

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO
CURSO DE ENGENHARIA DE COMPUTAÇÃO

ALLAN WENDLAND KRETZMANN

**GERAÇÃO DE GRADES HORÁRIAS ESCOLARES: AUTOMAÇÃO
COM BASE NA META-HEURÍSTICA SIMULATED ANNEALING**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO
2023

ALLAN WENDLAND KRETZMANN

GERAÇÃO DE GRADES HORÁRIAS ESCOLARES: AUTOMAÇÃO COM BASE NA META-HEURÍSTICA SIMULATED ANNEALING

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador: André Yoshiaki Kashiwabara
Universidade Tecnológica Federal do Paraná

**CORNÉLIO PROCÓPIO
2023**



4.0 Internacional

Esta é a mais restritiva das seis licenças principais Creative Commons. Permite apenas que outros façam download dos trabalhos licenciados e os compartilhem desde que atribuam crédito ao autor, mas sem que possam alterá-los de nenhuma forma ou utilizá-los para fins comerciais.

RESUMO

KRETZMANN, Allan. Geração de Grades Horárias Escolares: Automação com base na meta-heurística Simulated Annealing. 2023. 51 f. Trabalho de Conclusão de Curso – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2023.

O *High School Timetabling Problem* consiste na tarefa de alocar aulas a horários letivos em instituições de ensino, considerando restrições relacionadas aos professores, salas e disciplinas. Apesar de todos os avanços tecnológicos que experenciamos na atualidade, esta tarefa ainda é realizada primordialmente de maneira manual nas escolas brasileiras, produzindo grades horárias subótimas e consequentemente trazendo frustrações a todos os envolvidos, sejam estes alunos, docentes ou diretores. Todas as dificuldades relacionadas ao problema, somadas a novos desafios como os Itinerários Formativos do Novo Ensino Médio, pedem uma solução melhor. Este trabalho teve como objetivo implementar um sistema de geração automatizada das grades horárias atendendo o maior número de objetivos possível, através da aplicação de um algoritmo de otimização baseado na meta-heurística *Simulated Annealing*, técnica de otimização inspirada no processo de recozimento de materiais na metalurgia. A solução desenvolvida é uma aplicação *web* que segue uma arquitetura cliente-servidor empregando JavaScript no *frontend* e *backend*, além do otimizador desenvolvido na linguagem C++. Espera-se que a aplicação desenvolvida passe a fazer parte do ferramental de instituições de ensino e auxilie estas a entregar horários escolares que superem os diversos desafios associados e providenciem uma experiência melhor para todos.

Palavras-chave: High School Timetabling Problem. Simulated Annealing. Otimização multiobjetivo.

ABSTRACT

KRETZMANN, Allan. School Timetable Generation: Simulated Annealing based Generation. 2023. 51 f. Trabalho de Conclusão de Curso – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2023.

The High School Timetabling Problem involves the task of assigning classes to class periods in educational institutions while considering constraints related to teachers, classrooms, and subjects. Despite the technological advancements we experience today, this task is still primarily done manually in brazilian schools, resulting in suboptimal timetables and frustrations for all stakeholders, including students, teachers, and administrators. The difficulties associated with the problem, combined with new challenges provided by the new high school model implemented in the country, call for a better solution. This study has aimed to implement an automated timetabling system that satisfies as many objectives as possible by applying an optimization algorithm based on the Simulated Annealing metaheuristic, a technique inspired by the annealing process in metallurgy. The developed solution is a web application following a client-server architecture, employing JavaScript for both the frontend and backend, in addition to the optimizer implemented in the C++ language. We hope that the developed application becomes part of the toolkit for educational institutions and helps them deliver school timetables that overcome the various associated challenges and provide a better experience for all parties involved.

Keywords: High School Timetabling Problem. Simulated Annealing. Multi-objective optimization.

LISTA DE FIGURAS

Figura 1 – Aulas constantes	10
Figura 2 – Restrições	11
Figura 3 – Exemplo de região	13
Figura 4 – Exemplo de grupo de alinhamento sobre a grade horária gerada	14
Figura 5 – Efeito de grupo de alinhamento	15
Figura 6 – Configuração de preferências	17
Figura 7 – Modelo Entidade-Relacionamento	19
Figura 8 – Consultas de validação da modelagem	20
Figura 9 – Consulta SQL de grade horária com sete salas	21
Figura 10 – Resultado da consulta de uma grade no banco de dados	21
Figura 11 – Diagrama de Casos de Uso	22
Figura 12 – Tela - Listagem de Configurações de Grade	23
Figura 13 – Tela - Estrutura da Escola - Professores	24
Figura 14 – Tela - Estrutura da Escola - Turmas e Turnos	25
Figura 15 – Tela - Aulas por professor	25
Figura 16 – Tela - Restrições e Preferências	26
Figura 17 – Tela - Regiões e Grupos de Alinhamento	26
Figura 18 – Tela - Mínimos e Máximos	27
Figura 19 – Tela - Horários	27
Figura 20 – Modelo Entidade-Relacionamento com adição da tabela de usuários	28
Figura 21 – Tela de Acesso	29
Figura 22 – Método de Cadastro	30
Figura 23 – Método de Login	31
Figura 24 – Middleware de autenticação	32
Figura 25 – Middleware de validação da configuração	32
Figura 26 – Modelo Entidade-Relacionamento com Matérias	34
Figura 27 – Estrutura da Escola com Matérias	34
Figura 28 – Tela de configuração de quantidades de aulas por matéria	35
Figura 29 – Visualização de grade horária com matérias	35
Figura 30 – Mensagem de erro de validação	38
Figura 31 – Grade horária exportada para planilha	39
Figura 32 – Grade horária exportada para planilha	40
Figura 33 – Grade horária exportada para planilha	40
Figura 34 – Aulas configuradas na interface	42
Figura 35 – Constantes configuradas na interface	43
Figura 36 – Restrições e preferências configuradas na interface	44
Figura 37 – Regiões e grupos de alinhamento configuradas para o turno da “Manhã”	45
Figura 38 – Regiões e grupos de alinhamento configuradas para o turno da “Tarde”	45
Figura 39 – Mínimos de aulas configuradas na interface	46
Figura 40 – Listagem de grades horárias geradas	47
Figura 41 – Visualização de turno da manhã da melhor solução	47
Figura 42 – Grade exportada - Turno da Manhã	48
Figura 43 – Grade exportada - Turno da Tarde	49

LISTA DE QUADROS

Quadro 1 – Exemplo de dia com aulas separadas.	8
Quadro 2 – Exemplo de dia fragmentado.	9
Quadro 3 – Exemplo de dia com janela.	16
Quadro 4 – Configuração de aulas para validação.	41
Quadro 5 – Métricas de qualidade da melhor solução.	48

LISTA DE ALGORITMOS

Algoritmo 1 – Otimizador de grades inicial	6
Algoritmo 2 – Otimizador com persistência de grades horárias	18
Algoritmo 3 – Otimizador com alocação de matérias	37

SUMÁRIO

1 – INTRODUÇÃO	1
1.1 DELIMITAÇÃO DO TEMA	1
1.2 PROBLEMAS E PREMISSAS	1
1.3 OBJETIVOS	2
1.3.1 Objetivo Geral	2
1.3.2 Objetivos Específicos	2
1.4 ESTRUTURA ORGANIZACIONAL	2
2 – REVISÃO DE LITERATURA	3
2.1 TIMETABLING PROBLEM	3
2.1.1 Highschool Timetabling Problem	3
2.2 NOVO ENSINO MÉDIO	3
2.3 SIMULATED ANNEALING	4
3 – DESENVOLVIMENTO	5
3.1 RECURSOS UTILIZADOS	5
3.2 MÉTODOS	5
3.3 ANÁLISE E IMPLEMENTAÇÃO	6
3.3.1 Otimizador inicial	6
3.3.2 PESOS E RESTRIÇÕES	7
3.3.2.1 Agrupamento de aulas	7
3.3.2.2 Constantes	9
3.3.2.3 Restrições	10
3.3.2.4 Regiões	12
3.3.2.5 Grupos de Alinhamento	14
3.3.2.6 Janelas	16
3.3.2.7 Preferências	16
3.3.2.8 Armazenamento de soluções	17
3.3.3 SERVIDOR E BANCO DE DADOS	19
3.3.4 INTERFACE	22
3.3.5 USUÁRIOS E AUTENTICAÇÃO	27
3.3.5.1 Adaptação do banco de dados	28
3.3.5.2 Tela de login	28
3.3.5.3 Rotas de autenticação	29
3.3.5.4 Middlewares	31
3.3.6 ALOCAÇÃO DE MATÉRIAS	33
3.3.6.1 Alteração no banco de dados	33
3.3.6.2 Alteração de telas	34
3.3.6.3 Alteração de métodos no servidor	35
3.3.6.4 Matérias no otimizador	36
3.3.7 VALIDAÇÃO DE CONFIGURAÇÕES	38
3.3.8 EXPORTAÇÃO DE GRADES HORÁRIAS	39
4 – ANÁLISE E DISCUSSÃO DOS RESULTADOS	41

5 – CONSIDERAÇÕES FINAIS	50
Referências	51

1 INTRODUÇÃO

A cada ano letivo, se faz necessária a definição de grades horárias que atendam inúmeros requisitos para o bom funcionamento das mais diversas instituições de ensino. Estas exigências dependem e impactam diretamente os envolvidos, sejam estes professores, alunos, gestores, ou ainda, os responsáveis pelo planejamento dos horários. Os requisitos citados anteriormente podem mudar durante o ano, requerendo constantes atualizações das grades definidas. Esta tarefa, segundo [Bardadym \(1996\)](#), requer grande esforço quando realizada manualmente, frequentemente resultando em soluções subótimas.

Para o desenvolvimento dos horários escolares, vários critérios devem ser considerados, como o número de aulas de cada professor, salas, turnos, além das restrições associadas a cada professor em particular. A combinação destes fatores para a produção de um horário otimizado é conhecida como *school timetabling problem* ([FONSECA; SANTOS; CARRANO, 2016](#)). Para a escrita deste trabalho, foi realizada a leitura de artigos que abordam o tema em questão, neste modo os autores [Fonseca, Santos e Carrano \(2016\)](#), [Tan et al. \(2021\)](#) e [Abramson \(1991\)](#) se mostraram imprescindíveis para a sua realização.

Ao decorrer da revisão bibliográfica, foi reafirmada a relevância desta pesquisa e da elaboração de um projeto que solucione o problema, dada a complexidade e recorrência deste. [Poulsen \(2012\)](#) estima que a maioria das instituições de ensino brasileiras efetua essa tarefa de maneira manual, o que implica em atrasos para as escolas, além de desgaste entre os docentes envolvidos nas negociações de restrições. Isto justifica novamente a necessidade de uma solução automatizada que entregue grades otimizadas em um tempo hábil.

Diante deste contexto, fica evidente a necessidade de solucionar o seguinte problema: como automatizar a geração de grades horárias que atendam as diversas demandas das instituições de ensino? Por ser um problema combinatório, consideram-se técnicas viáveis *Simulated Annealing*, Busca Tabu, Algoritmos Genéticos, entre outros ([TAN et al., 2021](#)).

1.1 DELIMITAÇÃO DO TEMA

O presente trabalho visa abordar o tema de otimização de grades horárias escolares, tendo em mente especificamente as necessidades de instituições brasileiras de ensino fundamental e médio.

1.2 PROBLEMAS E PREMISSAS

Como mencionado anteriormente, as instituições de ensino necessitam de constantes atualizações de suas grades horárias. Este processo envolve muitas variáveis, e impacta diretamente alunos, professores e gestores. Devido a esta complexidade, o problema já é conhecido na literatura como *school timetabling problem*, sendo dividido nos tipos *exam timetabling*, *course timetabling* e *high school timetabling* ([TAN et al., 2021](#)). Este trabalho tem como foco esta última variação, que, segundo os mesmos autores, é definido em termos da disponibilidade dos professores, número de salas, número de aulas por professor em cada sala e restrições.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Desenvolver uma aplicação que automatize e simplifique o processo de criação de grades horárias escolares, otimizando diferentes características de qualidade destas, fazendo com que tais sejam livres de conflitos e atendam a restrições impostas pelas instituições de ensino.

1.3.2 Objetivos Específicos

1. Compreender quais são as reais necessidades das escolas quanto ao planejamento de grades horárias
2. Implementar métricas de qualidade para os horários, que refletem se as necessidades são atendidas com êxito
3. Aplicar um algoritmo de otimização na geração de grades que evite conflitos, atenda restrições de horários dos docentes, minimize a quantidade de janelas nas aulas dos professores, faça agrupamentos formando aulas duplas e evite excessos da mesma aula
4. Proporcionar uma experiência do usuário simples e agradável durante o uso da aplicação

1.4 ESTRUTURA ORGANIZACIONAL

O trabalho em questão está estruturado da seguinte forma: o capítulo 1 traz a introdução sobre o tema, uma visão geral sobre a pesquisa e seus objetivos; no capítulo 2 será desenvolvida a fundamentação teórica, listando todas as fontes de conhecimento consultadas para o desenvolvimento do projeto; a forma de implementação do projeto, seus materiais e métodos serão discutidos no capítulo ??; o capítulo ?? apresentará o cronograma das atividades planejadas e por fim, o capítulo 5 trará as considerações finais.

2 REVISÃO DE LITERATURA

Este capítulo tem como objetivo explicar o problema de *timetabling* e suas especificidades no caso de grades horárias de ensino médio, assim como a técnica de otimização *Simulated Annealing* e sua aplicação ao problema.

2.1 TIMETABLING PROBLEM

O problema de *timetabling* ou alocação de horários é definido em termos de quatro conjuntos, sendo estes horários, recursos, encontros e restrições. O problema consiste em associar os encontros desejados aos horários, utilizando os recursos disponíveis e minimizando as violações das restrições.

É evidente que existem inúmeras tarefas e indústrias que dependem da alocação de horários, portanto é necessário especializar o problema. Uma das variações deste problema é o *school timetabling problem*, que segundo [Tan et al. \(2021\)](#) subdivide-se em *exam timetabling*, *course timetabling* e *high school timetabling*. Esta última subdivisão, relacionada a escolas de ensino médio, será o foco deste trabalho.

2.1.1 Highschool Timetabling Problem

No caso específico da alocação de horários para escolas de ensino médio, o problema é definido em termos da disponibilidade dos professores, número de salas, número de aulas por professor em cada sala e restrições. Adequando estas necessidades à formulação do *timetabling problem*, os encontros que desejamos alocar são entre os professores e determinada sala de aula, dados os horários de aula em que a escola opera, visando a utilização dos recursos disponíveis, como salas de aula ou laboratórios, minimizando a violação de restrições associadas aos professores e recursos ([TAN et al., 2021](#)).

Segundo [Abramson \(1991\)](#), o problema envolve agendar aulas, professores e salas de aula de tal forma que nenhum professor, turma ou sala de aula seja utilizado mais de uma vez em determinado horário. Situações em que ocorra este agendamento de recursos duplicados serão tratadas como “conflitos” neste trabalho.

Levando em consideração a necessidade de agendar diversas aulas evitando conflitos, e o atendimento de restrições variadas, o *highschool timetabling problem* é um problema de otimização multiobjetivo, que segundo [Cooper e Kingston \(1995\)](#) é da classe NP-completo.

Considerando a natureza complexa do problema, este trabalho propõe desenvolver e analisar a eficiência de um algoritmo de *Simulated Annealing*, assim como implementar uma interface para interação com este.

2.2 NOVO ENSINO MÉDIO

O Novo Ensino Médio consiste na atualização das matrizes curriculares das salas desta etapa do sistema educacional (1º, 2º e 3º anos). Formulado através da Lei nº 13.415/2017 ([BRASIL, 2017](#)), o novo modelo traz ainda mais desafios para a tarefa de organização de grades horárias escolares.

O principal destes novos desafios no contexto da criação de grades horárias é o conceito dos Itinerários Formativos, conjuntos de atividades que os estudantes podem escolher realizar durante o ensino médio, como disciplinas, projetos, oficinas, entre outros.

No caso de grandes instituições de ensino, é possível que seja viável oferecer essas atividades optativas através da criação de turmas específicas, entretanto, em escolas de pequeno porte, a adição destas turmas e quantidade relativamente pequena de estudantes por turma pode facilmente desgastar os recursos da instituição.

Tendo estas limitações em mente, observou-se que uma estratégia aplicada por escolas de porte menor é a oferta de disciplinas optativas (Itinerários Formativos) em horários simultâneos em diferentes salas, nos períodos normais de ensino. No momento em que essas aulas acontecem, cada aluno das duas turmas se locomove para a sala adequada onde a aula que escolheu será ministrada, e com isso a escola não precisa alocar salas, professores ou horários adicionais.

