**UNIVERSIDADE PAULISTA – UNIP EaD**

**Projeto Integrado Multidisciplinar**

**Curso Superior de Tecnologia em**

**Análise e Desenvolvimento de Sistemas**

**Luan Da Silva Sabino – 0606739**

**Sistema de Reserva de equipamentos audiovisuais**

**ÁGUAS DE LINDÓIA**

**2023**

**Luan Da Silva Sabino – 0606739**

Sistema de Reserva de equipamentos audiovisuais

Projeto Integrado Multidisciplinar em

Análise e Desenvolvimento de Projetos

Projeto Integrado Multidisciplinar

para a obtenção do título de graduação

em Curso Superior Tecnologia em

Análise e Desenvolvimento de

Sistemas, apresentado à Universidade

Paulista – UNIP EaD.

ÁGUAS DE LINDÓIA

**2023**

**RESUMO**

Este projeto acadêmico tem a finalidade de desenvolver um software que administra reservas de equipamento audiovisuais para o “Colégio Vencer Sempre” e colégios de ensino fundamental e médio. Para a construção desse sistema colocaremos em pratica os conhecimentos obtidos em Engenharia de Software II, Projeto de Interface com o Usuário e Programação Orientada a Objetos, com isso temos as especificações de interfaces, prototipação de interface, análise de requisitos funcionais e não funcionais e roteiro de teste, também iremos aplicar o conhecimento obtido em economia e mercado com a viabilidade econômica do projeto e mapeamento de agentes econômicos.

Palavras chaves: Engenharia de Software, Economia, Desenvolvimento, Python, UI/UX, Interação Humano Computador

**ABSTRACT**

This academic project has the purpose of developing a software that manages reservations of audiovisual equipment for the "College to win always" and primary and secondary schools. For the construction of this system we will put into practice the knowledge obtained in software engineering || , user interface design and object-oriented programming |, with that we have the specifications of interfaces, interface prototyping, analysis of functional and non-functional requirements and test script, and we will also apply the knowledge obtained in economics and market with the economic viability of the project and mapping of economic agents.

Keywords: Software Engineering, Economics, Development, Python, UI/UX, Human Computer Interaction

**SUMÁRIO**

1.Introdução ................................................................................7

2.Documento e aplicação ...........................................................8

2.1 Cronograma .......................................................................8

3. Programação Orientada a Objetos .........................................9

3.1 Objetos .........................................................................................9

3.2 Classes .........................................................................................9

3.3 Encapsulamento ...........................................................................10

3.4 Herança ........................................................................................10

3.5 Polimorfismo .................................................................................90

4. Desenvolvimento do Soft.............................................11

4.1 Engenharia de Software II ..........................................................11

4.2 Requisitos Funcionais .................................................................11

4.3 Requisitos Não Funcionais e Regras de Negócios .....................12

4.4 Especificações do Sistema ..........................................................12

4.5 Metodologias ...............................................................................13

4.5.1 ISO – SPICE ............................................................................13

4.5.2 CMMI ........................................................................................14

4.5.3 MPS.BR ....................................................................................15

4.6 PROTOTIPAÇÃO DE TESTE.............................................16

4.6.1 Login .........................................................................................16

4.6.2 Cadastro ...................................................................................17

4.6.3 Menu Principal ..........................................................................18

4.6.4 Reserva ....................................................................................18

4.6.5 Registro de Reservas ..............................................................20

4.7 Roteiro de Teste .................................................................20

4.7.1 Login ........................................................................................21

4.7.2 Cadastro ..................................................................................21

4.7.3 Menu Principal .........................................................................22

4.7.4 Reserva ...................................................................................22

4.7.5 Registro de Reservas ..............................................................22

5. Investimento Financeiro ...........................................................23

6. Conclusão ................................................................................24

7. Referências ..............................................................................25

**7**

**1.Introdução**

Com as evoluções constantes e a informação e conhecimento tem sido a principal alavanca do desenvolvimento econômico global, garantindo uma qualidade melhor de vida para todos.

Com a evolução, o avanço de novas tecnologias, se torna uma necessidade de tal forma que é necessário pensar e novas formas de comunicação, aonde temos o início da Tecnologia da Informação, realizando novas pesquisas, desenvolvendo e inovando com hardwares novos e software para suprir as demandas que surgirem.

Com o avanço, os sistemas tornaram-se mais complexos de tal maneira que se precisa de evolução diária, tendo a necessidade de profissionais cada vez mais preparados e qualificados para resolver as demandas que for surgindo diariamente.

O Colégio Vencer Sempre, viu a necessidade de implantar um sistema mais avançado para controle de seus equipamentos de audiovisuais, obtendo maior valor em sua instituição de ensino e maior facilidade para professores e auxiliares.

8

**2. Documento e aplicação**

Pensando nas informações obtidas pelo cliente para o projeto, será desenvolvido na plataforma Microsoft visual Studio 2022, onde faremos todo o processo do sistema gerando maior confiabilidade.

O sistema terá total integração garantido que o usuário ao fazer um registro de um equipamento, ele será exibido em todas as telas dos equipamentos com sua descrição completa quando requisitado.

**2.1 Cronograma**

Um cronograma de desenvolvimento do sistema terá um prazo de 65 dias para execução dos sistemas para a entrega final.

Planejamento: 14 dias (112 horas).

Desenvolvimento: 35 dias (280 horas).

Teste: 16 dias (128 horas).

Tempo estimado de conclusão: 65 dias (520 horas).

Na fase de planejamento junto com o cliente, teremos o levantamento de regra de negócios, design e requisitos funcionais e não funcionais.

A próxima fase iremos dar início ao desenvolvimento começando pelo back-end, planejando como funcionará cada parte, criando suas classes, objetos e herança.

