

UNIVERSIDADE DE SÃO PAULO

JOÃO VICTOR MENECHINI

VITOR LUIZ LIMA CARAZZI

**MECAMAPS: PLATAFORMA DE LOCALIZAÇÃO
E GERAÇÃO DE ROTAS INDOOR UTILIZANDO
QR CODES**

São Paulo
2021

UNIVERSIDADE DE SÃO PAULO

JOÃO VICTOR MENEZHINI

VITOR LUIZ LIMA CARAZZI

MECAMAPS: PLATAFORMA DE LOCALIZAÇÃO E GERAÇÃO DE ROTAS INDOOR UTILIZANDO QR CODES

Trabalho apresentado à Escola Politécnica
da Universidade de São Paulo para obtenção
do Título de Engenheiro Mecatrônico.

São Paulo
2021

UNIVERSIDADE DE SÃO PAULO

JOÃO VICTOR MENEZHINI

VITOR LUIZ LIMA CARAZZI

MECAMAPS: PLATAFORMA DE LOCALIZAÇÃO E GERAÇÃO DE ROTAS INDOOR UTILIZANDO QR CODES

Trabalho apresentado à Escola Politécnica
da Universidade de São Paulo para obtenção
do Título de Engenheiro Mecatrônico.

Orientador:

Thiago de Castro Martins

São Paulo
2021

AGRADECIMENTOS

Vitor Luiz Lima Carazzi

Aos meus pais, Cristiano e Sheyla, pelo suporte e incentivo durante toda a minha graduação, assim como durante toda a minha vida.

Aos meus familiares, avós e tios, pelo amparo durante toda a minha trajetória.

Aos meus colegas da Engenharia Mecatrônica, em especial à turma de 2017, que sempre foram colaborativos e me ajudaram muito no caminho para me formar.

À todos os meus amigos que me acompanharam durante meus 6 anos na universidade, em especial aos que conheci pela Associação de Engenharia Química (AEQ), que estiveram comigo nos momentos mais marcantes da minha graduação.

Ao meu Orientador, Thiago de Castro Martins, por todas as instruções e discussões sobre o trabalho.

Por fim, ao meu parceiro de projeto, João Victor Meneghini, pelo trabalho e cooperação ao longo de todo o trabalho.

João Victor Meneghini

Aos meus pais, João e Silvia, pelo incentivo constante a lutar por meus objetivos, por todo o apoio e confiança no decorrer da minha graduação e pelo suporte tanto nos momentos de dor, quanto nos de conquistas.

Aos meus amigos da Engenharia Mecatrônica, pelos aprendizados, pela companhia nos piores e melhores momentos, pela ajuda em momentos de dificuldade, e, mais importante, pela amizade sincera e verdadeira.

Ao meu amigo Pedro Medeiros, pelas noites estudando para provas, pelas discussões e conversas importantes de carreira e por estar presente nos momentos mais marcantes da minha graduação.

Ao meu amigo Otávio Palma, pelo apoio durante o projeto, tanto pessoal quanto técnico.

Aos meus amigos de infância Matheus, Pedro Antônio, Pedro Marcos e Pedro Henrique, por todos esses anos de amizade e conquistas compartilhadas.

Ao meu Orientador, Thiago de Castro Martins, pelos direcionamentos e questionamento pertinentes para o desenvolvimento deste trabalho.

Por último, ao meu amigo, Vitor Carazzi, pelo trabalho duro e dedicação, mirando excelência no projeto.

RESUMO

O presente trabalho objetiva a criação de uma plataforma de localização indoor com o uso de QR codes, assim como a geração de rotas para o Prédio das Engenharias Mecânica, Mecatrônica e Naval da Escola Politécnica da Universidade de São Paulo. O desenvolvimento dessa plataforma poderá resolver o problema de encontrar localidades, como salas e laboratórios, dentro do prédio, problema esse que é enfrentado por muitos alunos e visitantes do local. A ferramenta foi desenvolvida em javascript, utilizando a biblioteca React. Nela, o usuário seleciona a posição atual e o local onde deseja ir e o site mostra o trajeto entre esses dois pontos. No decorrer do trajeto, ele pode atualizar sua posição manualmente ou escaneando algum QR Code espalhado pelo prédio.

Palavras-Chave – Localização indoor, Geração de rotas indoor, QR Code, SVG.

ABSTRACT

This work aims to create an platform of indoor location using QR codes, as well as the generation of routes to the Mechanical, Mechatronics and Naval Engineering Building of the Polytechnic School of the University of São Paulo. The development of this platform will solve the problem of finding locations, such as rooms and laboratories, inside the building, a problem that is faced by many students and visitors of the site. The tool was developed in javascript, using the React library. In it, the user selects the current position and the place where he wants to go and the website shows the path between these two points. Along the way, he can update his position manually or by scanning some QR Code scattered around the building.

Keywords – Indoor location, Indoor route generation, QR Code, SVG.

LISTA DE FIGURAS

1	Métodos mais comuns usados no IPS	15
2	Tipos de Qr Codes quanto ao armazenamento	18
3	Tipos de Qr Codes quanto ao estilo	18
4	Estrutura do QR Code	19
5	Leitura do QR Code	19
6	Mapa criado com Google Maps API	20
7	Mapa criado com iMapBuilder	20
8	Distribuição dos alunos de acordo com seus cursos	23
9	Opções mais procuradas ao visitar o prédio	24
10	Caminho Fácil	25
11	Caminho Médio	26
12	Caminho Difícil Térreo	27
13	Caminho Difícil Superior	28
14	Menu Principal	33
15	Visualizar Mapa	34
16	Selecionar Opção	35
17	Gerar Rota	36
18	Exemplo de grafo simples. Destaque para o menor caminho, com saída do nó A com destino ao nó M	37
19	Pontos térreo	46
20	Pontos superior	47
21	Exemplo de linha do Esqueleto de caminhos em JSX com a condicional de cor e opacidade	48
22	QR Code Portaria	49

23	Posicionamento dos QR Codes - Andar Térreo	49
24	Posicionamento dos QR Codes - Andar Superior	50
25	Menu Principal do MeCaMaps	52
26	Página do mapa térreo do MeCaMaps	52
27	Página do mapa superior do MeCaMaps	53
28	Página do QRCode	53

LISTA DE TABELAS

1	Comparativo de diferentes tecnologias para ISP	19
2	Quantidade de lugares nomeados por categoria	45
3	Teste de Funcionalidade - Caminhos	60
4	Teste Final - Destinos	61

SUMÁRIO

1	Introdução	12
1.1	Contexto	12
1.2	Motivações	13
1.3	Objetivos	13
2	Estado da Arte	14
2.1	Tecnologias utilizadas em IPS	14
2.1.1	Ultrassom	15
2.1.2	WiFi ou WLAN ou IEEE 802.11	16
2.1.3	BLE (Bluetooth Low Energy)	16
2.1.4	Radio Frequency Identification (RFID)	16
2.2	QR Codes	17
2.2.1	Panorama geral	17
2.2.2	QR Codes e IPS	18
3	Metodologia	22
3.1	Pesquisa de Demanda e Destinos	22
3.1.1	Resultado da Pesquisa	22
3.1.1.1	Pergunta 1	23
3.1.1.2	Pergunta 2	23
3.2	Teste de Funcionalidade	24
3.2.1	Caminhos	25
3.2.2	Resultados do teste	29
3.3	Requisitos do projeto	29

3.4	Casos de uso	30
3.4.1	Menu Principal	30
3.4.1.1	Visualizar mapa	31
3.4.1.2	Selecione onde está	31
3.4.1.3	Selecione seu destino	31
3.4.1.4	Gerar Rota	31
3.4.1.5	Fale conosco	31
3.4.2	Página do Mapa	32
3.4.2.1	Menu Principal	32
3.4.2.2	Trocar andar	32
3.4.2.3	Selecione onde está	32
3.4.2.4	Selecione seu destino	33
3.5	Prototipagem	33
3.5.1	Telas da plataforma	33
3.6	Desenvolvimento	36
3.6.1	Algoritmo de Dijkstra	36
3.6.2	Progressive Web App (PWA)	38
3.6.3	React	39
4	Resultados	41
4.1	Organização do Projeto	41
4.1.1	Componentes	41
4.2	Rotas	45
4.2.1	Esqueleto de caminhos	45
4.2.2	Geração de Rotas	45
4.2.2.1	Dicionário de Coordenadas de Pontos	47
4.2.2.2	Dicionário de ligações e distâncias de pontos	47

4.2.2.3	Esqueleto de caminhos em JSX com linhas nomeadas . . .	48
4.2.3	Mapas	48
4.2.4	QR Codes	49
4.3	Resultados	51
4.4	Manutenção	54
4.5	Teste final	54
5	Conclusão	56
	Referências	57
	Apêndice A	60
A.1	Pesquisa de Demanda e Destinos	60
A.2	Teste de Funcionalidade	60
A.3	Teste Final	60

1 INTRODUÇÃO

1.1 Contexto

No final do século passado, estudos foram iniciados acerca de uma nova aplicação utilizando tecnologias que pudessem, então, promover em ambientes internos o que o GPS (*Global Positioning System*) entregava externamente; e, assim, surgiu o IPS (*Indoor Positioning systems*).

Atualmente, o GPS se tornou uma tecnologia não mais exclusivamente militar – foi criado em 1957, em um período conflituoso conhecido como Guerra fria -, mas também comercial; existe uma rede de satélites artificiais espalhados ao redor do planeta, que permitem qualquer indivíduo com um celular moderno saber sua localização e, por conseguinte, ser guiado de um ponto a outro do mapa, com certa precisão, algo em torno de 3 metros (MAXIT, 2020). Contudo, apesar da sua importância e ampla usabilidade, o GPS não é uma tecnologia que se comporta corretamente em ambientes internos, a precisão é insuficiente para determinar localização correta e a comunicação com os satélites pode ficar comprometida devido às paredes no local; sendo assim, a ineficiência apresentada pelo GPS nesses casos, promoveu a necessidade de criação do IPS.

Ao longo dos últimos 20 anos, mais de 62 artigos foram publicados (SILVA; SOS-PEDRA; HUERTA, 2019) relatando a evolução do IPS e, apesar de existirem empresas que trabalhem com plataformas de localização indoor, ainda não há algo consolidado na área. Dentre as tecnologias utilizadas, podemos citar WiFi, Bluetooth, RFID, Ultrassom, Campo magnético, dentre outras, o que denota a gama de possibilidades que podem levar à solução esperada; todavia, todas elas são bastante complexas e tem custos relativamente altos.

De forma rudimentar, pode-se relacionar IPS a outras opções encontradas em locais de grande circulação de pessoas, como um mapa simples do ambiente, painéis interativos e guias ou guardas do estabelecimento. A grande dificuldade aqui se encontra na imprecisão das informações fornecidas e, também, do tempo gasto para encontrar um determinado

ponto, já que a navegação guiada não está presente em nenhuma dessas opções.

Em meio a este cenário, visando criar um sistema funcional, porém mais simples e menos custoso que os atuais existentes, pensou-se na utilização de QR Codes como forma de identificar o posicionamento e navegação em ambientes internos. No decorrer deste trabalho, será apresentado o processo de desenvolvimento desta plataforma, bem como os resultados obtidos, tendo como foco a implementação no Prédio das Engenharias Mecânica, Mecatrônica e Naval da Escola Politécnica da USP.

1.2 Motivações

Este estudo apresenta motivação de caráter acadêmico, visto que objetiva solucionar um problema vivido tanto pelos autores na busca por salas de aula, professores e laboratórios, quanto por outros colegas e visitantes que passaram por dificuldades similares.

Paralelamente, há motivação pessoal, uma vez que o campo estudado pode ser considerado recente, ainda mais em se tratando da tecnologia que se pretende trabalhar. Apesar de existirem algumas plataformas comerciais (nenhuma que utiliza QR Codes), ainda considera-se IPS pouco conhecido, o que abre espaço para inovações na área surgirem. Assim, acredita-se que seja possível criar uma solução satisfatória e, ao mesmo tempo, simples e funcional.

1.3 Objetivos

Considerando a problemática apresentada, tem-se como principal objetivo a construção de uma plataforma de localização e geração de rotas para o Prédio das Engenharias Mecânica, Mecatrônica e Naval da Escola Politécnica da USP, que possa ser utilizada sem a necessidade de instalar um aplicativo, por qualquer indivíduo que necessite encontrar algum lugar específico.

O sistema deve ser simples, porém funcional. a fim de que tenha aderência facilitada pelo usuário. Assim, espera-se que a utilização de QR Codes seja eficaz para entregar informações de forma rápida e precisa.

2 ESTADO DA ARTE

2.1 Tecnologias utilizadas em IPS

A partir da popularização e evolução do GPS, diversas pesquisas têm sido realizadas na tentativa de criar um sistema similar a ele quanto à simplicidade, robustez e acessibilidade que funcione em ambientes internos. Durante o processo de criação deste trabalho, foram encontradas pesquisas datadas a partir de 1998 (MOEGLEIN; KRASNER, 1998), bem como gama elevada de artigos sobre a aplicação de tecnologias já consolidadas para a implementação de um IPS. Segundo o artigo publicado por (SILVA; SOSPEDRA; HUERTA, 2019), no qual foi realizado um panorama sobre IPS a partir de outros 62 estudos dos 5 anos anteriores, existe uma quantidade muito grande de publicações sobre o tópico, que variam em quantidade e meios de comunicação.

