

LEITOR ACESSÍVEL: UMA PROPOSTA DE DISPOSITIVO PARA MEDIDAS DE MASSA EM AULAS PRÁTICAS DE QUÍMICA PARA PESSOAS COM DEFICIÊNCIA VISUAL

Resumo: A química é uma ciência experimental e um dos primeiros experimentos realizados em um laboratório é a medida de massa das substâncias. A massa é uma grandeza física presente em nosso cotidiano, sendo muito importante na realização de experimentos de química. Deste modo o uso da balança torna-se indispensável em um laboratório. Este projeto tem como proposta desenvolver uma aplicação para celulares na plataforma Android que realize reconhecimento de textos, mais especificamente de números emitidos por uma balança analítica, bem como realize operações matemáticas por comandos de voz, a fim de auxiliar na inclusão de estudantes portadores de deficiência visual em aulas de Química laboratoriais. O objetivo desta aplicação é que tais estudantes sejam incluídos e possuam mais independência ao realizarem experimentos em aulas práticas de Química. Para o desenvolvimento de tal aplicativo, é utilizada a linguagem de programação Java Orientada a Objetos no Ambiente de Desenvolvimento Integrado Android Studio.

Palavras-chave: Inclusão. Deficiência Visual. Aplicativo. Reconhecimento de Textos.

1 INTRODUÇÃO

A Química é uma ciência que deve apresentar atividades práticas e teóricas em sua metodologia de ensino. Para a parte prática é necessário o contato com o laboratório e a observação das transformações das substâncias nos experimentos, de modo a visualizar de forma macroscópica o que a teoria explica pelo lado microscópico (BASSOLI, 2014; ZUCCO, 1999, 2011; FRANCISCO JUNIOR, 2010; VAN BRAKEL, 1997). Proporcionar o ensino de forma que o aluno possa relacionar a prática com a teoria deve ser um método contínuo de ensino para todos os alunos, independente de suas necessidades ou diferenças. No que se refere ao processo de inclusão, poucos relatos de propostas de atividades de química experimental são encontrados. As diretrizes destacam em relação à educação inclusiva e a formação do docente a importância do professor frente à atual concepção da educação, cujo objetivo é a formação para o exercício pleno da cidadania. Dentre as metas traçadas para a docência, está: “assumir e saber lidar com a diversidade existente entre os alunos” (BRASIL, 2000). Neste contexto, educar significa que o docente deve considerar cada aluno como alguém singular e exclusivo. Deste modo é importante que os docentes estejam preparados e capacitados para as pessoas que possuem algum tipo de deficiência.

A justificativa para o desenvolvimento desta proposta dá-se pela instituição da Política Nacional de Educação Especial na Perspectiva da Educação Inclusiva, de 2008, advinda do Ministério da Educação, e que objetiva a promoção da inclusão no sistema de ensino brasileiro. Além do mais, de acordo com dados disponibilizados pelo Inep, entre os anos de 2004 e 2014, as matrículas de estudantes com deficiência no ensino superior aumentaram 518,66%, atingindo 33.377 matrículas (ALMEIDA, 2015; INEP, 2018). Segundo dados do Núcleo de Acessibilidade da Universidade Federal do Maranhão (UFMA) referentes ao ano de 2016, o Campus São Luís possui 148 alunos portadores de necessidades especiais, sendo que destes, 20,69% são de alunos com deficiência visual (DV) no Centro de Ciências Exatas e Tecnologias (UFMA, 2018).

