

UNIVERSIDADE VIRTUAL DO ESTADO DE SÃO PAULO

Antônio Fernandes Braga Neto, 2205996

Douglas Andrade Fernandes, 2210166

Elisandra Márcia Rodrigues, 2209506

Leonardo Mauricio Ramos, 2211065

Pablo Miguel dos Santos Andrade, 2214448

Rubens Antônio Rodrigues, 2200586

**Desenvolvimento do software assistente (Agenda da Beleza) para
agendamento e controle de horários e serviços para cabeleireiros e
barbeiros**

Araraquara – SP

2024

UNIVERSIDADE VIRTUAL DO ESTADO DE SÃO PAULO

**Desenvolvimento do software assistente (Agenda da Beleza) para
agendamento e controle de horários e serviços para cabeleireiros e
barbeiros**

Relatório Técnico-Científico apresentado na
disciplina de Projeto Integrador para o curso de
Engenharia de Computação da Universidade
Virtual do Estado de São Paulo (UNIVESP).

Araraquara – SP

2024

BRAGA NETO, Antônio Fernandes; FERNANDES, Douglas Andrade; RODRIGUES, Elisandra Márcia; RAMOS, Leonardo Maurício; ANDRADE, Pablo Miguel dos Santos; RODRIGUES, Rubens Antônio. Desenvolvimento do software assistente (**Agenda da Beleza**) para agendamento e controle de horários e serviços para cabeleireiros e barbeiros. 00f. Relatório Técnico-Científico. Engenharia da Computação - curso – **Universidade Virtual do Estado de São Paulo**. Tutor: Mariana Ruzzi. Polo Araraquara-SP, 2024.

RESUMO

Muitos prestadores de serviços, como cabeleireiros e barbeiros, possuem problemas relacionados ao agendamento de horários e serviços, sem controle de dados e, muitas vezes o cliente pode ser prejudicado com erros ocasionados pela falta de organização, falhas ao atualizar o cadastro de horários e monitoramento dos atendimentos prestados. Esse projeto visa desenvolver um sistema com framework web que seja capaz de gerir o banco de dados desses profissionais a fim de contribuir para a resolução de problemas relacionados ao agendamento de horários, bem como de seus serviços, além de oferecer comodidade aos seus clientes. Com esse intuito foram realizadas entrevistas com esses profissionais para coletar informações sobre sua rotina profissional, serviços prestados e as dificuldades recorrentes. No desenvolvimento desse software (Agenda da Beleza) foi utilizado a framework web Java Server Faces (JSF) com a linguagem Java EE no ambiente NetBeans e banco de dados MySQL. Para a versão de controle de versão foi usado o GitHub, por permitir criar o sincronismo mais facilmente. Após a implementação do software espera-se que os problemas recorrentes com agendamentos e gerenciamento dos serviços prestados sejam corrigidos e, que o software por ser Web possa tornar mais dinâmica a rotina desses profissionais, os quais poderão otimizar seu tempo e evitar falhas de agendamento de seus serviços, bem como oferecer comodidade aos seus clientes como favorecimento de mobilidade, já que o sistema poderá ser acessado de qualquer dispositivo conectado à internet.

PALAVRAS-CHAVE: Prestadores de Serviços; Banco de Dados; Framework; Java EE; Aplicação Web.

LISTA DE ILUSTRAÇÕES

Figura 1 - Metodologia Design Thinking.....	12
Figura 2 - Modelo Arquitetural do software Agenda da Beleza.....	15
Figura 3 - Pré-visualização da Área Pública do software Agenda da Beleza.....	16

SUMÁRIO

1 INTRODUÇÃO	1
2 DESENVOLVIMENTO	3
2.1 OBJETIVOS.....	3
2.2 JUSTIFICATIVA E DELIMITAÇÃO DO PROBLEMA.....	4
2.3 FUNDAMENTAÇÃO TEÓRICA.....	4
2.4 APLICAÇÃO DAS DISCIPLINAS ESTUDADAS NO PROJETO INTEGRADOR	7
2.5 METODOLOGIA.....	11
2.6 RESULTADOS PRELIMINARES: SOLUÇÃO INICIAL	14
REFERÊNCIAS.....	17
ANEXOS	20

1 INTRODUÇÃO

O setor de prestação de serviços é um dos mais importantes para a economia: em 2021, cresceu 10,9% mesmo com a pandemia, segundo o Instituto Brasileiro de Geografia e Estatística (IBGE). A Pesquisa Anual de Serviços - PAS do IBGE desenvolve esse trabalho com uso de dados por amostragem probabilística, que tem por função principal identificar as características estruturais básicas da atividade de serviços não financeiros no País, excetuando-se Saúde e Educação, e suas transformações no tempo. Seus principais objetivos são: dimensionar a produção, faturamento, custos e despesas, consumo intermediário, gastos com a folha de pagamento, volume de pessoas ocupadas, despesas com formação de capital, dentre outras características estruturais do segmento de serviços do País, por atividade detalhada da Classificação Nacional de Atividades Econômicas – CNAE (<https://www.ibge.gov.br>). Assim, fazem parte desse segmento, trabalhadores como cabeleireiros e barbeiros, os quais oferecem seus serviços à população muitas vezes num modelo com total informalidade. É de fácil percepção a falta de organização com falhas ao atualizar o cadastro de horários e monitoramento dos atendimentos prestados por esses profissionais.

Tem se tornado cada vez mais importante o foco das empresas em otimizar seus serviços e ganho de tempo, contando principalmente com resultado final, que é a satisfação de seus clientes. A facilidade e automação no atendimento ao público são boas para ambas as partes, servidor e cliente, pois otimiza tempo e é muito mais fácil de organizar, além de diminuir a probabilidade de equívocos na comunicação e agendamento. Além disso, é importante manter banco de dados funcionais que facilitem serviços administrativos e proporcionem segurança na manipulação de seus dados. Na interação com essa comunidade prestadora de serviços foi observado a necessidade da criação de um software eficiente que permita clientes solicitar horários de atendimento, controlar serviços prestados e, que ofereça autonomia aos usuários, permitindo assim, que os mesmos possam acessar o sistema para atualização de dados e agendamento de atendimentos.

