



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO E SISTEMAS  
CURSO ENGENHARIA DE PRODUÇÃO MECÂNICA

Gustavo Costa de Matos

**DESENVOLVIMENTO DE UMA VERSÃO WEB MODERNIZADA DO JOGO  
LSSP\_PCP1 COMO FERRAMENTA EDUCACIONAL PARA O ENSINO DE PCP**

Florianópolis

2024

Gustavo Costa de Matos

**DESENVOLVIMENTO DE UMA VERSÃO WEB MODERNIZADA DO JOGO  
*LSSP\_PCP1* COMO FERRAMENTA EDUCACIONAL PARA O ENSINO DE PCP**

Trabalho de Conclusão de Curso submetido ao curso de Engenharia de Produção Mecânica do Centro Tecnológico da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Engenharia Mecânica com Habilitação em Produção.

Orientador: Prof. Guilherme Ernani Vieira

Florianópolis

2024

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.  
Dados inseridos pelo próprio autor.

Matos, Gustavo Costa de

Desenvolvimento de uma versão web modernizada do jogo  
LSSP\_PCPI como ferramenta educacional para o ensino de PCP  
/ Gustavo Costa de Matos ; orientador, Guilherme Ernani  
Vieira, 2024.

103 p.

Trabalho de Conclusão de Curso (graduação) -  
Universidade Federal de Santa Catarina, Centro  
Tecnológico, Graduação em Engenharia de Produção Mecânica,  
Florianópolis, 2024.

Inclui referências.

1. Engenharia de Produção Mecânica. 2. planejamento e  
controle da produção. 3. jogos educacionais. 4. simulação.  
5. python. I. Vieira, Guilherme Ernani. II. Universidade  
Federal de Santa Catarina. Graduação em Engenharia de  
Produção Mecânica. III. Título.

Gustavo Costa de Matos

**DESENVOLVIMENTO DE UMA VERSÃO WEB MODERNIZADA DO JOGO  
LSSP\_PCP1 COMO FERRAMENTA EDUCACIONAL PARA O ENSINO DE PCP**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do título de Engenharia Mecânica com Habilitação em Produção e aprovado em sua forma final pelo Curso Engenharia de Produção Mecânica

Local Florianópolis, 12 de dezembro de 2024.

Coordenação do Curso

**Banca examinadora**

Prof. Guilherme Ernani Vieira, Dr.

Orientador

Prof. Enzo Morosini Frazzon, Dr.

Instituição UFSC

Prof. Lynceo Falavigna Braghirolli, Dr.

Instituição UFSC

Florianópolis, 2024.

Dedico este trabalho ao meu eterno companheiro Toby, que deixou em minhas memórias um legado de amor, força e lealdade.

## **AGRADECIMENTOS**

Primeiramente, gostaria de expressar minha profunda gratidão à minha família pelo apoio incondicional, esforço e por sempre acreditarem em mim ao longo desta jornada. Sem o incentivo e o carinho de vocês, este trabalho não seria possível. Cada palavra de encorajamento e cada gesto de suporte foram essenciais para superar os desafios e concluir esta etapa tão importante da minha vida.

Meu sincero agradecimento também ao meu orientador, Guilherme Ernani Vieira, pela paciência, disponibilidade e dedicação ao longo de todo o processo. Suas orientações e feedbacks foram fundamentais para o desenvolvimento deste trabalho, ajudando-me a refinar ideias e direcionar esforços na busca por resultados significativos. Sua contribuição foi indispensável para o sucesso deste projeto, e sou profundamente grato por seu apoio.

Agradeço igualmente aos meus amigos, que estiveram ao meu lado oferecendo apoio, motivação e companheirismo, especialmente nos momentos mais desafiadores desta trajetória. Vocês tornaram esse percurso mais leve e significativo. Em especial, agradeço ao Cidade, pelos incontáveis momentos de auxílio e parceria; ao Vicente, pelos ensinamentos indubitavelmente valiosos; ao Fontão, pelo apoio incansável durante minha caminhada; ao Caio, que sempre foi um exemplo de referência.

Estendo minha gratidão a todos os outros amigos e colegas que, direta ou indiretamente, fizeram parte dessa caminhada. Mesmo que seus nomes não estejam mencionados aqui, saibam que cada conversa, gesto ou palavra de incentivo tiveram um impacto significativo e foram fundamentais para que eu chegasse até aqui.

Gostaria também de agradecer à minha namorada, que esteve ao meu lado em todos os momentos, oferecendo amor, paciência e encorajamento. Sua compreensão nos períodos mais desafiadores e seu apoio constante foram cruciais para que eu pudesse seguir em frente com confiança e determinação.

Por fim, agradeço aos professores que, ao longo do curso, contribuíram para minha formação acadêmica e profissional. O aprendizado e as experiências compartilhadas foram valiosas e tiveram um impacto significativo na construção deste trabalho e para a minha formação acadêmica. Agradeço também à Universidade Federal de Santa Catarina, por proporcionar um ambiente que estimula o crescimento acadêmico, profissional e pessoal.

*“Ensinar não é transferir conhecimento, mas criar as possibilidades para a sua própria produção ou a sua construção”*

*(Paulo Freire, 1996, p. 25)*

## RESUMO

Este trabalho apresenta a modernização e o desenvolvimento de uma nova versão do jogo educacional *LSSP\_PCP1*, criado originalmente para apoiar o ensino de Planejamento e Controle da Produção em um ambiente acadêmico. A versão original, desenvolvida em uma plataforma desktop, mostrou-se limitada em acessibilidade e usabilidade frente às necessidades e tecnologias atuais. A nova implementação, agora como uma aplicação web, visa superar essas limitações, oferecendo uma experiência mais interativa e acessível aos usuários. Através do uso de tecnologias modernas, como *Python*, *HTML* e *CSS*, o jogo foi redesenhado para ser acessível via navegador em diversos dispositivos, eliminando a necessidade de instalação local e permitindo futuras expansões e melhorias. A estrutura do jogo mantém a lógica do original, simulando atividades de PCP, incluindo previsão de demanda, planejamento de produção e controle de estoques. No entanto, melhorias foram incorporadas para otimizar a experiência do usuário, como novos gráficos interativos e um painel de administrador para o gerenciamento dos grupos de jogadores. Além de atualizar o *LSSP\_PCP1*, o projeto foi desenvolvido para facilitar a futura incorporação dos demais jogos da série *LSSP\_PCP*, promovendo uma plataforma unificada para o ensino de PCP. Como código aberto, o projeto oferece à comunidade acadêmica a oportunidade de colaborar, permitindo que o jogo evolua continuamente para atender a demandas educacionais emergentes.

**Palavras-chave:** planejamento e controle da produção; jogos educacionais; simulação; python

## ABSTRACT

This work presents the modernization and development of a new version of the educational game *LSSP\_PCP1*, originally created to support the teaching of Production Planning and Control in an academic environment. The original version, developed on a desktop platform, proved limited in accessibility and usability when compared to current needs and technologies. The new implementation, now as a web application, aims to overcome these limitations by providing a more interactive and accessible user experience. Using modern technologies such as *Python*, *HTML* and *CSS*, the game was redesigned to be accessible via a browser on multiple devices, eliminating the need for local installation and allowing for future expansions and enhancements. The game structure retains the logic of the original, simulating PPC activities, including demand forecasting, production planning, and inventory control. However, improvements have been incorporated to optimize user experience, such as new interactive graphics and an administrator panel for managing player groups. In addition to updating *LSSP\_PCP1*, the project was developed to facilitate the future integration of other games in the *LSSP\_PCP* series, promoting a unified platform for PPC education. As an open-source project, it offers the academic community the opportunity to collaborate, enabling the game to continually evolve to meet emerging educational demands.

**Keywords:** production planning and control; educational games; simulation; python

## LISTA DE FIGURAS

Figura 1: Exemplo Gráfico de Gantt .....	24
Figura 2: Diagrama relação <i>Frontend</i> - <i>Backend</i> .....	26
Figura 3: Processo Produtivo .....	37
Figura 4: Tela Inicial .....	38
Figura 5: Dinâmica de PCP do Jogo .....	40
Figura 6: Tela Cadastro.....	41
Figura 7: Tela do Formulário Previsão Colméia .....	42
Figura 8: Tela do Formulário Erros de Previsão.....	43
Figura 9: Tela do Formulário Plano de Produção Colméia.....	44
Figura 10: Tela do Formulário Plano de Compras Fio 1.....	46
Figura 11: Tela Formulário Capacidade Necessária Tecelagem.....	48
Figura 12: Tela Formulário Capacidade Disponível Tecelagem.....	49
Figura 13: Tela Formulário Capacidade Futura Tecelagem .....	50
Figura 14: Tela Formulário Capacidade Disponível Purga/Tinturaria.....	51
Figura 15: Tela Formulário Capacidade Futura Purga/Tinturaria .....	52
Figura 16: Tela Formulário Plano de Produção Financeiro .....	53
Figura 17: Tela do Formulário Engenharia .....	54
Figura 18: Tela do Formulário Itens .....	55
Figura 19: Tela do Formulário Taxa Produção .....	56
Figura 20: Tela Formulário Estrutura.....	57
Figura 21: Tela Formulário Custos .....	57
Figura 22: Página do GitHub do Projeto.....	61
Figura 23: Trecho do código arquivo <i>production.html</i> .....	64
Figura 24: Trecho parte 1 da rota <i>production</i> .....	68
Figura 25: Trecho parte 2 da rota <i>production</i> .....	69
Figura 26: Diagrama Simplificado de Relacionamento de Entidade.....	72
Figura 27: Consulta Tabela Lead Time Máquinas.....	78
Figura 28: Gráfico Interativo Produção Planejada.....	84
Figura 29: Gráfico Interativo Produção Planejada com Filtro .....	84
Figura 30: Exemplo Indicador Visual Plano Tecelagem .....	85
Figura 31: Exemplo Mecanismo de Validação .....	86

Figura 32: Página Primeiro Acesso .....	87
Figura 33: Página de <i>Login</i> .....	89
Figura 34: Página <i>Dashboard</i> .....	90
Figura 35: Página Plano de Produção.....	91
Figura 36: Página Plano de Compras .....	92
Figura 37: Página Capacidade Tecelagem .....	93
Figura 38: Formulários Capacidade Disponível e Ampliações e Reduções .....	93
Figura 39: Página Relatório Financeiro .....	95
Figura 40: Página de Cadastro de Semestres.....	96
Figura 41: Página Cadastro de Turmas .....	96
Figura 42: Página Gerenciar Usuários .....	97

## LISTA DE QUADROS

Quadro 1: Grupo .....	73
Quadro 2: Estilo de Demanda .....	74
Quadro 3: Previsão de Demanda .....	74
Quadro 4: Plano de Produção .....	75
Quadro 5: Plano de Compras .....	76
Quadro 6: Capacidade Teares .....	77
Quadro 7: Taxa Produção .....	78
Quadro 8: Relatório Financeiro .....	79
Quadro 9: Custos .....	80

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>15</b>
1.1	OBJETIVOS DO TRABALHO .....	17
1.2	OBJETIVOS ESPECÍFICOS .....	18
1.3	JUSTIFICATIVA .....	18
1.4	ESTRUTURA DO TRABALHO.....	19
<b>2</b>	<b>REFERENCIAL TEÓRICO</b> .....	<b>20</b>
2.1	PREVISÃO DE DEMANDA .....	20
2.2	PLANEJAMENTO E CONTROLE DA PRODUÇÃO .....	21
<b>2.2.1</b>	<b>Planejamento Estratégico da Produção</b> .....	<b>22</b>
<b>2.2.2</b>	<b>Planejamento Mestre da Produção</b> .....	<b>22</b>
<b>2.2.3</b>	<b>Programação da Produção</b> .....	<b>23</b>
2.3	JOGOS NO PROCESSO DE APRENDIZAGEM .....	25
2.4	DESENVOLVIMENTO WEB .....	26
<b>2.4.1</b>	<b>Frontend</b> .....	<b>27</b>
<b>2.4.2</b>	<b>Backend</b> .....	<b>28</b>
<b>2.4.3</b>	<b>Banco de Dados</b> .....	<b>28</b>
<b>2.4.4</b>	<b>Transição para Plataformas Web</b> .....	<b>30</b>
<b>3</b>	<b>PROCEDIMENTOS METODOLÓGICOS</b> .....	<b>31</b>
3.1	CARACTERIZAÇÃO DA PESQUISA.....	31
3.2	PESQUISA APLICADA .....	32
3.3	<i>REDESIGN</i> INSTRUCIONAL.....	32
3.4	REENGENHARIA DE JOGOS.....	33
3.5	ROTEIRO METODOLÓGICO .....	34
<b>3.5.1</b>	<b>Análise da Versão Original do Jogo LSSP_PCP1</b> .....	<b>34</b>
<b>3.5.2</b>	<b>Desenvolvimento da Nova Versão e Mudanças Propostas</b> .....	<b>34</b>
<b>3.5.3</b>	<b>Avaliação Comparativa e Validação</b> .....	<b>35</b>
<b>4</b>	<b>DESENVOLVIMENTO</b> .....	<b>36</b>
4.1	ANÁLISE DO JOGO ATUAL.....	36
<b>4.1.1</b>	<b>Fluxo do Jogo</b> .....	<b>37</b>
<b>4.1.2</b>	<b>Principais Funcionalidades</b> .....	<b>41</b>
<b>4.1.2.1</b>	<b>Cadastro do Grupo</b> .....	<b>41</b>

4.1.2.2	<i>Previsão de Demanda</i> .....	42
4.1.2.3	<i>Planejamento de Produção</i> .....	44
4.1.2.4	<i>Plano de Compras</i> .....	45
4.1.2.5	<i>Cálculos de Capacidades</i> .....	47
4.1.2.6	<i>Plano de Produção Financeiros</i> .....	53
4.1.2.7	<i>Engenharia</i> .....	54
<b>4.1.3</b>	<b>Identificação de Limitações Tecnológicas e Pontos de Melhorias</b> .....	<b>58</b>
4.2	DESENVOLVIMENTO DO NOVO JOGO .....	59
<b>4.2.1</b>	<b>Tecnologias Utilizadas</b> .....	<b>59</b>
<b>4.2.2</b>	<b>Arquitetura Geral do Sistema</b> .....	<b>62</b>
4.2.2.1	<i>Frontend</i> .....	63
4.2.2.2	<i>Backend</i> .....	67
4.2.2.3	<i>Banco de Dados</i> .....	71
<b>4.2.3</b>	<b>Fluxo do jogo atualizado</b> .....	<b>82</b>
4.3	NOVAS FUNCIONALIDADES .....	82
<b>4.3.1</b>	<b>Jogo como Aplicação Web</b> .....	<b>83</b>
<b>4.3.2</b>	<b>Painel Dashboard Interativo</b> .....	<b>83</b>
<b>4.3.3</b>	<b>Indicadores Visuais</b> .....	<b>84</b>
<b>4.3.4</b>	<b>Contas de Acesso por Grupo</b> .....	<b>86</b>
<b>4.3.5</b>	<b>Painel de Administrador</b> .....	<b>88</b>
4.4	INTERFACES DO USUÁRIO.....	88
<b>4.4.1</b>	<b><i>Login</i></b> .....	<b>89</b>
<b>4.4.2</b>	<b><i>Dashboard</i></b> .....	<b>89</b>
<b>4.4.3</b>	<b>Planejamento da Produção</b> .....	<b>90</b>
<b>4.4.4</b>	<b>Plano de Compras</b> .....	<b>91</b>
<b>4.4.5</b>	<b>Telas de Capacidade</b> .....	<b>92</b>
<b>4.4.6</b>	<b>Financeiro</b> .....	<b>94</b>
<b>4.4.7</b>	<b>Painel Administrador</b> .....	<b>95</b>
<b>5</b>	<b>CONCLUSÃO</b> .....	<b>98</b>
	<b>REFERÊNCIAS</b> .....	<b>100</b>

## 1 INTRODUÇÃO

O Planejamento e Controle da Produção (PCP) é um sistema vital para o sucesso das empresas industriais. Ele se configura como uma ferramenta estratégica para a gestão da produção, pois permite o planejamento, organização e controle de todos os recursos envolvidos no processo produtivo, desde a aquisição de matéria-prima até a entrega do produto final ao cliente. Através do PCP, as empresas podem otimizar seus processos, reduzir custos, aumentar a produtividade e melhorar a qualidade dos seus produtos.

De acordo com Arenales et al. (2007), a vasta literatura existente nessa área é resultado da grande diversidade de sistemas de produção, abrangendo diferentes aspectos como a configuração dos processos produtivos, as características dos produtos e os critérios de desempenho dos programas de produção.

Essa tendência evidencia a importância e o interesse crescente na compreensão e aprimoramento das práticas e metodologias no âmbito do PCP. Essa crescente demanda por conhecimentos e abordagens eficazes nas áreas do PCP no ensino superior destaca a necessidade de estratégias pedagógicas que possam auxiliar os alunos a compreenderem e aplicar esses conceitos de forma prática e significativa. Nesse sentido os jogos educacionais surgem como uma ferramenta promissora pois proporcionam um ambiente de aprendizagem imersivo e interativo, permitindo que os estudantes experimentem diferentes situações e tomem decisões relacionadas às funções do PCP em um contexto simulado.

A oportunidade de transferir os princípios da aprendizagem baseada em ação do ambiente empresarial para o contexto da educação superior é uma maneira de conectar a formação do indivíduo com o futuro ambiente profissional. No contexto do processo de ensino em cursos de ensino superior, o uso de jogos desperta a necessidade do aluno de aprender em tempo real e compreender a relevância do que está aprendendo em suas experiências, tornando assim a aprendizagem significativa e transformadora.

Segundo Lewis e Maylor (2007) a aprendizagem baseada em jogos é uma estratégia de aprendizagem eficaz e os jogos educacionais são amplamente estudados e adotados para alunos que vão desde crianças até graduandos e adultos.

A aprendizagem com jogos desperta a necessidade de o aluno aprender em tempo real simulando situações reais e desafiadoras, o que contribui para o desenvolvimento de habilidades práticas e cognitivas. Ao se envolver ativamente em ambientes virtuais ou interativos, o estudante pode vivenciar diferentes cenários, aplicar conhecimentos teóricos na prática e obter feedback imediato sobre suas ações.

Além disso, essa abordagem promove a percepção da importância do que é aprendido, conectando os conhecimentos teóricos com experiências vividas. Isso torna o processo de aprendizagem mais significativo pois os estudantes reconhecem a relevância prática e aplicabilidade dos conceitos aprendidos em suas futuras trajetórias profissionais.

Segundo Savi e Ulbricht (2008), os jogos educacionais são utilizados nas instituições de ensino como uma ferramenta didática que apresenta atributos capazes de contribuir de forma positiva para as práticas de ensino e aprendizagem.

No entanto, a fim de serem empregados com propósitos educacionais, é necessário que os jogos possuam metas de aprendizagem claramente estabelecidas e sejam capazes de transmitir conteúdos disciplinares aos usuários. Além disso podem ser utilizados para promover o desenvolvimento de estratégias ou habilidades que contribuam para a expansão da capacidade cognitiva e intelectual dos alunos. (SAVI e ULBRICHT, 2008, p.2)

No contexto da engenharia de produção e do ensino relacionado ao Planejamento e Controle da Produção, o professor Dr. Dalvio Ferrari Tubino é uma referência importante na área. Com diversas publicações, ele desempenha um papel fundamental na disseminação do conhecimento em PCP, facilitando o aprendizado e tornando o tema mais acessível a estudantes e profissionais. Além disso, o seu livro *Planejamento e Controle da Produção: Teoria e Prática*, apresenta uma base sólida para o estudo de PCP. Para reforçar e facilitar a aplicação prática dessa teoria ele idealizou uma série de jogos educacionais desenvolvidos originalmente em *Microsoft Acces*, com o objetivo de oferecer uma ferramenta prática de ensino.

A série de jogos consiste em três jogos, *LSSP\_PCP1*, *LSSP\_PCP2*, *LSSP\_PCP3*, cada um com um escopo específico que aborda diferentes aspectos do PCP. Embora os jogos da série *LSSP\_PCP* tenham sido desenvolvidos em uma época em que as ferramentas tecnológicas disponíveis eram limitadas em

comparação com as soluções modernas, eles continuam a ter relevância pedagógica devido à sua capacidade de simular cenários de PCP. A defasagem tecnológica manifesta-se principalmente em aspectos como a falta de interfaces intuitivas, gráficos interativos mais sofisticados e a ausência de recursos que permitam maior acessibilidade, como a compatibilidade com diferentes dispositivos ou a execução diretamente em navegadores web.

Essas limitações, no entanto, não diminuem o mérito educacional desses jogos. Eles ainda conseguem transmitir conceitos importantes, como a importância do planejamento estratégico, o impacto das decisões operacionais e as consequências de erros de planejamento em um ambiente simulado. Por outro lado, a evolução tecnológica trouxe novas expectativas e demandas para ferramentas educacionais, como design responsivo, maior interatividade e capacidade de personalização. Isso tornou evidente a necessidade de modernização para que os jogos possam acompanhar as práticas educacionais contemporâneas.

Com a atualização tecnológica, como a migração para plataformas web, seria possível superar essas limitações, ampliando o alcance do público-alvo e proporcionando uma experiência mais imersiva e eficiente. Além disso, a modernização permitiria a introdução de novos recursos, como painéis de controle interativos, relatórios automatizados e a integração com outras ferramentas educacionais, garantindo que os jogos continuem relevantes no contexto atual.

Portanto, enquanto as limitações tecnológicas representam um desafio, elas também oferecem uma oportunidade para transformar essas ferramentas em plataformas mais modernas e adaptadas às necessidades do ensino atual. Essa atualização não só preservaria a essência dos jogos como ferramentas educacionais, mas também potencializaria sua eficácia no ensino prático de PCP, atendendo às demandas tanto de alunos quanto de educadores.

Diante do exposto, os objetivos deste trabalho serão apresentados a seguir, acompanhados de suas respectivas justificativas.

## 1.1 OBJETIVOS DO TRABALHO

O presente trabalho tem como objetivo geral desenvolver uma nova implementação do jogo *LSSP\_PCP1* já existente, criando uma plataforma que permita

a futura expansão para a incorporação dos demais jogos da série *LSSP\_PCP*. Essa nova implementação busca modernizar o jogo, incorporando novas funcionalidades e melhorias na interface, com o intuito de tornar a experiência de aprendizado mais interativa e eficaz.

## 1.2 OBJETIVOS ESPECÍFICOS

O objetivo geral do trabalho foi desdobrado nos seguintes objetivos específicos:

- a) Analisar as funcionalidades da versão original do jogo *LSSP\_PCP1*;
- b) Desenvolver a versão modernizada do jogo para acesso via navegador em dispositivos móveis e computadores;
- c) Incorporar novas funcionalidades para a versão modernizada do jogo;

## 1.3 JUSTIFICATIVA

A relevância deste trabalho está na modernização do jogo educacional *LSSP\_PCP1*, uma ferramenta utilizada para o ensino de PCP. No entanto, a versão original apresenta limitações de acessibilidade e funcionalidade, especialmente diante das tecnologias e expectativas atuais de interatividade e usabilidade em plataformas educacionais.

Esta nova implementação, agora como uma aplicação web, busca superar essas limitações, ampliando o alcance e a eficácia do jogo. Ao optar por uma versão web, o jogo torna-se mais acessível, podendo ser utilizado diretamente no navegador e em diferentes dispositivos, como computadores, tablets e smartphones. Essa modernização não apenas facilita o uso em contextos educacionais variados, mas também permite a integração de novos recursos que enriquecem a experiência do usuário. Além disso, o trabalho contribui para a disseminação dos conceitos de PCP de forma prática e intuitiva, alinhando-se à crescente demanda por ferramentas de ensino que combinam teoria e prática.

## 1.4 ESTRUTURA DO TRABALHO

Este trabalho está organizado em seis capítulos, cada um abordando uma etapa fundamental do desenvolvimento da nova versão do jogo educacional *LSSP\_PCP1*.

No Capítulo 1, foi apresentada a introdução, que contextualiza a importância PCP e dos jogos educacionais como ferramentas pedagógicas, além de descrever os objetivos, justificativas e relevância deste estudo.

O Capítulo 2 trata do referencial teórico, fundamentando as principais bases teóricas relacionadas ao PCP, à aprendizagem baseada em jogos e às tecnologias empregadas no desenvolvimento de aplicações web.

