

DESENVOLVIMENTO DE TELEMETRIA PARA MONITORAÇÃO DE TESTES EM PROPULSORES AEROESPACIAIS DE BAIXO EMPUXO COM COMUNICAÇÃO VIA INTERNET.

Tatiane Kimie Kakazu de Almeida (IC) e Fabio Raia (Orientador).

Apoio: PIBIC Mackenzie

RESUMO

As atividades da indústria aeroespacial podem ser compreendidas pelo seu desenvolvimento em projetos de fabricação e operação de aviões, foguetes e outros veículos de transporte aéreo e espacial. Também seu envolvimento em pesquisas relacionadas em sua área de atuação, que trouxe, e continua a proporcionar, aumento dos padrões de um país através do contínuo desenvolvimento do conhecimento que agregam as atividades científicas; militares; de comunicação; educação; e sensoriamento. No Brasil, essa atividade não tem sido incentivada devido a falta de entendimento do que representa as missões espaciais, e o quanto as atividades nesse setor são benéficas para o desenvolvimento do país. Por esta razão, foi desenvolvido o projeto, que viabiliza a entrada desse tipo de pesquisa e conhecimento em universidades e escolas, no qual tem como objetivo facilitar e propor incentivo de aplicação na área, a fim de trazer crescimento das atividades aeroespaciais no Brasil. Esse trabalho é um acréscimo em bancadas de testes já existentes que tem sido utilizada em ligas acadêmicas, em que os testes são realizados em locais isolados para a segurança dos alunos, e os dados coletados pelos sensores de pressão, temperatura e empuxo, são enviados a um microcontrolador que disponibiliza, em uma rede de comunicação que torna esses dados visíveis durante o teste, e acessíveis por qualquer dispositivo IoT. Com o uso de desenvolvedor simples e gratuito, com programação intuitiva, e tecnologia de baixo custo, o sistema de telemetria foi projetado, e pode ser facilmente replicado, adaptado e aplicado em qualquer sistema de testes.

Palavras-chave: Telemetria. Foguete. Aeroespacial.

ABSTRACT

The activities of the aerospace industry can be understood by its development in projects for the manufacture and operation of airplanes, rockets and other air and space transport vehicles. Also its involvement in related research in its field of activity, which has brought, and continues to provide, an increase in the standards of a country through the continuous development of knowledge that aggregate scientific activities; military; of communication; education; and sensing. In Brazil, this activity has not been encouraged due to the lack of understanding of what space missions represent, and how beneficial activities in this sector are for the development of the country. For this reason, the project was developed, which enables the

entry of this type of research and knowledge into universities and schools, which aims to facilitate and propose incentives for application in the area, in order to bring about growth in aerospace activities in Brazil. This work is an addition to existing test benches that have been used in academic leagues, where tests are carried out in isolated locations for student safety, and the data collected by pressure, temperature and thrust sensors are sent to a microcontroller that makes available, on a communication network, that data is visible during testing, and accessible by any IoT device. Using a simple and free developer, with intuitive programming, and low-cost technology, the telemetry system was designed and can be easily replicated, adapted and applied in any testing system.

Keywords: Telemetry. Rocket. Aerospace.

1. INTRODUÇÃO

A astronáutica atual teve seu início com a formulação teórica do cientista soviético de foguetes, Konstantin Tsiolkovsky, no entanto, a praticidade e executabilidade deve-se ao engenheiro e físico experimental norte americano Robert Goddard. O uso massivo de foguetes teve início ao final da segunda guerra mundial com os lançamentos dos foguetes alemães sobre Londres e continuou durante a corrida espacial entre Estados Unidos e União Soviética. Em 1957 os soviéticos, com o projeto Vostok, colocou o primeiro ser humano em órbita terrestre (NEWMAN, 2002). Dessa época em diante observa-se de dois a três lançamentos semanais. Um mercado – aeroespacial – crescente a ser explorado com muitas oportunidades e possibilidades para pesquisa, cujo valor agregado é extremamente elevado (TKATCHOVA, 2018).

No Brasil, o desenvolvimento de foguetes iniciou-se em São Paulo, na década de 60, no Centro Técnico Aeroespacial (CTA), em São José dos Campos (PALMERIO, 2017), e os primeiros foguetes produzidos foram os de sondagem, que tem como objetivo levar cargas úteis à atmosfera. A produção principal de foguetes civis no país, o Instituto de Aeronáutica e Espaço (IAE), foi criado em 1969. Atualmente, possui duas bases de lançamentos dos foguetes no nordeste do país, especificamente localizado em Alcântara, no Estado do Maranhão, e a outra em Natal, no Rio Grande do Norte.