Um framework é baseado em um conjunto de estruturas com ferramentas, bibliotecas e diretrizes com coerência em sua organização. Sua estrutura já é pré-estabelecida com objetivo de facilitar o desenvolvimento de softwares, aplicativos e sistemas em geral associada às ferramentas que facilitam o processo de criação com garantia na rapidez no desenvolvimento, uma vez que os programadores não precisam perder tempo com funções básicas, além de proporcionar maior produtividade, segurança, manutenção facilitada e redução de custos. Dessa

forma, trabalhar com essa estrutura de software com framework é bem vantajoso para otimizar o tempo para desenvolver e finalizar um programa, o que vem de encontro com esse projeto. O uso de controle de versão também agiliza o processo de programação por permitir que vários desenvolvedores trabalhem simultaneamente no mesmo projeto, tornando o processo mais rápido e eficiente. Dessa forma, o nosso grupo se propôs a desenvolver um software de acordo com o tema norteador da Univesp com framework web e controle de versão para agendamento de horários e serviços oferecidos por profissionais como cabeleireiros e barbeiros.

DESENVOLVIMENTO

2.1 OBJETIVOS

O objetivo geral desse projeto é o desenvolvimento de um software para agendamento de horários e serviços oferecidos por profissionais como cabelereiros e barbeiros visando a resolução de falhas e inconsistências ao agendar esse tipo de serviço.

Os objetivos específicos que permitem que o objetivo geral seja alcançado são:

- Analisar o cenário que é o ambiente de salões de cabelereiros e barbeiros;
- Desenvolver um software web usando ferramentas práticas;
- Utilizar a linguagem Java, no ambiente NetBeans para desenvolvimento do software web;

- Utilizar o banco de dados MySQL como Sistema Gerenciador de Banco de Dados (SGBD) para armazenar os dados do software;
- Fazer uso do JSF para oferecer uma interface amigável baseada em páginas HTML dinâmicas e interativas para o usuário
- Usar para controle de versão o GitHub, para criação de sincronismo

O software web deverá:

- Permitir o cadastro de atendimento de serviços prestados;
- Conceder acesso aos usuários do serviço;
- Possibilitar que os clientes realizem agendamento por serviços via Web;
- Apresentar protótipo;
- Realizar a testagem inicial da solução inicial (programa) junto a profissionais como cabelereiros e barbeiros para identificação de conflitos;
- Coletar sugestões de melhorias para o software;
- Implementar as alterações e adequar o sistema para alcançar melhorias das sugestões feitas pelos usuários;
- Entregar a solução final ao cliente.

2.2 JUSTIFICATIVA E DELIMITAÇÃO DO PROBLEMA

Profissionais que trabalham com a prestação de serviços e, que abrangem áreas como estética, beleza e, de cuidados com o bem-estar tem problemas em gerir corretamente seus serviços e os atendimentos que são prestados aos seus clientes, pois alguns atendimentos são realizados num contexto de informalidade, como agendamentos sem automação em anotações em papel e escrito à mão, os quais levam a falhas e equívocos como erros no controle ineficaz em datas e horários de atendimento, bem como de cancelamentos na agenda, onerando o tempo do cliente e do profissional que oferece o serviço. Portanto, desenvolver um sistema com framework web que seja capaz de gerir a agenda desses profissionais para sanar esses problemas se mostra de relevante importância, uma vez que o software gerado permitirá cadastrar, alterar e cancelar horários para fazer o gerenciamento dos atendimentos que serão oferecidos; e ainda oferecer comodidade e segurança aos usuários por permitir que os mesmos possam acessar o sistema de onde estiverem através do world wide web.

Para os desenvolvedores do sistema o maior benefício é a questão do controle de versão, uma vez que ao ser necessário realizar ajustes e melhorias no software para correção de possíveis erros, os desenvolvedores poderão fazer os seus registros, mantendo o histórico dos arquivos alterados e indicando quem fez as alterações nos arquivos garantindo agilidade e eficiência.

A pretensão deste projeto é a integração entre a teoria e a prática dos conteúdos apreendidos no decorrer do nosso curso, como, por exemplo, desenvolvimento de produtos, entendimento de algoritmos, programação e criação de banco de dados. Deste modo, criaremos um software web com as características demandadas pelo público-alvo, de acordo com os resultados de nossas pesquisas.

2.3 FUNDAMENTAÇÃO TEÓRICA

2.3.1 Banco de dados

O armazenamento de informações é uma preocupação desde os primórdios da humanidade. Ao longo do tempo, o principal meio de registro de informações tem sido o papel, porém esse tipo de registro tem alguns inconvenientes como a quantidade física de espaço necessário para o armazenamento de arquivos e a demanda de tempo para a localização das informações. Com a evolução da tecnologia e o aumento imensurável da quantidade de

informações, surgiu o arquivo nos computadores, um meio eficaz que se tornou respeitável, embora a maior preocupação não fosse somente como e onde armazenar as informações, mas também como consultá-las e acessá-las. Dessa forma, surgem os bancos de dados não somente como repositórios, mas também com estrutura bem elaborada de modelo relacional (Cardoso, V. M.; Cardoso, G.C., 2012, p. 15). Em nosso cotidiano, várias situações nos permite a interação com bancos de dados, como por exemplo: cadastrar e acessar algum aplicativo de banco no smartphone, tornando o banco de dados e os sistemas gerenciadores de bancos de dados (SGBDs) cada vez mais essenciais para armazenamento de informações. Praticidade, eficiência, rapidez e confiabilidade das informações são os principais benefícios que os bancos de dados computadorizados oferecem (Cardoso, V. M.; Cardoso, G. C., 2012, p.16).

O programa MySQL (<https://www.mysql.com/>) usado no desenvolvimento do software é um servidor robusto de bancos de dados SQL (Structured Query Language - Linguagem Estruturada para Pesquisas) e foi desenvolvido no ano de 1980 na Suécia por dois suecos e um finlandês; David Axmark, Allan Larsson e Michael "Monty" Widenius. A utilização de SQL destaca-se por ser muito rápido, multi-tarefa e multi-usuário e, gerou grande quantidade de SGBDs como o servidor MySQL (Machado, 2009, p. 244). A conexão com MySQL pode ser estabelecida usando o driver MySQL-Connector, que é uma implementação da interface JDBC (<https://dev.mysql.com/doc/connector-j/en/connector-j-usagenotes-connect-drivermanager.html>). Ele pode instalado e configurado no projeto por meio de ambiente de desenvolvimento Netbeans (<https://netbeans.apache.org/front/main/index.html>).

