

CrowdMusician: O poder do Crowdsourcing na assistência de produtores musicais com deficiência visual.

*Note: Sub-titles are not captured in Xplore and should not be used

Alexandre Borges Gonçalves
Escola de Ciências e Tecnologia
Universidade de Trás-os-Montes e Alto
Douro
Vila Real, Portugal
al70734@alunos.utad.pt

João Paulo Costa Rua
Escola de Ciências e Tecnologia
Universidade de Trás-os-Montes e Alto
Douro
Vila Real, Portugal
al70407@alunos.utad.pt

Luís Miguel Jesus
Escola de Ciências e Tecnologia
Universidade de Trás-os-Montes e Alto
Douro
Vila Real, Portugal
al70821@alunos.utad.pt

Abstract—Crowdsourcing is the act of using collective intelligence to obtain knowledge, services or goods. It involves getting information or opinions from a networked community, the crowd, who submit their data via the Internet. Crowdsourcing combines the efforts of volunteers or workers in an environment where each collaborator submit their small contribution in order to achieve a bigger result. Even though its use has benefited many sectors of society already, it has yet to be proven useful as a way to improve usability and/or accessibility in software usability, specifically when its users have some kind of disability. In this article we discuss and present a way to help improve the accessibility in an audio production software, e.g. FL Studio, to visually impaired people and facilitate the process of creating music.

Keywords—crowdsourcing, visually impaired, music production

Resumo—Crowdsourcing é uma estratégia que consiste em obter contribuições, ideias ou trabalho de uma grande quantidade de pessoas, geralmente através da Internet. Pode ser utilizado para resolver problemas, realizar tarefas ou criar novos produtos ou serviços. É uma forma de aproveitar o conhecimento e habilidades de um grande número de indivíduos de forma rápida e eficiente. O Amazon Mechanical Turk é uma plataforma de crowdsourcing que conecta empresas e indivíduos que precisam de trabalhos simples e rápidos realizados com trabalhadores remotos dispostos a realizar essas tarefas em troca de um pagamento. A Amazon fornece a plataforma e gere o pagamento dos trabalhadores. No âmbito de utilizar o crowdsourcing como prática de apoio a pessoas com necessidades especiais, foi criado o projeto CrowdMusician, com o objetivo de tornar a produção de música mais acessível para pessoas com deficiência visual ou que são iniciantes na área. Permite que o utilizador use a sua voz ou texto para solicitar elementos específicos de produção musical através do Amazon MTurk. Dessa forma, as pessoas cegas podem obter ajuda para criar música, mesmo que não tenham acesso a ferramentas visuais de produção.

I. INTRODUÇÃO

O processo de produzir música pode ser bastante confuso e demorado, especialmente se o utilizador não tiver muita experiência, não só com a área da música como com o software de produção musical.

Esta dificuldade torna-se muito maior quando o utilizador é um indivíduo com necessidades especiais, particularmente quando falamos de pessoas com deficiência visual. Tendo como referência o software de produção musical FL Studio,

sabemos que para produzir uma música são necessários diversos *samples*, isto é, um padrão que simboliza um som de um determinado instrumento. Obviamente que esta forma de produção musical é inacessível a pessoas com deficiência visual.

Tendo em conta este problema, pretendemos desenvolver um projeto, o CrowdMusician, que explora o potencial dos mecanismos de crowdsourcing como forma de apoio a pessoas cegas no trabalho de produção musical.

Através do CrowdMusician, o utilizador, através da interface da consola, tem a capacidade de usar a sua voz ou texto para criar uma campanha de crowdsourcing no Amazon Mechanical Turk (MTurk), de forma a que a crowd consiga fornecer padrões e elementos de produção musical que vão de acordo com o pedido do utilizador.

