uOttawa

Manager for Audio Notifications Accessibility: User Manual

GNG 2101: Introduction to Product Development and Management for Engineers

Authors

Team B34

[Michael Kagnew, 30011347]

[Fatimah Vakily, 300125671]

[Gabriel Cordovado, 300110852]

[Joseph Francis, 300116730]

[Jasmine Kokkat, 300115249]

[Emma Ballantyne, 300115563]

Faculty of Engineering

University of Ottawa

December 3, 2020

Abstract

This document is a user manual that describes how to operate our final version of Manager for Audio Notifications Accessibility and includes pertinent information for future developers to continue to improve our product. The Manager for Audio Notifications Accessibility is an application that has been created to improve visually impaired users experiences when using their mobile phones by reading their incoming messages aloud when the program is enabled. The program is compatible with the Android operating system with a minimum SDK of 29.0. The application can be downloaded from the Google Play Store and will not function with older phone models. The user must grant M.A.N.A. access to the notifications in setting and can then continue with registering/logging into the app. The application will then run once the 'Start' button is selected, and will cease to read your messages when 'Stop' is selected. The application is capable of being opened by Google Assistant and can read messages in both English and Spanish (one at a time). A common error which may occur is that the user has not allowed special access to notifications for M.A.N.A. in their settings page. There are also various code issues for which troubleshooting has been outlined. Additionally, there are certain safety/privacy issues which are currently associated with our product. Incoming messages will be read automatically regardless of the content which could lead to sensitive information being heard by passersby. Another concerning aspect is that the information being stored in Firebase is not encrypted and is therefore at risk of being stolen, changed, or deleted by others. In conclusion, while our application is functional there are still many aspects which can be improved, such as increasing security and increasing the features available using Google Assistant. These improvements would further aid visually impaired users with using their mobile phones, which is the ultimate goal of our product.

Table of Contents

List of Figures

Introduction

The Manager for Audio Notification Accessibility is an application that has been created to accommodate visually impaired users in their mobile notification experience. Before our product was created the issues that visually impaired users encounter with accessing their incoming messages had been largely overlooked and addressed with ineffective solutions that attempt to solve too many problems with a single product. This is a day-to-day issue, which affects millions of people worldwide, is a prevalent issue that should have solutions available on the market.

Our product has prioritized simplicity to ensure that the product is easy to use. M.A.N.A. values our subscribers and has worked to create the best possible user experience. Issues that occur with similar products, such as VoiceOver, have been taken into account and improved by our development team. This software aims to make an easy solution for visually-impaired users who are looking for a simple and inexpensive way to interact easily with notifications on their phone.

Main Body

Prototype Creation, Installation and Operation

The publicly available prototype for the M.A.N.A is compiled for devices running on the Android Operating system with a minimum SDK of 29.0. It is important that devices attempting to run M.A.N.A meet the required SDK level to ensure features including text-to-speech and translation are functional. For the average consumer, M.A.N.A can be downloaded onto your device using the Google Play Store. This will ensure your device contains the required tooling used by the application, so you do not have to worry about meeting daunting software requirements.

If you find the Play Store is stopping you from downloading M.A.N.A, it is possible that your device has not been updated to the required level. While this might seem unnecessary, M.A.N.A takes advantage of many features newly implemented into the Android Operating System, that are not included in older versions. We recommend you check your device settings for latest updates as it is possible automatic updates have been frozen or neglected. If this is not the case, visit your phone manufacturers website to verify your device has been approved to run at APK 29.0 or above as older hardware may not be sufficient to run the latest versions of the Android OS.

After installing M.A.N.A, before we can proceed to account setup, we need to enable one last setting that allows the application to extend the Android Notification Center. Without this, the application cannot persistently monitor incoming notification to your device and cannot read these out load unless the app is open. In order to become functional, the M.A.N.A requires special-access to notifications. This can be found by going into your Android Settings>Special Access>Notifications and granting M.A.N.A permission.

Once the application is installed and has been given special access to notifications. Opening M.A.N.A prompts the main-activity which displays the ability to start, stop and configure the application. For completing the setup of the app, we are focused on the top-right quadrant of the display, featuring the ‘login’ and ‘log-out’ buttons. Clicking the ‘login’ button will bring you to the sign-in and registration screen that will allow you create an account with the M.A.N.A app. This account is required by the application in-order to store your personal preferences of readable 3rd party applications and language settings that can be used across multiple devices. Another important feature you gain with a M.A.N.A account, is the ability to store past notifications across multiple devices and access them at any time. You can complete your account registration by entering you preferred email and password which can be used any-time you want to access your customizations of the application.