O Capítulo 3 apresenta os procedimentos metodológicos adotados para a realização deste trabalho, detalhando as etapas de análise da versão original, desenvolvimento da nova versão e validação preliminar.

No Capítulo 4, são descritos o desenvolvimento e os resultados obtidos, com ênfase nas tecnologias utilizadas, nas funcionalidades implementadas e nas melhorias introduzidas em comparação à versão original do jogo, abordando as discussões e análises dos resultados, destacando os avanços proporcionados pela nova versão do jogo e as suas limitações.

Por fim, o Capítulo 5 apresenta as considerações finais, incluindo uma síntese dos resultados alcançados, sugestões para trabalhos futuros e a relevância do trabalho para o ensino de PCP.

A organização do trabalho busca garantir uma progressão lógica e coerente entre os capítulos, de modo a facilitar a compreensão dos leitores e a contextualização dos resultados obtidos no âmbito do estudo. Adicionalmente, o escopo deste trabalho foi delimitado ao desenvolvimento da nova versão do jogo *LSSP\_PCP1*, sem incluir a implementação dos demais jogos da série, nem uma validação prática detalhada em contextos educacionais. Essas delimitações foram estabelecidas para viabilizar a conclusão do projeto dentro do cronograma proposto, mantendo o foco no aprimoramento da funcionalidade e acessibilidade do jogo principal.

## 2 REFERENCIAL TEÓRICO

Este capítulo dedica-se à apresentação do referencial teórico que fundamenta o trabalho. Nele, serão exploradas as principais características e conceitos teóricos que embasaram e orientaram o desenvolvimento da plataforma e a implementação do jogo.

### 2.1 PREVISÃO DE DEMANDA

A previsão de demanda é um componente crítico no PCP, desempenhando um papel central na tomada de decisões estratégicas e operacionais dentro das organizações. em períodos futuros permitindo que a empresa planeje sua produção, estoques e recursos de forma eficiente. A precisão na previsão de demanda é essencial para evitar tanto o excesso quanto a falta de produtos, o que pode resultar em custos elevados, perda de oportunidades de venda, e insatisfação do cliente.

Existem diversas técnicas utilizadas para a previsão de demanda, que variam desde métodos qualitativos, baseados em julgamentos e opiniões de especialistas, até métodos quantitativos, que utilizam dados históricos e modelos estatísticos para prever tendências futuras. A escolha da técnica adequada depende de fatores como a disponibilidade de dados, a natureza do mercado e a variabilidade da demanda. Nos últimos anos, avanços em tecnologia e análise de dados têm permitido o desenvolvimento de modelos mais sofisticados, como o uso de aprendizado de máquina e inteligência artificial, que melhoram a acurácia das previsões ao identificar padrões complexos nos dados.

No contexto do desenvolvimento de software de simulação para PCP, a incorporação de modelos de previsão de demanda é fundamental para a criação de cenários realistas e para a análise do impacto de diferentes estratégias de produção.

Como observado por Makridakis, Wheelwright e Hyndman (1998), a precisão nas previsões permite que os gestores avaliem como variações na demanda podem afetar o fluxo de produção, auxiliando na identificação de gargalos e na tomada de decisões mais informadas sobre alocação de recursos, programação de produção e gestão de estoques. Dessa forma, a previsão de demanda não só apoia o

planejamento de curto prazo, mas também contribui para a resiliência e adaptabilidade da cadeia de suprimentos em um ambiente competitivo e em constante mudança.

## 2.2 PLANEJAMENTO E CONTROLE DA PRODUÇÃO

De acordo com Tubino (2009), as atividades de PCP são organizadas em três níveis distintos: estratégico, tático e operacional.

No nível estratégico, o foco está no desenvolvimento do Plano de Produção de longo prazo (anos), alinhado com as projeções de vendas e a capacidade financeira e produtiva da empresa.

O nível tático concentra-se na elaboração do Planejamento Mestre da Produção (PMP), que é detalhado para o médio prazo (meses) e baseia-se nas previsões de vendas para esse período. Durante essa fase, é crucial que o PCP avalie as necessidades de recursos produtivos para identificar eventuais gargalos na produção.

Por fim, o nível operacional aborda a Programação da Produção, que é realizada a curto prazo (semanas e dias), determinando a quantidade, o momento e o local de produção de cada item.

Um sistema de PCP é considerado de qualidade quando apresenta a capacidade de oferecer a melhor solução para a utilização dos recursos de produção, garantindo uma alta eficiência produtiva. Dentre os principais objetivos do PCP, destacam-se:

1. Reduzir estoque;
2. Minimizar atrasos e não atendimento de ordens de fabricação;
3. Minimizar ociosidade de recursos produtivos;
4. Minimizar *lead-time* do processo.

Conforme a definição de Fernandes e Godinho Filho (2008), as atividades que compõem o PCP têm como propósito estabelecer as decisões relacionadas ao que será produzido, comprado e entregue, considerando o tempo e a quantidade envolvidos. Além disso, essas atividades abrangem a alocação adequada de recursos

humanos e locais de produção, buscando garantir uma execução eficiente do processo.

### **2.2.1 Planejamento Estratégico da Produção**

O Planejamento Estratégico da Produção (PEP) é uma fase essencial na gestão da cadeia de suprimentos, envolvendo decisões que afetam o desempenho de longo prazo da empresa. Durante o PEP, são estabelecidas as metas e os objetivos globais para a produção, além das diretrizes que orientarão as operações de manufatura ao longo dos anos. Chopra e Meindl (2016) destacam que decisões estratégicas incluem a configuração da cadeia de suprimentos, alocação de recursos e definição dos processos que cada estágio da cadeia executará. Essas decisões, que geralmente têm um horizonte de tempo longo, são fundamentais para garantir que a organização possa responder de maneira eficaz às mudanças do mercado e da tecnologia.

Complementando essa visão, Tubino (2009) enfatiza que o PEP deve alinhar as operações de produção com a estratégia corporativa, garantindo que as decisões tomadas promovam a competitividade e a sustentabilidade da empresa no mercado. Ele destaca que o sucesso do PEP depende da integração entre os diversos setores da empresa, como finanças, marketing e operações, de modo a criar uma cadeia de valor eficiente e coesa.

### **2.2.2 Planejamento Mestre da Produção**

O Planejamento Mestre da Produção (PMP) é uma fase intermediária no processo de gestão da produção, onde são desenvolvidos planos detalhados baseados nas diretrizes estratégicas estabelecidas no PEP. Nesta etapa, as empresas planejam a alocação de recursos para maximizar o valor gerado, considerando as restrições definidas pelo planejamento estratégico. Chopra e Meindl (2016) explicam que o PMP envolve a previsão de demanda, definição de políticas de inventário e determinação dos níveis de produção e capacidade para o próximo ano ou trimestre.

Tubino (2009) acrescenta que o PMP é crucial para garantir a continuidade operacional e a eficiência dos processos produtivos. Ele destaca que essa fase deve considerar tanto a demanda prevista quanto a capacidade disponível, ajustando o plano conforme as variabilidades do mercado. A flexibilidade no PMP é essencial para que a empresa possa responder rapidamente a mudanças na demanda e nas condições de mercado, minimizando riscos e aproveitando oportunidades.

### **2.2.3 Programação da Produção**

Segundo Tubino (2009), a Programação da Produção é uma das atividades pertencentes ao PCP e que tem como objetivo elaborar os planos que guiarão a produção e realizar o controle da produção, fundamentando-se nesses planos.

De acordo com Fuchigami (2005), a programação da produção envolve a organização e alocação de recursos para o processamento das tarefas em um determinado período. Esse processo pode ser dividido em dois meios distintos:

- 1) Determinação do tempo e local para a execução de cada tarefa no processo produtivo;
- 2) Determinação das datas de início e término para a conclusão de cada tarefa no processo.

No primeiro caso é necessário estabelecer a sequência correta das atividades, levando em consideração os recursos disponíveis, as restrições de capacidade e os fluxos de trabalho. É fundamental definir onde e quando cada tarefa deve ser realizada, garantindo uma operação eficiente e fluída.

No segundo caso, além da sequência, é necessário estabelecer prazos para a execução de cada atividade. Isso implica em definir as datas de início e término para cada tarefa, considerando fatores como a prioridade, a interdependência entre as atividades e os objetivos do projeto. Essa programação de datas permite um melhor controle do cronograma e o cumprimento dos prazos estabelecidos.

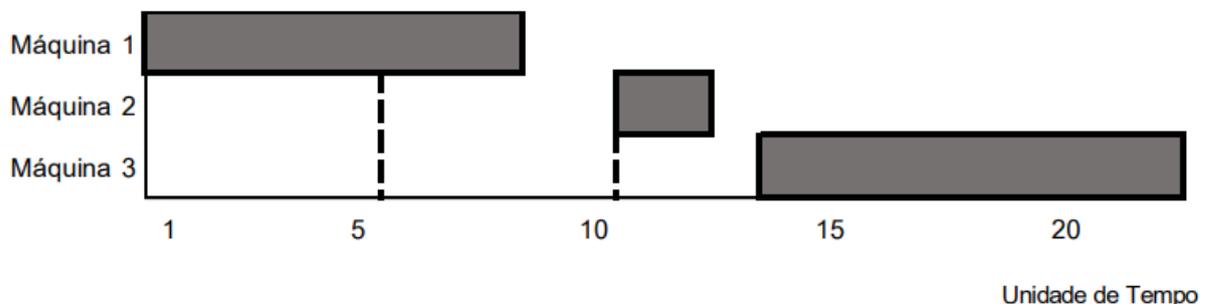
Dessa forma a programação da produção envolve não apenas a organização das tarefas, mas também a alocação eficiente dos recursos e a definição de prazos realistas. Ao considerar esses dois meios, é possível planejar e executar as atividades

de forma eficaz, garantindo o máximo desempenho e produtividade no processo produtivo.

A atividade de programação é reconhecida como uma das mais desafiadoras na gestão da produção, uma vez que envolve a organização das diversas tarefas em um determinado número de máquinas. Em essência, a programação determina a sequência, prioridade e momento exato em que cada ordem deve ser executada, além de especificar os recursos necessários para atingir o objetivo desejado. Essa complexidade implica na necessidade de considerar múltiplos aspectos simultaneamente (Slack et al., 2002).

O gráfico de Gantt é uma técnica de programação desenvolvida por H. L. Gantt em 1917, que continua sendo amplamente utilizada atualmente devido à sua capacidade de proporcionar uma visualização clara e simplificada do progresso do processo. Nessa representação gráfica, conforme mencionado por Slack (2002), o tempo é representado por meio de barras que indicam os momentos de início e término das atividades. A Figura 1 abaixo mostra um exemplo desse gráfico:

Figura 1: Exemplo Gráfico de Gantt



Fonte: Terao (2017)

Essa ferramenta é amplamente utilizada em diversos setores, como gestão de projetos, administração da produção e planejamento de eventos. Por sua simplicidade e eficácia na representação visual das atividades, o gráfico de Gantt tornou-se uma referência na programação e no acompanhamento de projetos, permitindo que os envolvidos tenham uma compreensão clara das etapas do processo e possam tomar decisões com base nessas informações.

## 2.3 JOGOS NO PROCESSO DE APRENDIZAGEM

Para Valente (1998), no jogo educacional o usuário está livre para aprender através de um ambiente exploratório, usando uma abordagem da exploração autodirigida, em contraste com a instrução explícita e direta. Segundo o autor, os aprendizes gostam, e do ponto de vista deles é a maneira mais divertida e gostosa de aprender. Os estudantes ficam mais motivados a usar a inteligência, pois querem jogar bem; sendo assim, esforçam-se para superar obstáculos, tanto cognitivos quanto emocionais. Estando mais motivadas durante o jogo, ficam também mais ativas mentalmente (Valente, 1998)

De acordo com Jucá (2011), um software é considerado educacional quando é utilizado de forma apropriada no contexto de ensino-aprendizagem. Sua principal característica é que ele é desenvolvido com base em uma teoria de aprendizagem, possibilitando que o aluno adquira conhecimento sobre um determinado assunto de maneira autônoma.

Os softwares educativos têm como objetivo principal apoiar os processos de ensino-aprendizagem e são desenvolvidos especificamente para facilitar a construção de conhecimento sobre um conteúdo didático. Entre suas principais características estão o enfoque didático, que visa ajudar o aluno ou usuário a desenvolver conhecimento em uma área específica, a capacidade de promover a interação entre o usuário e o programa, muitas vezes mediada pelo professor, e a facilidade de uso, que permite que qualquer usuário possa realizar suas atividades sem dificuldades.

Prensky (2001) defende que jogos digitais, quando bem estruturados, são ferramentas poderosas para a educação, uma vez que combinam aspectos de entretenimento com processos cognitivos profundos. Ele argumenta que jogos educacionais são particularmente eficazes para envolver estudantes, pois promovem um nível de concentração e dedicação que muitas vezes supera o observado em atividades tradicionais.

Por fim, o desenvolvimento de jogos educacionais web modernos requer uma abordagem multidisciplinar, que combina conhecimentos de programação, design, pedagogia e experiência do usuário. Ao alinhar esses elementos, é possível criar ferramentas educacionais inovadoras que não apenas transmitem conteúdo de forma

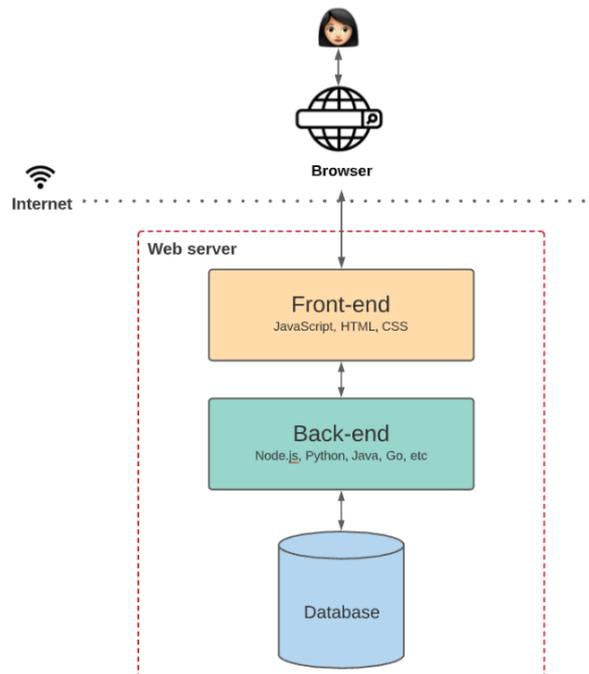
eficaz, mas também estimulam o engajamento e a motivação dos aprendizes, potencializando o processo de ensino-aprendizagem.

## 2.4 DESENVOLVIMENTO WEB

A escolha de tecnologias web modernas no desenvolvimento de jogos educacionais contribui significativamente para ampliar a acessibilidade e usabilidade dessas ferramentas. Segundo Lutz (2013), *Python*, uma linguagem de programação reconhecida por sua simplicidade e versatilidade, é amplamente usada no desenvolvimento de aplicações educacionais por facilitar a integração com bibliotecas e frameworks necessários para aplicações robustas e escaláveis.

Uma aplicação web, de forma geral e simplificada, é composta por dois componentes principais: o *backend* (parte do servidor) e o *frontend* (parte do usuário). O *backend* é responsável por lidar com toda a lógica de negócios, processamento de dados e comunicação com o banco de dados, sendo o núcleo funcional da aplicação. Ele gerencia as requisições dos usuários, processa dados e garante a segurança e integridade das informações trocadas durante a navegação (Tianfield, 2016). A Figura 2 ilustra de forma simplificada esse conjunto de componentes descritos.

Figura 2: Diagrama relação *Frontend - Backend*



Fonte: Preethi Kasireddy (2021)

Em sua dissertação sobre estilos de arquitetura de software, Fielding (2000) discute a importância de arquiteturas escaláveis para sistemas distribuídos, como é o caso de aplicações web, argumentando que um design cuidadoso na escolha de tecnologias permite que a aplicação seja ampliada sem perda de desempenho.

Dentro desse contexto, a seleção cuidadosa das tecnologias empregadas é essencial para o sucesso no desenvolvimento de jogos educacionais web. É fundamental considerar fatores como desempenho, escalabilidade, segurança e facilidade de manutenção. Além disso, Fielding (2000), reforça que a adoção de padrões de desenvolvimento e boas práticas contribui para a criação de um código mais limpo e eficiente, facilitando futuras atualizações e aprimoramentos.

A integração entre o *backend* e o *frontend* deve ser planejada com cuidado para garantir uma comunicação fluida e eficiente. A utilização de interfaces de programação bem definidas facilita essa interação, permitindo a troca segura e estruturada de informações. Essa sinergia entre os componentes é vital para o funcionamento harmonioso da aplicação como um todo.

#### **2.4.1 Frontend**

O *frontend*, é a camada visível ao usuário, responsável por criar a interface e controlar as interações visuais. Ele utiliza tecnologias como *HyperText Markup Language* (HTML) para estruturar o conteúdo, *Cascading Style Sheets* (CSS) para estilizar e definir o *layout* e *JavaScript* (JS) para adicionar interatividade e dinamismo. Esse componente é essencial para uma experiência de usuário positiva, especialmente em jogos educacionais, pois a interface precisa ser intuitiva e responsiva para manter o engajamento dos usuários (Ribeiro et al., 2019).

No *frontend*, é crucial oferecer uma interface intuitiva e responsiva, que proporcione uma experiência de usuário agradável e seja acessível em diferentes dispositivos e tamanhos de tela. Elementos interativos e recursos multimídia devem ser integrados de forma harmoniosa para manter o engajamento dos usuários e favorecer o processo de aprendizagem.

Além disso, para a criação de elementos visuais, Flanagan (2020) destaca que o uso de bibliotecas JS, como *Plotly* e *D3.js*, facilita a construção de gráficos dinâmicos que representam dados de maneira interativa e envolvente. Essas ferramentas tornam

a visualização de informações complexas mais acessível e visualmente atraente, melhorando a compreensão dos dados apresentados.

### **2.4.2 Backend**

O *backend* é a parte "invisível" de uma aplicação web, responsável por gerenciar a lógica de negócios, o processamento de dados e a comunicação com o banco de dados. Ele atua como o núcleo funcional da aplicação, recebendo requisições do *frontend*, processando as informações, e devolvendo os dados necessários para exibir resultados, realizar cálculos, autenticar usuários e executar outras ações.

O desenvolvimento de *backend* envolve diversas linguagens de programação, como *Python*, *Node.js*, *Ruby* e *Java*, e depende também de *frameworks* e bibliotecas que ajudam a organizar o código e facilitar o desenvolvimento. Além disso, conforme reforçado por Tianfield (2016), a arquitetura do *backend* deve considerar escalabilidade e segurança. Um *backend* escalável é capaz de suportar um grande número de acessos simultâneos sem perda de desempenho, enquanto um *backend* seguro protege os dados dos usuários contra acessos não autorizados, garantindo a integridade e confidencialidade das informações.

Para o *backend*, *Flask*, surge como uma opção prática e eficiente para criar Interfaces de Programação de Aplicação (API) e controlar o fluxo de dados em aplicações web. Sua leveza e simplicidade tornam *Flask* ideal para aplicações que precisam de rápida resposta e integração com tecnologias web adicionais, como HTML, CSS e JS (Grinberg, 2018).

### **2.4.3 Banco de Dados**

Os bancos de dados são fundamentais para o armazenamento, organização e recuperação de dados em sistemas computacionais. Eles permitem que as informações sejam estruturadas e gerenciadas de maneira eficiente, sendo essenciais em uma ampla variedade de aplicações, desde transações financeiras até plataformas educacionais (Connolly; Begg, 2021).

Conforme discutido por Shi, Wang e Li (2020), o uso de tecnologias de banco de dados é fundamental em jogos educacionais, especialmente devido à necessidade de gerenciar eficientemente os dados e otimizar o desempenho das aplicações. Entretanto, existem diferentes tipos de banco de dados, cada um com características que os tornam mais adequados para determinados contextos. Os bancos de dados relacionais, por exemplo, organizam as informações em tabelas e utilizam SQL para realizar operações de consulta, sendo amplamente utilizados por sua capacidade de manter a integridade dos dados e gerenciar relações complexas entre tabelas (Coronel; Morris, 2016).

Por outro lado, bancos de dados não relacionais, também chamados de *NoSQL*, como *MongoDB* e *Cassandra*, têm se destacado em aplicações que requerem flexibilidade e escalabilidade, especialmente em ambientes onde os dados são semiestruturados ou não estruturados. Esses bancos de dados utilizam estruturas como documentos e grafos, permitindo a manipulação de dados em grandes volumes e a adaptação a mudanças nos modelos de dados sem a necessidade de uma estrutura rígida de tabelas (Elmasri & Navathe, 2016). No contexto de jogos educacionais, onde é necessário armazenar diferentes tipos de dados, como resultados de simulação, histórico de desempenho e perfis de usuário, a escolha do banco de dados adequado impacta diretamente a eficiência e a experiência do usuário (Silberschatz, Korth, & Sudarshan, 2011).

Além disso, Shi, Wang e Li (2020) destacam a importância de otimizar a performance e a escalabilidade do banco de dados para garantir que a aplicação consiga lidar com o crescimento do número de usuários e dados ao longo do tempo, um aspecto fundamental em ambientes educacionais com múltiplos jogadores. Dessa forma, a escolha entre um banco de dados relacional ou *NoSQL*, ou até mesmo uma solução híbrida, deve considerar os requisitos específicos de cada jogo educacional, incluindo o volume de dados, a complexidade das relações e a necessidade de consultas rápidas e eficientes.

O *SQLAlchemy* é uma ferramenta ORM (*Object-Relational Mapping*) que promove a integração entre o código *Python* e o banco de dados relacional. Essa ferramenta permite que os desenvolvedores manipulem dados em um estilo orientado a objetos, o que facilita o armazenamento e recuperação de informações complexas de maneira intuitiva e menos propensa a erros (Skomoroch, Magnusson, 2017).

#### 2.4.4 Transição para Plataformas Web

A migração de uma plataforma *desktop*, como *Access*, para uma aplicação web traz benefícios e desafios específicos. Enquanto as aplicações *desktop* são limitadas aos dispositivos em que estão instaladas e exigem manutenção local, as plataformas web oferecem acessibilidade universal, permitindo que os usuários acessem o jogo a partir de qualquer dispositivo com acesso à internet (Bochicchio, Longo, 2011).

A flexibilidade das plataformas web permite atualizações contínuas, facilitando a correção de bugs e a implementação de novos recursos em tempo real, o que seria complexo em uma aplicação *desktop*. No entanto, segundo Rouse (2017), a mudança para a web também exige uma infraestrutura de *backend* mais segura e escalável para armazenar e gerenciar dados remotamente, um fator que pode impactar o desempenho dependendo da qualidade da conexão do usuário.

Embora as plataformas *desktop* ofereçam processamento rápido localmente, uma aplicação web com um *backend* robusto e design responsivo pode oferecer uma experiência de usuário mais interativa e fluida, ampliando o engajamento no contexto educacional (Bochicchio, Longo, 2011).

### 3 PROCEDIMENTOS METODOLÓGICOS

Serão apresentados neste capítulo a caracterização da pesquisa, as metodologias que serão utilizadas e uma breve descrição do roteiro empregado no trabalho.

Este trabalho utiliza um conjunto de metodologias acadêmicas que integram técnicas de pesquisa aplicada, *redesign* instrucional e reengenharia de jogos. Cada metodologia serve para estruturar o processo de análise, atualização e avaliação do jogo educacional *LSSP\_PCP1*, buscando aprimorar sua eficácia pedagógica e usabilidade. Abaixo estão descritas as metodologias escolhidas e como serão aplicadas neste trabalho, acompanhadas do embasamento teórico.

#### 3.1 CARACTERIZAÇÃO DA PESQUISA

A pesquisa científica pode ser caracterizada de acordo com sua natureza, objetivos e os procedimentos técnicos utilizados na sua elaboração (GIL, 2008). Este trabalho adota uma metodologia que mescla a pesquisa aplicada, pesquisa-ação e uma abordagem mista, com aspectos quantitativos e qualitativos, a fim de desenvolver uma nova versão do jogo instrucional *LSSP\_PCP1*.