2.3.2 Frameworks

O desenvolvimento de softwares e aplicativos contam com ferramentas como Frameworks, que tem como base códigos prontos, que fornecem infraestrutura de projeto disponibilizada aos desenvolvedores, bem como fluxo de controle de aplicação. O framework é uma estrutura que atinge a máxima reutilização de todo ou parte de um sistema por conterem o protocolo de controle de aplicação (Fayad *et al.*, 1997). Outro ponto é o nível de granularidade representada por um conjunto de classes abstratas e concretas, capazes de interagir entre suas instâncias, podendo gerar aplicações a partir de combinações de instâncias das classes concretas do framework ou pela definição as classes (Telligent, 1994).

A Web é composta por servidores conectados entre si e que fornecem grande quantidade de informação por serviços. Os frameworks possuem a vantagem de ferramenta altamente adaptativa a vários contextos, e dentre ele a Web. A composição do serviço é que definirão a aplicação a ser gerada (Pinto, 2000, p. 7). Nesse projeto foi utilizado a framework web JSF

(<https://www.oracle.com/java/technologies/javaserverfaces.html>) com a linguagem Java EE no ambiente NetBeans e banco de dados MySQL.

2.3.3 Controle de versão Git

Controle de versão é um sistema que registra alterações em um arquivo ou conjunto de arquivos ao longo do tempo para que você possa lembrar versões específicas mais tarde (<https://git-scm.com/book/pt-br/v2/Come%3A7ando-Sobre-Controle-de-Vers%C3%A3o>).

Além disso, muitos desses sistemas trabalham muito bem com vários repositórios remotos, tal que você possa colaborar em diferentes grupos de pessoas de maneiras diferentes ao mesmo tempo dentro do mesmo projeto. Isso permite que você configure vários tipos de fluxos de trabalho que não são possíveis em sistemas centralizados, como modelos hierárquicos (<https://git-scm.com/book/pt-br/v2/Come%3A7ando-Sobre-Controle-de-Vers%C3%A3o>).

2.4 APLICAÇÃO DAS DISCIPLINAS ESTUDADAS NO PROJETO INTEGRADOR

2.4.0.1 Programação Orientada a Objetos

O conceito apresentado na disciplina de programação orientada a objetos (POO) foi fundamental para o entendimento e abordagem desse tipo de modelo de análise, projeto e programação de software, baseado na composição e interação entre várias unidades conhecidas como objetos. Esse tipo de programação permite reduzir os programas em partes e resulta no alcance de metas na projeção e softwares de forma mais fácil e eficiente, com facilidade de projetar o código e organização, que decorrem de padrões comuns (Deitel, Harvey, 2016). Precisa ser observado na escolha da linguagem de programação alguns aspectos como: tempo de execução de um programa, paradigma explorado, reuso de trabalho e, principalmente flexibilidade.

No desenvolvimento desse projeto foi escolhida a linguagem Java, a qual foi modelada em C++, desenvolvida pela Sun Microsystems e, atual Oracle. Essa linguagem foi projetada para ser pequena, simples e portável. Além disso, todos os códigos serão abordados e disponibilizados em ambiente Netbeans IDE, o qual é um ambiente integrado, gratuito, de código aberto para desenvolvedores de software de inúmeras linguagens de programação, dentre elas Java (Deitel, Harvey, 2016).

Java é uma das linguagens de programação mais utilizadas no mundo e seu objetivo-chave é ser capaz de escrever programas a serem executados em uma grande variedade de sistemas computacionais e dispositivos controlados por computador. Isso às vezes é chamado de “escreva uma vez, execute em qualquer lugar”. Java é utilizada para desenvolver aplicativos corporativos de grande porte, aprimorar a funcionalidade de servidores da web (os computadores que fornecem o conteúdo que vemos em nossos navegadores), fornecer aplicativos para dispositivos voltados ao consumo popular (por exemplo, telefones celulares, smartphones, televisão, set-up boxes etc.) e para muitos outros propósitos (Deitel, Harvey, 2016).

2.4.0.2 Formação Profissional em Computação

A formação profissional em computação é uma disciplina que prevê a resolução de problemas de forma informatizada e/ou automatizada em situações do mundo real. Além disso,

apresenta diretrizes para escolha de linguagens de programação e uso de frameworks em desenvolvimento de softwares. Um software é um produto desenvolvido por engenheiros de software que engloba programas, documentos que englobem a operação e o uso de programas e, estrutura de dados para manipulação do programa de forma adequada (Sommerville, 2011).

A infraestrutura básica para o desenvolvimento web em Java é composto por Servelets presentes em frameworks e APIs. Com intuito de simplificar o desenvolvimento de aplicações web e melhorar a arquitetura foi desenvolvida a framework JavaServer Faces (JSF). Em nosso projeto foi a usada a framework JSF que tem como foco a interface de usuário, com coleção de componentes bastante robusta e melhor separação das camadas de aplicação e oferece códigos reutilizáveis para construção de aplicações web (Deitel, Harvey, 2016).

2.4.0.3 Fundamentos da Web

Em Fundamentos da Web foi fundamentado conceitos como rede de computadores que permitem troca de dados entre aplicações e da organização da arquitetura em camadas (Kurose, Ross, 2013). Também foram abordados conceitos de protocolo de comunicação para definição de formato e ordem de mensagens entre duas ou mais entidades comunicantes, bem como de ações que acontecem durante a transmissão e recepção das mesmas. Entendimento de componentes da Web como servidores Web, protocolo HTTP, browsers, cliente e servidor também foram de muita relevância para o desenvolvimento desse projeto Integrador, que tem como tema norteador o desenvolvimento de um software web. Também foram importantes conceitos desenvolvidos como a construção de páginas com linguagens de marcação como a mais conhecida, que é chamada de HyperText Markup Language – HTML), além da aplicação de estilos com uso de Cascading Style Sheets (CSS).