At this point, you have successfully installed and configured the M.A.N.A application. In order to start extending the functionality of the Android Notification Center by reading out notifications, you can click the ‘Start’ button on the main activity. While enabled, you can close the application or lock your device without compromising the functionality of the M.A.N.A. If you wish to disable notification-reading functionality, you must click the ‘Stop’ button on the main-activity once more.

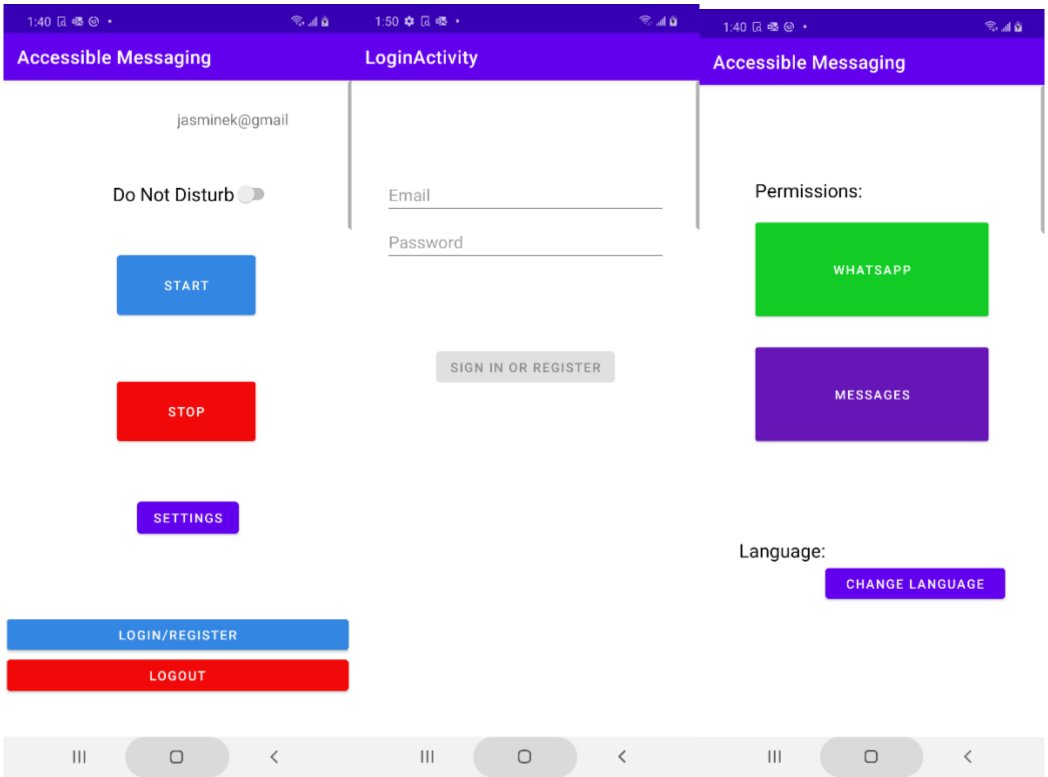


Figure 1: Final Prototype Design

Product Functions and Capabilities

The application is able to be opened using Google Assistant and can start listening for messages when the start button is clicked. When a notification appears on the screen, the name of the sender, the application the message was sent from and the message itself is read out loud to the user. The messages can be read in two languages, English and Spanish, by enabling this in the settings page. The application can only read one language at a time so the user must pick the language of their preference prior to receiving the notification. In order to save the permission settings selected, the user must login so that this information gets saved to the database. If they’re not an existing user, they will be directed to register for an account.

The application is currently being expanded to be more dynamic and include Google Assistant features to open the Settings page and the development team hopes to expand this to include Google Assistant being able to read missed notifications. These features are beyond the scope of our prototype and are features that would further improve our application.

