

SAFE LOGGER: SISTEMA DE APONTAMENTO DE HORAS COM GEOLOCALIZAÇÃO

37

SAFE LOGGER: TIMEKEEPING SYSTEM WITH GEOLOCATION

Cristiano Andrade de Aguiar¹, Danilo Gerardi de Faria¹, Marcio Rodrigues Sabino²,
Rita de Cássia Catini de Macedo³, Paulo César de Macedo⁴

1- Tecnólogos em Análise e Desenvolvimento de Sistema pela Faculdade de Tecnologia de São Paulo (FATEC – Unidade Mogi Mirim); 2- Mestre em Matemática Aplicada. Docente da FATEC - Mogi Mirim e da Fundação Hermínio Ometto (FHO – Araras); 3- Mestra em Ciências da Computação. Docente da FATEC - Mogi Mirim; 4- Doutor em Engenharia Biomédica. Docente da FATEC - Mogi Mirim e Faculdade Santa Lúcia - Mogi Mirim.

Contato: paulo.macedo@fatec.sp.gov.br

RESUMO

Atualmente, muitos colaboradores em diversas empresas registram suas horas trabalhadas em papel, um formato que exige grande cautela por parte dos responsáveis pelos registros e demanda considerável tempo do setor administrativo para a contagem das horas. Este cenário justifica a necessidade de um sistema digital que garanta a segurança das informações ao longo do tempo. O objetivo deste trabalho foi desenvolver um sistema de apontamento de horas (time-tracking) para os colaboradores de instituições ou empresas. Para alcançar esse objetivo, foi criado um microserviço que recebe requisições de uma aplicação web cliente e as armazena em um banco de dados não relacional. O diferencial deste projeto reside na utilização de geolocalização para a verificação da posição do usuário, permitindo o registro de horas apenas quando o colaborador estiver dentro de uma área permitida pela empresa, ou recusando-o com uma mensagem de alerta caso contrário. Esta solução tecnológica não apenas se mostra viável economicamente, mas também proporciona segurança para os registros e facilita o processo tanto para os usuários quanto para os funcionários administrativos.

Palavras-chave: registro eletrônico. microserviço. aplicação web. geolocalização.

ABSTRACT

Currently, many employees in many companies record their hours worked on paper, a format that requires great caution on the part of those responsible for the records and demands considerable time from the administrative department to count the hours. This scenario justifies the need for a digital system that guarantees the security of information over time. The objective of this work was to develop a time-tracking system for employees of institutions or companies. To achieve this objective, a microservice was created that receives requests from a client web application and stores them in a non-relational database. The unique feature of this project lies in the use of geolocation to verify the user's position, allowing the recording of hours only when the employee is within an area permitted by the company, or refusing it with an alert message otherwise. This technological solution is not only economically viable, but also provides security for the records and facilitates the process for both users and administrative staff.

Keywords: electronic register; microservice; web application; geolocation.

INTRODUÇÃO

É muito comum em empresas e instituições, o funcionário antes de iniciar a sua jornada de trabalho e após o término, “bater o ponto”, ou seja, realizar o seu registro para que o setor de RH possa verificar a quantidade de horas trabalhadas e tomar as medidas para pagamentos ou descontos. Um problema comum é que algumas instituições não possuem um sistema de apontamento de horas digital, recorrendo ao livro ponto físico ou algo semelhante.

Os registros de horas trabalhadas dos colaboradores são fundamentais para calcular salários, gerenciar banco de horas e executar outras atividades administrativas. No entanto, os registros físicos são naturalmente mais vulneráveis a rasuras e alterações, o que pode comprometer a integridade dos dados. Este sistema de registro em papel pode apresentar diversas falhas, incluindo a falta de registro por parte de um colaborador e a adulteração ou envelhecimento das folhas de registros. Ainda, poder-se-ia pensar em digitalizar tais registros de tempos em tempos, mas isso demandaria algum esforço humano, e portanto, eventualmente poderia haver erros. Como a tecnologia da informação poderia auxiliar para a resolução destes problemas?

Serviços de *Time-Tracking* como os pagos Harvest Monday e Jira, ou os gratuitos ClickUp e Clockify, já existem no mercado, mas com objetivos mais relacionados a uma agenda do que realizar o apontamento de horas.

O objetivo principal deste trabalho foi o desenvolvimento de uma aplicação *web* que armazena o registro de ponto de entrada e saída de um funcionário no trabalho com validação realizada por geolocalização utilizando o celular. Como objetivo secundário houve um estudo de caso no campus da faculdade de tecnologia de São Paulo – FATEC – unidade Mogi Mirim, mapeando a região e realizando os testes de validação.

As tecnologias utilizadas no desenvolvimento deste projeto foram: Java e JavaScript, fazendo uso dos *frameworks* Spring Boot para o *backend* e o React Js para o *frontend*. O banco de dados utilizado foi o Mongo DB.

REFERENCIAL TEÓRICO

Apresenta-se nesta seção a fundamentação teórica para trazer o embasamento científico e credibilidade à pesquisa, buscando esclarecer conceitos e termos chave, escolhas metodológicas, clareza e compreensão ao leitor.

Engenharia de *Software*

De acordo com Sommerville (2011), o *software* não é apenas um programa de computador, mas também todos os documentos relevantes e dados de configuração necessários para o funcionamento normal do programa. Pressman e Maxim (2016) asseveram que o *software* pode distribuir informação, que é o produto mais importante do nosso tempo. É capaz de transformar dados e gerenciar informações de negócios para serem úteis em todas as situações e melhorar a competitividade.

