	Série/Ano	Nº de participantes	Professor(a)	Ações
08/02/2022 15h00	Do Big Bang ao Sistema Solar	Escola Notre Dame Menino Jesus	Outro	25	Matheus e Adélia	  
07/02/2022 15h00	Stellarium: Simulação do Céu em Tempo Real	Escola Notre Dame Menino Jesus	Outro	25	Matheus e Adélia	  
20/01/2022 07h31	História da Astronomia e Instrumentos Observacionais	eeeee	3º Ano do Ensino Fundamental	50	Jose	  
10/01/2022 10h00	Stellarium: Simulação do Céu em Tempo Real	Rodrigo Diniz do Prado	1	20	Tatiana	  

Linhas por página: 5 1-5 de 48 |< < > >|

Fonte: Produção do autor.

Figura 18 – Tabela de Sessões da Área do Monitor

Sessões Virtuais					
ID	Título	Sinopse	Recomendação	Tópicos	Ações
1	Stellarium: Simulação do Céu em Tempo Real	A sessão consiste em simular o céu local em tempo real, uti...	Livre		  
2	Do Big Bang ao Sistema Solar	Nesta sessão, vemos qual a principal teoria sobre o surgime...	Livre		  
3	História da Astronomia e Instrumentos Observacionais	A sessão consiste em uma apresentação geral sobre a histó...	A partir de 12 anos de idade		  
4	Do Ponto de Vista da Terra	Nesta sessão, partimos da concepção do sistema solar, para ...	A partir de 10 anos de idade		  
5	Astronomia Moderna	É aqui onde vemos o que a astronomia estuda atualmente. P...	A partir de 14 anos de idade		  

Linhas por página: 5 1-5 de 7 |< < > >|

Fonte: Produção do autor.

Figura 19 – Tabela de Horários Disponíveis da Área do Monitor

Horários Disponíveis para Sessões Virtuais	
Horário	Ações
10h:00	  
15h:00	  

Linhas por página: 5 1-2 de 2 |< < > >|

Fonte: Produção do autor.

As ações de criação, visualização e edição abrem o mesmo componente React “Create-ReadUpdateModal”, um modal que se adapta de acordo com a ação e de acordo com o recurso, seja ele agendamento, sessão ou horário disponível. Este modal pode ser visualizado na Figura 20.

Figura 20 – Modal de Ação

The figure displays three screenshots of a modal form titled 'Criar', 'Informações', and 'Editar'. Each screenshot shows a form with a title field, a synopsis field, a topics field, and a recommendation field. A 'Virtual' toggle is present in the 'Criar' and 'Editar' views. A 'SALVAR' button is located at the bottom of each form. The 'Informações' view shows the synopsis field filled with text describing the Stellarium simulation.

(a) Ação de Criar

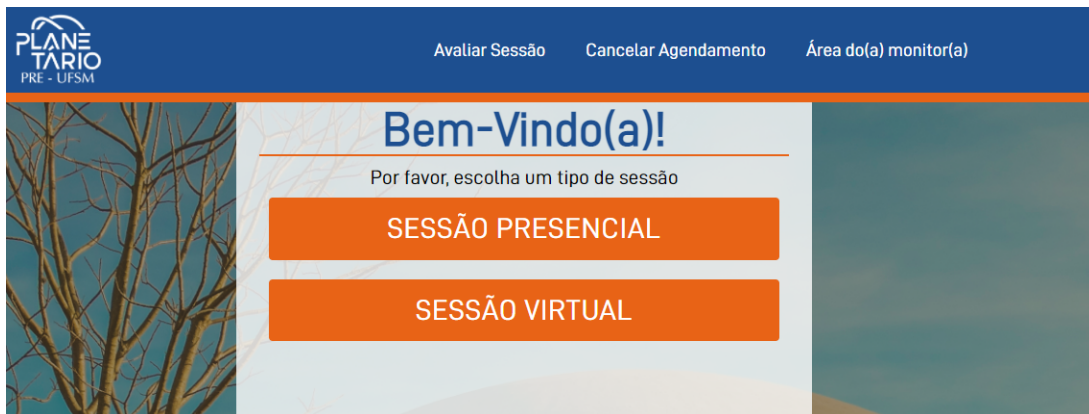
(b) Ação de Visualizar

(c) Ação de Editar

Fonte: Produção do autor.

Até esta iteração, todas as funcionalidades diziam respeito às sessões virtuais. Com a quarta história de usuário (história 10) da Tabela 3, os agendamentos, sessões e horários disponíveis foram separados em virtuais e presenciais através de um marcador booleano “Virtual” em cada um. Com isso, as tabelas de agendamento, sessões e horários disponíveis mostradas anteriormente foram duplicadas e adaptadas de acordo com este marcador. Estas tabelas foram separadas em duas páginas diferentes, uma para cada tipo de sessão (virtual ou presencial) e podem ser acessadas por meio de dois botões presentes na área do monitor. Além disso, na página inicial do sistema, o botão de encaminhamento para a página contendo o formulário de agendamento, foi substituído por dois botões, um para encaminhar para o formulário de agendamento de sessões virtuais e outro para o agendamento de sessões presenciais. A Figura 21 ilustra a página inicial do sistema após a implementação desta iteração.

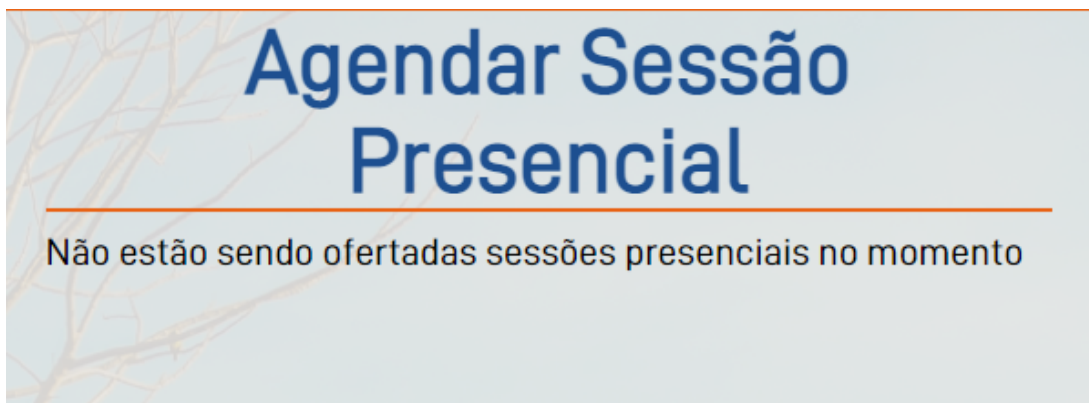
Figura 21 – Botões de Encaminhamento na Página Inicial do Sistema



Fonte: Produção do autor.