Visando suprir a necessidade de alunos que possuem DV de usufruir o direito de acesso às aulas de Química Experimental, este trabalho se propõe a desenvolver uma aplicação na plataforma Android capaz de realizar processamento e leitura de dados amostrais obtidos em balanças analíticas, bem como a realização de operações matemáticas, através de comandos e reconhecimento de voz. Sua relevância deve-se ao fato de que a medida de massa é uma das primeiras experiências vividas em um laboratório de química e trata-se uma etapa fundamental para a maioria dos experimentos realizados nas práticas da disciplina de Química Experimental. Alguns trabalhos mostram que para um aluno com DV, a ausência de um monitor impossibilitará a realização destas medidas. Uma solução encontrada na literatura é apresentada por Supalo e colaboradores (2009, 2007) uma balança de nome “*Ohaus Scout Pro*” desenvolvida para ser usada por pessoas com DV para medidas de massa. No entanto este equipamento tem um custo muito elevado e, além disso, necessita de um computador portátil equipado com um programa específico - JAWS e Logger Pro, tornando-se inviável, visto que é oneroso para as instituições de ensino e para o uso pessoal. Fernandes e colaboradores (2011) mostram em seu trabalho que alunos com DV, na realização de experimentos de química, utilizam como procedimento para medidas de massa a contagem de espátulas. Isso demonstra que não existe uma precisão na realização das medidas visto que a quantidade de matéria na espátula é variável. Já Santos e colaboradores (2015), adaptaram com um indicador sonoro de pesagem uma balança de tríplice escala, permitindo a realização da medida por alunos com DV. Uma quarta escala foi adicionada a balança com sulcos que permitem avaliar o peso do objeto com precisão de 0.12g.

Frente a estas publicações observa-se a importância e a necessidade de materiais ou metodologias que permitam as pessoas com DV realizarem medidas de massa e também o investimento no desenvolvimento de tecnologias assistivas, tal como o aplicativo “*Leitor Acessível*”, produto deste trabalho. Vale ressaltar que a aplicação poderá ser habilitada para o uso de alunos com ou sem deficiência visual; isto é, qualquer um poderá ser capaz de utilizá-la e compreender suas ações executadas.

2 METODOLOGIA

2.1 Escolha da abordagem a ser implementada

A fim de auxiliar estudantes com deficiência visual a realizarem medidas de massa em aulas práticas de Química, foi escolhida a abordagem via desenvolvimento de software. Para tanto, fora desenvolvido um aplicativo capaz de realizar reconhecimento de caracteres em busca da identificação de números emitidos em uma balança analítica, assim como capaz de fazer operações matemáticas através da implementação de reconhecimento de voz.

O uso da plataforma Android escolhida justifica-se na sua popularidade, uma vez que este é o sistema operacional móvel mais utilizado de acordo com um relatório de 2017 da Google Play, que fechou tal ano ultrapassando a marca de 2 bilhões de usuários ativos (SÉRVIO, 2017). De acordo com dados do relatório da IDC (*International Data Corporation*) de 2017, o Android lidera o mercado mobile com 85% dos *smartphones* em mais de 190 países ao redor do mundo. E, segundo relatórios da Google, a cada dia mais de um milhão de novos usuários chegam ao Android para consumo de jogos e aplicativos. Além do mais, outro aspecto relativo ao Android é que celulares não são os únicos a serem equipados com tal plataforma, podendo ser utilizado também por outros dispositivos como Tablets, consoles móveis, TVs, computadores, entre outros (FLORENZANO, 2015).

Para o desenvolvimento de aplicativos nesta plataforma, é necessário utilizarmos o *Software Development Kit* (SDK), que é um pacote de ferramentas para o desenvolvimento de

aplicações mobile, assim como uma IDE (*Integrated Development Environment*), que auxilia na implementação, sendo geralmente utilizada com a SDK para criar o ambiente de desenvolvimento (GANDHEWAR, 2010).

Os principais componentes do *framework* Android são: *Activities*, *Services*, *Intents* e *Broadcast receivers*. O *Activity* representa a tela na qual o usuário interage com a aplicação; já o *Service* é responsável por rodar tarefas em *background*, não possuindo componentes de interface; o componente *Intent* é responsável por iniciar os demais, isto é, para criar ou chamar uma *Activity* ou *Service*, utiliza-se *Intents*; e, por fim, o *Broadcast receiver* responde a eventos do sistema (MEDEIROS, 2014).