II. ESTADO DA ARTE

Tendo em conta que o nosso projeto tem como público-alvo pessoas com deficiência visual, temos obrigatoriamente de ter um método de reconhecimento de voz. Assim, pesquisando na literatura sobre a existência de trabalhos já realizados semelhantes ao nosso que relacionam campanhas de crowdsourcing com reconhecimento de voz, deparamo-nos com um conjunto de artigos que captaram o nosso interesse.

A ferramenta *Speak* [1] descreve uma abordagem para classificar imagens usando apenas a voz de *workers* de crowdsourcing, com o objetivo de criar uma maneira mais eficiente e económica de descrever imagens. A abordagem proposta foi testada com sucesso em um conjunto de imagens de objetos comuns e os resultados mostraram que é possível obter anotações de imagem precisas usando este método.

O artigo [2] discute um *marketplace* de crowdsourcing baseado em voz para pessoas com deficiência visual que vivem em regiões mais desfavorecidas. O objetivo do *marketplace* é proporcionar uma plataforma acessível para que as pessoas cegas possam participar de tarefas de crowdsourcing e, assim, ter a oportunidade de ganhar dinheiro através da transcrição de ficheiros de áudio. O artigo descreve a implementação do *marketplace* e apresenta resultados de uma avaliação com utilizadores reais.

III. METODOLOGIA

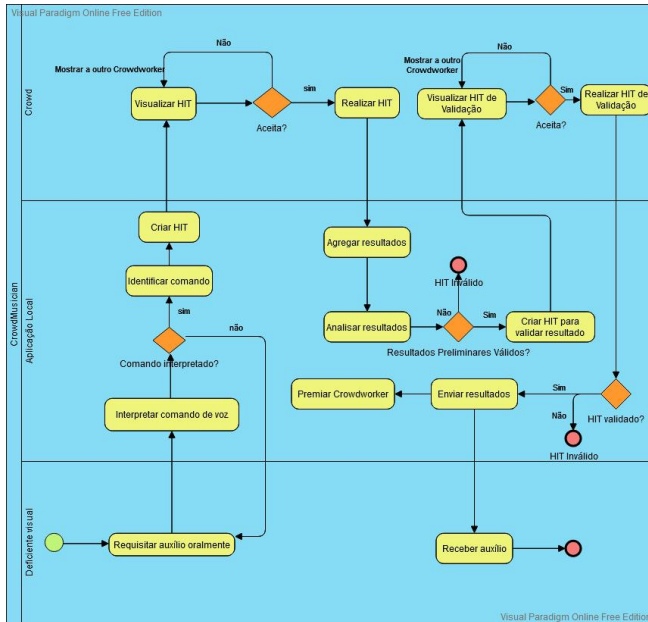


Figura 1 - Workflow do projeto

Antes de realizar qualquer desenvolvimento de código é importante entender o problema e como o resolver. Para isso criámos o nosso workflow no Visual Paradigm Online que descreve as atividades sequenciais que serão realizadas para o desenvolvimento do projeto.

De seguida, e tendo por base os exemplos de código que utilizámos durante as aulas práticas iniciais, iniciámos o desenvolvimento do projeto. Foi desenvolvida uma base de dados local, uma interface em consola que interage com o utilizador, onde este tem a possibilidade de pedir um sample, listar os samples já existentes e reproduzi-los (através de uma biblioteca externa NAudio), via comando de voz ou texto. De seguida, procedemos à criação de microtarefas no MTurk (figuras 6 e 7) onde os crowdworkers submetem ficheiros de áudio correspondentes aos samples requisitados pelo utilizador. Estes samples são, posteriormente, validados por um segundo crowdworker: se o sample for correto é enviado para o utilizador; se estiver incorreto, a resposta é invalidada;

O MTurk é útil porque permite aos empregadores rapidamente obter grandes quantidades de trabalho simples realizado a um custo relativamente baixo. Também é útil para os trabalhadores, pois lhes permite escolher o trabalho que desejam realizar e quanto tempo dedicar a este.