Neste momento, por consequência da pandemia de Covid-19, as sessões presenciais não são realizadas. Portanto, foi adicionada a regra de que se não houvessem horários disponíveis, tanto de sessões presenciais quanto virtuais, não haveria possibilidade de ser realizado algum agendamento. Por isso, com a falta de horários disponíveis para os agendamentos presenciais, a página contendo o formulário de agendamento exibe a mensagem “Não estão sendo ofertadas sessões presenciais no momento”, como ilustrado na Figura 22.

Figura 22 – Mensagem de Aviso de Sem Horarios Diponiveis



Fonte: Produção do autor.

Após a implantação, a equipe do Planetário utilizou o sistema e validou as funcionalidades desenvolvidas, aceitando todas as histórias de usuário nos testes de aceitação.

#### 4.4 QUARTA ITERAÇÃO

Na Reunião de Revisão e Priorização do *Backlog* de Produto foram especificadas as histórias 11 e 12 da Tabela 4 e a história 15 da Tabela 5. Todas elas correspondem à novas funcionalidades. As histórias de usuário 11 e 12 foram definidas pelos *stakeholders* como sendo as mais urgentes. Devido à estimativa de tempo para essas história ser pequena, a história 13 também foi selecionada para esta iteração.

Tabela 4 – Histórias de Usuários da Quarta Iteração

Nº	História de Usuário	Crítérios de aceitação
11	Eu como professor/coordenador pedagógico quero poder solicitar uma sessão em um horário diferente para que eu desfrute de uma sessão em um momento não pré-definido pelo monitores.	(i) A opção de solicitar um horário diferente deve estar presente no formulário de solicitação; (ii) o agendamento não deve constar na tabela de agendamentos da área do monitor; e (iii) deve-se avisar que o agendamento está pendente no e-mail enviado após a solicitação.
12	Eu como monitor do Planetário quero poder aceitar ou recusar uma solicitação de sessão em um horário diferente para que eu possa proporcionar uma sessão em um horário não pré-definido.	(i) Os agendamentos pendentes de aprovação ou recusa devem estar na página principal da área do monitor; e (ii) um e-mail deve ser enviado ao solicitante assim que a ação de aprovação e recusa for efetuada avisando sobre a situação;
13	Eu como professor/coordenador pedagógico quero receber o link da sessão virtual para que possa tê-lo desde o primeiro momento e compartilhá-lo.	(i) O link da sessão virtual deve ser uma reunião <i>online</i> pela plataforma Google Meet; e (ii) o link deve ser recebido no e-mail de confirmação de agendamento.

Fonte: Produção do Autor.

Para a implementação da história 11, uma caixa de seleção foi adicionada. Quando o usuário seleciona a opção, o componente de calendário desenvolvido na história 1 é substituído por um seletor padrão de data e hora. Com isso, o usuário pode selecionar a data e horário que quiser, visto que tal agendamento precisará ser aprovado posteriormente. A caixa de seleção e o seletor podem ser visualizados na Figura 23.



Figura 23 – Caixa de Seleção e Seletor

**Escolha o dia e horário da sessão**  
Dia e horário selecionado: 28/01/2022 10:00

28/01/2022 10:00

Solicitar um agendamento em um horário diferente (Mediante aprovação do Planetário da UFSM.)

Fonte: Produção do autor.

Para a implementação da história 12, uma tabela a qual contém os agendamentos pendentes de aprovação foi adicionada à área do monitor, como pode ser visualizado na Figura 24.

Figura 24 – Tabela de Agendamentos Pendentes de Aprovação

**Bem-Vindo(a) Monitor(a)**

**SESSÕES PRESENCIAIS** **SESSÕES VIRTUAIS**

**Agendamentos Pendentes de Aprovação**

Data e hora	Sessão Escolhida	Instituição	Série/Ano	Nº de participantes	Professor(a)	Ações
17/02/2022 15h00	Do Ponto de Vista da Terra	E.M.E.F	3º Ano do Ensino Fundamental	50	jose	<b>APROVAR</b> <b>RECUSAR</b>

Linhas por página: 5 1-1 de 1 |< < > >|

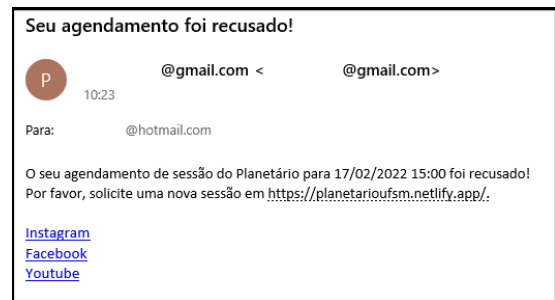
Fonte: Produção do autor.

Quando o agendamento é aprovado ou recusado por um monitor do Planetário, um e-mail informando a instituição interessada sobre a decisão é enviado. Os e-mails de aprovação e recusa podem ser visualizados na Figura 25.

Figura 25 – E-mails de Aprovação ou Recusa



(a) E-mail de Aprovação



(b) E-mail de Recusa

Fonte: Produção do autor.

A história 13 não foi possível implementar devido ao fato de que o objetivo deste trabalho é manter o sistema operacional de modo gratuito. Visto que a única maneira de criar um link do Google Meet<sup>12</sup> é a criação de um evento no Google Calendário<sup>13</sup>. Este por sua vez possui uma API que apenas permite a criação de um link do Google Meet se o usuário habilitar a delegação de autoridade para uma conta de serviço, um recurso disponível apenas para usuários do Google Workspace<sup>14</sup>, o qual é pago. Por isso, esta história foi totalmente reprovada nos testes de aceitação.

Portanto, apenas duas das três histórias de usuário foram aprovadas nos testes de aceitação realizados após a implantação do sistema.

#### 4.5 QUINTA ITERAÇÃO

Na última iteração, a Reunião de Revisão e Priorização do *Backlog* de Produto especificou as histórias 14, 16 e 17 da Tabela 5, todas correspondentes à novas funcionalidades. Estas e a história de usuário de número 15, também presente nesta tabela, foram selecionadas para compor o *Backlog* de *Sprint*.

