

SUMÁRIO

1	MODELAGEM GERAL DO SISTEMA	2
1.1	Estágios de execução	2
1.2	Iteração	3
1.3	Funcionamento	3
1.4	Modelo de banco de dados	5
1.5	API REST	6
	REFERÊNCIAS	7

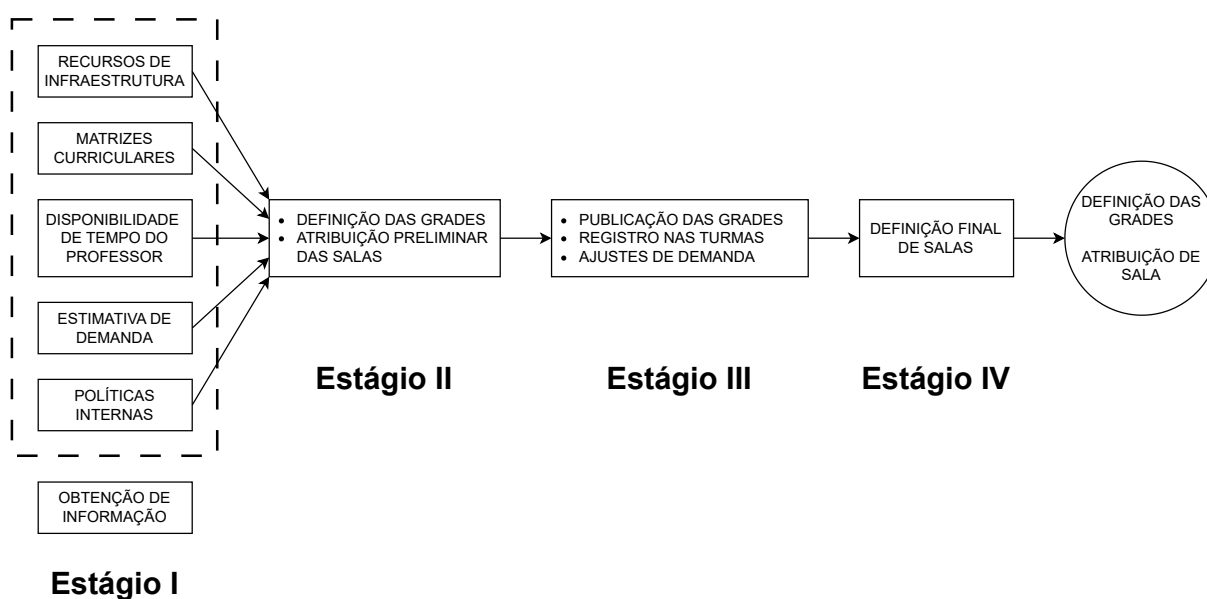
1 MODELAGEM GERAL DO SISTEMA

Tendo esclarecido sobre as questões gerais da área de estudo e de como se estrutura a instituição agora nos aprofundaremos na conceitualização do funcionamento geral do sistema e a forma como se dará a execução da metodologia proposta.

1.1 Estágios de execução

Em seu trabalho de aplicação prática, Miranda, Rey e Robles (2012) estruturaram estágios que compõem o processo necessário para que enfim se alcance a definição de tabelas horárias finais.

Figura 1 – Estágios para a obtenção de grade horária ótima



Fonte: Miranda, Rey e Robles (2012) - adaptado

Na Figura 1, estão dispostos 4 estágios principais. O primeiro dispõe da aquisição de informações, sendo elas a disponibilidade do professor, os recursos da infraestrutura, as grades dos cursos, as estimativas de demanda e as políticas internas. No segundo estágio são definidas grades horárias preliminares com atribuição preliminar das salas. No terceiro, os alunos se inscrevem e a demanda é ajustada, por fim, no quarto estágio, ocorre a alocação final das salas. Com sua conclusão, são definidos as grades horárias finais junto com as respectivas salas.

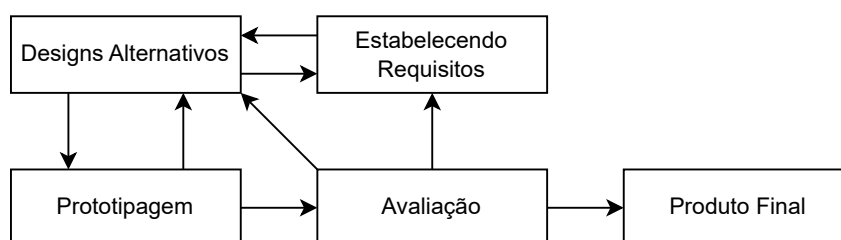
A sequência geral condiz com o processo de criação das grades horárias, porém é necessário adaptar a metodologia para o contexto da UENF, e também ao escopo do

trabalho. Sendo assim, no primeiro estágio a coleta de informações sobre a disponibilidade dos professores e as matrizes curriculares dos cursos não será necessária. Os recursos de infraestrutura serão coletados dos histórico de alocação das turmas, a estimativa de demandas será obtida através da média histórica de alunos inscritos nas turmas e as políticas internas serão compreendidas através da revisão documental e entrevistas com os *stakeholders*. O segundo tem caráter iterativo entre as coordenações e a diretoria do CCT. Os outros estágios não se distanciarão de forma significativa do que foi proposto.