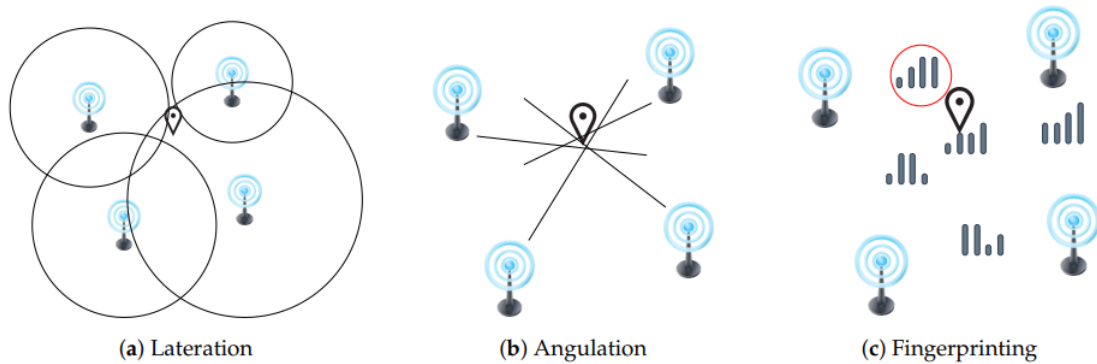
Algumas técnicas podem ser utilizadas para determinar o posicionamento em ambiente interno. As quatro mais comuns serão exploradas a seguir:

- a. **Tempo de chegada (*Time of arrival (TOA)*):** configura-se como sendo o tempo de um sinal recebido após a emissão. Sabendo a velocidade de propagação do sinal, é possível determinar a distância.
- b. **Diferença de tempo de chegada (*Time difference of arrival (TDOA)*):** mesmo que TOA, porém mede a diferença de distâncias entre cada emissor de sinal.
- c. **Ângulo de chegada (*Angle of arrival (AoA)*):** trata-se do ângulo que o sinal chega no sensor, sendo possível determinar a posição fixa.
- d. **Força do sinal recebido (*Received signal strength (RSS)*):** medição da intensidade que o sinal é recebido.

A combinação de tais técnicas dá origem ao que se chama de método, o qual é utilizado para determinar a posição estimada de um objeto. Os principais são:

- a. **Lateration:** TOA, TDOA e RSS, como já dito, são usados para estimar a distância até o sinal emitido. O conjunto de informações adquiridas por 3 ou mais sensores permitem determinar uma posição esperada do objeto (Figura 1a).
- b. **Angulation:** utiliza AoA a fim de determinar posição a partir do "cruzamento das retas" de ângulação (Figura 1b) do sinal obtido de vários sensores.
- c. **Fingerprinting:** por meio da técnica RSS, a intensidade dos sinais medidos a partir de algumas fontes é utilizada para estimar o quão distante o objeto está de cada uma (Figura 1c). Para tanto, esse processo é realizado em duas etapas; a primeira coleta de informações é utilizada como treino e a segunda utiliza este treino para melhor determinar a posição.

Figura 1: Métodos mais comuns usados no IPS



Fonte: (SILVA; SOSPEDRA; HUERTA, 2019)

Em meio às tecnologias que podem ser utilizadas para a criação de um IPS, discorrer-se-á sobre as mais relevantes.

2.1.1 Ultrassom

Os IPS baseados em ultrassom utilizam TOA ou TDOA na aplicação de *lateration* (OGUNTALA, 2018). Apesar do sistema acústico ter acurácia de alguns centímetros (NASSER, 2016), ele sofre com interferências e apresenta mudanças significativas conforme a velocidade do som é afetada pela temperatura e umidade. Além disso, requer hardware específico para a implementação, o que, por sua vez, mesmo sendo barato para espaços pequenos, acaba sendo inviável para lugares maiores (UREÑA, 2018).

Como referências de projetos utilizando a tecnologia de ultrassom pode-se citar *Dolphin* (FUKUJU, 2003) e *BeepBeep* (PENG, 2007).

2.1.2 WiFi ou WLAN ou IEEE 802.11

A tecnologia WLAN é utilizada em boa parte dos serviços de localização indoor existentes atualmente, tais como *The Indoor Lab*, uma plataforma que monitora pessoas dentro de um ambiente fechado, e *AnyPlace*, um aplicativo que tem como objetivo a navegação de um usuário no interior de um local.

No estudo (SILVA; SOSPEDRA; HUERTA, 2019), WiFi foi mencionado como tecnologia de suporte em todas as publicações analisadas; pode-se inferir, assim, que tal tecnologia é a mais difundida no meio de IPS.

Contudo, apesar do WLAN ser acessível, ele não é comumente utilizado de forma exclusiva para fins de posicionamento, o que pode acarretar travamentos e lentidão dependendo da qualidade da rede e da quantidade de usuários a utilizando.

2.1.3 BLE (Bluetooth Low Energy)

Os emissores BLE são conhecidos popularmente como beacons. São dispositivos pequenos, baratos e que consomem pouca bateria; além disso, são, mais acurados que WiFi, devido à maior quantidade difundida pelo local de uso (FARAGHER; HARLE, 2015).

No entanto, eles são mais lentos quando comparados a WLAN e, como são comprados e instalados separadamente, podem gerar um custo extra (RUTVIK, 2020).

As primeiras tentativas de utilizar beacons para criar um IPS são datadas em meados dos anos 2000 [(HALLBERG, 2020)(ANASTASI, 2003)(GWON, 2004)]; porém, a popularização veio em 2012, após o surgimento dos iBeacons, pela Apple.

2.1.4 Radio Frequency Identification (RFID)

A tecnologia RFID tem como principais componentes tags que armazenam algum tipo de informação, as quais podem ser ativas, possuindo baterias e emitindo sinais periodicamente, ou passivas, utilizando energia do leitor de sinal para transmitir a informação.

Apesar de ser uma tecnologia parecida com o BLE, o RFID sofre com várias limitações devido às questões de padronização de frequência de operação, infraestrutura adicional em técnicas de identificação de localização e natureza invasiva (KAUR, 2011). Além disso, faz-se necessária a instalação de várias tags na área de interesse, o que pode resultar em alto custo monetário e computacional.

Como exemplo de aplicação da tecnologia, tem-se mTag (KORHONEN, 2006), que tinha o intuito de realizar funções comparadas às do NFC atualmente.

2.2 QR Codes

2.2.1 Panorama geral

O QR Code, ou Quick Response Code, é uma invenção de 1994 da empresa japonesa Denso-Wave e surgiu como uma evolução dos códigos de barras. Em 2002, a leitura de QR codes foi introduzida como diferencial em alguns aparelhos de celulares e, a partir daí, muitas empresas também começaram a utilizar a tecnologia; no entanto, foi apenas em 2012 que passou-se a ver os tais códigos disseminados por todos os lugares (KASPERSKY, 2020). Desde então, o uso de QR Codes vem crescendo, cerca de 96% de 2018 para 2020 (BLUEBITE, 2021) e, no Brasil, visto que, em 2018, aproximadamente 80% da população com 10 anos ou mais tinha telefone celular (IBGE, 2020), pode-se dizer que o recurso tem se tornado cada vez mais acessível.

A principal crítica dos códigos de barras se encontra na limitação quanto a quantidade de informação que podem armazenar, cerca de 20 caracteres por vez. Os QR Codes, por sua vez, podem ser personalizados, tanto em capacidade de armazenamento (Figura 2) quanto em estilo (Figura 3). Sendo assim, eles podem ser considerados como uma tecnologia prática e versátil, visto que são gerados facilmente, devido terem código aberto (QRCODE.COM, 20-?), e podem ser utilizados para diversos fins, como levar o usuário a um site específico, autenticar contas, acessar a rede Wi-fi, enviar e receber informações de pagamento, dentre outros (KASPERSKY, 2020).

Figura 2: Tipos de Qr Codes quanto ao armazenamento



Fonte: (MARQUES, 2017)

Figura 3: Tipos de Qr Codes quanto ao estilo



Fonte: (KOZAK, 2020)

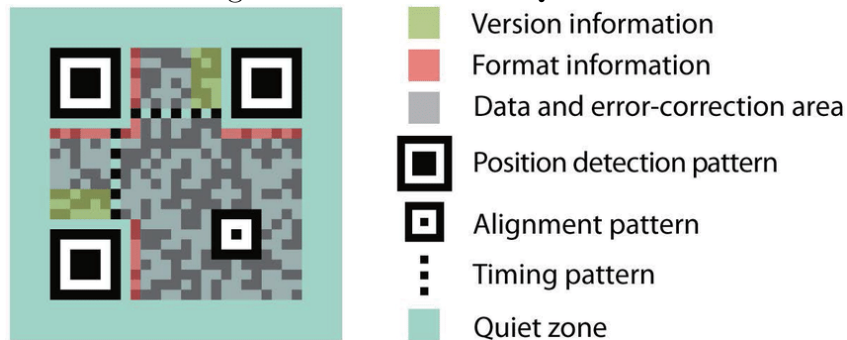
A estrutura do QR Code (Figura 4) foi pensada de modo a facilitar a usabilidade, podendo ser lido independente da sua orientação, além de ter redundâncias que, mesmo em caso de pequenos danos, ainda pode ser utilizado (MARQUES, 2017).

Assim como no código de barras tradicional, o QR Code funciona a partir da leitura sequencial de regiões claras e escuras (Figura 5) de cada linha, convertendo-as em bits (0 para claras e 1 para escuras); sendo assim, as informações contidas no código serão interpretadas após a leitura de todas as linhas do QR Code.

2.2.2 QR Codes e IPS

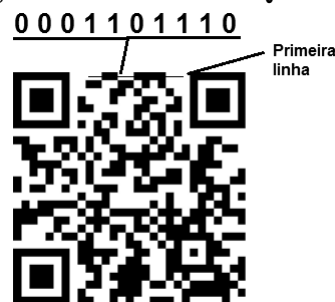
Mediante as facilidades de se trabalhar com QR Code, acredita-se que, para a proposta desejada, ele será uma opção satisfatória devido à simplicidade de manuseio, adaptabi-

Figura 4: Estrutura do QR Code



Fonte: (GAO; ZHAI; HU, 2015)

Figura 5: Leitura do QR Code



Fonte: (NCB, 2018)

lidade a diferentes ambientes, baixos custos de instalação e manutenção e flexibilidade de informações que podem ser transmitidas. Comparativamente com algumas tecnologias (Tabela 1), observa-se vantagem do QR Code perante as outras.

Tabela 1: Comparativo de diferentes tecnologias para ISP

Tecnologia	Acuracia	Custo de manutenção	Custo de instalação
GPS	baixa	alto	alto
Bluetooth	média	médio	alto
RFID	média	alto	alto
QR Code	alta	baixo	baixo

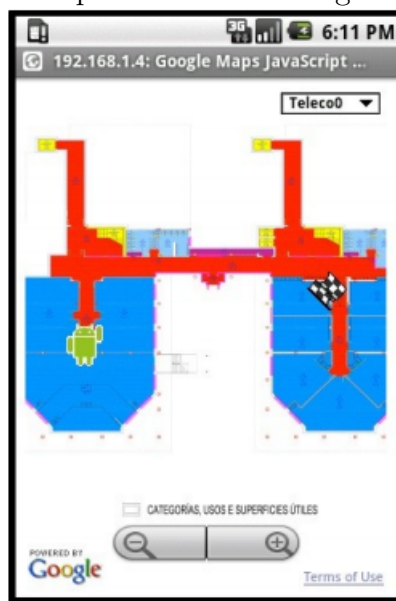
Fonte: (RAJ; TOLETY; IMMACULATE, 2013)

Uma vez que é necessário o escaneamento do QR Code para iniciar a navegação e, sabendo que o usuário precisa estar próximo ao código para isso, a posição inicial em que ele se encontrará tem acurácia alta; paralelamente, como a instalação configura-se apenas como a impressão de papéis e posterior posicionamento destes e a manutenção funcionaria

da mesma forma, pode ser dito que o custo é baixíssimo.

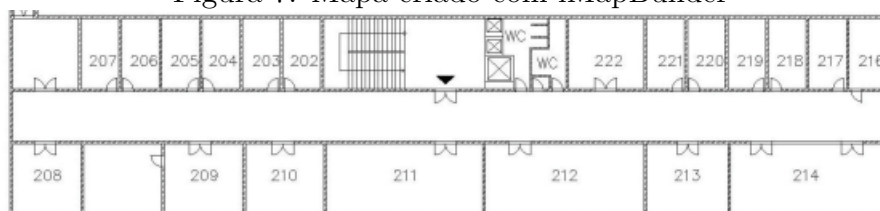
Apesar das vantagens observadas, nota-se disparidade em relação à quantidade de pesquisas que tem o QR Code como tecnologia base quando o assunto é IPS; da pesquisa ampla apresentada na seção anterior, ele não foi citado, e das publicações encontradas, menos de 10 envolviam o trabalho com smartphones e qr codes. Todavia, tais artigos se mostraram elucidativos, uma vez que apresentaram ferramentas de geração de mapas personalizados, como Google Maps API (Figura 6) e iMapBuilder (Figura 7), e, também, meios diferentes de implementação do sistema.