<sup>12</sup> <https://meet.google.com/>

<sup>13</sup> <https://calendar.google.com/>

<sup>14</sup> <https://workspace.google.com/>

Tabela 5 – Histórias de Usuários da Quinta Iteração

Nº	História de Usuário	Critérios de aceitação
14	Como monitor do Planetário, quero manter (criar, visualizar, editar e excluir) horários indisponíveis, para que não haja novos agendamentos nos horários especificados.	(i) O componente calendário do formulário de solicitação de agendamento não deve permitir selecionar algum horário indisponível; (ii) o sistema não deve permitir o agendamento em um horário indisponível com exceção de agendamentos que solicitem uma sessão em horário diferente; (iii) as ações de visualizar, editar e excluir devem estar localizadas na tabela de horários indisponíveis da área do monitor, em uma coluna de ações; (iv) a ação de criar um horário indisponível deve estar localizada ao final dos registros mostrados na tabela; (v) as ações de visualizar e editar um horário indisponível devem abrir um modal com as todas as informações do horário a serem visualizadas ou editadas; e (vi) a ação de excluir um horário deve exibir um alerta de confirmação da ação.
15	Como monitor do Planetário, quero poder gerar um relatório para que eu possa ter mais informações/estatísticas sobre os agendamentos de um determinado período.	(i) Deve ser necessário selecionar um período de tempo; (ii) o relatório deve conter as seguintes informações: (a) quantos alunos foram atendidos; (b) quantos agendamentos foram realizados para cada sessão; (c) quantos agendamentos foram realizados para tipo de escola; e (d) quantos agendamentos foram realizados para cada série/ano; (iii) o relatório gerado deve estar no formato .CSV.
16	Como professor/coordenador pedagógico, quero poder avaliar a sessão após seu término, para que eu possa dar um <i>feedback</i> sobre como foi a sessão.	(i) Deve-se inserir o identificador do agendamento; (ii) deve-se registrar a data e hora da avaliação; (iii) avaliação utilizando uma nota em escala de 1 à 5 e crítica em formato de texto; e (iv) deve ser possível avaliar apenas após o horário do agendamento.
17	Como monitor do planetário, quero poder visualizar as avaliações, para que eu possa obter <i>feedback</i> sobre as sessões realizadas.	(i) As avaliações deve estar localizadas na tabela de avaliações na área do monitor; (ii) a tabela de avaliações deve conter as informações de data e hora da sessão, sessão escolhida e se a sessão foi virtual ou presencial, além das informações da avaliação; (iii) a ação de visualizar a avaliação deve estar localizada na tabela de avaliações, em uma coluna de ações; e (iv) as avaliações devem estar ordenadas por data e hora de avaliação, as mais recentes primeiro.

Fonte: Produção do Autor.

No que se refere à história 14, a tabela e suas ações descritas nos últimos quatro crité-

rios de aceitação foram implementadas de maneira semelhante às outras tabelas presentes no sistema e descritas anteriormente. Em relação ao critério de aceitação que se refere ao componente calendário, a função “getUnavailableAgendamentosDates” do componente calendário, descrita na seção da primeira iteração, foi estendida para requisitar também os horários indisponíveis, impossibilitando o agendamento nestes horários. No que se refere ao segundo critério de aceitação, o fato de a solicitação em um horário diferente não utilizar o componente calendário, mas sim um seletor de data, permite o agendamento neste horário indisponível.

Quanto à história 15, foi criada uma página de gerar relatório, acessível por um botão na área do monitor, a qual exige uma data inicial e uma data final ao usuário. Após a inserção destas datas, o usuário pode gerar o relatório clicando no botão “Gerar”, a qual envia as mesmas à API. A API por sua vez realiza consultas ao banco, como mostrado na Figura 26, e retorna um arquivo .CSV contendo as informações descritas nos critérios de aceitação desta história. O arquivo .CSV gerado para a data inicial de 1/12/2021 a 31/1/2022 é ilustrado na Figura 27.

Figura 26 – Consultas de Informações Necessárias ao Arquivo CSV

```

public FileResult CreateRelatorio(CreateRelatorioRequest createRelatorioRequest)
{
    var totalAlunosPresentes = _dbContext.Agendamentos
        .Where(agenda =>
            !agenda.Excluido &&
            agenda.DataHoraSessao > createRelatorioRequest.DataInicial &&
            agenda.DataHoraSessao < createRelatorioRequest.DataFinal)
        .Sum(agenda => agenda.AlunosPresentes);

    var sessoesEscolhidas = _dbContext.Agendamentos
        .Where(agenda =>
            !agenda.Excluido &&
            agenda.DataHoraSessao > createRelatorioRequest.DataInicial &&
            agenda.DataHoraSessao < createRelatorioRequest.DataFinal)
        .Join(_dbContext.Sessoes, a => a.SessaoEscolhidaId, s => s.Id, (agenda, sessao) => new { sessao.Titulo })
        .GroupBy(s => s)
        .Select(s => new {
            s.Key.Titulo,
            Count = s.Count()
        });

    var seriesEscolhidas = (from agenda in _dbContext.Agendamentos
        where !agenda.Excluido && agenda.DataHoraSessao > createRelatorioRequest.DataInicial
            && agenda.DataHoraSessao < createRelatorioRequest.DataFinal
        group agenda by agenda.Serie into g
        select new
        {
            Serie = g.Key,
            Count = g.Count()
        });

    var tiposEscola = (from agenda in _dbContext.Agendamentos
        where !agenda.Excluido && agenda.DataHoraSessao > createRelatorioRequest.DataInicial
            && agenda.DataHoraSessao < createRelatorioRequest.DataFinal
        group agenda by agenda.TipoEscola into g
        select new
        {
            Tipo = g.Key,
            Count = g.Count()
        });
}

```

Fonte: Produção do autor.

Figura 27 – Arquivo CSV Gerado

	A	B
1	Total de Alunos Presentes	47
2		
3	Sessão	Quantidade
4	Do Ponto de Vista da Terra	1
5	Exploração Espacial	1
6	Stellarium: Simulação do Céu em Tempo Real	1
7		
8	Série	Quantidade
9	1º Ano do Ensino Fundamental	1
10	Múltiplas	2
11		
12	Tipo de Escola	Quantidade
13	Municipal	1
14	Estadual	1
15	Escola Especial	1
16		

Fonte: Produção do autor.

Para a história 16, um botão escrito “Avaliar Sessão” no cabeçalho de todas as páginas do sistema foi adicionado. Quando o professor ou coordenador pedagógico clica no mesmo, este abre um pequeno modal o qual é possível avaliar uma sessão já realizada inserindo o identificador do agendamento, o qual é a chave de agendamento enviada no momento da solicitação do mesmo, além da nota e crítica, positiva ou negativa, referente à sessão. O modal de avaliação pode ser visualizado na Figura 28. A regra referente ao quarto critério de aceitação e o registro da data e hora da avaliação foram implementadas na API.