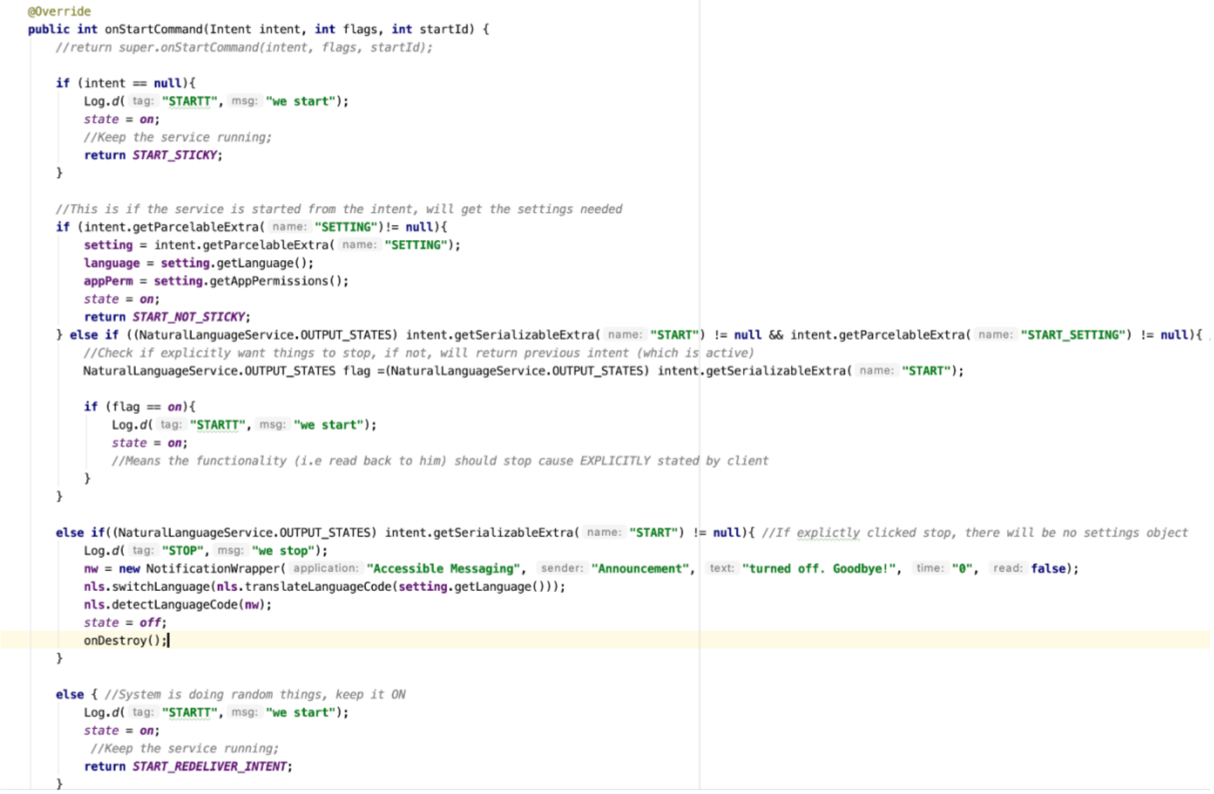
Troubleshooting

Due to the somewhat experimental nature of the application, a few issues could arise regarding the reading of notifications by the application and speaking out the text by voice. This section will start by explaining the less complicated, to more complex problems that could occur, and possible solutions to handle them.

The most likely issue to occur is the application not connecting to the status bar notifications system of android. The most likely reason would be the "special access" setting for the AccessibleMessaging app disabled in the settings of the phone that is being used. It is unclear if every android model has the same setting, but most likely, travelling to settings then searching for "special access" > notification access and enabling the app there will solve the inability to read incoming notifications.

The next possible issue is the lack of text-to-speech functionality. Once it is determined that the application is actually reading notifications by making log statements to check in the NotificationService class, then the first check is to ensure that the NaturalLanguageService is instantiated in the onCreate() function, otherwise the service will not properly start up.

An issue could arise where the onStartCommand() function of the NotificationService class does not properly register start and stop button selection. This issue will require a bit more checking between the activity that currently hosts the button, and if the onStartCommand is properly receiving the intent. In the activity with the start/stop button, one should make sure the service is being started with the startService() function. The stop button also uses this function as well because in our implementation, there was no way we could find a method in order to completely stop the application from listening to incoming messages explicitly, and as well as keeping the app running when Android randomly decides to stop it. Therefore, log statements will be the developer's best friend in determining where the issues arise.