Segundo sua natureza, esta pesquisa é aplicada, pois busca gerar conhecimento para aplicação prática, voltado à solução de problemas específicos na área de ensino de PCP, conforme apontado por Prodanov (2013). O objetivo principal é criar uma versão modernizada do jogo que seja mais acessível e eficaz, promovendo um aprendizado aprimorado para estudantes de Engenharia de Produção.

Quanto aos objetivos, esta pesquisa pode ser classificada como descritiva, pois visa analisar as características e funções essenciais para o desenvolvimento do jogo instrucional, com foco na modernização de suas mecânicas, interface e acessibilidade. Esta abordagem permite explorar os aspectos técnicos e educacionais da versão original e compará-los com as atualizações propostas.

Em relação aos procedimentos, a pesquisa se caracteriza como uma pesquisa-ação, uma vez que busca gerar conhecimento aplicável, com o pesquisador participando ativamente no desenvolvimento e teste do jogo junto aos usuários. De acordo com Prodanov (2013), a pesquisa-ação vai além da simples coleta de dados,

envolvendo o pesquisador em um papel ativo para influenciar a realidade dos fenômenos estudados. Neste trabalho, a pesquisa-ação permite que o jogo modernizado seja avaliado em contextos reais de ensino, coletando feedback dos estudantes e adaptando o recurso educacional conforme necessário.

A abordagem deste estudo utiliza-se principalmente, de métodos qualitativos com pouca ênfase em métodos quantitativos. A análise quantitativa concentra-se na coleta de dados objetivos do jogo. Já a abordagem qualitativa é utilizada para interpretar as percepções e feedback dos usuários, oferecendo uma visão mais profunda sobre a experiência de aprendizado.

### 3.2 PESQUISA APLICADA

A pesquisa aplicada é uma abordagem que visa a resolução de problemas específicos e a criação de soluções práticas com impacto direto na realidade dos envolvidos. Segundo Yin (2015), essa metodologia é especialmente adequada para estudos voltados ao desenvolvimento de produtos ou processos, em que a aplicação prática das descobertas é o principal objetivo. Na educação, a pesquisa aplicada permite transformar teorias e conceitos em ferramentas pedagógicas, promovendo a inovação no ensino (GIL, 2008).

Neste trabalho, a pesquisa aplicada fundamenta o desenvolvimento da nova versão do jogo instrucional *LSSP\_PCP1*. Em vez de apenas analisar teoricamente o planejamento e controle da produção (PCP), busca-se uma aplicação prática por meio da modernização do jogo, transformando-o em uma ferramenta educacional mais acessível e eficaz. Essa abordagem permite que o produto final – o jogo modernizado – seja utilizado para aprimorar o ensino de PCP em Engenharia de Produção.

### 3.3 REDESIGN INSTRUCIONAL

O *redesign* instrucional é uma metodologia que envolve a adaptação de recursos educacionais para novos contextos e públicos, mantendo o objetivo original de aprendizagem. De acordo com Reigeluth (1999), o *redesign* instrucional deve garantir que, mesmo com mudanças no formato ou na mídia, a essência pedagógica do recurso seja preservada e aprimorada. Smith e Ragan (2005) destacam que o

*redesign* instrucional frequentemente envolve a modernização de interfaces e a aplicação de teorias de design instrucional e de usabilidade, adaptando o conteúdo para novos suportes tecnológicos.

Para a atualização do jogo *LSSP\_PCP1*, o *redesign* instrucional é utilizado para adaptar a interface e a usabilidade do jogo para o ambiente web, garantindo compatibilidade com dispositivos móveis e computadores. Esse processo envolve manter os princípios de ensino de PCP e as mecânicas de jogo originais, ao mesmo tempo que moderniza a apresentação visual e os controles, tornando o recurso mais intuitivo e acessível. A metodologia de *redesign* instrucional aqui aplicada assegura que a lógica pedagógica do jogo seja mantida, promovendo uma experiência de aprendizado que permanece fiel à versão de 2002.

#### 3.4 REENGENHARIA DE JOGOS

A reengenharia de jogos é uma metodologia frequentemente usada na indústria de jogos, onde se busca reformular jogos antigos para melhor atender às expectativas e necessidades dos usuários contemporâneos. Segundo Collins e Kapralos (2015), a reengenharia de jogos educacionais envolve o aprimoramento de interfaces, funcionalidades e acessibilidade, mantendo a mecânica central do jogo para preservar o conteúdo educacional. Em contextos educacionais, McGonigal (2011) aponta que a reengenharia pode incluir a incorporação de teorias de gamificação, tornando o recurso mais atrativo e envolvente para os alunos.

Nesta modernização do *LSSP\_PCP1*, a reengenharia de jogos é aplicada para reconstruir a interface e a lógica do jogo em uma plataforma compatível com navegadores modernos. As atualizações incluem o uso das linguagens de programação como: *Python*, HTML, CSS e JS, garantindo que o jogo funcione tanto em computadores quanto em dispositivos móveis. A reengenharia aqui também considera a experiência do usuário, permitindo que os alunos possam acessar e utilizar o jogo sem restrições de software, como o *Access 2003*, presente na versão original.

### 3.5 ROTEIRO METODOLÓGICO

Este estudo se desenvolve utilizando um modelo de pesquisa aplicada e de *redesign* instrucional, com foco na remasterização do jogo *LSSP\_PCP1* como ferramenta de apoio educacional. A metodologia é dividida em três etapas principais, conforme detalhado a seguir:

#### 3.5.1 Análise da Versão Original do Jogo LSSP\_PCP1

Nesta fase, foi realizada uma análise do jogo *LSSP\_PCP1*, criado em 2002 pelo Laboratório de Simulação de Sistemas de Produção (LSSP) da UFSC, com base nas diretrizes do livro *Planejamento e Controle da Produção - Teoria e Prática* de Dálvio Ferrari Tubino. Esta análise inclui:

1. **Contextualização:** Revisão dos objetivos educacionais e dos fundamentos do PCP implementados no jogo.
2. **Descrição Funcional e Educacional:** Documentação das principais mecânicas de jogo, como previsões de demanda, planejamento estratégico de produção e avaliação financeira. Dados das telas e formulários operacionais são apresentados para ilustrar o processo original.
3. **Identificação de Limitações Tecnológicas:** Análise das barreiras da versão original, como a limitação ao *software* Access 2003 e a necessidade de atualização para compatibilidade com dispositivos modernos (PC e celular).

#### 3.5.2 Desenvolvimento da Nova Versão e Mudanças Propostas

Com base nas limitações identificadas, são introduzidas melhorias de design e usabilidade na nova versão do jogo, utilizando a metodologia de reengenharia de jogos:

1. **Comparação por Etapas (Interface, Funcionalidade e Acessibilidade):** Implementação de um ambiente web responsivo permitindo o acesso via navegador. Cada seção do jogo é documentada em comparação com a versão anterior, abordando as mudanças na interface.

2. **Utilização de Novas Tecnologias:** Implementação de *frameworks* e tecnologias modernas com foco em acessibilidade e compatibilidade multiplataforma, melhorando a experiência do usuário e garantindo que o jogo funcione em diferentes dispositivos.

### 3.5.3 Avaliação Comparativa e Validação

Após a implementação, a nova versão do jogo é submetida a testes internos para garantir sua funcionalidade e avaliar o potencial de impacto educacional:

1. **Análise das Melhorias Implementadas:** A eficácia da versão modernizada é avaliada com base em uma análise comparativa com a versão original, focando em aspectos como usabilidade e facilidade de navegação. As melhorias são examinadas em termos de acessibilidade e intuitividade da interface, a fim de verificar se atendem aos objetivos propostos.
2. **Avaliação das Funcionalidades e Usabilidade:** O impacto das mudanças é verificado por meio de testes internos, onde são analisados elementos como a clareza das instruções, a lógica de navegação e o tempo de resposta do sistema.

## 4 DESENVOLVIMENTO

O presente capítulo se propõe a detalhar o processo de desenvolvimento do jogo, desde a concepção até a implementação final. O desenvolvimento foi dividido em etapas abrangendo a análise do jogo atual, a identificação de melhorias, a contextualização e das ferramentas utilizadas na programação e desenvolvimento.

Considerando que o trabalho consiste no desenvolvimento de uma aplicação web, que possui diversos arquivos e elementos para funcionar, todo o código fonte pode ser encontrado no repositório do *GitHub* presente nas referências deste trabalho.

### 4.1 ANÁLISE DO JOGO ATUAL

O jogo *LSSP\_PCP1* é parte da série de jogos educacionais *LSSP\_PCP*, desenvolvidos pelo LSSP da UFSC, e tem como objetivo explorar e debater as características da previsão de demanda e do planejamento estratégico da produção.

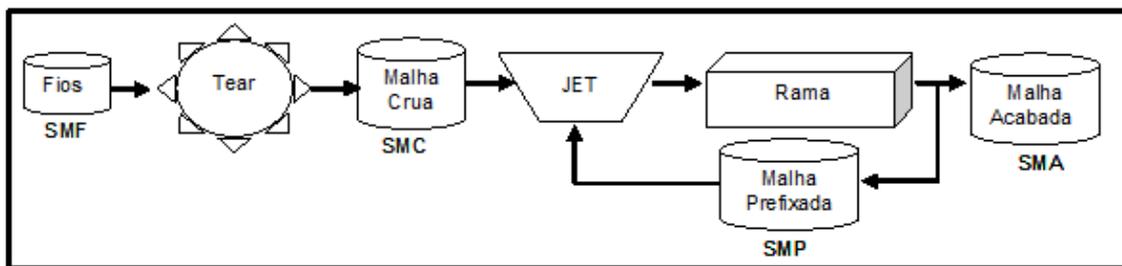
Desenvolvido com foco educacional, o jogo é uma ferramenta prática para o ensino de (PCP). Ele permite que os participantes compreendam e apliquem conceitos teóricos em situações simuladas, promovendo uma experiência interativa que reforça o aprendizado por meio de tomada de decisões estratégicas e análise de resultados. Ele simula um ambiente produtivo de longo prazo, abrangendo 12 períodos mensais, do mês 13 ao mês 24 de operação. O jogo foca na fabricação de malhas, com três famílias de produtos: Colméia, Piquet e Maxim. A dinâmica do jogo envolve a previsão de demanda, planejamento de produção, cálculo de capacidade e gestão de recursos, permitindo ao grupo tomar decisões estratégicas e avaliar seus impactos financeiros e operacionais.

O principal objetivo do *LSSP\_PCP1* é proporcionar uma experiência prática na aplicação dos conceitos de PCP. Os participantes são desafiados a desenvolver habilidades em previsão de demanda, alocação de recursos e gerenciamento de estoques, enfrentando situações típicas de uma empresa do setor têxtil. A simulação permite que os participantes testem diferentes estratégias de produção e adaptação de capacidade, considerando três tipos de mercados: de massa, repetitivo em lotes e sob encomenda, cada um com características e desafios específicos.

Conforme ilustrado na Figura 3, a empresa adquire fios (de algodão e sintéticos) no mercado, que são processados na etapa de tecelagem, resultando em malhas cruas das famílias Colméia, Piquet e Maxim, produzidas em teares circulares. Posteriormente, essas malhas cruas passam por um processo técnico denominado prefixação. Esse processo inclui uma lavagem nos jets, chamada de purga, seguida pela fixação na rama, originando as malhas fixadas (Colméia, Piquet e Maxim).

Na etapa seguinte, as malhas fixadas são submetidas novamente ao fluxo produtivo jet-rama, onde recebem a adição de corantes, também adquiridos no mercado, para serem tingidas nos jets e finalizarem o acabamento na rama. Os fios, as malhas cruas, as malhas fixadas e as malhas acabadas são armazenadas em supermercados específicos, destinados a cada estágio do processo produtivo

Figura 3: Processo Produtivo



Fonte: Manual Operacional *LSSP\_PCP1* (2007)

Dado o horizonte de longo prazo do jogo, o objetivo é montar um plano estratégico de produção para atender a determinada previsão de demanda, estruturando os recursos físicos da empresa.

Nas próximas subseções será discutido sobre o funcionamento do fluxo do jogo e as etapas necessárias para sua execução. As informações apresentadas são baseadas no *Manual Operacional do LSSP\_PCP1* além de no próprio jogo em si.

#### 4.1.1 Fluxo do Jogo

Ao iniciar o jogo pela primeira vez, os jogadores se deparam com a tela inicial, conforme ilustrado na Figura 4.

Figura 4: Tela Inicial



Fonte: Jogo LSSP\_PCP1 (2024)

A primeira etapa necessária é o cadastro do grupo, e após essa etapa, o jogo começa, com a previsão de demanda das três famílias de malhas (Colméia, Piquet e Maxim) para os próximos 12 meses. Essa previsão é essencial para planejar a produção e a capacidade produtiva ao longo dos períodos, considerando possíveis ajustes de recursos, como ampliações ou reduções, e seus impactos financeiros. Durante o jogo, os grupos podem acompanhar o desempenho das previsões e o erro acumulado ao longo dos períodos.

Em seguida, é feito o planejamento de produção para cada uma das famílias de malhas, definindo as quantidades a serem produzidas ao longo dos 12 meses de simulação. O plano de produção impacta diretamente os estoques e as vendas, sendo automaticamente ajustado com base na demanda e produção reais a cada período simulado. Conforme os meses avançam, os períodos passados são bloqueados, e as simulações consideram as variações aleatórias que ocorrem na produção e demanda reais.

O plano de produção também gera a necessidade de matérias-primas, que se traduz no plano de compras de insumos (Fio de Algodão, Fio Sintético e Corantes). Neste plano, os grupos definem as quantidades de compras para cada mês, considerando o consumo previsto, estoque atual e possíveis compras de emergência

para cobrir déficits de matéria-prima. Cada simulação atualiza os estoques e substitui o consumo previsto pelo consumo real.

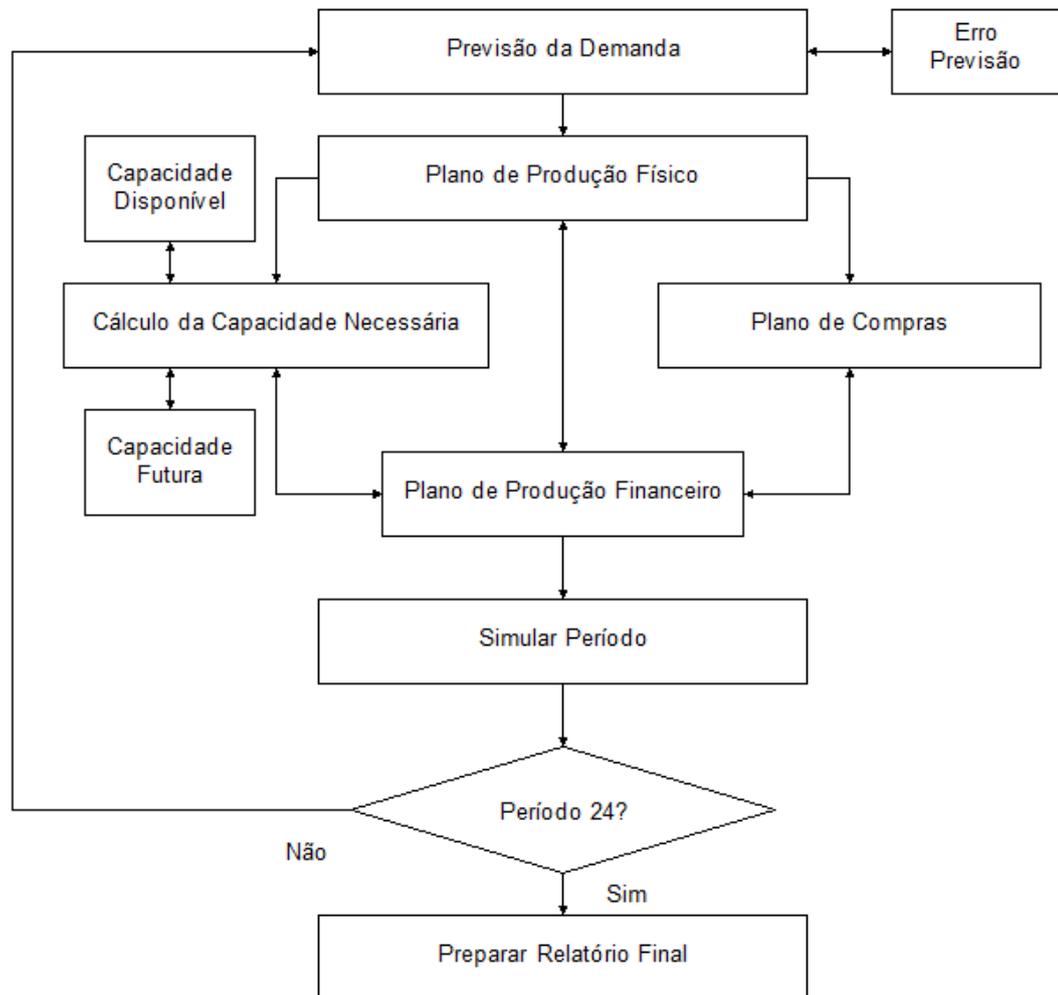
Além disso, o planejamento de produção define a capacidade necessária nos três setores produtivos: tecelagem, purga/tinturaria e fixação/acabamento. Para que o plano estratégico seja viável, a capacidade disponível deve ser suficiente em cada setor. Os grupos podem ajustar a capacidade por meio de turnos adicionais, terceirização ou mudanças na capacidade futura para atender às demandas produtivas.

Por fim, os reflexos financeiros de cada decisão tomada são exibidos em relatórios detalhados de custos e receitas, permitindo aos grupos analisar o impacto financeiro ao longo dos 12 períodos. Após completar todas essas etapas, o jogo está pronto para a simulação, onde então são simuladas demandas e produções reais, vendas, compras de equipamentos, faltas de matéria-prima e outras variáveis operacionais.

Ao final dos 12 períodos simulados (do mês 13 ao 24), o jogo é concluído, e os relatórios finais de desempenho são usados para analisar as decisões tomadas durante a simulação.

A Figura 5 a seguir ilustra o esquema do fluxo do jogo descrito anteriormente.

Figura 5: Dinâmica de PCP do Jogo



Fonte: Manual Operacional *LSSP\_PCP1* (2008)

Conforme o exposto, esse fluxo detalhado do jogo oferece aos grupos uma experiência prática de tomada de decisões estratégicas e operacionais, permitindo que testem diferentes abordagens para o Planejamento e Controle da Produção. Cada etapa do jogo proporciona um *feedback* imediato sobre os impactos das escolhas feitas, facilitando o aprendizado dos conceitos aplicados ao longo da simulação. Ao final, a análise dos relatórios finais permite que os grupos reflitam sobre o desempenho geral e considerem possíveis melhorias para aprimorar suas estratégias em simulações futuras.

## 4.1.2 Principais Funcionalidades

O *LSSP\_PCP1* possui diversas funcionalidades que suportam o planejamento estratégico e a programação da produção. Abaixo, cada funcionalidade é detalhada com mais profundidade, acompanhada de imagens ilustrativas que demonstram o funcionamento do jogo.

### 4.1.2.1 Cadastro do Grupo

Antes de iniciar a simulação, os jogadores precisam realizar o cadastro do grupo de participantes. Esta etapa permite que seja armazenado as informações sobre o grupo e configure os parâmetros iniciais do jogo. O cadastro é simples e envolve a inserção de dados básicos dos participantes, como nome, e-mail, identificação do grupo, demanda e porte da empresa, como pode ser observado na Figura 6.

Figura 6: Tela Cadastro

Cadastro

Nome da Empresa

Analista 1

Analista 2

Analista 3

Analista 4

Analista 5

Escolha o Tipo de Demanda

Escolha o Tamanho da Empresa

Fonte: Jogo *LSSP\_PCP1* (2024)

Essas informações além de configurarem os parâmetros iniciais do jogo, dependendo do tipo de demanda e do tamanho da empresa selecionados, servem

também para identificar o grupo posteriormente, quando o arquivo for enviado para avaliação.

#### 4.1.2.2 *Previsão de Demanda*

Os grupos devem prever a demanda das três famílias de malhas ao longo dos 12 períodos de simulação. Para isso, é preciso considerar tendências, sazonalidades e variações aleatórias, com o objetivo de planejar a capacidade e as compras de matéria-prima de forma eficaz. Essa etapa inicial é essencial, pois as previsões definem a base para o planejamento de produção e gestão de estoques subsequente.

O jogo não conta com um mecanismo próprio para realizar as previsões diretamente na plataforma. No entanto, ele disponibiliza um histórico com a demanda real dos períodos 1 a 12 para cada família de malhas. Com base nesses dados, os grupos podem criar suas próprias fórmulas de previsão, utilizando séries temporais ou outras metodologias que considerarem mais adequadas. Após calcular as previsões, os grupos podem inserir os resultados na tela de previsão de demanda.

Na Figura 7 a seguir, é apresentada a tela onde as previsões de demanda dos períodos futuros (13 a 24) para cada família de malhas devem ser inseridas.

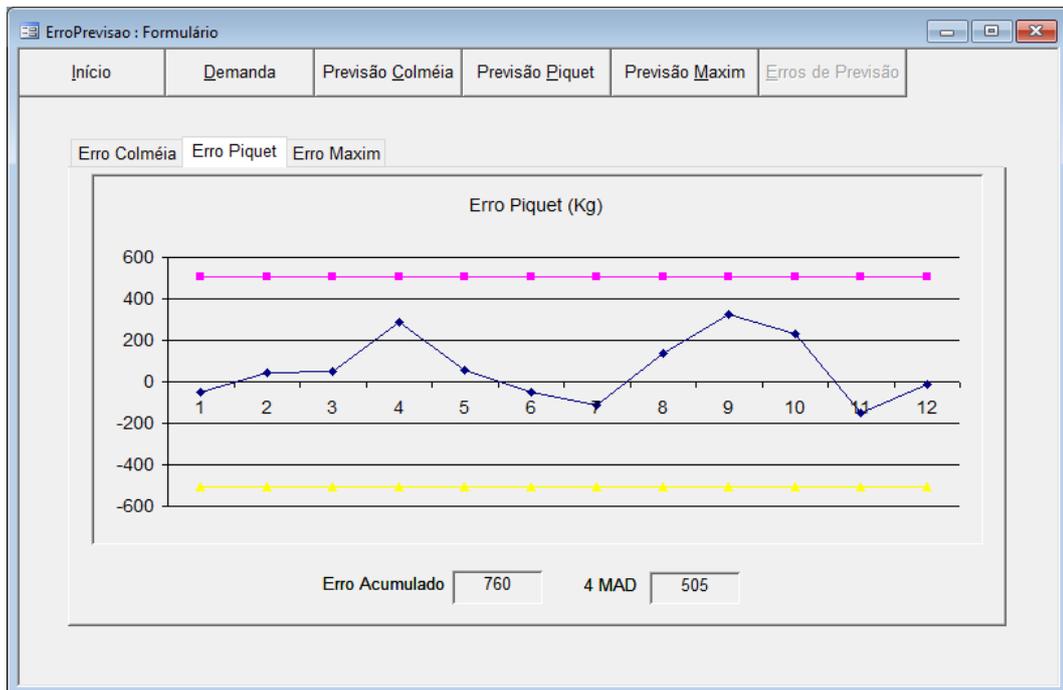
Figura 7: Tela do Formulário Previsão Colméia

Período	13	14	15	16	17	18	19	20	21	22	23	24
Previsão	3.000	3.000	3.000	3.000	4.000	3.000	3.400	3.300	3.600	4.000	3.700	4.200
Real												

Fonte: Jogo LSSP\_PCP1 (2024)

Além dessa tela, há também a tela de Erros de Previsão, ilustrada na Figura 8, onde é apresentado um gráfico de controle que monitora os erros de previsão para cada família de malhas. À medida que as simulações são realizadas, o gráfico é atualizado com as novas informações dos períodos simulados. No entanto, observa-se que, na versão atual do jogo representada na imagem, o gráfico não exibe os dados adequadamente, indicando uma limitação que foi identificada como uma oportunidade de melhoria. Essa questão foi abordada na nova implementação, garantindo que os erros de previsão sejam apresentados de forma clara e interativa, permitindo uma análise mais eficiente dos resultados.