Figura 28 – Modal de Avaliação de Sessão

Fonte: Produção do autor.

No que diz respeito à história 17, a tabela ilustrada na Figura 29 foi produzida. Nela, as avaliações são ordenadas pelo momento de avaliação, mostrando as informações requeridas descritas nos critérios de aceitação, bem como a ação de visualizar na coluna de ações.

Figura 29 – Tabela de Avaliações da Área do Monitor

Avaliações						
Data e hora da sessão	Sessão escolhida	Tipo	Avaliado em	Crítica	Nota	Ações
25/11/2021 15h00	História da Astronomia e Instrumentos Observacionais	Virtual	19/01/2022 14h16	Ótima sessão, conteúdo muito bem explicado e didática excelente!	Excelente	
24/11/2021 15h00	Do Big Bang ao Sistema Solar	Presencial	17/01/2022 14h17	A sessão foi boa mas deixou a desejar quanto à didática.	Boa	

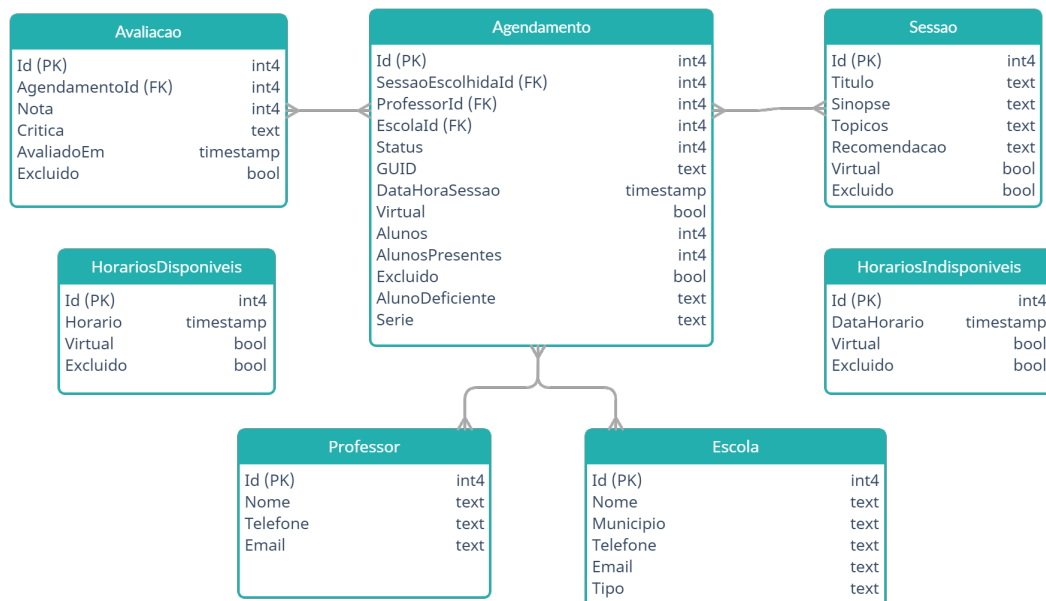
Linhas por página: 5 ▾ 1-2 de 2 < > >>

Fonte: Produção do autor.

#### 4.6 ESTADO FINAL DO BANCO DE DADOS

Após as cinco iterações, o diagrama de entidade-relacionamento do banco de dados, representado na Figura 30, se refere ao estado final do banco de dados.

Figura 30 – Diagrama Entidade-Relacionamento



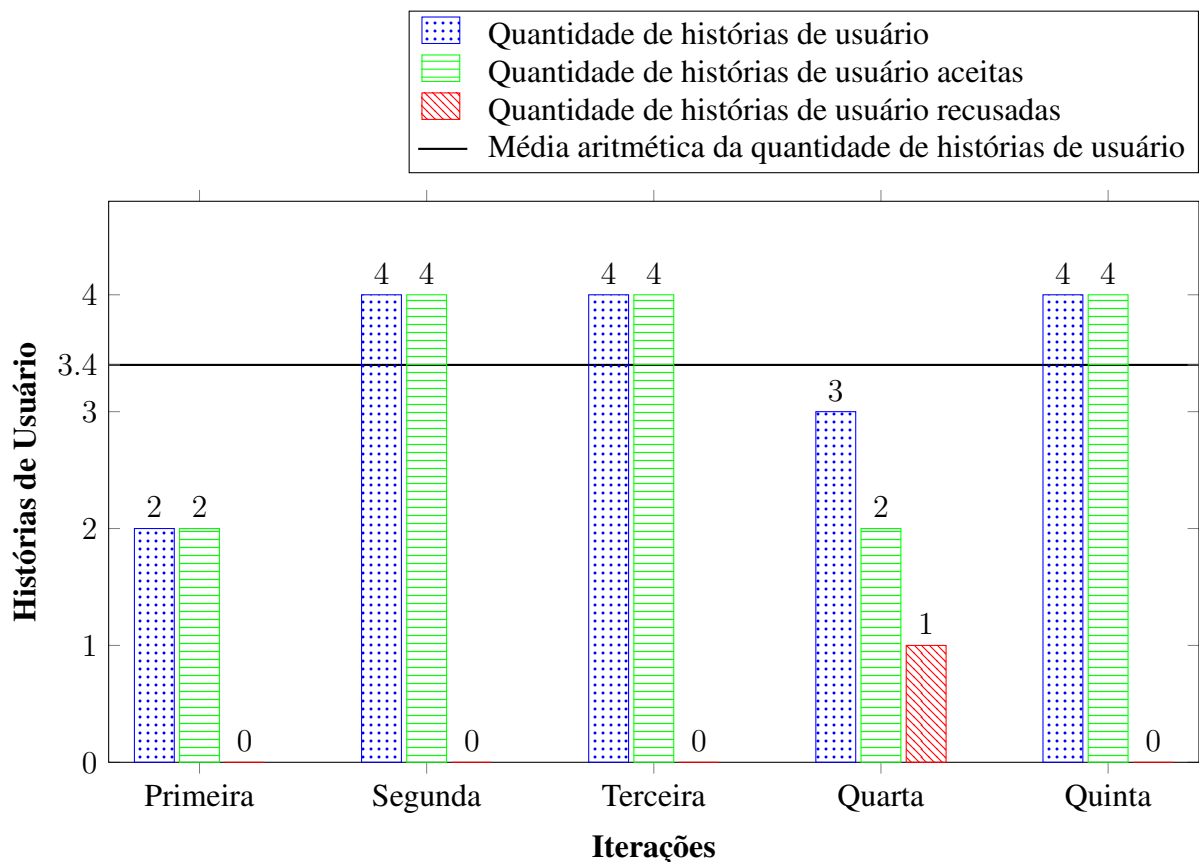
Fonte: Produção do autor.