2.4.0.4 Banco de Dados

Também foi nos disponibilizada aulas de Banco de Dados com intuito de aprendizagem e criação de modelos lógico e físico em banco da dados, através de várias ferramentas, especialmente o MySQL. Os conhecimentos que foram apresentados como normalização, Modelo-Entidade-Relacionamento e Modelo Relacional com uso e chaves primárias e estrangeiras, bem como de suas simbologias proporcionaram o conhecimento para operações que foram realizadas pelo sistema gerenciador de banco de dados (SGBD). SGBD é um sistema

de software que ajuda no processo de definição, construção, compartilhamento, bem como de sua manipulação entre usuários e aplicações (Elmasri, Navathe, 2018).

2.4.0.5 Algoritmos e Programação de Computadores I e II

Algoritmos de computador é um conjunto de etapas para executar uma tarefa descrita com precisão suficiente para que um computador possa executá-la (Cormen, 2012). As disciplinas de Algoritmos e Programação I e II proporcionaram o conhecimento para construção de programas com definição de algoritmos e sua representação por fluxogramas, uso de expressões aritméticas e lógicas e operadores, além de conhecimento em listas, tuplas, variáveis com uso de estruturas de repetição e de condição e de bibliotecas associadas às linguagens de programação. Também foi exemplificado a importância de documentação de programas e gerenciamento de alterações de códigos, o qual foi incorporado no projeto por meio da plataforma de hospedagem Github e seu respectivo controle de versão Git.

2.4.0.6 Estrutura de Dados

A disciplina de Estrutura de Dados abordou modelagem e implementação de diferentes estruturas de dados, além de algoritmos para gerenciá-los. Foi nos ensinado sintaxe básica de classes de operações e classes, uso de ponteiros, alocação dinâmica, tratamento de colisões, uso de árvores binárias, utilização de arranjos, implementação de pilhas e filas para garantirmos alinhamento de componentes e uso de recursos compartilhados em processos, respectivamente (Lorenzi, *et al.*, 2006).

Esses conhecimentos adquiridos nessa disciplina permitiu o desenvolvimento do sistema de forma eficiente e otimizado para lidar com dados.

2.4.0.7 Gestão de Inovação e Desenvolvimento de Produtos

Essa disciplina foi muito interessante e de grande relevância no desenvolvimento do projeto, porque abordou o conceito de Design Thinking com a metodologia Brainstorming. A disciplina também ajudou no entendimento do processo criativo pelo modelo proposto por Graham Walls, disposto em quatro pilares: Preparação (coleta de informações), Incubação (descanso mental), Iluminação (solução) e Verificação (ajuste e implementação). Foram aplicadas noções de Lean Startup, onde se tem um conjunto mínimo de recursos para resolver

o problema, obtendo como resultado a tração inicial do produto, ou seja, quando 40% ou mais dos clientes ficarem muito desapontados se o produto deixar de existir (Pearson Education do Brasil, 2011). Esses conceitos nos deram suporte para elaboração das entrevistas com a comunidade e nos capacitou a decidir por uma solução eficaz na resolução de seus problemas.

2.4.0.8 Leitura e Produção de Textos

A disciplina Leitura e Produção de Textos foi disponibilizada no início do nosso curso de graduação e favoreceu na elaboração dos relatórios com conceito e mecanismos de coesão textual e clareza na comunicação. A disciplina reafirmou a importância da leitura e escrita como ferramentas de formação cidadã e, estratégias eficientes para textos acadêmicos (Koch, 2010). Também dimensionou a diferença e importância da linguagem verbal e não-verbal, efetivando-se a linguagem como instrumento de comunicação (Porfirio, 2015). Todas essas abordagens foram fundamentais para interação com a comunidade durante o questionamento de seus problemas de forma clara e elegível para que pudéssemos construir o projeto Integrador e fazer inferências durante o desenvolvimento do mesmo para capacitar melhorias no sistema.

2.4.0.9 Projetos e Métodos para a Produção do Conhecimento

No curso de Projetos e Métodos para a Produção do Conhecimento obtivemos diretrizes sobre o senso comum resultante da relação homem e natureza como forma de ciência com conhecimento racional, metódico e sistemático. A ciência envolve o estudo crítico do método para conhecimento e análise da realidade social (Cervo, Bervian, Da Silva, 2006). Foram ensinados procedimentos científicos através de técnicas de pesquisa com documentação direta e indireta. Também foi discutido como examinar um trabalho com senso crítico com fidelidade ao pensamento do autor, facilitando em nossas buscas bibliográficas (Barros, Lehfeld, 2012).

As abordagens proporcionaram o entendimento do método que seria usado no processo de investigação da comunidade a ser abordada nesse projeto. Além disso, ajudou na postura científica para busca de soluções para o problema que foi abordado no desenvolvimento do software web.

2.4.1.0 Ética, Cidadania e Sociedade

Essa disciplina nos ensinou que a ética é um sistema de normas, princípios e valores, as quais regulamentam relações mútuas entre os indivíduos e, ao interiorizarmos essas normas construímos nossa formação ética (Sánchez Vásquez, 2018). A informação científica é o resultado de pesquisas registradas e divulgadas com atitudes éticas, que envolvem princípios éticos, consentimento, confiabilidade, anonimato e proteção dos dados (Barros, Lehfeld, 2012).

Esses ensinamentos nos levaram a desenvolver um cenário de como seria desenvolvido o estudo, qual público-alvo, como seria a abordagem da comunidade, demonstrar a confiabilidade em nosso trabalho e principalmente garantir a proteção dos dados das pessoas que se disponibilizaram em responder o formulário que direcionou o desenvolvimento do presente projeto.

2.4.1.1 Inglês

O conhecimento da língua inglesa é importante para os profissionais que desejam alavancar suas carreiras, porém em tecnologia tem se tornado primordial. É a linguagem padrão usada no desenvolvimento de códigos e padronizado em muitas linguagens. Na aula de inglês aprendemos a buscar referências e significados de palavras, bem como de termos usados em nossa área (<https://www.wordreference.com/>). Também foi abordado como participar de Congressos internacionais, disponibilizando conhecimentos desde a inscrição do evento, interação com demais participantes, bem como ter desenvoltura nesse tipo de evento.

A disciplina também proporcionou o entendimento de livros e artigos escritos em inglês, os quais foram referenciados nesse projeto. Além disso, o conhecimento do inglês será determinante para prática de trabalhos e futuros projetos que exijam internacionalização.