A atividade aeroespacial no Brasil está em desenvolvimento lento devido a uma parcela da população que condena a plena atividade espacial nacional, pois julgam haver melhor uso dos recursos econômicos-financeiros. De acordo com Palmerio (2017), “isso se deve em sua essência, ao desconhecimento do que representa a missão espacial em termos de arregimentação, formação e alavancagem de todas as camadas da população.” A atividade espacial tem beneficiado áreas que não são diretamente relacionadas, mas que induz tudo que eleva os padrões de um país, dinamizando o contínuo desenvolvimento do conhecimento ao incluir as atividades científicas, militares, de comunicação, de educação e sensoriamento, beneficiando e cooperando na evolução dos países que nela investem. As nações que não são desenvolvedores dos recursos espaciais também pagam o investimento dos países que produzem, mas não usufruem do benefício completo alcançado pelos executores, portanto, trata-se de um dever cívico proporcionar a mesma oportunidade ao Brasil (PALMERIO, 2017).

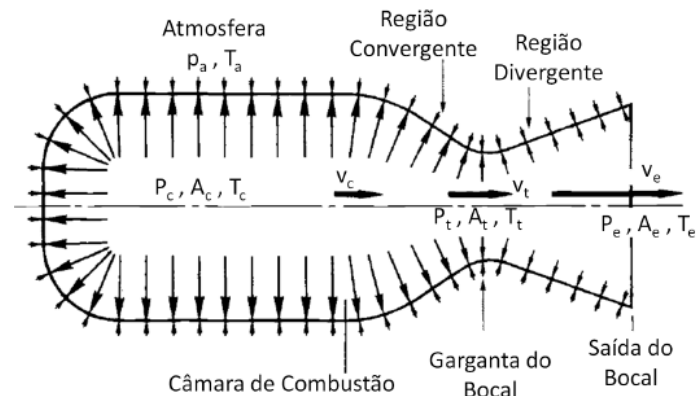
O sucesso de uma missão está ligado a correta interpretação de dados físicos oriundos das diversas partes de um foguete. A robustez e o fluxo de dados entre partes *onboard* e estação em terra traçam um panorama de como o foguete ou sonda está se comportando. Essa telemetria, assim chamada, compõe o sistema de comunicação (NEWMAN, 2002).

Partes específicas de um foguete, tal qual o sistema de propulsão, são também alvo de monitoração para efeito de projeto, controle ou validação de proposições analíticas. Ribeiro (2013), realiza um levantamento de parâmetros para desempenho de um motor a propulsão sólida. Trata da aquisição de dados de um motor experimental, onde as grandezas, temperatura e pressão são dados de saída para validação de cálculos analíticos.

O sucesso das espaçonaves ou sondas está ligado a exaustivos testes práticos que consolidam as avaliações numéricas. Uma maneira de realizar experimentos antes da utilização final, se faz por meio de uma plataforma de testes instrumentada com sistema de aquisição de dados fixo e telemétrico, essa forma é uma maneira de prover os dados físicos ao grupo de estudos e comunidade acadêmica interessada, que está distante do local de teste. Uma bancada construída para essa finalidade foi concebida para ter mobilidade de transporte e de adaptação a tipos de combustíveis, no que se trata de modelos de foguetes acadêmicos e irá seguir as recomendações da NR – 12 (Norma Regulamentadora).

Ensaio realizados em bancada de teste, são fundamentais para compreensão de processos (SUTTON, 2001), onde a teoria e a prática se encontram (PALMERIO 2017). Nessa bancada serão realizadas medidas das grandezas força (empuxo), temperatura e tempo de queima, deformação e pressão. Por certo, o levantamento de uma curva de empuxo é fundamental para a validação de motores foguete, que contribui para dinâmica da trajetória além de ser um parâmetro importante para o projeto do veículo. Mas, as curvas de pressão, temperatura e de deformação estrutural também são importantes e servem como entrada do projeto e dimensionamento do motor. Toma-se como exemplo a câmara de combustão de um foguete a propelente sólido (SUTTON, 2001), figura 1, adaptado de Ribeiro (2013). Observa-se as variáveis de entrada no dimensionamento de câmara e tubeira P_c (pressão da câmara de combustão), T_c (temperatura da câmara de combustão) e V_c (velocidade da saída de gases).

Figura 1. Modelo convencional de câmara e tubeira e grandezas a serem medidas para dimensionamento ou validação do motor foguete.



Fonte: Rocket Propulsion Elements. Sutton (2001)

O presente trabalho é parte do projeto de uma bancada de testes para motores a propulsão sólida que conta com fomento da Fundação de apoio para projetos de pesquisa de ciência e tecnologia espaciais (FUNCATE). A bancada de testes, concebida para testes estáticos, possui suporte para a célula de carga e para o sensor de temperatura. Medidas de pressão serão realizadas por meio de transdutores de deformação (*strain gages*) aplicados na estrutura do motor. A medida de pressão é importante, pois o motor foguete deve suportar a combustão do grão propelente. Contafer (2009), apresenta um sistema de aquisição de dados para desenvolvimento de projeto de foguete em bancada de teste, instrumentada com sensores e sistema de aquisição da *National Instruments* composto chassi compacto modelo CDAQ 9172¹, e dois módulos NI 9211, atendidos por um VI (*virtual instrument*) específico desenvolvido por meio do software *Labview*®. Observa-se que a prática da coleta de dados experimentais é corriqueira e comum nesses casos. Nota-se, no entanto, que os dados oriundos da aquisição ficam a espera para serem manipulados, interpretados e analisados no local de testes.