Evidentemente, alocar horários simultâneos para determinadas aulas dificulta ainda mais o planejamento do horário escolar, sendo assim mais uma questão que pode ser auxiliada por um sistema de otimização da grade.

2.3 SIMULATED ANNEALING

Diversos problemas podem ser resolvidos imitando o que ocorre em fenômenos físicos ou naturais. Segundo [Laarhoven \(1987\)](#), *Simulated Annealing* é uma meta-heurística que se inspira no processo de recocimento na área da metalurgia. Este processo tem como objetivo reduzir as tensões internas de determinado material, da seguinte forma:

1. O material inicia com uma alta temperatura, com seus átomos desordenados e livres para vibrarem e se deslocarem
2. O sistema é resfriado gradualmente, fazendo com que os átomos encontrem posições cada vez mais estáveis. Em outras palavras, conforme a temperatura diminui, torna-se cada vez mais improvável um átomo se deslocar para uma posição menos estável
3. Por fim, atinge-se certa temperatura em que não ocorrem mais mudanças significativas no material.

Esta meta-heurística foi escolhida para o desenvolvimento deste trabalho considerando seu potencial verificado durante a revisão de literatura, e a clareza dos paralelos que podem ser realizados com o processo de têmpera real: os átomos representam os professores e aulas, que podem ser deslocados na grade horária até que atinjam uma posição estável, ou seja, que viole o mínimo possível de restrições do problema.

3 DESENVOLVIMENTO

Este capítulo tem como objetivo evidenciar todas as atividades executadas durante o desenvolvimento do projeto. A [Seção 3.1](#) apresentará uma listagem dos *softwares* utilizados no desenvolvimento do projeto; a [Seção 3.2](#) explicitará o método de desenvolvimento, e a [Seção 3.3](#) evidenciará a execução das atividades propriamente ditas.

3.1 RECURSOS UTILIZADOS

Tratando-se do desenvolvimento de um projeto *full stack*, utilizaram-se diversas tecnologias e ferramentas, sendo as de maior importância:

- **VSCode:** Editor de código-fonte desenvolvido pela Microsoft, escolhido por sua versatilidade e aplicabilidade para desenvolver todos os componentes do sistema
- **DBeaver:** Ferramenta de código aberto de administração de bancos de dados, utilizada para validação da modelagem e interações manuais com a base de dados
- **PostgreSQL:** Sistema gerenciador de banco de dados relacional, escolhido por sua robustez, possibilitará a persistência dos dados da aplicação em tabelas
- **JavaScript:** Linguagem de programação ubíqua na engenharia de software, escolhida por sua versatilidade para desenvolver a interface *web* e servidor
- **VueJS:** *Framework* para desenvolvimento frontend na linguagem JavaScript, escolhido por sua versatilidade e familiaridade do graduando com este
- **VuetifyJS:** *Framework* de componentes e estilização para o VueJS, utilizado para padronizar o design da aplicação
- **NodeJS:** *Runtime* de JavaScript que permite a utilização dessa linguagem para escrever aplicações no lado do servidor;
- **ExpressJS:** *Framework* para desenvolvimento backend na linguagem JavaScript, utilizando a plataforma NodeJS
- **C++:** Linguagem de programação compilada, escolhida por sua alta performance para o desenvolvimento do otimizador
- **G++:** Compilador utilizado para converter o código fonte em C++ do otimizador para um arquivo executável
- **GDB:** Debugador da linguagem C++, utilizado para localizar problemas no código fonte do otimizador

3.2 MÉTODOS

Para o processo de software da aplicação, optou-se pelo processo de desenvolvimento incremental. Este processo consiste na divisão da implementação do projeto em incrementos, os quais são executados linearmente, porém de forma escalonada.([PRESSMAN; MAXIM, 2016](#))

Este processo de software foi escolhido por possibilitar o planejamento de execução do projeto em etapas lógicas, que podem ser sobrepor através de escalonamento, de acordo com as necessidades encontradas. Conforme os requisitos levantados na seção anterior, dividiram-se as tarefas de desenvolvimento nos incrementos pertinentes:

- **Incremento 1:** Desenvolvimento inicial do otimizador, aplicando Simulated Annealing apenas para a resolução de conflitos;
- **Incremento 2:** Implementação de restrições no otimizador;
- **Incremento 3:** Criação do servidor e banco de dados;

- **Incremento 4:** Desenvolvimento da interface web;
- **Incremento 5:** Adição do sistema de usuários;
- **Incremento 6:** Adaptação da modelagem para incluir matérias aos horários alocados pelo otimizador;
- **Incremento 7:** Implementação de validações das configurações de grade inseridas pelo usuário;
- **Incremento 8:** Desenvolvimento de sistema de exportação de grades horárias.

3.3 ANÁLISE E IMPLEMENTAÇÃO

3.3.1 Otimizador inicial

Segundo Abramson (1991), uma parte fundamental para a produção de grades horárias escolares é a resolução de conflitos de recursos. Estes conflitos são caracterizados por recursos alocados simultaneamente, levando a grades horárias não aplicáveis na prática. Um exemplo disso é a alocação de um professor para ministrar aulas em mais de uma turma ao mesmo tempo, o que é impossível.

Neste projeto, a primeira parte desenvolvida foi a versão inicial do otimizador, cuja tarefa era gerar uma grade horária válida, evitando conflitos, ou seja, professores alocados para mais uma turma ao mesmo tempo. Para realizar esta tarefa, aplicou-se a meta-heurística de *simulated annealing*, originando o algoritmo 1.

Algoritmo 1: Otimizador de grades inicial

Input: Lista de professores LP , lista de turmas LT , matriz de aulas por professor por turma MA , temperatura inicial TI , Taxa de resfriamento TR

Output: Grade horária de professores otimizada

$temperatura \leftarrow TI$

$grade \leftarrow \text{CriaGradeInicial}(LP, LT, MA)$

$minConflitos \leftarrow \text{NumeroConflitos}(grade)$

while condição de parada não atingida **do**

for passo = 0 até numeroPassos **do**

$turma \leftarrow \text{EscolheTurmaAleatoria}()$

$linhas \leftarrow \text{EscolheHorariosAleatoriosValidos}(sala)$

$delta \leftarrow \text{CalculaDelta}(sala, linhas)$

$probabilidade \leftarrow e^{-delta/temperatura}$

$valorAceite \leftarrow \text{Aleatorio}(0, 1)$

if $delta < 0$ ou $probabilidade \geq valorAceite$ **then**

$\text{PermutaProfessores}(sala, linha1, linha2)$

if $\text{NumeroConflitos}(grade) < minConflitos$ **then**

$\text{Imprime}(grade)$

$minConflitos \leftarrow \text{NumeroConflitos}(grade)$

end

end

end

$temperatura \leftarrow temperatura * TR$

end

No algoritmo 1, o valor da taxa de resfriamento é de 0,99, sendo a temperatura inicial e o número de passos por iteração escolhidos empiricamente. Um valor apropriado

para a temperatura inicial também pode ser definido empiricamente, ou utilizando o método explicitado por Abramson et al. (1999), em que é calculada a temperatura em que as soluções passam por uma transição de fase.

Além disso, durante o desenvolvimento deste primeiro incremento, utilizou-se como condição de parada o esgotamento da temperatura, ou seja, o algoritmo finaliza sua execução assim que a temperatura atinge um valor próximo de zero, quando não ocorrem mais permutações de professores.

Explicando melhor o algoritmo, o método “CriaGradeIncial” gera uma matriz com as turmas e números de aulas de cada professor alocados corretamente. Esta grade inicial provavelmente possui inúmeros conflitos, portanto são aplicados os passos de otimização.

Para cada passo de otimização, são escolhidas aleatoriamente uma turma e duas linhas (posições) da grade horária. Com estas informações, é calculada a variação do número de conflitos que a permutação dos professores nas linhas escolhidas ocasionaria. De acordo com este valor de variação, é determinado se a troca dos professores deve ou não ser realizada: uma troca que diminua o número de conflitos sempre é aceita, enquanto uma troca que aumenta o número de conflitos pode ser aceita probabilisticamente, de acordo com o valor da temperatura na iteração atual.

3.3.2 PESOS E RESTRIÇÕES

Apesar da importância da resolução de conflitos, existem diversas outras nuances durante o planejamento das grades horárias que precisam ser levadas em conta para que as grades produzidas sejam aplicáveis na prática.

Abramson (1991) sugere uma forma de permitir a otimização de múltiplas características simultaneamente utilizando: uma função de custo ponderado. Esta função foi implementada, utilizando-se um sistema de métricas, cada qual mensura quantitativamente determinada característica da grade horária, e contém um peso que define a sua importância para a grade como um todo.

A função de custo ponderado retorna um valor que representa a qualidade geral de uma grade horária, através do cálculo da média ponderada de todas as métricas de qualidade. Em outras palavras, quanto menor o custo, melhor a solução encontrada.

Adicionalmente, as métricas foram implementadas de forma que pudessem ser rígidas ou suaves. As métricas rígidas precisam ser perfeitamente atendidas para que a grade horária seja considerada viável, enquanto as métricas suaves promovem a melhoria da qualidade, mas não necessariamente precisam ser perfeitamente atendidas.

As próximas seções terão como objetivo explicar cada uma destas métricas e os desafios associados. A Subseção 3.3.2.8 entrará em mais detalhes sobre como as grades horárias passaram a ser salvas após a implementação das métricas.

3.3.2.1 Agrupamento de aulas

Observando grades horárias escolares existentes, é possível notar que existe uma motivação para realizar agrupamentos, formando aulas duplas. Isso aumenta a produtividade das aulas, à medida que diminui trocas e deslocamento de professores entre as salas. Para produzir estes agrupamentos, foram adicionadas algumas métricas que fazem o otimizador penalizar:

1. Aulas separadas;
2. Aulas desagrupadas;
3. Excessos de aulas iguais para determinada turma em um dia;

4. Dias com todas aulas planejadas distintas para determinada turma;
Para contextualizar estas situações, a seguir serão apresentados alguns quadros.

Quadro 1 – Exemplo de dia com aulas separadas.

Aula	Professor	Matéria
1	Marcos	Matemática
2	Fábio	Física
3	Marcos	Matemática
4	Luciana	Português
5	Luciana	Português
6	Luciana	Português

Fonte: Autoria própria

Observando o quadro 1, é possível notar:

- Aulas separadas: 1, 2 e 3, visto que não estão em nenhum agrupamento;
- Aulas desagrupadas: 1 e 3, pois consistem em múltiplas aulas iguais, no mesmo dia, que não foram agrupadas;
- Excessos de aulas: 4, 5, 6, pois consiste em uma aula tripla, algo considerado não desejável neste trabalho

Por fim, a situação de dias com todas aulas diferentes, que serão referidos como "Dias fragmentados" neste trabalho, pode ser observada no quadro 2.

Quadro 2 – Exemplo de dia fragmentado.

Aula	Professor	Matéria
1	Marcos	Matemática
2	Fábio	Física
3	Luciana	Português
4	Roberto	Geografia
5	Renato	História

Fonte: Autoria própria

3.3.2.2 Constantes

Durante o desenvolvimento, concebeu-se o conceito de “Aulas constantes”, como aulas que absolutamente devem ser alocadas em determinada posição da grade horária. Isto é útil para guiar o otimizador rumo a uma solução desejada, quando já são conhecidas algumas aulas que devem ser fixas.

Como exemplo de caso de uso, uma escola com seis aulas diárias pode ter alguns dias da semana com menos aulas, e pode ser interessante definir explicitamente quais dias devem ter a última aula da grade horária não alocada (vazia). A figura 1 mostra um exemplo de configuração de aulas constantes para determinada grade horária.

Figura 1 – Aulas constantes

Constantes		6 Ano	1 EM	7 Ano	8 Ano	9 Ano
	Dia	Aula				
Segunda-feira	1	Verônica				
	2	Verônica				
	3					
	4					
	5					
	6					
Terça-feira	1	Luciana	Adriana			
	2	Luciana	Adriana			
	3					
	4					
	5					
	6					
Quarta-feira	1					
	2					
	3					
	4					
	5					
	6					
Quinta-feira	1					
	2					
	3					
	4					
	5					
	6					
Sexta-feira	1					
	2					
	3					
	4					
	5					
	6					

Fonte: Autor

Considerando a natureza absoluta das aulas constantes, estas não são consideradas um métrica de qualidade, e não possuem um peso próprio. Em vez disso, no método “GeraGradeInicial” do algoritmo 1, o otimizador já aloca as aulas constantes nas posições desejadas, e a função “EscolheHorariosAleatoriosValidos” não seleciona posições que estejam alocadas com aulas constantes. Desta forma, um vez que as aulas constantes são posicionadas na grade horária, estas nunca são movidas durante os passos de otimização.

3.3.2.3 Restrições

As restrições representam o oposto das aulas constantes: posições em que determinadas aulas não devem ser alocadas. Implementaram-se no otimizador restrições suaves e rígidas,

cada uma podendo também receber um peso customizado. Dessa forma, é possível configurar o otimizador para nunca alocar determinada aula em certa posição da grade, ou apenas evitar isso.

Como exemplo de uso dessa funcionalidade, a figura 2 demonstra uma configuração de restrições para determinado professor que não pode ser alocado nas duas últimas aulas de qualquer dia da grade horária.

Figura 2 – Restrições

		Restrições						
		Dia	Aula	6 Ano	1 EM	7 Ano	8 Ano	9 Ano
Segunda-feira	1							
	2							
	3							
	4							
	5							
	6							
	1							
Terça-feira	2							
	3							
	4							
	5							
	6							
	1							
	2							
Quarta-feira	3							
	4							
	5							
	6							
	1							
	2							
	3							
Quinta-feira	4							
	5							
	6							
	1							
	2							
	3							
	4							
Sexta-feira	5							
	6							

Fonte: Autor

No caso das restrições, a métrica associada mensura a quantidade de restrições violadas, ou seja, posições em que determinada aula foi alocada, mas não deveria ter sido.

3.3.2.4 Regiões

O conceito de regiões foi concebido como uma forma de proporcionar ainda mais controle ao usuário, sobre o posicionamento das aulas na grade horária. Cada região consiste em um grupo arbitrário de posições da grade horária, associado a uma regra relacionada a uma quantidade de aulas. Com as regiões, é possível determinar mínimos, máximos ou quantidades exatas de aulas que devem ser alocadas em certas posições da grade horária.

Como exemplo de uso das regiões, a figura 3 demonstra uma configuração utilizada para assegurar que em todos os dias da grade horária, o professor tenha alguma aula alocada no primeiro horário.

Figura 3 – Exemplo de região

Região (Verônica - Exatamente 5 aulas)						
Dia	Aula	6 Ano	1 EM	7 Ano	8 Ano	9 Ano
Segunda-feira	1					
	2					
	3					
	4					
	5					
	6					
Terça-feira	1					
	2					
	3					
	4					
	5					
	6					
Quarta-feira	1					
	2					
	3					
	4					
	5					
	6					
Quinta-feira	1					
	2					
	3					
	4					
	5					
	6					
Sexta-feira	1					
	2					
	3					
	4					
	5					
	6					

Fonte: Autor

A região configurada na 3 pode ser interpretada da seguinte forma: a professora “Verônica” deve ter exatamente cinco aulas alocadas dentro das posições marcadas pela cor azul. Como o otimizador não permite conflitos, cada uma das cinco aulas deverá ser alocada em um dos diferentes dias, garantindo que a professora terá uma aula agendada no primeiro horário de cada um dos dias.

Neste caso, a métrica associada mensura o erro das regiões, ou seja, a diferença entre

a quantidade de aulas esperada de acordo com a regra de cada região e a quantidade real de aulas alocadas.

3.3.2.5 Grupos de Alinhamento

Os grupos de alinhamento representam uma forma de configurar o otimizador para agendar aulas para diferentes turmas, nos mesmos horários. Isto é útil para atender a restrições relacionadas ao conceito dos Itinerários Formativos do Novo Ensino Médio, conforme citado na Seção 2.2.

Como exemplo de utilização dos grupos de alinhamentos, tem-se a configuração da figura 4.

Figura 4 – Exemplo de grupo de alinhamento sobre a grade horária gerada

Grupos de Alinhamento			NOVO GRUPO		
6 Ano		7 Ano			
Professor	Aulas	Ações	Professor	Aulas	Ações
Verônica	3		Adriana	3	

Fonte: Autor

O efeito da configuração do grupo de alinhamento da figura 4, pode ser observado na grade horária final gerada na figura 5, com as aulas corretamente alocadas em horários simultâneos.