2.5 METODOLOGIA

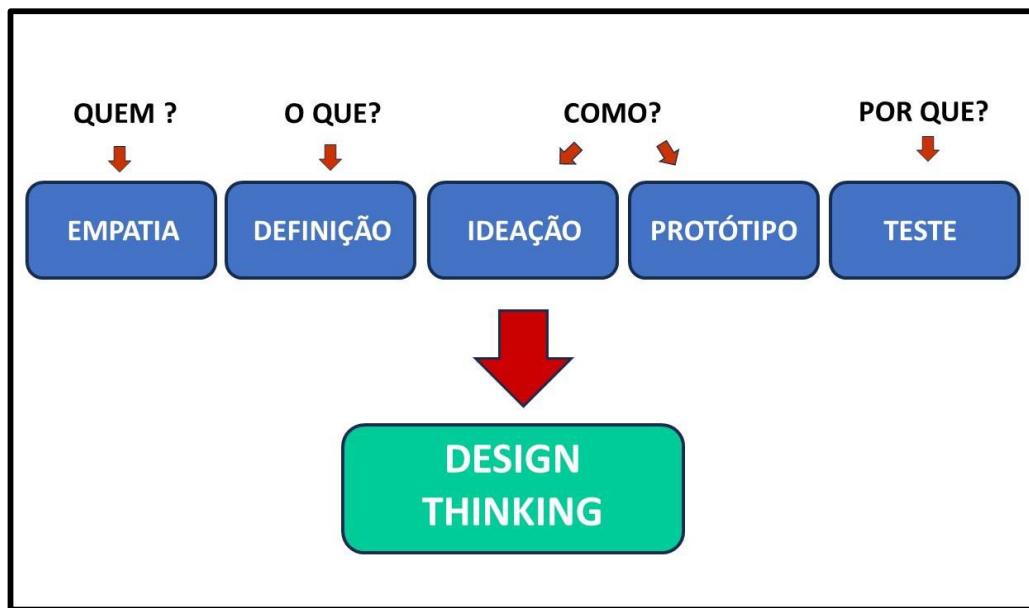
2.5.1 Interpretação do contexto e coleta de dados

Para chegarmos na definição da comunidade e qual tema seria abordado pelo grupo no desenvolvimento do Projeto Integrador abordamos a metodologia de Brainstorming com conceito de Desing Thinking. No Design Thinking é importante procurar por respostas para questionamentos como: Quem?, O que?, Como? e Por que? (Figura 1).

Essas questões devem ser abordadas em etapas do Desing Thinking: empatia, definição, ideação, protótipo e teste. A empatia busca por soluções que são intrinsecamente desejáveis

para satisfazer as necessidades explícitas ou latentes (Brown, 2010). Na etapa de definição eram realizadas reuniões semanais com os participantes do projeto através da ferramenta Collaborate (Univesp) para decidirmos quais comunidades seriam abordadas e o tema do trabalho de acordo com o tema norteador disponibilizado pela Univesp para o desenvolvimento de um software com framework web, fazendo uso de banco de dados e técnicas de controle de versão. Na etapa de ideação os dados foram coletados via pesquisa de campo *on-line*, o que tornou possível a decisão de desenvolver um software para prestadores de serviços, mais especificamente cabelereiros e barbeiros, os quais apresentaram como problema recorrente, falhas em sua forma de agendamento de horários. Na etapa seguinte, o grupo distribuiu tarefas entre seus membros, seguindo as normas apresentadas no plano de ação.

Figura 1 - Metodologia Design Thinking



Fonte: Elaborado pelos autores.

2.5.2 Protótipo do Projeto

O desenvolvimento do software abordou a linguagem Java com framework JSF, banco de dados MySQL e controle de versão, usando Git. A justificativa para uso dessas ferramentas se dá ao fato de que a linguagem Java apresenta portabilidade do sistema para o sistema operacional desejado, e também possui integração com a frameworks JSF e APIs que amplia a funcionalidade do sistema.

O sistema contará com a seguinte estrutura:

- Área administrativa para possíveis agendamentos, alterações ou reagendamentos;
- Área pública gerida pelo próprio cliente, onde pode solicitar um bloco de horários vagos para ser atendido;

2.5.3 Testagem do Protótipo

Nessa fase do Design Thinking, o foco não é desenvolver um protótipo que seja perfeito, mas que seja possível identificar pontos positivos e negativos e, correção dos mesmos (Brown, 2010).

O teste do protótipo do software será realizado em um primeiro momento pelos integrantes desse projeto, para que cada um descreva sua percepção de funcionamento bem como de erros que poderão ocorrer.

Em uma segunda etapa, o software desenvolvido será testado por proprietários de salões de cabeleireiros e barbeiros, onde será possível verificar as funcionalidades do programa, e permitir a realização de testes, a fim de que sejam identificadas possíveis falhas e levantadas sugestões e críticas por parte desses usuários.

