



AUTOMAÇÃO DE SISTEMA DE KARAOKÊ E RESTAURANTE

ADRIANO THOMAZ A. F. DE SOUZA, FELIPE A. PITA, MURILO K. P. OKADO

THIAGO SCHUMACHER BARCELOS

IFSP - CAMPUS GUARULHOS

Resumo

A automação de sistemas torna-se cada vez mais presente para executar tarefas complexas dentro das empresas. Mesmo com suas vantagens, empecilhos como custos elevados, pouca possibilidade de customização e falta de capacitação podem resultar na não implementação deste processo ou, se implementado incorretamente, em prejuízos financeiros. Esse trabalho tem como objetivo desenvolver um sistema para controle dos pedidos, reservas e tempo de utilização das salas em um karaokê no bairro da Liberdade, São Paulo. A pesquisa foi realizada utilizando métodos ágeis, com foco em Scrum e Extreme Programming, visando o desenvolvimento gradual do software por meio de sprints. Inicialmente, foi feita uma coleta piloto utilizando inteligências artificiais para simular uma entrevista com os donos do estabelecimento, para que, em seguida fosse realizada a entrevista factual para o levantamento de requisitos. Com as entrevistas concluídas, foram definidas user stories responsáveis por orientar os requisitos do sistema e o desenvolvimento dos protótipos da interface.

Os resultados obtidos até o momento demonstraram que a interação presencial com os clientes possibilitou identificar necessidades que não haviam sido previstas inicialmente, como a automação da comunicação entre diferentes setores do estabelecimento. Além disso, as user stories contribuíram para maior alinhamento entre as funcionalidades planejadas e as demandas reais do cliente.

Conclui-se que o processo de levantamento de requisitos possui papel fundamental no desenvolvimento de sistemas, especialmente em projetos baseados em métodos ágeis. Espera-se que a versão inicial do software no site Figma ajude a melhorar a organização, o controle operacional e a experiência dos usuários, norteando também o design do software real.

Palavras-chave: Automação. Desenvolvimento web. Karaokê. Métodos ágeis.



1. Introdução

A automação, caracterizada como o processo de execução de tarefas manuais por métodos automáticos e tecnológicos, utilizando cada vez menos a mão de obra humana, vem se tornando cada vez mais necessária no ramo empresarial. Ela é observada em diferentes áreas, como a administrativa, com o envio de e-mails e tratamento de currículos; a industrial, com a introdução de máquinas para executar tarefas complexas; e a tecnológica, com a criação de sistemas que agilizam processos (IBM, [s.d.]).

Assim como outros tipos de automação, a automação de sistemas toma cada vez mais espaço no gerenciamento de estabelecimentos comerciais por garantir melhor controle dos fluxos de clientes, reservas e pedidos. Restaurantes, lanchonetes e bares, por exemplo, lidam com constante demanda de pedidos. Quando é realizado manualmente, por garçons e caixas, o processo fica mais suscetível a erros de registro, estressando o ambiente de trabalho e dificultando o controle do fluxo de pedidos e clientes.

Mesmo assim, os altos custos, pouca possibilidade de customização, interface não atrativa para o usuário e falta de capacitação para a implementação destes sistemas fazem com que pequenas e médias empresas ainda optem por continuar com anotações físicas, utilizando papel e caneta, o que pode resultar em prejuízos e desorganização generalizada dentro do estabelecimento.

No tocante a restaurantes com outros serviços no mesmo local, como o karaokê, essa limitação torna-se ainda mais evidente, pois, além dos processos comuns, ainda é necessário controlar agendamentos de salas, tempo de uso, separação das músicas solicitadas previamente e fluidez no atendimento, mantendo-o eficiente. A falta de um sistema personalizado impacta negativamente na experiência dos clientes, dos funcionários e na gestão interna do estabelecimento.

