

Downtracker - A System for Monitoring People with Down Syndrome at the Labour Market

Alexandre Alves Pereira
Souza

Pontifícia Universidade Católica de
Goiás, PUC Goiás
Goiânia/Go, Brasil

Juliana Santos de Souza
Hannum

Pontifícia Universidade Católica de
Goiás
PUC Goiás

Talles Marcelo Gonçalves de
Andrade Barbosa

Pontifícia Universidade Católica de
Goiás
PUC Goiás

ABSTRACT

Several studies in the literature describe the capacity for the autonomous development of people with atypical development when they are included in the professional environment, becoming individuals capable of making their own decisions. Besides, there are some obstacles to be overcome, such as the family overprotection that the subject is submitted until the individual can achieve his autonomy. Thus, this work presents an Android application called DownTracker, proposed to help monitor people with Down Syndrome (DS) by tracking the individual's real-time location and quantifying emotions and expressions all day long. DownTracker can perform the location tracking of the person with DS to guarantee their safety and autonomy and provide the individual's classification of emotions to help psychologists, teachers, and caregivers learn and assess their emotional state.

Keywords

Tracking; Emotion Recognition; Down's Syndrome.

1. INTRODUCTION

Various applications and technologies developed to provide the ability to track an individual's location and classify emotions. Ozen et al. proposed an Android app for real-time location tracking with an energy-saving mobile device mechanism [9]. Hadjioannou et al. present an application based on tracking location by specific area and triggering particular events based on the user's gesture [10]. The application informs the monitor's place to a list of registrants when the user is outside the configured area or when the user performs a specific movement with the device, carrying out communication between patients and monitoring in emergency cases. Szturm et al. describe the development of an emotion classification system using Afdex SDK, which allows the capture of emotional and behavioral information from people with DS in a learning environment [11]. It was developed to facilitate health professionals to carry out studies on people's emotional state with DS.

Previous works describe systems capable of tracking location or classifying emotions. However, no work has yet been found in the literature capable of monitoring location and classifying emotions for people with DS in one work environment that could favor the user's safety and autonomy.

Pires et al. [6] discuss the difficulties encountered by people with DS in the social and professional environment, discussing the low expectation of skills, family overprotection, and failures to adapt these professionals' qualifications. However, these difficulties are minimized from the moment of inclusion of the subject with DS in the work

environment, promoting their autonomy and cognitive skills, essential factors for the person with DS to obtain a better quality of life [8].

The Pontifical Catholic University of Goiás (PUC Goiás) maintains an extension and research project, called Alfadown, which proposes to help people with DS in activities that promote interpersonal development through technological resources. In the project, activities are carried out according to each participant's needs, help literacy, improve skills, and develop talents, providing more autonomy to these people to carry out their daily activities [7].

This work presents the development of an Android application called DownTracker. The application uses Android's native functionality to monitor individuals with DS by tracking their location in real-time and classifying emotions and expressions by image using the Cognitive Services API. It also allows users to send chat messages to communicate and configure location zones using Google Maps that the user will visit frequently. After configuring a location zone, the user can customize the perimeter of the chosen area. The user's location is continuously monitored using the device's GPS. Suppose such zones are configured in the application. In that case, it is evaluated if the user is within the limits of any of the parameterized zones. If the user is not located in such zones, a message is sent by the application chat to notify the user's monitor. Downtracker has been created in partnership with the Alfadown project to help monitor people with DS to guarantee their safety and provide psychologists, teachers, and caregivers to assess the user's emotional state with inserted DS in the business market.

This section aimed to present the theme, the object of study, and the main justifications that guided this work's accomplishment. Section 2 offers the materials and methods, with a description of the technologies and prototype developed. Section 3 presents the results and the discussion of the tests carried out to validate the system. In section 4, aspects related to the originality, complexity, and usefulness of this work are analyzed. Besides, proposals for future work are presented.

2. MATERIALS AND METHODS

2.1 Application Development

Figure 1 summarizes the process used for the development of DownTracker, based on the incremental bottom-up model. The requirements analysis and elicitation process were carried out through discussions in weekly meetings with all the research group participants. The group has an interdisciplinary background, formed by specialists in Speech Therapy and Computer Science. After collecting the requirements, the

application's functionalities were implemented during the "Implementation of Functionalities" activity, shown in Fig. 1.

With each new functionality completed, white-box tests were performed to find possible errors in logic and coding. Then, in the "Black Box Tests" activity, integration tests were carried out between the application's functionalities to validate the project requirements. In the end, with the validated tests, the software user manuals were documented.