Figura 5 – Efeito de grupo de alinhamento

Dia	Aula	6 Ano	7 Ano	
Segunda-feira	1	Jack	Português	História
	2	Jack	Português	História
	3	Renata	História	Verônica
	4	Verônica	Geografia	Adriana
	5	Verônica	Geografia	Adriana
	6			
Terça-feira	1	Jack	Português	Artes
	2	Jack	Português	Artes
	3	Jéssica	Inglês	Rosa
	4	Rosa	Matemática	Luciana
	5	Rosa	Matemática	Beatriz
	6	Beatriz	ED. Física	
Quarta-feira	1	Renata	História	Jack
	2	Renata	História	Jack
	3	Rosa	Matemática	Renata
	4	Luciana	Of. Texto	Rosa
	5	Verônica	Geografia	Adriana
	6			
Quinta-feira	1	Adriana	Ciências	Espanhol
	2	Adriana	Ciências	Português
	3	Jack	Português	Filosofia
	4	Jéssica	Inglês	Rosa
	5	Jéssica	Filosofia	Verônica
	6			Verônica
Sexta-feira	1	Rosa	Matemática	Inglês
	2	Rosa	Matemática	Inglês
	3	Jack	Espanhol	Matemática
	4	Adriana	Ciências	Matemática
	5	Cristiane	Artes	Português
	6	Cristiane	Artes	Português

Fonte: Autor

Implementaram-se duas métricas de qualidade relacionadas aos grupos de alinhamento, uma que mensura a quantidade de grupos formados e a quantidade de aulas “desalinhadas”, ou seja, aulas que deveriam ser agendadas no mesmo horário, mas que não foram.

No caso da [5](#), foram formados dois grupos (demarcados em vermelho), e nenhuma aula ficou desalinhada, ou seja, todas as seis aulas configuradas no grupo de alinhamento foram corretamente alocadas em horários simultâneos.

3.3.2.6 Janelas

As janelas consistem em situações em que a jornada de trabalho de determinado professor apresenta um horário sem aulas alocadas, fazendo com que o docente fique ocioso. O quadro [3](#) exemplifica esta situação.

Quadro 3 – Exemplo de dia com janela.

Aula	6ºAno	7ºAno	8ºAno
1	Jéssica	Rosa	Adriana
2	Jéssica	Rosa	Adriana
3	Fábio	Jéssica	Rosa
4	Adriana	Jéssica	Rosa
5	Beatriz	Adriana	Jéssica

Fonte: Autoria própria

No quadro [3](#), a professora Adriana tem uma janela na terceira aula, visto que tem aulas alocadas antes e depois (aulas 1, 2, 4 e 5). Em contrapartida, apesar de não ter nenhuma aula planejada no quinto horário, Rosa não tem janelas neste exemplo, pois pode encerrar sua jornada de trabalho na quarta aula.

A métrica de janelas mensura o número de horários na grade que apresentam estes horários ociosos, para cada professor.

3.3.2.7 Preferências

As preferências consistem em configurações que informam ao otimizador características preferíveis para a alocação das aulas de cada professor. Implementaram-se os seguintes tipos de preferência no otimizador:

1. Preferência de primeiras aulas;
2. Preferência de últimas aulas;
3. Preferência específica de última aula

A [Figura 6](#) representa um exemplo de configuração de preferências para determinado professor. Neste caso, a configuração informa ao otimizador que na segunda-feira, as aulas de “Verônica” devem ser alocadas desde o início do dia, não ultrapassando um total de aulas; e que na turma do “6º Ano”, caso haja uma aula dessa professora, esta deve ser alocada na última aula.

Vale ressaltar que o diferencial da preferência específica de última aula é que esta considera a possibilidade de aulas vazias, ou seja, caso o último horário do dia não tenha uma aula alocada, a preferência considera que a aula deve ser alocada na penúltima aula.

Figura 6 – Configuração de preferências

Preferências						Verônica
Dia	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	
Segunda-feira	2 primeiras aulas	2 primeiras aulas	2 primeiras aulas	2 primeiras aulas	2 primeiras aulas	
Terça-feira	Forçar última aula					
Quarta-feira						
Quinta-feira						
Sexta-feira						

Fonte: Autor

3.3.2.8 Armazenamento de soluções

Com a adição das métricas, o critério para considerar uma grade horária como uma solução válida ficou mais estrito, e em muitos casos tornou-se impossível obter uma grade que atenda perfeitamente todas as nuances. Tendo isto em mente, foi necessário implementar uma funcionalidade de armazenamento de grades horárias um pouco mais detalhada. O resultado disso, pode ser visto no algoritmo 2.

Algoritmo 2: Otimizador com persistência de grades horárias

Input: Lista de professores LP , lista de turmas LT , matriz de aulas por professor por turma MA , temperatura inicial TI , Taxa de resfriamento TR

Output: Grade horária de professores otimizada

```

temperatura ← TI
grade ← CriaGradeInicial( $LP, LT, MA$ )
iteracoesSemAlteracao ← 0
solucoes ← lista vazia
while condição de parada não atingida do
    deltaTotal ← 0
    for passo = 0 até numeroPassos do
        turma ← grade.EscolheTurmaAleatoria()
        linhas ← grade.EscolheHorariosAleatoriosValidos(sala)
        delta ← grade.CalculaDelta(sala, linhas)
        probabilidade ←  $e^{-\delta/\text{temperatura}}$ 
        valorAceite ← Aleatorio(0, 1)
        if delta < 0 ou probabilidade ≥ valorAceite then
            grade.PermutaProfessores(sala, linha1, linha2)
            deltaTotal ← deltaTotal + delta
            if grade não existe na lista de soluções then
                insere grade na lista de soluções
                limita lista de soluções às 100 melhores grades
            end
        end
    end
    if delta = 0 then
        iteracoesSemAlteracao ← iteracoesSemAlteracao + 1
    else
        iteracoesSemAlteracao ← 0
    end
    if iteracoesSemAlteracao ≥ 15 then
        salvaGradesRelevantes()
        apaga lista de soluções
        temperatura ← TI
        iteracoesSemAlteracao ← 0
    end
    temperatura ← temperatura * TR
end

```

Considerando as diversas métricas de qualidade que as grades horárias passaram a ter, o método “salvaGradesRelevantes” do algoritmo 2 salva, para cada métrica escolhida, as 5 grades que tiveram a melhor pontuação em cada métrica. A princípio, optou-se por utilizar as métricas de qualidade geral (média ponderada de todas as métricas), janelas, agrupamento de aulas e número de preferências resolvidas, mas este método pode ser expandido para qualquer uma das métricas implementadas no sistema.

3.3.3 SERVIDOR E BANCO DE DADOS

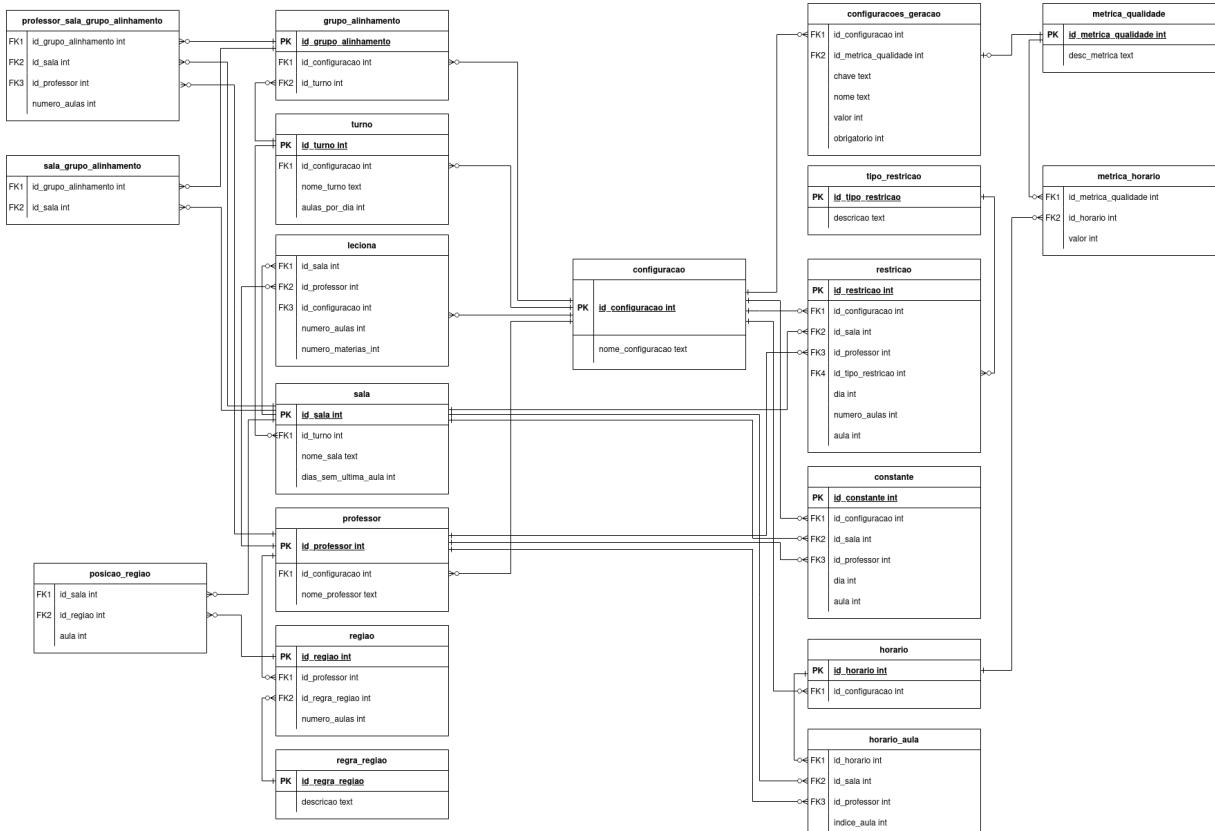
A aplicação do lado do servidor é responsável por receber as requisições da interface, persistir as configurações no banco de dados e realizar a comunicação com o otimizador, a fim de produzir e armazenar as grades horárias.

Para o desenvolvimento deste componente, optou-se pelo framework *Express.js*, a ser executado na plataforma *Node.js*, devido à simplicidade de implementação que estas tecnologias proporcionam. Em relação ao banco de dados, será utilizado o sistema de gerenciamento de banco de dados *PostgresSQL*, devido à sua robustez e grande uso no mercado.

Conforme as premissas do problema sendo tratado, a modelagem do banco de dados é centrada na entidade “Configuração”, que agrupa as configurações de determinada instituição de ensino para a geração de suas grades horárias. Cada uma dessas entidades tem turnos, turmas, professores, e as configurações de quantas aulas cada professor deve ministrar para cada turma, e as respectivas restrições.

A modelagem comentada está representada na Figura 7:

Figura 7 – Modelo Entidade-Relacionamento

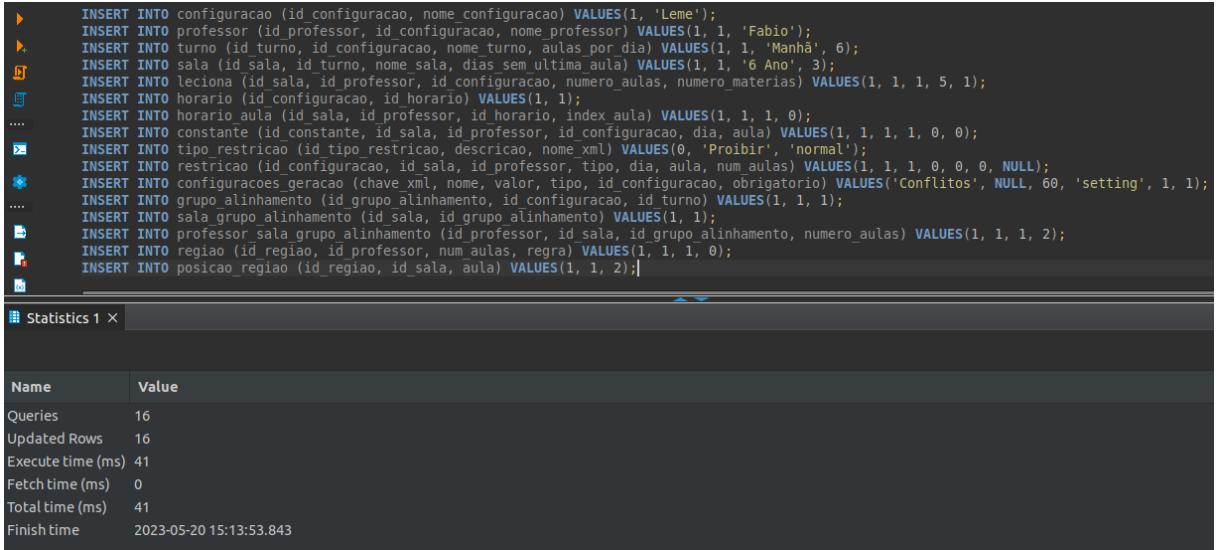


Fonte: Autor

Na [Figura 7](#), é possível notar a presença das entidades elacionadas às diversas métricas de qualidade comentadas na [Subseção 3.3.2](#), como restrições, contratantes, regiões, grupos de alinhamento, entre outras.

Para validar a modelagem do banco de dados, realizaram-se inserções de informações de exemplo nas diferentes tabelas. A [Figura 8](#) mostra algumas dessas inserções e os vínculos instituídos pelos identificadores escolhidos.

[Figura 8 – Consultas de validação da modelagem](#)



The screenshot shows a database interface with two main sections. The top section displays a series of SQL INSERT statements used to populate various tables with sample data. The bottom section, titled 'Statistics 1', provides performance metrics for the executed queries.

```

INSERT INTO configuracao (id_configuracao, nome_configuracao) VALUES(1, 'Leme');
INSERT INTO professor (id_professor, id_configuracao, nome_professor) VALUES(1, 1, 'Fabio');
INSERT INTO turno (id_turno, id_configuracao, nome_turno, aulas_por_dia) VALUES(1, 1, 'Manhã', 6);
INSERT INTO sala (id_sala, id_turno, nome_sala, dias_sem_ultima_aula) VALUES(1, 1, '6 Ano', 3);
INSERT INTO leciona (id_sala, id_professor, id_configuracao, numero_aulas, numero_materias) VALUES(1, 1, 1, 5, 1);
INSERT INTO horario (id_configuracao, id_horario) VALUES(1, 1);
INSERT INTO horario_aula (id_sala, id_professor, id_horario, index_aula) VALUES(1, 1, 1, 0);
INSERT INTO constante (id_constante, id_sala, id_professor, id_configuracao, dia, aula) VALUES(1, 1, 1, 1, 0, 0);
INSERT INTO tipo_restricao (id_tipo_restricao, descricao, nome_xml) VALUES(0, 'Proibir', 'normal');
INSERT INTO restricao (id_configuracao, id_sala, id_professor, tipo, dia, aula, num_aulas) VALUES(1, 1, 1, 0, 0, 0, NULL);
INSERT INTO configuracoes_geracao (chave_xnl, nome, valor, tipo, id_configuracao, obrigatorio) VALUES('Conflitos', NULL, 60, 'setting', 1, 1);
INSERT INTO grupo_alinhamento (id_grupo_alinhamento, id_configuracao, id_turno) VALUES(1, 1, 1);
INSERT INTO sala_grupo_alinhamento (id_sala, id_grupo_alinhamento) VALUES(1, 1);
INSERT INTO professor_sala_grupo_alinhamento (id_professor, id_sala, id_grupo_alinhamento, numero_aulas) VALUES(1, 1, 1, 2);
INSERT INTO regiao (id_regiao, id_professor, num_aulas, regra) VALUES(1, 1, 1, 0);
INSERT INTO posicao_regiao (id_regiao, id_sala, aula) VALUES(1, 1, 2);

```

Name	Value
Queries	16
Updated Rows	16
Execute time (ms)	41
Fetch time (ms)	0
Total time (ms)	41
Finish time	2023-05-20 15:13:53.843

Fonte: Autor

Validou-se também o armazenamento das grades horárias no banco de dados. A Figura 9 traz um exemplo de consulta de uma grade horária com sete salas:

Figura 9 – Consulta SQL de grade horária com sete salas

```

@SELECT
    CASE (a.n / 6)
        WHEN 0 THEN 'Segunda'
        WHEN 1 THEN 'Terça'
        WHEN 2 THEN 'Quarta'
        WHEN 3 THEN 'Quinta'
        WHEN 4 THEN 'Sexta'
        ELSE NULL
    END AS "DiaSemana",
    (a.n % 6) + 1 AS "Aula",
    p1.nome_professor AS "6Ano",
    p2.nome_professor AS "7Ano",
    p3.nome_professor AS "8Ano",
    p4.nome_professor AS "9Ano",
    p5.nome_professor AS "1EM",
    p6.nome_professor AS "2EM",
    p7.nome_professor AS "3EM"
FROM generate_series(0, 29) AS a(n)
LEFT JOIN horario_aula hal ON hal.id_sala = 2 AND hal.index_aula = a.n AND hal.id_horario = 10848
LEFT JOIN professor p1 ON p1.id_professor = hal.id_professor
LEFT JOIN horario_aula ha2 ON ha2.id_sala = 3 AND ha2.index_aula = a.n AND ha2.id_horario = 10848
LEFT JOIN professor p2 ON p2.id_professor = ha2.id_professor
LEFT JOIN horario_aula ha3 ON ha3.id_sala = 4 AND ha3.index_aula = a.n AND ha3.id_horario = 10848
LEFT JOIN professor p3 ON p3.id_professor = ha3.id_professor
LEFT JOIN horario_aula ha4 ON ha4.id_sala = 5 AND ha4.index_aula = a.n AND ha4.id_horario = 10848
LEFT JOIN professor p4 ON p4.id_professor = ha4.id_professor
LEFT JOIN horario_aula ha5 ON ha5.id_sala = 6 AND ha5.index_aula = a.n AND ha5.id_horario = 10848
LEFT JOIN professor p5 ON p5.id_professor = ha5.id_professor
LEFT JOIN horario_aula ha6 ON ha6.id_sala = 7 AND ha6.index_aula = a.n AND ha6.id_horario = 10848
LEFT JOIN professor p6 ON p6.id_professor = ha6.id_professor
LEFT JOIN horario_aula ha7 ON ha7.id_sala = 8 AND ha7.index_aula = a.n AND ha7.id_horario = 10848
LEFT JOIN professor p7 ON p7.id_professor = ha7.id_professor
ORDER BY a.n;
```

Fonte: Autor

A consulta anterior traz como resultado a tabela visível na Figura 10, com linhas e colunas correspondentes a horários de aulas e salas respectivamente.