Dentre os principais pontos de planejamento e elaboração almejados no desenvolvimento de um aplicativo, tem-se: simplicidade, funcionalidade, navegabilidade, performance e suportabilidade (PORTAL EDUCAÇÃO, 2018). Tratando-se de desenvolvimento de aplicativos voltados a pessoas com necessidades especiais, em especial a deficiência visual, a característica principal que deve estar corretamente elaborada é a simplicidade, uma vez que, no caso com maior dificuldade de interação com o usuário - quando o usuário possui cegueira total -, o aplicativo deverá garantir que este também consiga obter acesso a todos os recursos da aplicação. Portanto, devido aos motivos aqui listados, adotou-se a solução baseada em software, em que o aplicativo elaborado foi aqui chamado de “*Leitor Acessível*”.

2.2 Definição da aplicação Android que dará suporte à execução dos experimentos por parte dos discentes

Dentro da abordagem via *software*, o conjunto de dispositivos eletrônicos usados para processamento dos dados obtidos na balança analítica são compostos por: um Tablet Android 5.1.1, que é o dispositivo responsável por suportar e executar a aplicação; e a própria aplicação Android programada para Orientação a Objetos em linguagem Java.

Na busca por soluções de *software* similares à proposta aqui desenvolvida, vale a pena ressaltar a criação de aplicativo Android de identificação de soluções químicas para deficientes visuais, que através de métodos de processamento de imagens em dispositivos móveis, reconhece a cor de soluções químicas usando a câmera do dispositivo (SILVA, 2017).

Quanto à especificação dos requisitos funcionais e não-funcionais do sistema aqui desenvolvido, podemos citar os seguintes requisitos funcionais: método para interagir com o usuário através da vocalização de orientações a serem tomadas pelo mesmo; elaboração de método de reconhecimento de comandos de voz do usuário em português brasileiro; captura e processamento de imagens em tempo real; reconhecimento de textos e caracteres presentes nas imagens; armazenamento dos números emitidos na balança analítica; cálculo da média dos números emitidos na balança analítica; cálculo do desvio médio; vocalização do resultado obtido.

Dentre alguns dos requisitos não-funcionais presentes na aplicação, temos: a aplicação deve ser compatível com a versão 5.1.1 do dispositivo Android; o tempo de transição automática da tela inicial para a tela de comando deve ser de 5 segundos; diminuição da taxa de velocidade de vocalização enquanto a balança analítica ainda não estabilizou em 0.5 vezes; o tempo de leitura das imagens deve ser menos de 0,5 segundo; a vocalização dos textos e números reconhecidos também deve ser feita em tempo real, não ultrapassando 1 segundo de *delay*.

2.3 Implementação do sistema proposto

No desenvolvimento do aplicativo, foram realizadas as seguintes tarefas: criação de uma aplicação na IDE *Android Studio*; elaboração de uma tela principal como guia para o usuário,

utilizando os recursos da API *Text-To-Speech* para tanto; elaboração de um tutorial próprio para o “*Leitor Acessível*”, a fim de que o usuário tenha conhecimento de todas as funcionalidades disponíveis na aplicação; elaboração de uma *Activity* responsável unicamente pelo reconhecimento de caracteres usando OCR com a API *Mobile Vision Text*, da Google, para Android; implementação do *Text-To-Speech* para vocalização dos resultados obtidos e de textos como um todo; implementação da API *RecognizerIntent* para reconhecimento de comandos de voz em português brasileiro emitidos pelo usuário; realização de cálculos de média e desvio dos números obtidos na balança analítica; informar o usuário a massa obtida na balança analítica com sua margem de erro.

3 RESULTADOS

3.1 Criando a aplicação Android

O aplicativo aqui desenvolvido deve ser capaz de realizar reconhecimento de textos, que é um processo que detecta blocos de textos latinos em imagens em tempo real no dispositivo, assim como reconhecimento de voz para produção de comandos do usuário. Para tanto, utiliza-se a Interface de Programação de Aplicativos (API, em inglês) do Google Mobile Vision para Reconhecimento Óptico de Caracteres (OCR, em inglês) e a *RecognizerIntent*, que suporta o reconhecimento de voz através de uma inicialização de *Intent*.