A última fase temos o teste, onde iniciaremos diversos testes no sistema para garantir que não haverá erros no programa aos ser executado, confirmando a utilização do sistema no computador do cliente

9

**3. Programação Orientada a Objetos**

A programação orientada objetos tem o propósito em desenvolver um software que cumpra todos os requisitos solicitado, garantindo a entrega que irá satisfazer o cliente.

A orientação objeto é utilizada na maioria das linguagens de programação, embora que alguns aspectos se diferenciam, eles seguem os mesmos princípios, porém pode utilizar para resolver problemas na realidade e acabar sendo uma base para outros problemas que não está ligado diretamente na linguagem de programação, mas podemos aprender pois não há mudança.

Será identificado nesse projeto, os conceitos de objetos, classes, heranças e polimorfismo, onde iremos identificar todos estes conceitos em nosso sistema.

**3.1 Objetos**

Na programação, um objeto é igualado a uma variável, em um sistema com orientação a objetos, identificamos o total de objetos que podem mostrar a conclusão para problemas, temos atributos que reproduzem suas propriedades e suas características, que podemos usar para alterar seus dados.

Um objeto possui um espaço na memória que armazena seu valor, sendo capaz de ser manipulado por comandos ou outras operações.

**3.2 Classes**

Uma classe é uma propriedade que define a forma para os objetos, por meio de uma definição de uma classe, é capaz de criarmos atributos e métodos variados.

No sistemas temos duas classes, primeira classe "entrar" e segundo classe "cadastro", que possui atributos para preenchimento de dados obrigatórios como, "E-mail Institucional" e "Senha" e possui métodos "entrar" e "cadastrar".

10

**3.3 Encapsulamento**

Encapsulamento é um processo que impede o vazamento de dados dos atributos e métodos que não devem ser vistos por outros, somente prover uma interface bem definida, temos como exemplo o controle remoto, quando o controle remoto estraga, apenas é trocado ou consertado o controle e não a televisão inteira.

**3.4 Herança**

Um dos principais conceitos, se não o mais importante, é o conceito de "Herança", onde permite a reutilização do código que garante a confiabilidade do sistemas. Para manter a qualidade dos sistemas, é preciso fazer apenas uma vez o cadastro em código, quando for necessário usar as informações do cadastro em outra parte, ele busca dentro do código e reutiliza.

No paradigma estruturado, cada informação é necessário ser feito manualmente, como exemplo um cadastro de veículo, se for preciso obter as informações novamente, podemos reutilizar o código, copiando e colando em outra área, assim o programador não precisa refazer todo o código.

**3.5 Polimorfismo**

O polimorfismo é um mecanismo por meio do qual selecionamos a funcionalidades utilizadas, e a forma de como se chega a ação ou método. Com o polimorfismo os mesmos atributos e objetos podem ser utilizados em objetos distintos, porém, com implementações lógicas diferentes, temos como exemplo o carro, a três carros, o carro "A" reproduz música pelo bluetooth, USB e rádio, o carro "B" reproduz música pelo cabo auxiliar e rádio, e o cabo o carro "C" reproduz música somente pelo CD e rádio, os três veículos possuem o método de reproduzir música, mas a forma de cada um muda.

11

**4. Desenvolvimento do Software**

**4.1 Engenharia de Software II**

Com o avanço da tecnologia nos últimos anos, vem surgindo melhorias para o desenvolvimento de software e passando por inovações constantes, hoje as primeiras metodologias não conseguem acompanhar seus requisitos, pois os sistemas da segunda década do século XXI vem se tornando mais complexo, sendo necessário ter uma inovação constante de acordo com o projeto e sistema desenvolvido.

Criar um software sem possuir as metodologias adequadas, pode parecer ser melhor e relativamente mais fácil, porém ao decorrer do tempo seu sistema fica cada vez mais complexo criando a necessidade de contratar empresas especializadas para reconstruir o software utilizando as metodologias adequadas no processo de desenvolvimento.

A necessidade dos sistemas serem construídos e distribuídos de formas mais eficientes, possuímos novas técnicas que ajudam a construir sistemas maiores e mais complexos, permitindo novas capacidades que antes eram consideradas impossíveis.

**4.2 Requisitos Funcionais**

Controle dos equipamentos: O sistema irá controlar as reservas de equipamentos audiovisuais do Colégio Vencer Sempre, controlando registro em log do responsável, equipamentos, data e hora, bem como confirmações, cancelamentos e devolução.

Inventario: Todo equipamento possuirá cadastro no software, sendo que cada reserva, sendo confirmada ou cancelada ficará salva em relatório individual de cada equipamento. Os equipamentos podem ser rastreados através de um QRCode fixado em cada um possuindo identificação única.

12

**4.3 Requisitos Não Funcionais e Regras de Negócios**

Permissão de acesso: O sistema terá dois níveis de permissões acesso administrativos, os diretores e os coordenadores, que poderão fazer alteração de status disponível, adicionar equipamentos novos e manutenção. Os professores, assistentes e estagiários, não serão possíveis realizar nenhuma alteração nos equipamentos, será somente permitido o acesso para reservas ou cancelamento de reserva.

Todos os dados serão administrados somente pelo usuário que possui a permissão de administrador segundo a lei geral de proteção de dados (Lei n° 13.719/2018).

Configuração do Sistema: o sistema será desenvolvido apenas para Windows desktop. Para os relatórios devem seguir um padrão de espaçamento e fonte conforme a ABNT. O sistema será desenvolvido em linguagem de programação C#, utilizando a plataforma visual Studio 2022.

**4.4 Especificações do Sistema**

O sistema será possível acessá-lo somente pelos computadores do diretor, coordenador, sala dos professores e almoxarifado, onde será implantado o sistema no servidor existente no colégio.

Nas telas dos equipamentos será fácil a visualização e interpretação, com botões grandes e de fácil usabilidade e entendimento.