Figura 10 – Resultado da consulta de uma grade no banco de dados

DiaSemana	1	2	3	4	5	6	7	8	9	1EM	2EM	3EM
Segunda	1	Renata	Rosa	Cristiane	Luciana	Adriana	Jéssica	Glaucia				
Segunda	2	Renata	Rosa	Cristiane	Luciana	Glaucia	Jéssica	Adriana				
Segunda	3	Beatriz	Renata	Rosa	Jéssica	Glaucia	Luciana	Adriana				
Segunda	4	Cristiane	Renata	Rosa	Adriana	Jéssica	Glaucia	Luciana				
Segunda	5	Cristiane	Adriana	Beatriz	Adriana	Jéssica	Glaucia	Luciana				
Segunda	6	[NULL]	Beatriz	[NULL]	[NULL]	Luciana	Jéssica	Adriana				
Terça	1	Rosa	Luciano	Luciana	Renata	Fabio	Adriana	Jéssica				
Terça	2	Rosa	Adriana	Luciana	Fabio	Luciano	Glaucia	Jéssica				
Terça	3	Luciano	Adriana	Renata	Fabio	Jéssica	Glaucia	Luciana				
Terça	4	Luciano	Renata	Adriana	Jéssica	Fabio	Luciana	Glaucia				
Terça	5	Adriana	Jéssica	Luciana	Beatriz	Fabio	Luciano	Glaucia				
Terça	6	Adriana	[NULL]	[NULL]	[NULL]	Glaucia	Luciano	Jéssica				
Quarta	1	Luciano	Jack	Rosa	Luciana	Adriana	Renata	Jéssica				
Quarta	2	Rosa	Jack	Jéssica	Luciano	Adriana	Renata	Luciana				
Quarta	3	Jack	Luciano	Jéssica	Fabio	Luciana	Adriana	Renata				
Quarta	4	Jéssica	Luciano	Luciana	Fabio	Jack	Adriana	Renata				
Quarta	5	Jéssica	Jack	Luciano	Luciana	Glaucia	Fabio	Adriana				
Quarta	6	[NULL]	[NULL]	Luciano	Luciana	Glaucia	Jack	Adriana				
Quinta	1	Luciana	Jack	Rosa	Renata	Jéssica	Fabio	Adriana				
Quinta	2	Jéssica	Jack	Rosa	Renata	Adriana	Luciana	Fabio				
Quinta	3	Jack	Rosa	Renata	Jéssica	Fabio	Adriana	Luciana				
Quinta	4	Jack	Rosa	Renata	Jéssica	Luciana	Adriana	Fabio				
Quinta	5	Adriana	Jéssica	Luciana	Jack	Luciano	Glaucia	Fabio				
Quinta	6	[NULL]	[NULL]	Jack	Luciana	Luciano	Adriana	Glaucia				
Sexta	1	Jack	Rosa	Luciana	Jéssica	Adriana	Renata	Luciano				
Sexta	2	Renata	Jack	Jéssica	Fabio	Adriana	Luciana	Luciano				
Sexta	3	Rosa	Jéssica	Luciano	Adriana	Renata	Luciana	Fabio				
Sexta	4	Rosa	Luciana	Adriana	Luciano	Renata	Jéssica	Jack				
Sexta	5	Jack	Jéssica	Adriana	Luciano	Luciana	Fabio	Glaucia				
Sexta	6	Jack	Jéssica	[NULL]	[NULL]	Luciana	Fabio	Luciano				

Fonte: Autor

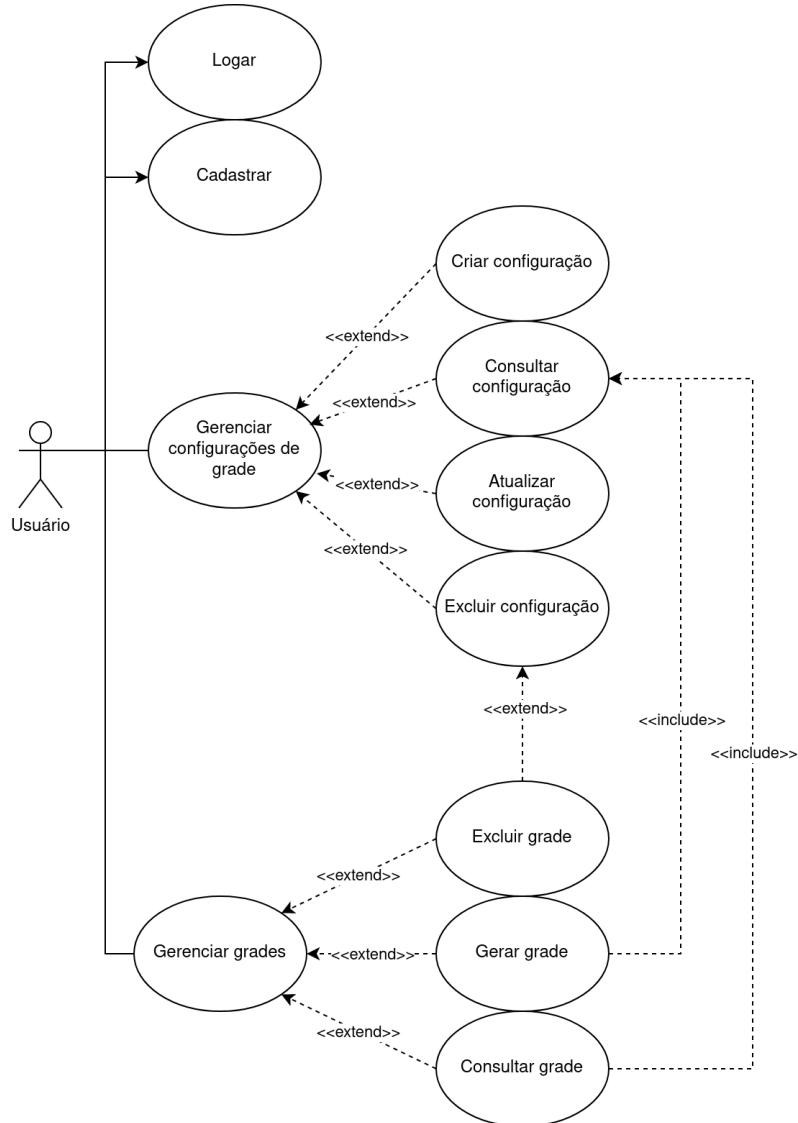
3.3.4 INTERFACE

A interface web é responsável pela interação do usuário final com a aplicação. Portanto, implementaram-se funcionalidades que possibilitam a configuração das restrições e características das grades horárias a serem geradas, além de mostrar ou exportar tais grades após a geração.

Para o desenvolvimento deste componente, optou-se pelo framework *Vue.js*, devido à riqueza de seu ecossistema de desenvolvimento *frontend* e familiaridade do graduando com este. Complementando este framework, utilizou-se também o *Vuetify.js*, a fim de facilitar e padronizar o design da aplicação.

A [Figura 11](#) mostra os casos de uso que a interface web é responsável por disponibilizar:

Figura 11 – Diagrama de Casos de Uso



Fonte: Autor

A [Figura 12](#) mostra a tela de listagem de configurações de grade. Como comentado anteriormente, alguns dos casos de uso da aplicação envolvem o gerenciamento de configurações de grades horárias, as quais são listadas nessa tela.

Figura 12 – Tela - Listagem de Configurações de Grade

Horário	CONFIGURAÇÕES	HORÁRIOS
Configurações de horário		
Nome	NOVA CONFIGURAÇÃO	Ações
Teste		⊕ ✓ ━
Leme		⊕ ✓ ━
LemeMultiTurno		⊕ ✓ ━

Fonte: Autor

A tela mostrada na [Figura 13](#) é responsável por permitir que o usuário cadastre os professores da escola. Nessa tela é possível notar que a aplicação foi estruturada seguindo uma noção de etapas de configuração até a geração da grade horária final. No topo da tela, a etapa “Estrutura da Escola” encontra-se selecionada.

Figura 13 – Tela - Estrutura da Escola - Professores

Professores	
Nome	Ações
Fabio	✓
Adriana	✓
Luciana	✓
Luciano	✓
Jack	✓
Gláucia	✓
Renata	✓
Beatriz	✓
Rosa	✓
Jéssica	✓
Cristiane	✓

Turnos		
Nome	Aulas por dia	Ações
...	-	-

Fonte: Autor

Na [Figura 14](#), tem-se a continuação da tela de estrutura da escola, mostrada na [Figura 13](#). Esta parte da tela é responsável pela configuração das turmas e turnos da escola.

Figura 14 – Tela - Estrutura da Escola - Turmas e Turnos

Fonte: Autor

Na tela da [Figura 15](#), o usuário pode realizar a configuração de número de aulas que cada professor deve ministrar em cada sala, informação fundamental para a geração das grades horárias.

Figura 15 – Tela - Aulas por professor

Professor	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM	Total
Fabio	0	0	0	5	4	4	4	17
Adriana	4	3	3	3	6	6	6	31
Luciana	1	1	6	6	5	5	5	29
Luciano	3	3	3	3	3	2	3	20
Jack	6	6	1	1	1	1	1	17
Glauclia	0	0	0	0	5	5	5	15
Renata	3	3	3	3	2	3	2	19
Beatriz	1	1	1	1	0	0	0	4
Rosa	5	5	5	0	0	0	0	15
Jéssica	3	5	3	5	4	4	4	28
Cristiane	2	0	2	0	0	0	0	4

Fonte: Autor

A tela da [Figura 16](#) é responsável pela configuração das restrições e preferências. Nesta, o usuário pode configurar horários na grade que devem ser evitados ou proibidos para determinado docente.

[Figura 16 – Tela - Restrições e Preferências](#)

Fonte: Autor

Já a tela da [Figura 17](#) possibilita a configuração das regiões e grupos de alinhamento:

[Figura 17 – Tela - Regiões e Grupos de Alinhamento](#)

Fonte: Autor

A última tela relacionada a configurações pode ser vista na [Figura 18](#), e permite a definição de quantidade mínimas e máximas de aulas para cada professor em cada dia da grade horária:

A etapa final no fluxo da aplicação é representada pela tela da [Figura 19](#). Nesta, o usuário pode requisitar a geração da grade horária utilizando as configurações realizadas nas etapas anteriores, e acessar as grades geradas anteriormente.

Figura 18 – Tela - Mínimos e Máximos

Mínimos		Limites diários									
Professor	Segunda	Terça	Quarta	Quinta	Sexta	Professor	Segunda	Terça	Quarta	Quinta	Sexta
Verônica	0	0	0	0	0	Verônica	0	0	0	0	0
Adriana	0	0	0	0	0	Adriana	0	0	0	0	0
Luciana	0	0	0	0	0	Luciana	0	0	0	0	0
Luciano	0	0	0	0	0	Luciano	0	0	0	0	0
Jack	0	0	0	0	0	Jack	0	0	0	0	0
Gláucia	0	0	0	0	0	Gláucia	0	0	0	0	0
Renata	0	0	0	0	0	Renata	0	0	0	0	0
Beatriz	0	0	0	0	0	Beatriz	0	0	0	0	0
Rosa	0	0	0	0	0	Rosa	0	0	0	0	0
Jéssica	0	0	0	0	0	Jéssica	0	0	0	0	0
Cristiane	0	0	0	0	0	Cristiane	0	0	0	0	0
Fábio	0	0	0	0	0	Fábio	0	0	0	0	0

Fonte: Autor

Figura 19 – Tela - Horários

Horário (10832)												Horários Gerados					
Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM	Aulas separadas	Restrições não resolvidas	Restrições opcionais não resolvidas	Prefeências não resolvidas	Asunções de aulas duplas	Exposas juvenis	Qualidade ↑ (3)	Após		
1	Rosa	Renata	Adriana	Jéssica	Fábio	Gláucia	Luciana	80	0	0	1	1	60	222	II		
2	Rosa	Renata	Adriana	Jéssica	Gláucia	Fábio	Luciana	80	0	0	1	1	60	222	II		
3	Adriana	Rosa	Renata	Luciana	Gláucia	Jéssica	Fábio	80	0	0	1	1	60	222	II		
4	Cristiane	Rosa	Renata	Fábio	Luciana	Jéssica	Adriana	80	0	0	1	1	60	222	II		
5	Cristiane	Jéssica	Luciana	Beatriz	Fábio	Adriana	Gláucia	80	0	0	1	1	60	222	II		
6	Beatriz				Fábio	Adriana	Gláucia	80	0	0	2	1	70	236	II		
7	Renata	Jéssica	Rosa	Luciana	Adriana	Luciano	Gláucia	80	0	0	2	1	70	236	II		
8	Jéssica	Rosa	Luciano	Adriana	Gláucia	Luciana	Renata	82	0	0	2	1	70	238	II		
9	Luciano	Jéssica	Rosa	Cristiane	Luciano	Luciana	Adriana	82	0	0	2	1	70	238	II		
10	Beatriz	Luciana	Cristiane	Luciano	Adriana	Gláucia	Jéssica	82	0	0	2	1	70	238	II		
11	Renata	Jéssica	Luciana	Beatriz	Luciano	Jéssica	Adriana	80	0	0	2	1	50	242	II		
12	Rosa	Jack	Renata	Fábio	Luciana	Jéssica	Adriana	80	0	0	1	2	50	242	II		
13	Luciano	Jack	Rosa	Adriana	Jéssica	Renata	Luciano	80	0	0	1	2	50	242	II		
14	Jéssica	Luciano	Rosa	Adriana	Renata	Luciana	Jack	80	0	0	1	2	50	242	II		
15	Luciano	Jack	Luciana	Jéssica	Renata	Adriana	Fábio	80	0	0	1	2	50	242	II		
16	Jack	Adriana	Luciana	Jéssica	Luciano	Gláucia	Fábio	80	0	0	1	2	50	242	II		
17	Jack					Adriana	Gláucia	Luciana									
18	Renata	Jéssica	Luciana	Jack	Luciano	Fábio	Adriana										
19	Rosa	Jéssica	Luciana	Renata	Luciano	Jack	Adriana										
20	Rosa	Jack	Luciana	Renata	Adriana	Jéssica	Adriana										
21	Jack	Luciano	Luciana	Jéssica	Gláucia	Adriana	Fábio										