Dessa maneira, é possível observar que, neste capítulo, foram abordadas: (i) as iterações previstas no trabalho, descrevendo as histórias de usuário especificadas e implementadas, bem como de que maneira as funcionalidades foram implementadas; (ii) o resultado da validação em cada iteração; e (iii) o estado final do banco de dados.

## 5 RESULTADOS E AVALIAÇÃO

Como resultado da implementação, a produção deste sistema colaborou para o acatamento de grande maioria das necessidades existentes identificadas durante o processo de identificação de requisitos. Durante todo o processo de desenvolvimento, a maior dificuldade foi concluir a iteração em duas semanas. Para isso, a urgência definida pela equipe do Planetário na Reunião de Revisão e Priorização do *Backlog* de Produto e a estimativa por parte do desenvolvedor foram fundamentais para a correta escolha das histórias de usuário de cada iteração. A quantidade total de histórias de usuário foi de 17, das quais 16 foram aceitas e 1 foi recusada, devido ao fato do objetivo específico deste trabalho ser manter o sistema de modo gratuito. A média de histórias de usuário por iteração foi de 3,4. A Figura 31 ilustra: (i) a quantidade de histórias de usuário; (ii) a quantidade de histórias de usuário aceitas; (iii) a quantidade de histórias de usuário recusadas por iteração; e (iv) a média aritmética da quantidade de histórias de usuário.

Figura 31 – Histórias de Usuário por Iteração

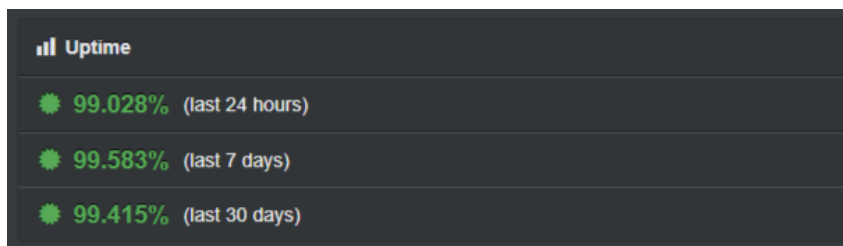


Fonte: Produção do autor.



O uso da metodologia e ferramentas utilizadas para a hospedagem proporcionou um sistema gratuito com disponibilidade de 99,4% em um mês, de acordo com o *bot* utilizado. Este dado nos indica que o sistema é capaz de operar, de maneira gratuita, satisfatoriamente no que diz respeito a um sistema que é usado de maneira esporádica, e não de maneira contínua, por seus usuários finais. O tempo de disponibilidade de um mês de operação do sistema pode ser visualizado na Figura 32.

Figura 32 – Tempo de Disponibilidade do Sistema

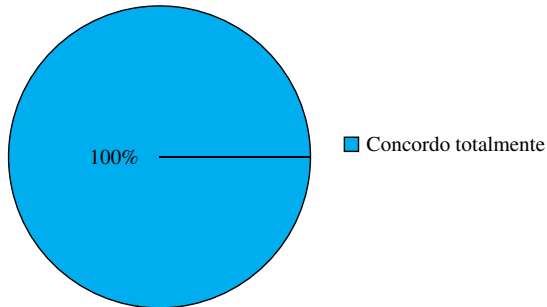


Fonte: Produção do autor.

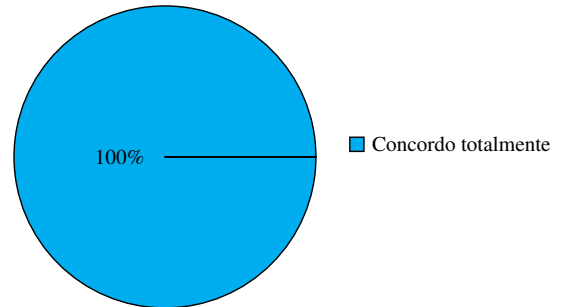
De acordo com a avaliação realizada em relação à usabilidade, percebe-se que todos e todas da equipe do Planetário da UFSM concordaram totalmente que o sistema é apropriado para suas necessidades, que é fácil aprender a usar o sistema, que o sistema é fácil de ser operado e controlado, e que a interface é agradável. Além disso, 40% dos monitores da equipe concordam parcialmente que o sistema te protege de errar, enquanto 60% concordam totalmente. A Figura 33 ilustra um gráfico de pizza para cada afirmação avaliada nesta parte do questionário.

Figura 33 – Avaliação de Usabilidade

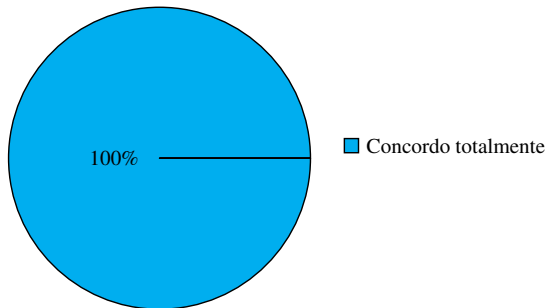
(a) O sistema é apropriado para suas necessidades



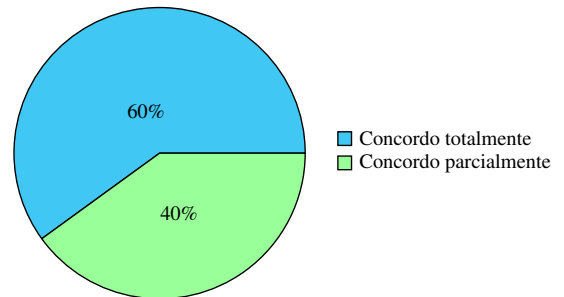
(b) É fácil aprender a usar o sistema



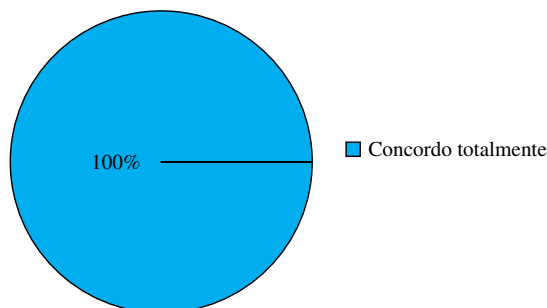
(c) É fácil de operar e controlar o sistema



(d) O sistema te protege de errar



(e) A interface é agradável



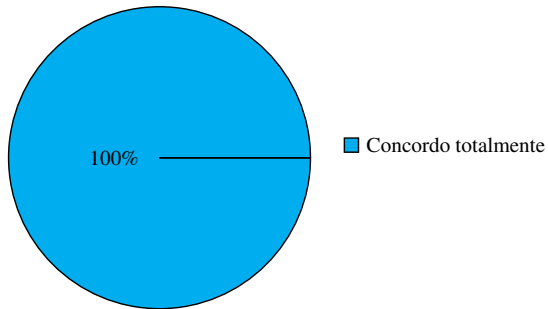
Fonte: Produção do autor.