Figura 6: Mapa criado com Google Maps API



Fonte: (MONTENEGRO, 2014)

Figura 7: Mapa criado com iMapBuilder



Fonte: (ILKOVICOVÁ; ERDÉLYI; KOPÁCIK, 2014)

Em todos os casos, a implementação foi realizada em ambiente mobile; contudo, acredita-se que a necessidade de instalar um aplicativo no dispositivo celular seria uma barreira para o uso do sistema, já que, em se tratando de um prédio universitário o local

base do projeto, pedir informações poderia ser mais rápido e eficaz, o que diminuiria a adesão à plataforma. Sendo assim, acredita-se que a criação de um sistema web seria uma solução mais eficiente para o problema.

O âmbito mais complexo do projeto encontra-se na integração do posicionamento com a navegação do usuário, visto que não foram encontradas referências sobre tal processo. Nos artigos estudados, todos ressaltavam como passo futuro esta integração; contudo, nenhum deles chegou a implementá-la. A sugestão mencionada por (MONTENEGRO, 2014) se baseia na utilização da bússola e acelerômetro do smartphone a fim de promover a navegação.

3 METODOLOGIA

Para se definir do escopo do projeto, foram feitos dois experimentos, sendo eles:

- Pesquisa de Demanda e Destinos
- Teste de Funcionalidade

3.1 Pesquisa de Demanda e Destinos

A pesquisa de demanda e destinos foi feita via Formulário do Google e teve o objetivo se entender a demanda de pessoas que procuram algum local no Prédio das Engenharias Mecânica, Mecatrônica e Naval da POLI-USP e também quais lugares são procurados por essas pessoas. Com isso, pode-se definir quais necessidades o MeCaMaps precisa solucionar.

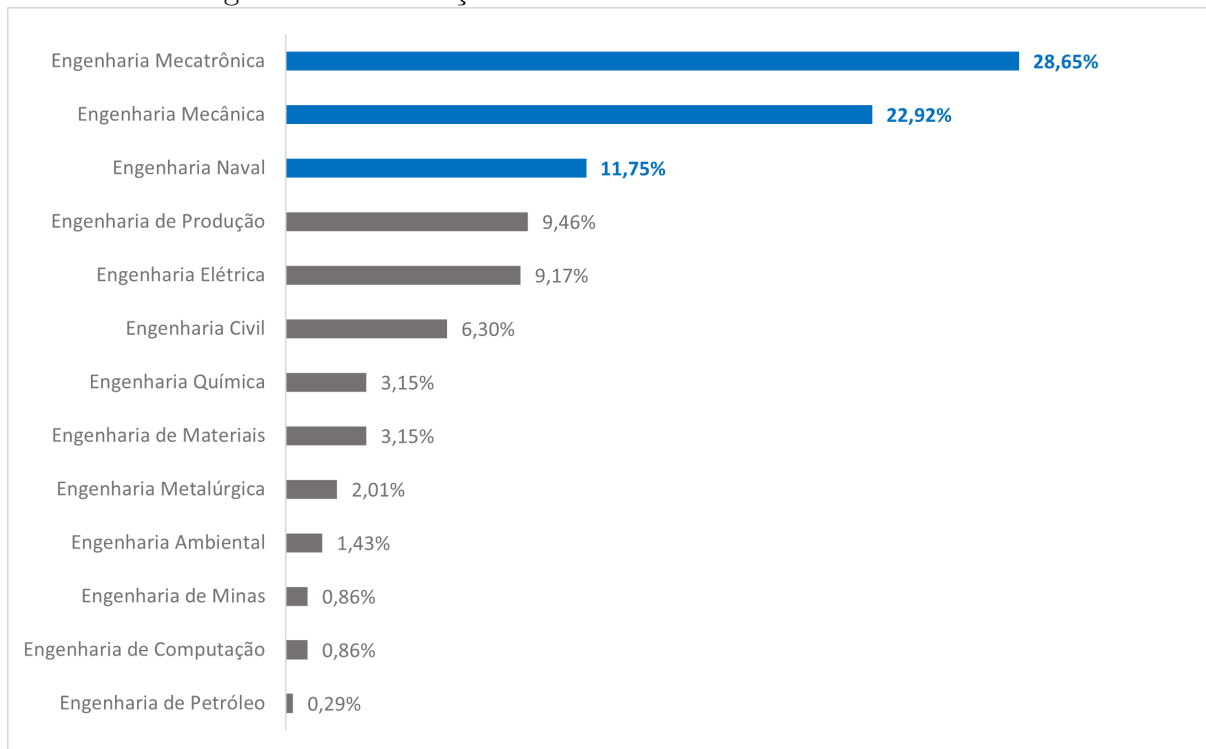
A consulta teve como público os alunos da Escola Politécnica da Universidade de São Paulo que ingressaram entre 2016 e 2020. Para atingir esse público, o link do formulário foi compartilhado em todos os grupos de sala de Whatsapp de todos os cursos da POLI.

Foram obtidas com 350 respostas, sendo 221 delas de alunos das engenharias Mecânica, Mecatrônica e Naval (Figura 8), ou seja, alunos que visitam o prédio com alta frequência.

3.1.1 Resultado da Pesquisa

As duas principais perguntas da pesquisa eram: "Você já precisou pedir informação para achar algo no prédio da Mecânica POLI-USP?" (Pergunta 1) e "O que você precisava encontrar ao chegar no prédio da Mecânica POLI-USP? Considere a informação que você tinha no momento que entrou no prédio." (Pergunta 2)

Figura 8: Distribuição dos alunos de acordo com seus cursos



Fonte: Autoria Própria

3.1.1.1 Pergunta 1

Dos 350 alunos que responderam a pesquisa, 321 deles já precisaram pedir informação para achar algo no prédio da Mecânica POLI-USP, mostrando, assim, que a dor que o MeCaMaps visa solucionar pode atingir parte relevante deste público.

3.1.1.2 Pergunta 2

A Figura 9 mostra as opções mais procuradas ao visitar o prédio.

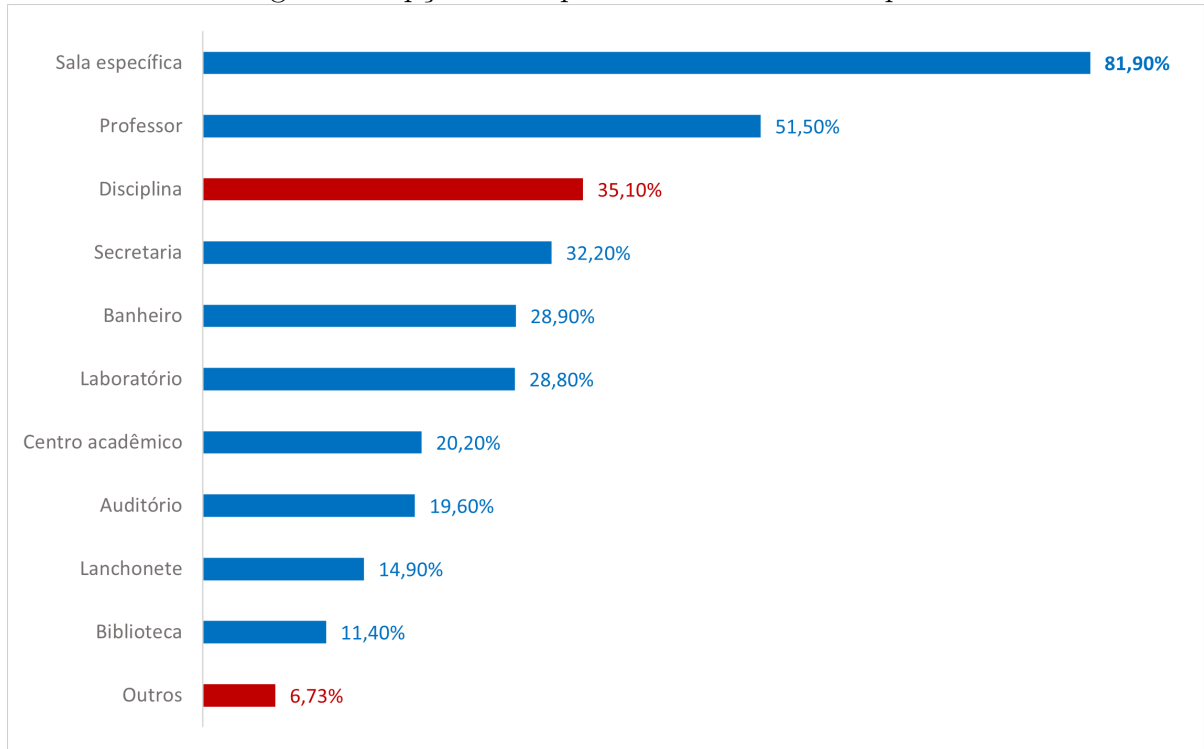
Dentro da opção "Outros", 95% das respostas foram "Grupo de Extensão". Com isso, essa categoria de destinos estará inclusa no site.

A opção "Disciplina" se refere a levar o usuário até a sala de determinada matéria da POLI. Como a relação sala-disciplina é variável ano a ano, essa categoria de destinos não será inclusa na plataforma.

A opção "Banheiro" possui mais de uma localidade no prédio e é assumido que seja algo simples de se encontrar, de forma que poucas pessoas abririam o celular para procurar essa opção. Dessa forma, essa opção de destino não estará inclusa no MeCaMaps.

Por fim, todas as demais opções estarão incluídas na lista de possíveis destinos do trabalho.

Figura 9: Opções mais procuradas ao visitar o prédio



Fonte: Autoria Própria

3.2 Teste de Funcionalidade

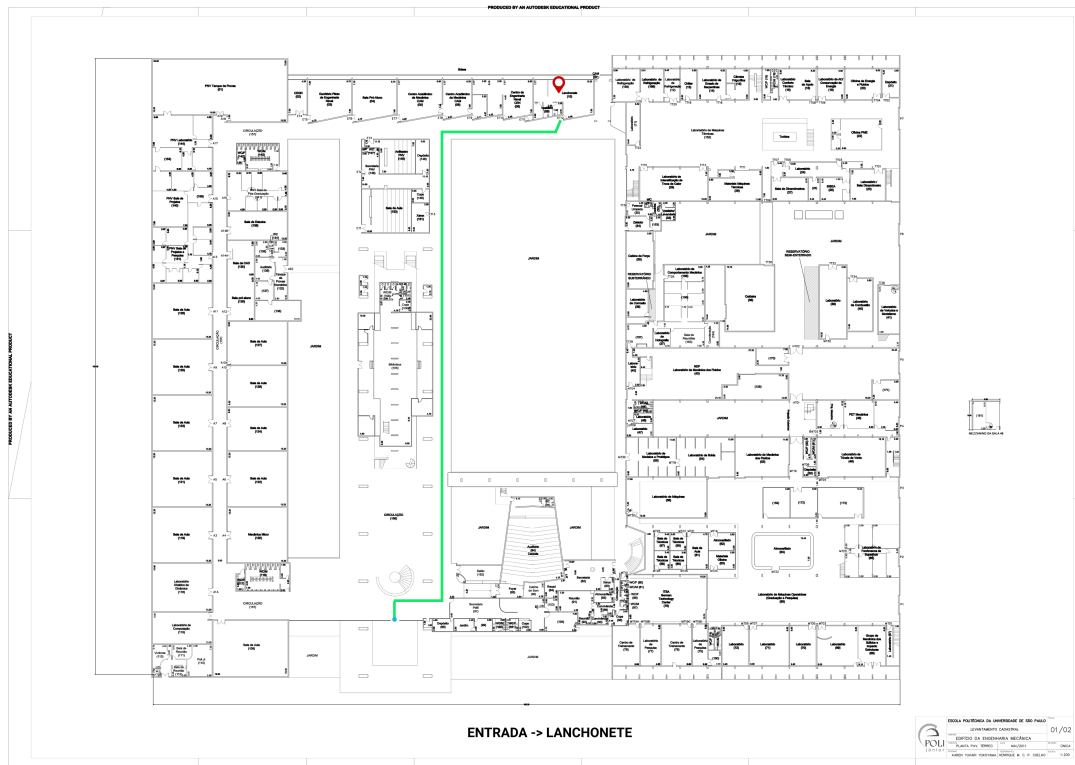
Visto que o objetivo do projeto visa o desenvolvimento de um sistema que solucione a localização dentro do prédio, questionou-se a necessidade de implementar localização em tempo real do usuário enquanto se move. Sendo assim, foi feito um teste de funcionalidade, no qual alguns indivíduos que nunca entraram no local tentarão se deslocar de um ponto a outro a partir de um mapa no celular com uma rota traçada e um pin vermelho no destino final.

Foram fornecidas três rotas, que foram nomeadas como fácil, médio e difícil, de acordo com a quantidade de corredores a se virar e com a necessidade de mudar de andar para chegar ao destino. Para facilitar o teste, o final de uma rota é o início da seguinte.