Fonte: Autor

3.3.5 USUÁRIOS E AUTENTICAÇÃO

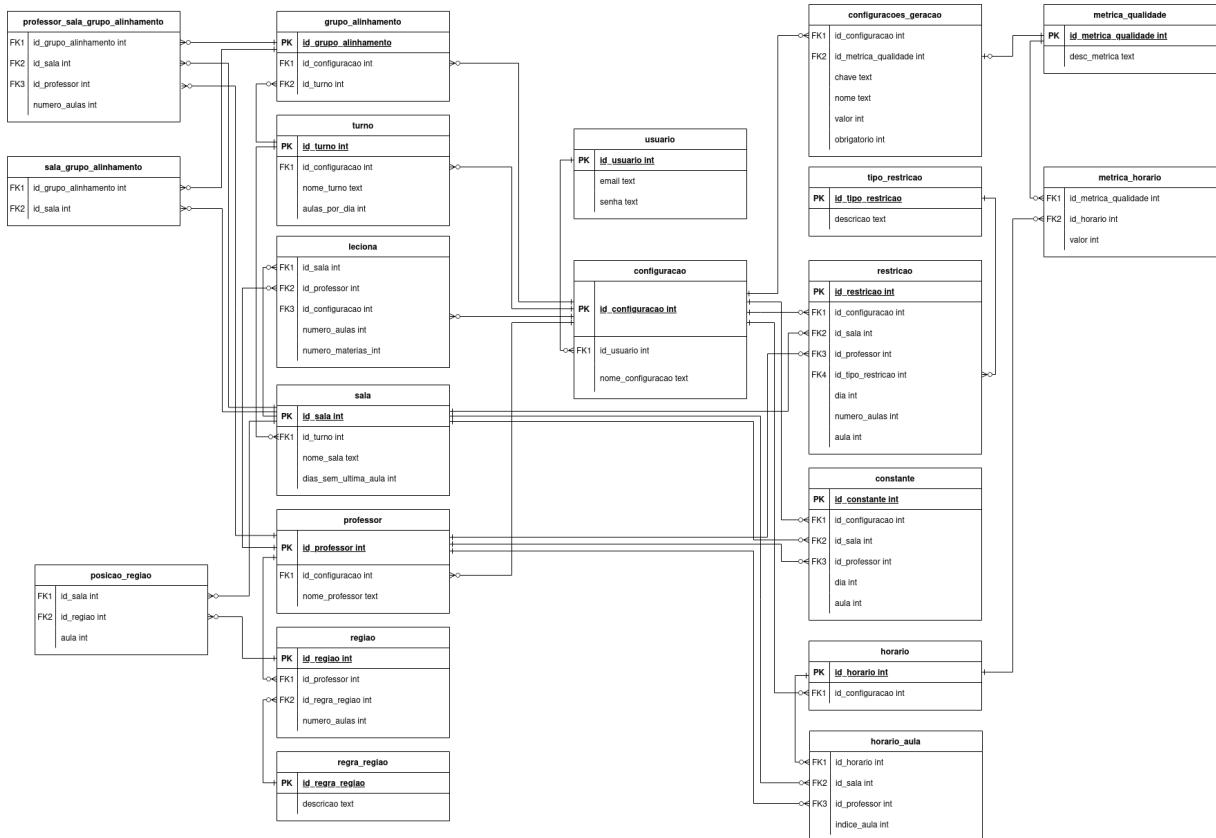
Para controlar a visibilidade de informações no sistema desenvolvido, implementou-se um sistema de usuários. A melhoria consistiu na:

1. Criação de tabela de usuários no banco de dados;
2. Associação da tabela “configuracao” com a tabela de usuários, para que fosse possível armazenar o usuário responsável por cada configuração;
3. Criação da tela de login na interface web;
4. Criação das rotas de cadastro e *login* no servidor
5. Criação do *middleware* de autenticação
6. Criação do *middleware* de validação da configuração

3.3.5.1 Adaptação do banco de dados

A aplicação dos itens 1 e 2 foi realizada diretamente no banco de dados, através da criação da tabela mencionada e a chave estrangeira possibilitando a associação de cada usuário a múltiplas configurações. Após estas alterações, o diagrama entidade relacionamento do banco de dados passa a ser representado pela figura 20.

Figura 20 – Modelo Entidade-Relacionamento com adição da tabela de usuários

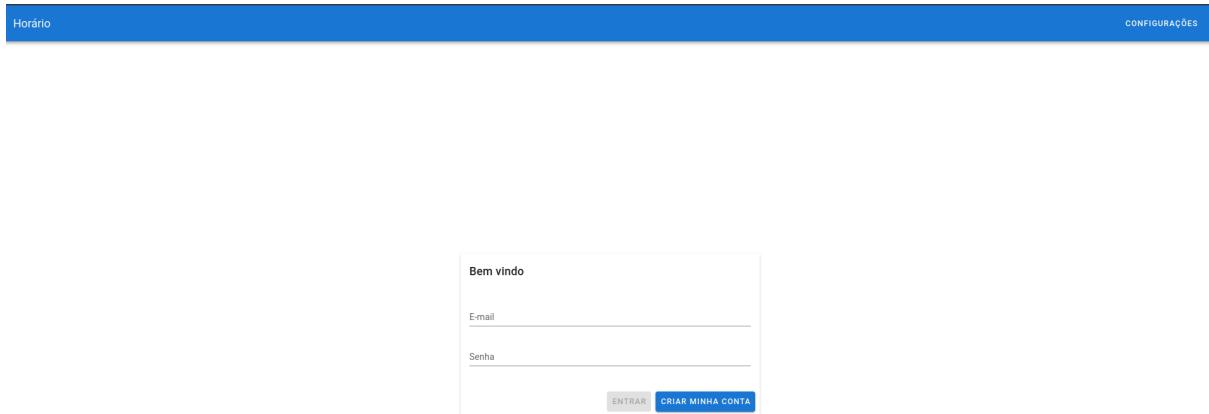


Fonte: Autor

3.3.5.2 Tela de login

Como pode ser visto na figura 21, desenvolveu-se uma tela simples de *login*, a qual também desempenha a função de cadastro de novos usuários.

Figura 21 – Tela de Acesso



Fonte: Autor

3.3.5.3 Rotas de autenticação

Para a implementação das rotas de cadastro e *login*, utilizou-se além do *framework* ExpressJS, os pacotes JWT (*Json Web Token*) e *bcrypt*. O pacote JWT é utilizado para gerar *tokens*, que são enviados para a interface web e podem ser utilizados para autenticar os usuários; já o *bcrypt* é utilizado para gerar e validars as *hashes* das senhas dos usuários, para que nunca sejam armazenadas senhas em texto pleno no banco de dados.

A função utilizada pela nova rota de cadastro pode vista na figura 22. A função “register” realiza a validação dos parâmetros, e cria um usuário no banco de dados, caso já não exista outro com o mesmo e-mail. Além disso, a função retorna um *token* JWT para a interface web, para que seja possível verificar a autenticidade das próximas requisições realizadas pelo usuário.

Figura 22 – Método de Cadastro

```

const register = async (req, res) => {
  const { email, senha } = req.body;
  if (email == null || senha == null) {
    return res.status(400).json({ message: "E-mail e senha são obrigatórios" });
  }

  const emailRegex = /^[^@\s]+@[^\s]+\.[^\s]+$/;
  if (!emailRegex.test(email)) {
    return res.status(400).json({ message: "E-mail inválido" });
  }

  if (senha.length < 6) {
    return res
      .status(400)
      .json({ message: "A senha deve conter pelo menos 6 caracteres" });
  }

  const resUsuario = await pool.query(
    "select 1 from usuario where email = $1",
    [email]
  );
  if (resUsuario.rows.length > 0) {
    return res.status(400).json({
      message: "Este e-mail já está sendo utilizado por outro usuário",
    });
  }

  const saltRounds = 10;
  const hash = await bcrypt.hash(senha, saltRounds);

  const resUsuarioInserido = await pool.query(
    "insert into usuario (email, senha) values ($1, $2) returning id_usuario",
    [email, hash]
  );
  const id_usuario = resUsuarioInserido.rows[0].id_usuario;

  const secret = process.env.JWT_SECRET;
  const token = jwt.sign({ id_usuario }, secret);
  return res
    .status(200)
    .json({ token, message: "Usuário registrado com sucesso" });
};

```

Fonte: Autor

A função de *login*, visível na figura 23 é similar, realizando validação dos parâmetros, autenticação do usuário através da comparação de senhas utilizando o pacote *bcrypt* e geração de *token* JWT. Vale citar que em ambas as rotas de autenticação, é inserido no *payload* do *token* o identificador do usuário, que posteriormente pode ser utilizado pelos *middlewares* para a realização de validações.

Figura 23 – Método de Login

```
const login = async (req, res) => {
  const { email, senha } = req.body;
  if (email == null || senha == null) {
    return res.status(400).json({ message: "E-mail e senha são obrigatórios" });
  }

  const resUsuario = await pool.query(
    "select id_usuario, senha from usuario where email = $1",
    [email]
  );
  if (resUsuario.rows.length === 0) {
    return res.status(400).json({ message: "Credenciais inválidas" });
  }

  const { senha: hash, id_usuario } = resUsuario.rows[0];
  const correctPassword = await bcrypt.compare(senha, hash);

  if (!correctPassword) {
    return res.status(400).json({ message: "Credenciais inválidas" });
  }

  const secret = process.env.JWT_SECRET;
  const token = jwt.sign({ id_usuario }, secret);

  return res
    .status(200)
    .json({ token, message: "Login realizado com sucesso" });
};
```

Fonte: Autor

3.3.5.4 Middlewares

Foram criadas duas funções *middleware* relacionadas aos usuários. A primeira é responsável por assegurar que apenas usuários propriamente autenticados tenham acesso aos recursos protegidos do sistema. Como pode ser visto na figura 24, essa verificação é realizada através da verificação da presença de um *token* JWT válido no header “*authorization*” da requisição.

Figura 24 – Middleware de autenticação

```

const jwt = require("jsonwebtoken");

function authenticateToken(req, res, next) {
    const authHeader = req.headers["authorization"];
    const token = authHeader && authHeader.split(" ")[1];

    if (token == null) return res.sendStatus(401);

    jwt.verify(token, process.env.JWT_SECRET, (err, data) => {
        if (err || data.id_usuario == null) return res.sendStatus(403);
        req.id_usuario = data.id_usuario;
        next();
    });
}

module.exports = authenticateToken;

```

Fonte: Autor

O outro middleware criado tem como objetivo assegurar que um usuário possa consultar apenas as configurações pelas quais seja responsável. Isso é feito utilizando o identificador do usuário, presente no *payload* do *token* JWT, conforme a figura 25. Caso o usuário não tenha um vínculo com a configuração que está tentando acessar, a requisição é bloqueada.

Figura 25 – Middleware de validação da configuração

```

const pool = require("../config/db");

async function validateConfigId(req, res, next) {
    const ids = [req.params.id, req.body.id_configuracao];
    if (ids.every((id) => id == null)) {
        return next();
    }
    const { id_usuario } = req;
    for (let id_configuracao of ids) {
        if (id_configuracao == null) {
            continue;
        }

        const result = await pool.query(
            "select 1 from configuracao where id_configuracao = $1 and id_usuario = $2",
            [id_configuracao, id_usuario]
        );
        if (result.rows.length == 0) {
            return res.sendStatus(401);
        }
    }
    return next();
}

module.exports = validateConfigId;

```

Fonte: Autor

3.3.6 ALOCAÇÃO DE MATÉRIAS

Durante os incrementos anteriores, o otimizador gerava grades horárias que continham apenas os nomes dos professores alocados para aula. Em outras palavras, as grades eram matrizes nas quais as linhas eram os horários disponíveis, as colunas eram as salas e cada posição na matriz representava qual professor deveria ministrar a aula naquele horário.

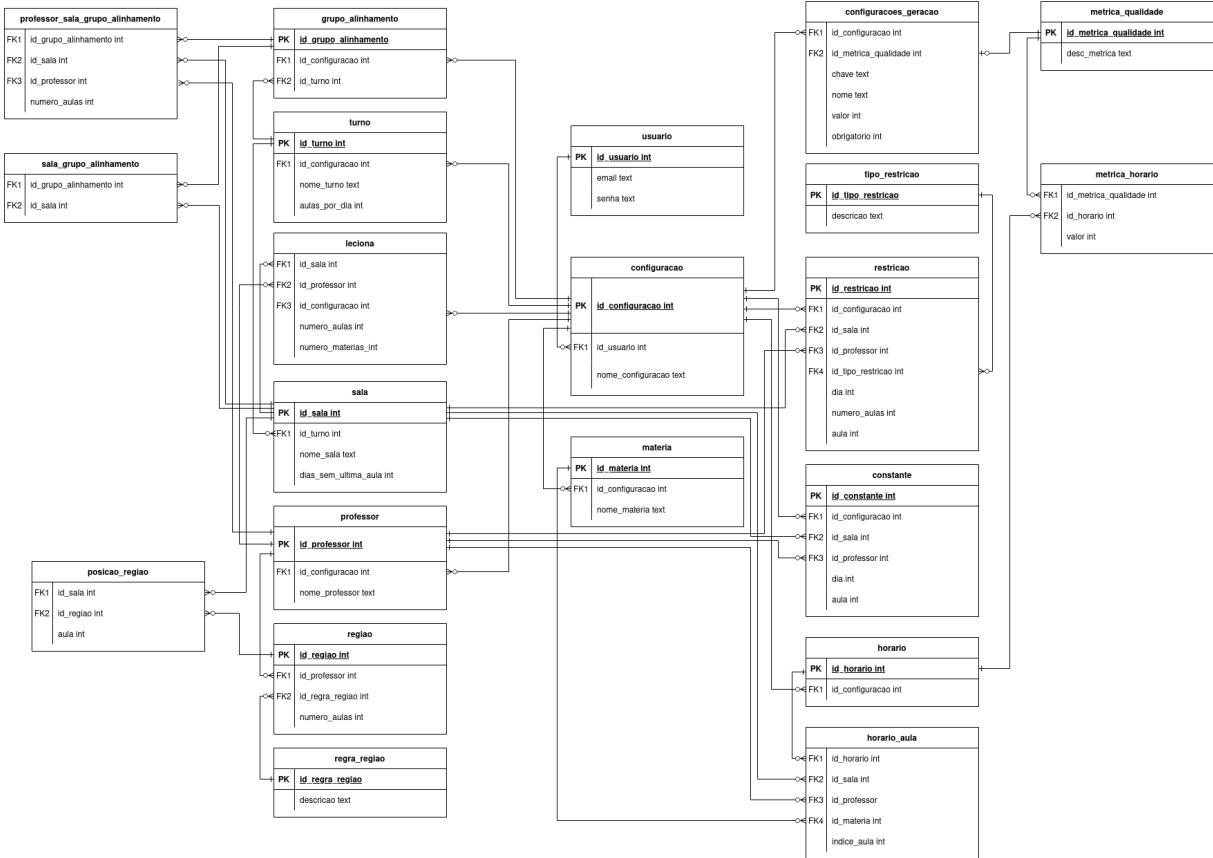
A alocação dos professores é muito importante para a resolução do problema, pois apresenta a maior parte das restrições e dificuldades relacionadas, como os conflitos, por exemplo. Entretanto, em termos de completude, as grades horárias devem ter também a alocação de matérias em cada horário de aula, visto que cada professor pode ministrar aulas de mais de uma matéria. As alterações necessárias para possibilitar isso foram:

1. Alterar a modelagem do banco de dados para comportar informações relacionadas às matérias;
2. Alterar telas da interface web para que fosse possível configurar as matérias;
3. Alterar rotas do servidor para persistir as matérias;
4. Alterar código do otimizador para produzir grades horárias com matérias

3.3.6.1 Alteração no banco de dados

Para armazenar informações relacionadas às matérias, a modelagem do banco de dados foi alterada com a criação de uma tabela para armazenar as matérias, e a adição de uma coluna à tabela “horario” aula para que seja possível especificar matérias alocadas nas grades horárias. Com estas alterações, a modelagem do banco de dados passa a ser representada pela figura 26.

Figura 26 – Modelo Entidade-Relacionamento com Matérias



Fonte: Autor

3.3.6.2 Alteração de telas

Com a adição do conceito das matérias ao sistema, algumas telas da interface precisaram ser modificadas. A primeira destas, foi a tela inicial da configuração, ou seja, a tela "Estrutura da Escola", cuja alteração pode ser vista na figura 27 consistiu na adição de uma seção para cadastro e visualização de matérias, semelhante ao componente de cadastro de professores.