Figura 8: Tela do Formulário Erros de Previsão



Fonte: Jogo *LSSP\_PCP1* (2024)

Os gráficos possuem limites inferiores e superiores definidos como quatro vezes o valor do Desvio Médio Absoluto (MAD). De acordo com o manual, esses limites são estabelecidos com base no erro aleatório gerado pelo próprio jogo. Dessa forma, qualquer valor que ultrapasse a faixa de controle indica erros significativos nas previsões feitas pelo grupo, sugerindo a necessidade de ajustes nas estratégias de previsão.

#### 4.1.2.3 Planejamento de Produção

Nesta tela, os grupos precisam elaborar planos de produção para cada uma das famílias de malhas. Eles devem definir a quantidade de produção por período, de forma a atender à demanda prevista. A gestão de estoques é uma parte crítica deste processo, pois a produção insuficiente pode resultar em vendas não atendidas, enquanto o excesso de produção pode gerar custos adicionais de estoque. A tela do plano de produção é ilustrada na Figura 9 abaixo, onde os grupos podem inserir as projeções de demanda para cada família de malhas.

Figura 9: Tela do Formulário Plano de Produção Colméia

Período	13	14	15	16	17	18	19	20	21	22	23	24
Demanda Prevista	3.000	3.000	3.000	3.000	3.500	3.000	3.400	3.300	3.600	4.000	3.700	4.200
Demanda Real	0	0	0	0	0	0	0	0	0	0	0	0
Estoques Iniciais	1.000	1.000	0	1.000	1.500	1.500	0	0	0	0	0	0
Produção Planejada	3.000	2.000	4.000	3.500	3.500	0	0	0	0	0	0	0
Produção Real	0	0	0	0	0	0	0	0	0	0	0	0
Estoques Finais	1.000	0	1.000	1.500	1.500	-1.500	-3.400	-3.300	-3.600	-4.000	-3.700	-4.200
Vendas Perdidas	0	0	0	0	0	1.500	3.400	3.300	3.600	4.000	3.700	4.200
Vendas	3.000	3.000	3.000	3.000	3.500	1.500	0	0	0	0	0	0

Fonte: Jogo *LSSP\_PCP1* (2024)

O único campo que o grupo pode preencher é o de Produção Planejada, enquanto os valores de Demanda Prevista são automaticamente preenchidos com base nas informações inseridas anteriormente na tela de previsão de demanda. À medida que o grupo insere os valores de produção planejada, os demais campos, exceto Demanda Real e Produção Real, são calculados automaticamente para refletir a produção planejada e a demanda prevista.

Os valores de Demanda Real e Produção Real são atualizados somente após a simulação do período e possuem uma variação aleatória. A Produção Real possui

uma taxa de desempenho de 95% da Produção Planejada, fazendo com que o valor varie de forma aleatória  $\pm 5\%$  em relação a produção planejada, o que introduz um elemento de imprevisibilidade que simula condições reais de desempenho produtivo. Após essa atualização, os demais campos são recalculados para refletir os valores reais, garantindo que todas as informações estejam alinhadas com os dados mais recentes da simulação.

A lógica dos cálculos é baseada em uma série de interações entre as variáveis principais. Para atender à demanda em cada período, o jogo utiliza os Estoques Iniciais do período, somados à Produção Planejada ou Produção Real (dependendo de se o período foi simulado ou não). A quantidade de malha vendida no período, representada pelo campo Vendas, é calculada com base no valor da Demanda Prevista ou Real, novamente dependendo se o período já foi simulado ou não, e o estoque total (Estoques Iniciais + Produção). Os Estoques Finais são então calculados subtraindo as vendas realizadas do estoque total. Esses estoques finais são transferidos como estoques iniciais para o período seguinte.

Caso os estoques iniciais somados à produção do período não sejam suficientes para atender toda a demanda, surgem Vendas Perdidas, e os estoques finais são zerados, o que reflete um impacto direto nas decisões e na estratégia de planejamento do grupo. Essa lógica de cálculo assegura que cada decisão tomada influencie os resultados subsequentes.

#### 4.1.2.4 *Plano de Compras*

A tela Plano de Compras, ilustrada na Figura 10, é projetada para permitir que os grupos planejem a aquisição de matérias-primas de maneira estratégica. Com base nas previsões de demanda e no planejamento de produção, os grupos precisam definir a quantidade de matérias-primas necessárias para atender à produção planejada, garantindo um nível adequado de estoques sem comprometer o fluxo produtivo ou incorrer em custos excessivos de armazenamento.

O plano é construído a partir do Consumo Previsto para períodos futuros ou do Consumo Real para períodos já simulados. Esses valores são gerados automaticamente com base nos planos de produção das três famílias de malhas, considerando a proporção específica de fios de algodão, fios sintéticos e a adição de

corantes em cada família. Enquanto um período ainda não foi simulado, o consumo previsto é utilizado para estimar as necessidades físicas e financeiras de compras. À medida que os períodos são simulados, as decisões se tornam fixas, e o consumo real substitui o consumo previsto.

Figura 10: Tela do Formulário Plano de Compras Fio 1

Plano de Compras - Fio 1 (Kg)		Período Atual											
Período		13	14	15	16	17	18	19	20	21	22	23	24
Consumo Previsto		5.000	3.500	5.500	5.000	5.000	6.500	1.500	4.500	4.500	5.400	5.500	6.500
Consumo Real		0	0	0	0	0	0	0	0	0	0	0	0
Estoques Iniciais		1.000	0	0	0	1.000	0	0	3.000	3.000	0	0	0
Compras Planejadas		4.000	3.500	5.500	6.000	4.000	6.500	4.500	4.500	1.500	5.400	5.500	6.500
Compras Reais		0	0	0	0	0	0	0	0	0	0	0	0
Compras Emergên.		0	0	0	0	0	0	0	0	0	0	0	0
Estoques Finais		0	0	0	1.000	0	0	3.000	3.000	0	0	0	0

Fonte: Jogo *LSSP\_PCP1* (2024)

Para garantir que haja matéria-prima suficiente para atender ao consumo dos planos de produção em cada período, são considerados os Estoques Iniciais do período, somados às Compras Planejadas, que são definidas pelo grupo. Assim como acontece com o consumo, as compras planejadas são substituídas pelas Compras Reais geradas pelo jogo, que têm um desempenho de 90% para fornecedores de fios e 85% para fornecedores de corantes, o que introduz uma variação realista e simula a incerteza no fornecimento.

Os Estoques Finais de matéria-prima são calculados com base no consumo (previsto ou real), nos estoques iniciais e nas compras (planejadas ou reais). Estes estoques finais são então transferidos como estoques iniciais para o período seguinte. Se, em algum momento, os estoques iniciais e as compras do período não forem

suficientes para atender ao consumo, o jogo automaticamente realiza Compras de Emergência para conseguir atender a produção.

#### 4.1.2.5 *Cálculos de Capacidades*

Os jogadores têm a capacidade de ajustar a produção nos três setores: tecelagem, purga/tinturaria e fixação/acabamento. O jogo permite ajustes de capacidade tanto para o momento atual quanto para períodos futuros, incluindo o aumento ou redução de turnos e a utilização de terceirização, se necessário. Essas funcionalidades são necessárias para garantir que a capacidade produtiva seja compatível com a demanda projetada e a estratégia de produção definida.

No setor de tecelagem, por exemplo, existem três telas principais que facilitam o gerenciamento da capacidade: Capacidade Necessária, Capacidade Disponível e Capacidade Futura. A tela de Capacidade Necessária, conforme ilustrado na Figura 11, apresenta a necessidade de teares em horas, calculada com base no plano de produção estratégico das famílias Colméia, Piquet e Maxim. O cálculo considera, além da capacidade em horas para atender ao plano de produção, o tempo médio de setup, estimado em quatro setups de 0,25 horas por mês por tear, e uma perda de produtividade de 10%. Para que o plano seja viável, a capacidade disponível precisa ser maior ou igual à capacidade necessária, o que é indicado pelo campo de validação na tela.

Figura 11: Tela Formulário Capacidade Necessária Tecelagem

PPTecelagem : Formulário

Tecelagem - Capacidade em Horas      Período Atual: 12

Período	13	14	15	16	17	18	19	20	21	22	23	24
Cap. Disponível	1.400	1.400	1.120	1.120	1.120	1.680	1.120	1.120	1.120	1.120	1.120	1.680
Cap. Necessária	984	896	1.093	1.043	1.043	1.192	697	994	994	1.083	1.093	1.192
Colméia	360	180	360	315	315	450	0	270	270	351	360	450
Piquet	200	300	300	300	300	300	300	300	300	300	300	300
Maxim	330	330	330	330	330	330	330	330	330	330	330	330
Setup *	5	5	4	4	4	4	4	4	4	4	4	4
Produtividade (10%)	89	81	99	94	94	108	63	90	90	98	99	108
Validação	SIM											

\* Setup = Num. Teares x 0,25 horas/setup x 4 setup/mês/tear

Fonte: Jogo *LSSP\_PCP1* (2024)

Para ajustar a Capacidade Disponível, os jogadores podem acessar a tela de Cap. Disponível Tecelagem, ilustrada na Figura 12, e fazer alterações no número de turnos de trabalho dos teares ou contratar serviços terceirizados. A capacidade instalada é ajustada selecionando 1, 2 ou 3 turnos, com cada turno representando 7 horas de trabalho em 20 dias úteis por mês. Além disso, os jogadores podem adicionar capacidade terceirizada em múltiplos de 420 horas, o equivalente a alugar teares por três turnos durante um mês inteiro. É importante notar que, uma vez simulados, os períodos passados ficam bloqueados e não podem ser alterados.

Figura 12: Tela Formulário Capacidade Disponível Tecelagem

PP TecelagemDisponível : Formulário

Início Cap. Necessária Cap. Disponível **Cap. Futura**

Tecelagem - Capacidade em Horas Período Atual 12

Período	13	14	15	16	17	18	19	20	21	22	23	24
Cap. Disponível	1.400	1.400	1.120	1.120	1.120	1.680	1.120	1.120	1.120	1.120	1.120	1.680
Cap. Instalada *	1.400	1.400	1.120	1.120	1.120	1.680	1.120	1.120	1.120	1.120	1.120	1.680
1 Turno	1	1	1	1	1	1	1	1	1	1	1	1
2 Turnos	2	2	2	2	2	2	2	2	2	2	2	2
3 Turnos	3	3	3	3	3	3	3	3	3	3	3	3
Cap. Terceirizada **	0	0	0	0	0	0	0	0	0	0	0	0

\* Cap. Instalada = Num. Teares x 7 horas/turno x num. turnos/dia x 20 dias/mês

\*\* Cap. Terceirizada em múltiplos de 420 horas = 1 tear x 7 horas/turno x 3 turnos/dia x 20 dias/mês

Fonte: Jogo LSSP\_PCP1 (2024)

A tela de Capacidade Futura, ilustrada na Figura 13, permite aos jogadores planejar ampliações ou reduções de recursos no setor de tecelagem. As ampliações ou reduções são realizadas preenchendo os campos correspondentes, e essas mudanças só têm efeito após o tempo de *lead time* definido. Para garantir a integridade da simulação, as reduções nos dois últimos períodos são bloqueadas pelo jogo, evitando a venda de ativos apenas para melhorar artificialmente o resultado operacional. Essa funcionalidade permite um planejamento estratégico mais eficaz, alinhando a capacidade produtiva futura com as necessidades previstas do jogo.

Figura 13: Tela Formulário Capacidade Futura Tecelagem

PP Tecelagem Ampliaco es : Formulário

Início    Cap. Necessária    Cap. Disponível    Cap. Futura

Tecelagem - Número de Teares    Período Atual    12

Período	13	14	15	16	17	18	19	20	21	22	23	24
Num. Teares	5	5	4	4	4	4	4	4	4	4	4	4
Ampliações *	0	0	0	0	0	0	0	0	0	0	0	0
Reduções **	1	0	0	0	0	0	0	0	0	0	0	0
Cap. Inst. Turno	700	700	560	560	560	560	560	560	560	560	560	560

\* Lead Time Ampliações = 2 meses

\*\* Lead Time Reduções = 2 meses

Fonte: Jogo *LSSP\_PCP1* (2024)

Para o setor de fixação/acabamento, que compreende as ramas, o processo de ajuste de capacidade segue a mesma lógica descrita para a tecelagem, com algumas variações nos parâmetros específicos. A principal diferença está nos valores de lead time para as ampliações e reduções de capacidade, que agora é de 3 meses, o que significa que qualquer mudança planejada nesse setor só terá efeito após esse período. O tempo médio de setup para as ramas corresponde a 10% da capacidade necessária para as famílias, enquanto a perda de produtividade é de 15%.

Além disso, para adicionar capacidade terceirizada no setor de fixação/acabamento, os jogadores devem inserir valores em múltiplos de 70 horas. Esse valor corresponde ao aluguel de ramas por 1 turno por dia durante meio mês, o que oferece uma flexibilidade diferente em comparação com a terceirização de teares. Assim, os jogadores precisam considerar esses ajustes cuidadosamente para garantir que a capacidade disponível no setor de fixação/acabamento seja suficiente para atender à demanda projetada, sem comprometer a eficiência ou gerar custos desnecessários.

Por fim, considerando o setor de purga/tinturaria, que utiliza os jets, o processo de ajuste de capacidade segue uma lógica semelhante à dos outros setores, mas com algumas particularidades importantes. Na tela de Capacidade Necessária, o cálculo

da necessidade de capacidade leva em conta o tempo de setup, que varia dependendo do número de cores processadas por família: o setup corresponde a 5% da capacidade total se houver apenas 1 cor por família, 20% se houver 3 cores por família e 35% se houver 10 cores por família, sendo essas definidas quando selecionado o tipo de demanda no começo do jogo. Além disso, a perda de produtividade para os jets é fixada em 10%, refletindo as especificidades operacionais desse setor.

Na tela de Capacidade Disponível para os jets, apresentada na Figura 14, a lógica também é análoga às demais, com algumas diferenças relacionadas aos três tipos de jets que podem ser utilizados: jets de 30, 120 ou 480 quilos. A capacidade disponível é calculada como a soma da capacidade instalada para cada tamanho de jet, mais a capacidade terceirizada. A capacidade instalada pode ser ajustada selecionando 1, 2 ou 3 turnos, afetando todos os jets independentemente do tamanho. A capacidade instalada é calculada com base no número de jets de cada tamanho, no número de turnos escolhido, e em uma carga média de 4,5 horas entre os processos de purga e tinturaria.

Figura 14: Tela Formulário Capacidade Disponível Purga/Tinturaria

Purga / Tinturaria - Capacidade em Kg												Período Atual	
Período	13	14	15	16	17	18	19	20	21	22	23	24	12
Cap. Disponível	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400
Cap. Instalada Jet_1	0	0	0	0	0	0	0	0	0	0	0	0	0
Cap. Instalada Jet_2	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400	22.400
Cap. Instalada Jet_3	0	0	0	0	0	0	0	0	0	0	0	0	0
1 Turno	1	1	1	1	1	1	1	1	1	1	1	1	1
2 Turnos	2	2	2	2	2	2	2	2	2	2	2	2	2
3 Turnos	3	3	3	3	3	3	3	3	3	3	3	3	3
Cap. Terceirizada **	0	0	0	0	0	0	0	0	0	0	0	0	0

\* Cap. Instalada = [(Num. turnos/dia x 7 horas/turno x 20 dias/mês) / 4,5 horas/carga] x Num. Jet x Kg/Jet  
 \*\* Cap. Terceirizada em múltiplos de 11.200 Kg = [(3 turnos/dia x 7 horas/turno x 20 dias/mês) / 4,5 horas/carga] x 1 Jet x 120 Kg/Jet

Fonte: Jogo LSSP\_PCP1 (2024)

Para adicionar capacidade terceirizada, os jogadores devem preencher os campos de Capacidade Terceirizada com valores em múltiplos de 11.200 quilos. Esse valor corresponde ao aluguel de jets de 120 quilos por 3 turnos durante um mês inteiro.

De forma similar aos demais setores, os jogadores podem modificar a capacidade instalada para períodos futuros por meio de ampliações ou reduções dos diferentes tipos de jets no setor de purga e tinturaria. Isso é feito na tela de Capacidade Futura, conforme ilustrado na Figura 15 abaixo, onde os campos de Ampliações e Reduções podem ser preenchidos com o número de jets desejado para cada tipo (30, 120 ou 480 quilos). Sendo que para esse setor, uma ampliação de capacidade só terá efeito após três meses, enquanto uma redução se concretiza após dois meses. Assim como nos outros setores, os períodos passados ficam bloqueados após serem simulados, o que impede alterações retroativas.

Figura 15: Tela Formulário Capacidade Futura Purga/Tinturaria

Período	13	14	15	16	17	18	19	20	21	22	23	24
Num. Jet_1 480 Kg	0	0	0	0	0	0	0	0	0	0	0	0
Ampliações	0	0	0	0	0	0	0	0	0	0	0	0
Reduções	0	0	0	0	0	0	0	0	0	0	0	0
Num. Jet_2 120 Kg	3	3	3	3	3	3	3	3	3	3	3	3
Ampliações	0	0	0	0	0	0	0	0	0	0	0	0
Reduções	0	0	0	0	0	0	0	0	0	0	0	0
Num. Jet_3 30 Kg	0	0	0	0	0	0	0	0	0	0	0	0
Ampliações	0	0	0	0	0	0	0	0	0	0	0	0
Reduções	0	0	0	0	0	0	0	0	0	0	0	0
Cap. Inst. Turno	11.200	11.200	11.200	11.200	11.200	11.200	11.200	11.200	11.200	11.200	11.200	11.200

\* Lead Time Ampliações = 3 meses      \*\* Lead Time Reduções = 2 meses

Fonte: Jogo LSSP\_PCP1 (2024)

Essas diferenças específicas para os jets exigem que os jogadores façam um planejamento cuidadoso, considerando o impacto do número de cores no tempo de setup e gerenciando eficientemente os diferentes tamanhos de jets disponíveis.

#### 4.1.2.6 Plano de Produção Financeiros

O jogo gera Planos de Produção Financeiros detalhados que permitem aos jogadores avaliar o desempenho das estratégias adotadas. Esses planos incluem informações abrangentes sobre custos fixos, custos de matéria-prima, receitas de vendas, custos relacionados à capacidade produtiva e custos decorrentes de vendas perdidas. A visualização clara desses resultados financeiros é crucial para que os jogadores compreendam o impacto de suas decisões de planejamento e ajustes de capacidade, ajudando-os a refletir sobre a eficiência das estratégias implementadas.

Após desenvolver todo o planejamento estratégico da produção, incluindo a previsão de demandas, a criação dos planos de produção e compras, e o balanceamento da capacidade produtiva da fábrica, os jogadores podem acessar a tela do Plano de Produção Financeiro, ilustrada na Figura 16. O acesso a essa tela é feito a partir da tela principal do jogo, já apresentada na Figura 3, clicando no botão P. Produção e, em seguida, no botão PP Financeiro. Nessa tela, os custos e receitas são apresentados de forma detalhada por período: para os períodos já simulados, esses valores são fixos, enquanto para os períodos futuros, eles são projetados com base no planejamento realizado.

Figura 16: Tela Formulário Plano de Produção Financeiro

Plano de Produção Financeiro - \$												Período Atual	
Período	13	14	15	16	17	18	19	20	21	22	23	24	12
Custos Fixos	31.933	29.133	27.667	27.667	27.667	30.467	27.667	27.667	27.667	27.667	27.667	30.467	
C. Compras MP	10.100	13.850	16.250	16.550	16.750	15.450	17.250	11.250	11.850	16.150	16.250	17.810	
Custos Estoques	850	613	613	806	918	1.118	1.070	729	463	310	333	461	
C. Terceirização	0	0	0	0	0	0	0	0	0	0	0	0	
Custos Capital	10.110	9.968	8.168	8.284	8.351	8.471	8.442	8.237	8.078	7.986	8.000	8.077	
Custos V. Perdas	0	0	0	0	0	0	0	0	0	0	0	0	
Custos Totais	52.993	53.563	52.697	53.307	53.685	55.505	54.429	47.883	48.057	52.112	52.250	56.814	
Receita Vendas	64.500	58.500	58.500	58.500	61.250	58.500	60.700	60.150	61.800	64.000	62.350	65.100	
Resul. Operacional	11.507	4.937	5.803	5.193	7.565	2.995	6.271	12.267	13.743	11.888	10.100	8.286	
R. Op. Acumulado	11.507	16.443	22.247	27.440	35.006	38.001	44.272	56.540	70.283	82.171	92.271	100.557	
Rec. Vendas	Custos Fixos		C. Compras MP		Custos Estoques		C. Terceirização		Custos Capital		Custos V. Perd.		

Fonte: Jogo LSSP\_PCP1 (2024)

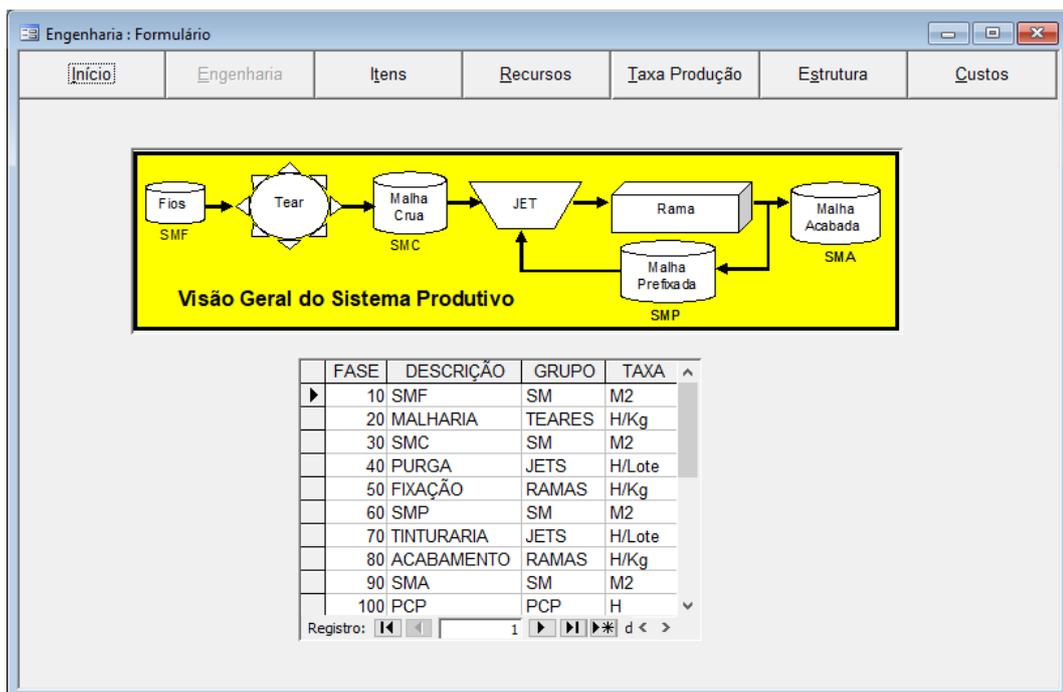
Além disso, na parte inferior da tela do Plano de Produção Financeiro, há botões que levam a detalhes específicos de cada componente de custos e receitas, sendo eles Receita de Vendas, Custos Fixos, Custos Compras Matéria Prima, Custos de Estoques, Custos de Terceirização, Custos Capital, e Custos de Venda Perdida.

Nessas telas, as fórmulas de cálculo são explicadas, permitindo que os jogadores compreendam melhor como suas decisões influenciam o desempenho financeiro da fábrica e facilitando uma análise mais precisa das implicações de suas estratégias.

#### 4.1.2.7 Engenharia

A tela da Engenharia no *LSSP\_PCP1*, ilustrada na Figura 17, permite que os jogadores visualizem informações técnicas detalhadas sobre o processo produtivo, servindo como uma referência essencial para as decisões de planejamento e produção. Neste módulo, os jogadores não podem modificar parâmetros ou ajustar configurações, mas têm acesso a dados fundamentais que orientam outras decisões estratégicas no jogo.

Figura 17: Tela do Formulário Engenharia



Fonte: Jogo *LSSP\_PCP1* (2024)

A partir desta tela, os jogadores podem acessar, por meio de botões específicos, informações sobre os itens envolvidos na produção das malhas, os recursos produtivos, as taxas de produção, a estrutura (ou árvore) das malhas, e o sistema de custos para avaliação das decisões tomadas ao longo do jogo.