3.2.1 Caminhos

O caminho fácil (Figura 10) leva o indivíduo da entrada do prédio até a lanchonete, aos fundos. A linha verde tem o objetivo de guiá-lo do ponto azul, onde inicia o trajeto, até o ícone vermelho.

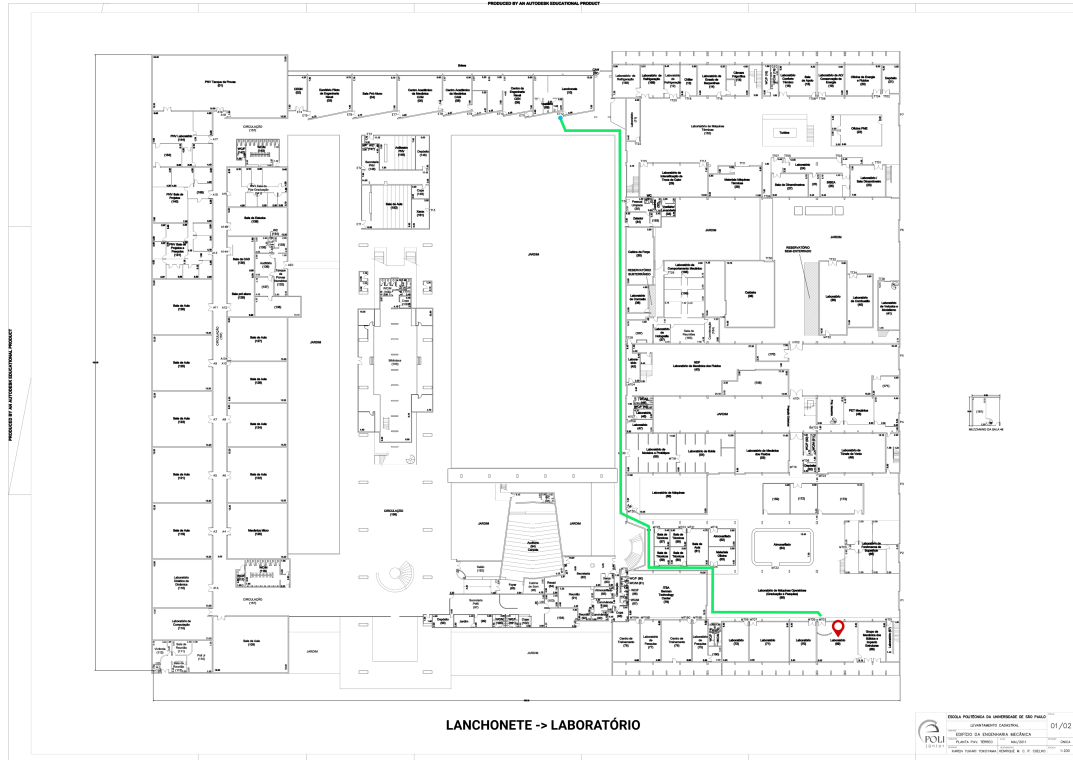
Figura 10: Caminho Fácil



Fonte: Autoria Própria

O caminho médio (Figura 11) inicia-se na lanchonete e vai até um laboratório na extremidade direita do prédio.

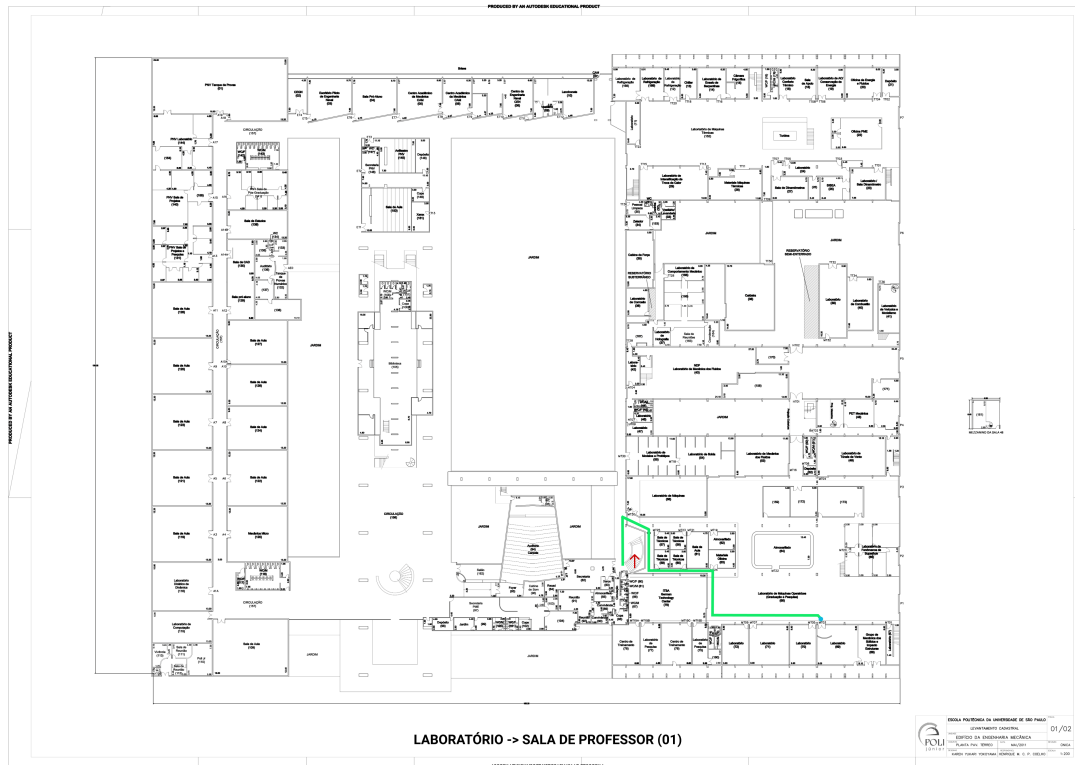
Figura 11: Caminho Médio



Fonte: Autoria Própria

O caminho difícil (Figuras 12,13) tem um elemento dificultador que é a troca de andares. Assim, o trajeto começa no laboratório e leva até a escada mais próxima, indicando a subida para o andar superior.

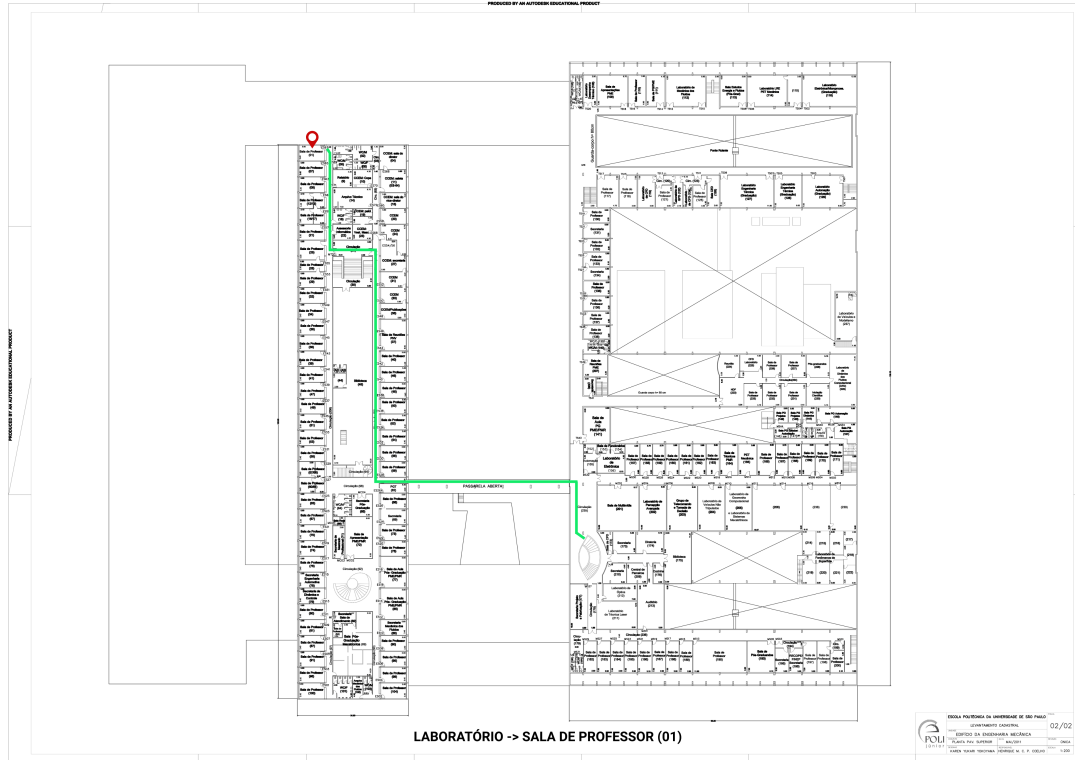
Figura 12: Caminho Difícil Térreo



Fonte: Autoria Própria

Depois de subir as escadas, o indivíduo é direcionado até uma sala de professores no porção esquerda no prédio.

Figura 13: Caminho Difícil Superior



Fonte: Autoria Própria

3.2.2 Resultados do teste

Os testes foram feitos com quatro pessoas, em junho de 2021. A amostragem ficou limitada devido ao cenário de pandemia da cidade de São Paulo.

Todos os participantes do teste conseguiram atingir seus destinos em todas as rotas propostas. Além disso, os quatro participantes relataram não terem tido dificuldades com o uso dos mapas.

Um dos participantes cometeu um erro durante a rota do Caminho Difícil, após passar pela passarela do segundo andar do prédio ele não virou no primeiro corredor à direita, conforme consta na rota, e virou no corredor seguinte à direita. Apesar disso, o participante percebeu rapidamente o erro e adaptou seu caminho, de forma que a chegada ao destino final da rota não foi prejudicada.

Com isso, conclui-se que a um mapa estático com uma rota traçada e uma indicação de destino final é funcional e resolve o problema de busca por salas, professores e laboratórios dentro do Prédio da Mecatrônica POLI-USP.

3.3 Requisitos do projeto

A partir dos da Pesquisa de Demanda e Destinos e do Teste de Funcionalidade, foram definidos os seguintes requisitos do projeto:

- Requisitos de Escopo
 1. O MeCaMaps deverá mostrar rotas traçadas em um mapa estático entre a localização do usuário e o destino desejado;
 2. O MeCaMaps deverá indicar no mapa a localização do usuário;
 3. O MeCaMaps deverá indicar no mapa o destino do usuário;
 4. O MeCaMaps deverá indicar no mapa a troca de andar, com uma cor diferente na rota e uma indicação em legenda;
 5. A lista de possíveis destinos do MeCaMaps deverá conter salas, professores (sala dos professores), laboratórios, grupos de extensão e outros (secretarias, biblioteca, auditório, lanchonete, etc)
- Requisitos de Experiência do Usuário

1. O Prédio das Engenharias Mecânica, Mecatrônica e Naval da Escola Politécnica da Universidade de São Paulo deverá ter QR codes instalados em diferentes localizações, para que os usuários possam escaneá-los;
2. A leitura de um QR code deverá redirecionar o usuário para a plataforma do MeCaMaps com a sua localização já definida, a partir do QR code lido;
3. O MeCaMaps não deverá exigir a instalação de um aplicativo no celular do usuário, mas deverá dar essa possibilidade a ele;
4. O MeCaMaps deverá possibilitar que o usuário atualize a sua localização durante a rota, manualmente ou com a leitura de outro QR code

3.4 Casos de uso

Para a funcionalidade da plataforma, foram espalhados pelo prédio da Mecânica POLI-USP vários QR Codes acompanhados do texto "Perdido no Prédio?". Os QR Codes serão usados como ferramenta para se fazer a localização do usuário no prédio. A plataforma contará com um dicionário de localização exata de cada QR Code. Para a geração das rotas, será considerado que o usuário se encontra exatamente na posição do QR escaneado.

O site conta com duas páginas, o Menu Principal e a Página do Mapa.

O fluxo considerado como principal do site, por ser mais intuitivo para o usuário, é aquele onde a pessoa vai escanear o QR Code antes mesmo de entrar no site do projeto. Com isso, o usuário será redirecionado para o Menu Principal, com sua Localização atual já definida (a partir do QR Code lido).

Caso o usuário opte por entrar na plataforma, ele irá visualizar o Menu Principal sem a localização atual definida.

Neste capítulo vamos detalhar como se dá a navegação do usuário da plataforma.

3.4.1 Menu Principal

O Menu Principal possui cinco funcionalidades, sendo elas:

- a. Visualizar Mapa: Botão
- b. Selecione onde está: Caixa de Seleção
- c. Selecione seu destino: Caixa de Seleção

d. Gerar Rota: Botão

e. Fale conosco: Botão

3.4.1.1 Visualizar mapa

Esta função é um botão, onde o usuário clica e é redirecionado para a Página do Mapa.

3.4.1.2 Selecione onde está

Esta função é uma caixa de seleção onde o usuário vai informar ao sistema qual é a sua posição dentro do Prédio da Mecatrônica POLI-USP, com duas opções para isso, o QR Code e a Seleção Manual.

A opção do QR Code é um botão com o ícone de uma câmera, onde o usuário clica e o site abre a câmera do celular para escanear o QR Code.