Para o propósito do aplicativo aqui desenvolvido, a interface de usuário deverá objetivar a simplicidade. Ou seja, é desejável que haja poucos elementos de interação com usuário, e que estes sejam de fácil acesso. No caso da tela principal - isto é, a primeira tela com que o usuário irá interagir antes de utilizar os recursos de reconhecimento de caracteres -, foram implementados apenas dois recursos visuais: o nome do aplicativo (*Leitor Acessível*) em tamanho 35sp e negrito, o que facilitará o uso para alunos que possuem deficiência visual moderada, e também serve para indicar aos outros estudantes presentes o que o usuário está prestes a fazer; um botão com largura máxima e 512dp de altura, para ampliar a área de toque do usuário e permitir com que este tenha grandes possibilidades de não errar a área de toque.

3.2 Interação com o usuário

Na primeira tela de interação com o usuário, este pode optar por uma, de duas escolhas disponíveis: escutar o tutorial do aplicativo a fim de descobrir todas as funcionalidades oferecidas; ou aguardar 5 segundos para ser redirecionado para a tela de comandos.

Para ter acesso ao tutorial, é necessário apenas que o usuário clique na tela uma única vez e será redirecionado para o tutorial, o qual usa a mesma biblioteca *Text-To-Speech* para explicar as funcionalidades do “*Leitor Acessível*” ao usuário.

3.3 Identificação de comandos de voz do usuário

Para fazer a identificação dos comandos de voz do usuário, é utilizada a API *RecognizerIntent*, que suporta o reconhecimento de voz em português brasileiro através de uma inicialização de *Intent*.

O método *ReceberComando()*, que realiza o reconhecimento de voz, é implementado tanto na tela de decisão de funcionalidade, chamada *SecondActivity*, em que o usuário opta pela leitura de caracteres ou pelo uso da calculadora; e é implementado também na *Activity* da calculadora, para o recebimento da expressão matemática que o usuário deseja que a calculadora opere.

O método *ReceberComando()* é chamado no *onStart()* de cada uma das duas *Activities* (*SecondActivity* e *CalculadoraActivity*), em que é utilizada uma *Intent* que solicita ao usuário



a fala e o envia por meio de um reconhecedor de fala. Os resultados são retornados por meio do método *onActivityResult*.

A implementação dessa API transmite o áudio recebido para servidores remotos a fim de executar o reconhecimento de fala. Sua implementação pode ser vista na Figura 1.

Figura 1- Implementação do método *ReceberComando()* na *CalculadoraActivity*.

```
private void ReceberComando() {  
    new Handler().postDelayed(new Runnable() {  
        @Override  
        public void run() {  
            Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);  
            intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);  
            intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, value: "pt-BR");  
  
            try {  
                startActivityForResult(intent, REQ_CODE_SPEECH_INPUT);  
            } catch (ActivityNotFoundException a) {  
            }  
        }  
    }, delayMillis: 15000);  
}  
  
//Recebendo os comandos de voz  
@Override  
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
    super.onActivityResult(requestCode, resultCode, data);  
    switch (requestCode) {  
        case REQ_CODE_SPEECH_INPUT: {  
            if (resultCode == RESULT_OK && null != data) {  
                ArrayList<String> result = data  
                    .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);  
                expressao.setText(result.get(0));  
            }break;  
        }  
    }  
    calcular();  
}
```

Fonte: Próprio autor

3.4 Implementação da calculadora

As operações matemáticas suportadas pela calculadora são quatro: soma, subtração, multiplicação e divisão. Para realizar tais operações, é requisitado ao usuário que informe a expressão matemática desejada por comando de voz.

Em seguida, através do método *calcular()*, o comando recebido é armazenado em uma String, em que separa-se os elementos da String em um vetor de String, e, para cada caso de operação (soma, subtração, multiplicação e divisão), usa-se o primeiro e último elemento do vetor de String - pois estes são os que contêm os números informados pelo usuário - e transforma-os de String para o tipo Double. O resultado de cada operação é vocalizado para o usuário através da API Text-To-Speech.

O método de calcular a expressão recebida do usuário pode ser visualizado na Figura 3.