Na tela Itens, exibida na Figura 18, são apresentados detalhes cadastrais como o tipo de item (corante, fio, malha crua, malha fixada ou malha acabada), o código, a descrição, a família a que pertence e o número de cores, que varia conforme o tipo de demanda escolhido inicialmente.

Figura 18: Tela do Formulário Itens

ID_ITEM	TIPO	CÓDIGO	DESCRIÇÃO	FAMÍLIA	NUM. CORES
1	CORANTE	COR	CORANTES	MATÉRIA PRIMA	3
2	FIO	FIO_1	FIO DE ALGODÃO	MATÉRIA PRIMA	1
3	FIO	FIO_2	FIO SINTÉTICO	MATÉRIA PRIMA	1
4	MALHA_CRUA	MC_1	MALHA CRUA COLMEIA	COLMÉIA	1
5	MALHA_CRUA	MC_2	MALHA CRUA PIQUET	PIQUET	1
6	MALHA_CRUA	MC_3	MALHA CRUA MAXIM	MAXIM	1
7	MALHA_FIXADA	MF_1	MALHA FIXADA COLMEIA	COLMÉIA	1
8	MALHA_FIXADA	MF_2	MALHA FIXADA PIQUET	PIQUET	1
9	MALHA_FIXADA	MF_3	MALHA FIXADA MAXIM	MAXIM	1
10	MALHA_ACABADA	MA_1	MALHA ACABADA COLMEIA	COLMÉIA	3
11	MALHA_ACABADA	MA_2	MALHA ACABADA PIQUET	PIQUET	3
12	MALHA_ACABADA	MA_3	MALHA ACABADA MAXIM	MAXIM	3

Registro: 1 de 12

Fonte: Jogo *LSSP\_PCP1* (2024)

Na tela de Recursos, os jogadores podem visualizar os recursos produtivos disponíveis na fábrica, incluindo supermercados, teares, jets e ramas. Essa tela oferece um resumo geral ou uma visão filtrada por grupo de recursos, os quais também podem ser visualizados nas telas de Capacidades de cada setor discutidas anteriormente.

A tela de Taxa de Produção exibe informações sobre o consumo de recursos por cada item produzido, com a possibilidade de filtrar por família de produto, como ilustrado para todas as famílias na Figura 19. Os dados incluem o nome da família, o

código e a descrição do item, a fase em que o recurso é consumido, o grupo de recursos envolvidos, a taxa de produção e a unidade de medida correspondente.

Figura 19: Tela do Formulário Taxa Produção

The screenshot shows a software window titled 'TaxaProdução : Formulário'. It has a menu bar with 'Início', 'Engenharia', 'Itens', 'Recursos', 'Taxa Produção', 'Estrutura', and 'Custos'. The main area contains a table with the following data:

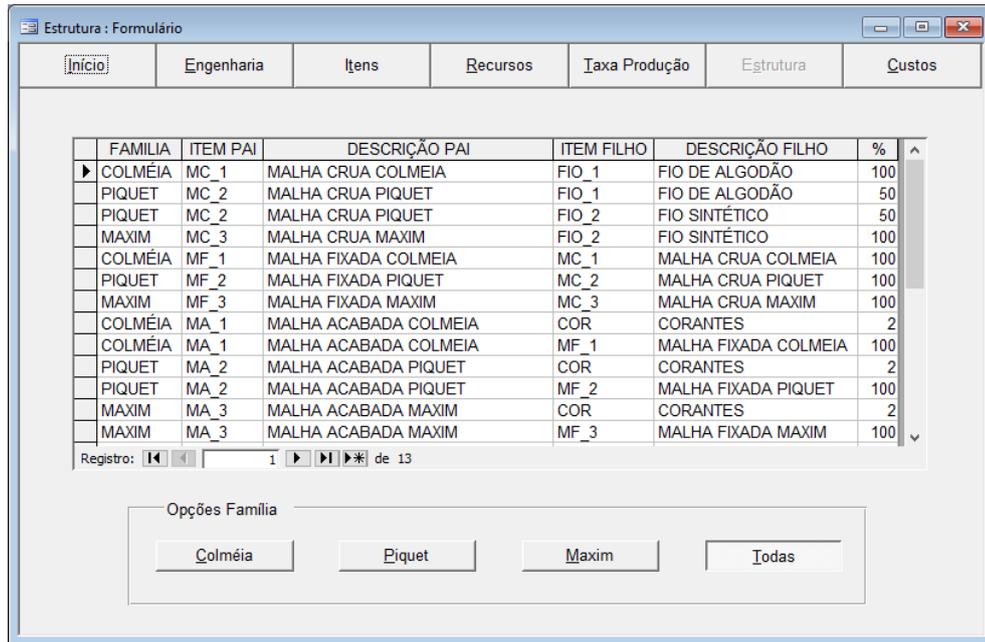
FAMILIA	ITEM	DESCRICAÇÃO	FASE	GRUPO	TAXA	UNIDADE
COLMÉIA	MC_1	MALHA CRUA COLMEIA	MALHARIA	TEARES	0,090	H/Kg
PIQUET	MC_2	MALHA CRUA PIQUET	MALHARIA	TEARES	0,100	H/Kg
MAXIM	MC_3	MALHA CRUA MAXIM	MALHARIA	TEARES	0,110	H/Kg
COLMÉIA	MF_1	MALHA FIXADA COLMEIA	PURGA	JETS	1,000	H/Lote
COLMÉIA	MF_1	MALHA FIXADA COLMEIA	FIXAÇÃO	RAMAS	0,002	H/Kg
PIQUET	MF_2	MALHA FIXADA PIQUET	PURGA	JETS	1,000	H/Lote
PIQUET	MF_2	MALHA FIXADA PIQUET	FIXAÇÃO	RAMAS	0,002	H/Kg
MAXIM	MF_3	MALHA FIXADA MAXIM	PURGA	JETS	1,500	H/Lote
MAXIM	MF_3	MALHA FIXADA MAXIM	FIXAÇÃO	RAMAS	0,003	H/Kg
COLMÉIA	MA_1	MALHA ACABADA COLMEIA	TINTURARIA	JETS	3,000	H/Lote
COLMÉIA	MA_1	MALHA ACABADA COLMEIA	ACABAMENTO	RAMAS	0,003	H/Kg
PIQUET	MA_2	MALHA ACABADA PIQUET	TINTURARIA	JETS	3,500	H/Lote
PIQUET	MA_2	MALHA ACABADA PIQUET	ACABAMENTO	RAMAS	0,003	H/Kg
MAXIM	MA_3	MALHA ACABADA MAXIM	TINTURARIA	JETS	4,000	H/Lote
MAXIM	MA_3	MALHA ACABADA MAXIM	ACABAMENTO	RAMAS	0,004	H/Kg

Below the table, there is a 'Registro:' field showing '1 de 15'. At the bottom, there is a section titled 'Opções Família' with four buttons: 'Colméia', 'Piquet', 'Maxim', and 'Todas'.

Fonte: Jogo LSSP\_PCP1 (2024)

Já na tela de Estrutura, os jogadores podem acessar a árvore dos itens da fábrica, com a opção de visualizar a estrutura completa ou filtrá-la por família, conforme exemplificado na Figura 20 para todas as famílias. Nesta tela, são apresentados o nome da família, o item pai (com seu código e descrição), o item filho (também com código e descrição), e o percentual de uso do item filho no item pai, facilitando o entendimento das relações entre os componentes.

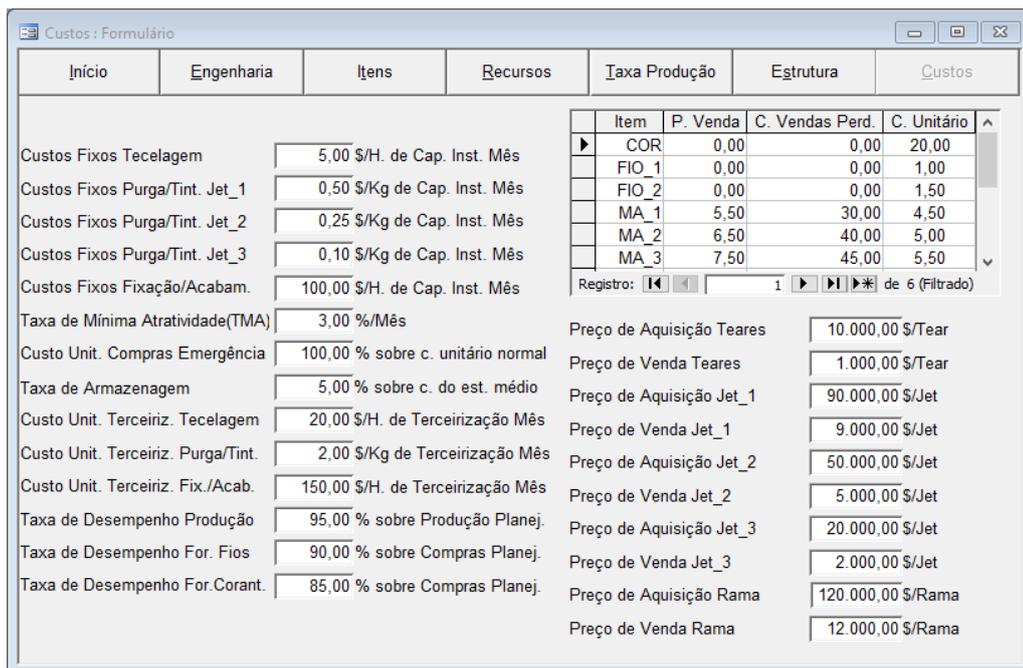
Figura 20: Tela Formulário Estrutura



Fonte: Jogo LSSP\_PCP1 (2024)

Por fim, tem-se a tela de Custos, ilustrada na Figura 21, que detalha as informações financeiras associadas às decisões do jogo. Os jogadores podem consultar custos fixos, custos de compras de matérias-primas, custos de manutenção de estoques, custos de terceirização, custos de capital e custos de vendas perdidas.

Figura 21: Tela Formulário Custos



Fonte: Jogo LSSP\_PCP1 (2024)

Além dessas informações, os jogadores ainda encontram as taxas de desempenho tanto para a produção, quanto para os fornecedores de fios e corantes.

A avaliação financeira do jogo é baseada no Resultado Operacional, que é calculado como a diferença entre as receitas mensais (receitas de vendas de malhas e receitas de vendas de equipamentos) e os custos mensais (fixos, de matérias-primas, de manutenção de estoques, de terceirização, de capital e de vendas perdidas). Essa estrutura completa da tela de Engenharia permite que os jogadores tenham uma visão abrangente do processo produtivo e das implicações financeiras de suas decisões, e em conjunto com as telas do Plano de Produção Financeiro, apresentadas anteriormente, os jogadores podem explorar as fórmulas de cálculo detalhadas para cada componente financeiro, facilitando a análise e a compreensão do impacto de suas decisões.

#### **4.1.3 Identificação de Limitações Tecnológicas e Pontos de Melhorias**

O jogo *LSSP\_PCP1*, que serve de inspiração e base para o desenvolvimento deste novo projeto, não recebe atualizações desde 2008, conforme a página para *download* do jogo no *site* do LSSP, a última versão disponível data de 2008, refletindo a ausência de melhorias ao longo do tempo. Essa defasagem é perceptível em diversos aspectos, tanto tecnológicos quanto funcionais, que não atendem mais às expectativas modernas de usabilidade e acessibilidade.

Além da falta de atualizações, a versão original do *LSSP\_PCP1* foi desenvolvida como um software de desktop, exigindo a instalação local para ser utilizado. Esse formato representa uma limitação significativa em termos de acessibilidade, já que o uso do jogo fica restrito aos dispositivos onde o software foi instalado, limitando a flexibilidade e dificultando a adoção em ambientes de ensino mais dinâmicos.

Outra limitação tecnológica do *LSSP\_PCP1* reside na implementação de recursos de visualização interativos, como gráficos. Embora alguns gráficos estejam presentes, eles frequentemente apresentam problemas de funcionamento, com dados não sendo exibidos corretamente ou telas em branco onde os gráficos deveriam aparecer. Essa instabilidade compromete a experiência do usuário, dificultando a

visualização e interpretação dos dados de simulação. Conseqüentemente, o jogo se torna menos intuitivo e menos envolvente.

Além das limitações nos recursos de visualização, a interface do *LSSP\_PCP1* frequentemente contribui para confundir o usuário em um primeiro momento. A organização e disposição dos elementos na tela não são intuitivas, o que leva a uma curva de aprendizado acentuada para entender as funcionalidades básicas do jogo. Sem uma navegação clara e uma apresentação visual bem estruturada, os jogadores podem ter dificuldades em acessar informações relevantes rapidamente, especialmente durante os primeiros usos.

Essas limitações tecnológicas e funcionais ressaltam a necessidade de uma atualização, tanto em termos de plataforma quanto de funcionalidades. A nova versão proposta, portanto, busca não apenas resolver esses pontos de melhoria, mas também introduzir recursos modernos que ampliem as possibilidades de uso e de análise.

## 4.2 DESENVOLVIMENTO DO NOVO JOGO

Esta seção tem como objetivo demonstrar o processo de desenvolvimento, destacando os passos realizados para transformar as propostas iniciais em funcionalidades concretas e os resultados obtidos ao longo do projeto. Serão apresentadas as tecnologias utilizadas, as novas funcionalidades e interfaces obtidas com o desenvolvimento.

### 4.2.1 Tecnologias Utilizadas

Para o desenvolvimento da nova versão do jogo, foi escolhida a linguagem *Python* devido à sua simplicidade, eficiência e ampla gama de bibliotecas que apoiam diferentes necessidades do projeto. *Python* proporciona uma sintaxe clara, ideal para prototipagem rápida e implementação de funcionalidades complexas, além de contar com uma grande comunidade, o que garante suporte contínuo e recursos adicionais.

Com o intuito de trazer o jogo ao ambiente web, foi utilizado o *Flask*, um *microframework* web em *Python*. O *Flask* permite o desenvolvimento ágil de aplicações web e, com sua arquitetura minimalista, fornece apenas os componentes

essenciais, como roteamento e controle de requisições HTTP, possibilitando ao desenvolvedor adicionar funcionalidades conforme a necessidade do projeto. No caso do jogo, o *Flask* é utilizado como *backend*, gerenciando a lógica do servidor e facilitando a comunicação entre o navegador do usuário e a aplicação. Dessa forma, o jogo pode ser jogado diretamente no navegador, eliminando a necessidade de instalações locais.

No *frontend*, foram empregadas as linguagens HTML, CSS e JS. O HTML é a linguagem responsável pela estrutura básica das páginas web, o CSS cuida do design e *layout*, conferindo ao jogo uma interface mais amigável e intuitiva, sendo assim, responsável por estilizar e melhorar a aparência visual do jogo. JS, por sua vez, foi crucial para adicionar interatividade em tempo real, adicionando interatividade e dinamismo, permitindo que o jogo responda rapidamente às ações do usuário. Para complementar a estética e responsividade, foi utilizado o *framework Bootstrap*, que facilita a criação de uma interface moderna e adaptável, ajustando automaticamente o layout para diferentes tamanhos de tela e dispositivos.

O mecanismo de *templates Jinja2*, integrado ao *Flask*, possibilita a renderização dinâmica de páginas HTML, facilitando a integração entre o *backend* em *Python* e a interface web. Com o *Jinja2*, o *HTML* é gerado dinamicamente, incorporando dados processados pelo servidor diretamente nas páginas exibidas ao usuário. Dessa forma, o uso combinado de *Jinja2* e *Bootstrap* permite que o *frontend* mantenha uma estrutura visual consistente e adaptativa, ao mesmo tempo em que exibe dados atualizados em tempo real para o jogador.

Para o armazenamento e gestão dos dados do jogo, foi utilizado o *SQLite* como banco de dados. O *SQLite* é uma solução de banco de dados relacional leve e embutida, o que o torna ideal para aplicações de pequeno a médio porte, como este projeto. Ao contrário de servidores de banco de dados mais robustos, o *SQLite* armazena os dados localmente em um único arquivo, simplificando a configuração e reduzindo a sobrecarga do sistema.

No contexto do jogo, o *SQLite* facilita o armazenamento persistente de dados, como informações dos jogadores, configurações de simulação e resultados financeiros, permitindo o acesso rápido e eficiente a essas informações pelo *backend* em *Flask*. A integração direta entre o *Flask* e o *SQLite* é garantida através da biblioteca em *python SQLAlchemy*, que fornece uma interface de mapeamento objeto-relacional

(ORM). Essa combinação permite manipular os dados de maneira simplificada e intuitiva, utilizando classes *Python* para interagir com o banco de dados e realizar operações CRUD (*Create, Read, Update, Delete*) de forma mais eficiente.

Para o controle de versão, foram utilizados o *Git* e o *GitHub*. O *Git* é uma ferramenta de controle de versão distribuído que permite o rastreamento detalhado de todas as alterações no código ao longo do desenvolvimento. Cada mudança pode ser registrada em um “*commit*”, possibilitando a criação de um histórico de versões do projeto e facilitando o retorno a versões anteriores, se necessário. Esse recurso foi essencial para manter a organização e o controle sobre o código, especialmente em um projeto com diversas iterações e atualizações.

Na Figura 22 abaixo, é exibida a página do repositório do projeto, onde estão armazenadas as versões do código, o histórico de commits e demais informações relevantes para o controle e acompanhamento do desenvolvimento.

Figura 22: Página do GitHub do Projeto

The screenshot shows the GitHub repository page for 'matos1396 / PCP-TCC-WEB'. The repository is public and has 1 star, 1 watching, and 0 forks. The main content area displays a list of files and folders with their commit messages and dates. The right sidebar shows repository statistics like 1 star, 1 watching, and 0 forks, along with sections for Releases, Packages, Languages (Python 56.0%, HTML 43.6%, CSS 0.4%), and Suggested workflows.

File/Folder	Commit Message	Time Ago
dados	fix: typo e nome variaveis	last week
simulacao	feat: adicionado logica corretamente para lidar com a...	3 days ago
static	feat: adicionado estilos css para melhorar e estilizar as...	3 days ago
templates	feat: adicionado estilos css para melhorar e estilizar as...	3 days ago
utils	feat: adicionado logica corretamente para lidar com a...	3 days ago
.gitignore	gitignore atualizado	3 days ago
app.py	feat: adicionado parada quando periodo atual for o ul...	3 days ago
config.py	feat: melhorado sistema de gerenciamento de sessao ...	3 weeks ago
forms.py	feat: adicionado inicio de rota para cadastrar grupos	3 days ago
init_db.py	feat: adicionado inicio de rota para cadastrar grupos	3 days ago
models.py	fix: typo	3 days ago
routes.py	first commit: upload github	2 months ago
utils_db.py	feat: adicionado inicio de rota para cadastrar grupos	3 days ago

Fonte: *GitHub* (2024)

O *GitHub*, uma plataforma de hospedagem de repositórios *Git* na nuvem, foi utilizado para armazenar o projeto de maneira segura e centralizada. Além de fornecer

um *backup* (cópia de segurança) confiável do código, o *GitHub* facilita o acesso ao repositório a partir de qualquer local e dispositivo. Com sua interface intuitiva, ele permite visualizar o histórico de *commits* e acompanhar a evolução do projeto de forma clara e estruturada, o que foi fundamental para a organização e manutenção do projeto ao longo do desenvolvimento.

Durante o desenvolvimento do jogo, foi essencial contar com ambientes de teste para verificar a funcionalidade e a estabilidade da aplicação conforme novas funcionalidades eram implementadas. Para isso, além da execução local, utilizou-se a plataforma *PythonAnywhere*, que permitiu a hospedagem gratuita do projeto e possibilitou simulações em um ambiente próximo ao real.

A escolha por *PythonAnywhere* ofereceu diversas vantagens. Primeiramente, a plataforma permitiu testar o jogo em um contexto de servidor remoto, semelhante ao que seria utilizado pelos jogadores ao acessar a versão final. Esse tipo de ambiente garantiu uma avaliação mais precisa das interações entre o cliente e o servidor, da performance e do tempo de resposta, além de outros aspectos críticos que podem não ser detectados em um ambiente de desenvolvimento local. Adicionalmente, *PythonAnywhere* facilitou a verificação de questões de compatibilidade e de disponibilidade de recursos, proporcionando uma visão mais completa da aplicação em condições de uso real.

Esse processo de testes em um ambiente de hospedagem foi fundamental para a identificação e correção de possíveis problemas e para ajustes na arquitetura, melhorando a experiência final do usuário.

#### **4.2.2 Arquitetura Geral do Sistema**

O sistema do jogo foi estruturado com uma arquitetura cliente-servidor, onde o *frontend* interage com o usuário final, enquanto o *backend* gerencia a lógica do jogo, operações de banco de dados e cálculos necessários para a simulação. Essa arquitetura garante uma separação clara entre a apresentação visual e a lógica de negócios, o que facilita a manutenção, escalabilidade e segurança do sistema.

O *backend* é responsável pela execução das simulações, armazenamento de dados e autenticação de usuários. No *frontend*, a interface foi implementada de forma

a criar uma experiência interativa e responsiva, permitindo que os jogadores interajam intuitivamente com o sistema.

#### 4.2.2.1 *Frontend*

Conforme discutido anteriormente, o *frontend* abrange toda a interação com o usuário, incluindo a exibição de informações e o processamento dos inputs recebidos. Ele é estruturado em arquivos HTML e CSS, responsáveis pela disposição e aparência visual dos elementos, enquanto o JS lida com a atualização dinâmica de dados em tempo real nos formulários e painéis. Os scripts em JS realizam cálculos diretos na interface, como a atualização automática de valores e a verificação das capacidades de máquinas, alterando a cor de fundo das caixas conforme os valores atendem ou não às necessidades operacionais.

Dentro do escopo principal do *frontend*, destacam-se os arquivos HTML que estruturam as páginas, ou rotas, acessíveis aos jogadores no site. Para simplificar e agilizar o desenvolvimento, o uso do *Jinja2* possibilitou a criação de templates base que permitem o compartilhamento de elementos comuns entre diferentes rotas, evitando a necessidade de duplicação de código. Assim, todas as rotas herdam o template “base.html”, que contém os elementos comuns a todas as páginas, como a barra de navegação e o rodapé. Essa abordagem não só facilita a manutenção, mas também proporciona uma experiência de usuário mais consistente ao longo de todo o jogo.

Os elementos comuns definidos no template base, proporcionam uma estrutura consistente e intuitiva para todas as páginas da aplicação. No topo da página, tem-se a barra de navegação que serve como um ponto central de acesso para todas as principais funcionalidades do jogo. Esta barra se ajusta ao tipo de usuário logado: se o usuário for administrador, são exibidos links específicos de administração, como o gerenciamento de usuários; já para os jogadores comuns, a barra de navegação inclui links para as diferentes etapas e telas do jogo, como o plano de produção, compras, simulação, entre outros.

A barra de navegação é configurada com *Bootstrap*, que permite um design responsivo. Em dispositivos móveis ou telas menores, a barra se transforma em um menu colapsável, facilitando a navegação e otimizando a experiência do usuário.

Além disso, com o *Jinja2* foi possível gerar formulários de maneira dinâmica, permitindo que os campos e valores fossem ajustados conforme os dados vindos do *backend*. Com o intuito de exemplificar a implementação desse processo, tomando como exemplo os formulários de planejamento da produção, o *Jinja2* é usado para iterar sobre os períodos e gerar campos de entrada específicos para cada período e variável, como Demanda Prevista e Demanda Real. Dessa forma, a interface reflete automaticamente o estado atual do jogo, promovendo uma interação mais fluida e precisa entre o jogador e sistema.