O crowdworker recebe uma recompensa de 0,70\$ na submissão do primeiro HIT e um bónus de 1,00\$ se o sample for devidamente validado. O segundo crowdworker recebe 0,10\$ na submissão do HIT de validação.

Foram também desenvolvidas threads assíncronas para manter sincronizados os resultados obtidos no MTurk.

IV. APRESENTAÇÃO DA TECNOLOGIA

De forma a construir um projeto que implementasse a solução proposta neste papel, foram utilizadas diversas tecnologias e ferramentas.

Inicialmente, o projeto consistiu em uma simples consola baseada no framework C# dotnet core, versão seis,

com reconhecimento e interpretação da voz do utilizador, de forma a suprir a necessidade do público com deficiência visual. Posteriormente, ocorreu o desenvolvimento de mais opções na interface, permitindo ao utilizador escolher entre interfaces de voz ou texto e suporte a ambos. Por fim, de forma a globalizar o uso da ferramenta a múltiplas plataformas e sistemas, onde pode vir a suprir a necessidade de uma gama maior de pessoas, foi desenvolvida uma REST API Web, onde encontram-se as diferentes funcionalidades implementadas, para além de haver documentação automática através do *Swagger*, suportado pelo framework WebAPI dotnet core, que é demonstrada através da figura abaixo.



Figura 2: Documentação da API via Swagger

Por forma a manter a aplicação sincronizada com os resultados obtidos no MTurk sem a necessidade de reinicialização, foram implementadas threads assíncronas que funcionam em *background*, garantindo que a base de dados local se encontra atualizada e sincronizada com os resultados em nuvem. Tal atualização ocorre a cada dez segundos. Nos casos de haver novos resultados sugeridos e posteriormente validados pela crowd, a aplicação emite um evento para quaisquer participantes registados e adiciona-o aos dados locais, garantindo assim maior consistência do sistema.

A aplicação com interface em consola permite ao utilizador solicitar novos samples, listar e tocar os samples recebidos e validados, via comandos de voz ou texto. Ao solicitar novos samples, a aplicação carrega o formulário que será enviado aos crowdworkers, contendo como parâmetro o instrumento desejado. Este parâmetro está limitado a um grupo de oito instrumentos, nomeadamente *electric guitar*, *classic guitar*, *piano*, *bass*, *violin*, *drums*, *saxofone* e *flute*. Este HIT (figura 6) permite obter os seguintes dados: sample, criador do sample, se o crowdworker sabe tocar ou não o instrumento solicitado e quantos anos de experiência possui. As samples são salvas no armazenamento cloud S3 Bucket, da Amazon, permitindo assim facilitar a gestão, sendo apenas necessário trabalhar com a referência para o ficheiro.

Assim, seguindo o workflow já demonstrado neste ensaio, a aplicação recebe esta nova resposta e automaticamente cria outro formulário (figura 7), desta vez para validar a resposta obtida no primeiro. Neste, é possível obter a confirmação do instrumento musical, se o utilizador conhece o artista e se confirma quem é, bem como a avaliação referente à qualidade de áudio do sample obtido. Se este for confirmado e devidamente validado, o utilizador

recebe o sample e o crowdworker responsável por enviá-lo ganha uma bonificação adicional.

A base de dados local foi criada recorrendo ao EntityFramework e a funcionalidade de tocar o sample foi implementada recorrendo a uma biblioteca externa, a qual permite versatilidade ao funcionar em diferentes sistemas operativos, a NAudio. Esta pilha de tecnologias pode ser encontrada no repositório github do projeto [3]. Para aceder, é necessário solicitar acesso prévio.

Por fim, o projeto foi desenvolvido num repositório no github, plataforma de hospedagem de código fonte com controle de versão utilizando Git. Este novo repositório possui testes automáticos a cada *push* enviado para o *branch* principal da aplicação, facilitando a deteção e tratamento de erros assim que enviados; prática muito comum em métodos ágeis.