Logo, há necessidade de desenvolver um sistema próprio para auxiliar na gestão de um restaurante e karaokê, facilitando o controle de pedidos, reservas e tempo de utilização das salas, por meio de uma interface simples e adaptada ao ambiente. Pretende-se, com este trabalho, criar este sistema para um estabelecimento localizado no bairro Liberdade (SP), na Avenida da Liberdade, a fim de solucionar o problema de desorganização e melhorar a gestão do local com o menor custo de produção possível.



2. Objetivos

Desenvolver um sistema personalizado para um restaurante e karaokê situado na cidade de São Paulo, baseando-se na metodologia ágil e a partir de levantamento de requisitos, mapeamento de processos e análise de campo, com o intuito de automatizar os processos de lançamento de despesas, reservas e demais atividades exercidas no estabelecimento, além da própria metodologia empregada servir como parte do estudo para verificar sua viabilidade na aplicação em projetos reais de desenvolvimento de software.

2.1 Objetivos específicos

- Identificar os requisitos do sistema por meio de entrevistas e análise presencial do ambiente de trabalho;
- Manter contato frequente com o cliente, a fim de verificar a assertividade do que foi produzido nas sprints anteriores e alinhar o que será feito nas próximas;
- Elaborar protótipos antes de iniciar o desenvolvimento real do projeto, utilizando ferramentas online ou programação web;
- Implementar o sistema pouco a pouco, utilizando pair programming sempre que possível;
- Testar e validar cada resultado no final das sprints.

3. Fundamentação teórica

A engenharia de software baseia-se em processos estruturados com a finalidade de orientar a construção de sistemas confiáveis e adequados às necessidades dos usuários. No livro Engenharia de Software (SOMMERVILLE, 2018), o autor define que esses processos não se resumem a programação, mas envolvem um conjunto de atividades com a intenção de compreender um problema, e utilizar sua solução para produzir um produto. Assim, o desenvolvimento de software depende do planejamento, análise e definição de métodos.

Sommerville entende que é necessário um processo que organize o trabalho para possibilitar o desenvolvimento de um sistema. A estrutura de organização varia



conforme as necessidades, mas deve facilitar previsibilidade, comunicação clara entre os envolvidos e formas de verificar a qualidade.

Assim, torna-se clara a necessidade de definição e gerenciamento das necessidades dos usuários, constituindo a base sobre qual as futuras decisões do projeto são tomadas. Além de uma descrição do que o sistema deve fazer, é também o registro de restrições operacionais e expectativas que precisam ser atendidas. Por isso, a engenharia de requisitos envolve investigação, análise e validação - atividades que exigem comunicação com os administradores do ambiente em que o sistema será aplicado.

Outro aspecto importante a ser mencionado são as naturais mudanças nos requisitos. Conforme o problema é cada vez mais compreendido e o sistema toma forma, ou com o surgimento de novas necessidades, ajustes tornam-se inevitáveis. Assim, a gestão dessas mudanças deve ser incorporada ao desenvolvimento, para garantir que o software permaneça alinhado ao objetivo do projeto e às práticas do estabelecimento.

Com esses fundamentos, a engenharia de software fornece princípios que orientam a construção de sistemas mais adequados, eficientes e alinhados às demandas do usuário. Essa base teórica sustenta o desenvolvimento de ferramentas capazes de melhorar processos internos, aumentar o controle operacional e oferecer suporte às atividades do karaokê.

Neste contexto de constantes mudanças, metodologias como o modelo de cascata (SOMMERVILLE, 2018, p.34) se tornam inadequadas no quesito de acompanhamento destas tarefas, uma vez que, devido à estrutura fixa de planejamento do modelo, onde cada etapa deve ser feita e apenas é mostrado ao cliente no final, qualquer mudança requerida exige retrabalho e possui um custo de erro muito alto. Portanto, se vê necessário um desenvolvimento mais flexível, como os métodos ágeis. O Manifesto Ágil, descrito e explorado por Daniel Wildt et al. (2015), é uma declaração de princípios que propõe uma filosofia baseada em colaboração, ciclos curtos de programação que variam de 3 a 6 semanas (sprints) e capacidade de resposta rápida às alterações.