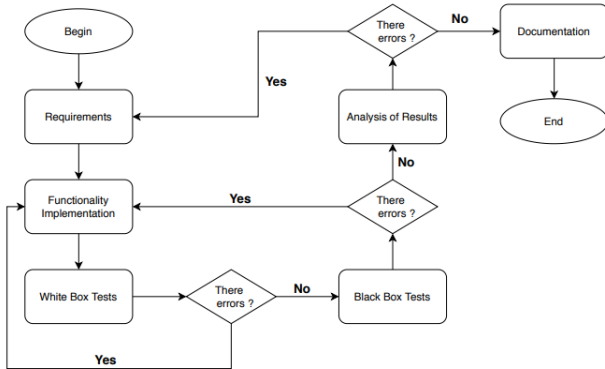


Fig. 1. The application development process

2.2 Prototypes and Technologies

A. Technologies the Visual Studio Code [2] development environment was used For the development of the application. This tool is provided by Microsoft and allows mobile applications that have the Android operating system. The Android operating system was chosen for its greater hardware independence and accessibility due to the lower cost of purchasing a device that uses the Android system [13].

The development SDK chosen for creating the application was Flutter. Its SDK is made available by Google, in an open source way, in which it allows the creation of natively compiled multiplatform applications [1]. It is possible to develop applications for different platforms with code reuse. However, only the Android version was developed and tested because most of the Alfadown participants are Android users.

The Firebase platform was used as the database server. Firebase is a platform made available by Google to create mobile and web applications that facilitate rapid integration and configuration of a back-end and the application [4]. Firebase is also used as a user authentication server, allowing users to control access to the application with greater security.

The API Cognitives Services [3] was used to obtain the classification of users' emotions. Microsoft's API provides cloud-based cognitive services with REST APIs and SDKs that facilitate integration with mobile applications. The API emotion rating service can score Ekman's seven raw emotions, plus neutral ones [12].

B. Structural Description

The class diagram shown in Figure 7 illustrates the software structure developed in this work, presenting the main components and the interrelationship between them. It also shows the association between the blocks representing the application and the other APIs used in developing the application. The "Application" block contains the classes responsible for the main functionalities. The "Firebase Cloud" comprises the Firebase platform's functionalities available for data persistence between the application. The "Maps API" includes the features implemented by the Google Maps API to

view maps with the user's location. The "Microsoft API" provides the necessary resources for the classification of emotions of individuals with DS.

The "ModulesScreen" class, present in the "Application" block, is responsible for presenting and monitoring the application's screen events, rendering graphics components. The "MainServiceProvider" and "HttpClientRest" classes are responsible for the functionality of sending and reading data stored in the Firebase database and communicating with the services of the emotion classification APIs "CognitiveServices" and location monitoring "GoogleMapsApi" and "GeolocationApi".

The "Application" block also includes a "JobServiceBackground" class, where the user location monitoring algorithm is implemented in the background. Even if the user is not using the app in the foreground on their device, their location will be monitored in real-time. The "JobServiceBackground" class is still responsible for controlling the receipt and sending of notifications between users.

C. Prototype

The prototype was developed based on two user profiles, the monitor profile and the DS monitored profile. Only the monitored user has their location tracked together with the classification of their emotions. In contrast, the monitor profile has access to the location and emotion classification information of monitored users linked to your profile. The application provides the monitored user with the configuration of location zones with perimeters between 100 and 500 meters, as shown in Fig. 2. According to the configured zones, the application's monitoring service can obtain the user's location and infer if it is inside or outside the determined area, using the Haversine formula to calculate the distance between two points in a sphere with longitude and latitude data [14]. When the monitored user is outside the location zone, a notification message is automatically sent to the application's monitor every 5 minutes if the monitored subject remains outside the location zone.

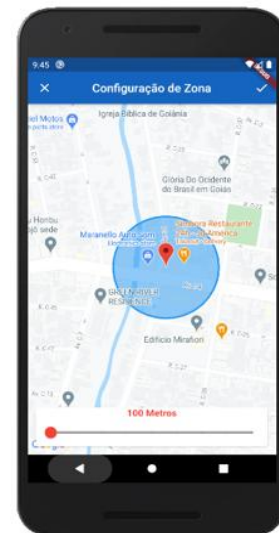


Fig. 2. Location zone configuration screen

The emotion classification functionality of the application is performed, starting from the configuration of customizable questionnaires. When beginning the quiz, the user is asked to register an image of his face to later perform his emotions. It is worth mentioning that the questionnaire is only initialized

after taking the image capture. In this way, the monitor of the person with DS can analyze the user's response and compare it with the classification of the emotion recognized from the photo recorded at the beginning of the questionnaire later, as shown in Figure 3. Figure 4 illustrates the screens for capturing the user's image and presenting the application quiz.