Figura 27 – Estrutura da Escola com Matérias

Fonte: Autor

Além da tela de estrutura, a segunda aba da configuração ("Aulas/Matérias") foi alterada para que fosse possível vincular matérias aos professores, e configurar a quantidade de aulas de cada matéria deve ser ministrada semanalmente, conforme a figura 28.

Figura 28 – Tela de configuração de quantidades de aulas por matéria

Fonte: Autor

Por fim, na tela final da configuração, responsável por exibir as grades horárias geradas, foi necessário alterar o componente da grade para incluir, além dos nomes dos professores, os nomes das matérias alocadas para cada horário, como pode ser visto na figura 29.

Figura 29 – Visualização de grade horária com matérias

Fonte: Autor

3.3.6.3 Alteração de métodos no servidor

A adição das matérias também envolveu algumas alterações no servidor. As rotas alteradas para acomodar a melhoria foram rotas de armazenamento e consulta da estrutura da escola, aulas e grades horárias.

Evidentemente, foi necessário criar rotas também para o gerenciamento de matérias, e as demais rotas do sistema precisaram ter suas funções de tratamento modificadas para interagir corretamente com esta nova tabela.

3.3.6.4 Matérias no otimizador

Após as alterações na interface, servidor e banco de dados, o sistema estava pronto para lidar com as informações relacionadas às matérias, faltando apenas a incorporação destas na geração de grades horárias por parte do otimizador.

Para realizar isso, o otimizador foi alterado para alocar também matérias na grade horária, de forma que enquanto antes a grade horária armazenava em cada posição um professor que foi alocado para aquele horário, agora passa a armazenar uma dupla de professor e matéria alocados. Durante os passos de otimização, estas duplas são permutadas, garantindo assim que cada matéria sempre corresponde a um professor compatível. Com estas alterações, o algoritmo 3 passa a representar o otimizador.

Algoritmo 3: Otimizador com alocação de matérias

Input: Lista de professores LP , lista de turmas LT , matriz de aulas e matérias por professor por turma MA , temperatura inicial TI , Taxa de resfriamento TR

Output: Grade horária de matérias e professores otimizada

```

temperatura ← TI
grade ← CriaGradeInicial( $LP, LT, MA$ )
iteracoesSemAlteracao ← 0
solucoes ← lista vazia
while condição de parada não atingida do
    deltaTotal ← 0
    for passo = 0 até numeroPassos do
        turma ← grade.EscolheTurmaAleatoria()
        duplas ← grade.EscolheDuplasAleatoriasValidas(turma)
        delta ← grade.CalculaDelta(turma, duplas)
        probabilidade ←  $e^{-\delta/\text{temperatura}}$ 
        valorAceite ← ValorAleatorioEntre(0, 1)
        if delta < 0 ou probabilidade ≥ valorAceite then
            grade.PermutaDuplas(sala, duplas)
            deltaTotal ← deltaTotal + delta
            if grade não existe na lista de soluções then
                insere grade na lista de soluções
                limita lista de soluções às 100 melhores grades
            end
        end
    end
    if delta = 0 then
        iteracoesSemAlteracao ← iteracoesSemAlteracao + 1
    else
        iteracoesSemAlteracao ← 0
    end
    if iteracoesSemAlteracao ≥ 15 then
        salvaGradesRelevantes()
        apaga lista de soluções
        temperatura ← TI
        iteracoesSemAlteracao ← 0
    end
    temperatura ← temperatura * TR
end
```

3.3.7 VALIDAÇÃO DE CONFIGURAÇÕES

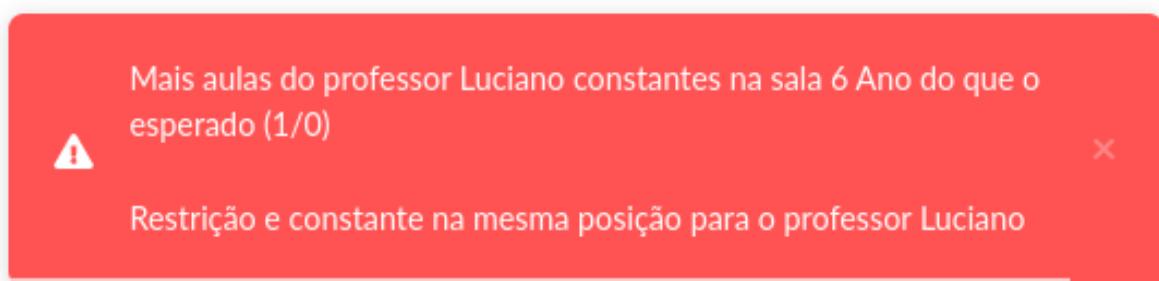
Devido à grande quantidade de configurações necessárias para parametrizar a geração de uma grade horária, erros de configurações por parte dos usuários são inevitáveis. Tendo isto em mente, implementou-se uma seção de código no otimizador, responsável pela validação das informações providenciadas pelo usuário.

Vale ressaltar que esta validação não garante que é possível gerar uma grade horária de acordo com as configurações providas, mas ela deve evitar alguns dos erros mais comuns. As validações realizadas foram:

1. Existem turnos, turmas, professores e matérias configurados;
2. O total de aulas configurado para cada sala deve ser correto, de acordo com o número de dias da grade horária e quantidade de aulas por dia;
3. Para cada sala, nenhum professor pode ter mais aulas agendadas do que a sala acomoda;
4. Nenhum professor pode ter mais aulas constantes configuradas do que o seu total de aulas naquela sala;
5. Não pode haver uma restrição e aula constante para determinado professor na mesma posição da grade horária;
6. Cada grupo de alinhamento deve relacionar duas turmas diferentes, e referenciar o alinhamento da mesma quantidade de aulas em ambas as salas, sendo esta quantidade compatível com o total de aulas configurado para cada professor;
7. O número de aulas esperado para cada região deve ser compatível com o total de aulas configurado para as salas englobadas na região

Estas validações são executadas antes do início das otimizações das grades horárias, e caso haja algum erro de configuração, é exibida uma mensagem alertando o usuário, conforme a [Figura 30](#), e a geração da grade horária é cancelada.

Figura 30 – Mensagem de erro de validação



Fonte: Autor

3.3.8 EXPORTAÇÃO DE GRADES HORÁRIAS

Considerando a necessidade de exportar as grades horárias para fora do sistema, e a sua natureza tabular, implementou-se a funcionalidade de *download* das grades como planilhas Excel, no formato XLSX.

Para possibilitar a exportação neste formato, utilizou-se o pacote *Node ExcelJs*, que providencia a classe “*WorkBook*”, a qual facilita a construção de arquivos de planilhas Excel utilizando a linguagem *JavaScript*. O resultado final desta implementação pode ser visto na [Figura 31](#).

Figura 31 – Grade horária exportada para planilha

Dia	Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM					
Segunda-feira	1	Matemática	Rosa	Português	Jack	Geografia	Verônica	Inglês	Jéssica	Matemática	Glaucia
	2	Matemática	Rosa	Português	Jack	Geografia	Verônica	Inglês	Jéssica	Matemática	Glaucia
	3	Espanhol	Jack	Matemática	Rosa	Português	Luciana	História	Renata	Filosofia	Jéssica
	4	Português	Jack	Matemática	Rosa	Português	Luciana	História	Renata	Artes	Jéssica
	5	Ciências	Adriana	ED. Física	Beatriz	Espanhol	Jack	Português	Luciana	História	Renata
	6	Ciências	Adriana					Português	Luciana	História	Renata
Terça-feira	1	Of. Texto	Luciana	Matemática	Rosa	Geografia	Verônica	Filosofia	Jéssica	P. Vida	Fabio
	2	Geografia	Verônica	Ciências	Adriana	Matemática	Rosa	Português	Luciana	Física	Fabio
	3	Geografia	Verônica	Ciências	Adriana	Matemática	Rosa	Português	Luciana	Espanhol	Jack
	4	Matemática	Rosa	Português	Jack	Of. Texto	Luciana	Geografia	Verônica	Biologia	Adriana
	5	Matemática	Rosa	Português	Jack	Artes	Cristiane	Geografia	Verônica	Biologia	Adriana
	6	ED. Física	Beatriz			Artes	Cristiane			Sociologia	Verônica
Quarta-feira	1	Filosofia	Jéssica	Geografia	Verônica	História	Renata	Ciências	Adriana	Matemática	Glaucia
	2	Artes	Cristiane	Geografia	Verônica	Português	Luciana	Artes	Jéssica	Química	Adriana
	3	Artes	Cristiane	Espanhol	Jack	Português	Luciana	Artes	Jéssica	Química	Adriana
	4	Português	Jack	Artes	Jéssica	Ciências	Adriana	Português	Luciana	Física	Fabio
	5	Português	Jack	Artes	Jéssica	Ciências	Adriana	Of. Texto	Luciana	Física	Fabio
	6					Matemática	Rosa	Matemática	Fabio	Biologia	Adriana
Quinta-feira	1	Geografia	Verônica	Ciências	Adriana	Inglês	Jéssica	Espanhol	Jack	Matemática	Glaucia
	2	Português	Jack	Matemática	Rosa	Inglês	Jéssica	Geografia	Verônica	Matemática	Glaucia
	3	Português	Jack	Inglês	Jéssica	Matemática	Rosa	História	Renata	Geografia	Verônica
	4	História	Renata	Inglês	Jéssica	Matemática	Rosa	Matemática	Fabio	Geografia	Verônica
	5	História	Renata	Geografia	Verônica	ED. Física	Beatriz	Matemática	Fabio	Português	Luciana
	6			História	Renata					Português	Luciana
Sexta-feira	1	História	Renata	Filosofia	Jéssica	Ciências	Adriana	Matemática	Fabio	Of. Texto	Luciana
	2	Inglês	Jéssica	Of. Texto	Luciana	História	Renata	Matemática	Fabio	Química	Adriana
	3	Inglês	Jéssica	Matemática	Rosa	História	Renata	Ciências	Adriana	Português	Luciana
	4	Matemática	Rosa	História	Renata	Filosofia	Jéssica	Ciências	Adriana	Literatura	Luciana
	5	Ciências	Adriana	História	Renata	Português	Luciana	ED. Física	Beatriz	Inglês	Jéssica
	6			Português	Jack					Inglês	Jéssica

Fonte: Autor

Para possibilitar o *download* das grades horárias, acrescentou-se um botão à interface. Adicionalmente, foram adicionados botões para navegar entre as diferentes grades geradas, e exibir mais informações sobre a grade selecionada. Os botões podem ser vistos, realçados em vermelho na figura [32](#):

Figura 32 – Grade horária exportada para planilha

Horário										CONFIGURAÇÕES	SAIR
	Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM					
Segunda-Feira	1	Verônica	Geografia	Adriana	Ciências	Renata	História	Fábio	Matemática	Luciana	Português
	2	Verônica	Geografia	Adriana	Ciências	Renata	História	Fábio	Matemática	Luciana	Português
	3	Luciana	Of. Texto	Jéssica	Filosofia	Christiane	Artes	Renata	História	Adriana	Biologia
	4	Rosa	Matemática	Jéssica	Inglês	Christiane	Artes	Renata	História	Adriana	Química
	5	Rosa	Matemática	Renata	História	Jéssica	Filosofia	Jack	Espanhol	Glaucia	Matemática
	6	Jéssica	Filosofia	Rosa	Matemática					Glaucia	Matemática
Terça-Feira	1	Christiane	Artes	Renata	História	Adriana	Ciências	Fábio	Matemática	Jéssica	Inglês
	2	Christiane	Artes	Renata	História	Adriana	Ciências	Fábio	Matemática	Jéssica	Inglês
	3	Jack	Espanhol	Jéssica	Artes	Renata	História	Adriana	Ciências	Glaucia	Matemática
	4	Jack	Português	Jéssica	Inglês	Rosa	Matemática	Adriana	Ciências	Renata	História
	5	Verônica	Geografia	Luciana	Of. Texto	Rosa	Matemática	Jéssica	Artes	Renata	História
	6							Jéssica	Artes	Verônica	Sociologia
Quarta-Feira	1	Renata	História	Jéssica	Artes	Rosa	Matemática	Verônica	Geografia	Glaucia	Matemática
	2	Renata	História	Rosa	Matemática	Jéssica	Inglês	Fábio	Matemática	Glaucia	Matemática
	3	Beatriz	ED. Física	Rosa	Matemática	Jéssica	Inglês	Luciana	Português	Fábio	P. Vida
	4	Rosa	Matemática	Jack	Português	Beatriz	ED. Física	Luciana	Português	Fábio	Física
	5	Rosa	Matemática	Jack	Português	Luciana	Português	Beatriz	ED. Física	Adriana	Química
	6					Beatriz	ED. Física	Luciana	Of. Texto	Adriana	Química
Quinta-Feira	1	Jack	Português	Rosa	Matemática	Verônica	Geografia	Luciana	Português	Adriana	Biologia
	2	Jack	Português	Rosa	Matemática	Verônica	Geografia	Luciana	Of. Texto	Adriana	Biologia
	3	Adriana	Ciências	Jack	Espanhol	Rosa	Matemática	Verônica	Geografia	Luciana	Português
	4	Adriana	Ciências	Jack	Português	Rosa	Matemática	Verônica	Geografia	Luciana	Of. Texto
	5	Renata	História	Verônica	Geografia	Luciana	Português	Jéssica	Inglês	Fábio	Física
	6					Luciana	Português	Jéssica	Filosofia	Fábio	Física
Sexta-Feira	1	Adriana	Ciências	Jack	Português	Verônica	Geografia	Jéssica	Inglês	Luciana	Literatura
	2	Jéssica	Inglês	Jack	Português	Adriana	Ciências	Luciana	Português	Verônica	Geografia
	3	Jéssica	Inglês	Adriana	Ciências	Jack	Espanhol	Luciana	Português	Verônica	Geografia
	4	Jack	Português	Verônica	Geografia	Luciana	Português	Adriana	Ciências	Jéssica	Artes
	5					Luciana	Português	Jéssica	Filosofia	Fábio	Física
	6										

Fonte: Autor

Ao clicar no botão de informações, são mostradas todas as métricas de qualidade da grade horária, conforme a figura 33:

Figura 33 – Grade horária exportada para planilha

Horário (1366)										CONFIGURAÇÕES	SAIR
	Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM					
Segunda-Feira	1	Verônica	Geografia	Adriana	Informações do horário				Fábio	Matemática	Luciana
	2	Verônica	Geografia	Adriana					Fábio	Matemática	Português
	3	Luciana	Of. Texto	Jéssica					Renata	História	Biologia
	4	Rosa	Matemática	Jéssica					Renata	História	Química
	5	Rosa	Matemática	Renata					Jack	Espanhol	Matemática
	6	Jéssica	Filosofia	Rosa						Glaucia	Matemática
Terça-Feira	1	Christiane	Artes	Renata					Fábio	Matemática	Jéssica
	2	Christiane	Artes	Renata					Fábio	Matemática	Inglês
	3	Jack	Espanhol	Jéssica					Adriana	Ciências	Glaucia
	4	Jack	Português	Jéssica					Adriana	Ciências	Matemática
	5	Verônica	Geografia	Luciana					Renata	Português	Fábio
	6								Beatriz	ED. Física	P. Vida
Quarta-Feira	1	Renata	História	Jéssica					Fábio	Matemática	Jéssica
	2	Renata	História	Rosa					Fábio	Matemática	Inglês
	3	Beatriz	ED. Física	Rosa					Adriana	Ciências	Glaucia
	4	Rosa	Matemática	Jack					Adriana	Ciências	Matemática
	5	Rosa	Matemática	Jack					Jéssica	Artes	Renata
	6								Beatriz	ED. Física	Sociologia
Quinta-Feira	1	Jack	Português	Rosa					Fábio	Matemática	Jéssica
	2	Jack	Português	Rosa					Fábio	Matemática	Inglês
	3	Adriana	Ciências	Jack					Luciana	Português	Fábio
	4	Adriana	Ciências	Jack					Verônica	Geografia	Glaucia
	5	Renata	História	Verônica					Luciana	Português	Fábio
	6								Beatriz	ED. Física	Física
Sexta-Feira	1	Adriana	Ciências	Jack					Fábio	Matemática	Jéssica
	2	Jéssica	Inglês	Jack					Fábio	Matemática	Biologia
	3	Jéssica	Inglês	Adriana					Luciana	Português	Geografia
	4	Jack	Português	Verônica					Verônica	Geografia	Geografia
	5								Jéssica	Artes	Sociologia
	6										

Fonte: Autor

4 ANÁLISE E DISCUSSÃO DOS RESULTADOS

Este capítulo será responsável por evidenciar o estado final do *software*, trazendo a validação das suas funcionalidades.