1.2 Iteração

Para se alcançar uma alta satisfação por parte dos *stakeholders*, vê-se necessária a constante interação com os mesmos. Para isto, será seguida a estrutura utilizada por Andre e Dinata (2018).

Figura 2 – Etapas do Design de Interação



Fonte: Andre e Dinata (2018) - adaptado

Seguindo o conceito do Design de Interação, a Figura 2 ilustra o ciclo de ações a serem tomadas durante o desenvolvimento do sistema, caso este venha a ser necessário. Nesse modelo de pesquisa, os *stakeholders* serão consultados continuamente enquanto lhes for apresentados protótipos do sistema, para que assim informem quanto às suas percepções. Esta dinâmica tem como finalidade encontrar um design tal que seja adequado aos desejos e necessidades de seus usuários. Pois, considerando que para que o sistema seja efetivo, é necessário que ele seja aceito e utilizado pelos usuários finais.

1.3 Funcionamento

O sistema proposto funcionará de forma a auxiliar a coordenação do curso de Ciência da Computação da UENF criação de grades horárias para os semestres letivos. Para isso, o sistema deverá ser capaz de gerenciar as informações referentes às disciplinas, professores, salas e horários disponíveis.

Para tanto, mesmo que disponha de informações pré-cadastradas, o sistema deverá permitir a inserção de novas informações, como disciplinas, professores e salas, pois, apesar

de não ser o foco principal do sistema, é recorrente que haja alterações nessas informações ao longo do tempo, e caso o sistema não as comporte, não poderá cumprir seu propósito.

Como forma de agregar as informações cadastradas, o usuário será capaz de criar turmas para as disciplinas, informando o professor responsável, a sala e o horário em que a turma ocorrerá. Além disso, o sistema deverá permitir a visualização das turmas já cadastradas, bem como a edição e exclusão das mesmas, caso necessário. Ao criar uma turma, o sistema viabiliza a definição de qual ano e semestre a turma pertence.

As turmas contarão também com a informação de demanda estimada, que será utilizada como comparativo para a alocação das salas, de forma a evitar que uma sala seja alocada para uma turma cuja demanda seja maior do que a capacidade da sala. Outra informação contida nas turmas será um código descritor da turma em questão. Esse código será utilizado para auxiliar na compreensão das turmas, visto que ao longo do processo de criação das grades horárias é recorrente a criação de turmas criadas com um propósito específico, como por exemplo contemplar apenas repetentes ou alunos de um determinado conjunto de cursos.

Como ponto chave do funcionamento do sistema, o mesmo deverá ser capaz de ilustrar conflitos que vierem a surgir durante a alocação dos recursos. Esses conflitos podem ser de diversas naturezas, como por exemplo, a alocação de uma turma em um horário em que o professor responsável não está disponível, ou a alocação de duas turmas em uma mesma sala e horário. A identificação desses conflitos é essencial para que a coordenação possa corrigi-los de forma prática e direta, assim evitando que ocorram problemas durante a execução das turmas no semestre letivo.

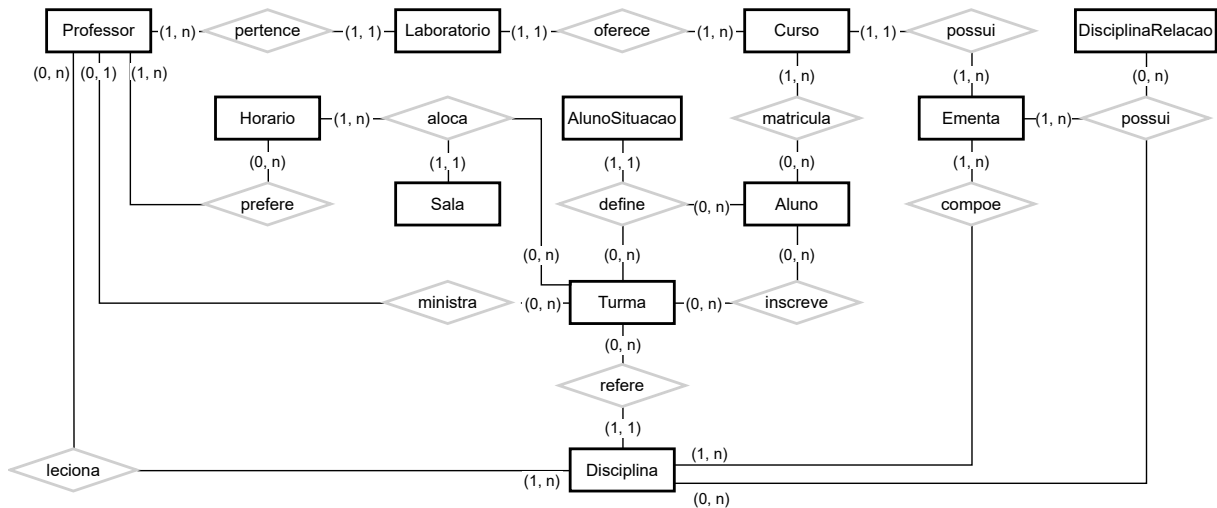
Por fim, o sistema viabiliza também um método de rápida criação de turmas para todas as disciplinas esperadas para determinado semestre letivo para o curso de Ciência da Computação. Esse método consiste em analisar as disciplinas esperadas para os estudantes de computação em determinado semestre letivo, e, através do cálculo das maiores recorrências de professor, sala e horário atribuídos às turmas dessa mesma disciplina em semestres anteriores, criar uma turma com essas informações. Esse método visa agilizar o processo de criação das turmas, visto que a coordenação não precisará criar cada turma manualmente, mas sim apenas revisar as informações geradas pelo sistema.