Figura 3 - Implementação do método calcular() na *CalculadoraActivity*.

```
private void calcular() {  
    String resposta = expressao.getText().toString();  
    String[] separa = resposta.split( regex: "\\s+");  
    Double n1 = Double.parseDouble(separa[0]);  
    Double n2 = Double.parseDouble(separa[separa.length-1]);  
    if (resposta.contains("+")){  
        Double soma = n1 + n2;  
        resultado.setText(" = " + soma);  
        final String sum = "0 resultado da soma é: " + soma;  
        fala = new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {  
            @Override  
            public void onInit(int status) {  
                if (status == TextToSpeech.SUCCESS) {  
                    fala.setLanguage(new Locale( language: "pt", country: "br"));  
                }  
                fala.speak(sum, TextToSpeech.QUEUE_FLUSH, params: null);  
            }  
        });  
    } else if (resposta.contains("-")) {  
        Double subtracao = n1 - n2;  
        resultado.setText(" = " + subtracao);  
        final String sub = "0 resultado da subtração é: " + subtracao;  
        fala = new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {  
            @Override  
            public void onInit(int status) {  
                if (status == TextToSpeech.SUCCESS) {  
                    fala.setLanguage(new Locale( language: "pt", country: "br"));  
                }  
                fala.speak(sub, TextToSpeech.QUEUE_FLUSH, params: null);  
            }  
        });  
    } else if (resposta.contains("x")){  
        Double multi = n1 * n2;  
        resultado.setText(" = " + multi);  
        final String mult = "0 resultado da multiplicação é: " + multi;  
        fala = new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {  
            @Override  
            public void onInit(int status) {  
                if (status == TextToSpeech.SUCCESS) {  
                    fala.setLanguage(new Locale( language: "pt", country: "br"));  
                }  
                fala.speak(mult, TextToSpeech.QUEUE_FLUSH, params: null);  
            }  
        });  
    } else if (resposta.contains("/")) {  
        Double div = n1 / n2;  
        resultado.setText(" = " + div);  
        final String divisao = "0 resultado da divisão é: " + div;  
        fala = new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {  
            @Override  
            public void onInit(int status) {  
                if (status == TextToSpeech.SUCCESS) {  
                    fala.setLanguage(new Locale( language: "pt", country: "br"));  
                }  
                fala.speak(divisao, TextToSpeech.QUEUE_FLUSH, params: null);  
            }  
        });  
    }  
}
```

Fonte: Próprio autor

3.5 Reconhecimento de caracteres com a API *Mobile Text Vision*

Dentro do método *onCreate()*, é chamado o método responsável pela inicialização da câmera do dispositivo. Dentro deste método, aqui chamado de *startCameraSource()*, é criado um objeto da classe *TextRecognizer*, que habilita a câmera do dispositivo e define seus parâmetros, e então após fazer a verificação das permissões de acesso à câmera, chama o método de detecção de texto da API.

No método de detecção de caracteres, os itens detectados são transformados para *String* e, em seguida, alocados em um vetor do tipo *SparseArray*. Em seguida, são alocados os números reais em uma lista; isso porque, de tal forma, é possível isolar os números emitidos pela balança dos possíveis textos detectados, a fim de realizar cálculo de média e de desvio. Em seguida, o resultado, contendo o último número emitido pela balança, juntamente com sua margem de erro, é vocalizado para o usuário.

Vale ressaltar que, quando a *String* que contém os caracteres detectados contém a expressão "Please wait", que é uma expressão emitida pela balança analítica, é vocalizado para o usuário a expressão "Por favor, aguarde", com uma desaceleração da taxa de velocidade de fala.

O resultado pode ser visualizado na Figura 4 abaixo.



Figura 4 – Representação da tela (a) inicial e (b) em uso do aplicativo com reconhecimento da balança analítica disponível.



Fonte: Próprio autor

4 CONSIDERAÇÕES FINAIS

Neste trabalho, foi desenvolvido o aplicativo “*Leitor Acessível*” para leitura e processamento de imagens em tempo real, bem como realização de operações matemáticas através de comandos de voz do usuário.