Sommerville (2011) define quatro atividades fundamentais ao processo:

- Especificação de *software*: o *software* e suas restrições são estabelecidos pelo time de desenvolvedores e os clientes;
- Desenvolvimento de *software*: o *software* é projetado e desenvolvido;
- Validação de *software*: o *software* é verificado para assegurar que possui aquilo que foi requerido pelo cliente;
- Evolução de *software*: é alterado para se adequar às modificações dos requisitos.

Pressman e Maxim (2016 p. 16) estabelecem o processo como sendo o “conjunto de atividades, ações e tarefas realizadas na criação de um artefato [...]”. Ainda para o autor, o processo é um método adaptativo que permite à equipe selecionar um conjunto adequado de ações e tarefas para executar o projeto no prazo e com a melhor qualidade. Eles definiram cinco atividades comuns que se aplicam a todos os projetos: Comunicação, Planejamento, Modelagem, Construção e Entrega.

Método Ágil *Design Thinking*

Métodos ágeis introduzem uma nova visão no desenvolvimento de *software*. Eles estão associados a uma nova maneira de pensar e mudanças culturais. Seu diferencial em termos de modelos antigos é que é mais focado no homem e não em processos, e também tem um número de valores, princípios e práticas que permitem a adaptação de fatores e resposta rápida aos custos de mudanças no mercado (PRIKLADNICKI, 2014; SBROCCO e MACEDO, 2012).

Brown (2010) diz que o método ágil *Design Thinking* (DT) é baseado no processo de *designers* visando entender quais as verdadeiras necessidades humanas, quais os recursos disponíveis e como integrá-los criando produtos desejáveis, tecnológicos e viáveis financeiramente para o público.

Segundo Vianna et al. (2012), o DT pode ser dividido em 3 principais etapas:

1. **Imersão:** onde ocorre o entendimento do problema, contexto e perspectiva sob o ponto de vista do público-alvo, traçando objetivos sob o comando da empatia. São possíveis de se realizar entrevistas, pesquisas, observações, mapeamento de empatia, análise de dados etc. Nesta fase, pode existir uma etapa de Análise e Síntese, na qual ocorre a organização e interpretação das informações coletadas para identificar padrões, obter *insights* e oportunidades.
2. **Ideação:** Nesta etapa ocorre a geração de um amplo leque de soluções criativas e inovadoras para o problema definido por meio de *brainstorming*, mapas mentais, *sketching*, prototipagem rápida etc.
3. **Prototipação:** Nesta fase ocorre o desenvolvimento de protótipos tangíveis para testar e validar as ideias geradas na etapa anterior. Para isso, pode-se haver a criação de protótipos simples, testes com usuários, *feedback*, iteração e refinamento dos protótipos.

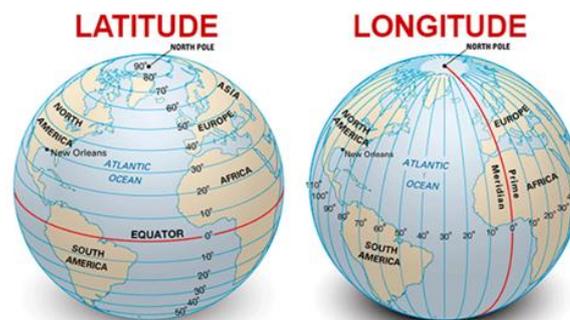
Por fim, se o intuito é fazer do protótipo um produto, as fases anteriores podem demonstrar a aptidão para aplicações em cenários reais, devendo haver novos testes de usabilidade, entrevistas pós-teste, análise de resultados, implementação da solução final, monitoramento e aprimoramento contínuo.

Geolocalização

Um serviço baseado em localização (SBL) tem o objetivo de integrar a localização de um dispositivo móvel com outras informações de interesse, podendo aplicar seus resultados em funcionalidades de interesses diversos, como por exemplo o Sistema de Posicionamento Global, do inglês o Global Positioning System (GPS) (CORREIA, 2004; OLIVEIRA, 2014).

Como medidas de posicionamento sobre o globo terrestre, ou seja, uma coordenada geográfica, a latitude e longitude são largamente utilizadas. Antes de abordar suas definições, lembremos dois referenciais importantes no globo terrestre. A linha do Equador se situa em distância iguais ao polo norte e sul, seccionando o planeta Terra nos hemisférios norte e o hemisfério sul. Já o meridiano de Greenwich divide o planeta nos hemisférios leste ou oriental e o oeste, sendo colocado no mapa de forma vertical. A figura 1 ilustra estes referenciais.

Figura 1: Latitude e Longitude.



Fonte: Giovanini (2021).

A latitude trata-se da distância em graus de um ponto sobre o globo até a linha do Equador medida ao longo de paralelas do meridiano de Greenwich, variando entre 0° e 90° para Norte (N) ou para Sul (S). A longitude é a distância ao meridiano de Greenwich medida ao longo de paralelas da linha do Equador, podendo variar entre 0° e 180° para Leste (E) ou para Oeste (W) (GIOVANINI, 2021).

PROCEDEIMENTOS METODOLÓGICOS

Por se adaptar muito bem a estudos de casos com problemáticas de clientes, o *DT* foi escolhido como método ágil para nortear o desenvolvimento deste projeto. Na sequência, são apresentadas as etapas do *DT* utilizadas na construção do projeto.