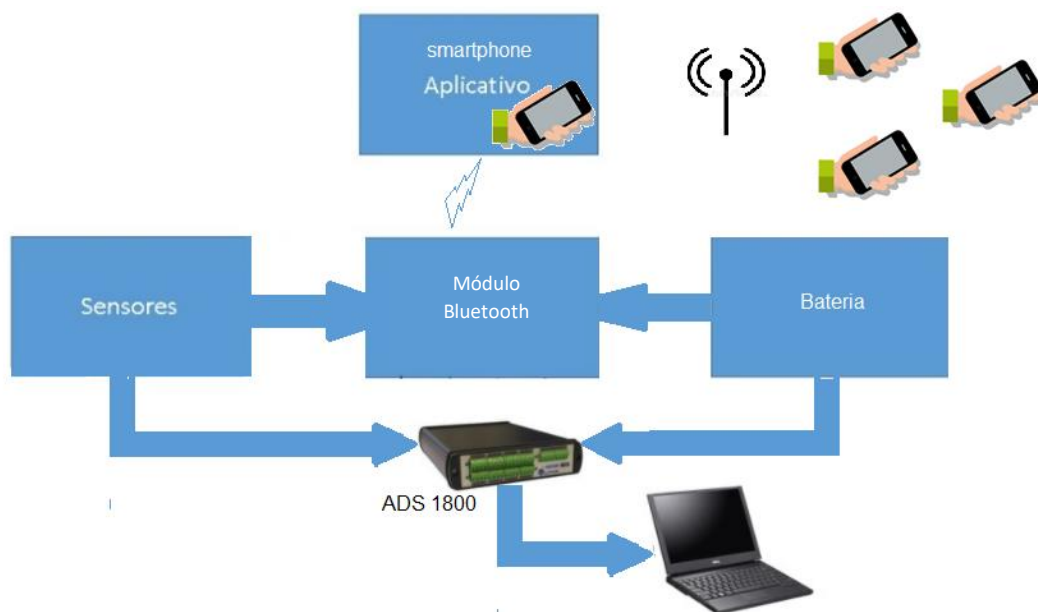
Baseado nos argumentos anteriores justifica-se o desenvolvimento de tecnologia para providenciar dados via *Internet*, com rapidez em tempo real para análises e avaliações de grupos específicos que, por motivos outros, não podem estar presentes no local ou no momento adequado. As aplicações na indústria e as interfaces gráficas do tipo interface homem máquina (IHM) permitem acesso a funcionalidades com interfaces de comunicação como *WiFi*, *USB*, *4G*. Essas facilidades são amplamente utilizadas em sistemas de telemetria, pois podem oferecer supervisão, coleta, armazenamento e controle com criptografia ponta a ponta (RILEY, 2012).

¹ National Instruments, Disponível em: <https://www.ni.com/pt-br/support/model.cdaq-9172.html>, acesso 15/04/2021

O objetivo do presente trabalho foi desenvolver um sistema de telemetria baseado em um aplicativo para *smartphone* com comunicação com a *Internet*. Especificamente, atrelado ao desenvolvimento, será utilizado a plataforma *Android*, presente em grande parte dos *smartphones*, bem como aplicativos prontos de domínio público. Necessário que esse sistema seja acessível economicamente e de fácil reprodução e compreensão.

O acompanhamento remoto foi realizado por meio de um *smartphone* utilizado como *host* e, com um aplicativo desenvolvido exclusivo, que receberá os dados via Bluetooth e disponibilizará via internet que serão recebidos em outros *smartphones* que possuam o aplicativo e a senha de acesso. O aplicativo irá apresentar, em sua tela, as curvas de empuxo, temperatura e pressão. Os dados gravados poderão ser compartilhados com outros aplicativos permitindo a visualização em outros dispositivos conectados a rede de internet. A figura 2, mostra a forma esquemática da comunicação de *software* e *hardware* para publicação de dados via aplicativo com *smartphone*.

Figura 2. Visão esquemática da telemetria com *smartphone* + módulo *bluetooth* + ADS 1800



Fonte: Acervo do autor

O desenvolvimento do aplicativo ocorreu, de acordo com Gamma et al. (1995), foi necessário fazer um levantamento dos requisitos funcionais e não- funcionais, que serviram de condições de contorno que deram forma e direção as funcionalidades do programa. A partir daí o sistema foi modelado na linguagem de programação desejada, que nesse caso, foi em blocos lógicos pré-programados da plataforma *MIT App Inventor*. Essa plataforma foi escolhida para a programação desse aplicativo pois além de ser gratuita, é intuitiva. Não é necessário ser um programador para utilizar essa plataforma, nem ter amplos conhecimentos

em lógica de programação, tornando a réplica e a manutenção do projeto por outras pessoas mais fácil.