13

**4.5 Metodologias**

**4.5.1 ISO – SPICE**

O início da norma 15504 remonta ao ano de 1991, quando o JTCI iniciou um estudo sobre a necessidade de uma Norma para avaliação de processo de software, em 1998 foi publicada a ISO/IEC 15504, é o resultado dos estudos e combinação do modelo de qualidade de processo de software CMM, ISO/IEC 12207 (Processos do ciclo de vida), ISO 9001, ISO 9000-3 entre outras.

A SPICE (Software Process Improvement and Capability Determination), sendo a maturidade das empresas na construção de software, descrevendo o conjunto de processos que agregam a Engenharia de Software e classificam as empresas em seis níveis de maturidade que determinam o grau de qualidade do desenvolvimento de software.

|  |  |
| --- | --- |
| Nível de Capacidade | Características Básicas |
| 0 – Incompleto | Processo Inexistente ou geralmente falho. |
| 1 – Executado | Atinge os objetivos, mas sem controle de escopo, prazo e custos e sem padrões de qualidade. |
| 2 – Gerenciado | Atinge os objetivos de prazo, custo e qualidade, e os produtos são gerenciados. |
| 3 – Estabelecido | Processo estabelecido, executado e gerenciado mediante adaptação ao padrão definido. |
| 4 – Previsível | Processo estabelecido e totalmente controlado por medições especificas. |
| 5 - Otimizado | Melhoria de forma contínua e disciplinada. |

Tabela 01 – Níveis de Maturidade ISO

Fonte: ISO (2003c).

É considerado um dos processos de implantação de maior valor, para empresas de pequeno/médio porte, para fazer sua implantação é quase impossível, levando até 10 anos para chegar-se ao nível 5.

14

**4.5.2 CMMI**

Capability Maturity Model Integration, ou CMMI, possui também 5 níveis de desenvolvimento, de acordo com a maturidade da empresa, é um conjunto de boas práticas para o processo de desenvolvimentos de software. Tem uma visão macro do desenvolvimento e suas regras podem ser aplicadas em diversos tipos de metodologias de desenvolvimento.

|  |  |
| --- | --- |
| Nível de Capacidade | Características Básicas. |
| 1 – Realizado. | Todas as metas especificas da área de processo foram satisfeitas. |
| 2 – Gerido. | Práticas de gerenciamento mais sofisticadas, tais como: Gestão de risco, orçamentos, cronogramas etc. |
| 3 – Definido. | Todos os requisitos do nível 2 foram atingidos e a empresa está alinhado com as ferramentas e comprometida com o modelo. Todo o processo é mapeado e documentado. |
| 4 - Quantitativamente Gerido. | Área de processo controlada e aperfeiçoada usando medições e avaliação quantitativa. Indicadores de avaliação de desempenho dos projetos são criados. |
| 5 – Otimizado. | Processo em melhoria contínua, a maturidade está em nível que não se é mais necessário criar subprocessos, apenas refinar os existentes através de modelos estatísticos, garantindo maior qualidade. |

Tabela 02 – Níveis de Maturidade CMMI

Fonte: UNIP – Engenharia de Software II, 2023.

15

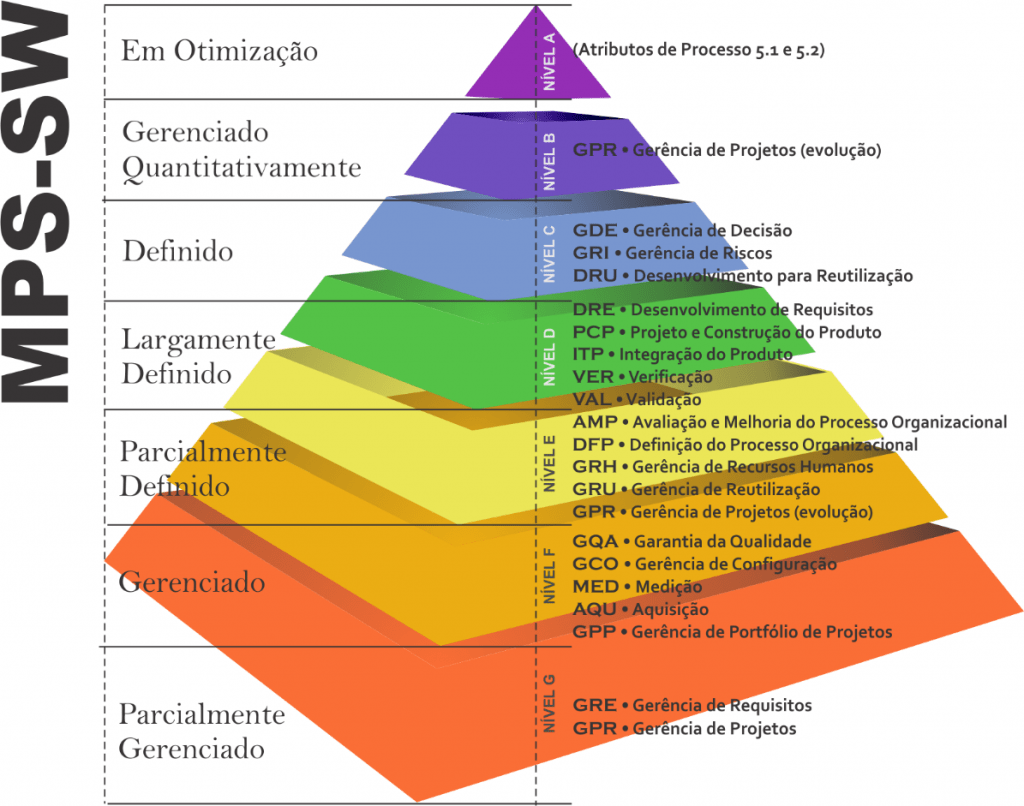
**4.5.3 MPS.BR**

A metodologia MPS.BR foi criada em 2003, pela coordenação da SOFTEX - Associação da Promoção da Excelência do Software Brasileiro, desenvolvido com base nas normas ISO 15504 e CMMI visando ter-se mais qualidade no processo de desenvolvimento de software no Brasil, a MPS.BR é um programa mobilizador de longo prazo para a Melhoria de Processo de Software e Serviços, com metas e níveis definidos para longo prazo.