A. Design da tecnologia

A tecnologia desenvolvida foi projetada com recurso a diferentes classes. Os diagramas abaixo apresentam seus atributos e correlações umas com as outras, tanto de forma local quanto de conexão com o MTurk.

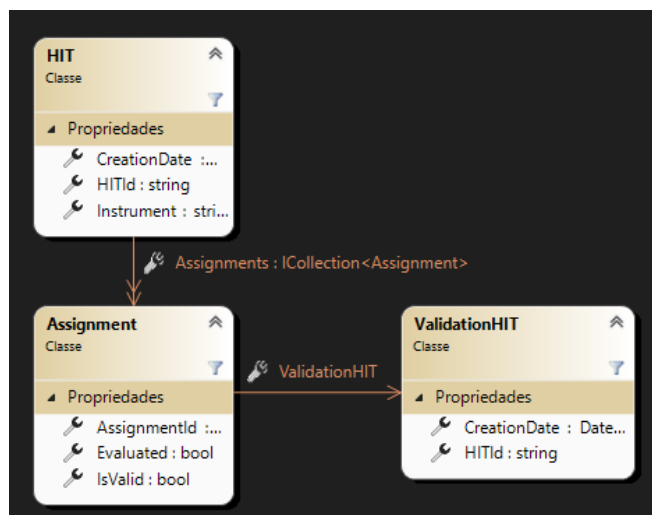


Figura 3: Modelo de Dados Local

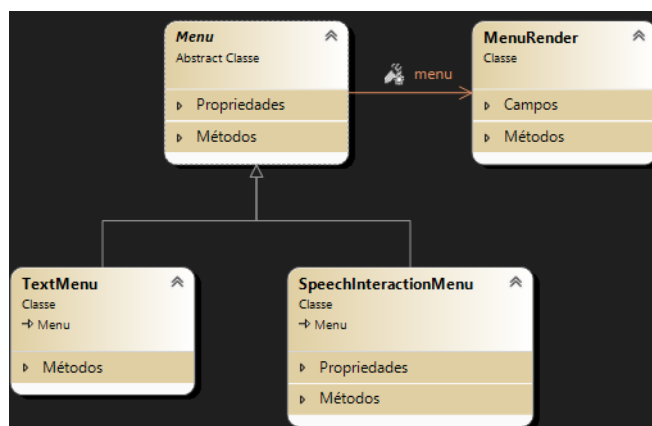


Figura 4: Diagrama de classes dos menus da interface da console

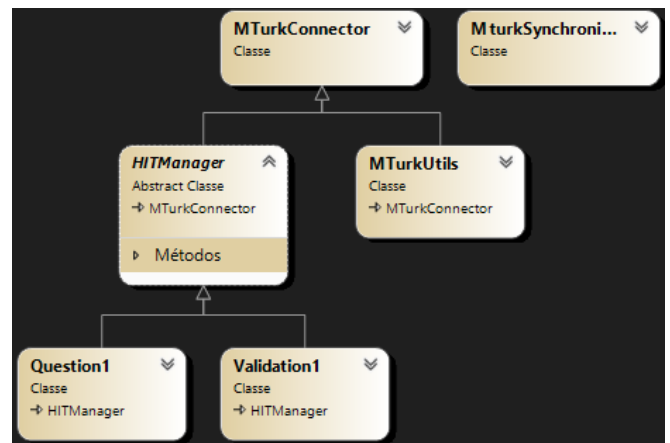


Figura 5: Diagrama de classes dos gestores dos dados do MTurk

V. AVALIAÇÃO DA USABILIDADE

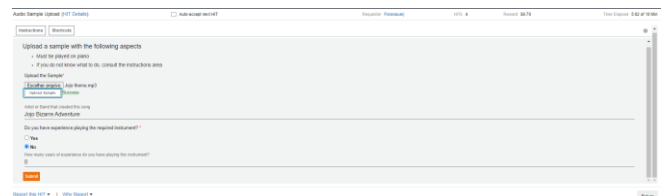


Figura 6: Exemplo de HIT solicitando sample de piano

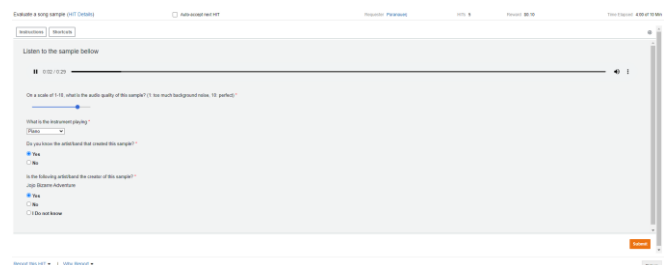


Figura 7: Exemplo de HIT de validação

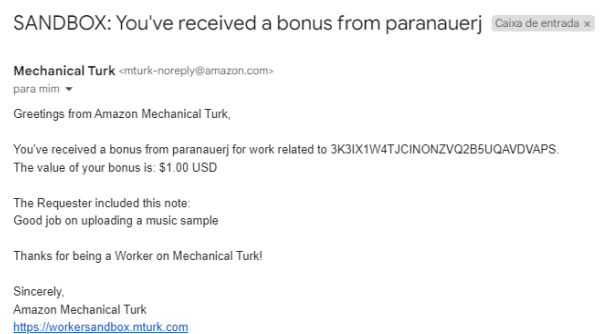


Figura 8: Email enviado a crowdworker após seu sample ser validado

Uma limitação inicial do nosso projeto é a necessidade de termos de instalar o idioma Inglês dos Estados Unidos de maneira pois o reconhecimento de voz por parte da aplicação só funciona neste idioma, nesta fase de desenvolvimento.