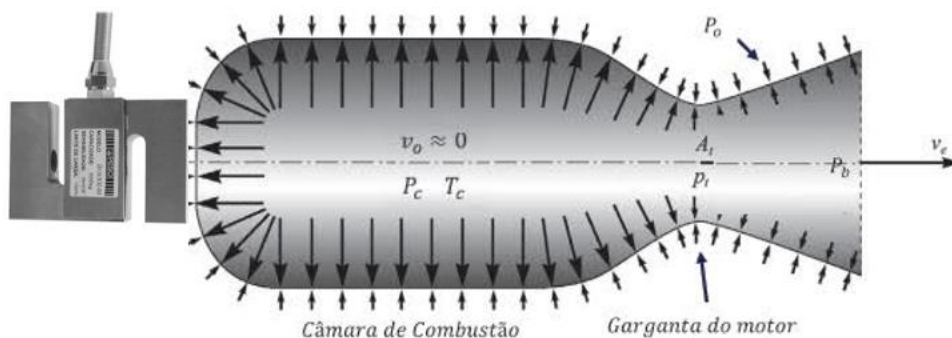
O *MIT App Inventor* foi criado pela *Google*, mas pertence ao *Massachusetts Institute of Technology (MIT)*. O objetivo dessa plataforma é tornar o desenvolvimento de aplicativos mais acessível para aqueles que não possuem experiência no assunto. É focado no desenvolvimento de aplicativos para pesquisas escolares, introduzindo a nova geração ao mundo tecnológico.

Com o aplicativo já programado, os testes são realizados para a aplicação das melhorias que serão identificadas. São necessários testes até que o programa esteja atendendo aos requisitos. Demais melhorias podem ser feitas posteriormente conforme sugestões do próprio cliente/usuário, (GAMMA ET AL. 1995).

2. DESENVOLVIMENTO DO ARGUMENTO

A telemetria é um sistema de medição de dados. Para isso é necessário um aplicativo, sensores e um microcontrolador para conversar com o módulo de rede. As grandezas medidas selecionadas são temperatura, pressão e empuxo. O empuxo é obtido pela célula de carga do tipo Z com sensibilidade de 2mV disposta conforme figura 3. A pressão dentro da câmara de combustão é medida com um Strain Gage colado do lado de fora, e a temperatura com um termopar. A taxa de amostragem foi de 80Hz.

Figura 3. Modelo convencional de câmara e tubeira e grandezas com célula de carga.



Fonte: Rocket Propulsion Elements. Sutton (2001)

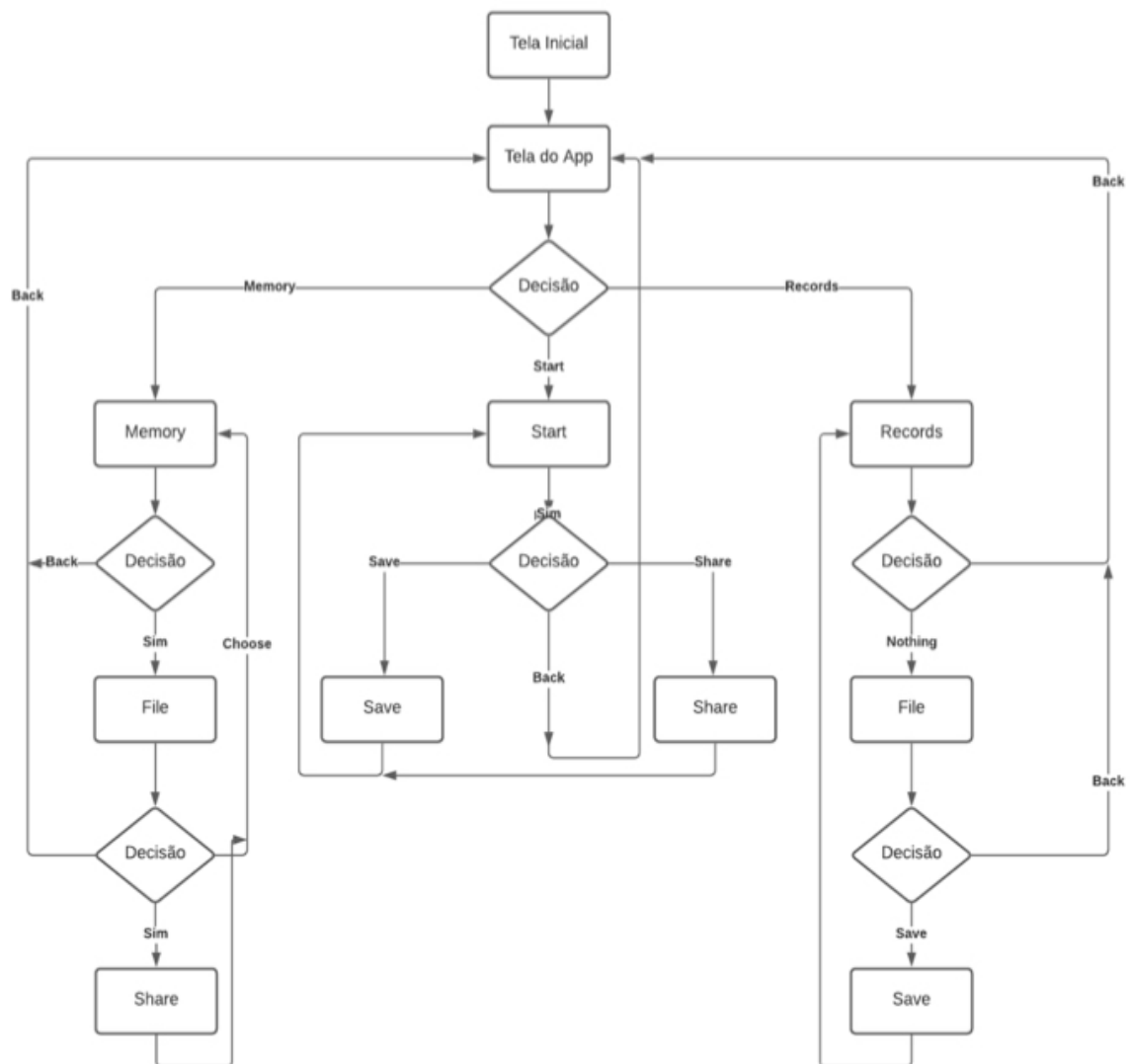
O aplicativo foi desenvolvido a partir da plataforma de desenvolvimento gratuita MIT App Inventor. A linguagem de programação está disposta em blocos, no qual é possível fazer toda a programação necessária para a construção da lógica e do layout do aplicativo. Para dar início na criação, os requisitos funcionais foram estudados e listados conforme esquema apresentado na figura 4 (HAREL, 1987).