A contribuição deste trabalho é permitir a inclusão de alunos portadores de deficiência visual em aulas práticas de Química, mais especificamente em medidas de massa, para que estes possam ter a experiência de participar de práticas na disciplina de forma mais independente, superando suas limitações.

Em trabalhos futuros, pretende-se realizar experiências no laboratório de Química em situações reais com estudantes portadores de deficiência visual, a fim de avaliar a eficácia da aplicação aqui desenvolvida.

Agradecimentos

A Universidade Federal do Maranhão e a Fundação de Amparo à Pesquisa e ao Desenvolvimento Científico e Tecnológico do Maranhão (FAPEMA): Projeto “*QuiKits - Desenvolvimento de Kits para a Inclusão de Portadores de Necessidades Educacionais Especiais em Aulas Práticas de Química*” (Processo UNIVERSAL-01269/17).

REFERÊNCIAS

ALMEIDA, J. G. A.; BELLOSI, T. C.; FERREIRA, E. L. Evolução da matrícula de pessoas com deficiência na educação superior brasileira: subsídios normativos e ações institucionais para acesso e permanência, Revista Ibero-Americana de Estudos em Educação, v.10, n. esp. 2015.

Android on Board. **TextToSpeech! Faça sua aplicação falar.** Disponível em: <https://androiddevbr.wordpress.com/2012/08/31/texttospeech-faca-a-sua-aplicacao-falar/>.

Acesso em: 20 ago. 2018.

BASSOLI, F. Atividades práticas e o ensino-aprendizagem de ciência(s): mitos, tendências e distorções, **Ciênc. Educ.**, Bauru, v. 20, n. 3, p. 579-593, 2014.



BRANDIZZI, Adam. **Orientação a objetos no Java**. Disponível em: <https://www.devmedia.com.br/programacao-orientada-a-objetos-com-java/>. Acesso em: 24 ago. 2018.

BRASIL, Ministério da Educação. Proposta de diretrizes para a formação inicial de professores da educação básica em cursos de nível superior. Brasília: Ministério da Educação, 2000.

BRITO, Halina Loos. Desenvolvimento de Kits para a Inclusão de Portadores de Necessidades Especiais em Aulas Práticas de Química. **Relatório de pesquisa – Universidade Federal do Maranhão**, São Luís, 2017.

CORDEIRO, Fillipe. **Android SDK: O que é? Para que Serve? Como Usar?**. Disponível em: <https://www.androidpro.com.br/blog/android-studio/android-sdk/>. Acesso em: 23 ago. 2018.

DEVELOPERS. **TextToSpeech**. Disponível em: <https://developer.android.com/reference/android/speech/tts/TextToSpeech>. Acesso em: 13 ago. 2017.

FERNANDES, T. C.; HUSSEIN, F. R. G. S.; DOMINGUES, R. C. P. R. Ensino de química para deficientes visuais: a importância da experimentação num enfoque multissensorial, **Química Nova na Escola**, São Paulo, v. 39, n. 2, p. 195-203, 2017.

FLORENZANO, Cláudio. **Porque você deveria desenvolver aplicativos Android**. Disponível em: <https://www.cbsi.net.br/2015/11/porque-voce-deveria-desenvolver-aplicativos-android.html>. Acesso em: 20 ago. 2018.

FRANCISCO JUNIOR, W. E. Estratégias de Leitura e Educação Química: Que relações? **Química Nova na Escola**, v. 32, n. 4, p. 220-226, 2010.

GANDHEWAR, N.; SHEIKH, R. Google Android: An Emerging Software Platform For Mobile Devices, **International Journal on Computer Science and Engineering**, Special Issue, 2010.

Google APIs for Android. **com.google.android.gms.vision.text**. Disponível em: <https://developers.google.com/android/reference/com/google/android/gms/vision/text/package-summary>. Acesso em: 20 ago. 2018.

<http://www.techtudo.com.br/tudo-sobre/android.htm/>. Acesso em: 28 abr. 2018.