Os níveis de maturidade do MPS.BR iniciam-se em Nível G progredindo até o Nível A, dividido em 7 níveis com o objetivo de implementação de micros, pequenas e médias empresas.

Por sermos uma microempresa, estamos no Nível G de maturidade, onde nossos processos são Parcialmente Gerenciador por Requisitos (GRE) e Projetos (GPR).

Para prover informações sobre o andamento dele permitam a realização de correções quando houver desvios significativos, e estabelecemos planos que definem atividades, recurso e responsabilidade do projeto.



Fonte: <https://promovesolucoes.com/quais-sao-os-niveis-de-maturidade-do-mps-br/>

16

**4.6 PROTOTIPAÇÃO E TESTES**

Para garantir a validação e verificação do software, será utilizada a metodologia Desenvolvimento Orientado/Guiado por Testes ou TDD (Test Driven Development), que tem como objetivo orientar cada etapa de testes antes da codificação do sistema.

A seguir temos o passo a passo do sistema, que foi separado por telas, onde são especificados para sanar as dúvidas.

**4.6.1 Login**

Campos de login do usuário:

E-mail Institucional => e-mail institucional do usuário, admite letras, números e caracteres especiais, preenchimento obrigatório para acesso.

Senha => senha para acesso ao sistema, admite letras, números, caracteres especiais e com mínimo 8 caracteres, preenchimento obrigatório para acesso.

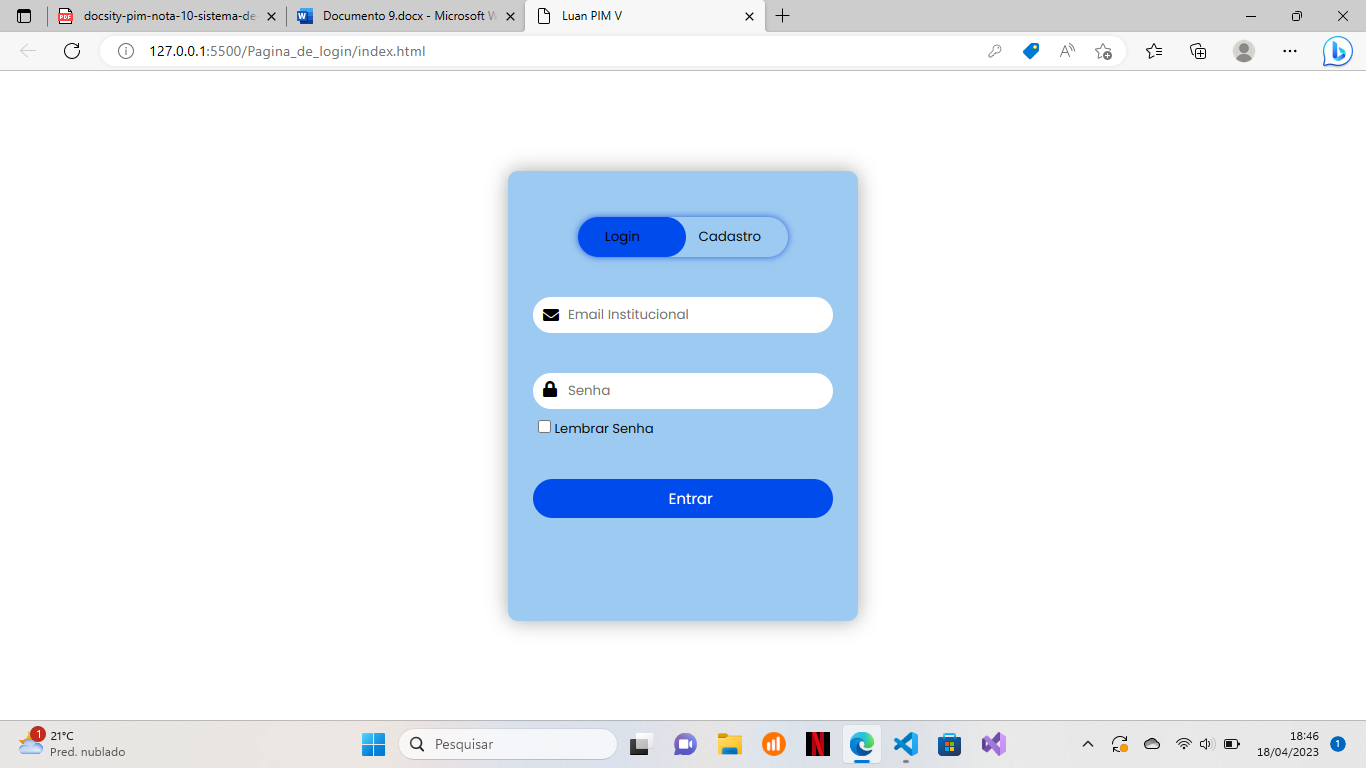


Figura – Tela de Login de Usuários.

Fonte: Autoria Própria.

17

Se usuário não tiver o login de usuário para o acesso no sistema, deverá clicar em “Cadastro” onde será levado para próxima sessão.

**4.6.2 Cadastro**

Campos de cadastro do usuário:

E-mail Institucional => E-mail do ator, admite letras, números e caracteres especiais, preenchimento obrigatório para acesso.

Senha => Senha para acesso ao sistema, admite letras, números, caracteres especiais e com mínimo 8 caracteres, preenchimento obrigatório para acesso.

Confirmar a senha => Repetir senha.