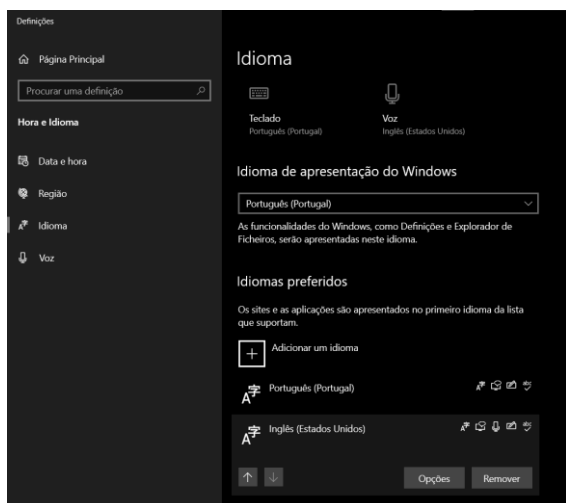


Figura 9 - Instalação do idioma Inglês (EUA)

Nas figuras 6 e 7 estão apresentados exemplos de dois HITs, o primeiro é HIT a requisitar um sample de um piano e o segundo HIT é um HIT de validação, cujo objetivo é validar a resposta do HIT anterior. Neste primeiro HIT, para além de permitir a submissão de um ficheiro áudio correspondente ao pedido do utilizador, permite-nos também obter um conjunto informações sobre o criador do sample submetido, se o *worker* tem experiência com o instrumento e quantos anos de experiência tem. Estas questões servem para despistar incoerência nas respostas, ou seja, a resposta será automaticamente invalidada se o *worker* responder que tem experiência com o instrumento e de seguida responder que tem 0 anos de experiência, por exemplo. No HIT de validação, o *worker*, para além de verificar se o áudio submetido é, de facto, o áudio correto e se o criador do áudio corresponde ao referido no HIT anterior, também avalia a qualidade do sample submetido, de 0 a 10.