42

Imersão

Nesta etapa, foram realizadas entrevistas com alguns funcionários da Fatec Arthur de Azevedo de Mogi Mirim para compreender como era realizado o processo de registro de ponto na faculdade. Verificou-se que o registro de ponto é feito de maneira manual pelos funcionários. Professores devem se dirigir até a sala da coordenação a cada início de aula para o registro enquanto outros funcionários devem se deslocar até o prédio do setor administrativo para o processo de assinatura.

Ao fechamento do mês, os funcionários do setor administrativo realizam de maneira manual a verificação dos registros de ponto, o fechamento e pagamento dos funcionários. Além disso, uma vez que estes registros são físicos, ficam sujeitos a sofrerem erros, rasuras e alterações, evidenciando a falta de segurança, confidencialidade e integridade das informações nele contidas.

Diante destas situações, foi possível definir um escopo de pesquisa e iniciar as discussões para possíveis soluções exequíveis que poderíamos utilizar para solucionar a problemática levantada.

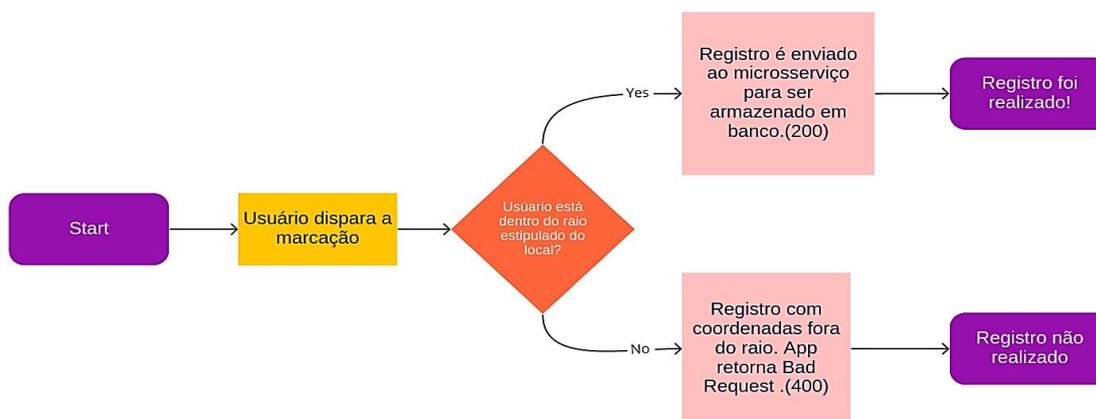
Ideação

Foram realizados *brainstorms* entre os autores deste trabalho para decidir que tipo de solução seria apresentada para sanar para a problemática levantada. Dentre as diversas possibilidades, um cenário passível de ser alcançado seria aquele em que houvesse um aplicativo que realizasse o *time tracking* dentro de um ambiente onde todos estivessem habituados com o uso desta tecnologia, e partindo desta premissa, entende-se que um aplicativo dedicado a prover o registro, a consulta e o ajuste de horas trabalhadas deveria, em teoria, resolver os problemas citados anteriormente.

Por fim, ficou decidido que o desenvolvimento de um sistema de apontamento de horas com o auxílio da geolocalização como confirmação da presença do indivíduo dentro do raio do local de trabalho seria uma ótima solução. A Figura 2 ilustra o fluxo principal da aplicação idealizada:

43

Figura 2: Fluxo principal da aplicação.



Fonte: O próprio autor.

Com um objetivo definido, decidiu-se por utilizar as seguintes ferramentas de desenvolvimento e bases de pesquisa:

- **Java e JavaScript:** linguagem de programação interpretada estruturada e de *script* em alto nível (MDN WEB DOCS, 2024).
- **Spring Boot:** *Framework* de Java para construir o microserviço que fará as tratativas para armazenamento dos registros (SPRING, 2024);
- **React Js:** biblioteca JavaScript para construção de interfaces de usuário (ABRAMOV; NABORS, 2023).
- **MongoDB:** Banco de dados não relacional (MONGODB, 2024);
- **Geocoding API Usage and Billing:** API para coordenadas geográficas (GOOGLE MAPS PLATFORM, 2024).

Como requisitos importantes para a utilização da aplicação, o usuário deverá ter *internet* disponível e o aplicativo instalado no seu *smartphone*. Ainda, ficará a