2.6 RESULTADOS PRELIMINARES: SOLUÇÃO INICIAL

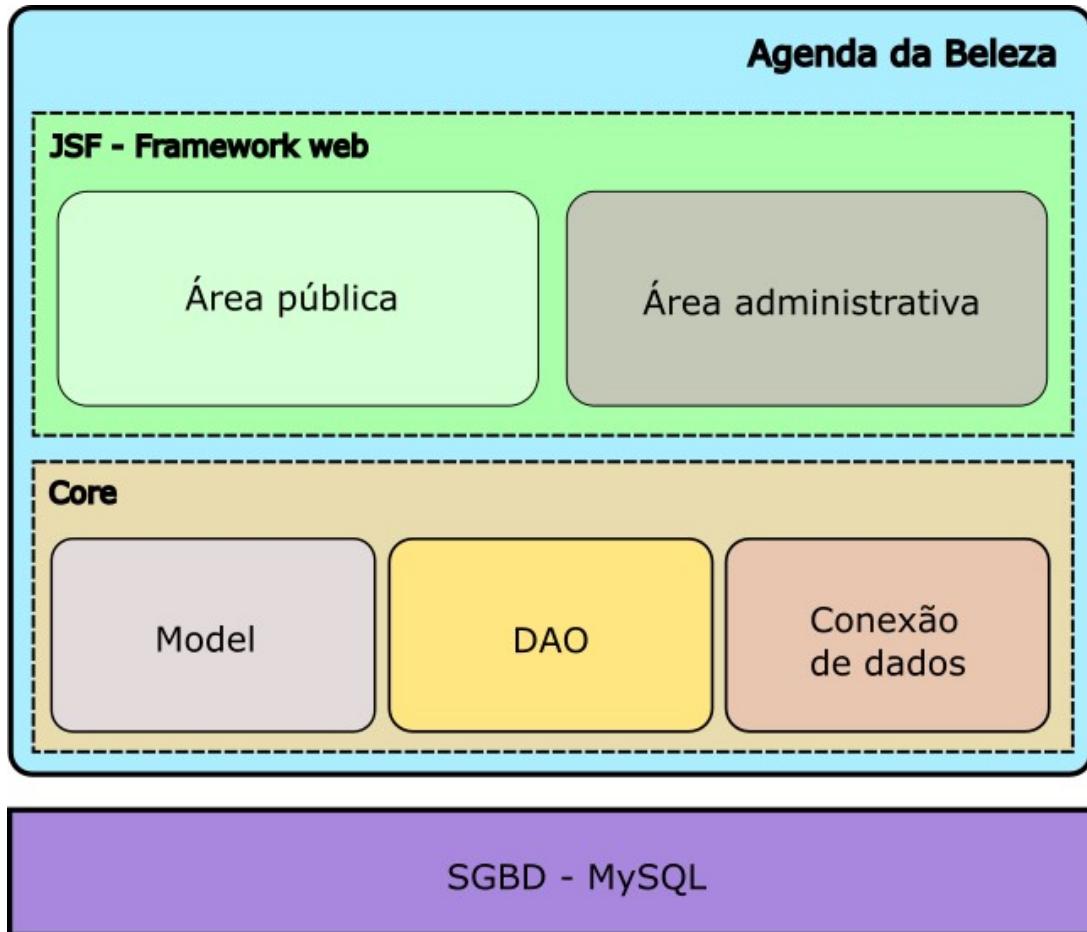
Os resultados gerados pelo projeto descrevem como foi elaborada a solução inicial que visava como objetivo resolver o problema de agendamentos de horários que prestadores de serviços tais como cabeleireiros e barbeiros sofrem pela falta de automação desse processo via Web.

O software Agenda da Beleza é uma aplicação web feita em Java e com o framework web JSF. Para persistência dos seus dados, é utilizado o Sistema Gerenciador de Banco de Dados (SGBD) MySQL. Seu modelo arquitetural pode ser visualizado na figura 2. Numa visão bottom-up, o primeiro elemento é o SGBD. Sua função é armazenar os dados por meio de recursos de otimização e segurança tais como tabelas, índices, autenticação e autorização de usuários, backup, restore entre outros. Em sequência, o bloco grande em azul claro é a aplicação em si. Ela é subdividida em duas partes principais: 1) core responsável pela comunicação e gerenciamento dos dados no SGBD e 2) JSF - Framework web que recebe e trata as requisições do usuário e devolve os resultados das requisições em páginas HTML.

O core é subdividido em três partes: Model, DAO e Conexão de dados. O Model é constituído pelas representações das entidades da base de dados, para elucidar, pode ser pensado como uma representação do Modelo Entidade Relacionamento em nível mais alto. O DAO é responsável por traduzir e gerenciar a persistência dos dados, intermediando entre a sintaxe da aplicação e a do banco de dados. A conexão com o banco de dados é o mecanismo que estabelece uma via segura para comunicação com o SGBD, por meio de autenticação via usuário e senha.

O bloco JSF - Framework web, nesta aplicação, é composto por dois blocos menores, área pública e área administrativa. A área pública é o local da aplicação que pode ser acessado por qualquer usuário que tenha uma conexão com a internet. Nessa área ele pode visualizar o estado da agenda e também alocar um horário de acordo com seu interesse. A área administrativa é o espaço que só pode ser acessado pelo usuário responsável por fazer trabalhos de administração do sistema tais como, parametrização do sistema, inclusão, alteração e exclusão de registros de dados, entre outros.

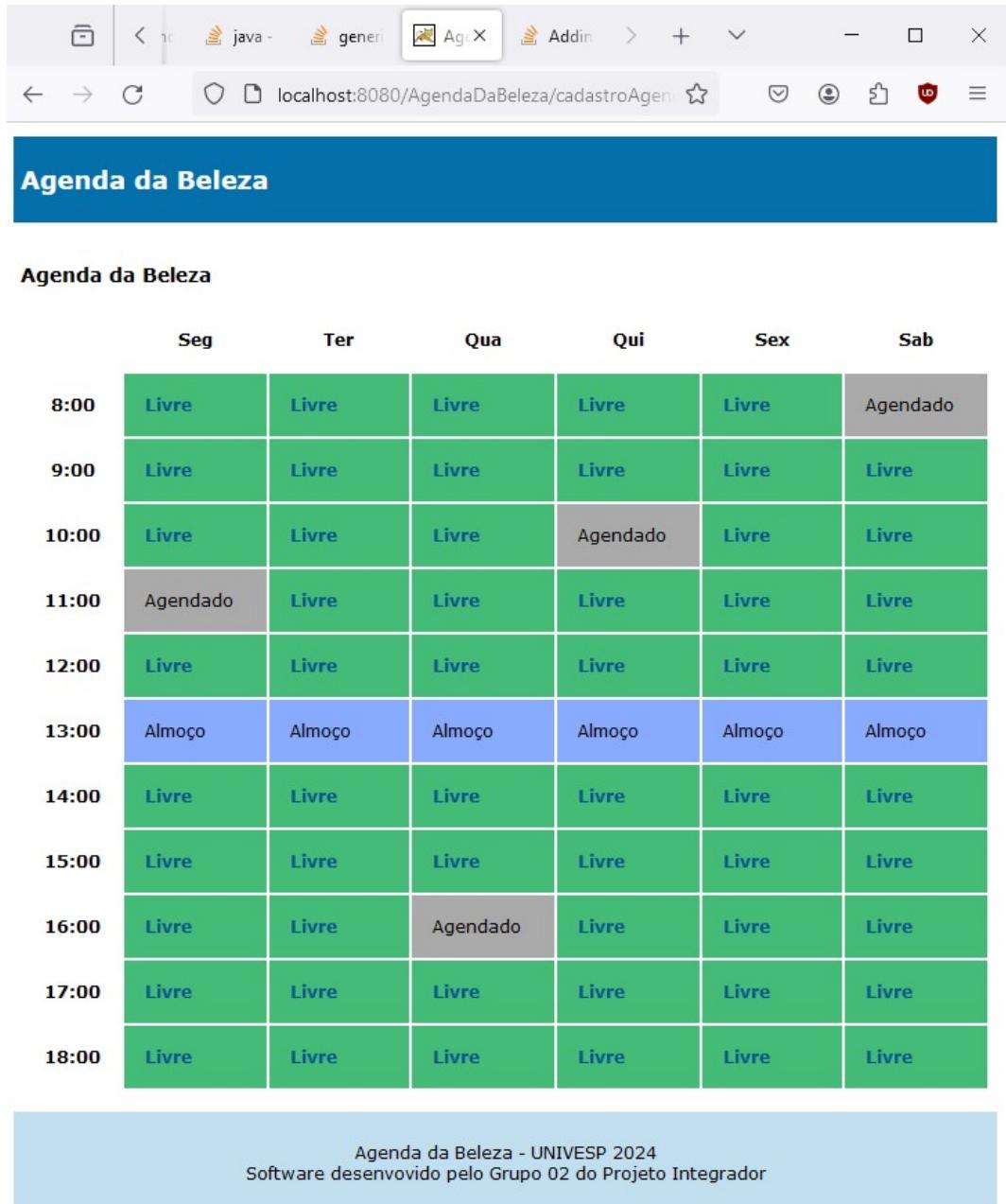
Figura 2 - Modelo Arquitetural do software Agenda da Beleza