Para levantar uma visão geral de sua funcionalidade, foi necessário a discriminação do caso de uso, atores, dados, estímulo e resposta. O caso de uso foi mostrar o *status* dos sensores em tempo real. Os atores são telemetria dos dados dos sensores. Os dados são que os sensores de temperatura, pressão e empuxo enviarão suas leituras para o microcontrolador, que se comunicará via *Bluetooth* com o aplicativo, que deverá receber esses dados e plotados em um gráfico. O estímulo é comunicação via *Bluetooth* com o celular, e *Wi-Fi* para outros aparelhos. A resposta esperada é que os dados possam ser lidos e compartilhados. Esses dados foram organizados dessa forma de acordo com o que está na literatura que descreve um quadro de caso de uso (BECK e CUNNINGHAM, 1989).

Para cada tela foi necessário o diagrama de sequência que modelava a sequência das ocorrências de cada solicitação feita pelo usuário. Dessa forma também pôde-se prever demais necessidades do sistema antes de ser programado e testado. Esse método fez com que botões e outros recursos surgissem na interface de comunicação com o usuário.

O fluxograma da figura 4, foi o ponto inicial do desenvolvimento. A partir dele as telas foram programadas e testada uma a uma, no qual outras alterações foram levantadas e discriminadas ao longo do texto.

Figura 4. Fluxograma das telas do aplicativo.



Fonte: Acervo do autor

Os requisitos funcionais para a tela inicial tiveram duas entradas de dados do usuário, conforme mostrado na figura 5, sendo uma discreta e uma em *string*. A saída é uma *string* nomeada de *password*, uma caixa para receber a senha que foi digitada pelo usuário e um botão para confirmar. Se a senha estiver incorreta, ela simplesmente irá sumir da caixa de texto e continuará na mesma tela, e uma mensagem abaixo da caixa de texto na cor vermelha aparecerá como um aviso com a mensagem que a senha está errada. Caso a senha esteja correta, passará para a tela principal do aplicativo.

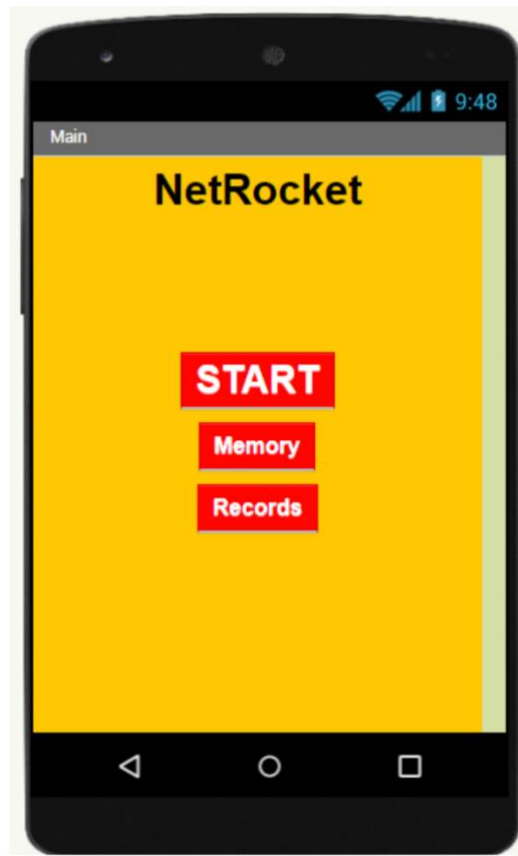
Figura 5. Tela inicial do aplicativo programado no *MIT App Inventor*.