De acordo com os resultados evidenciados na Figura 32, percebe-se que todos e todas da equipe do Planetário da UFSM concordam totalmente que o uso de histórias de usuário e seus critérios de aceitação facilitou a especificação e testagem dos requisitos. Eles e elas também concordaram totalmente que o tempo de entrega de duas semanas foi satisfatório, bem como que sua expectativa em relação ao sistema desenvolvido foi cumprida. Em relação à afirmação que questiona se o monitor ou monitora do Planetário da UFSM se envolveu com a Reunião de Revisão e Priorização do *Backlog* de Produto, cerca de 40% dos mesmos responderam que não concordam nem discordam da afirmação, enquanto outros 40% responderam que concordam

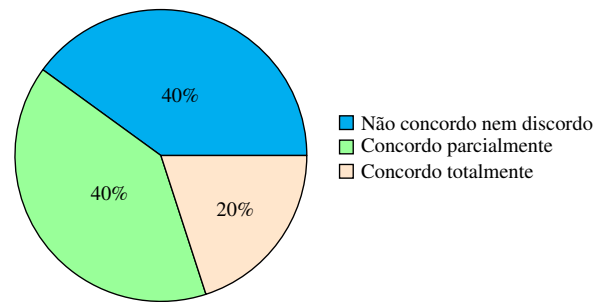
parcialmente, e outros 20% responderam que concordam totalmente.

Figura 34 – Avaliação de Satisfação

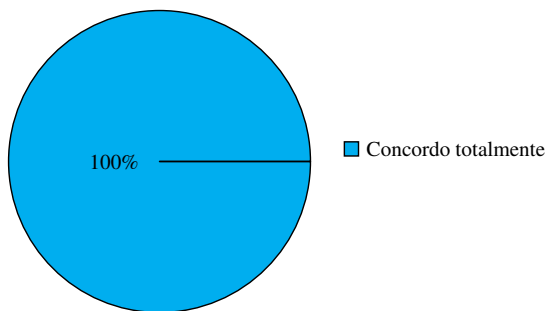
(a) O uso de histórias de usuário e seus critérios de aceitação facilitou a especificação e testagem dos requisitos



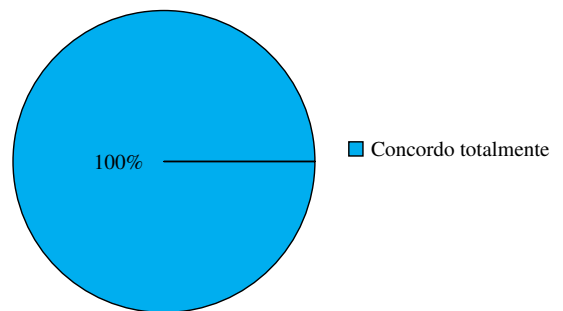
(b) Você se envolveu e se comprometeu com a Reunião de Revisão e Priorização do *Backlog* de Produto



(c) Você ficou satisfeito com tempo de entrega de duas semanas



(d) Sua expectativa foi cumprida em relação ao sistema desenvolvido



Fonte: Produção do autor.

## 6 CONCLUSÃO

O desenvolvimento do Sistema de Agendamento de Sessões para o Planetário da UFSM, a partir da constatação da inexistência de um sistema que atendesse todas as necessidades do Planetário da UFSM, possibilitou a aplicação prática do processo de desenvolvimento iterativo e incremental, bem como a de práticas e artefatos advindos de diferentes metodologias ágeis, mantendo, desta maneira, entregas frequentes aos *stakeholders*.

A aplicação dos métodos e ferramentas de hospedagem corroborou para um sistema sem despesas ao desenvolvedor e ao Planetário da UFSM, e com alto tempo de disponibilidade, cumprindo com o objetivo de manter operacional o sistema de forma gratuita.

Para a equipe do Planetário da UFSM, de acordo com a avaliação realizada, o sistema atendeu todas as características de usabilidade. Em relação ao envolvimento e comprometimento com a reunião de iteração, a equipe do Planetário majoritariamente respondeu que não concorda nem discorda ou concorda parcialmente com a afirmação de que a mesma se envolveu e comprometeu com a reunião de iteração. A equipe também avaliou que o uso de histórias de usuário e seus critérios de aceitação facilitaram a especificação e testagem dos requisitos, que o tempo de entrega de duas semanas foi satisfatório e que a expectativa da mesma em relação ao sistema desenvolvido foi cumprida.

Dentre as principais funcionalidades que o desenvolvimento do SASP-UFSM promoveu, podem ser destacadas a geração de um relatório sobre os agendamentos, o agendamento de sessões tanto presenciais quanto virtuais, o agendamento de sessões em horários diferentes dos pré-definidos pelos monitores (mediante aprovação) e a possibilidade de avaliação pelos usuários de uma sessão já realizada.

Através do resultado da implementação, notou-se que o sistema produzido supriu grande parte das necessidades relacionadas ao agendamento de sessões do Planetário da UFSM. Um dos fatores que corroborou para o atingimento dessas necessidades, foi o fato de que a equipe do Planetário foi capaz de proporcionar um bom *feedback* e visão do que seria prioritário para a próxima iteração.

Com a realização deste trabalho, observa-se uma contribuição de ordem prática, pois foi possível agregar valor aos serviços prestados pelo Planetário da UFSM no que diz respeito ao processo de agendamento de sessões. Do mesmo modo, o trabalho desenvolvido reforça a teoria de que a aplicação de métodos ágeis possibilita o desenvolvimento de *softwares* que se

adaptam rapidamente às mudanças, contribuindo para a área do conhecimento. De outro modo, o desenvolvimento do SASP-UFSM possibilitou o aperfeiçoamento da compreensão do autor sobre estes assuntos, tanto no âmbito teórico, quanto no âmbito prático.