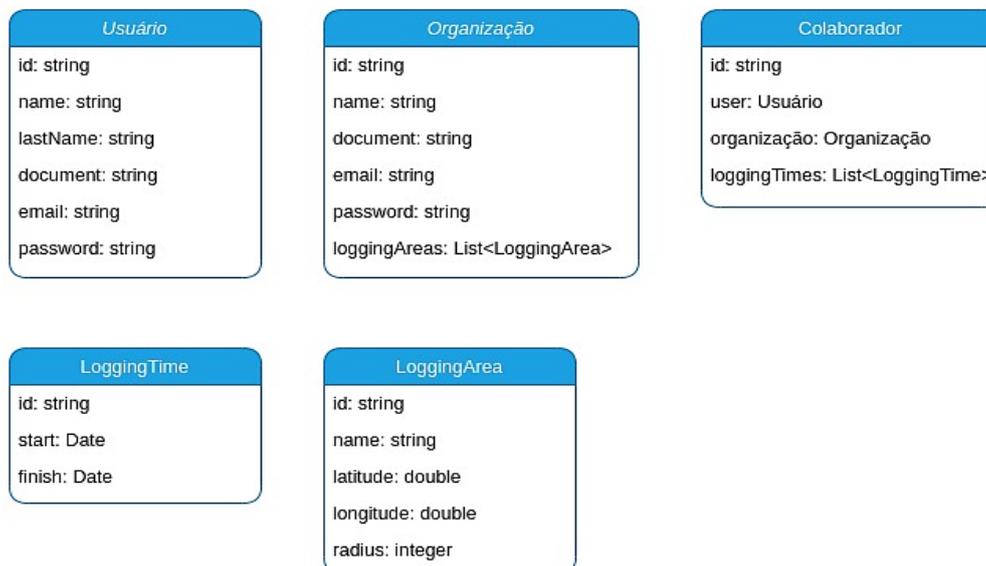
responsabilidade da empresa/instituição, fornecer informações importantes dos tipos de dados que necessitam retirar do sistema.

Pensando na utilização do protótipo pelo público-alvo idealizaram-se personas que devem ser funcionários de uma empresa que deve prestar contas com horários de entrada e saída, podendo este ser um alto executivo ou um auxiliar. Para atingir todos os perfis, a aplicação deve ser intuitiva e de simples utilização, trazendo informações essenciais para o seu propósito que é registrar o ponto.

Prototipação

A prototipação abrangeu duas partes principais: o *frontend* e o *backend*. A seguir, detalha-se a integração entre essas partes e a comunicação entre as telas da aplicação. O *backend* é a parte da aplicação onde os dados são processados e armazenados. O *frontend*, responsável pela interface *web*, coleta os dados inseridos pelo usuário final e os envia para a API ou aplicação *backend*.

Para o controle de versão, utilizou-se o Git, e o repositório de códigos foi hospedado na plataforma GitHub. No desenvolvimento, empregou-se o *framework* Spring Boot em conjunto com o MongoDB, um banco de dados não relacional, cuja modelagem é ilustrada na Figura 3.

Figura 3: Modelagem de Banco de Dados.

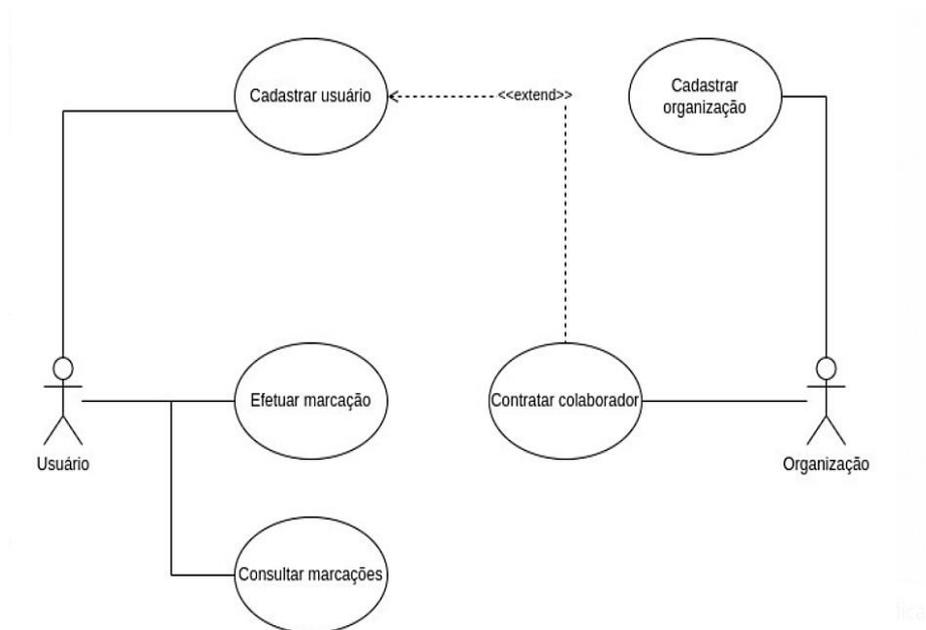
Fonte: O próprio autor.

A modelagem consiste em documentos ao invés de tabelas, cujo relacionamento se dá por referência do campo de identificação (id).

A aplicação *frontend* foi feita utilizando a biblioteca React Js para aplicações SPA, permitindo o usuário colaborador registrar e consultar suas horas, bem como o perfil de uma empresa (organização) admitir novos colaboradores, e por conseguinte, habilitar para os usuários colaboradores a funcionalidade de registro e consulta de tempo trabalhado.

Nele, está contida uma biblioteca de geolocalização que retorna a latitude e a longitude do usuário no momento que ele realiza seus registros, de modo que não seja permitido ao colaborador realizá-los fora da área que contém a empresa ou instituição.

O diagrama de caso de uso ilustrado na figura 4 ilustra as principais funcionalidades da aplicação:

Figura 4: Diagrama de caso de uso da aplicação.

Fonte: O próprio autor.

Observa-se na figura 4, cinco principais funcionalidades dadas por Cadastrar usuário, Cadastrar Organização, Efetuar Marcação, Consultar Marcação e Contratar Colaborador.