A Seleção Manual é feita em uma lista que conta com salas, laboratórios, professores (salas dos professores), grupos de extensão e outros (secretarias, centros acadêmicos, etc).

3.4.1.3 Selecione seu destino

Esta função é uma caixa de seleção onde o usuário vai informar ao sistema qual é o seu destino desejado. A lista de destinos conta com Salas, Laboratórios, Professores (salas dos professores), Grupos de Extensão e Outros (Secretarias, Centros Acadêmicos, etc).

3.4.1.4 Gerar Rota

Esta função é um botão onde só é possível se clicar quando as funções "Selecione onde está" e "Selecione seu destino" estiverem preenchidas. Ao clicar no botão, o usuário é direcionado para a página do mapa com a sua rota traçada.

3.4.1.5 Fale conosco

Esta função é um botão onde, ao clicar, o usuário é direcionado para um Formulário do Google. Neste formulário podem ser enviadas sugestões de melhorias, críticas, elogios e relatos de possíveis erros.

3.4.2 Página do Mapa

A Página do Mapa possui quatro funcionalidades, sendo elas:

- a. Menu Principal: Botão
- b. Trocar Andar: Botão
- c. Selecione onde está: Caixa de Seleção
- d. Selecione seu destino: Caixa de Seleção

Além disso, a página conta com uma legenda das cores e ícones do mapa e com uma distância estimada do trajeto.

3.4.2.1 Menu Principal

Ao clicar na função "Menu Principal" o usuário será redirecionado para o Menu Principal do site.

3.4.2.2 Trocar andar

Ao clicar na função "Troca de andar", o mapa exibido vai mudar para o andar oposto ao andar atual. Os andares disponíveis são o térreo e o superior, sendo que o térreo será o andar padrão da plataforma. Caso o usuário chegue até a Página do Mapa com sua localização já definida, o site vai mostrar primeiro o andar referente a essa localização.

3.4.2.3 Selecione onde está

Esta função é uma caixa de seleção que funciona da mesma maneira que a função do Menu Principal. Na Página do Mapa ela poderá servir para o usuário atualizar a sua localização durante o trajeto, caso encontre outro QR Code no caminho ou caso opte por atualizar manualmente.

Quando esta função estiver preenchida, um ícone de um boneco amarelo aparece no mapa, indicando onde o usuário está

3.4.2.4 Selecione seu destino

Esta função é uma caixa de seleção que funciona da mesma maneira que a função "Selecione seu destino" no Menu Principal.

Quando esta função estiver preenchida, um ícone de um pin vermelho aparece no mapa, indicando onde está o destino do usuário.

3.5 Prototipagem

3.5.1 Telas da plataforma

A fim de nortear o desenvolvimento da parte gráfica do projeto, criou-se algumas telas mostrando o funcionamento da plataforma. Assim, ao acessar o MeCaMaps, o usuário poderá ver alguma das telas mostradas a seguir. A primeira refere-se ao menu principal, onde serão mostrados os botões que o levarão a gerar a rota até o destino desejado.

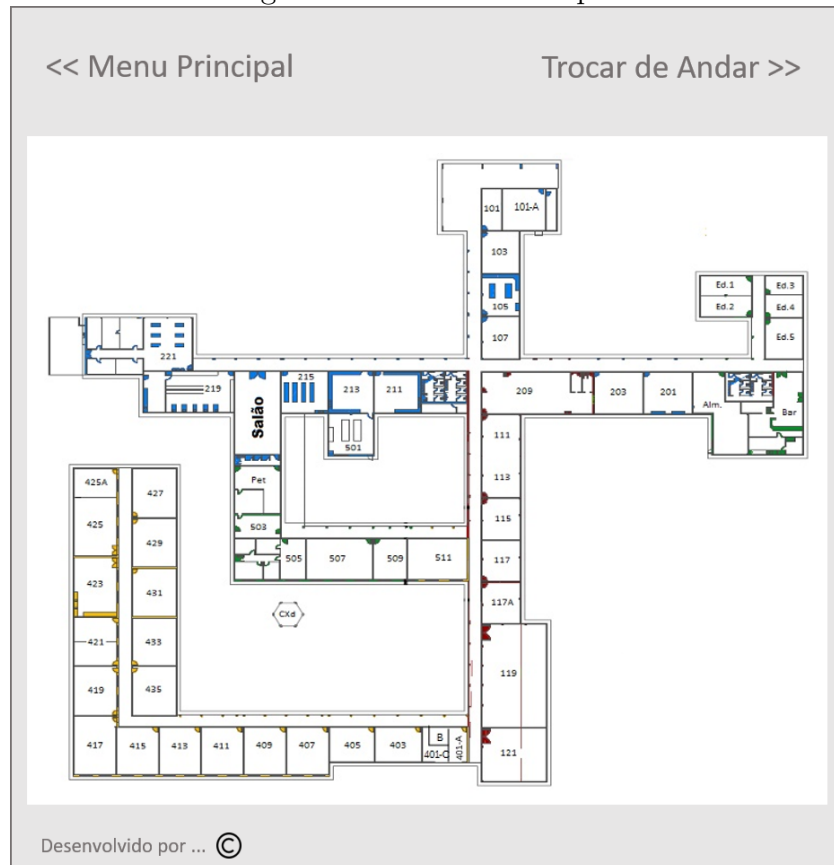
Figura 14: Menu Principal



Fonte: Autoria Própria

A segunda mostra a visualização ao clicar no botão "Visualizar Mapa". Nota-se a presença do botão "Trocar de andar", que permitirá a mudança para o andar superior do prédio e vice-versa e do botão "Menu Principal", que promove o retorno à tela que é mostrada na Figura 14.

Figura 15: Visualizar Mapa



Fonte: Autoria Própria

A terceira refere-se à seleção de uma opção. A tela seria a mesma para selecionar destino. O botão com o ícone de uma câmera permite escanear o QR Code da posição em que o usuário se encontra. O usuário poderia mudar as opções, bem atualizar sua localização atual ao escanear outro QR Code.

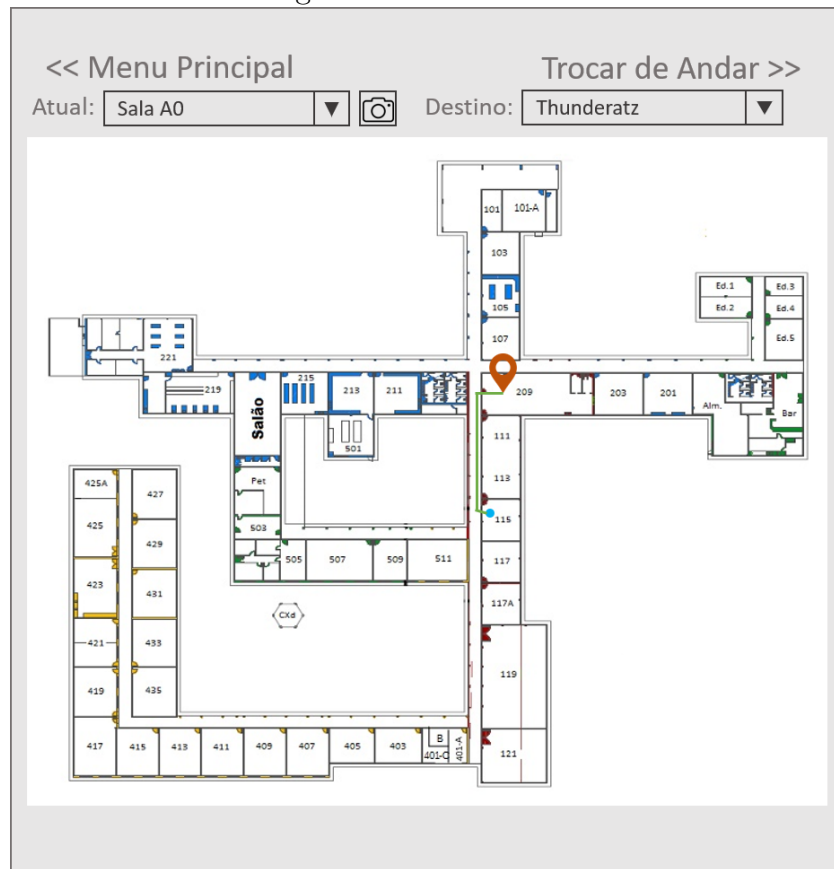
Figura 16: Selecionar Opção



Fonte: Autoria Própria

A quarta mostra a funcionalidade principal da plataforma, a geração de rota. O ponto azul seria a localização atual e o ícone laranja, o destino.

Figura 17: Gerar Rota



Fonte: Autoria Própria

3.6 Desenvolvimento

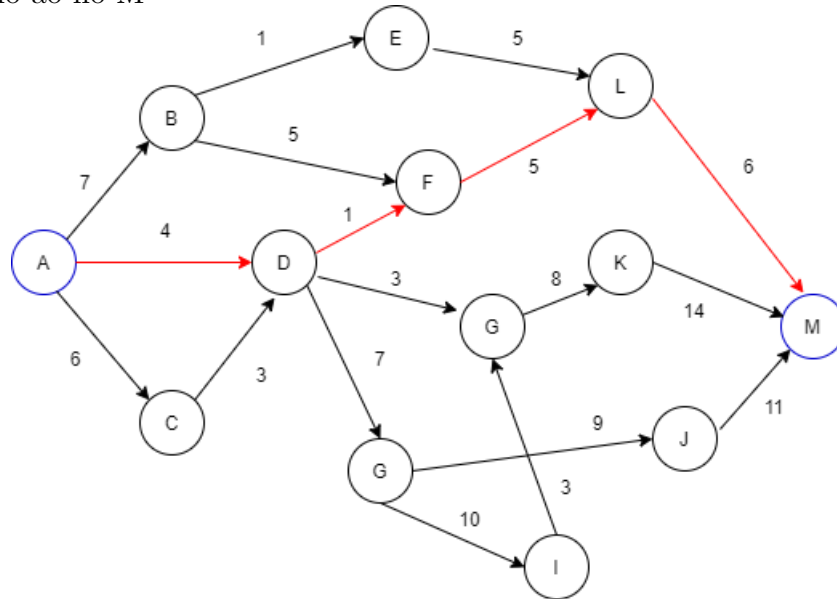
3.6.1 Algoritmo de Dijkstra

Para este projeto, optou-se pela utilização do algoritmo de Dijkstra (DIJKSTRA, 1959), o qual visa encontrar o menor caminho entre dois nós de um grafo (Figura 20), o que, neste trabalho, representa a rede de caminhos no prédio. Cada um dos nós corresponde a um ponto selecionável no mapa, ou melhor, a um QR Code ou a um ponto fictício que serve para promover o deslocamento por caminhos possíveis.

Existem algumas variantes do algoritmo, sendo a versão original encontrar o menor caminho dado dois nós; no entanto, a mais comum fixa um único nó como origem e encontra o caminho deste para todos os outros no grafo, gerando, assim, uma árvore de menores caminhos.

O Algoritmo pode ser desenvolvido da seguinte forma:

Figura 18: Exemplo de grafo simples. Destaque para o menor caminho, com saída do nó A com destino ao nó M



Fonte: Autoria Própria

Seja o nó inicial o nó em que se começa e a distância do nó Y a distância do nó inicial ao nó Y. O algoritmo de Dijkstra atribuirá alguns valores de distância iniciais e tentará melhorá-los passo a passo:

1. Marca-se todos os nós não visitados e cria-se um conjunto com todos eles, denominado conjunto não visitado;
2. Atribui-se a cada nó um valor de distância provisório: zero para nosso nó inicial e infinito para todos os outros nós. Define-se o nó inicial como atual;
3. Para o nó atual, considera-se todos os seus vizinhos não visitados e calcula-se suas distâncias provisórias até o nó atual. Compara-se a distância provisória recém-calculada com o valor atribuído atual e o menor é mantido. Por exemplo, se o nó atual A está marcado com uma distância de 6, e a aresta que o conecta a um vizinho B tem comprimento 2, então a distância de B a A será $6 + 2 = 8$. Se B foi previamente marcado com uma distância maior que 8, então muda-se para 8. Caso contrário, o valor atual será mantido;
4. Quando se termina de considerar todos os vizinhos não visitados do nó atual, este é marcado como visitado e removido do conjunto não visitado. Um nó visitado nunca será verificado novamente;

5. Se o nó de destino foi marcado como visitado (ao planejar uma rota entre dois nós específicos) ou se a menor distância provisória entre os nós no conjunto não visitado é infinita (ocorre quando não há conexão entre o nó inicial e os nós não visitados restantes), então o algoritmo termina;
6. Caso contrário, seleciona-se o nó não visitado que está marcado com a menor distância provisória, define-se como o novo "nó atual" e volta-se para a etapa 3.

Ao planejar uma rota, na verdade não é necessário esperar até que o nó de destino seja "visitado" como acima: o algoritmo pode parar uma vez que o nó de destino tem a menor distância provisória entre todos os nós "não visitados" (e, portanto, pode ser selecionado como o próximo "atual").

3.6.2 Progressive Web App (PWA)

Com o objetivo de melhorar a experiência do usuário, optou-se pelo desenvolvimento de um Progressive Web App (PWA), que se trata de uma aplicação híbrida entre Web e Mobile. O PWA proporciona a mesma experiência que se tem pelo browser, porém sem nenhuma informação em tela além da aplicação, ou seja, toda interface do navegador, como barra de endereço, botões, favoritos, etc, são removidos (DOMES, 2017).