Dispondo de todas essas funcionalidades, o sistema deverá ser capaz de gerar uma grade horária final, que será utilizada como base para a criação da grade horária oficial do curso de Ciência da Computação da UENF.

1.4 Modelo de banco de dados

Considerando as informações necessárias para o presente trabalho, e também o preparo de campo para potenciais aplicações futuras, foi elaborado um diagrama conceitual parcial do banco de dados (Figura 3).

Figura 3 – Diagrama Conceitual do banco de dados



Fonte: autoria própria

O diagrama conceitual foi desenvolvido utilizando a ferramenta draw.io¹ citada na metodologia e ilustra as relações entre diversas entidades presentes na realidade da UENF. O emaranhamento presente no diagrama ilustra a complexidade envolvida na criação de uma grade horária, onde diversas entidades se relacionam entre si.

Como principais apontamentos, podemos citar a parte principal do modelo que é a alocação de turmas. Ela, como já descrito, envolve a correlação entre alunos de diferentes cursos, professores, disciplinas, salas e horários. Além disso, também é possível notar a presença de entidades que não são diretamente relacionadas à alocação de turmas, mas que podem se mostrar úteis no futuro, como a relação entre professores e laboratórios; e a de disciplinas e ementas.

Embora o diagrama apresente uma visão mais completa das interconexões possíveis, é importante ressaltar que o presente trabalho foca primordialmente na alocação das turmas para o curso de Ciência da Computação, e que a implementação do banco de dados será feita de forma a atender a essas necessidades, fazendo então uso de apenas uma parte do diagrama conceitual apresentado pela Figura 3. Sendo posteriormente ilustrado (??, ?? e ??).

¹ <https://www.drawio.com>

1.5 API REST

Para exemplificar o funcionamento geral da permanência dos dados, consideremos o uso de uma *API REST* utilizando de 4 “camadas”: o **frontend**, os **endpoints**, as **funções** de execução e o **banco de dados**.

O **frontend** é a interface do sistema, onde o usuário interage com o sistema. Essa interface precisa se comunicar com o *backend* para realizar as quatro operações básicas no banco de dados (criação, leitura, atualização e deleção) por sobre as entidades existentes (turmas, professores, disciplinas, salas, etc.). Elas assim o fazem ao enviar requisições HTTP (*GET*, *POST*, *PUT* e *DELETE*), contendo pacotes de informações em formato JSON para os **endpoints**.

Os **endpoints** são as rotas que o *backend* disponibiliza para a recepção das requisições HTTP. Eles são responsáveis por encaminhar as requisições recebidas. Seu funcionamento consiste em rotear as requisições recebidas junto com sua carga útil. Para tanto, as rotas criadas refletem diretamente a qual entidade do banco de dados a requisição se refere, sendo então assim sabido qual **função** deve ser executada.

As **funções** são as responsáveis por executar as operações no banco de dados. Elas processam o pacote de informações recebido, e então realizam a operação desejada no **banco de dados**.

O **banco de dados** recebe a requisição, processa a operação, e então retorna o status da operação. Esse retorno é então repassado camada por camada, até chegar ao *frontend*, onde o usuário final pode visualizar o resultado da operação.

Resumidamente: O **frontend** envia uma requisição HTTP com uma carga de informações a um **endpoint**, que encaminha a requisição a uma **função** específica que executa uma operação no **banco de dados**, assim retornando o status da operação ao *frontend*.

REFERÊNCIAS

ANDRE, A.; DINATA, H. Interaction design to enhance ux of university timetable plotting system on mobile version. IOP Conference Series: Materials Science and Engineering, IOP Publishing, v. 407, n. 1, p. 012174, set. 2018. ISSN 1757-899X. Disponível em: <https://doi.org/10.1088/1757-899X/407/1/012174>. Acesso em: 25/04/2024. Citado na página 3.

MIRANDA, J.; REY, P. A.; ROBLES, J. M. udpskeder: A web architecture based decision support system for course and classroom scheduling. Decision Support Systems, v. 52, n. 2, p. 505–513, jan. 2012. ISSN 01679236. Disponível em: <https://doi.org/10.1016/j.dss.2011.10.011>. Acesso em: 25/04/2024. Citado na página 2.