Existem dois atores principais definidos por Usuário e Organização, sendo que:

- o ator usuário terá acesso a apenas três funcionalidades: cadastrar usuário, efetuar marcação e contratar colaborador;
- o ator organização possui acesso a duas funcionalidades: cadastrar organização e contratar colaborador, sendo esta última dependente de haver usuários cadastrados.

Cálculo da Distância entre o Colaborador e o Centro da Cerca Virtual

Para garantir que o funcionário esteja dentro da empresa ao tomar o registro do ponto, a organização poderá criar um raio de validação formando uma região em que os registros poderão ser realizados a qual será chamada de cerca virtual.

A geometria sobre a superfície da Terra é não euclidiana, mas utilizando a hipótese de que para pequenas distâncias a superfície de interesse será um plano, são válidas as relações trigonométricas no triângulo retângulo e a geometria plana. A figura 5 ilustra uma distância entre dois pontos em um plano e o triângulo formado:

Figura 5: Exemplo do cálculo.



Fonte: O próprio autor.

Supondo dois pontos $P_1(x_1, y_1)$ e $P_2(x_2, y_2)$ em um plano, então aplicando-se o teorema de Pitágoras, a distância entre estes dois pontos será dada por

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1.0)$$

Considerando dois pontos com suas coordenadas de latitude e longitude, foi implementado no *backend* a função (1.1) para determinar a distância desejada entre um colaborador e um ponto referência dado pelo centro da cerca virtual:

$$D_g = \sqrt{(latOrg - latUser)^2 + (longOrg - longUser)^2} \quad (1.1)$$

onde D_g é distância entre o colaborador e centro da cerca virtual, em graus; $latOrg$ e $longOrg$ são latitude e longitude da cerca virtual, em graus; $latUser$ e $longUser$ são latitude e longitude do colaborador, em graus.

Sabe-se que um minuto de latitude Norte para sul é equivalente a 1 [milha] náutica ou convertendo, 1.852,1341 [m]. Assim, como um grau equivale a 60 minutos, segue aproximadamente que $1^\circ = 111.128$ [m]. De maneira alternativa, um minuto de latitude é 1.852 [m], então $1^\circ = (60).(1852)$ [m] = 111.120 [m]. Desta forma, para converter a distância para metros, ou seja, obter a distância entre o colaborador e o

centro da cerca virtual em metros, realizou-se uma regra de três simples aproximando $1^\circ = 111.120$ [m], como na fórmula (1.2):

$$D_m = 111.120 * D_g \quad (1.2)$$

Após isso, compara-se a diferença entre a distância encontrada o raio da circunferência pré-definida para a região de validação, como na fórmula (1.3)

$$V = R - D_m \quad (1.3)$$

e caso o resultado seja igual ou maior que zero, isso significa que o usuário se encontra dentro da área permitida para os registros, caso contrário, estará fora.

RESULTADOS DE DISCUSSÃO

Nesta seção são apresentados os resultados obtidos com ênfase nas telas desenvolvidas e passos a serem seguidos nos perfis de Usuário Colaborador e Organização. Por fim, será apresentado um teste de campo realizado dentro da instituição Fatec Arthur de Azevedo.

Perfil Usuário Colaborador

A imagem da figura 6 representa a página inicial para a qual o usuário é levado quando acessa a aplicação.

Figura 6: Página inicial da aplicação para o usuário colaborador.



Fonte: O próprio autor.

O primeiro passo que uma pessoa precisa dar para ter acesso aos serviços do Safe Logger é o de criar uma conta de usuário colaborador. Para isso, basta que ele navegue até a página de criação de contas, clicando no link “Crie uma aqui”, como ilustrado na figura 7.

49

Figura 7: Página de criação de conta para o usuário colaborador.

Crie sua conta

Nome

Sobrenome

Cpf

Email

Senha

Confirme a senha

Criar

Criar Conta Business

VOLTAR AO LOGIN

Fonte: O próprio autor.

Na imagem da figura 8, é mostrada a página de criação de contas para o usuário colaborador. Ao inserir seus dados e clicar no botão criar ele será redirecionado para tela home da aplicação, onde poderá registrar suas horas trabalhadas, caso esteja vinculado à uma organização.

Figura 8: Página home da aplicação para o usuário colaborador.

Safe Logger Logout

Olá, Danilo!

Quinta-feira, 25 de novembro de 2021

01:00

Inicio Fim

--- --

Registrar tempo

Ver minhas horas

Fonte: O próprio autor.

Além de uma conta criada, para que um colaborador possa acessar a página home, precisa também ter um vínculo com a organização que trabalha, caso contrário será levado a uma página de aviso, como ilustrado na Figura 8. O usuário, ao clicar no botão “Registrar Tempo” pela primeira vez no dia, terá sua marcação exibida na tela logo abaixo da palavra “Início”. No entanto, para que isso ocorra é preciso que o usuário se encontre dentro do raio da cerca virtual definida pela organização onde trabalha, caso contrário o registro de tempo não será realizado. A mesma lógica serve para a segunda marcação, exibida logo abaixo da palavra “Fim”, no caso de um fluxo feliz.

Em havendo registrado as duas marcações possíveis dentro de um dia, a tentativa de uma terceira marcação lançaria uma exceção no fluxo por parte do *backend*, dizendo não ser possível realizar mais um registro em um mesmo dia.

Figura 9: Página de aviso ao colaborador não vinculado a empresa no momento.



Fonte: O próprio autor.