Fonte: Acervo do autor.

A tela principal do aplicativo, na figura 6, tem duas entradas discretas de dados do usuário, sendo um botão de *Start* e um botão de *Memory*. O botão de *Records* previamente projetado, como apresentado no fluxograma da figura 4, foi removido devido ao tempo necessário de programação do mesmo. A programação desses botões é simples: quando cada botão for acionado, irá direcionar o usuário para a tela referente a sua escolha.

Figura 6. Tela principal do aplicativo.

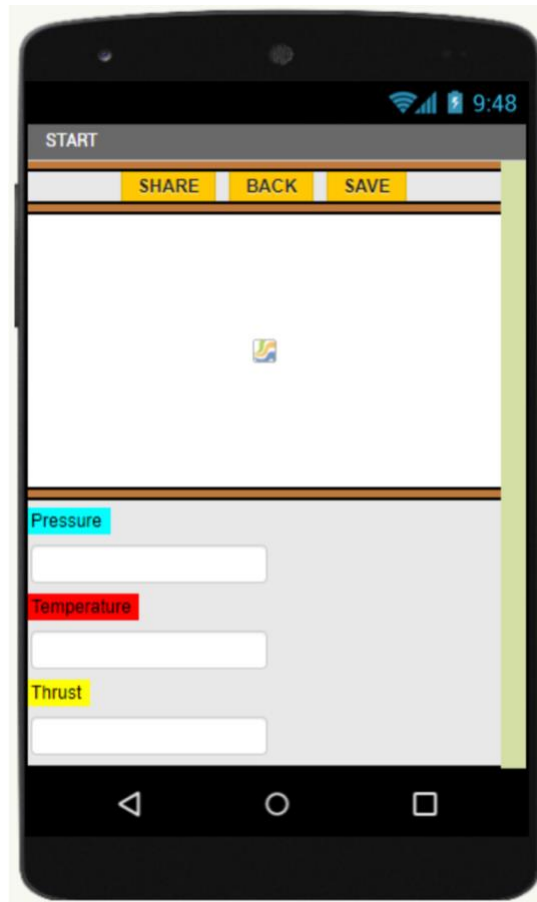


Fonte: Acervo do autor

Na tela referente ao botão de *Start*, na figura 7, mostra os dados em tempo real coletados pelos sensores aparecendo em um gráfico contendo as linhas dos sensores, baseada no gráfico que aparece no software do ADS 1800. Essa tela tem também duas entradas de dados discretas do usuário, que são as opções de botões: *Back*; e *Save*. Todos os botões *Back* conduzem o usuário para tela principal do aplicativo.

O botão *Memory* leva o usuário para uma tela, mostrada na figura 8, onde está todas as pastas salvas contendo os dados coletados pelos sensores permitindo que o usuário tenha acesso aos dados sempre que necessário com apenas um click na pasta desejada. Quando a pasta é selecionada, o usuário é direcionado a uma tela onde será possível visualizar os dados. Na tela *Memory* não houve a necessidade de ter um botão além do botão de *Back*.

Figura 7. Tela *Start* para visualizar os dados da simulação.



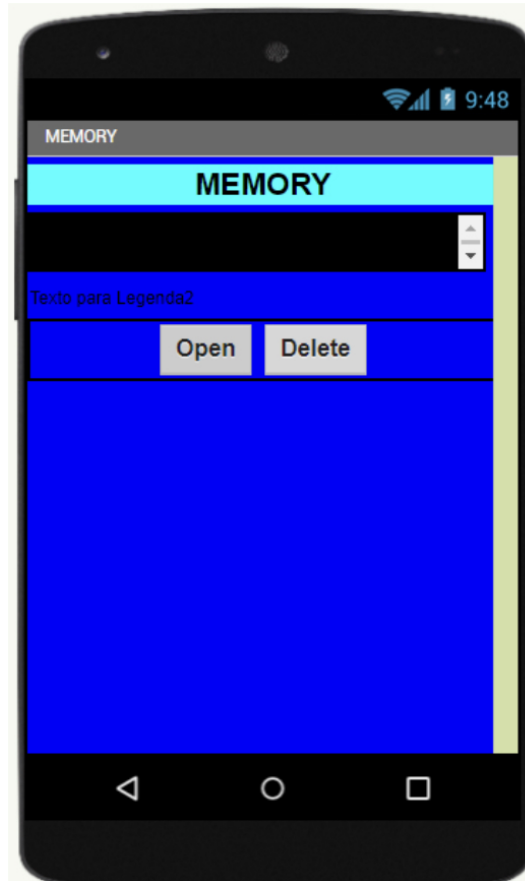
Fonte: Acervo do autor

Na tela apresentada após a seleção da pasta contida na tela *Memory*, o usuário poderá ver os dados coletados pelos sensores de ensaios da bancada anteriores. Nessa tela há, portanto, três entradas de dados discretas, que são os botões: *Share*; *Back*; e *Choose*. O botão *Choose* redireciona o usuário para a tela anterior afim de que o usuário escolha outra pasta de dados para abrir. O botão *Share* abre os aplicativos no qual o usuário poderá enviar os dados em formato de imagem.