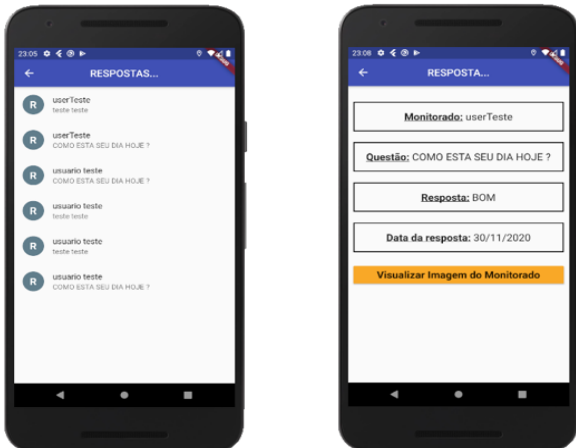


Fig. 3. Visualization of the quiz responses given by the monitored users in the portuguese language

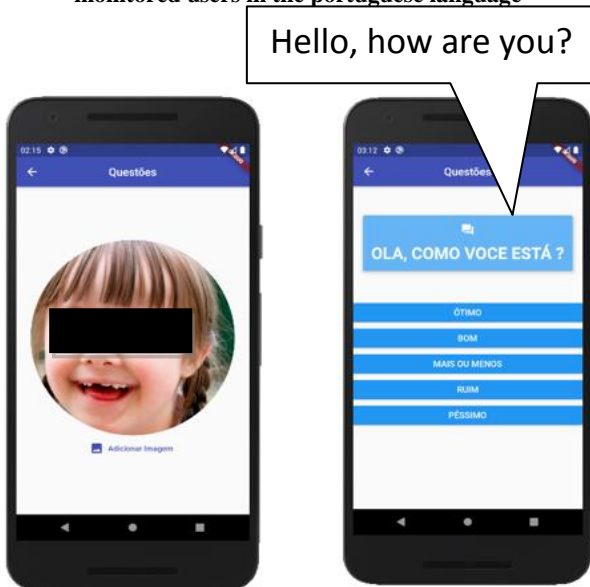


Fig. 4. Initial Data Capturing

D. Tests

To verify the user's location tracking and emotion classification characteristics, due to impediments caused by the COVID-19 pandemic, the tests were raised with the Android emulator's support from IDE Android Studio [15]. The user's location tracking in real time was validated, alternately defining the location points and then analyzed whether DonwTracker would return the correct location, as shown in Figure 5. Also, for testing tracking functionalities, some configured location zones were analyzed in parallel, monitoring the moment when the user is located inside or outside the zone, thus being able to inform the monitor, through notification messages, the instant that the monitored user is not found in any of the defined location zones.

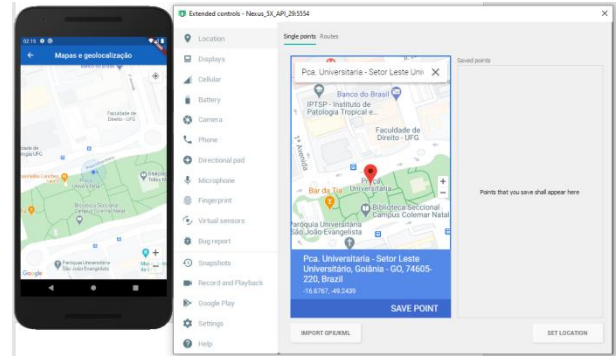


Fig. 5. Evaluating at Android Studio Emulator

The classification of emotions was also tested with the support of the Android Studio emulator. The user's image was registered by the emulator and linked to the responses that this user returned when initializing the application's iterative questionnaire. The application with the monitor user profile linked to the monitored user who answered the questionnaire was accessed. The emotion classification was verified by recording the image obtained from the monitored user and the inference of the answers given in this questionnaire. Figure 6 illustrates the interface presentation of the image with the result of the classification obtained.