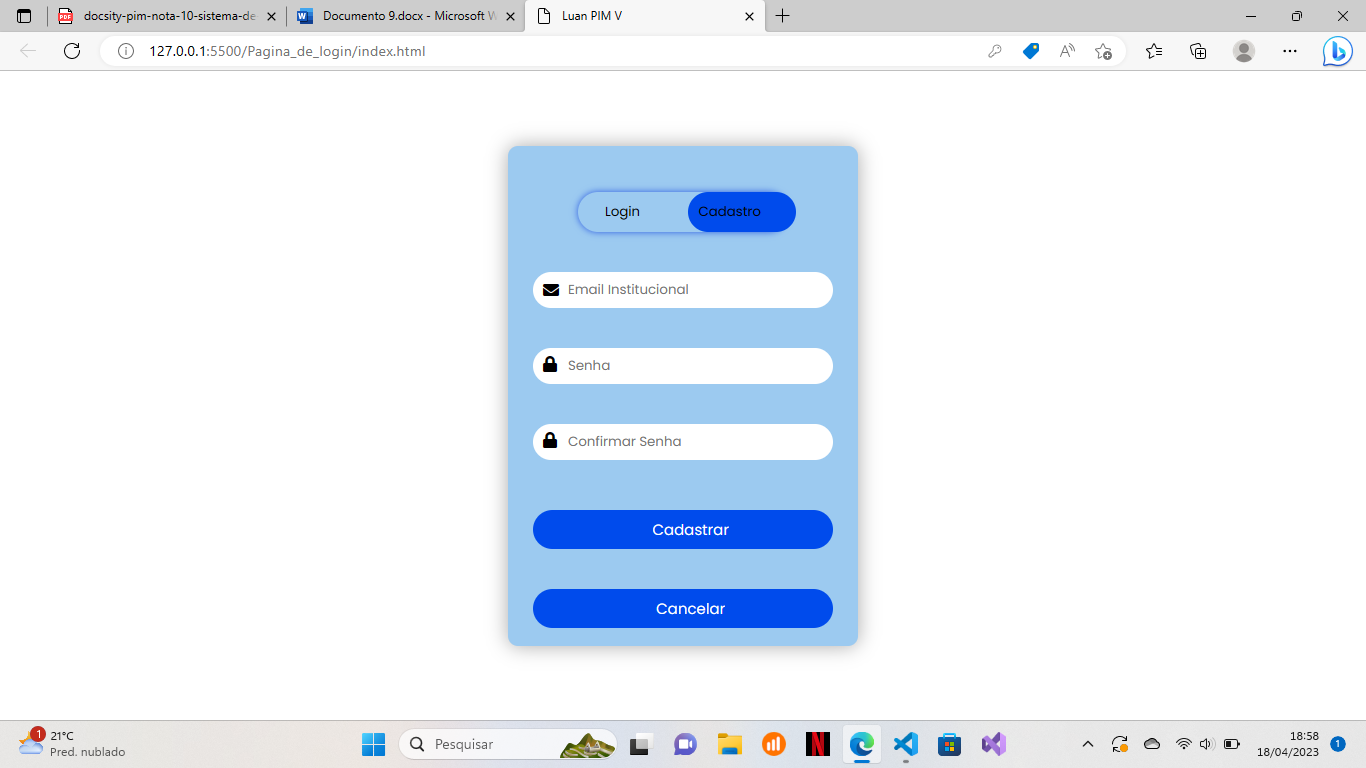


Figura – Tela de Cadastro de Usuários.

Fonte: Autoria Própria.

Após finalizar preenchimento, o usuário poderá confirmar e retorna a tela de login com dados gravados, ou voltar para tela de Login sem finalizar o cadastro.

18

**4.6.3 Menu Principal**

Menu principal apresenta ao usuário, qual tipo de ação deseja fazer ou se deseja sair do sistema, voltando para o menu principal.