A tela não implementada no projeto teste do aplicativo para a telemetria é a tela *Records*, que vale a pena cita-lá pois o aplicativo continuará passando por melhorias contínuas, e possivelmente será programada. Nessa tela, a ideia é de ver algo semelhante a tela *Memory*, porém, as pastas contidas nela não serão as mesmas das que foram salvas. Serão *files* contendo as simulações anteriores que poderão ser abertas dentro do período de quinze dias após a simulação, ou até serem salvas e conduzidas para a tela *Memory*. Depois dos 15 dias, essas pastas que não foram salvas serão apagadas do aplicativo

automaticamente. Ela não é requisito obrigatório no projeto, mas traz segurança no quesito de manter os dados salvos para uma futura decisão de adquiri-los e guardá-los.

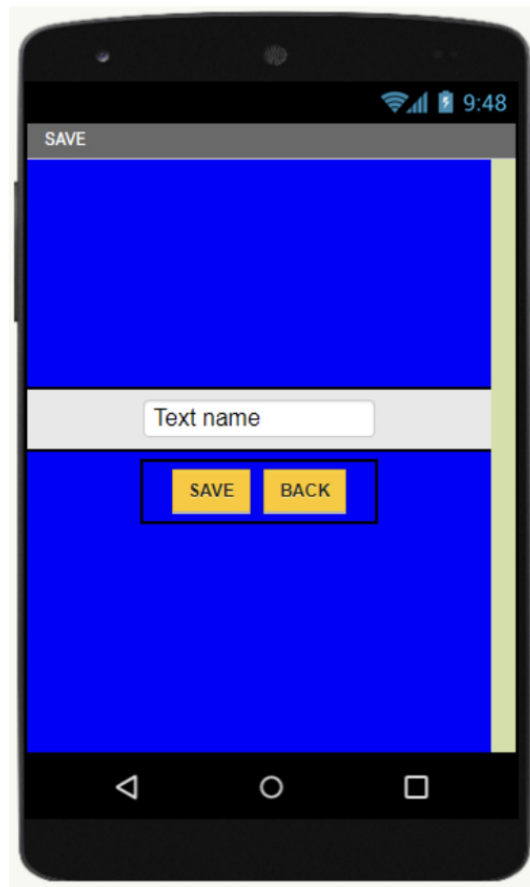
Figura 8. Tela de *Memory* do aplicativo.



Fonte: Acervo do autor

O botão *Save* conduz o usuário para uma tela onde ele terá que inserir o nome da pasta que estará salvando e confirmar com a opção, conforme figura 9, também como entrada de dado discreta, salvar. Foi programado um botão com a opção cancelar caso o usuário desista de salvar. Tanto o botão de salvar quanto o botão de cancelar, após executar o código referente as respectivas funções, fará o usuário retornar para a tela anterior, porém o botão de salvar direcionará essa pasta para a tela de *Memory*.

Figura 9. Tela de Save das simulações no aplicativo.



Fonte: Acervo do autor

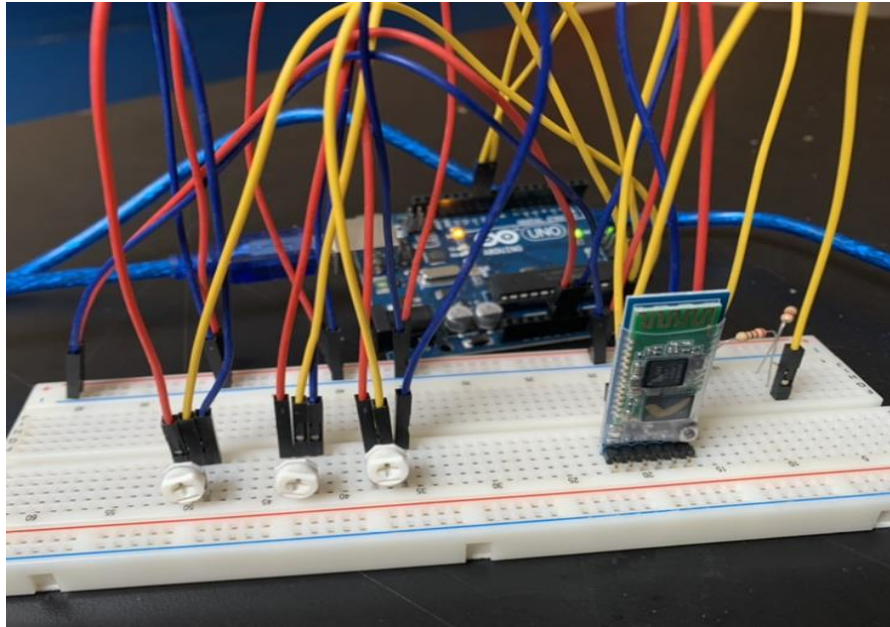
Na tela que aparecerá após a seleção do botão *Share*, vamos ver as opções dos aplicativos que queremos enviar, como por exemplo, o *Google Meet*. Com isso, será compartilhado a simulação, até mesmo em tempo real dos dados coletados pelos sensores como uma chamada de vídeo, levando os dados que chegaram ao celular por bluetooth para a rede de internet, caso o botão esteja na tela *Start*.