Fig. 6. Result of the classification of emotions

3. RESULTS AND DISCUSSIONS

The test of the user's location functionality from the configuration of locality zones was subjected to ten thousand unit iterations, divided into two distinct scenarios: inside the zone and outside the area. Results show 47.51% of correct answers were obtained for the user classified as outside the area, 48.23% of correct answers for the user classified as within the zone, and 4.26% of errors in the classifications. Thus, 95.74% successfully obtained the right category of the user's location, focusing on whether or not he would be located within the perimeter of the zone configured in the application. Errors in the classification of the user's location obtained in the test can be addressed by increasing the accuracy factor in getting the location. However, this will increase the resource consumption of the device.

Figure 8 shows the results obtained after the execution of approximately four thousand test iterations, in which the emotion classification functionality of the application was submitted. The test routines were performed with images obtained from the iCV-MEFED database [5]. This database contains 31250 images of facial expressions with different emotions from 115 people whose gender distribution is almost uniform. The base is composed of the seven raw emotions plus the neutral one. The test results were divided according to

feelings: anger, contempt, disgust, fear, happiness, sadness, surprise, and neutral. The blue columns represent the correct answers in the correct classification of emotions, and the orange columns the percentage of error. For all emotions labeled, the right answers rate is higher than 90%, demonstrating an excellent performance of the correct classification of emotions in still images using the Microsoft Cognitive Services API.