A. Participantes

Os participantes envolventes no CrowdMusician são: (1) o utilizador com deficiência visual que requisita um sample através da interface da aplicação; (2) o crowdworker que fornece o sample através no HIT de submissão e, por último, (3) o crowdworker que valida este primeiro HIT num segundo HIT, de validação;

B. Procedimento

A avaliação dos resultados obtidos, i.e., dos samples submetidos, é feita através de uma dupla camada de validação. O HIT de submissão de samples, presente na figura 6, assim como o de validação, presente na figura 7, são validados automaticamente. Portanto, se um resultado for inválido, é rejeitado sem necessidade de validação da crowd. A outra camada de validação, através da crowd, garante que os resultados obtidos são de qualidade. Assim, é possível mapear procedimentos de qualidade para obtenção de melhores resultados.

VI. RESULTADOS E DISCUSSÃO

A. Resultados obtidos

Os resultados obtidos, no contexto do nosso projeto, tratam-se dos samples fornecidos pela crowd aos utilizadores. De lembrar que o utilizador só receberá o

sample que requisitou depois de estarem devidamente validados.

Embora não tivéssemos testado a nossa aplicação com um grande conjunto de pessoas com deficiência visual, ainda assim é possível inferir que o CrowdMusician é um sistema eficaz para a obtenção de samples de qualidade.

B. Discussão

Após corretamente fornecidos os samples ao utilizador, este poderá utilizá-los no ambiente de produção musical da maneira que melhor pretender. Assim verificamos que o CrowdMusician pode ser considerada uma ferramenta com grande potencial, bastante útil no auxílio à produção musical por parte de pessoas com deficiências visuais. Em relação a possíveis aplicações futuras deste projeto, consideramos que possa vir a ter uma utilização em outras áreas para além do apoio a cegos, através de uma rápida integração via API da plataforma.

VII. CONCLUSÃO

Em conclusão, o CrowdMusician demonstrou ser um recurso valioso para permitir aos utilizadores cegos criarem e produzirem música de forma relativamente independente.

Através da criação de campanhas de crowdsourcing, a aplicação foi capaz de fornecer aos utilizadores os elementos musicais necessários para compor e editar músicas de forma intuitiva e acessível.

Além disso, a aplicação também foi capaz de proporcionar uma experiência de utilização intuitiva para utilizadores com visão, sugerindo que pode ter um alcance mais amplo no futuro. Este projeto deu-nos também uma ideia de como esta tecnologia pode ser utilizada para expandir as oportunidades e a independência dos utilizadores cegos no futuro.

AGRADECIMENTOS

Gostaríamos de agradecer aos professores da cadeira de Crowdsourcing: ao Prof. Hugo Paredes, da componente teórica, pela orientação e ensino dos fundamentos teóricos basilares ao entendimento necessário para o projeto; e ao Prof. Diogo Guimarães, pela exposição do tema e acompanhamento prático imprescindível durante todo o semestre. Um agradecimento especial a Universidade de Trás-os-Montes e Alto Douro, em especial a Escola de Ciências e Tecnologia, ECT, por proporcionar esta oportunidade de investigação aos estudantes.

REFERÊNCIAS

- [1] Song, C., Harwath, D., Alhanai, T., & Glass, J. (2022). Speak: A Toolkit Using Amazon Mechanical Turk to Collect and Validate Speech Audio Recordings. Proceedings of the Thirteenth Language Resources and Evaluation Conference, 7253–7258. <https://aclanthology.org/2022.lrec-1.787>
- [2] Vashistha, A., Sethi, P., & Anderson, R. (2018). BSpeak: An Accessible Voice-based Crowdsourcing Marketplace for Low-Income Blind People. Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 1–13. <https://doi.org/10.1145/3173574.3173631>
- [3] Paranauerj/MusicCrowd: Trabalho de Crowdsourcing. ([s.d.]). Recuperado 2 de janeiro de 2023, de <https://github.com/Paranauerj/MusicCrowd>