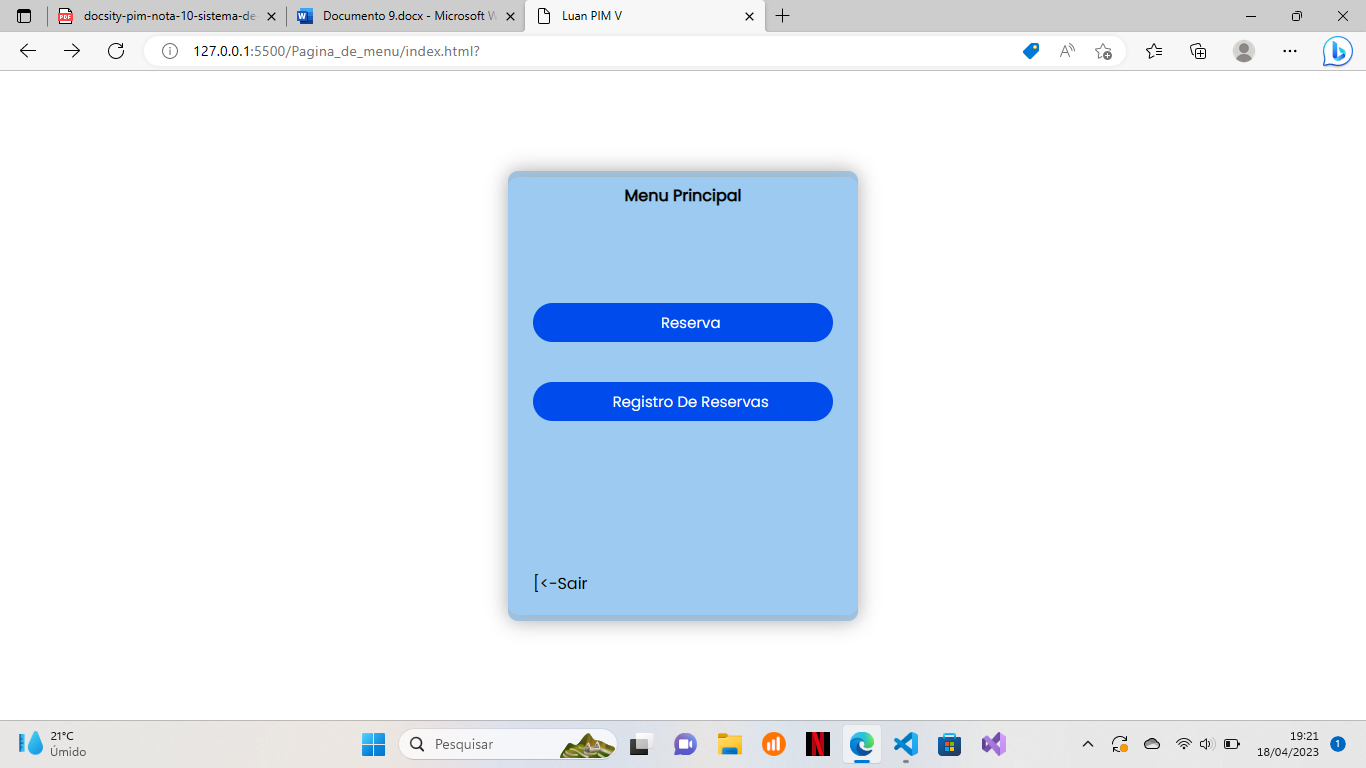


Figura – Tela de Menu Principal.

Fonte: Autoria Própria.

**4.6.4 Reserva**

Campos de cadastro de reservas:

RESPONSÁVEL PELO PEDIDO => Nome do usuário responsável pela reserva, admite apenas letras.