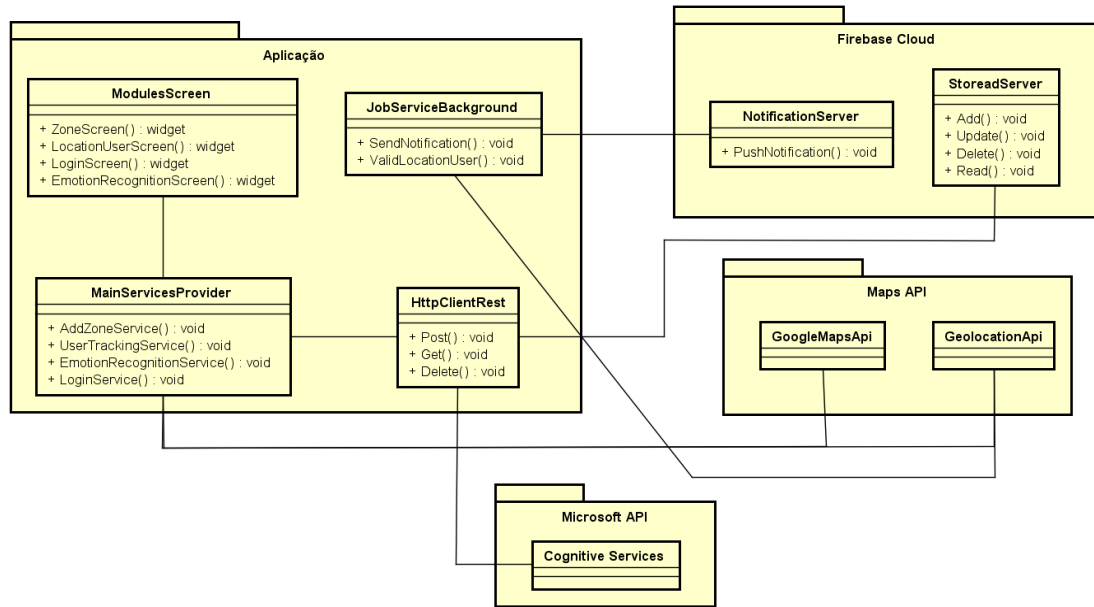


Fig. 7. The Application Components

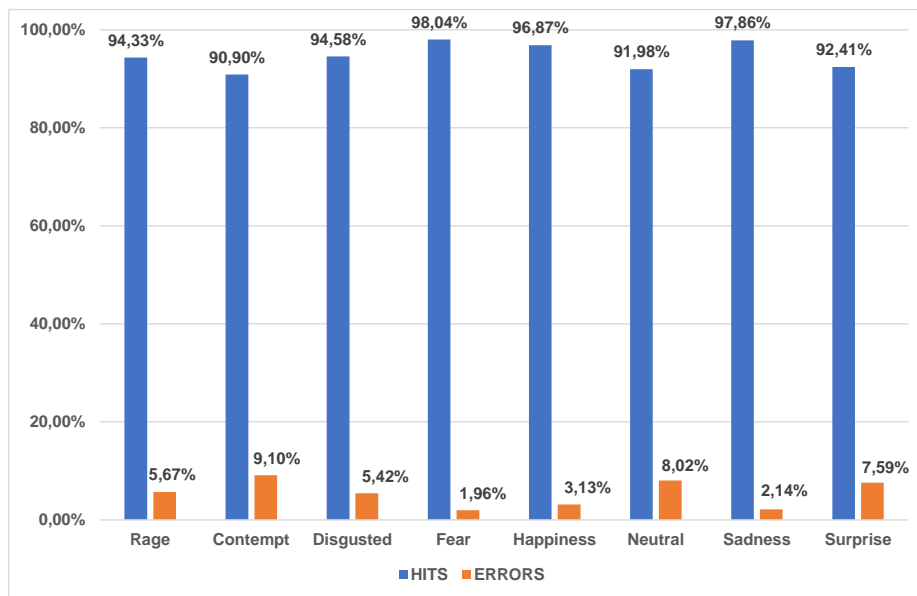


Fig. 8. Correct Answers and Errors in the Pictures Emotion's Classification

4. CONCLUSIONS

This article presents an Android application to perform real-time location tracking and classification of emotions of a person with DS, providing resources to provide autonomy and security. Also, it offers the ability to analyze the emotional state of the user with DS. It cannot assist in the psychological study of people with DS in the workplace.

The prototype tests showed that the communication between the monitor device and the monitored person's device operates successfully.

The classification of the user's emotions is made when the app records users' photographs. When answering the questionnaire, it requires the user to open his camera to perform this procedure. It is possible to automate this

functionality to record the photographs and stream in the background, providing more excellent usability for future work users. The user's location tracking functionality can also be improved to consume fewer mobile device resources, improving the location information methods only when the user moves using the device's native sensors.

The DownTracker project can be accessed at <https://github.com/alexandre-avess/DownTracker>. A version of the application for the Android platform is available at this address.

5. REFERENCES

- [1] Flutter. Available in: <https://flutter.dev/>.
- [2] Visual Studio Code. Available in: <https://code.visualstudio.com/>.
- [3] Microsoft Cognitive Emotion API. Available in: <https://azure.microsoft.com/pt-br/services/cognitive-services/face/>.
- [4] Firebase. Available in: <https://firebase.google.com/>.
- [5] LÜSI, Iris et al. Joint challenge on dominant and complementary emotion recognition using micro emotion features and head-pose estimation: Databases. In: 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017). IEEE, 2017. p. 809-813.
- [6] PIRES, A. B.; BONFIM, D.; BIANCHI, L.C.A.P.; Inclusão social da pessoa com síndrome de down: uma questão de profissionalização. *Arq Ciênc Saúde*, v.14, n4, p.203-210, 2007.
- [7] PUC, Goiás. (2020) [Online]. Available in: <http://sites.pucgoias.edu.br/extensao/coordenacoes/coordenacao-de-extensao/programa-de-referencia-em-inclusao-social/>.
- [8] BARBOSA, T. M. M. F. Et al. Contribuições da Fonoaudiologia na inserção de pessoas com síndrome de Down no mercado de trabalho. *CoDAS*, Paraíba, p. 1-8, jun. 2017.
- [9] OZEN, Yunus; OZDEMIR, Oguzhan; BANDIRMALI, Necla. Android-based energy-aware real-time location tracking system. In: 2015 Seventh international conference on ubiquitous and future networks. IEEE, 2015. p. 842-844.
- [10] HADJIOANNOU, Vasos et al. Context-awareness location-based android application for tracking purposes in assisted living. In: 2016 International Conference on Telecommunications and Multimedia (TEMU). IEEE, 2016. p. 1-7.
- [11] SZTURM, D. R. et al. Emotion Analysis Tool to Support People with Down Syndrome. In: 2018 IEEE MIT Undergraduate Research Technology Conference (URTC). IEEE, 2018. p. 1-4.
- [12] EKMAN, Paul. Basic emotions. *Handbook of cognition and emotion*, v. 98, n. 45-60, p. 16, 1999.
- [13] SOUSA, Marcus Vinicius Girolneto et al. Lori's Help: um aplicativo para auxílio na alfabetização de pessoas com Síndrome de Down. *Revista Brasileira de Computação Aplicada*, v. 9, n. 2, p. 2-13, 2017.
- [14] Yiannos Kryftis, George Mastorakis, Constandinos X. Mavromoustakis, Jordi Mongay Batalla, Evangelos Pallis and Georgios Kormentzas, Efficient Entertainment Services Provision over a Novel Network Architecture, accepted/to appear to the IEEE Wireless Communication magazine, July 2015.
- [16] Android Studio. Available in: <https://developer.android.com/studio/index.html>.