As vantagens do PWA quanto à experiência do usuário podem ser resumidas em quatro fatores:

- Praticidade: como não há necessidade da realização do download, essa solução é um facilitador para o usuário. Visto que a plataforma não seria utilizada de forma recorrente por um mesmo indivíduo, acredita-se que a obrigatoriedade de se instalar um aplicativo faça com que ele acabe desistindo de utilizar o serviço;
- Agilidade: não será necessário esperar até que o arquivo seja baixado para completar sua jornada. Basta que escaneie o QR Code ou acesse o site, preencha algumas informações rápidas e o trajeto está pronto para ser seguido;
- Acessibilidade: o acesso a aplicações progressivas é outro elemento facilitado. Ele funciona com quaisquer tipos de conexões, sejam elas fixas, móveis e até mesmo offline. Além disso, o PWA é multiplataforma, podendo ser utilizado tanto em celulares Android quanto IOS ou, até mesmo, pelo navegador de um computador;

- Responsividade: como possuem design responsivo, adaptam-se a diferentes tipos de tela. Isso inclui tablets, smartphones, notebooks, desktops e outros tipos de aparelhos

Sendo assim, o PWA é uma aplicação web com tecnologias que permitem se ter experiência de uso muito próxima da oferecida pelos mobile apps. As funcionalidades que estas tecnologias permitem são:

- Poucas alterações no código do site;
- Utilização de HTML/CSS/Javascript;
- Acesso à API's nativas como geolocalização, câmera, microfone, etc;
- Envio de notificações push;
- Aplicação muito leve (menos de 1MB geralmente);
- Suporte à utilização offline;
- Possibilidade de adicionar um ícone de acesso rápido na tela inicial do dispositivo.

3.6.3 React

O React é a biblioteca mais popular do JavaScript e é usada para construir uma interface de usuário (IU), oferecendo uma resposta excelente para o desenvolvedor adicionar comandos usando um novo método de renderizar sites (DOMES, 2017). Optou-se por trabalhar com React, visto que apresenta, como principais vantagens:

- Facilidade de uso: por se tratar de uma biblioteca de código aberto para interfaces gráficas (GUI), tem como foco em tornar a experiência do usuário com a interface mais eficiente;
- Suporte à reusabilidade de componentes: permite o reuso de componentes que tenham sido desenvolvidos em outras aplicações e que usem a mesma função;
- Componentes com facilidade de escrita: também conhecido como React JS, o componente do React é fácil de escrever porque ele usa JSX, uma extensão de sintaxe opcional para JavaScript. Ela permite a combinação de HTML com JavaScript e simplifica a estrutura de codificação escrita de um site, além de fazer com que a renderização de múltiplas funções seja mais fácil (BODUCH, 2017);

- Melhor desempenho com Virtual DOM: melhora o processo de atualização do DOM (Document Object Model) - é uma interface que representa como os documentos HTML e XML são lidos pelo browser -, o qual pode causar muita frustração em projetos baseados em aplicações web. O React usa Virtual DOMs, o que permite contornar esses problemas, uma vez que quando ocorrer uma mudança no DOM real, o virtual se modifica imediatamente. O sistema vai impedir que o DOM real faça atualizações constantemente, o que acarreta a manutenção da velocidade da aplicação.

4 RESULTADOS

4.1 Organização do Projeto

O projeto consiste em uma aplicação web. Todavia, pelo fato da tecnologia PWA, pode se comportar visualmente como uma aplicação android ou IOS se instalado no dispositivo. Devido à especificidade da plataforma - gerar rotas para um único prédio, com quantidade de caminhos relativamente pequena -, foi possível desenvolvê-la toda no *client-side*.

A fim de melhor apresentar a arquitetura do projeto, utilizar-se-a o modelo C4 (BROWN,), que divide a estrutura em quatro camadas hierárquicas:

1. **Contexto:** camada maior de abstração, descreve do que se trata a aplicação de uma visão macro. No caso, um software para geração de rotas;
2. **Containers:** representa uma aplicação. Para este projeto, uma aplicação web client-side que gera rotas para o prédio da Engenharia Mecatrônica da Poli-USP;
3. **Componentes:** grupo de funcionalidades ou função específica que permitem uma interface funcionar. Neste caso, os componentes react para cada uma das três páginas da plataforma;
4. **Códigos:** especificidades de cada componente a nível código. Para mais detalhes quanto a este tópico: [MeCaMaps](#)

4.1.1 Componentes

A página *Main* é a primeira a ser vista ao acesar o site. Nela, encontramos os seguintes componentes:

1. **Dropdown**

- Descrição: utilizado para exibir a lista suspensa com as opção de seleção. Ele possui a relação de todos os pontos do mapa, bem como a categoria que eles pertencem;
- Propriedades:
 - (a) *placeholder*: indentificação se é uma lista de origem ou de destino;
 - (b) *place*: variável que armazena a opção selecionada;
 - (c) *setPlace*: função que muda o valor armazenado pela variável anterior;
- Retorno: renderiza uma lista suspensa na tela. Ao clicar em uma opção, esta fica destacada em azul e persiste ao trocar de página

2. **qr_correlations**

- Descrição: variável funcional que armazena um dicionário contendo a relação entre o código de pontos no mapa e o nome dos lugares onde existem QR Codes instalados no prédio. Encontra-se na pasta de componentes *Floor*.

Em seguida, após o usuário preencher as informações de localização solicitadas, a página *RouteGenerator* é o próximo destino. Os seguintes componentes são utilizados:

1. **grafo**

- Descrição: classe responsável por criar a rede de caminhos do algoritmo de Dijkstra e, posteriormente, calcular a menor rota de um ponto a outro. Ao ser iniciado, cria um dicionário de distâncias e outro que guarda o caminho percorrido pelo algoritmo. Encontra-se na pasta de componentes *Dijkstra*.

2. **Floor**

- Descrição: recebe propriedades de posicionamento dos pinos, andar da posição inicial ou andar atual e o vetor de linhas do trajeto, e decide qual dos andares vai ser mostrado e onde os pins estarão localizados;
- Propriedades:
 - (a) *showMan*: representa a relação entre o andar atual e a posição de origem. Se forem no mesmo andar, o boneco amarelo é mostrado no mapa;
 - (b) *showPin*: representa a relação entre o andar atual e a posição de destino. Se forem no mesmo andar, o pin vermelho é mostrado no mapa;

- (c) *coordMan*: vetor de coordenadas da posição de origem; utilizado para posicionar o boneco amarelo;
- (d) *coordPin*: vetor de coordenadas da posição de destino; utilizado para posicionar o pin vermelho;
- (e) *place*: referência à imagem do andar que deve ser mostrado;
- (f) *color*: vetor de linhas do trajeto para serem coloridas;
- (g) *option*: andar atual.
- Retorno: renderiza o mapa do andares e as rotas, bem como os pins de localização.

3. correlations

- Descrição: variável funcional que armazena um dicionário contendo a relação entre o código de pontos no mapa e o nome dos lugares disponíveis para seleção. Encontra-se na pasta de componentes *Paths*.

4. coords

- Descrição: variável funcional que armazena um dicionário contendo a relação entre o código de pontos no mapa e suas respectivas coordenadas. Encontra-se na pasta de componentes *Paths*.

5. Man_pin

- Descrição: svg funcional que mandem o código que gera o boneco amarelo;
- Propriedades:
 - (a) *floor*: indica qual o andar atual;
 - (b) *coordMan*: vetor de coordenadas da posição de origem; utilizado para posicionar o boneco amarelo.
- Retorno: renderiza o boneco sobre as rotas do mapa.

6. Pin

- Descrição: svg funcional que mandem o código que gera o pin vermelho;
- Propriedades:
 - (a) *floor*: indica qual o andar atual;
 - (b) *coordMan*: vetor de coordenadas da posição de destino; utilizado para posicionar o pin vermelho.

- Retorno: renderiza o pin sobre as rotas do mapa.

7. **Routes_zero**

- Descrição: svg funcional que mandem o código que gera as rotas do andar térreo;
- Propriedades:
 - (a) *color*: vetor com as linhas a serem coloridas no mapa;
- Retorno: renderiza as linhas do térreo da rota selecionada, de acordo com o vetor color.

8. **Routes_first**

- Descrição: svg funcional que mandem o código que gera as rotas do primeiro andar;
- Propriedades:
 - (a) *color*: vetor com as linhas a serem coloridas no mapa;
- Retorno: renderiza as linhas do primeiro andar da rota selecionada, de acordo com o vetor color.

Vale dizer que os componentes *Dropdown* e *qr_correlations* também são utilizados nesta página.

Por fim, outro caso de uso decorre da utilização do QR Code para atualizar a posição de origem. O seguinte componente se apresenta:

1. **QrReader**

- Descrição: componente utilizado para a leitura dos QR Codes. Ao clicar no botão da câmera, uma página com um quadrado vermelho aparece na tela; ao encontrar um QR Code válido, retorna para a página anterior com a opção atual alterada;
- Propriedades:
 - (a) *onScan*: ao escanear um QR Code, uma função é chamada para alterar o valor da posição atual;
 - (b) *onError*: caso a leitura não funcionar, o erro é tratado;

(c) *facingMode*: opção que permite escolher qual das câmeras do dispositivo será utilizada por padrão, no caso, a traseira.

- Retorno: renderiza uma lista suspensa na tela. Ao clicar em uma opção, esta fica destacada em azul e persiste ao trocar de página

4.2 Rotas

4.2.1 Esqueleto de caminhos

O esqueleto de caminhos foi feito no AutoCAD e inserido na plataforma em um arquivo .svg transcrito para JSX. O esqueleto foi feito ponto a ponto nos corredores do mapa. Por exemplo, no corredor de salas 1, 2, 3...., existe uma linha entre cada sala, de forma que cada linha poderá ser colorida ou não de acordo com a rota do usuário.

4.2.2 Geração de Rotas

Para a geração de rotas, foram nomeados todos os pontos do esqueleto de caminhos, ao todo o esqueleto possui 594 pontos interligados por 297 linhas. Foi criado um dicionário para relacionar os pontos com as opções disponíveis nas caixas "Selecione onde está" e "Selecione seu destino". Com isso, uma vez que elas estiverem preenchidas, a plataforma interpreta que o usuário deseja ir do ponto A ao ponto B, por exemplo.

A quantidade de lugares nomeados por categoria está indicada na Tabela 2. Para se adicionar um novo lugar ao site, é necessário adicionar uma linha ao dicionário que relaciona os lugares aos pontos do esqueleto de caminhos, indicando em qual ponto o novo lugar se encontra.

Tabela 2: Quantidade de lugares nomeados por categoria

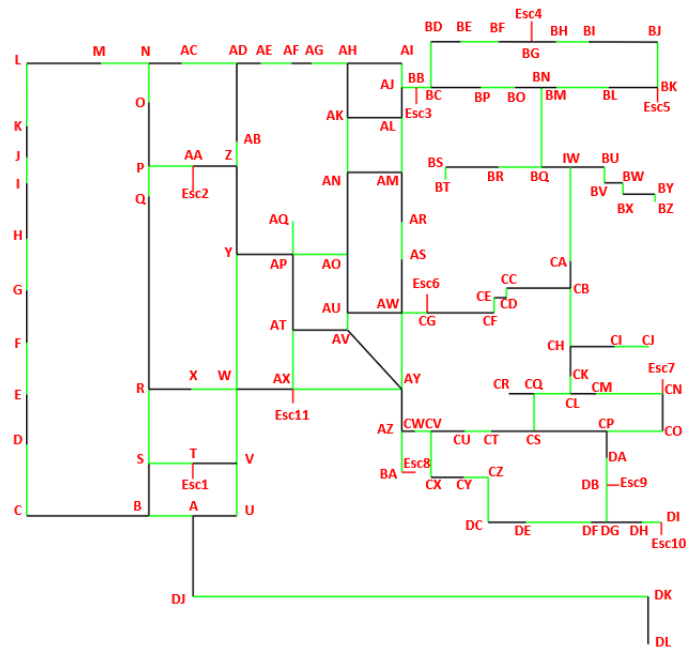
Categoria	Quantidade de lugares
Salas	234
Sala de professores	94
Laboratórios	34
Grupos de extensão	9
Outros	16

A partir dessa nomeação, foram criados os seguintes arquivos:

- a. Dicionário de coordenadas dos pontos;

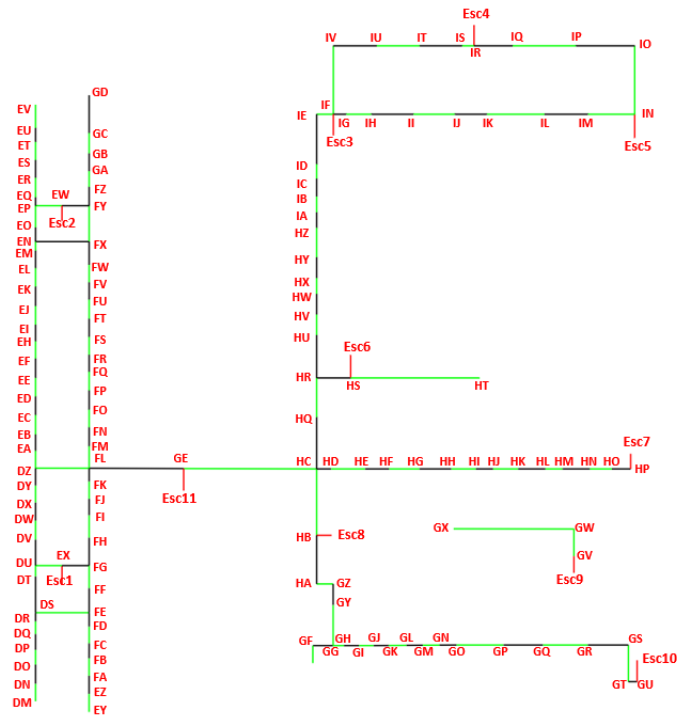
- b. Dicionário de ligações e distâncias entre pontos;
- c. Esqueleto de Caminhos em JSX com linhas nomeadas.