Fonte: Elaborado pelos autores.

A figura 3 é uma pré-visualização da tela inicial da área pública. Percebe-se um conjunto de blocos para alocação de horários disponibilizado em forma de tabela. Os que estão em cor verde e com o texto “Livre”, são os blocos disponíveis para agendamento. Os que estão em cor cinza e com texto “Agendado” são os blocos dos horários já reservados e que não podem mais ser alocados. Em lilás, um conjunto de blocos com texto “Almoço”, são os blocos do intervalo de almoço e não podem também ser alocados. As colunas da tabela são os dias da semana, de segunda a sábado e as linhas são os horários para atendimento, das 8:00 às 18:00.

Figura 3 - Pré-visualização da Área Pública do software Agenda da Beleza



The screenshot shows a web browser window with the title 'Agenda da Beleza' in the tab. The URL is 'localhost:8080/AgendaDaBeleza/cadastroAgen'. The main content is a 10x6 grid representing a weekly availability calendar. The columns are labeled 'Seg', 'Ter', 'Qua', 'Qui', 'Sex', and 'Sab'. The rows represent time slots from 8:00 to 18:00. The grid contains the following data:

	Seg	Ter	Qua	Qui	Sex	Sab
8:00	Livre	Livre	Livre	Livre	Livre	Agendado
9:00	Livre	Livre	Livre	Livre	Livre	Livre
10:00	Livre	Livre	Livre	Agendado	Livre	Livre
11:00	Agendado	Livre	Livre	Livre	Livre	Livre
12:00	Livre	Livre	Livre	Livre	Livre	Livre
13:00	Almoço	Almoço	Almoço	Almoço	Almoço	Almoço
14:00	Livre	Livre	Livre	Livre	Livre	Livre
15:00	Livre	Livre	Livre	Livre	Livre	Livre
16:00	Livre	Livre	Agendado	Livre	Livre	Livre
17:00	Livre	Livre	Livre	Livre	Livre	Livre
18:00	Livre	Livre	Livre	Livre	Livre	Livre

At the bottom, a footer bar displays the text 'Agenda da Beleza - UNIVESP 2024' and 'Software desenvolvido pelo Grupo 02 do Projeto Integrador'.

Fonte: Elaborado pelos autores.

Através dessa solução inicial foram realizados testes necessários no software e relatadas falhas que ocorreram. Dessa forma, na etapa seguinte será realizada a implementação e alterações para adequar o sistema com objetivo de melhoramento na funcionalidade do software.

REFERÊNCIAS

APACHE NETBEANS 21. **Development Environment, Tooling Platform and Application** <<https://netbeans.apache.org/front/main/index.html>> Acesso em 02/04/2024.

ABNT – Associação Brasileira de Normas Técnicas. **NBR 14724: Informação e documentação. Trabalhos Acadêmicos** - Apresentação. Rio de Janeiro: ABNT, 2023.

BARROS, A. J., LEHFELD, N. A. S. **Metodologia científica**. 3 ed São Paulo: Pearson Prentice Hall, 2012.

BROWN, T. **Uma metodologia poderosa para decretar o fim das velhas ideias design thinking**. Rio de Janeiro: Elsevier, 2010.

CARDOSO, V. M; CARDOSO, G. C. **Sistema de Banco de Dados**. 1 ed. São Paulo: Saraiva, 2012.

CERVO, A. L., BERVIAN, P. A., DA SILVA, R. **Metodologia Científica**. 1 ed. São Paulo: Pearson Prentice Hall, 2011.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. **Algoritmos - Teoria e Prática**., 3 ed. São Paulo: GEN LTC, 2012.

DEITEL, P.; DEITEL, H. **Java Como Programar**. 10 ed. Londres: Pearson, 2016.

ESMASRI, R.; NAVATHE, S. K. **Sistemas de Banco de Dados**. 7 ed. Londres: Pearson, 2018.

FAYAD, M.; Douglas C. S. **Object-Oriented Application Frameworks - Introduction**. Communications Of the ACM, 40: 32-38, October, 1997.

GIT. **Git-Local-Branching-on-the-cheap**. Book- Chapter 2 ed <<https://git-scm.com/book/pt-br/v2/Come%7ando-Sobre-Controle-de-Vers%C3%A3o>> Acesso em 19/03/2024.

IBGE - INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **PAS – Pesquisa Anual de Serviços**. <<https://www.ibge.gov.br>> Acesso em 19/03/2024.