A Figura 23 a seguir representa um trecho de código do arquivo *production.html*, nela é possível observar essa implementação para os campos de Demanda Prevista e Demanda Real:

Figura 23: Trecho do código arquivo *production.html*

```

<tr>
  <th>Demanda Prevista</th>
  {% for period in periods %}
    <td>
      {% if period <= periodo_atual %}
        <!-- Campo somente leitura para períodos passados -->
        <input type="number" class="form-control demanda-prevista"
              id="colmeia_demanda_prevista_{{ period }}"
              name="{{ previsoes_por_familia['Colmeia'][period].name }}"
              data-familia="Colmeia" data-period="{{ period }}"
              value="{{ previsoes_por_familia['Colmeia'][period].data }}" readonly>
      {% else %}
        {{ previsoes_por_familia['Colmeia'][period] (class="form-control demanda-prevista",
              id="colmeia_demanda_prevista_" ~ period,
              data_familia="Colmeia",
              data_period=period) }}
      {% endif %}
    </td>
  {% endfor %}
</tr>
<tr>
  <th>Demanda Real</th>
  {% for period in periods %}
    <td>
      <input type="number" class="form-control demanda-real"
            id="colmeia_demanda_real_{{ period }}"
            name="{{ demanda_real_por_familia['Colmeia'][period].name }}"
            value="{{ demanda_real_por_familia['Colmeia'][period].data }}" readonly>
    </td>
  {% endfor %}
</tr>

```

Fonte: Elaborado pelo autor (2024)

É possível observar na Figura 23 como a utilização do *Jinja2* é responsável por gerar uma tabela dinâmica onde são exibidos dados referentes à Demanda Prevista e Demanda Real para a família de produtos Colméia ao longo de vários

períodos. A seguir, será descrita uma explicação de cada parte desse processo no código:

## 1. Demanda Prevista:

- 1.1. A linha `<th>Demanda Prevista</th>` cria o cabeçalho da linha, indicando que a linha contém valores da "Demanda Prevista".
- 1.2. `{% for period in periods %}` é uma estrutura de *loop* em *Jinja2* que itera sobre a lista de períodos (*periods*). Para cada período, uma nova célula (`<td>`) é criada.
- 1.3. Dentro do loop, há um condicional `{% if period <= periodo_atual %}` que verifica se o período atual é menor ou igual ao período em questão:
  - 1.3.1. Se o período já passou ou é o período atual, um campo de entrada (`<input type="number">`) é exibido, mas em modo somente leitura (*readonly*). Isso significa que o valor está bloqueado para edição em períodos passados. O valor exibido é definido pela variável `previsoes_por_familia['Colmeia'][period].data`.
  - 1.3.2. Se o período é futuro, o valor é exibido com o campo de entrada editável, permitindo ao usuário definir o valor de demanda para períodos ainda não ocorridos.
- 1.4. Os atributos `id`, `name`, `data-familia`, e `data-period` são configurados dinamicamente:
  - 1.4.1. `id` e `name`: Esses identificadores são gerados usando `period` para criar IDs e nomes únicos para cada período, facilitando o acesso a cada campo no JS e no *backend*.
  - 1.4.2. `data-familia` e `data-period`: Esses atributos personalizados (`data-*`) armazenam metadados úteis (neste caso, a família e o período), o que ajuda a manipular esses dados no *frontend* de maneira mais organizada.

## 2. Demanda Real (*segundo <tr>*):

- 2.1. A linha *<th>Demanda Real</th>* é o cabeçalho da linha, indicando que a linha contém valores da "Demanda Real".
- 2.2. Novamente, um *loop {% for period in periods %}* é usado para iterar sobre cada período, criando uma célula (*<td>*) para cada um.
- 2.3. Para cada célula, um campo de entrada de número (*<input type="number" class="form-control demanda-real">*) é gerado, mas com *readonly*, significando que o campo é sempre somente leitura. Isso indica que a "Demanda Real" para cada período é exibida, mas não pode ser alterada manualmente pelo usuário.
- 2.4. Assim como na "Demanda Prevista", os atributos *id* e *name* são configurados dinamicamente usando o nome da família e o período, facilitando a manipulação de cada campo.

## 3. Uso de *Jinja2* para renderização dinâmica:

- 3.1. *{{ }}* e *{% %}* são comandos *Jinja2* usados para inserir variáveis ou lógica no HTML. O *Jinja2* permite a criação de conteúdo dinâmico (como loops e condicionais) diretamente no HTML, adaptando o layout da página conforme os dados do *backend*.
- 3.2. Exemplo: *{{ previsoes\_por\_familia["Colmeia"][period].data }}* insere diretamente o valor da demanda prevista para um período específico no HTML, permitindo que cada célula da tabela seja preenchida com dados únicos.
- 3.3. Isso significa que, ao renderizar a página, o *Jinja2* compila o código dinâmico com os valores do *backend*, gerando uma página HTML completa e personalizada com base nos dados fornecidos pelo servidor.

Esse exemplo demonstra como o *Jinja2* foi útil para criar interfaces dinâmicas e responsivas, com campos de entrada e exibição de dados condicionados, melhorando a interação com o usuário ao permitir que informações se adaptem automaticamente ao contexto dos dados.

Para todos os demais formulários do jogo, a lógica utilizada é similar: os dados específicos de cada família de produtos, máquina ou período são renderizados

dinamicamente com base nas variáveis do *backend*, adaptando os campos conforme necessário, seja para leitura ou edição, e gerando elementos HTML únicos e interativos para cada situação. Essa abordagem proporciona uma consistência no design e na usabilidade das diferentes telas do sistema, enquanto garante que todos os elementos compartilhados entre as páginas sejam manipulados de forma eficiente e centralizada.

#### 4.2.2.2 *Backend*

Assim como o *frontend* é responsável pela interface e interação do usuário com o sistema, o *backend* sustenta a lógica do jogo, realiza cálculos e gerencia as operações com o banco de dados. Essa camada foi implementada com o *Flask*. O *backend* é o núcleo do jogo, manipulando dados dos usuários, regras de simulação e informações financeiras, e responde a solicitações enviadas a partir do *frontend*, fornecendo dados processados e armazenando resultados.

No *backend*, cada rota corresponde a uma página acessada pelo jogador ou administrador. O *Flask* gerencia essas rotas, recebendo e respondendo a solicitações HTTP, o que permite a execução de funcionalidades como o login, o processamento do plano de produção, a simulação, e a visualização de resultados financeiros. Cada rota também verifica as permissões do usuário, garantindo que apenas usuários autenticados ou administradores autorizados possam acessar informações específicas.

Para a persistência de dados, o *backend* utiliza um banco de dados relacional gerenciado por *SQLAlchemy*, — que será discutido na próxima subseção — que abstrai as operações de banco de dados em *Python*. A estrutura de dados foi organizada em modelos que representam cada entidade do jogo.

A relação entre esses modelos permite consultas e atualizações eficientes de dados, facilitando o acesso a informações específicas, como o desempenho financeiro de um grupo em um determinado período. Essas consultas são usadas para gerar tabelas de resultados e painéis com gráficos, que são exibidos no *frontend*.

Quando um usuário interage com o *frontend*, os dados enviados (por exemplo, no formulário de Planejamento de Produção) são processados pelo *backend*. O *Flask*

coleta essas informações e as repassa para funções de cálculo ou operações de banco de dados:

No caso do planejamento de produção, os dados de cada família de produtos são enviados ao *backend*, onde são verificados e processados.

Para simulações de produção, o *backend* utiliza esses dados, aplica as regras de negócio e, em seguida, armazena os resultados no banco de dados.

Após o processamento, os resultados são renderizados e enviados de volta ao *frontend*. Por exemplo para a rota de planejamento de produção, o *backend* realiza uma série de etapas para garantir que os dados inseridos pelo usuário sejam processados, validados e armazenados corretamente. Quando o método *POST* é acionado, ele indica que o usuário submeteu um formulário para salvar o plano de produção. A Figura 24 e Figura 25 ilustram um trecho do código da rota *production* no arquivo *app.py* para como forma de exemplo.

Figura 24: Trecho parte 1 da rota *production*

```

if request.method == 'POST':
    for familia in ['Colmeia', 'Piquet', 'Maxim']:
        for period in periods:
            if period >= periodo_atual:
                # Obter os valores atuais do forms
                demanda_prevista = previsoes_por_familia[familia][period].data
                demanda_real = demanda_real_por_familia[familia][period].data
                producao_planejada = producao_planejada_por_familia[familia][period].data
                producao_real = producao_real_por_familia[familia][period].data
                estoque_inicial = estoques_iniciais_por_familia[familia][period].data
                estoque_final = estoques_finais_por_familia[familia][period].data
                vendas_perdidas = vendas_perdidas_por_familia[familia][period].data
                vendas = vendas_por_familia[familia][period].data

                # Atualizar ou criar novo plano de produção
                existing_plan = PlanoProducao.query.filter_by(
                    periodo_numero=period, familia=familia, grupo_id=current_user.id
                ).order_by(PlanoProducao.periodo_modificado.desc()).first()

                if existing_plan:
                    if existing_plan.periodo_modificado == periodo_atual:
                        # Atualizar o plano existente
                        existing_plan.demanda_prevista = demanda_prevista
                        existing_plan.demanda_real = demanda_real
                        existing_plan.producao_planejada = producao_planejada
                        existing_plan.producao_real = producao_real
                        existing_plan.estoques_iniciais = estoque_inicial
                        existing_plan.estoques_finais = estoque_final
                        existing_plan.vendas_perdidas = vendas_perdidas
                        existing_plan.vendas = vendas

```

Fonte: Elaborado pelo autor (2024)

Figura 25: Trecho parte 2 da rota *production*

```

else:
    # Criar um novo plano
    new_plan = PlanoProducao(
        grupo_id=current_user.id,
        periodo_numero=period,
        periodo_modificado=periodo_atual,
        familia=familia,

        demanda_prevista=demanda_prevista,
        demanda_real=demanda_real,
        producao_planejada=producao_planejada,
        producao_real=producao_real,
        estoques_iniciais=estoque_inicial,
        estoques_finais=estoque_final,
        vendas_perdidas=vendas_perdidas,
        vendas=vendas
    )

    db.session.add(new_plan)

atualizar_plano_compras(current_user)
atualizar_capacidade_maquinas(current_user)
atualizar_financeiro(current_user)

# Flag de controle
set_flag_controle(current_user, "producao")

db.session.commit()
return redirect(url_for('dashboard'))

```

Fonte: Elaborado pelo autor (2024)

O fluxo de execução no *backend* pode ser detalhado da seguinte maneira para a rota *production*:

- 1. Iteração por Família e Período:** O código inicia iterando por cada família de produtos (Colméia, Piquet, e Maxim) e por cada período relevante. A combinação de cada família e período representa um conjunto de dados de produção a ser salvo para o respectivo grupo de produtos.

- 2. Obtenção de Dados do Formulário:** Para cada combinação de família e período, o código extrai os valores enviados pelo formulário para variáveis como *demanda\_prevista*, *demanda\_real*, *producao\_planejada*, *producao\_real*, *estoque\_inicial*, *estoque\_final*, *vendas\_perdidas*, e *vendas*.
- 3. Verificação de Planos Existentes:** Após obter os dados, o código verifica se já existe um plano de produção para o grupo e período específicos no banco de dados. Para isso, ele usa a consulta *PlanoProducao.query.filter\_by(...)* com uma ordenação para pegar o plano mais recente (com base no campo *periodo\_modificado*). Esse passo garante que, se um plano para aquele período já foi salvo, ele será atualizado com os novos dados; caso contrário, um novo plano será criado.
- 4. Atualização ou Criação de Novo Plano:**

  - 4.1.** Se um plano de produção já existe para o período atual, o código atualiza seus campos com os novos dados extraídos do formulário.
  - 4.2.** Se não há um plano existente, o código cria um registro *PlanoProducao* e adiciona os dados do formulário como atributos. A linha *db.session.add(new\_plan)* insere o novo plano na sessão, para que ele seja salvo no banco de dados no próximo *commit*.
- 5. Atualização de Outras Tabelas:** Uma vez que os dados do plano de produção são processados, o *backend* chama funções específicas para atualizar outras tabelas no banco de dados, como:

  - 5.1.** *atualizar\_plano\_compras(...)*: Atualiza o plano de compras do usuário.
  - 5.2.** *atualizar\_capacidade\_maquinas(...)*: Recalcula e ajusta a capacidade das máquinas com base nos novos dados de produção.
  - 5.3.** *atualizar\_financeiro(...)*: Ajusta as informações financeiras, incluindo custos e receitas, baseando-se nos valores atualizados do plano de produção.

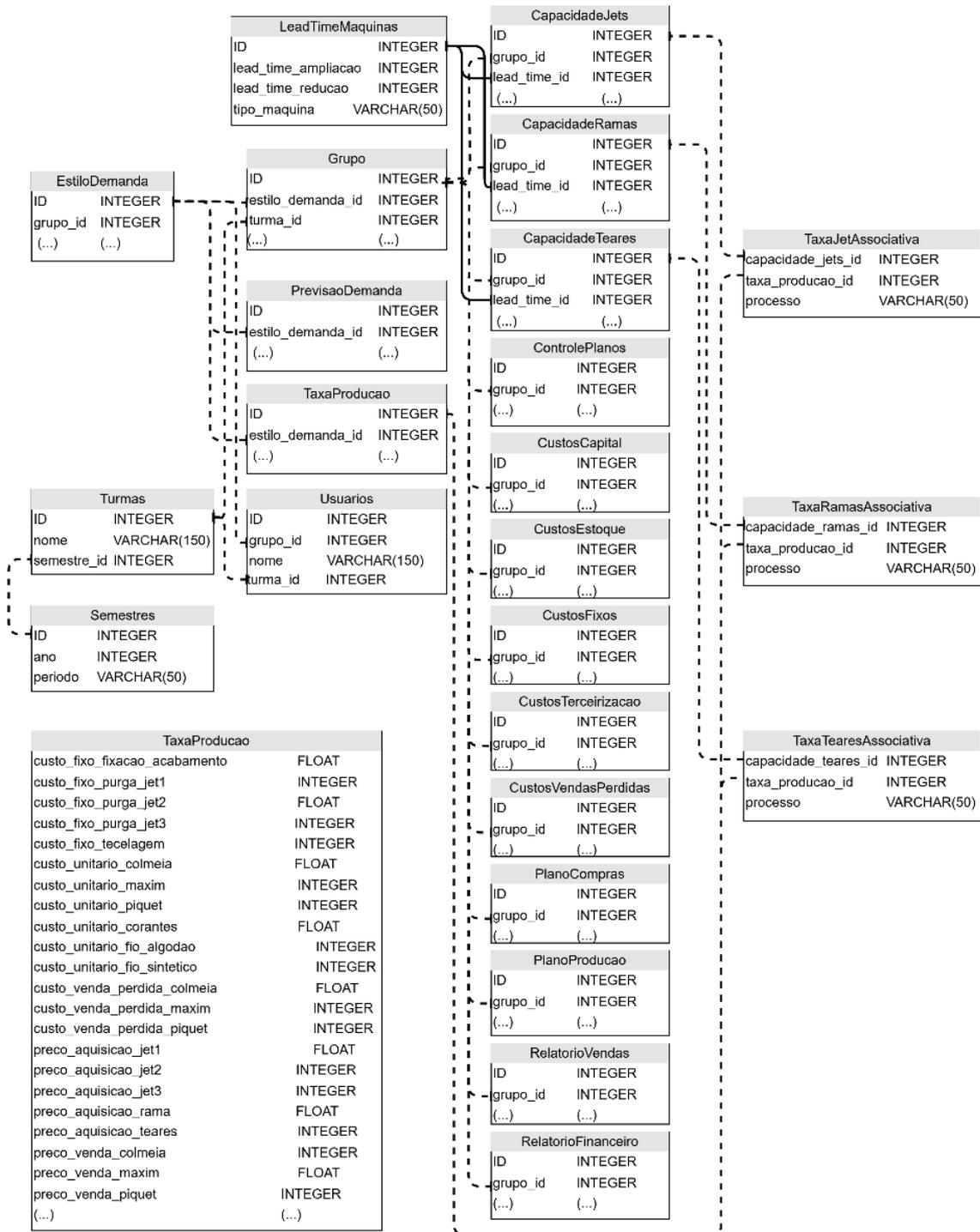
6. **Controle de Fluxo:** A função `set_flag_controle(...)` é usada para marcar o progresso no fluxo do jogo, indicando que o plano de produção foi salvo para o período atual. Essa *flag* serve como uma medida de controle para o *backend*.
7. **Persistência e Feedback ao Usuário:** Após todas as operações, o *backend* realiza o `db.session.commit()` para salvar todas as mudanças no banco de dados. Isso inclui qualquer plano atualizado ou criado, bem como as atualizações nas tabelas de compras, capacidade das máquinas e informações financeiras. O sistema, então, gera uma mensagem de *feedback* ao usuário, informando que o plano foi salvo com sucesso. Essa mensagem é exibida no *frontend* para o jogador, garantindo uma experiência de usuário mais completa e informativa.
8. **Redirecionamento e Métricas de Desempenho:** Após o salvamento, o *backend* redireciona o usuário para o *dashboard* que agora estará atualizado com as novas informações.

#### 4.2.2.3 Banco de Dados

O banco de dados é fundamental para armazenar e gerenciar as informações do jogo, como os dados dos grupos de jogadores, planos de produção, capacidades das máquinas, custos e receitas. A modelagem foi realizada utilizando o *SQLAlchemy*, uma ferramenta ORM que facilita a interação entre o código *Python* e o banco de dados relacional.

A seguir, serão apresentadas figuras, geradas por meio da biblioteca *SQLAlchemy Data Model Visualizer* em *Python*, que representam as tabelas do banco de dados, juntamente com a explicação de suas respectivas funções e relações, conforme ilustrado no diagrama de relacionamento de entidade do banco de dados simplificado, Figura 26, destacando as interconexões entre as tabelas e como elas sustentam a lógica do jogo.

Figura 26: Diagrama Simplificado de Relacionamento de Entidade



Fonte: Elaborado pelo autor (2024)

No centro dessa modelagem está a tabela *Grupo*, representada pelo Quadro 1, que representa os usuários do sistema, ou seja, os grupos de jogadores participantes do jogo. Cada grupo possui informações essenciais, como um identificador único,

nome, senha criptografada, período atual no jogo, referências ao estilo de demanda associado e referência a turma associada. Além disso, cada grupo mantém relacionamentos com diversas outras tabelas, como planos de produção, planos de compras, capacidades de máquinas, estilo de demandas e relatórios financeiros, o que permite um gerenciamento integrado das informações relacionadas ao grupo.

Quadro 1: Grupo

Grupo	
Atributo	Tipo de Variável
id	INTEGER (PK)
is_admin	BOOLEAN
grupo_nome	VARCHAR(150) (Unique)
password	VARCHAR(150)
periodo_atual	INTEGER
estilo_demanda_id	INTEGER
quantidade_teares	INTEGER
quantidade_ramas	INTEGER
quantidades_jets_tipo1	INTEGER
quantidades_jets_tipo2	INTEGER
quantidades_jets_tipo3	INTEGER
turma_id	INTEGER

Fonte: Elaborado pelo Autor (2024)

O *EstiloDemanda*, representado pelo Quadro 2, define diferentes padrões de demanda que podem ser atribuídos aos grupos. Essa tabela permite simular cenários variados no jogo, influenciando diretamente as previsões de demanda e as taxas de produção. Cada estilo de demanda está associado a quantidades iniciais de equipamentos, como teares, ramas e os diferentes tipos de jets, que são utilizados na

definição da estrutura inicial dos grupos, e relaciona-se com as tabelas de *PrevisaoDemanda* e *TaxaProducao* além da tabela *Turmas*.

Quadro 2: Estilo de Demanda

EstiloDemanda	
Atributo	Tipo de Variável
id	INTEGER (PK)
nome_estilo	VARCHAR(50) (Unique)
quantidade_teaes	INTEGER
quantidade_ramas	INTEGER
quantidades_jets_tipo1	INTEGER
quantidades_jets_tipo2	INTEGER
quantidades_jets_tipo3	INTEGER

Fonte: Elaborado pelo Autor (2024)

A tabela *PrevisaoDemanda*, representada pelo Quadro 3, armazena as previsões de demanda para cada família de produtos em diferentes períodos, de acordo com o estilo de demanda. Cada previsão de demanda está vinculada a um estilo de demanda específico.

Quadro 3: Previsão de Demanda

PrevisaoDemanda	
Atributo	Tipo de Variável
id	INTEGER (PK)
numero_periodo	INTEGER
familia	VARCHAR(50)
valor_previsao	FLOAT
estilo_demanda_id	INTEGER

Fonte: Elaborado pelo Autor (2024)

Os Planos de Produção e Planos de Compras registram as estratégias elaboradas pelos grupos para atender às demandas previstas. A tabela *PlanoProducao*, representada pelo Quadro 4, inclui informações como demanda prevista e real, estoques iniciais e finais, produção planejada e real, vendas e vendas perdidas para cada família de produtos em cada período. Adicionalmente, a tabela conta com a variável *periodo\_modificado*, que serve como uma referência para rastrear cada plano final salvo pelos grupos em cada período. Essa variável é essencial para identificar alterações realizadas ao longo do tempo, permitindo monitorar o histórico das decisões e ajustes feitos pelos grupos durante a simulação.

Quadro 4: Plano de Produção

PlanoProducao	
Atributo	Tipo de Variável
id	INTEGER (PK)
grupo_id	INTEGER
periodo_numero	INTEGER
periodo_modificado	INTEGER
familia	VARCHAR(50)
demanda_prevista	FLOAT
demanda_real	FLOAT
estoques_iniciais	FLOAT
producao_planejada	FLOAT
producao_real	FLOAT
estoques_finais	FLOAT
vendas_perdidas	FLOAT
vendas	FLOAT

Fonte: Elaborado pelo Autor (2024)

Já o plano a tabela *PlanoCompras*, representada pela Quadro 5, compreende o planejamento de aquisição de matérias-primas, incluindo consumo previsto e real, estoques, compras planejadas e emergenciais.

Quadro 5: Plano de Compras

PlanoCompras	
Atributo	Tipo de Variável
id	INTEGER (PK)
grupo_id	INTEGER
periodo_numero	INTEGER
periodo_modificado	INTEGER
material	VARCHAR(50)
consumo_previsto	FLOAT
consumo_real	FLOAT
estoques_iniciais	FLOAT
compra_planejada	FLOAT
compra_real	FLOAT
compra_emergencial	FLOAT
estoques_finais	FLOAT

Fonte: Elaborado pelo Autor (2024)

A capacidade produtiva dos diferentes setores de cada grupo, é gerenciada pelas tabelas *CapacidadeTeares*, *CapacidadeJets* e *CapacidadeRamas*. Essas tabelas controlam as quantidades de máquinas disponíveis, as ampliações e reduções realizadas, a capacidade disponível e necessária, bem como parâmetros de desempenho como produtividade, setup e número de turnos. Cada uma dessas tabelas está relacionada a um grupo específico e incorpora mecanismos para validar se a capacidade disponível atende às necessidades de produção. No Quadro 6 a seguir, é representada a tabela *CapacidadeTeares*.

Quadro 6: Capacidade Teares

CapacidadeTeares	
Atributo	Tipo de Variável
id	INTEGER (PK)
grupo_id	INTEGER
lead_time_id	INTEGER
periodo_numero	INTEGER
periodo_modificado	INTEGER
quantidade	FLOAT
ampliacoes	INTEGER
reducoes	INTEGER
capacidade_disponivel	FLOAT
capacidade_necessaria	FLOAT
capacidade_instalada	FLOAT
capacidade_terceirizada	INTEGER
colmeia	FLOAT
piquet	FLOAT
maxim	FLOAT
setup	FLOAT
produtividade	FLOAT
numero_turnos	INTEGER
validacao	BOOLEAN

Fonte: Elaborado pelo Autor (2024)

Para as demais máquinas, ramas e jets, as tabelas *CapacidadeRamas* e *CapacidadeJets* seguem uma estrutura análoga à de *CapacidadeTeares*, controlando os mesmos aspectos. No caso da tabela *CapacidadeJets*, há parâmetros específicos para cada tipo de jet, o que permite ajustar e monitorar a capacidade individualmente para cada categoria, assim como no jogo original. Essa distinção possibilita uma gestão mais detalhada e precisa, considerando as particularidades de cada tipo de jet no planejamento produtivo.

As taxas de produção são definidas na tabela *TaxaProducao*, ilustrada na Quadro 7 abaixo, que especifica as taxas para cada família de produtos, processo

produtivo e tipo de equipamento. Essa tabela permite relacionar as capacidades das máquinas às taxas de produção correspondentes, influenciando diretamente o planejamento e a execução das operações produtivas.

Quadro 7: Taxa Produção

TaxaProducao	
Atributo	Tipo de Variável
id	INTEGER (PK)
familia	VARCHAR(50)
processo	VARCHAR(50)
tipo_equipamento	VARCHAR(50)
taxa	FLOAT
estilo_demanda_id	VARCHAR(50)

Fonte: Elaborado pelo Autor (2024)

O gerenciamento dos tempos necessários para ampliação ou redução de máquinas é realizado pela tabela *LeadTimeMaquinas*. Essa tabela armazena os *lead times* (tempo de espera) associados a cada tipo de máquina, permitindo que o sistema considere esses tempos nas simulações e planejamentos.