Figura 19: Pontos térreo



Fonte: Autoria Própria

Figura 20: Pontos superior



Fonte: Autoria Própria

4.2.2.1 Dicionário de Coordenadas de Pontos

Este dicionário relaciona cada um dos pontos nomeados com a sua respectiva coordenada no plano XY do AutoCAD, no seguinte formato: A: [x, y].

Com isso, a partir das seleções de posição atual e destino, os ícones do boneco amarelo e do pin vermelho são posicionados no mapa.

4.2.2.2 Dicionário de ligações e distâncias de pontos

Este dicionário relaciona cada ponto com os pontos adjacentes a ele, incluindo, também, a distância entre eles e um rótulo, que será usado para identificar, no componente de rotas, quais linhas deverão ser mostradas. Apresenta-se no seguinte formato:

"A": {"B": [123, "A-B"], "C": [456, "A-C"]}

Onde o ponto A se liga com os pontos B e C, com as distâncias 123 e 456, e pelas linhas "A-B" e "A-C" respectivamente.

A nomeação de linhas foi padronizada em ordem alfabética, para que o algoritmo entenda que deve mostrar a mesma linha ao ir de A para B ou de B para A, por exemplo.

Assim, tanto o caminho de A para B quanto de B para A se chama "A-B".

A geração de rotas é feita com o Algoritmo de Dijkstra, que retorna o vetor de linhas do trajeto contendo todas as linhas por onde o usuário vai passar para chegar ao seu destino.

4.2.2.3 Esqueleto de caminhos em JSX com linhas nomeadas

No arquivo JSX do Esqueleto de caminhos, cada linha possui um parâmetro "stroke" que determina a sua cor. Além disso, também foi adicionado o parâmetro "strokeOpacity", que determina a sua opacidade.

Dessa forma, dentro desses dois parâmetros foi adicionado uma condicional, na qual se a linha estiver dentro do vetor de linhas do trajeto, ela será colorida e opaca (visível, neste caso), caso contrário a linha será transparente - ou seja, opacidade zero. A coloração das linhas é azul, com exceção das linhas troca de andar, que são vermelhas.

Um exemplo pode ser visto na Figura 21, para a linha "B-S", onde "color" é o vetor de linhas do trajeto.

Figura 21: Exemplo de linha do Esqueleto de caminhos em JSX com a condicional de cor e opacidade

```
<path
  fill="none"
  stroke={props.color.includes('B-S') ? color_path : color_test}
  strokeOpacity={props.color.includes('B-S') ? 1 : visible}
  strokeWidth={stroke}
  d="M282.2 378.42L282.2 409.92"
/>
```

Fonte: Autoria Própria

Após a verificação se as linhas estão contidas no vetor de trajeto, a rota é traçada sobre o mapa.

4.2.3 Mapas

Os mapas foram inseridos na plataforma em arquivos .png atrás do Esqueleto de caminhos e dos pins. As configurações de estilo do css permitem definir a sobreposição a partir do parâmetro "z-index", iniciando por zero. Assim, os mapas tem "z-index" igual a zero, os caminhos igual a um, e os pins igual a dois.

4.2.4 QR Codes

Foi aprovada pela CAEMEN uma solicitação de intervenção no prédio da Mecatrônica POLI-USP para a implantação dos 34 QR Codes, nas dimensões 15x15cm. Os qr codes foram instalados com o layout da Figura 22

Figura 22: QR Code Portaria

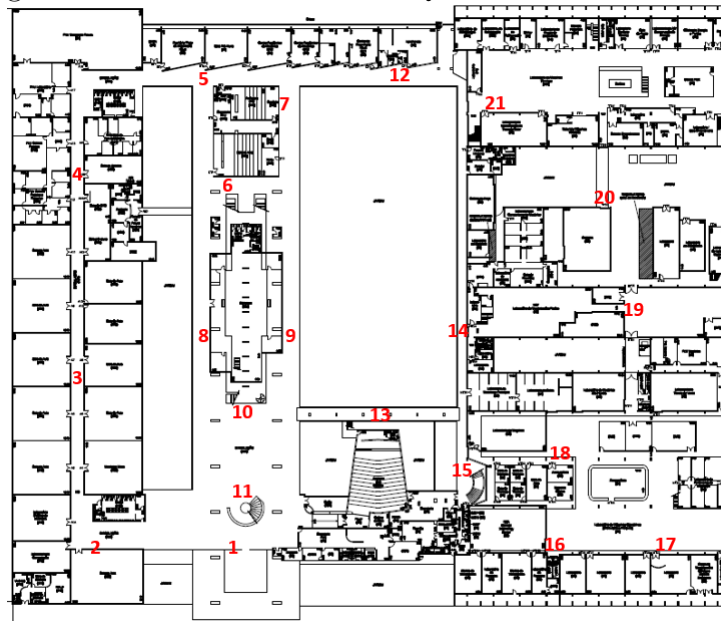


Fonte: Autoria Própria

Cada QR Code carrega o link da plataforma com o respectivo ponto do Esqueleto de caminhos, por exemplo A, B e C.

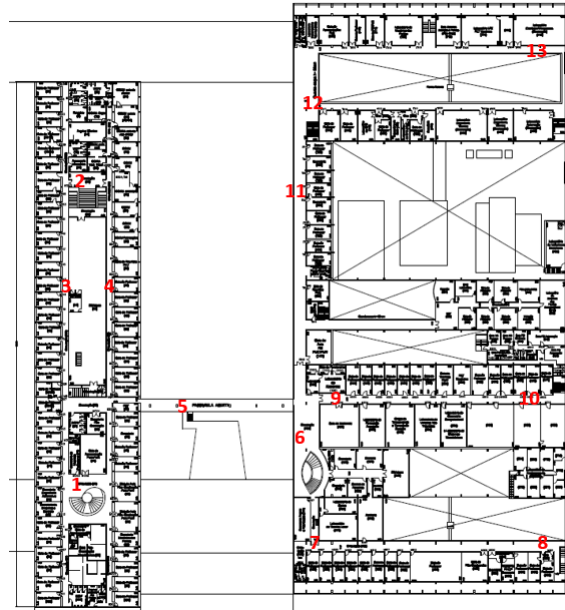
Os QR Codes foram posicionados como se segue:

Figura 23: Posicionamento dos QR Codes - Andar Térreo



Fonte: Autoria Própria

Figura 24: Posicionamento dos QR Codes - Andar Superior



Fonte: Autoria Própria

- Térreo

1. Portaria
2. Sala A0
3. Sala A4
4. Sala A14
5. Sala ET6
6. Sala ET1
7. CAM
8. Biblioteca Lado Esquerdo
9. Biblioteca Lado Direito
10. Biblioteca
11. Escada Portaria
12. Lanchonete
13. Passarela Térreo
14. PET Mecânica
15. Secretaria PMR Térreo

16. MT15
17. MT05
18. MT19
19. Laboratório de Mecânica dos Fluidos
20. Sala TT09
21. Sala TT15

- Superior

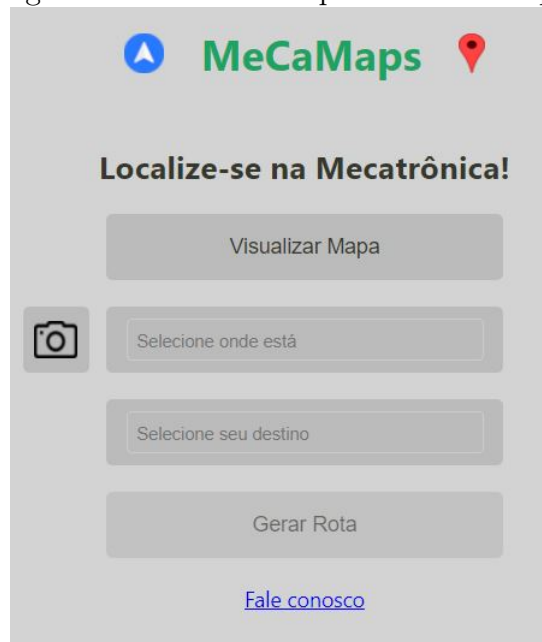
1. Sala MC02
2. Secretaria Marinha
3. Sala ES41
4. Sala ES42
5. Passarela Superior
6. Secretaria PMR Superior
7. Sala MS27
8. Sala MS01
9. Sala MZ02
10. Sala MZ16
11. Sala TS27
12. Sala TS17
13. Sala TS02

4.3 Resultados

O acesso à plataforma pode ser feito pelo domínio [MeCaMaps](#).

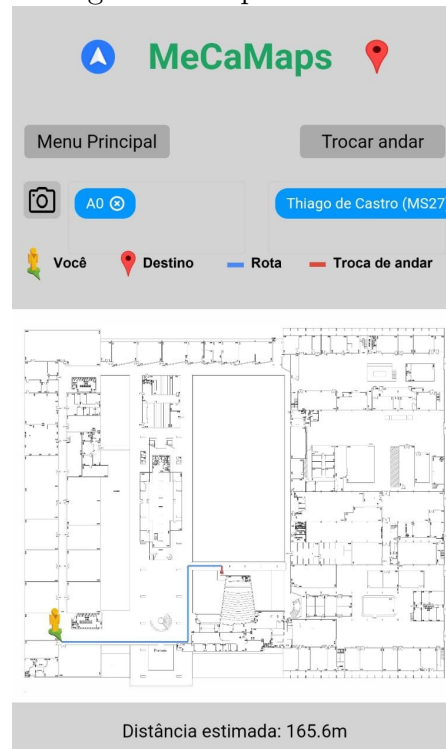
O resultado final consiste em três telas, mostradas a seguir:

Figura 25: Menu Principal do MeCaMaps



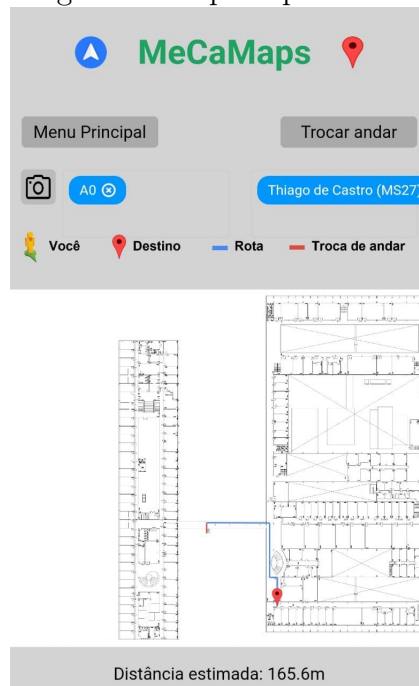
Fonte: mecamaps.poli.usp.br

Figura 26: Página do mapa térreo do MeCaMaps



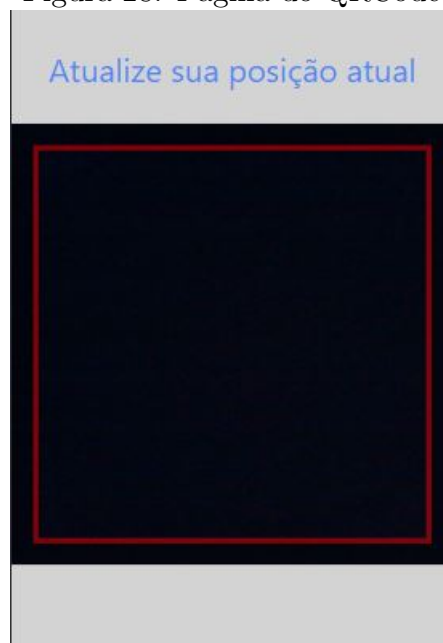
Fonte: mecamaps.poli.usp.br

Figura 27: Página do mapa superior do MeCaMaps



Fonte: mecamaps.poli.usp.br

Figura 28: Página do QRCode



Fonte: mecamaps.poli.usp.br

O funcionamento é bem simples e intuitivo. Após a seleção da origem e do destino, uma linha azul surge indicando o caminho, bem como um indicador da posição atual e um do destino. Após isso, basta que o usuário siga o trajeto, podendo, também, durante o percurso, atualizar sua posição selecionando uma nova posição atual ou escaneando algum QR Code que encontrar no trajeto.