Para finalizar a experiência do usuário, ele ainda pode consultar todas as suas horas registradas clicando no botão “Ver minhas horas” presente na Figura 8. Ao fazê-lo, será redirecionado para a página de consulta, a seguir.

Figura 10: Página para ver os registros de tempo do colaborador.



Fonte: O próprio autor.

Perfil Organização

Este perfil é destinado às empresas e instituições que desejam contratar os serviços do Safe Logger. Vale lembrar que para este trabalho, a única funcionalidade presente para este perfil é a de indicar pessoas como colaboradores da organização, deixando para o futuro funcionalidades que permitirão uma melhor visibilidade e controle das horas trabalhadas das pessoas.

A figura 10 ilustra o protótipo da página inicial para o perfil Organização, também chamado de perfil Business. Analogamente ao perfil de usuário colaborador, trata-se de uma página de login ou sign-in, para quando o perfil já possui uma conta de Organização (Conta Business).

51

Figuras 10 e 11: Página inicial e de criação de conta Business.

Fonte: O próprio autor.

De forma análoga ao perfil de usuário colaborador, basta clicar em “Crie uma Aqui” presente na Figura 11 para navegar até a página de criação de uma conta Business, a seguir.

Ao inserir os dados da empresa, basta clicar no botão “Criar” para a criação da conta ser realizada. Neste momento, o usuário perfil organização é automaticamente levado à sua respectiva página home. Uma informação importante é a de que para este trabalho, foi acordado que o usuário não informaria o raio da cerca virtual, ficando ela definida em 10 metros. Esta distância adotada serviu para facilitar os testes da aplicação quando os graduandos e o orientador foram à Fatec de Mogi Mirim para

verificar a aplicação funcionando na prática. Na figura 12, o usuário insere o CPF de um usuário colaborador e clica no botão “Admitir Colaborador”, finalizando o fluxo principal do perfil de organização.

52

Figura 12: Página home da aplicação para o usuário perfil organização.



Fonte: O próprio autor.

Testes

Para a validação da aplicação, realizaram-se diversos testes, sendo um destes dentro da instituição Fatec Arthur de Azevedo de Mogi Mirim, que consistiu em verificar se o registro de um usuário realmente era efetuado com êxito dentro da cerca virtual estabelecida, que no caso teve um raio de 100 metros. O registro foi efetuado com sucesso, como esperado.

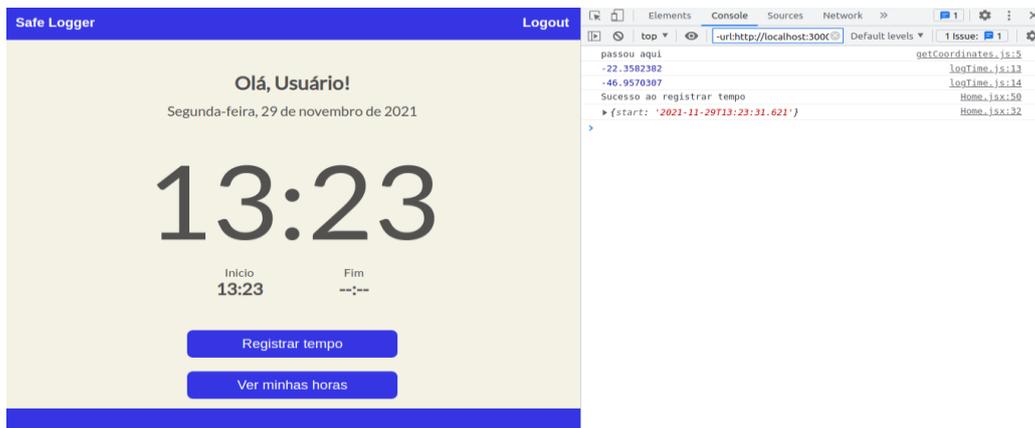
Figura 13: Exemplo de uma cerca virtual de 100 metros.



Fonte: O próprio autor.

A figura 14 mostra o primeiro registro do dia efetuado com sucesso.

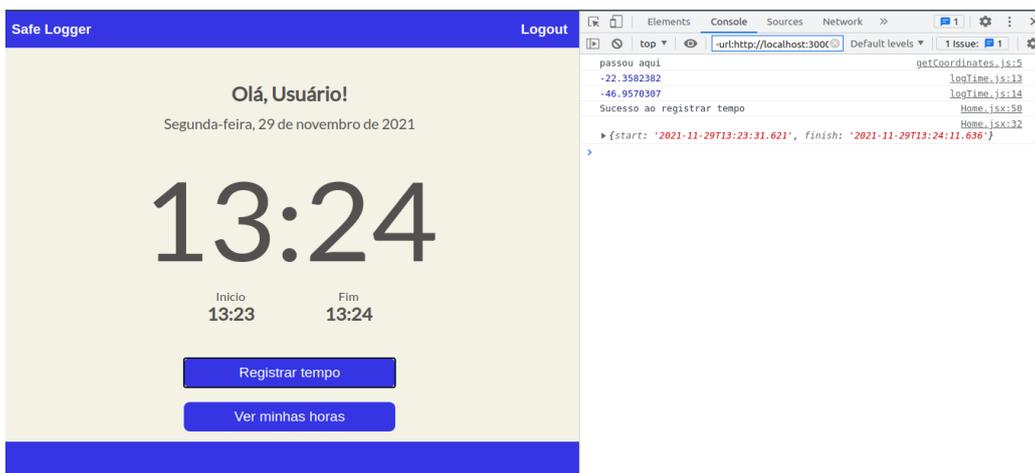
Figura 14: Primeiro registro do dia realizado com sucesso.