Por exemplo, a Figura 27 abaixo mostra como seria os dados da tabela fazendo uma consulta com SQL:

Figura 27: Consulta Tabela Lead Time Máquinas

The image shows a screenshot of a SQL query interface. At the top, it says 'SQL' with a dropdown arrow. To the right, there are navigation icons: a left arrow, a box containing '1', a slash, another '1', a right arrow, and '1 - 3 of 3'. Below this, the SQL query is displayed: 'SELECT \* FROM lead\_time\_maquinas;'. The result is shown as a table with four columns: 'id', 'tipo\_maquina', 'lead\_time\_ampliacao', and 'lead\_time\_reducao'. The data rows are as follows:

id	tipo_maquina	lead_time_ampliacao	lead_time_reducao
1	Teares	2	2
2	Jets	3	2
3	Ramas	3	3

Fonte: Elaborado pelo Autor (2024)

O desempenho financeiro dos grupos é monitorado através da tabela *RelatorioFinanceiro*, ilustrada pelo Quadro 8, que consolida os resultados financeiros de cada grupo em cada período, incluindo custos fixos, custos de compra de matérias-primas, custos de estoques, custos de terceirização, custos de capital, custos de vendas perdidas, custos totais, receitas de vendas, resultado operacional e resultado operacional acumulado. Para um detalhamento ainda maior, existem tabelas específicas que registram diferentes categorias de custos, como *CustosFixos*, *CustosCompraMP*, *CustosEstoques*, *CustosTerceirizacao*, *CustosCapital* e *CustosVendasPerdidas*, além da tabela *ReceitasVendas*, que registra as receitas provenientes das vendas realizadas.

Quadro 8: Relatório Financeiro

RelatorioFinanceiro	
Atributo	Tipo de Variável
Id	INTEGER (PK)
grupo_id	INTEGER
periodo	INTEGER
custos_fixos	FLOAT
custos_compras_mp	FLOAT
custos_estoques	FLOAT
custos_terceirizacao	FLOAT
custos_capital	FLOAT
custos_vendas_perdidas	FLOAT
custos_totais	FLOAT
receitas_vendas	FLOAT
resultado_operacional	FLOAT
ro_acumulado	FLOAT

Fonte: Elaborado pelo Autor (2024)

A tabela *Custos*, ilustrada na Quadro 9, atua como uma configuração geral dos parâmetros de custos utilizados no jogo. Ela armazena informações como custos fixos por departamento e equipamento, custos variáveis como taxa mínima de atratividade

(TMA) e taxa de armazenagem, preços de aquisição e venda de equipamentos, preços de venda e custos de vendas perdidas por família de produtos, além de custos unitários de materiais. Embora não possua relacionamentos diretos com outras tabelas, essa tabela é fundamental para os cálculos financeiros e operacionais realizados no *backend*.

Quadro 9: Custos

(continua)

<b>Custos</b>	
<b>Atributo</b>	<b>Tipo de Variável</b>
id	INTEGER (PK)
custos_fixos_tecelagem	FLOAT
custos_fixos_purga_jet1	FLOAT
custos_fixos_purga_jet2	FLOAT
custos_fixos_purga_jet3	FLOAT
custo_fixo_fixacao_acabamento	FLOAT
tma	FLOAT
custo_unitario_compra_emergencia	FLOAT
taxa_armazenagem	FLOAT
custo_terceirizacao_tecelagem	FLOAT
custo_terceirizacao_purga_tinturaria	FLOAT
custo_terceirizacao_fixacao_acabamento	FLOAT
taxa_desempenho_producao	FLOAT
taxa_desempenho_fornecimento_fios	FLOAT
taxa_desempenho_fornecimento_corantes	FLOAT
preco_aquisicao_teares	FLOAT
preco_venda_teares	FLOAT
preco_aquisicao_jet1	FLOAT
preco_venda_jet1	FLOAT
preco_aquisicao_jet2	FLOAT

Quadro 9: Custos

(conclusão)

Custos	
Atributo	Tipo de Variável
preco_venda_jet2	FLOAT
preco_aquisicao_jet3	FLOAT
preco_venda_jet3	FLOAT
preco_aquisicao_rama	FLOAT
preco_venda_rama	FLOAT
preco_venda_colmeia	FLOAT
preco_venda_piquet	FLOAT
preco_venda_maxim	FLOAT
custo_venda_perdida_colmeia	FLOAT
custo_venda_perdida_piquet	FLOAT
custo_venda_perdida_maxim	FLOAT
custo_unitario_colmeia	FLOAT
custo_unitario_piquet	FLOAT
custo_unitario_maxim	FLOAT
custo_unitario_corantes	FLOAT
custo_unitario_fio_algodao	FLOAT
custo_unitario_fio_sintetico	FLOAT

Fonte: Elaborado pelo Autor (2024)

Para controlar o progresso dos grupos no jogo, existe ainda uma tabela *ControlePlanos* que monitora se os planos de produção e compras foram salvos em determinado período. Isso assegura que os grupos sigam o fluxo correto do jogo e permite ao sistema gerenciar adequadamente as etapas de planejamento e execução.

Os relacionamentos entre as tabelas foram estruturados para refletir as interdependências das diferentes entidades no jogo. Por exemplo, os grupos estão no centro dos relacionamentos, conectando-se a planos de produção, planos de compras, capacidades de máquinas e relatórios financeiros. Os estilos de demanda influenciam as previsões de demanda e as taxas de produção, impactando diretamente o planejamento dos grupos. As capacidades das máquinas refletem as decisões dos grupos em relação à infraestrutura produtiva, afetando a capacidade de produção e os custos associados. Os relatórios financeiros e as tabelas de custos permitem avaliar o desempenho financeiro dos grupos, considerando receitas e despesas de diferentes naturezas.

Além disso, tabelas associativas foram utilizadas para facilitar os relacionamentos muitos-para-muitos entre taxas de produção e capacidades das máquinas. Isso permite uma gestão flexível das taxas de produção em relação aos equipamentos disponíveis, essencial para simular cenários realistas de produção.

#### **4.2.3 Fluxo do jogo atualizado**

O fluxo do jogo atualizado preserva a lógica do jogo original, com aprimoramentos para melhorar a interatividade e a experiência do usuário. Além disso, os valores/dados utilizados para a simulação foram mantidos iguais aos dados extraídos do jogo original; entretanto, é possível modificar tais valores caso o administrador deseje. Sendo assim, todas as taxas, parâmetros e variáveis de simulação foram configuradas com base no jogo original.

No entanto, com a centralização do jogo em uma plataforma web, foi necessário realizar algumas alterações no processo de cadastro de grupos e usuários para unificar e gerenciar essas informações de forma mais eficiente. Essas mudanças serão detalhadas na próxima seção.

### **4.3 NOVAS FUNCIONALIDADES**

Para aprimorar a experiência do usuário e facilitar o acompanhamento dos resultados pelo administrador, a nova versão do jogo incorpora diversas funcionalidades que ampliam significativamente a interatividade, acessibilidade e

praticidade da plataforma. A seguir, são descritas as novas funcionalidades e os aprimoramentos em relação à versão original, seguidos de uma apresentação das interfaces finais com ilustrações das principais telas.

#### **4.3.1 Jogo como Aplicação Web**

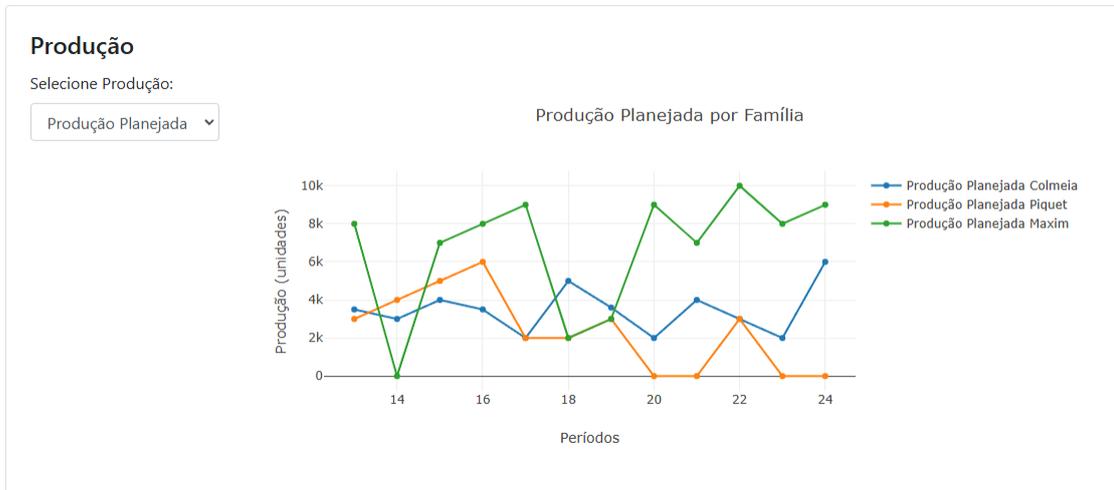
Uma das principais melhorias, que também reflete a motivação central para o desenvolvimento deste trabalho, é que o jogo agora é acessível diretamente como uma aplicação web. Essa mudança elimina a necessidade de instalações ou downloads de programas adicionais, bastando aos jogadores o uso de um navegador para acessar a plataforma. Esse formato facilita o acesso ao jogo a partir de qualquer dispositivo com conexão à internet, proporcionando flexibilidade e conveniência. Com essa arquitetura, a barreira de entrada para os jogadores é reduzida, permitindo que o jogo seja acessado de forma rápida e prática, seja em ambientes de estudo ou em casa

#### **4.3.2 Painel Dashboard Interativo**

O jogo inclui agora um painel *dashboard* interativo, que permite a visualização de dados e indicadores de desempenho ao longo das rodadas. Este painel apresenta gráficos dinâmicos que ilustram métricas importantes para o planejamento e controle de produção, como dados financeiros, de produção, de estoque e de capacidades das máquinas. Além de tornar a análise de dados mais intuitiva, o dashboard oferece uma funcionalidade de exportação que permite aos jogadores baixar os gráficos em formato de imagem. Dessa forma, os dados podem ser salvos ou compartilhados externamente, auxiliando na documentação dos resultados e na análise detalhada após as sessões de jogo.

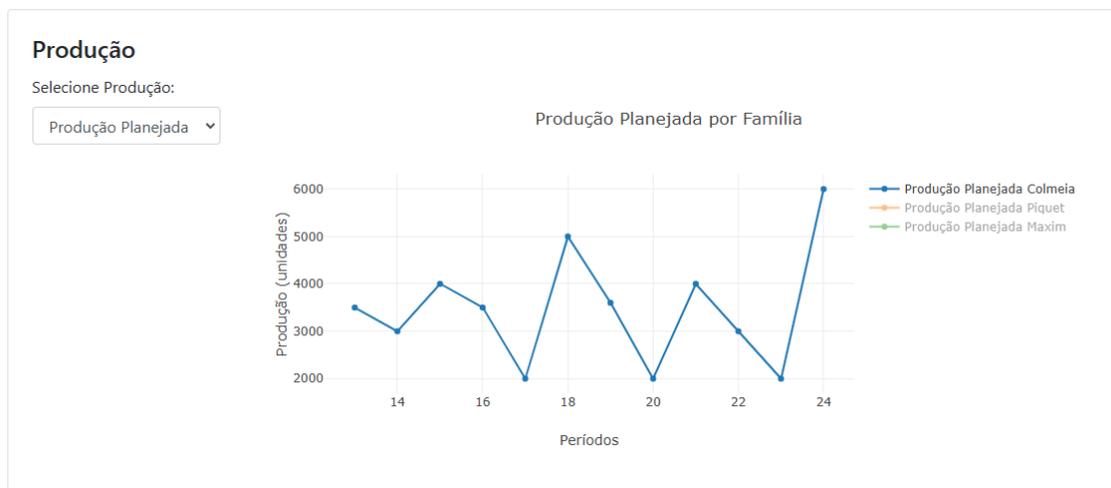
As Figuras 28 e 29, ilustra um dos novos gráficos interativos disponíveis para produção, onde é possível selecionar por produção planejada ou produção real, além de filtrar por família ou ampliar a imagem sendo ainda possível baixar o gráfico como um arquivo de imagem.

Figura 28: Gráfico Interativo Produção Planejada



Fonte: Elaborado pelo Autor (2024)

Figura 29: Gráfico Interativo Produção Planejada com Filtro



Fonte: Elaborado pelo Autor (2024)

### 4.3.3 Indicadores Visuais

Para simplificar o acompanhamento das capacidades de produção e alertar os jogadores sobre possíveis gargalos, foram implementados indicadores visuais nas telas relevantes do jogo. Esses indicadores permitem que os jogadores identifiquem rapidamente se as capacidades disponíveis das máquinas são suficientes para atender às necessidades da produção. Quando a capacidade disponível é insuficiente

para cobrir a capacidade necessária, a caixa numérica correspondente fica com o fundo vermelho, indicando uma restrição de capacidade. Em contrapartida, quando a capacidade disponível é suficiente, o fundo da caixa se torna verde, sinalizando que o planejamento está dentro dos limites operacionais. Esse recurso visual melhora a tomada de decisões ao longo do jogo, orientando os jogadores a ajustar suas estratégias de produção com mais precisão. A Figura 30 ilustra esse funcionamento para o setor de tecelagem.

Figura 30: Exemplo Indicador Visual Plano Tecelagem

Plano de Tecelagem

Você está atualmente no período 12.

Capacidade Necessária												
Período	13	14	15	16	17	18	19	20	21	22	23	24
Capacidade Disponível	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400
Capacidade Necessária	5615	545	5	5	5	5	5	5	5	5	5	5
Colmeia	900	540	0	0	0	0	0	0	0	0	0	0
Piquet	2000	0	0	0	0	0	0	0	0	0	0	0
Maxim	2200	0	0	0	0	0	0	0	0	0	0	0
Setup	5	5	5	5	5	5	5	5	5	5	5	5
Produtividade	510	0	0	0	0	0	0	0	0	0	0	0

Fonte: Elaborado pelo Autor (2024)

Uma nova funcionalidade implementada para melhorar a usabilidade e a experiência do usuário é o sistema de validação em tempo real nos campos de entrada de dados, conforme ilustrado na imagem acima. Essa funcionalidade assegura que os campos obrigatórios, como o de Produção Planejada, sejam preenchidos antes que as informações possam ser salvas ou processadas. Quando o usuário tenta avançar sem preencher um campo necessário, é exibida uma mensagem clara e visível, alertando-o sobre a necessidade de fornecer os dados requeridos. Esse mecanismo pode ser observado na Figura 31.

Além desse caso, para os campos destinados ao preenchimento de capacidade terceirizada nos diferentes setores, como tecelagem, jets e ramas, foi incorporado um sistema de validação específico. Caso o grupo insira um valor que não seja múltiplo do valor correto para aquele setor, a mensagem de erro exibida não apenas alerta

sobre o problema, mas também sugere os valores mais próximos permitidos, indicando de forma precisa a posição do campo que necessita ser corrigido.

Figura 31: Exemplo Mecanismo de Validação

Fonte: Elaborado pelo Autor (2024)

Esse sistema de validação reduz erros comuns de entrada de dados e garante que as informações críticas para a simulação sejam inseridas corretamente, além de tornar o processo mais intuitivo para o usuário.

#### 4.3.4 Contas de Acesso por Grupo

Cada grupo de jogadores possui uma conta de acesso personalizada, pré-configurada com as informações necessárias para iniciar o jogo, como o tipo de demanda e a configuração inicial da empresa. Essa funcionalidade permite que os grupos entrem no jogo com configurações específicas, alinhadas com as dinâmicas do jogo e os objetivos educacionais. O acesso centralizado evita a necessidade de configurações manuais por parte dos jogadores, otimizando o tempo e garantindo que todos os grupos estejam no mesmo ponto de partida.

O cadastro inicial é realizado pelo administrador, que é responsável por configurar previamente as turmas e os integrantes no Painel de Administrador, cuja funcionalidade será detalhada na subseção seguinte.

Do lado dos usuários, o processo de cadastro do grupo começa na página de *login*, onde é necessário clicar no *link* "Primeiro Acesso", localizado na parte inferior da tela. Na página subsequente, ilustrada pela Figura 32, um dos integrantes do grupo deve completar o processo de registro. Isso inclui a seleção do semestre, a turma, o

tipo de demanda, a inserção do número do grupo, a criação de uma senha para o grupo e o registro dos integrantes que o compõem.

Figura 32: Página Primeiro Acesso

**Primeiro Acesso**

Semestre

Turma

Tipo de Demanda

Número do Grupo

Definir Senha do Grupo

Confirmar Senha do Grupo

Integrantes Seleccionados:

Rodrigo Silva	<input type="button" value="Remover"/>
Roberta Pereira Santos	<input type="button" value="Remover"/>

Fonte: Elaborado pelo Autor (2024)

Após a conclusão desse cadastro, os integrantes dos grupos podem realizar o login utilizando as credenciais criadas para acessar a conta do grupo e, assim, iniciar o jogo.

#### 4.3.5 Painel de Administrador

Foi adicionado um *login* especial para administradores, o que permite a centralização da visualização e análise dos resultados dos grupos sem a necessidade de envio de arquivos externos por partes dos grupos. Com esse acesso exclusivo, o administrador (como um professor, por exemplo) pode acessar a tela de administração, onde é possível visualizar e consultar os resultados de cada grupo em tempo real. A tela de administração funciona também como um painel onde o administrador pode realizar pesquisas no banco de dados e exibir os resultados diretamente na tela, sem precisar de programas externos. Essa funcionalidade centralizada oferece uma visão abrangente dos desempenhos dos grupos, facilitando a supervisão e a análise pedagógica dos resultados além de tornar o jogo mais interativo, acessível e adaptado tanto para o uso dos jogadores quanto para o acompanhamento dos resultados pelo administrador, com o intuito de cumprir seu papel como uma ferramenta educacional e prática.

#### 4.4 INTERFACES DO USUÁRIO

Nesta seção, são apresentadas as interfaces atualizadas das diferentes telas do jogo, ilustrando as melhorias feitas para tornar o sistema mais interativo e intuitivo. As imagens a seguir mostram como os elementos visuais e os controles foram organizados para facilitar o uso e proporcionar uma experiência de navegação mais fluida. As mudanças visam garantir que os usuários tenham acesso rápido e claro às funcionalidades do jogo, contribuindo para um aprendizado mais eficaz e uma interação mais dinâmica com a simulação.

Além disso, as novas interfaces foram projetadas para serem responsivas, permitindo que o jogo funcione adequadamente em dispositivos móveis, como tablets e smartphones. Essa adaptação automática assegura que os elementos se ajustem conforme o necessário, proporcionando uma experiência consistente e acessível, independentemente do dispositivo utilizado.

#### 4.4.1 *Login*

A página de *login*, conforme ilustrada na Figura 33, apresenta uma interface simples e direta, projetada para facilitar o acesso ao jogo.

Figura 33: Página de *Login*

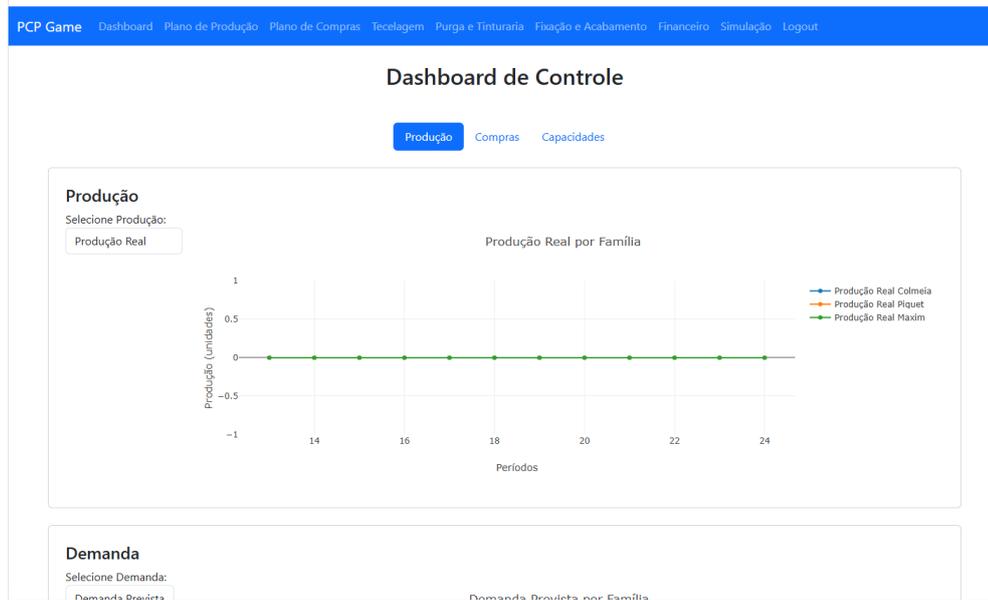


Fonte: Elaborado pelo Autor (2024)

O usuário ao tentar entrar em alguma das páginas do jogo, deverá estar autenticado, caso contrário será redirecionado para esta página.

#### 4.4.2 *Dashboard*

Ao realizar o acesso com os dados fornecidos pelo administrador, o usuário é automaticamente redirecionado para a tela do Dashboard, ilustrada na Figura 34. Essa tela centraliza as informações principais e oferece uma visão geral do desempenho e do progresso no jogo.

Figura 34: Página *Dashboard*

Fonte: Elaborado pelo Autor (2024)

No Dashboard, o usuário encontra gráficos, tabelas e outros elementos visuais que facilitam a análise de dados em tempo real. Essas informações podem incluir, por exemplo, a previsão de demanda ou a demanda real, dados relacionados a compras de matérias primas, ou então as capacidades das máquinas, oferecendo uma visão ampla e integrada do processo produtivo.

Cada componente do Dashboard foi projetado para ser interativo e responsivo, possibilitando que o usuário explore os dados detalhadamente ou filtre informações conforme a necessidade.

#### 4.4.3 Planejamento da Produção

Ao acessar a página Plano de Produção, ilustrada na Figura 35, o usuário encontra os formulários destinados à inserção dos dados necessários. Visualmente, apenas os campos com fundo branco estão disponíveis para edição, enquanto os campos com fundo cinza são exclusivamente para visualização, garantindo uma distinção clara entre as informações que podem ser modificadas e aquelas que são apenas consultadas.

Figura 35: Página Plano de Produção  
Plano de Produção

Você está atualmente no período 12.

Família Colméia												
Período	13	14	15	16	17	18	19	20	21	22	23	24
Demanda Prevista	3638	3585	3732	3561	3371	3296	3608	3429	3485	3639	3505	3482
Demanda Real	0	0	0	0	0	0	0	0	0	0	0	0
Estoques Iniciais	1000	0	0	0	0	0	0	0	0	0	0	0
Produção Planejada	0	0	0	0	0	0	0	0	0	0	0	0
Produção Real	0	0	0	0	0	0	0	0	0	0	0	0
Estoques Finais	0	0	0	0	0	0	0	0	0	0	0	0
Vendas Perdidas	2638	3585	3732	3561	3371	3296	3608	3429	3485	3639	3505	3482
Vendas	1000	0	0	0	0	0	0	0	0	0	0	0

Fonte: Elaborado pelo Autor (2024)

Conforme mencionado anteriormente na Seção 4.3, à medida que os campos editáveis são preenchidos com novos valores, os dados relacionados são atualizados automaticamente graças à funcionalidade implementada em JS. Essa atualização em tempo real facilita a entrada de dados e assegura que todas as informações interdependentes estejam sempre sincronizadas, proporcionando uma experiência de uso mais ágil e precisa.

Nessa mesma página, ao rolar para baixo são encontrados os planos para as outras famílias.

#### 4.4.4 Plano de Compras

A página Plano de Compras, ilustrada na Figura 36, segue uma estrutura semelhante à página Plano de Produção, mas é voltada especificamente para o planejamento das matérias-primas necessárias ao processo produtivo. Nessa interface, o usuário encontra formulários destinados à inserção de dados relacionados às quantidades de matérias-primas, como fios e corantes, necessárias para atender à demanda prevista inserida na página anterior.