4.4 Manutenção

Nota-se a presença do botão *Fale conosco* na tela principal, que redireciona para um formulário, o qual pode ser preenchido com sugestões de melhorias e problemas com a plataforma, por exemplo, possibilitando, também, o anexo de arquivos. Por meio desta comunicação direta com os usuários da ferramenta, será possível conseguir fazer eventuais correções e alterações, tanto em características visuais do site, quanto nos mapas. Tem-se a intenção de realizar a manutenção por nós pelo período de um ano, no caso, até o final de 2022, e, depois disso, ensinar outra pessoa após este período para que ela assuma a responsabilidade pelo MeCaMaps.

4.5 Teste final

Para validar a eficiência do MeCaMaps, foram feitos testes com oito voluntários, todos ingressantes das Engenharias Mecânica e Mecatrônica dos anos 2020 e 2021.

Dos oito voluntários, seis não haviam visitado o prédio da Mecatrônica POLI-USP; os outros dois, sim, porém conheciam apenas o trajeto até o centro acadêmico da mecânica e mecatrônica (CAM). Ou seja, nenhum deles conhecia o prédio por completo previamente.

Cada voluntário recebeu três destinos diferentes no prédio, sendo dois deles no andar térreo e um no andar superior. Os voluntários precisavam chegar até os destinos utilizando a plataforma MeCaMaps, sendo que o primeiro trajeto sempre partiria da Portaria do prédio.

O resultado foi que todos os voluntários conseguiram acessar a plataforma MeCaMaps e chegar aos seus três destinos. Seis usuários não conseguiram localizar um QR Code nas proximidades durante o segundo ou o terceiro trajeto, mas em todos esses casos eles conseguiram selecionar sua localização atual manualmente. Além disso, quatro deles utilizaram o QR Code para atualizarem o trajeto.

Após os testes, os oito voluntários responderam que recomendariam o MeCaMaps

a outras pessoas e acreditam que uma plataforma como essa seria benéfica para outros prédios da Escola Politécnica.

Por fim, foram sugeridas algumas melhorias: mapa mais didático, utilização da câmera traseira - alteração realizada - e localização em tempo real. Tanto a primeira como a terceira sugestões fogem do escopo inicial do projeto e, portanto, não foram implementadas. Todavia, poderiam vir a ser incrementos em futuras versões.

Logo, o uso das diferentes funcionalidades do site permitiu que todos os alunos, que não conheciam o prédio, chegassem a todos os seus destinos com sucesso.

5 CONCLUSÃO

Nota-se que os requisitos de projeto foram cumpridos, criou-se uma plataforma prática e simples de utilizar. O PWA trouxe como vantagens a possibilidade de instalar a plataforma no dispositivo, como se fosse uma aplicação nativa, além de permitir a utilização offline, depois de já ter acessado o site pela primeira vez. Assim, removeu-se a obrigatoriedade de instalar uma aplicação, porém sem tirar a possibilidade de tê-la no dispositivo celular para aqueles que a quiserem.

A escolha de utilizar React foi bastante feliz, uma vez que a curva de aprendizado com a linguagem javascript e mais especificamente com JSX é bem acentuada. Como React é uma biblioteca bastante modular, a utilização de componentes existem, tais como o dropdown e a leitura de QR Codes, auxiliaram no desenvolvimento da plataforma.

Por fim, a convergência de respostas positivas a respeito da plataforma, somado ao fato da portaria do prédio indicá-la aos visitantes, mostra que é, de fato, funcional.

REFERÊNCIAS

- MAXIT. **Qual a precisão do GPS que você utiliza?** 2020. Disponível em: <https://www.maxit.com.br/qual-a-precisao-do-gps-que-voce-utiliza/>.
- SILVA, G. M. M.; SOSPEDRA, J. T.; HUERTA, J. **A Meta-Review of Indoor Positioning Systems**. 2019. DOI: <https://doi.org/10.3390/s19204507>.
- MOEGLEIN, M.; KRASNER, N. **An Introduction to SnapTrack Server-Aided GPS Technology**. 1998. *Proceedings of the 11th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1998)*, Nashville, TN, pp. 333-342.
- OGUNTALA, G. t. **Indoor location identification technologies for real-time IoT-based applications: An inclusive survey**. 2018. DOI: <https://doi.org/10.1016/j.cosrev.2018.09.001>.
- NASSER, Y. e. a. **Recent Advances in Indoor Localization: A Survey on Theoretical Approaches and Applications**. 2016. DOI: [10.1109/COMST.2016.2632427](https://doi.org/10.1109/COMST.2016.2632427).
- UREÑA, J. e. a. **Acoustic Local Positioning With Encoded Emission Beacons**. 2018. DOI: [10.1109/JPROC.2018.2819938](https://doi.org/10.1109/JPROC.2018.2819938).
- FUKUJU, Y. e. a. **DOLPHIN: An Autonomous Indoor Positioning System in Ubiquitous Computing Environment**. 2003. WSTFES,3,53.
- PENG, C. e. a. **BeepBeep: a high accuracy acoustic ranging system using COTS mobile devices**. 2007. <https://doi.org/10.1145/1322263.1322265>.
- FARAGHER, R.; HARLE, R. **Location Fingerprinting With Bluetooth Low Energy Beacons**. 2015. DOI: [10.1109/JSAC.2015.2430281](https://doi.org/10.1109/JSAC.2015.2430281).
- RUTVIK, J. **Indoor Positioning System**. 2020. <https://dx.doi.org/10.37622/IJAER/15.12.2020.1142-1146>.
- HALLBERG, J. e. a. **Indoor Positioning System**. 2020. <https://dx.doi.org/10.37622/IJAER/15.12.2020.1142-1146>.
- ANASTASI, G. e. a. **Positioning with bluetooth**. 2003. In *Proceedings of the 10th International Conference on Telecommunications, Papeete, French Polynesia, 23 February–1 March 2003; Volume 2*, pp. 954–958.
- GWON, Y. e. a. **Robust indoor location estimation of stationary and mobile users**. 2004. In *Proceedings of the IEEE INFOCOM 2004, Hong Kong, China, 7–11 March 2004; Volume 2*, pp. 1032–1043.
- KAUR, M. e. a. **RFID Technology Principles, Advantages, Limitations Its Applications**. 2011. *International Journal of Computer and Electrical Engineering vol. 3, no. 1*, pp. 151-157.

KORHONEN, J. e. a. **mTag – Architecture for Discovering Location Specific Mobile Web Services Using RFID and Its Evaluation with Two Case Studies**. 2006. DOI: 10.1109/AICT-ICIW.2006.134.

KASPERSKY. **Um guia sobre códigos QR e como fazer sua leitura**. [S.I], 2020. Disponível em: <https://www.kaspersky.com.br/resource-center/definitions/what-is-a-qr-code-how-to-scan>. Acesso em: 04.05.2021.

BLUEBITE. **The State of QR in 2021**: Updated qr code stats and exploration of qr usage today. [S.I], 2021. Disponível em: <https://www.bluebite.com/nfc/qr-code-usage-statistics#:~:text=The\%2094\%25\%\%20in\%20the,number\%20of\%20interactions\%20per\%20object.&text=This\%20all\%20adds\%20to\%20a,same\%202018\%2D2020\%20time\%20period.> Acesso em: 05.05.2021.

IBGE. **PNAD Contínua TIC 2018: Internet chega a 79,1% dos domicílios do país**. [S.I], 2020. Disponível em: <https://agenciadenoticias.ibge.gov.br/agencia-sala-de-imprensa/2013-agencia-de-noticias/releases/27515-pnad-continua-tic-2018-internet-chega-a-79-1-dos-domicilios-do-pais>. Acesso em: 05.05.2021.

QRCODE.COM. **History of QR Code**. 20–? Disponível em: <https://www.qrcode.com/en/history/>. Acesso em: 04.05.2021.

MARQUES, S. **QR Code**. *Knoow.net*, 2017. Disponível em: <https://knoow.net/ciencinformtelec/informatica/qr-code/>. Acesso em: 04.05.2021.

KOZAK, A. **Como criar um código QR com logo: instruções passo a passo**. *Logaster*, 2020. Disponível em: <https://www.logaster.com.br/blog/qr-code-logo/>. Acesso em: 04.05.2021.

GAO, Z.; ZHAI, G.; HU, C. **The Invisible QR Code**. 2015. DOI: <https://dl.acm.org/doi/10.1145/2733373.2806398>.

NCB, I. **History of QR Code**. 2018. Disponível em: <https://www.newtoncbraga.com.br/index.php/como-funciona/15548-como-funciona-o-qr-code-art4052.html>. Acesso em: 05.05.2021.

RAJ, C. P. R.; TOLETY, S.; IMMACULATE, C. **QR code based navigation system for closed building using smart phones**. 2013. DOI: 10.1109/iMac4s.2013.6526488.

MONTENEGRO, E. C. e. a. **QR-Maps: an Efficient Tool for Indoor User Location Based on QR-Codes and Google Maps**. 2014. DOI: 10.1109/CCNC.2011.5766643.

ILKOVICOVÁ, L.; ERDÉLYI, J.; KOPÁCIK, A. **Positioning in Indoor Environment using QR Codes**. 2014.

DIJKSTRA, E. "A Note on Two Problems in Connexion with Graphs.." *numerische mathematik* 1. 1959. P. 269-271. Disponível em: <http://eudml.org/doc/131436>.

DOMES, S. **Progressive Web Apps with React: Create lightning fast web apps with native power using React and Firebase**. Packt Publishing, 2017. 10 - 11 p. ISBN 9781788296137. Disponível em: <https://books.google.com.br/books?id=8RhKDwAAQBAJ>.

BODUCH, A. ***React and React Native***. Packt Publishing, 2017. 18 - 32 p. ISBN 9781786465658. Disponível em: <https://books.google.com.br/books?id=r9JJvgAACAAJ>.

BROWN, S. **The C4 model for visualising software architecture**. Disponível em: <https://c4model.com/>. Acesso em: 05.12.2021.

APÊNDICE A

A.1 Pesquisa de Demanda e Destinos

- Data da realização: De 3 de maio de 2021 a 14 de maio de 2021
- Canal: Formulários do Google enviados via grupos de Whatsapp
- Quantidade de Respostas: 350

A.2 Teste de Funcionalidade

- Data da realização: De 26 de junho de 2021
- Quantidade de Voluntários: 4
- Caminhos testados:

Tabela 3: Teste de Funcionalidade - Caminhos

Caminho	Partida	Destino
Fácil	Portaria	Lanchonete
Médio	Lanchonete	Sala MT05
Difícil	Sala MT05	Sala ES67

- Resultado: os 4 voluntários percorreram 3 caminhos e todos os destinos foram atingidos com sucesso, sendo que um dos voluntários atingiu um dos destino com um desvio de rota.

A.3 Teste Final

- Data da realização: De 24 de novembro de 2021 a 31 de novembro de 2021
- Quantidade de Voluntários: 8
- Destinos de cada voluntário:

Tabela 4: Teste Final - Destinos

Voluntário	Destino 1	Destino 2	Destino 3
Voluntário 1	Sala A6	Lab de MecFlu	Secretaria PMR Superior
Voluntário 2	CEN	PET Mecânica	Arturo Cordero
Voluntário 3	Secretaria PMR	Sala A6	Sala TS02
Voluntário 4	PET Mecânica	CEN	Sala MZ02
Voluntário 5	Lab de MecFlu	Sala A6	Secretaria PMR Superior
Voluntário 6	PET Mecânica	CEN	Arturo Cordero
Voluntário 7	Sala A6	Secretaria PMR	Sala TS02
Voluntário 8	CEN	PET Mecânica	Sala MZ02

• Perguntas:

1. Qual seu nome?
2. Qual seu Número USP?
3. Qual é a sua engenharia?
4. Qual é o seu ano de ingresso?
5. Você já tinha visitado o Prédio das Engenharias Mecânica, Mecatrônica e Naval antes?
6. Você conseguiu acessar a plataforma MeCaMaps?
7. Você conseguiu chegar ao seu 1° destino usando o MeCaMaps?
8. Você conseguiu chegar ao seu 2° destino usando o MeCaMaps?
9. Você conseguiu chegar ao seu 3° destino usando o MeCaMaps?
10. Caso não tenha conseguido chegar a algum destino, poderia descrever qual problema você teve no uso do site?
11. Você conseguiu localizar e utilizar os QR Codes que estão no prédio para definir sua localização?
12. Caso não tenha utilizado o QR Code em algum percurso, você conseguiu selecionar manualmente sua localização?
13. Durante seus trajetos, você utilizou um novo QR Codes para atualizar sua localização?
14. Você recomendaria o uso do MeCaMaps aos seus colegas?
15. Você acredita que outros prédios da USP se beneficiariam de um sistema igual ao MeCaMaps?

16. Você tem alguma sugestão de melhoria ou gostaria de relatar alguma dificuldade que teve durante o uso do MeCaMaps?

- Resultados:

1. Todos os voluntários chegaram aos respectivos 3 destinos.
2. Todos os voluntários recomendariam o uso do MeCaMaps para um colega.
3. Em todos os percursos os usuários conseguiram atualizar sua localização para realizar o percurso seguinte, seja com a leitura de um novo QR code noo prédio ou manualmente.