Então, a partir da metodologia ágil serão utilizados sprints de 3 semanas onde, no período entre as sprints, o cliente será contatado para mostrar o resultado do solicitado na sprint anterior e alinhar as demandas seguintes. Além disso, o grupo que produz o trabalho também será objeto de análise para verificar a usabilidade dos métodos ágeis numa situação real de desenvolvimento de software.



4. Materiais e Métodos

Durante o planejamento deste projeto, optou-se pela adoção de métodos ágeis, com foco na combinação entre Scrum e Extreme Programming, para o futuro desenvolvimento. Além da flexibilidade, esta metodologia favorece reflexões contínuas sobre o próprio processo de aprendizagem, objetivo adicional deste trabalho.

Assim, a futura fase de codificação ocorrerá de forma iterativa e incremental, dividido em ciclos curtos de entrega (sprints), nos quais a equipe realizará planejamento, implementação, testes, revisão com o cliente e retrospectiva do trabalho realizado.

4.1 Definição de requisitos

Esta é a etapa inicial, onde, por meio de comunicação com os usuários (neste caso, os donos do estabelecimento) a partir de entrevistas e questionários, e posteriormente mapeamento de processos, estabelece-se às necessidades específicas do sistema que deverão ser implementadas nas próximas etapas.

4.1.1 Entrevistas e preparo prévio utilizando IAs

A coleta de dados efetiva fundamenta-se em entrevistas diretas com o proprietário e funcionários, visando compreender o fluxo real de trabalho e as necessidades do estabelecimento. Como etapa de preparo para estas entrevistas, foi realizada uma coleta piloto utilizando ferramentas de inteligência artificial (ChatGPT, Claude e Gemini). Através de prompts específicos, as IAs simularam o comportamento de funcionários, prevendo possíveis dúvidas e estimando o tempo de coleta, aprimorando o roteiro antes da aplicação prática no local.

4.1.2 Histórias de usuários (user stories)

Para a tradução das necessidades do cliente em requisitos técnicos, será utilizada a técnica de “User Stories”. Cada funcionalidade do sistema é descrita sob a ótica do usuário final, seguindo o padrão: "Como [perfil], eu desejo [ação] para [benefício]". Essa abordagem permite que a equipe priorize o desenvolvimento com base nos interesses do karaokê e na experiência do usuário, garantindo que o software de automação seja funcional e intuitivo.

4.2 Projeto do sistema e do software

Após conversar com os usuários, será necessário trazer as ideias desenvolvidas para uma representação gráfica e visual, para avaliação dos que usufruirão



do produto final, apresentando-lhes modelos da interface do software criados com base nas user stories desenvolvidas na etapa anterior, podendo estes serem feitos em plataformas online como Figma ou Canva. Atualmente, o projeto encontra-se exatamente nesta fase.

4.3 Desenvolvimento colaborativo (pair programming)

A etapa de implementação será guiada pelas práticas de Extreme Programming, garantindo qualidade técnica e aprendizado contínuo. Cada funcionalidade relevante será desenvolvida por dois membros da equipe simultaneamente, favorecendo colaboração, troca de conhecimento e revisão contínua do código.

4.4 Integração e teste de sistema

Ao final de cada sprint, a equipe entregará um incremento funcional do sistema, que deverá integrar as partes desenvolvidas anteriormente, atender aos critérios definidos previamente e ser demonstrado ao cliente.

4.5 Operação e manutenção

Após a finalização de todo o processo de atendimento de todos os requisitos do cliente, sua respectiva aprovação e instalação do sistema, há possibilidade de surgimento de erros, que deverão ser corrigidos enquanto este já encontra-se sendo utilizado. Além disso, descobrir pontos que podem ser aperfeiçoados ou refeitos também ocorre nesta fase.