Conforme amplamente abordado no [Capítulo 3](#), foram desenvolvivos ao longo do projeto três componentes principais: a interface web, a aplicação de servidor e o otimizador de grades. Para validar o funcionamento do sistema como um todo, realizou-se um teste de ponta a ponta, envolvendo os seguintes passos:

1. Criar uma nova conta através da interface;
2. Criar uma nova configuração de grades;
3. Configurar as quantidades de aulas, e todas as métricas de qualidade;
4. Requisitar a geração das grades horárias;
5. Verificar se as configurações foram atendidas corretamente;
6. Realizar a exportação da grade horária no formato XLSX

Após a execução dos itens 1 e 2, foi criada uma configuração que foi utilizada para a validação do *software*. Esta configuração de teste contém dois turnos (“Manhã” e “Tarde”), cada qual com sete turmas. O quadro [4](#) contém a relação de professores, matérias e turmas do turno da manhã.

Quadro 4 – Configuração de aulas para validação.

Professor	Matéria	6ºAno	7ºAno	8ºAno	9ºAno	1ºEM	2ºEM	3ºEM
Adriana	Biologia	0	0	0	0	3	3	3
Adriana	Ciências	3	3	3	3	0	0	0
Adriana	Química	0	0	0	0	3	3	3
Beatriz	ED. Física	1	1	1	1	0	0	0
Cristiane	Artes	2	0	2	0	0	0	0
Fabio	Física	0	0	0	0	3	3	3
Fabio	Matemática	0	0	0	5	0	0	0
Fabio	P. Vida	0	0	0	0	1	1	1
Glaucia	Matemática	0	0	0	0	5	5	5
Jack	Espanhol	1	1	1	1	1	1	1
Jack	Português	5	5	0	0	0	0	0
Jéssica	Artes	0	2	0	2	1	1	1
Jéssica	Filosofia	1	1	1	1	1	1	1
Jéssica	Inglês	2	2	2	2	2	2	2
Luciana	Literatura	0	0	0	0	1	1	1
Luciana	Of. Texto	1	1	1	1	1	1	1
Luciana	Português	0	0	5	5	3	3	3
Renata	História	3	3	3	3	2	3	2
Rosa	Matemática	5	5	5	0	0	0	0
Verônica	Geografia	3	3	3	3	2	2	3
Verônica	Sociologia	0	0	0	0	1	0	0

Fonte: Autoria própria

O quadro [4](#) contém o número de aulas que cada professor deve ministrar de cada

matéria, em cada uma das salas. Por exemplo, seguindo esta configuração, a professora “Adriana” deve ministrar três aulas de química e três aulas de biologia na turma do “3º EM”. Algumas dessas quantidades podem ser vistas configuradas através da interface web na figura 34.

Figura 34 – Aulas configuradas na interface

Aulas

Professor

Adriana

Aulas de Adriana em Manhã

Matéria	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM	Ações
Biologia	0	0	0	0	3	3	3	
Ciências	3	3	3	3	0	0	0	
Química	0	0	0	0	3	3	3	

VINCULAR MATERIA

Beatriz

Aulas de Beatriz em Manhã

Matéria	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM	Ações
ED. Física	1	1	1	1	0	0	0	

VINCULAR MATERIA

Cristiane

Aulas de Cristiane em Manhã

Matéria	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM	Ações
Artes	2	0	2	0	0	0	0	

VINCULAR MATERIA

Fonte: Autor

Por questão de simplicidade, o turno da tarde contém turmas análogas ao turno da manhã, porém com o sufixo “B” para que possam ser diferenciadas. Por exemplo, se no turno da manhã consta a turma “6º Ano”, no turno da tarde constará a turma “6º Ano B”.

Para que o teste contemplasse as diversas métricas de qualidade do sistema, configuraram-se também as outras diversas características das grades horárias, que podem ser verificadas nas figuras a seguir. As figuras representam o estado final das interfaces desenvolvidas durante o projeto, e após sua apresentação, constará uma explicação mais detalhada de cada métrica configurada.

Figura 35 – Constantes configuradas na interface

Constantes		Manhã						
Dia	Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM
Segunda-feira	1	Cristiane						
	2	Cristiane						
	3							
	4							
	5							
	6							
Terça-feira	1							
	2							
	3							
	4							
	5							
	6							
Quarta-feira	1							
	2							
	3							
	4							
	5							
	6							
Quinta-feira	1							
	2							
	3							
	4							
	5							
	6							
Sexta-feira	1							
	2							
	3							
	4							
	5							
	6							

Fonte: Autor

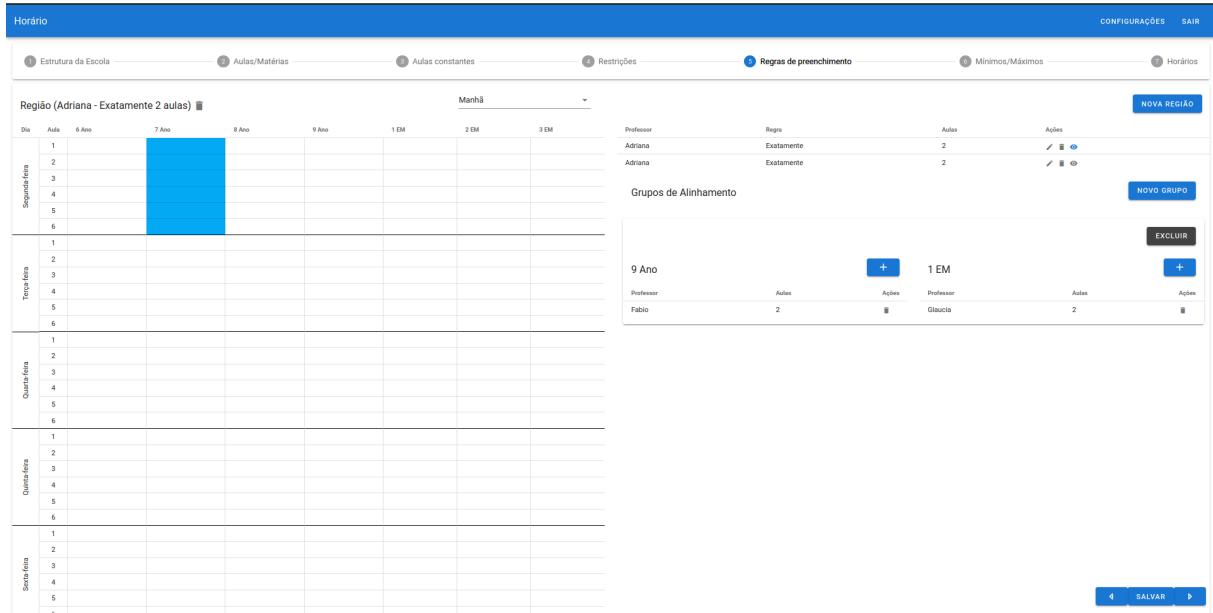
Figura 36 – Restrições e preferências configuradas na interface

Preferências								Manhã	Beatriz
Dia	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM		
Segunda-feira	Forçar última aula								
Terça-feira	Forçar última aula								
Quarta-feira									
Quinta-feira									
Sexta-feira									

Restrições										<input checked="" type="radio"/> Proibir	<input type="radio"/> Evitar
Dia	Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM			
Segunda-feira	1										
	2										
	3										
	4										
	5										
	6										
Terça-feira	1										
	2										
	3										
	4										
	5										
	6										
Quarta-feira	1										
	2										
	3										
	4										
	5										
	6										
Quinta-feira	1										
	2										
	3										
	4										
	5										
	6										
Sexta-feira	1										
	2										
	3										
	4										
	5										
	6										

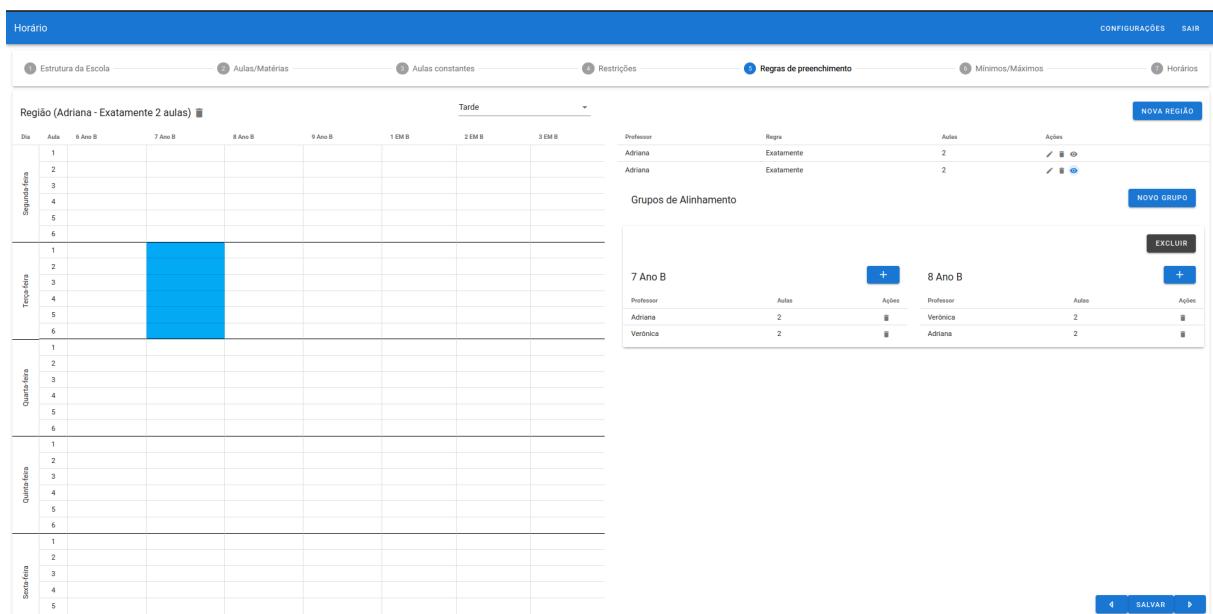
Fonte: Autor

Figura 37 – Regiões e grupos de alinhamento configuradas para o turno da “Manhã”



Fonte: Autor

Figura 38 – Regiões e grupos de alinhamento configuradas para o turno da “Tarde”



Fonte: Autor

Figura 39 – Mínimos de aulas configuradas na interface

Horário						CONFIGURAÇÕES		SAIR															
Estrutura da Escola		Aulas/Matérias		Aulas constantes		Restrições		Regras de preenchimento		Mínimos/Maximos		Horários											
Mínimos																							
Tarde							Limites diários																
Professor	Segunda	Terça	Quarta	Quinta	Sexta		Professor	Segunda	Terça	Quarta	Quinta	Sexta											
Adriana	0	0	0	0	0		Adriana	0	0	0	0	0											
Beatriz	2	1	0	2	1	0	Beatriz	0	0	0	0	0											
Cristiane	0	0	0	0	0		Cristiane	0	0	0	0	0											
Fábio	0	0	0	0	0		Fábio	0	0	0	0	0											
Glaucia	0	0	0	0	0		Glaucia	0	0	0	0	2											
Jack	0	0	0	0	0		Jack	0	0	0	0	0											
Jessica	0	0	0	0	0		Jessica	0	0	0	0	0											
Luciana	0	0	0	0	0		Luciana	0	0	0	0	0											
Luciano	0	0	0	0	0		Luciano	0	0	0	0	0											
Renata	0	0	0	0	0		Renata	0	0	0	0	0											
Rosa	0	0	0	0	0		Rosa	0	0	0	0	0											
Verônica	0	0	0	0	0		Verônica	0	0	0	0	0											

Fonte: Autor

- Constantes:** Conforme a figura 35, definiu-se que “Cristiane” deveria lecionar as duas primeiras aulas da segunda-feira na turma do “6º Ano”, em ambos os turnos;
- Restrições e preferências:** Na figura 36, é possível notar que no turno da manhã, “Beatriz” não deve ter aulas alocadas na quarta, quinta ou sexta-feira. Além disso, informou-se pela interface que suas aulas devem ser lecionadas obrigatoriamente como a última aula do expediente de cada turma;
- Regiões:** Utilizando o conceito das regiões, definiu-se que “Adriana” deveria lecionar exatamente duas aulas para a turma do “7º Ano” na segunda-feira, e exatamente 2 aulas para “7º Ano B” na terça-feira. Estas configurações abrangem tanto o turno da “Manhã”, quanto da “Tarde”, como pode ser visto nas figuras 37 e 38;
- Grupos de alinhamento:** Definiu-se que duas aulas do professor “Fábio” no “9º Ano” devem ser alocadas simultaneamente a duas aulas da professora “Glaucia” no “1º EM”. Além disso, duas aulas de “Adriana” no “7º Ano B” devem ser alocadas simultaneamente a duas aulas de “Verônica” no “8º Ano B”, e vice-versa. Assim como as regiões, as configurações dos grupos de alinhamento podem ser conferidas nas figuras 37 e 38;
- Mínimos e limites diários:** Como pode ser visto na figura 39, configurou-se que no turno da tarde, “Beatriz” deve ministrar no mínimo duas aulas na segunda-feira e 2 aulas na quarta-feira; além disso, “Gláucia” deve ministrar no máximo duas aulas na sexta-feira, contabilizando ambos os turnos.

Realizadas as configurações, o otimizador foi ativado, e utilizou-se como condição de parada a passagem de um intervalo de tempo de um minuto. Como pode ser visto na figura 40 Durante este intervalo, o otimizador produziu 65 soluções diferentes, cada qual contendo uma grade horária para o turno da manhã e uma para o turno da tarde.

Figura 40 – Listagem de grades horárias geradas

Horário									CONFIGURAÇÕES	SAIR		
Estrutura da Escola		Aulas/Matérias		Aulas constantes		Restrições		Regras de preenchimento		Mínimos/Máximos		Horários
Horários Gerados												GERAR
	Aulas separadas	Restrições não resolvidas	Restrições opcionais não resolvidas	Preferências não resolvidas	Ausências de aulas duplas	Espaços janelas	Problemas	Detalhes	Ações			
142	0	0	0	0	0	10	0	480				
150	0	0	0	0	0	0	0	484				
152	0	0	0	0	0	0	0	486				
152	0	0	0	0	0	0	0	486				
152	0	0	0	0	0	0	0	486				
152	0	0	0	0	0	0	0	486				
150	0	0	0	0	0	0	0	488				
152	0	0	0	0	0	0	0	488				
150	0	0	0	0	0	0	0	488				
150	0	0	0	0	0	0	0	488				
Linhas por página: 10 100 de 103 < < 1 > >												

Fonte: Autor

Figura 41 – Visualização de turno da manhã da melhor solução

Horário (1727) < > 🔍 ⓘ											
	Dia	Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM		
Segunda-feira	1	Cristiane	Artes	Jéssica	Inglês	Luciana	Português	Renata	História	Adriana	Química
	2	Cristiane	Artes	Renata	História	Luciana	Of. Texto	Jéssica	Inglês	Adriana	Biologia
	3	Verônica	Geografia	Jack	Espanhol	Rosa	Matemática	Adriana	Ciências	Jéssica	Filosofia
	4	Luciana	Of. Texto	Adriana	Ciências	Rosa	Matemática	Jack	Espanhol	Fábio	Física
	5	Rosa	Matemática	Adriana	Ciências	Beatriz	ED. Física	Luciana	Português	Jéssica	Inglês
	6		Beatriz		ED. Física			Luciana	Português	Jéssica	Inglês
Terça-feira	1	Renata	História	Verônica	Geografia	Jéssica	Inglês	Luciana	Of. Texto	Glaucia	Matemática
	2	Renata	História	Verônica	Geografia	Jéssica	Filosofia	Luciana	Português	Glaucia	Matemática
	3	Jéssica	Inglês	Renata	História	Verônica	Geografia	Fábio	Matemática	Adriana	Química
	4	Jack	Português	Renata	História	Verônica	Geografia	Jéssica	Filosofia	Adriana	Biologia
	5	Jack	Português	Rosa	Matemática	Adriana	Ciências	Beatriz	ED. Física	Luciana	Português
	6	Beatriz	ED. Física							Renata	História
Quarta-feira	1	Adriana	Ciências	Luciana	Of. Texto	Jack	Espanhol	Renata	História	Verônica	Geografia
	2	Adriana	Ciências	Jack	Português	Luciana	Português	Renata	História	Verônica	Geografia
	3	Renata	História	Jack	Português	Jéssica	Inglês	Verônica	Geografia	Luciana	Português
	4	Jack	Português	Jéssica	Artes	Renata	História	Fábio	Matemática	Glaucia	Matemática
	5	Jack	Português	Jéssica	Inglês	Cristiane	Artes	Fábio	Matemática	Adriana	Química
	6	Rosa	Matemática	Adriana	Ciências	Cristiane	Artes	Fábio	P. Vida	Jack	Espanhol
Quinta-feira	1	Jéssica	Filosofia	Rosa	Matemática	Luciana	Português	Fábio	Matemática	Adriana	Química
	2	Jéssica	Inglês	Rosa	Matemática	Luciana	Português	Fábio	Matemática	Glaucia	Matemática
	3	Jack	Espanhol	Jéssica	Filosofia	Rosa	Matemática	Verônica	Geografia	Adriana	Biologia
	4	Rosa	Matemática	Jack	Português	Adriana	Ciências	Verônica	Geografia	Luciana	Português
	5	Rosa	Matemática	Jack	Português	Adriana	Ciências	Jéssica	Inglês	Verônica	Geografia
	6									Jéssica	Inglês
Sexta-feira	1	Adriana	Ciências	Jack	Português	Rosa	Matemática	Jéssica	Artes	Renata	História
	2	Jack	Português	Verônica	Geografia	Rosa	Matemática	Jéssica	Artes	Renata	História
	3	Verônica	Geografia	Rosa	Matemática	Renata	História	Luciana	Português	Luciana	Português
	4	Verônica	Geografia	Rosa	Matemática	Renata	História	Luciana	Português	Jack	Espanhol
	5	Rosa	Matemática	Jéssica	Artes	Verônica	Geografia	Adriana	Ciências	Fábio	Matemática
	6					Luciana	Português	Adriana	Ciências	Verônica	Sociologia

Fonte: Autor

A listagem da figura 40 é dada em ordem de qualidade, ou seja, a primeira grade horária da lista apresenta os melhores valores para as métricas de qualidade. A figura 41 contém a grade horária do turno da manhã presente na melhor solução encontrada, cujas métricas de qualidade foram transcritas para o quadro 5. Para melhor leitura e análise, as informações de ambos os turnos dessa solução foram exportadas para as planilhas nas figuras 42 e 43, respectivamente.