Fonte: O próprio autor.

A figura 15 mostra o segundo e último registro do dia também efetuado com sucesso.

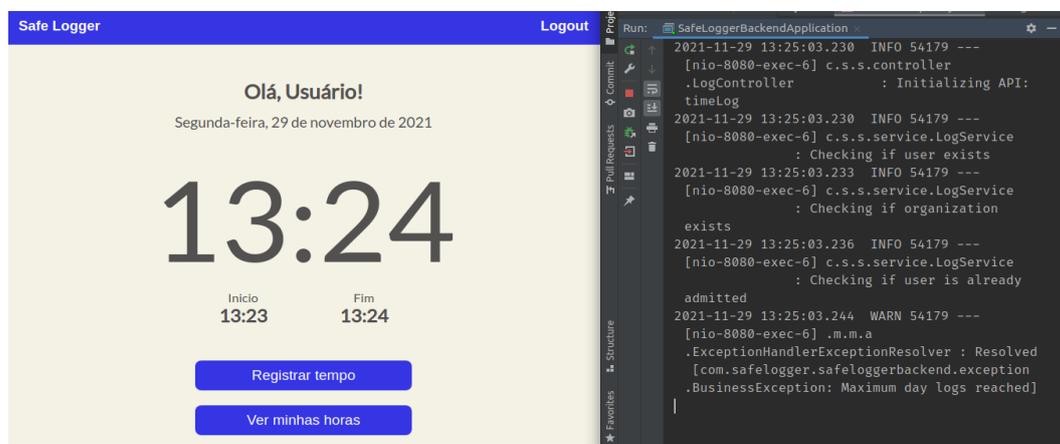
Figura 15: Segundo registro do dia realizado com sucesso.



Fonte: O próprio autor.

Como já mencionado anteriormente, um terceiro registro de tempo em um mesmo dia lançará uma exceção do sistema, conforme ilustrado na figura 16.

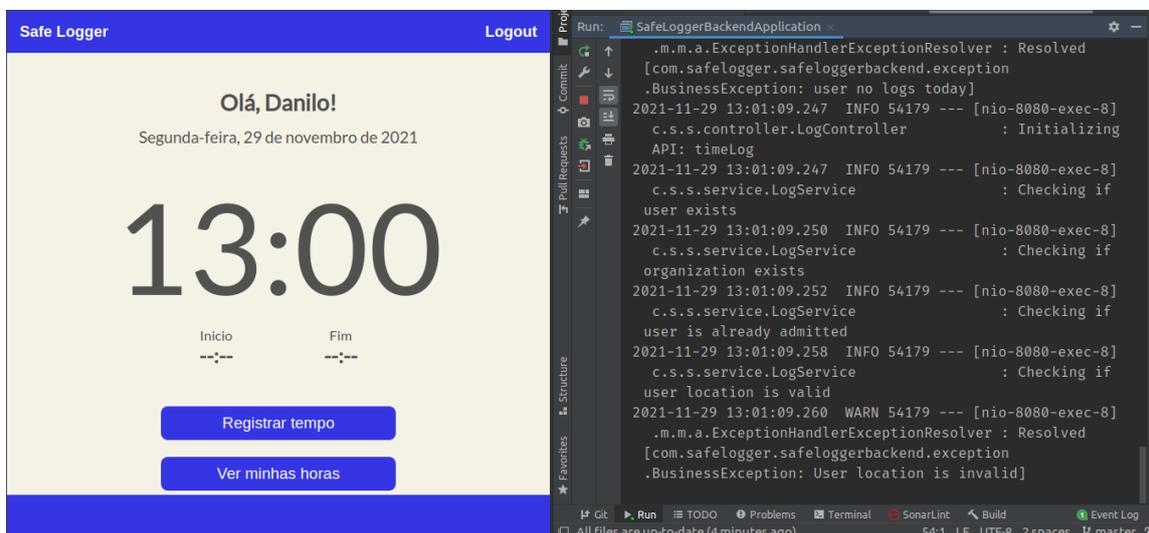
Figura 16: Demonstração da exceção de quantidade de registros máxima em um dia.



Fonte: O próprio autor.

Ainda, a figura 17 mostra o caso de o colaborador ter tentado o registro fora da cerca virtual, que como consequência resulta em uma exceção do sistema com a mensagem apropriada.

Figura 17: Demonstração da exceção de colaborador estando fora cerca virtual.



Fonte: O próprio autor.

Foram realizadas tentativas de registro de tempo próximo ao portão de entrada da faculdade, para garantir que também não seria possível o registro em uma

localização fora da área permitida. De fato, a aplicação funcionou novamente como esperado.

Durante o teste, foi percebido que por alguma razão ainda desconhecida, ao realizar uma segunda marcação de tempo, a aplicação se valia das mesmas coordenadas obtidas anteriormente da primeira marcação, de modo que estando ou não dentro da cerca virtual, processava os registros com sucesso, comportando-se de forma não desejada. Felizmente, após algumas tentativas, descobriu-se que ao desativar a internet móvel e em seguida ativá-la novamente, a aplicação conseguiu buscar a latitude e longitude, conforme a regra de negócio definida.