EQUIPAMENTOS DISPONÍVEIS => nome do equipamento que será reservado, campo obrigatório, campo seleção dos equipamentos cadastrados.

DATA DA RESERVA => data da reserva, admite datas contidas no calendário, campo obrigatório.

19

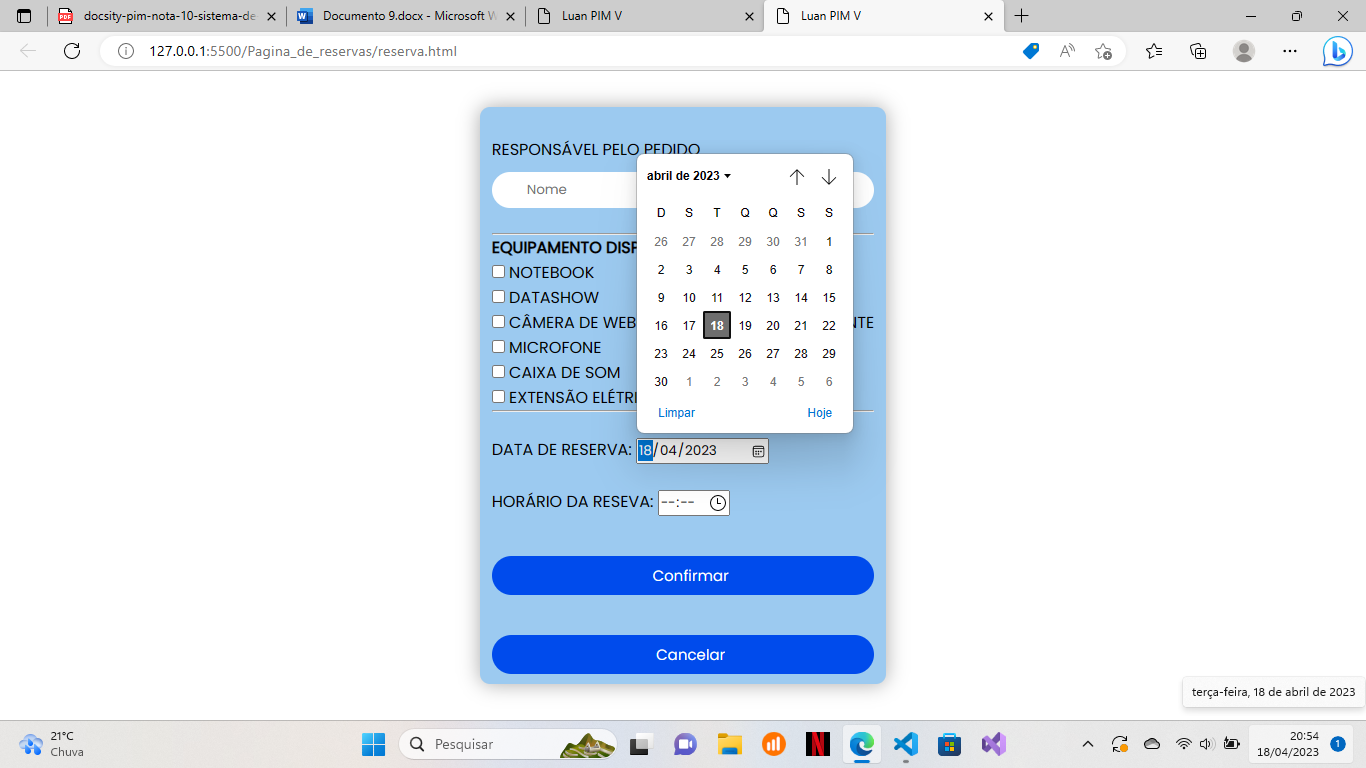
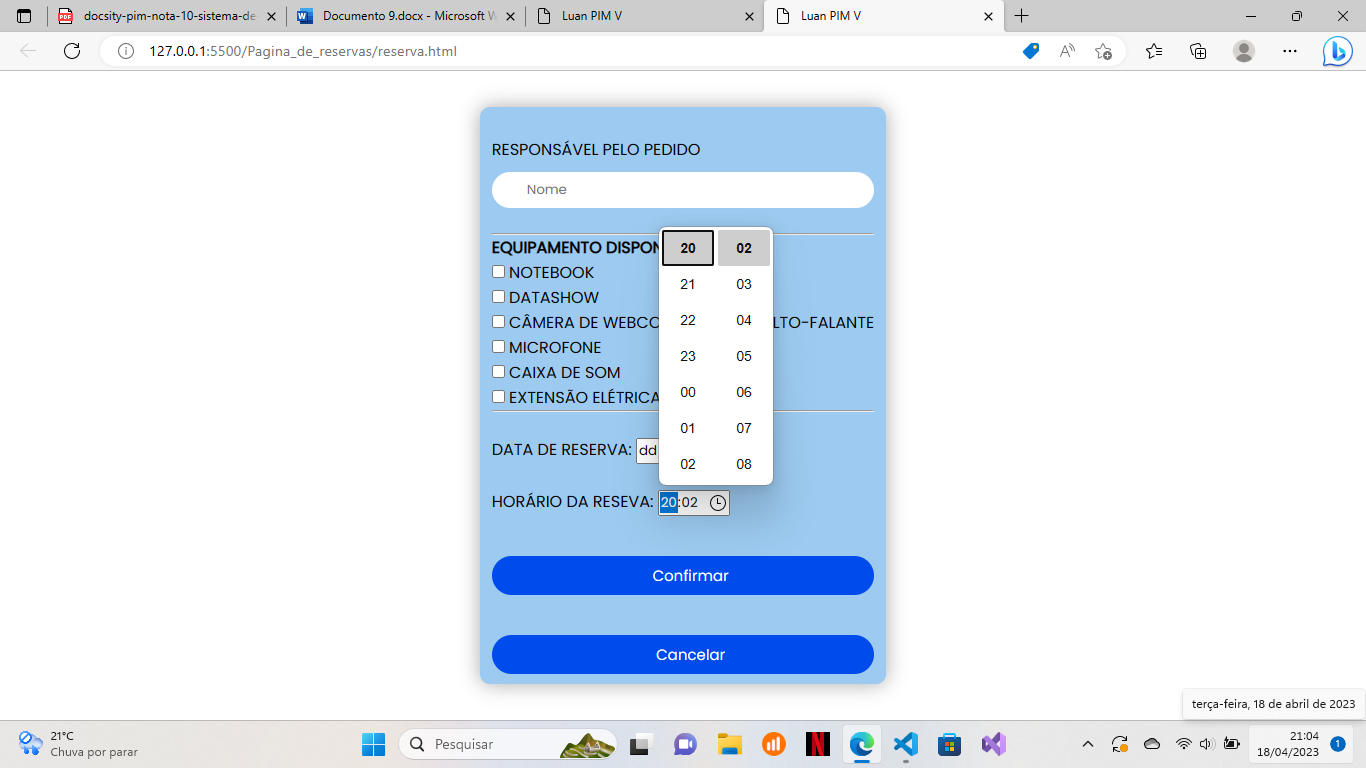