Como limitação deste trabalho se apresenta a de que nem sempre toda a equipe do Planetário da UFSM estava presente em todas as reuniões de iteração. Isto se deveu principalmente ao fato de que a realização de duas reuniões, da terceira e quarta iterações, ocorreram em período de recesso letivo, conforme o calendário acadêmico da UFSM (2021). Portanto, não estando presente toda a equipe do Planetário, o *feedback* e a visão sobre as necessidades prioritárias no momento da iteração podem não ter sido as ideais. Contudo, considera-se que esta limitação não foi capaz de prejudicar a validade do desenvolvimento do sistema em si.

O desenvolvimento do SASP-UFSM possibilitou avaliá-lo em relação à usabilidade, à satisfação dos *stakeholders* em relação ao processo de desenvolvimento e à expectativa dos mesmos em relação ao sistema desenvolvido. Entretanto, há oito características no modelo de qualidade de *software* proposto pela ISO 25010. Portanto, recomenda-se como futuros estudos, a avaliação do sistema em relação às outras características, como adequação funcional, eficiência de desempenho, compatibilidade, confiabilidade, segurança, manutenibilidade e portabilidade.



## REFERÊNCIAS

- ANDERSON, D.; CARMICHAEL, A. **Essential Kanban Condensed**. [S.l.]: Lean-Kanban University, 2015. ([Essential Kanban]).
- BECK, K. **Test-driven Development: by example**. [S.l.]: Addison-Wesley, 2002. (Addison-Wesley signature series).
- BECK, K.; ANDRES, C. **Extreme Programming Explained: embrace change** (2nd edition). [S.l.]: Addison-Wesley Professional, 2004.
- BUCHAN, J.; LI, L.; MACDONELL, S. G. Causal Factors, Benefits and Challenges of Test-Driven Development: practitioner perceptions. **2011 18th Asia-Pacific Software Engineering Conference**, [S.l.], Dez 2011.
- COHN, M. **User Stories Applied: for agile software development**. [S.l.]: Pearson Education, 2004. (Addison-Wesley Signature Series (Beck)).
- FOWLER, M.; HIGHSMITH, J. et al. The agile manifesto. **Software development**, [S.l.], v.9, n.8, p.28–35, 2001.
- GARRETT, F. **O que é Microsoft Azure? Veja como funciona e preços do serviço de nuvem**. Brasil, 2020. Disponível em:<<https://www.techtudo.com.br/listas/2020/07/o-que-e-microsoft-azure-veja-como-funciona-e-precos-do-servico-de-nuvem.ghml>>. Acesso em 12 nov. 2021.
- LONGEN, A. S. **O Que é React e Como Funciona?** Brasil, 2021. Disponível em:<<https://www.hostinger.com.br/tutoriais/o-que-e-react-javascript>>. Acesso em 12 nov. 2021.
- MOURA, F. **Manual da marca: sua importância e o que ele deve conter**. Brasil, 2020. Disponível em:<<https://www.treinaweb.com.br/blog/manual-da-marca-sua-importancia-e-o-que-ele-deve-conter>>. Acesso em 8 nov. 2021.
- POSTGRESQL. **About PostgreSQL**. Brasil, 1996-2022. Disponível em:<<https://www.postgresql.org/about/>>. Acesso em 12 nov. 2021.

PRESSMAN, R. **Software Engineering**: a practitioner's approach. 7.ed. USA: McGraw-Hill, Inc., 2009.

ROVEDA, U. **React**: o que é, como funciona e porque usar e como aprender. Brasil, 2019. Disponível em:<<https://devaprender.com/como-hospedar-um-site-gratuitamente-com-netlify/>>. Acesso em 12 nov. 2021.

SCHWABER, K. **Agile Project Management with Scrum**. [S.l.]: Microsoft Press, 2004. (Best practices).

SOMMERVILLE, I. **Software Engineering**. 9.ed. USA: Addison-Wesley Publishing Company, 2010.

SOUZA, J. de. **Como Hospedar um Site Gratuitamente em 2020 com Netlify**. Brasil, 2021. Disponível em:<<https://devaprender.com/como-hospedar-um-site-gratuitamente-com-netlify/>>. Acesso em 10 nov. 2021.

SYOZI, R. **O que é a AWS? [Amazon Web Services]**. Brasil, 2021. Disponível em:<<https://tecnoblog.net/489091/o-que-e-a-aws-amazon-web-services/>>. Acesso em 10 nov. 2021.

TUTORIALSTEACHER. **What is Web API?** Brasil, 2022. Disponível em:<<https://www.tutorialsteacher.com/webapi/what-is-web-api/>>, Acesso em 12 nov. 2021.

UNIVERSIDADE FEDERAL DE SANTA MARIA. **Planetário**. Santa Maria, 2021. Disponível em:<<https://www.ufsm.br/orgaos-suplementares/planetario/>>, Acesso em 5 jul. 2021.



## APÊNDICE A - HISTÓRIAS DE USUÁRIO

Tabela 6 – Histórias de Usuários de Todas as Iterações

Nº	História de Usuário	Critérios de aceitação
1	Como professor/coordenador pedagógico, quero solicitar uma sessão, para obter um agendamento e desfrutar de uma sessão remota.	(i) A solicitação deve ser feita por um formulário; (ii) a escolha da data e hora de uma sessão deve ser um calendário, o qual: (a) os dias que já tiverem todos os horários com agendamento são desabilitados e marcados com vermelho; (b) os dias de final de semana são desabilitados e marcados com cinza; e (c) quando selecionado um dia, deve-se exibir apenas os horários disponíveis. (iii) exibir as sessões e tópicos abordados; e (iv) enviar um e-mail para quem agendou e um para o Planetário contendo as informações do agendamento.
2	Como professor/coordenador pedagógico, quero poder cancelar meu agendamento, para que caso ocorra algum imprevisto eu não precise enviar um e-mail ao Planetário.	(i) Deve-se inserir o identificador do agendamento para cancelar.
3	Eu como monitor do Planetário, quero efetuar login no sistema, para poder gerenciar recursos privados	(i) Deve ser feita com o e-mail e senha definidos pelos monitores do Planetário; e (ii) após o login, devo ser redirecionado à área do monitor.
4	Eu como monitor do Planetário, quero efetuar logout no sistema, para caso eu queira não deixar meu navegador logado no sistema	(i) Deve existir um botão de logout na área do monitor.
5	Eu como professor/coordenador pedagógico quero visualizar as informações de cada sessão no momento do preenchimento do formulário para que eu não precise lembrar das mesmas no momento do preenchimento do formulário	(i) Deve-se mostrar título, descrição, tópicos abordados e recomendação; (ii) cada tipo de sessão deve ser um radio button; e (iii) quando clicar em um tipo, deve-se mostrar as informações da mesma.