4.6 Tecnologias de desenvolvimento

Para o desenvolvimento do sistema, serão utilizadas tecnologias relevantes ao desenvolvimento web, permitindo a criação de um produto funcional, simples e com baixo custo, sendo elas HTML, CSS, JavaScript, PHP e MySQL.

O PHP foi escolhido como linguagem de programação para o lado do servidor (server-side) devido à sua compatibilidade com servidores web e facilidade da integração do banco de dados (PHP, [s.d.]). Por ser uma tecnologia de código aberto e amplamente documentada, ela possibilita a criação do sistema sem a necessidade de altos investimentos em licença de software.

O MySQL foi escolhido para o gerenciamento dos dados devido à sua eficiência no modelo relacional, organizando as informações em tabelas permitindo o



estabelecimento de relações entre elas (MYSQL, [s.d.]). Além disso, a combinação de PHP e MySQL é consolidada no mercado para aplicações que exigem agilidade e segurança no armazenamento de registros.

Com as tecnologias apresentadas acima, o desenvolvimento de um software web é possibilitado, uma vez que se complementam e permitem a criação de um projeto que cumpre com as necessidades projetadas pelo grupo.

5. Resultados e Discussão

A coleta piloto de dados, realizada com o suporte de inteligências artificiais (ChatGPT, Claude e Gemini), permitiu testar o roteiro de perguntas e prever a dinâmica das entrevistas. Embora úteis para o planejamento inicial, as respostas das IAs divergiram da realidade prática do estabelecimento, com respostas muito exageradas ou com pouco nexos. Na coleta efetiva, realizada presencialmente no karaokê, a interação direta com a proprietária revelou necessidades que não prevíamos, como a de uma comunicação automatizada entre o balcão e a cozinha para substituir as anotações manuais.

Com base nesses dados coletados na entrevista, foram estruturadas as user stories que servirão de guia para o desenvolvimento do software, como por exemplo:

- “Como um cliente eu gostaria de reservar uma sala de karaokê previamente para que eu possa utilizá-la quando chegar no estabelecimento”;
- “Como um funcionário eu devo alocar uma sala de karaokê para um cliente que reservou previamente ou solicitou a reserva na hora”.

6. Considerações Finais

Com base nos resultados obtidos até o momento, conclui-se que a etapa das coletas de dados possui papel fundamental no desenvolvimento do sistema, haja vista que o núcleo dos métodos ágeis é a comunicação constante a fim de verificar a qualidade e efetividade das interações futuras. As entrevistas e a observação presencial mostraram-se mais eficazes que simulações, fornecendo ainda mais informações precisas sobre a rotina do estabelecimento.

Atualmente, o projeto encontra-se com os requisitos iniciais previamente definidos por meio das user stories e a estrutura do sistema em desenvolvimento por meio



da plataforma Figma. As próximas etapas envolvem o desenvolvimento dos recursos de reserva, comandas e comunicação interna entre setores, seguidos de testes e validação com o cliente.

Espera-se que uma versão inicial do sistema seja concluída nos próximos ciclos de desenvolvimento, permitindo futura checagem com o proprietário do estabelecimento e a realização de ajustes conforme o uso real.

7. Referências

IBM. **O que é automação?** Disponível em: <https://www.ibm.com/br-pt/think/topics/automation>. Acesso em: 16 nov. 2025.

MYSQL. **MySQL Documentation.** Disponível em: <https://dev.mysql.com/doc/>. Acesso em: 21 abr. 2026.

PHP. **PHP: Hypertext Preprocessor.** Disponível em: <https://www.php.net/docs.php>. Acesso em: 21 abr. 2026.

SOMMERVILLE, Ian. **Engenharia de software.** 10. ed. São Paulo, SP: Pearson, 2018. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 16 nov. 2025.

WILDT, Daniel *et al.* **Extreme programming:** práticas para o dia a dia no desenvolvimento ágil de software. São Paulo, SP: Casa do Código, 2015. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 28 nov. 2025.