MEDEIROS, Higor. **Projetando e criando aplicativos para dispositivos móveis**. Disponível em: <https://www.devmedia.com.br/projetando-e-criando-aplicativos-para-dispositivos-moveis/3067>. Acesso em: 24 ago. 2018.

Portal Educação. **Plataforma Android - Conceitos básicos**. Disponível em: <https://www.portaleducacao.com.br/conteudo/artigos/informatica/plataforma-android-conceitos-basicos/63389>. Acesso em: 24 ago. 2018.

PRAKASH, Pun. **Text recognition for Android using Google Mobile Vision**. Disponível em: https://medium.com/@prakash_pun/text-recognition-for-android-using-google-mobile-vision-a8ffabe3f5d6. Acesso em: 26 jul. 2017.

SANTOS, S. R. B.; DANIEL, L. X. L.; SILVA, A. A.; SILVA, P. R. A.; MEDEIROS, E. A. S.; SANTOS, L. M. Química experimental para deficientes visuais, **Latin American Journal of Science Education**, 2, 12015, 2015.

SILVA, Lucas; FERREIRA, Jailson M.; REGIS, Danilo. Aplicativo Android de Identificação de Soluções Químicas para Deficientes Visuais. In: XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, 2017, São Pedro, 2017.

SÉRVIO, Gabriel. **Sucesso absoluto: Android já ultrapassou a marca de 2 bilhões de usuários**. Disponível em: <https://www.tudocelular.com/android/noticias/n93431/android-ultrapassa-2-bilhoes-usuarios.html>. Acesso em: 07 maio 2018.

- SUPALO, C. A.; MALLOUK, T. E.; AMOROSI, C.; LANOUE, J.; WOHLERS, H. D.; MCENNIS, K. Using Adaptive Tools and Techniques To Teach a Class of Students Who Are Blind or Low-Vision, **Journal of Chemical Education**, v. 86, n. 5, p. 587-591, 2009.
- SUPALO, C. A.; MALLOUK, T. E.; AMOROSI, C.; RANKEL, L.; WOHLERS, H. D.; ROTH, A.; GREENBERG, A. Talking tools to assist students who are blind in laboratory courses, **Journal of Science Education for Students with Disabilities**, v. 12, n. 1- winter, 2007.
- UNIVERSIDADE FEDERAL DO MARANHÃO. Relatório de Gestão do Exercício, p.26-27, 2018.
- VAN BRAKEL, J. Chemistry as the science of the transformation of substances, **Synthese**, v. 111, n. 3, p. 253-282, 1997
- INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA. Sinopse Estatística da Educação Superior 2017. Brasília: Inep, 2018. Disponível em: <[HTTP://portal.inep.gov.br/basica-censo-escolar-sinopse-sinopse](http://portal.inep.gov.br/basica-censo-escolar-sinopse-sinopse)>. Acesso em 09.05.2019.
- ZUCCO, C. Química para um Mundo Melhor, **Química Nova**, v. 34, n. 5, p. 733, 2011.
- ZUCCO, C.; PESSINE, F. B.; ANDRADE, J. B. Diretrizes curriculares para os cursos de química, **Química Nova**, v. 22, n. 3, p. 454-461, 1999.

“LEITOR ACESSÍVEL”: A DEVICE PROPOSAL FOR MASS MEASURES IN EXPERIMENTAL CHEMICAL CLASSES WITH VISUAL IMPAIRMENT

Abstract: Chemistry is an experimental science and one of the earliest experiments performed in a laboratory is the mass measurement of substances. Mass is a physical quantity present in our daily life, being very important in the conduct of chemistry experiments. In this way the scale becomes one of the indispensable equipment in a laboratory. This project aims to develop an application for mobile phones in the Android platform that performs text recognition, specifically of numbers issued by an analytical scale, as well as performs mathematical operations by voice commands, to assist in the inclusion of students with visual disabilities in laboratory chemistry classes. The purpose of this application is for such students to be included and to have more independence when conducting experiments in practical classes. For the development of such an application, Java programming language is used, along with Android Studio Integrated Development Environment.

Keywords: Student inclusion. Visual impairment. Android app. Text Recognition.