---

6	Eu como monitor do Planetário, quero visualizar uma tabela de agendamentos para obter conhecimento dos mesmos	(i) Deve-se mostrar a tabela na área do monitor; (ii) a tabela deve mostrar as seguintes informações: (a) data e hora de sessão; (b) sessão escolhida; (c) instituição; (d) série/ano; (e) número de participantes; e (f) professor; (iii) a tabela deve ser ordenada por data e hora de sessão em ordem decrescente, ou seja, a data mais tardia deve ser mostrada por primeiro; e (iv) a tabela apenas deve mostrar os agendamentos não excluídos.
7	Eu como monitor do Planetário quero poder visualizar, editar e excluir um agendamento para que eu possa manter um agendamento atualizado	(i) As ações de visualizar, editar e excluir devem estar localizadas na tabela de agendamentos da área do monitor, em uma coluna de ações; (ii) as ações de visualizar e editar um agendamento devem abrir um modal com as todas as informações do agendamento a serem visualizadas ou editadas; e (iii) a ação de excluir um agendamento deve exibir um alerta de confirmação da ação.
8	Eu como monitor do Planetário, quero poder manter (criar, visualizar, editar e excluir) as sessões para que haja agendamentos para novos tipos de sessões e para que as sessões já existentes fiquem atualizadas.	(i) As ações de visualizar, editar e excluir devem estar localizadas na tabela de sessões da área do monitor, em uma coluna de ações; (ii) a ação de criar uma sessão deve estar localizada ao final dos registros mostrados na tabela; (iii) as ações de visualizar e editar uma sessão devem abrir um modal com as todas as informações da sessão a serem visualizadas ou editadas; e (iv) a ação de excluir uma sessão deve exibir um alerta de confirmação da ação.

---

9	Eu como monitor do Planetário, quero poder manter (criar, visualizar, editar e excluir) os horários disponíveis para que haja novos agendamentos apenas para os horários especificados.	(i) As ações de visualizar, editar e excluir devem estar localizadas na tabela de horários disponíveis da área do monitor, em uma coluna de ações; (ii) a ação de criar um horário disponível deve estar localizada ao final dos registros mostrados na tabela; (iii) as ações de visualizar e editar um horário disponível devem abrir um modal com as todas as informações do horário a serem visualizadas ou editadas; e (iv) a ação de excluir um horário deve exibir um alerta de confirmação da ação.
10	Eu como professor/coordenador pedagógico quero poder solicitar tanto uma sessão virtual quanto uma sessão presencial para que dependendo da minha escolha, seja realizado um agendamento.	(i) A escolha deve ser feita na página inicial do sistema; (ii) deve haver uma página para gerenciamento de agendamentos, sessões e horários de sessões virtuais e outra para gerenciamento de agendamentos, sessões e horários de sessões presenciais, as duas devem poder ser acessadas pela área do monitor.
11	Eu como professor/coordenador pedagógico quero poder solicitar uma sessão em um horário diferente para que eu desfrute de uma sessão em um momento não pré-definido pelo monitores.	(i) A opção de solicitar um horário diferente deve estar presente no formulário de solicitação; (ii) o agendamento não deve constar na tabela de agendamentos da área do monitor; e (iii) deve-se avisar que o agendamento está pendente no e-mail enviado após a solicitação.
12	Eu como monitor do Planetário quero poder aceitar ou recusar uma solicitação de sessão em um horário diferente para que eu possa proporcionar uma sessão em um horário não pré-definido.	(i) Os agendamentos pendentes de aprovação ou recusa devem estar na página principal da área do monitor; e (ii) um e-mail deve ser enviado ao solicitante assim que a ação de aprovação e recusa for efetuada avisando sobre a situação;
13	Eu como professor/coordenador pedagógico quero receber o link da sessão virtual para que possa tê-lo desde o primeiro momento e compartilhá-lo.	(i) O link da sessão virtual deve ser uma reunião <i>online</i> pela plataforma Google Meet; e (ii) o link deve ser recebido no e-mail de confirmação de agendamento.

---

14	Como monitor do Planetário, quero manter (criar, visualizar, editar e excluir) horários indisponíveis, para que não haja novos agendamentos nos horários especificados.	(i) O componente calendário do formulário de solicitação de agendamento não deve permitir selecionar algum horário indisponível; (ii) o sistema não deve permitir o agendamento em um horário indisponível com exceção de agendamentos que solicitem uma sessão em horário diferente; (iii) as ações de visualizar, editar e excluir devem estar localizadas na tabela de horários indisponíveis da área do monitor, em uma coluna de ações; (iv) a ação de criar um horário indisponível deve estar localizada ao final dos registros mostrados na tabela; (v) as ações de visualizar e editar um horário indisponível devem abrir um modal com as todas as informações do horário a serem visualizadas ou editadas; e (vi) a ação de excluir um horário deve exibir um alerta de confirmação da ação.
15	Como monitor do Planetário, quero poder gerar um relatório para que eu possa ter mais informações/estatísticas sobre os agendamentos de um determinado período.	(i) Deve ser necessário selecionar um período de tempo; (ii) o relatório deve conter as seguintes informações: (a) quantos alunos foram atendidos; (b) quantos agendamentos foram realizados para cada sessão; (c) quantos agendamentos foram realizados para tipo de escola; e (d) quantos agendamentos foram realizados para cada série/ano; (iii) o relatório gerado deve estar no formato .CSV.
16	Como professor/coordenador pedagógico, quero poder avaliar a sessão após seu término, para que eu possa dar um <i>feedback</i> sobre como foi a sessão.	(i) Deve-se inserir o identificador do agendamento; (ii) deve-se registrar a data e hora da avaliação; (iii) avaliação utilizando uma nota em escala de 1 à 5 e crítica em formato de texto; e (iv) deve ser possível avaliar apenas após o horário do agendamento.

---

---

17	Como monitor do planetário, quero poder visualizar as avaliações, para que eu possa obter <i>feedback</i> sobre as sessões realizadas.	(i) As avaliações deve estar localizadas na tabela de avaliações na área do monitor; (ii) a tabela de avaliações deve conter as informações de data e hora da sessão, sessão escolhida e se a sessão foi virtual ou presencial, além das informações da avaliação; (iii) a ação de visualizar a avaliação deve estar localizada na tabela de avaliações, em uma coluna de ações; e (iv) as avaliações devem estar ordenadas por data e hora de avaliação, as mais recentes primeiro.
----	--	--

---

Fonte: Produção do Autor.