CONSIDERAÇÕES FINAIS

Muitas empresas e instituições ainda realizam o processo de registro de horário de início e término da jornada de trabalho de maneira manual, utilizando cartões perfurados, marcações impressas ou assinaturas manuais. Nesses casos, o setor de RH deve, ao final do mês, analisar a quantidade de horas trabalhadas e repassar os dados para o setor responsável pelos pagamentos ou descontos. A Fatec Arthur de Azevedo de Mogi Mirim é um exemplo de instituição que utiliza registros manuais, onde os funcionários assinam um campo no início e no término da jornada, e o setor administrativo verifica os dados individualmente ao final do mês.

Para resolver essa problemática, desenvolveu-se uma aplicação de registro automático de ponto com o intuito de realizar os registros de forma segura, armazená-los em um banco de dados, fornecer os dados ao setor responsável quando necessário e agregar valor tecnológico com a geolocalização. Assim, o registro pode ser realizado em qualquer local mapeado pela empresa, bastando que o funcionário utilize a aplicação dentro dessa área, eliminando a necessidade de equipamentos fixos em pontos específicos da empresa.

A aplicação foi desenvolvida nas linguagens Java e JavaScript, utilizando os frameworks Spring Boot para o backend e React.js para o frontend. Testes demonstraram resultados satisfatórios em termos de usabilidade tanto para os

administradores quanto para os colaboradores. Os administradores podem inserir colaboradores, mapear o raio de captura onde os registros podem ser realizados e acessar os dados obtidos. Os colaboradores podem registrar suas horas quando estão dentro da cerca virtual e são alertados caso tentem registrar fora da área permitida.

Como projetos futuros, idealiza-se a implementação de melhorias e novas funcionalidades, tais como: permitir o registro automático de tempo quando o usuário entra ou sai da cerca virtual; permitir que o usuário defina filtros de data para consulta; incluir registros de horário intrajornada (almoço); mapear áreas em setores isolados dentro da empresa ou instituição; possibilitar a definição de múltiplas cercas virtuais com raios ajustáveis; fornecer acesso às horas trabalhadas pelos colaboradores; e criar dashboards que mostrem a frequência de registros tanto para a organização quanto para os colaboradores.

Espera-se que este projeto proporcione segurança e otimização para as organizações, facilite o uso para os colaboradores e agregue valor ao setor de equipamentos de registros de ponto com uma solução criativa, tecnologicamente viável e econômica.

REFERÊNCIAS

BROWN, Tim. **Design Thinking: Uma Metodologia Poderosa para Deletar o Fim das Velhas Ideias**. Rio de Janeiro: Elsevier, 2010.

CORREIA, A. G. S. **Aplicações e Serviços Baseados em Localização**. Pontifícia Universidade Católica do Rio de Janeiro, 2004. Disponível em < <http://www-di.inf.puc-rio.br/~endler/courses/Mobile/Monografias/04/AdolfoCorreia-Mono.pdf> > Acessado em 14 maio 2024.

GIOVANINI, Adenilson. **Latitude e longitude: o que são e como calcular!** Disponível em: <https://adenilsongiovanini.com.br/blog/latitude-e-longitude-o-que-sao-e-como-calculiar/>. Acesso em: 22 maio. 2024.

GOOGLE MAPS PLATFORM. **API Geocoding**. Disponível em: <https://developers.google.com/maps/documentation/geocoding?hl=pt-br>. Acesso em: 15 jun. 2024.

MDN WEB DOCS. **JavaScript**. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>. Acesso em: 10 jun. 2024.

MONGODB. **Documentação do Mongo DB**. Disponível em: <https://www.mongodb.com/pt-br/docs/>. Acesso em: 15 jun. 2024.

ABRAMOV, Dan; NABORS, Rachel. **Apresentando react.dev**. 2023. Disponível em: <https://pt-br.react.dev/blog/2023/03/16/introducing-react-dev>. Acesso em: 12 jun. 2024.

OLIVEIRA, Luciano da Silva. **Desenvolvimento de um aplicativo Android utilizando geolocalização**. Coordenadoria do Curso de Ciência da Computação. Universidade Estadual de Mato Grosso do Sul. 2014. Disponível em : <https://www.comp.uems.br/~ricardo/PFCs/PFC%20151.pdf>. Acesso em: 25 maio 2024.

PRESSMAN, Roger S; MAXIM, Bruce. **Engenharia de Software. Uma Abordagem Profissional**. 7ª edição. Porto Alegre: AMGH Editora Ltda, 2016.

PRIKLADNICKI, Rafael; WILLI, Renato; MILANI, Fabiano. **Métodos Ágeis para Desenvolvimento de Software**. Porto Alegre: Bookman, 2014

SBROCCO, José Henrique Teixeira de Carvalho; MACEDO, Paulo Cesar de. **Metodologias Ágeis: Engenharia de Software Sob Medida**. São Paulo: Érica, 2012.

SPRING. **Spring Boot: O que é, Etapas e como Aplicar**. Disponível em: <https://upis.br/blog/design-Thinking/>. Acesso em: 08 mai. 2024.

SOMMERVILLE, Ian. **Engenharia de Software**. 9ª edição. São Paulo: Pearson, 2011.

VIANNA, M. et al. **Design Thinking: Inovação em negócios**. Rio de Janeiro: MJV Press, 2012.

Os autores declararam não haver qualquer potencial conflito de interesses referente a este artigo.