KOCH, I. V. **A coesão textual**. 22 ed. São Paulo: Contexto, 2010.

KUROSE, J.; Ross, K. **Rede de computadores e a Internet: Uma Abordagem Top-Down**. 6 ed. Londres: Pearson, 2013.

MACHADO, F. N. R. **Banco de Dados – Projeto e Implementação**. 2 e d. São Paulo: Érika, 2009.

LORENZI, F. et al. **Estruturas De Dados**. 1 ed. São Paulo: Elsevier, 2006.

MYSQL. **Downloads, Documentation and Developer Zone** <<https://www.mysql.com/>> Acesso em 02/04/2024.

MYSQL. **JDBC Driver Manager** <<https://dev.mysql.com/doc/connector-j/en/connector-j-usagenotes-connect-drivermanager.html>> Acesso em 02/04/2024.

PEARSON EDUCATION DO BRASIL. **Criatividade e inovação**. 1 ed. São Paulo: Pearson Prentice Hall, 2011.

PINTO, S. C. C. S. **Composição em WebFrameworks**. 2000. Tese (Doutorado em Informática) - Pontifícia Universidade Católica do Rio de Janeiro.

PORFIRIO, S.; SOUZA, F. E. B.; CIPRIANO, L. C. **Textos multimodais: a nova tendência na comunicação**. Observatório da Imprensa, Edição 861, 29 jul. 2015. Disponível em: <<https://www.observatoriodaimprensa.com.br/diretorio-academico/textos-multimodais-a-nova-tendencia-na-comunicacao/>> Acesso em 20/03/2024.

SÁNCHEZ-VÁSQUEZ, A. **Ética**. 39 ed. Civilização Brasileira: São Paulo, 2018.

SOMMERVILLE, I. **Engenharia de Software**. 9 ed. Prentice Hall: São Paulo, 2011.

TALIGENT. **Building object-oriented frameworks**. [S.I]: Taligent, 1994. White paper. Disponível em: <<https://developer.ibm.com/>> Acesso em 20/03/2024.

WORDREFERENCE <<https://www.wordreference.com/>> Acesso em 02/04/2024.

ANEXOS

Modelo de questionário respondido por um dos voluntários (José Francisco Catapani):

1. Como você atualmente gerencia os agendamentos de horários em seu salão de cabeleireiro? Existem desafios específicos que você enfrenta nesse processo?

R: Anotando em uma caderneta com nome do cliente e o horário que ele agendou. Desafio é que todo dia tem que escrever os horários disponíveis e ir anotando quando um cliente agenda um horário, quando cliente pergunta sobre horário disponível tem que consultar a agenda pra saber quais horários estão livres para oferecer ao cliente.

2. Quais seriam as principais necessidades que você identifica ao agendar horários para os clientes? Isso inclui considerações como gestão de tempo, organização de agendas e comunicação eficiente.

R: facilidade para o cliente agendar horário, organização dos horários disponíveis, sugestão de horário disponível, avisar o cliente no dia agendado com pelo menos uma hora de antecedência, pois caso ele não possa comparecer ele tenha também a opção de cancelar o agendamento, abrindo assim vaga para um futuro cliente que deseje agendar próximo aquele horário, possibilitar ao cliente a escolha prévia do(s) serviço(s) a ser executado, possibilitando a organização dos horários com base nos serviços escolhidos pelo cliente.

Ter um pré cadastro do cliente com dados simplificados como nome, celular (whatsapp) e um histórico dos 2 últimos serviços feito na barbearia.

3. Existem situações em que você sente que a falta de um sistema automatizado afeta a eficiência do salão?

R: sim, principalmente no horário de abrir e fechar e intervalo de almoço, pois sem agendamento, fica a disponibilidade do cliente chegar e aguardar na fila de espera, fazendo com que muita das vezes passe do horário de almoço ou de fechar a barbearia. Além da anotação da quantidade de atendimentos feito no dia e do fechamento do caixa e a gestão dos clientes que estiveram presentes naquele dia.

4. Na sua experiência, quais características específicas um software de agendamento de horários deveria ter para atender às necessidades do seu salão?

R: cadastro simplificado de cliente, com nome, celular e o sistema guardar a informação dos dois últimos agendamentos do cliente, informações básicas como corte realizado e a data que o cliente veio ao salão.

Cadastro de serviços com opção de valor e configuração de horário de execução de cada serviço para facilitar o agendamento.

Fechamento de caixa, com configuração de pagamento, se foi dinheiro ou cartão, dar essa opção ao cliente na hora do agendamento (com possibilidade de configurar margens de desconto da máquina de cartão), quantidade de clientes atendidos no dia.

Identificação do cliente pelo número do celular, assim toda vez q ele tentar agendar um novo serviço o login seja o celular, não necessitanto o cliente salvar ou gravar senhas para futuros agendamentos.

5. Como você gostaria que um sistema de agendamento ajudasse na comunicação com os clientes, por exemplo, enviando lembretes automáticos ou confirmações de horários?

R: ajudaria se o software enviasse um lembrete ao cliente no dia agendado com pelo menos uma hora de antecedência, possibilitando ao cliente a confirmação ou o cancelamento do agendamento abrindo assim vaga para um outro cliente.

6. Como você lidaria com o gerenciamento de cancelamentos, reagendamentos? Tem em mente alguma ideia de processo eficiente?

R: o software deveria enviar ao cliente um lembrete no dia do agendamento, com pelo menos uma hora de antecedência, dando a opção do cliente confirmar se poderá comparecer no horário agendado ou não, ou caso necessário, ter a opção de reagendar o horário, e ainda se fosse possível, com a confirmação do agendamento uma hora antes, com 20min antes do atendimento enviar um lembrete do agendamento na barbearia.

7. Além do agendamento de horários, quais outras funcionalidades você gostaria que um software oferecesse para melhorar a gestão geral do seu salão? Algo específico que você sente falta nas soluções disponíveis atualmente no mercado?

R: gestão de clientes, fluxo de caixa, cadastro simplificado de clientes, agendamento por QR code ou endereço virtual simplificado de fácil acesso ao cliente, cadastro de serviços com opção de tempo de execução e produtos, horário de atendimento personalizados.