Após a programação, o código passou por correção três vezes, no qual houve o surgimento das necessidades de implementar avisos, e tomada de decisões automáticas caso o *bluetooth* não fosse identificado pelo aplicativo.

Para testar o seu funcionamento, foi necessário desenvolver um sistema de leitura de sensores, no qual enviaria os dados pela rede *bluetooth* para o aplicativo em *smartphone*. Para isso, o sistema foi montado utilizando o microcontrolador Arduino Uno, que fez o papel do ADS 1800, com o uso de potenciômetro no lugar dos sensores, na ordem em que apareceriam as leituras dos sensores principais do projeto, conforme figura 10. Embora não

sejam sensores, os dados enviados são recebidos pelo aplicativo como entrada de dados analógicas, igualmente nos sensores da bancada de testes, e por isso, foi possível testar o aplicativo e implementar melhorias a cada teste.

Figura 10. Sistema para teste do aplicativo utilizando Arduino Uno e módulo HC-05.



Fonte: Acervo do autor

Figura 11. Gráfico das resistências variáveis do arduino para teste 3



Fonte: Acervo do autor

3. CONSIDERAÇÕES FINAIS

Inicialmente o projeto foi desenvolvido para utilização do *myRIO*, que é um dispositivo de *hardware* embarcado, que faria a comunicação com o *smartphone*. A programação seria desenvolvida pelo *software AndroidStudio* com a linguagem *kotlin*, que é uma linguagem de programação que compila para máquina virtual Java. Com a caracterização dos requisitos para desenvolvimento do aplicativo durante a pandemia, adaptações foram feitas, mesmo assim, visando o projeto original de implementação.

O sistema recebe os valores medidos pelos sensores em ordem de coleta, pois ele não diferencia pelas grandezas. O microcontrolador usado não precisa ser o ADS 1800, como apresentado acima, o que torna o projeto de aquisição de dados ainda mais viável financeiramente. Além disso, o arduino é de fácil manuseio e aquisição, incluindo a compatibilidade com os sensores.

A plataforma utilizada para o desenvolvimento do aplicativo é gratuita, e com tutorial igualmente grátis, o que facilita o desenvolvimento e a introdução no ensino fundamental e

médio. A plataforma permite criar interface de compartilhamento de dados do aplicativo com outros via internet. Portanto, os dados chegam ao aplicativo por *bluetooth*, e saem dele via *internet*. Esse sistema de aquisição e compartilhamento não é algo novo, mas sim, adaptações de diversos trabalhos baseados em compartilhamento de dados via *Wi-Fi*, portanto, é facilmente replicável, concluindo a viabilidade e a facilidade de implementação.

4. REFERÊNCIAS

BECK, K.; CUNNINGHAM, W. A Laboratory for Teaching Object-Oriented Thinking. Proc. OOPSLA'89 (Conference on Object-oriented Programming, Systems, Languages and Applications), ACM Press, 1989.

CONDER, Shane; DARCEY, Lauren. Android Wireless Application Development. 2. ed, Addison Wesley, Boston (EUA), 2011.

CONTAIFER, Ricardo Amaral. Desenvolvimento de uma bancada de testes de propulsores híbridos de baixo empuxo. (Dissertação de Mestrado em Engenharia e Tecnologia Espaciais), São José dos Campos. 2009.

GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. Design Patterns: Elements of Reusable Object-oriented Software. Reading, Mass.: Addison-Wesley, 1995.

GUNASEKERA, Sheran. Android Apps Security. Apress, 2012.

HAREL, D. Statecharts: A Visual Formalism for Complex Systems. Sci. Comput. Programming, v. 8, n. 3, 1987.

MARTINS, Dener L. F. Redes Privadas Virtuais com IPSec. Escola de Extensão, Ciência da Computação, Universidade de Brasília: Brasília, 2000.

NEWMAN, Dava. Interactive Aerospace Engineering and Design. New York: McGraw-Hill. 2002.

PALMERIO, F. Ariovaldo. Introdução à Tecnologia de Foguetes, São José dos Campos: SindCT, 2017.

RIBEIRO, Marcus Vinícius Fernandes. Metodologia de projeto e validação de motores foguete a propelente sólido (dissertação de mestrado em Mecânica), Universidade de São Paulo, São Carlos. 2013. 78p

RILEY, Mike. Programming Your Home, Automate with Arduino, Android, and Your Computer. Ed.ThePragmatic Bookshelf. Dallas (EUA), 2012.

SUTTON, George. P. Biblarz. Rocket Propulsion Elements. New York: John Wiley & Sons, 2001.

TKATCHOVA, Stella. Emerging Space Markets. Springer-Verlag. GmbH Germany. 2018.

Contatos: tatianekakazu@outlook.com e fabio.raia@mackenzie.br