Quadro 5 – Métricas de qualidade da melhor solução.

Nome	Valor	Tipo
Excessos de matérias iguais	0	Rígida
Restrições violadas	0	Rígida
Limites diários violados	0	Rígida
Aulas em grupos de alinhamento não alinhadas	0	Rígida
Mínimos opcionais violados	0	Rígida
Mínimos violados	0	Rígida
Erros nas regiões	0	Rígida
Matérias desagrupadas	0	Rígida
Conflitos	0	Rígida
Aulas desagrupadas	0	Rígida
Excessos de aulas iguais	0	Rígida
Dias com todos professores diferentes	0	Rígida
Restrições opcionais violadas	0	Suave
Horários com janela (total de todos professores)	10	Suave
Número de grupos de alinhamento formados	4	Suave
Preferências não resolvidas	0	Suave
Aulas separadas	142	Suave
Dias de professores sem aulas duplas	0	Suave
Matérias separadas	208	Suave

Fonte: Autoria própria

Figura 42 – Grade exportada - Turno da Manhã

Dia	Aula	6 Ano	7 Ano	8 Ano	9 Ano	1 EM	2 EM	3 EM							
Segunda-feira	1	Artes	Cristiane	Inglês	Jéssica	Português	Luciana	História	Renata	Química	Adriana	Matemática	Glaucia	Geografia	Verônica
	2	Artes	Cristiane	História	Renata	Of. Texto	Luciana	Inglês	Jéssica	Biologia	Adriana	Matemática	Glaucia	Geografia	Verônica
	3	Geografia	Verônica	Espanhol	Jack	Matemática	Rosa	Ciências	Adriana	Filosofia	Jéssica	Literatura	Luciana	Matemática	Glaucia
	4	Of. Texto	Luciana	Ciências	Adriana	Matemática	Rosa	Espanhol	Jack	Física	Fabio	Ingles	Jéssica	Matemática	Glaucia
	5	Matemática	Rosa	Ciências	Adriana	ED. Física	Beatriz	Português	Luciana	Inglês	Jéssica	P. Vida	Fabio	Espanhol	Jack
	6			ED. Física	Beatriz			Português	Luciana	Ingles	Jéssica	Física	Fabio	Biologia	Adriana
Terça-feira	1	História	Renata	Geografia	Verônica	Inglês	Jéssica	Of. Texto	Luciana	Matemática	Glaucia	Química	Adriana	P. Vida	Fabio
	2	História	Renata	Geografia	Verônica	Filosofia	Jéssica	Português	Luciana	Matemática	Glaucia	Biologia	Adriana	Física	Fabio
	3	Inglês	Jéssica	História	Renata	Geografia	Verônica	Matemática	Fabio	Química	Adriana	Matemática	Glaucia	Literatura	Luciana
	4	Português	Jack	História	Renata	Geografia	Verônica	Filosofia	Jéssica	Biologia	Adriana	Of. Texto	Luciana	Matemática	Glaucia
	5	Português	Jack	Matemática	Rosa	Ciências	Adriana	ED. Física	Beatriz	Português	Luciana	História	Renata	Ingles	Jéssica
	6			ED. Física	Beatriz			Literatura	Luciana	História	Renata	Biologia	Adriana		
Quarta-feira	1	Ciências	Adriana	Of. Texto	Luciana	Espanhol	Jack	História	Renata	Geografia	Verônica	Ingles	Jéssica	Física	Fabio
	2	Ciências	Adriana	Português	Jack	Português	Luciana	História	Renata	Geografia	Verônica	Filosofia	Jéssica	Física	Fabio
	3	História	Renata	Português	Jack	Inglês	Jéssica	Geografia	Verônica	Português	Luciana	Física	Fabio	Biologia	Adriana
	4	Português	Jack	Artes	Jéssica	História	Renata	Matemática	Fabio	Matemática	Glaucia	Português	Luciana	Química	Adriana
	5	Português	Jack	Inglês	Jéssica	Artes	Cristiane	Matemática	Fabio	Matemática	Glaucia	Química	Adriana	Português	Luciana
	6	Matemática	Rosa	Ciências	Adriana	Artes	Cristiane			P. Vida	Fabio	Espanhol	Jack	Português	Luciana
Quinta-feira	1	Filosofia	Jéssica	Matemática	Rosa	Português	Luciana	Matemática	Fabio	Química	Adriana	Matemática	Glaucia	História	Renata
	2	Inglês	Jéssica	Matemática	Rosa	Português	Luciana	Matemática	Fabio	Biologia	Adriana	Matemática	Glaucia	História	Renata
	3	Espanhol	Jack	Filosofia	Jéssica	Matemática	Rosa	Geografia	Verônica	Matemática	Glaucia	Biologia	Adriana	Português	Luciana
	4	Matemática	Rosa	Português	Jack	Ciências	Adriana	Geografia	Verônica	Português	Luciana	Artes	Jéssica	Matemática	Glaucia
	5	Matemática	Rosa	Português	Jack	Ciências	Adriana	Inglês	Jéssica	Of. Texto	Luciana	Geografia	Verônica	Matemática	Glaucia
	6									Artes	Jéssica	Geografia	Verônica	Química	Adriana
Sexta-feira	1	Ciências	Adriana	Português	Jack	Matemática	Rosa	Artes	Jéssica	História	Renata	Português	Luciana	Geografia	Verônica
	2	Português	Jack	Geografia	Verônica	Matemática	Rosa	Artes	Jéssica	História	Renata	Português	Luciana	Química	Adriana
	3	Geografia	Verônica	Matemática	Rosa	História	Renata	Português	Luciana	Espanhol	Jack	Química	Adriana	Filosofia	Jéssica
	4	Geografia	Verônica	Matemática	Rosa	História	Renata	Português	Luciana	Física	Fabio	Biologia	Adriana	Ingles	Jéssica
	5	Matemática	Rosa	Artes	Jéssica	Geografia	Verônica	Ciências	Adriana	Física	Fabio	História	Renata	Of. Texto	Luciana
	6					Português	Luciana	Ciências	Adriana	Sociologia	Verônica	Física	Fabio	Artes	Jéssica

Fonte: Autor

Figura 43 – Grade exportada - Turno da Tarde

Dia	Aula	6 Ano B	7 Ano B	8 Ano B	9 Ano B	1 EM B	2 EM B	3 EM B	
Segunda-feira	1	Artes	Cristiane	Português	Jack	História	Renata	Ciências	Adriana
	2	Artes	Cristiane	Português	Jack	História	Renata	Artes	Jéssica
	3	Português	Jack	ED. Física	Beatriz	Português	Luciana	Filosofia	Jássica
	4	Matemática	Rosa	Filosofia	Jéssica	Português	Luciana	Biologia	Adriana
	5	Inglês	Jéssica	Matemática	Rosa	Espanhol	Jack	Matemática	Glaucia
	6	Filosofia	Jéssica			Of. Texto	Luciana	Sociologia	Gláucia
Terça-feira	1	Matemática	Rosa	Of. Texto	Luciana	Inglês	Jéssica	Geografia	Verônica
	2	Geografia	Verônica	Matemática	Rosa	Inglês	Jéssica	História	Renata
	3	História	Renata	Geografia	Verônica	Ciências	Adriana	Literatura	Luciana
	4	Espanhol	Jack	Geografia	Verônica	Ciências	Adriana	Matemática	Glaucia
	5	Português	Jack	Ciências	Adriana	Geografia	Verônica	Matemática	Jássica
	6			Ciências	Adriana	Geografia	Verônica	Física	Fábio
Quarta-feira	1	Matemática	Rosa	História	Renata	Artes	Cristiane	Espanhol	Biologia
	2	Matemática	Rosa	História	Renata	Artes	Cristiane	Espanhol	Química
	3	Geografia	Verônica	Matemática	Rosa	História	Renata	Português	Adriana
	4	Geografia	Verônica	Matemática	Rosa	Ciências	Adriana	P. Vida	Fábio
	5	ED. Física	Beatriz	Artes	Jéssica	Matemática	Rosa	Geografia	Verônica
	6					Matemática	Rosa	Filosofia	Jássica
Quinta-feira	1	Ciências	Adriana	Português	Jack	Matemática	Rosa	Geografia	Verônica
	2	Ciências	Adriana	Português	Jack	Matemática	Rosa	Geografia	Verônica
	3	História	Renata	Geografia	Verônica	Espanhol	Jack	Filosofia	Renata
	4	História	Renata	Espanhol	Jack	Português	Luciana	Português	Português
	5	Português	Jack	Inglês	Jéssica	Português	Luciana	Matemática	Luciana
	6					Matemática	Fabio	Geografia	Verônica
Sexta-feira	1	Português	Jack	Matemática	Rosa	Filosofia	Jéssica	Matemática	Português
	2	Português	Jack	Artes	Jéssica	Matemática	Fabio	Biologia	Português
	3	Matemática	Rosa	Inglês	Jéssica	Of. Texto	Luciana	Geografia	Luciana
	4	Inglês	Jéssica	Português	Jack	Português	Luciana	História	Espanhol
	5	Ciências	Adriana	História	Renata	Geografia	Verônica	Matemática	Fábio
	6	Of. Texto	Luciana	Ciências	Adriana	Inglês	Jéssica	Física	Geografia

Fonte: Autor

Observando as figuras 42 e 43, constata-se que:

1. Foi alocada a quantidade correta de aulas para cada professor e matéria, nas salas corretas, de acordo com o quadro 4;
2. As aulas foram alocadas sem causar conflitos, ou seja, em nenhum horário das grades, um professor está alocado para mais de uma turma simultaneamente;
3. Aproximadamente 64% das aulas foram agrupadas: conforme o quadro 5, apenas 142 do total de 396 aulas permaneceram separadas;
4. As aulas constantes mencionadas no item 1 foram alocadas corretamente: Cristiane foi alocada nas duas primeiras aulas da segunda-feira na turma do 6º Ano e 6º Ano B;
5. As restrições e preferências do item 2 foram atendidas: “Beatriz” não teve aulas alocadas na quarta, quinta ou sexta-feira no turno da manhã, e neste mesmo turno suas aulas ocuparam sempre os últimos horários do expediente de cada turma;
6. Em conformidade com o item 3, na segunda-feira a professora “Adriana” teve exatamente duas aulas alocadas na turma do “7º Ano”, e duas no “7º Ano B”;
7. O grupo de alinhamento da figura 37 foi respeitado, já que duas aulas do professor “Fábio” no “9º Ano” foram alocadas simultaneamente a duas aulas de “Adriana” no “1º EM”. As aulas do grupo da figura 38 também foram alocadas corretamente;
8. As quantidades mínimas e máximas de aulas do item 5 foram respeitadas: foram alocadas para “Beatriz” exatamente duas aulas na segunda e quarta-feira no período da tarde; e exatamente duas aulas para “Gláucia” na sexta-feira, somando as aulas de ambos os turnos;

5 CONSIDERAÇÕES FINAIS

Em conclusão, este trabalho abordou o *High School Timetabling Problem*, e a demanda associada por um software de otimização de grades horárias. Verificou-se durante a revisão de literatura a dificuldade que a tarefa de planejamento de grades horárias representa, e que grande parte das instituições de ensino do país ainda realiza essa tarefa manualmente.

O presente trabalho trouxe também algumas demonstrações do estado atual do protótipo da aplicação desenvolvida, assim como melhorias sugeridas para a continuação da pesquisa. Com o desenvolvimento dos primeiros incrementos da aplicação, foi possível verificar a viabilidade do projeto aplicando a técnica de *Simulated Annealing* para produzir e otimizar grades horárias escolares, atendendo múltiplos objetivos tradicionais do problema, assim como a nova preocupação trazida pelo Novo Ensino Médio, os Itinerários Formativos.

Resta portanto como atividade de crucial importância a coleta de dados de ponteciais usuários futuros da aplicação, para que seja possível verificar a necessidade de adições e melhorias ao otimizador, e a execução dos incrementos do software a fim de cumprir os objetivos propostos.

Referências

- ABRAMSON, D. Constructing school timetables using simulated annealing: Sequential and parallel algorithms. **Management Science**, v. 37, n. 1, p. 98–113, 1991. Disponível em: <<https://EconPapers.repec.org/RePEc:inm:ormnsc:v:37:y:1991:i:1:p:98-113>>. Citado 4 vezes nas páginas 1, 3, 6 e 7.
- ABRAMSON, D. et al. Simulated annealing cooling schedules for the school timetabling problem. **Asia Pacific Journal of Operational Research**, Citeseer, v. 16, p. 1–22, 1999. Citado na página 7.
- BARDADYM, V. A. Computer-aided school and university timetabling: The new wave. In: BURKE, E.; ROSS, P. (Ed.). **Practice and Theory of Automated Timetabling**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996. p. 22–45. ISBN 978-3-540-70682-3. Citado na página 1.
- BRASIL. Lei nº 13.415, de 16 de fevereiro de 2017. **Diário Oficial [da] República Federativa do Brasil**, Brasília, DF, 2017. ISSN 1677-7042. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2017/lei/l13415.htm>. Citado na página 3.
- COOPER, T. B.; KINGSTON, J. H. The complexity of timetable construction problems. In: **International Conference on the Practice and Theory of Automated Timetabling**. [S.l.: s.n.], 1995. Citado na página 3.
- FONSECA, G. H.; SANTOS, H. G.; CARRANO, E. G. Integrating matheuristics and metaheuristics for timetabling. **Computers And Operations Research**, v. 74, p. 108–117, 2016. ISSN 0305-0548. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0305054816300879>>. Citado na página 1.
- LAARHOVEN, P. J. **Simulated annealing theory and applications**. [S.I.]: Kluwer, 1987. Citado na página 4.
- POULSEN, C. J. B. **Desenvolvimento de um Modelo para o School Timetabling Problem Baseado na Meta-Heurística Simulated Annealing**. 141 p. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012. Citado na página 1.
- PRESSMAN, R.; MAXIM, B. **Engenharia de Software - 8ª Edição**. [s.n.], 2016. ISBN 9788580555349. Disponível em: <<https://books.google.com.br/books?id=wexzCwAAQBAJ>>. Citado na página 5.
- TAN, J. S. et al. A survey of the state-of-the-art of optimisation methodologies in school timetabling problems. **Expert Systems with Applications**, v. 165, p. 113943, 2021. ISSN 0957-4174. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0957417420307314>>. Citado 2 vezes nas páginas 1 e 3.