Figura – Tela de Reserva, Calendário.

Fonte: Autoria Própria.

HORA DA RESERVA => hora de reserva, admite horas, campo obrigatório.



21

Figura – Tela de Reserva, Horário.

Fonte: Autoria Própria.

**4.6.5 Registro de Reservas**

Registro de reservas apresenta ao usuário os registros de dados, recebidos do cadastro de reservas, aqui pode editar os dados, ou excluir a reserva.

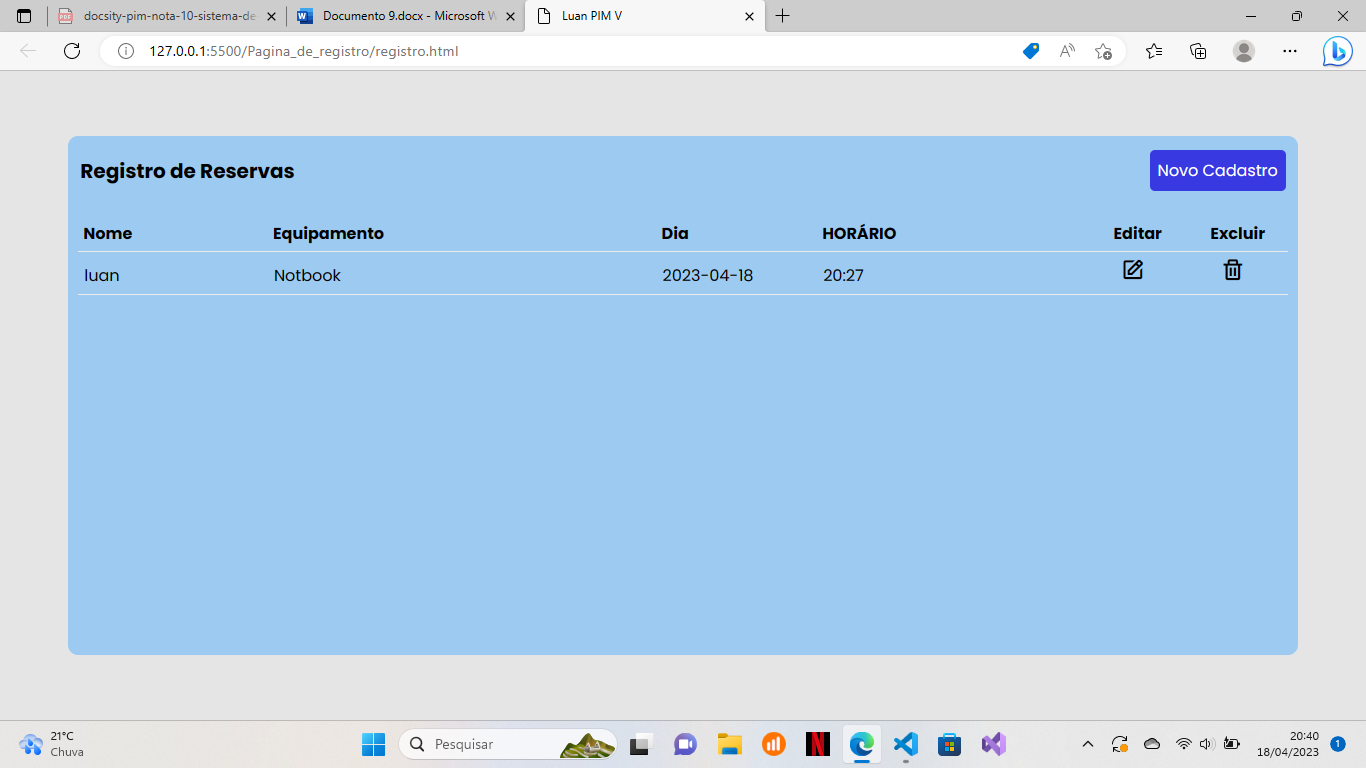


Figura – Tela de Registro de Reservas.

Fonte: Autoria Própria.

**4.7 Roteiro de Teste**

Cada etapa foi detalhada através de um roteiro já solicitado através do enredo do projeto, a seguir será mostrado os casos para cada etapa do software:

CADASTRO DE USUÁRIO: Preenchimentos de dados > Dados corretos > Dados incorretos;

LOGIN: Preenchimentos de dados > Dados corretos > Dados incorretos;

22

MENU PRINCIPAL: Reservas > Registro de Reservas > Sair;

RESERVA: Preencher dados Responsável Pelo Pedido, preencher dados de Equipamento, preencher dados de Data da Reserva, preencher dados de Hora da Reserva > Data e Hora não disponíveis > Dados confirmado;

REGISTRO DE RESERVAS: Dados gravados> Editar> Excluir> Nova Cadastro;

**4.7.1 Login**

1. Digitar E-mail Institucional > Preenchido, vazio;

2. Digitar Senha > (1) Preenchido, (2) vazio;

3. Entrar;

4. Se o preenchimento não seguir a prototipação será visualizado mensagens de Alerta: “Usuário não identificado, verificar e-mail ou senha.”;

5. Tela de Menu Principal, caso o e-mail e senha estiver correto.

**4.7.2 Cadastro**

1. Digitar nome completo > Preenchido, vazio;

2. Digitar E-mail Institucional > Preenchido, vazio;

3. Digitar Senha > (1) Preenchido, (2) vazio;

4. Confirmar Senha > (1) Preenchido da mesma forma do passo 3, (2) Preenchido não confere com a senha digita anteriormente, (3) vazio;

5. Entrar;

Se o preenchimento não seguir a prototipação será visualizado mensagens de Alerta: “Usuário não identificado, verificar e-mail ou senha.”

23

**4.7.3 Menu Principal**

1. Reserva;

2. Registro de Reservas;

**4.7.4 Reserva**

Para processo de reserva, o usuário precisa fazer seguir cada passo para ter sucesso no preenchimento do formulário.

1. Digitar Nome > Preenchido, vazio;

2. Selecionar Equipamento > (1) Selecionado, (2) não selecionado;

3. Selecionar Data da Reserva > (1) Selecionado, (2) não selecionado;

4. Selecionar Horário da Reserva > (1) Selecionado, (2) não selecionado;

5. Confirmar

Se o preenchimento não seguir a prototipação será visualizado mensagens de Alerta: “Digitar nome, selecionar Equipamento, selecionar data da reserva, selecionar horário da reserva.”

**4.7.5 Registro de Reserva**

1. Dados Gravados > Visualizar;

2. Editar > Selecionar dados gravados para editar;

3. Excluir > Selecionar dados de registro para excluir.

24

**5. Investimento Financeiro**

Um projeto de pequeno porte, com uma equipe devidamente preparada mesmo sendo uma fábrica de software pequena gera custos, valendo-se desde a hora trabalhada por cada membro da equipe como dos custos dos equipamentos e impostos que agregam o valor final.

O investimento feito pelo Colégio Vencer Sempre será de R$ 67,546.30 já com todos os valores de imposto calculados.

Os agentes econômicos que interferem no projeto são: Famílias, pais pagam a mensalidade, sendo o colégio o provedor pelos serviços de ensino.

O governo, como garantidor e administrador dos impostos recolhidos pelas famílias e empresas.

25

**6. Conclusão**

Com o investimento deste software para o Colégio Vencer Sempre, terão maior valor em seus serviços, oferecendo aos professores e auxiliares uma nova plataforma, ajudando a administrar as solicitações dos equipamentos de áudio visual com data, hora, garantindo o controle de uso dos equipamentos do colégio.

Será possível no futuro haver um dimensionamento deste sistema para outras áreas do colégio, desde a gestão de projetos para as turmas e professores até mesmo o demonstrativo de resultados para notas.

24

**7. Referencias**

LOPES, P. H. F. Programação Orientada a Objetos I. 9. ed. [S.l.]: UNIP, 2015.

MANZALLI, P. M. F. Economia e Mercado. 9. ed. [S.l.]: UNIP, 2020.

ISO – SPICE Disponível em:[ISO/IEC 15504 Norma de qualidade de desenvolvimento de software (dqsglobal.com)](https://www.dqsglobal.com/pt-br/certifique/certificacao-iso-iec-15504-spice) . Acesso em: 09 abril 2023.

PROMOVE SOLUÇÕES. MPS.BR IMPLANTAÇÃO INTEGRADA DO MPS PARA SOFTWARE E MPS PARA SERVIÇOS DESENVOLVENDO SOFTWARE COMO SERVI. Disponível em: https://promovesolucoes.com/mps-br/. Acesso em: 04 abril 2023.