Assim como na página de Plano de Produção, apenas os campos com fundo branco estão liberados para edição, enquanto os campos com fundo cinza são apenas para visualização, mantendo uma distinção clara entre os dados que podem ser alterados e os que são exibidos apenas para consulta.

Figura 36: Página Plano de Compras

Plano de Compras

Você está atualmente no período 12.

Fio de Algodão												
Período	13	14	15	16	17	18	19	20	21	22	23	24
Consumo Previsto	0	0	0	0	0	0	0	0	0	0	0	0
Consumo Real	0	0	0	0	0	0	0	0	0	0	0	0
Estoque Iniciais	500	500	500	500	500	500	500	500	500	500	500	500
Compras Planejadas	0	0	0	0	0	0	0	0	0	0	0	0
Compras Reais	0	0	0	0	0	0	0	0	0	0	0	0
Compras Emergenciais	0	0	0	0	0	0	0	0	0	0	0	0
Estoque Finais	500	500	500	500	500	500	500	500	500	500	500	500

Fonte: Elaborado pelo Autor (2024)

E de forma análoga, à medida que o usuário insere ou ajusta os valores das quantidades de matérias-primas, os dados relacionados são automaticamente atualizados. Essa funcionalidade garante que o planejamento de compras esteja sempre alinhado com as necessidades de produção, permitindo ao usuário visualizar imediatamente os impactos das mudanças em tempo real, e facilitando o ajuste de suprimentos de maneira eficaz e dinâmica.

#### 4.4.5 Telas de Capacidade

Para as páginas de Capacidade de cada setor, as mesmas possuem três formulários. Primeiramente tem-se o formulário da Capacidade Necessária, ilustrada pela Figura 37, que serve apenas para exibição, não sendo possível modificar manualmente nenhum dos valores.

Figura 37: Página Capacidade Tecelagem  
Plano de Tecelagem

Você está atualmente no período 12.

Capacidade Necessária												
Período	13	14	15	16	17	18	19	20	21	22	23	24
Capacidade Disponível	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400
Capacidade Necessária	5	5	5	5	5	5	5	5	5	5	5	5
Colméia	0	0	0	0	0	0	0	0	0	0	0	0
Piquet	0	0	0	0	0	0	0	0	0	0	0	0
Maxim	0	0	0	0	0	0	0	0	0	0	0	0
Setup	5	5	5	5	5	5	5	5	5	5	5	5
Produtividade	0	0	0	0	0	0	0	0	0	0	0	0

Fonte: Elaborado pelo Autor (2024)

Abaixo deste formulário, encontram-se os formulários de Capacidade Disponível e Ampliações e Reduções, conforme ilustrado na Figura 38.

Figura 38: Formulários Capacidade Disponível e Ampliações e Reduções

Capacidade Disponível												
Período	13	14	15	16	17	18	19	20	21	22	23	24
Capacidade Disponível	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400
Capacidade Instalada	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400	1400
Número de Turnos	2	2	2	2	2	2	2	2	2	2	2	2
Capacidade Terceirizada	0	0	0	0	0	0	0	0	0	0	0	0

Ampliações e Reduções												
Período	13	14	15	16	17	18	19	20	21	22	23	24
Quantidade	5	5	5	5	5	5	5	5	5	5	5	5
Ampliações	0	0	0	0	0	0	0	0	0	0	0	0
Reduções	0	0	0	0	0	0	0	0	0	0	0	0

Salvar Plano de Tecelagem

Fonte: Elaborado pelo Autor (2024)

No formulário de Capacidade Disponível, o usuário pode ajustar o número de turnos, sendo aceitos apenas os valores 1, 2 ou 3, garantindo que a operação seja planejada de acordo com as restrições estabelecidas. Além disso, é possível modificar a capacidade terceirizada, permitindo que o usuário selecione múltiplos dos valores permitidos com base nas configurações do banco de dados.

No formulário de Ampliações e Reduções, o usuário pode planejar aumentos ou diminuições na capacidade de cada setor. Este formulário possui uma validação que impede o salvamento de qualquer plano em que as quantidades resultem em valores negativos, garantindo que as capacidades sejam sempre consistentes e respeitem as limitações operacionais da produção.

Para o setor de Fixação e Acabamento, que abrange as ramas, o formulário mantém uma estrutura semelhante ao do setor de Tecelagem. No entanto, para o setor de Fixação e Acabamento que utiliza jets, há uma diferença específica nos dois últimos formulários, onde o usuário encontra entradas separadas que permitem planejar ampliações ou reduções para cada tipo de jet individualmente. Essa configuração permite um controle mais detalhado e preciso das capacidades de cada tipo de equipamento, atendendo às necessidades específicas do processo produtivo.

#### **4.4.6 Financeiro**

A página Relatório Financeiro exibe uma visão abrangente dos dados financeiros gerados durante o jogo, permitindo que o usuário acompanhe o desempenho econômico do planejamento de produção ao longo dos períodos simulados. Conforme ilustrado na Figura 39, o relatório é organizado em uma tabela que abrange os períodos 13 a 24, e apresenta uma série de indicadores financeiros essenciais para a avaliação do impacto das decisões tomadas.

Na seção superior da página, há uma série de abas que permitem ao usuário alternar entre diferentes categorias de custos e receitas. Estas abas incluem Custos Fixos, Custos de Compra de Matéria-Prima (MP), Custos de Estoques, Custos de Terceirização, Custos de Capital, Custos de Vendas Perdidas e Receitas de Vendas. Esse layout facilita a navegação e permite uma análise detalhada de cada tipo de custo ou receita, proporcionando uma compreensão mais profunda dos fatores que afetam o resultado financeiro.

Figura 39: Página Relatório Financeiro

**Relatório Financeiro**

Resumo Financeiro
Custos Fixos
Custos de Compra MP
Custos de Estoques
Custos de Terceirização
Custos de Capital
Custos de Vendas Perdidas
Receitas de Vendas

**Resumo Financeiro**

	Período 13	Período 14	Período 15	Período 16	Período 17	Período 18	Período 19	Período 20	Período 21	Período 22	Período 23	Período 24
Custos Fixos	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Custos Compra MP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Custos Estoques	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Custos Terceirização	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Custos Capital	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Custos Vendas Perdidas	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Custos Totais	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Receitas Vendas	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Resultado Operacional	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RO Acumulado	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Fonte: Elaborado pelo Autor (2024)

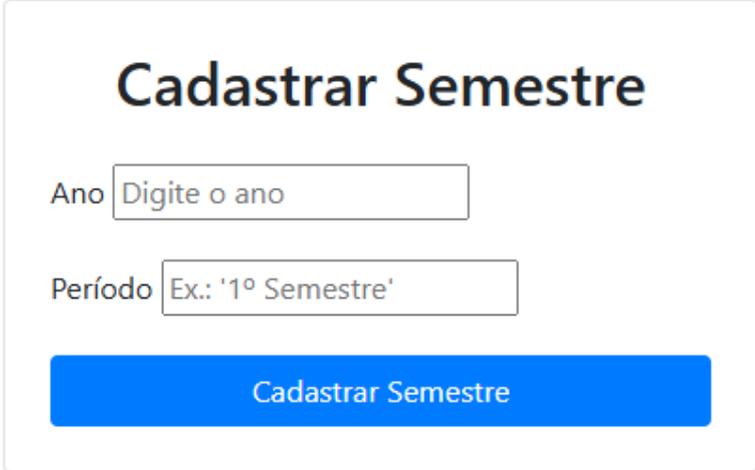
Dentro da tabela principal, cada linha representa um tipo de custo ou receita, como Custos Fixos, Custos de Compra MP, Custos de Estoques, Custos de Terceirização, Custos de Capital, Custos de Vendas Perdidas, Custos Totais, Receitas de Vendas, Resultado Operacional e RO Acumulado. Esses valores são atualizados automaticamente à medida que as decisões do jogador impactam os resultados financeiros, permitindo que o usuário veja os efeitos de suas escolhas ao longo dos períodos.

A disposição clara e organizada dos dados financeiros, com a possibilidade de explorar diferentes categorias de custos e receitas, oferece uma ferramenta poderosa para a análise de desempenho e tomada de decisões. Essa funcionalidade ajuda os usuários a entenderem o impacto econômico de suas estratégias, oferecendo insights sobre a eficiência e a sustentabilidade das operações simuladas no jogo.

#### 4.4.7 Painel Administrador

A nova funcionalidade do Painel de Administrador traz maior controle e flexibilidade ao gerenciamento dos grupos dentro do jogo. Na página de Cadastro de Semestres, conforme ilustrado na Figura 40, o administrador pode criar novos semestres.

Figura 40: Página de Cadastro de Semestres



O formulário 'Cadastrar Semestre' possui o seguinte layout:

- Um campo de texto para 'Ano' com o placeholder 'Digite o ano'.
- Um campo de texto para 'Período' com o placeholder 'Ex.: '1º Semestre''.
- Um botão azul de 'Cadastrar Semestre'.

Fonte: Elaborado pelo Autor (2024)

Na página seguinte, Cadastro de Turmas, ilustrada na Figura 41, o administrador deve cadastrar as turmas vinculadas a um dos semestres já cadastrados. Esse processo inclui a seleção do semestre correspondente, a inserção do nome das turmas e o envio de um arquivo contendo os nomes dos alunos presentes em cada turma. Após o cadastro das turmas, os alunos podem acessar a página de Primeiro Acesso para realizar seus cadastros de forma autônoma, vinculando-se à turma correta e iniciando o processo de criação do grupo para participação no jogo.

Figura 41: Página Cadastro de Turmas



O formulário 'Cadastrar Turmas' possui o seguinte layout:

- Um menu suspenso para 'Semestre' com o valor selecionado '2025 - 1'.
- Um campo de texto para 'Nome da turma'.
- Um botão 'Escolher arquivo' e o texto 'Nenhum arquivo selecionado'.
- Um botão 'Adicionar Turma'.
- Um botão azul de 'Salvar Turmas'.

Fonte: Elaborado pelo Autor (2024)

Por fim a página de Consulta de Usuários e Resultados Financeiros, ilustrada na Figura 42, o administrador tem acesso a uma visão detalhada do progresso e dos resultados financeiros de cada grupo. A interface permite que o administrador filtre os dados com base em diferentes critérios, como o período inicial e final, ou visualizar

todos os grupos de uma vez. A seção "Usuários" exibe informações gerais sobre cada grupo, incluindo o nome do grupo e o período atual. Já a seção "Resultados Financeiros" apresenta uma análise detalhada dos custos e receitas, com categorias como Custos Fixos, Custos de Compra de MP, Custos de Terceirização, Custos de Capital, Custos de Vendas Perdidas, Custos Totais, Receitas de Vendas, e Resultado Operacional.

Figura 42: Página Gerenciar Usuários  
Admin - Gerenciar Usuários

**Consulta de Usuários e Resultados Financeiros**

Critério de Consulta de Usuários:

Todos os Usuários
▼

Período Inicial:  Período Final:

Consultar

**Usuários**

ID	Nome do Grupo	Período Atual
1	Admin	12
2	Grupo Teste	12

**Resultados Financeiros**

Grupo ID	Nome Grupo	Período	Custos Fixos	Custos Compras MP	Custos Terceirização	Custos Capital	Custos Vendas Perdidas	Custos Totais	Receitas Vendas	Resu Ope
2	Grupo Teste	13	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	Grupo	14	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Fonte: Elaborado pelo Autor (2024)

## 5 CONCLUSÃO

Em conclusão, este trabalho alcançou o objetivo de implementar uma versão atualizada do jogo *LSSP\_PCP1*, incorporando as melhorias propostas para tornar a experiência mais interativa, intuitiva e acessível. A modernização da interface, a adição de novas funcionalidades e a estrutura de *backend* flexível, com base nas observações realizadas durante o desenvolvimento, indicam um aprimoramento significativo do potencial educacional do jogo, proporcionando aos usuários uma ferramenta potencialmente mais eficiente e adaptável para o aprendizado de PCP.

O projeto, concebido como uma plataforma de código aberto, continuará em desenvolvimento, possibilitando a participação da comunidade interessada em contribuir com novas ideias e melhorias. Essa abordagem aberta permite não apenas o enriquecimento contínuo da ferramenta, mas também facilita a implementação dos dois jogos restantes da série *LSSP\_PCP*. Além disso, a flexibilidade e modularidade da nova arquitetura possibilitam a adaptação para criar novos jogos focados em outros conteúdos ou áreas de ensino.

Dessa forma, esta plataforma se apresenta não apenas como uma versão aprimorada do jogo original, mas como uma base sólida e expansível para futuras inovações educacionais, beneficiando a comunidade.

Para trabalhos futuros, diversas possibilidades de expansão e aprimoramento da plataforma podem ser consideradas. Uma das principais sugestões é a implementação e unificação dos outros jogos da série *LSSP\_PCP*, integrando-os em uma única plataforma web. Essa unificação permitiria que os usuários tivessem acesso ao conjunto completo dos jogos de simulação de PCP, facilitando o aprendizado contínuo e sequencial das diferentes fases e complexidades do PCP. Com essa abordagem, os jogos poderiam compartilhar dados, aumentando a coesão entre as simulações e proporcionando uma experiência mais integrada.

Além disso, recomenda-se a realização de uma validação mais aprofundada da implementação do jogo web em ambientes educacionais, como em salas de aula de cursos de Engenharia de Produção. Essa validação, por meio de estudos experimentais ou análises de usabilidade com estudantes, permitiria avaliar o impacto das melhorias propostas sobre o aprendizado e a motivação dos alunos. Com o feedback dos usuários em um contexto real de uso, seria possível identificar pontos

de melhoria e realizar ajustes que tornem o jogo ainda mais eficaz como ferramenta pedagógica.

Outras sugestões incluem o desenvolvimento de novos módulos de simulação, abordando tópicos adicionais do PCP ou áreas correlatas, como Gestão de Estoques, Logística ou Controle da Qualidade. A possibilidade de expandir o conteúdo permite que a plataforma evolua como uma ferramenta abrangente de ensino de operações industriais. Além disso, a integração de funcionalidades como dashboards personalizados para análise de desempenho, relatórios detalhados de simulação e ferramentas de benchmarking entre grupos podem aumentar o engajamento e a aplicabilidade da plataforma.

Outra perspectiva interessante é a implementação de recursos gamificados adicionais, como conquistas, rankings e níveis de dificuldade personalizáveis, que podem contribuir para a motivação e o engajamento dos estudantes. A adoção de tecnologias emergentes, como inteligência artificial para fornecer sugestões estratégicas baseadas em análises preditivas e padrões de desempenho dos jogadores, pode agregar ainda mais valor à plataforma. Além disso, o uso de aprendizado de máquina para personalizar o conteúdo do jogo, identificando as principais dificuldades e pontos de melhoria de cada grupo, poderia tornar a experiência mais eficaz e adaptada.

Finalmente, a continuidade do projeto como código aberto facilita a contribuição da comunidade acadêmica e profissional, permitindo que novas funcionalidades e aprimoramentos sejam constantemente incorporados, mantendo a plataforma atualizada e alinhada às demandas educacionais e tecnológicas.

## REFERÊNCIAS

- ANACONDA, **Python Anywhere**. Disponível em: <https://www.pythonanywhere.com/>  
Acesso em: 10 set. 2024.
- ARENALES, Marcos; ARMENTANO, Vinícius Amaral; MORABITO, Reinaldo (org.). **Pesquisa Operacional**. Rio de Janeiro: Elsevier Editora Ltda., 2007. 524 p.  
Disponível em:  
[https://www.academia.edu/36203037/Pesquisa\\_Operacional\\_Arenales\\_et\\_al\\_2007\\_](https://www.academia.edu/36203037/Pesquisa_Operacional_Arenales_et_al_2007_).  
Acesso em: 21 jun. 2023.
- BOCHICCHIO, M.; LONGO, A. **A web-based platform for the design and execution of learning games for education**. *Computers & Education*, v. 57, n. 1, p. 1307-1317, 2011.
- COLLINS, Karen; KAPRALOS, Bill. **Game Re-engineering for Serious Games**. In: *Serious Games and Edutainment Applications*. London: Springer, 2015. p. 93-116.
- CONNOLLY, Thomas; BEGG, Carolyn. **Database Systems: a practical approach to design, implementation, and management**. 6. ed. [S. I.]: Pearson, 2021.
- CORONEL, C; MORRIS, S. **Database Systems: design, implementation, & management**. 12. ed. [S. I.]: Cengage Learning, 2016.
- ELMASRI, Ramez; NAVATHE, Shamkant B. **Fundamentals of Database Systems**. 7. ed. [S. I.]: Pearson, 2016.
- FERNANDES, Flavio Cesar Faria; GODINHO FILHO, Moacir. **Planejamento, Programação, e Controle da Produção: dos fundamentos ao essencial**. São Paulo: Atlas, 2010. 296 p.
- FIELDING, Roy Thomas. **Architectural Styles and the Design of Network-based Software Architectures**. 2000. 162 f. Dissertação (Mestrado) - Curso de Information And Computer Science, University Of California, Irvine, 2000. Disponível em: [https://ics.uci.edu/~fielding/pubs/dissertation/fielding\\_dissertation.pdf](https://ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf). Acesso em: 23 ago. 2024.
- FREIRE, Paulo. **Pedagogia da Autonomia: saberes necessários à prática educativa**. 25. ed. São Paulo: Paz e Terra, 1996.
- FUCHIGAMI, Hélio Yochihiro. **Métodos Heurísticos Construtivos para o Problema de Programação da Produção em Sistemas Flow Shop Híbridos com Tempos de Preparação das Máquinas Assimétricos e Dependentes da Sequência**. 2005. Dissertação (Mestrado em Engenharia de Produção) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2005. doi:10.11606/D.18.2005.tde-25042006-153732. Acesso em: 12 jun. 2023.
- GIL, Antonio Carlos. **Métodos e Técnicas de Pesquisa Social**. 6. ed. São Paulo: Atlas, 2008.

GRINBERG, Miguel. **Flask Web Development**: developing web applications with python. 2. ed. [S. l.]: O'Reilly Media, 2018.

IN, Robert K. **Estudo de Caso**: Planejamento e Métodos. 5. ed. São Paulo: Bookman, 2015.

JUCÁ, S. C. S. A relevância dos softwares educativos na educação profissional. **Ciências & Cognição**, v. 8, 1 mar. 2011.

LEWIS, Michael A.; MAYLOR, Harvey R. Game playing and operations management education. **International Journal Of Production Economics**. Bath, p. 134-149. jan. 2007.

MATOS, Gustavo Costa de. **PCP-TCC-WEB** (Repositório no GitHub) disponível em: <https://github.com/matos1396/PCP-TCC-WEB>. Acesso em: 17 nov. 2024.

MAKRIDAKIS, Spyros; WHEELWRIGHT, Steven Charles; HYNDMAN, Rob John. **Forecasting Methods and Applications** 3. ed. New York: John Wiley & Sons, 1998

MCGONIGAL, Jane. **Reality Is Broken**: Why Games Make Us Better and How They Can Change the World. New York: Penguin Press, 2011.

PREETHIKA. The Architecture of a Web 3.0 application. 2022. Disponível em: <https://www.preethikasireddy.com/post/the-architecture-of-a-web-3-0-application>. Acesso em: 10 nov. 2024.

PRENSKY, Marc. **Digital Game-Based Learning**. 1. ed. New York: McGraw-Hill, 2001.

PRODANOV. **Metodologia do Trabalho Científico**: Métodos e Técnicas da Pesquisa e do Trabalho Acadêmico. 2. ed. Novo Hamburgo, Rio Grande do Sul: Editora Feevale, 2013.

REIGELUTH, Charles Morgan. **Instructional design theories and models**: a new paradigm of instructional theory. New Jersey: Lawrence Erlbaum Associates, 1999. 2 v.

RIBEIRO, L.; GONÇALVES, R.; SANTOS, R. **Responsiveness in web applications**: importance and techniques. *Journal of Web Development and Design*, v. 6, n. 2, p. 45-58, 2019.

ROUSE, M. **Web application**. TechTarget, 2017. Disponível em: [www.techtarget.com](http://www.techtarget.com) Acesso em: 4 nov. 2024.

SAVI, Rafael; ULBRICHT, Vania Ribas. **Jogos Digitais Educacionais**: Benefícios e Desafios. Porto Alegre: Revista Novas Tecnologias na Educação, v. 6, 10 dez. 2008. Quadrimestral. Disponível em: <https://seer.ufrgs.br/index.php/renote/article/view/14405/8310>. Acesso em: 17 jun. 2023.

SHI, H.; WANG, J.; LI, Z. **Application of Database Technology in Educational Games**. Journal of Educational Technology Development and Exchange, v. 9, n. 4, p. 50-60, 2020.

SILBERSCHATZ, A.; SUDARSHAN, S.; KORTH, F. **Database System Concepts**. 7. ed. [S. l.]: McGraw-Hill, 2019.

SLACK, Nigel; CHAMBERS, Stuart; JOHNSTON, Robert. **Administração da Produção**: parte 1. 2. ed. São Paulo: Atlas, 2002. 55 p. Disponível em: <http://folgueral.com.br/producao/arquivos/administracao%20da%20producao/capitulo%201-administracao%20da%20producao.pdf>. Acesso em: 06 jun. 2023.

SMITH, Patricia L.; RAGAN, Tillman J. **Instructional Design**. 3. ed. Hoboken, NJ: John Wiley & Sons, 2005.

SKOMOROCH, M.; MAGNUSSON, D. **Using SQLAlchemy with Flask for a data-driven web application**. Disponível em: <https://cmmorrow.medium.com/using-sqlalchemy-and-flask-to-build-a-simple-data-driven-web-app-17e2d43778bb>. Acesso em: 5 nov. 2024.

TERAO, Amanda Naoimi. **Comparativo de Desempenho de Sistemas de Produção Clássico e com Restrição No-Wait no Ambiente Flow Shop Permutacional em uma Função Bi-Objetivo**. 2017. 69 f. TCC (Graduação) - Curso de Engenharia de Produção, Departamento de Engenharia de Produção, Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2017. Cap. 24. Disponível em: [https://repositorio.utfpr.edu.br/jspui/bitstream/1/16141/1/PG\\_DAENP\\_2017\\_2\\_21.pdf](https://repositorio.utfpr.edu.br/jspui/bitstream/1/16141/1/PG_DAENP_2017_2_21.pdf). Acesso em: 23 jun. 2023.

TIANFIELD, H. **Cloud computing architectures**. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE AND ITS APPLICATIONS, 16., 2016.

TUBINO, Dalvio Ferrari. **Manual de Planejamento e Controle da Produção**. 2. ed. São Paulo: Atlas, 2007. 536 p. Disponível em: [https://edisciplinas.usp.br/pluginfile.php/7263423/mod\\_resource/content/2/LIVRO%20PCP.pdf](https://edisciplinas.usp.br/pluginfile.php/7263423/mod_resource/content/2/LIVRO%20PCP.pdf). Acesso em: 22 jun. 2023.

TUBINO, Dalio Ferrari. **Planejamento e Controle da Produção**: teoria e prática. 2. ed. São Paulo: Atlas, 2009

UNIVERSIDADE FEDERAL DE SANTA CATARINA. **Manual operacional do jogo LSSP\_PCP1**. Florianópolis: Departamento de Engenharia de Produção, Laboratório de Simulação de Sistemas de Produção, [s.d.]. Disponível em: <http://www.deps.ufsc.br/lssp>. Acesso em: 22 abril. 2024.

VALENTE, José Armando (org.). **Computadores e Conhecimento**: repensando a educação. 2. ed. Campinas: Unicamp, 1998. 513 p. Disponível em: <https://www.nied.unicamp.br/wp-content/uploads/other-files/livro-computadores-e-conhecimento.pdf>. Acesso em: 14 maio 2023.

ZALUSKI, Felipe Cavalheiro; OLIVEIRA, Tarcisio Dorn de. **A Utilização de Jogos Como Proposta de Metodologia Ativa: REFLEXÕES DO PROCESSO DE ENSINO E APRENDIZAGEM NO ENSINO SUPERIOR**. 2019. 11 f. Dissertação (Mestrado) - Curso de Administração, Grupo de Pesquisa Espaço Construído, Sustentabilidade e Tecnologias, Universidade Regional do Noroeste do Estado do Rio Grande do Sul, Campo Grande, 2019.