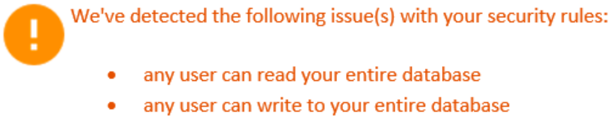


```
@Override
public int onStartCommand(Intent intent, int flags, int startId) {
    //return super.onStartCommand(intent, flags, startId);

    if (intent == null){
        Log.d( tag: "STARTT", msg: "we start");
        state = on;
        //Keep the service running;
        return START_STICKY;
    }

    //This is if the service is started from the intent, will get the settings needed
    if (intent.getParcelableExtra( name: "SETTING")!= null){
        setting = intent.getParcelableExtra( name: "SETTING");
        language = setting.getLanguage();
        appPerm = setting.getAppPermissions();
        state = on;
        return START_NOT_STICKY;
    } else if ((NaturalLanguageService.OUTPUT_STATES) intent.getSerializableExtra( name: "START") != null && intent.getParcelableExtra( name: "START_SETTING") != null){
        //Check if explicitly want things to stop, if not, will return previous intent (which is active)
        NaturalLanguageService.OUTPUT_STATES flag =(NaturalLanguageService.OUTPUT_STATES) intent.getSerializableExtra( name: "START");

        if (flag == on){
            Log.d( tag: "STARTT", msg: "we start");
            state = on;
            //Means the functionality (i.e read back to him) should stop cause EXPLICITLY stated by client
        }
    }

    else if((NaturalLanguageService.OUTPUT_STATES) intent.getSerializableExtra( name: "START") != null){ //If explictly clicked stop, there will be no settings object
        Log.d( tag: "STOP", msg: "we stop");
        nw = new NotificationWrapper( application: "Accessible Messaging", sender: "Announcement", text: "turned off. Goodbye!", time: "0", read: false);
        nls.switchLanguage(nls.translateLanguageCode(setting.getLanguage()));
        nls.detectLanguageCode(nw);
        state = off;
        onDestroy();
    }

    else { //System is doing random things, keep it ON
        Log.d( tag: "STARTT", msg: "we start");
        state = on;
         //Keep the service running;
        return START_REDELIVER_INTENT;
    }
```

Figure 2: OnStartCommand Code for Start/Stop Functionality

Health and Safety

Though there are no health concerns related to usage of this product, the nature of the app produces safety concerns in regards to privacy and security. The M.A.N.A. app is designed to read aloud all notifications regardless of the sender and content. Though users can slightly adjust the apps whose notifications will be read aloud via the Permissions on the settings page, there is no way for it to tell if the contents of the message are sensitive and they will immediately be announced to everyone within hearing distance while the app is enabled. In addition, to allow the user to check missed notifications, the information read from the app is sent to a Firebase database. This information is not encrypted and is set to public access. The lack of privacy caused Firebase to repeatedly send out the following warning about the current settings.



We've detected the following issue(s) with your security rules:

- any user can read your entire database
- any user can write to your entire database

This means that anyone who knows the address of the database used can steal information from, delete information from, and generally change the information stored inside the app's database, which is greatly invasive as it stores information in regards to any notification received by a user.

To improve upon this, it would be beneficial to restrict access to the database information, encrypt the data, and have settings that allow users to restrict messages from certain senders and messages containing chosen words from being announced. In addition, it would be optimal for users to be able to set periods of time where the app would automatically be disabled/enabled to best suit their schedule.

Conclusions and Recommendations for Future Work

In conclusion, our final product for this project was functional and we believe will improve the experience of visually impaired users using their mobile phones. However, there where various issues we encountered which affected the functionality of the application. We were unable to create an application that functioned with IOS and was forced to switch our work to be compatible with Android. Within our Android app there are some key aspects we would have liked to improve if our timeline was extended. An important improvement would be to increase the security of the user's information. This could be done by encrypting the data that is stored in Firebase. Features using Google Assistant could also be increased, such as being able to open the Settings page and to read missed notifications. Overall, we are proud of the product we have created and hope that it can continue to be developed to be as useful as possible for our visually impaired users.

Appendices

All project files and code have been uploaded to our teams makerepo account.
https://makerepo.com/GroupB34MANA/gng2101b34mana

A bill of materials and parts (BOM) is not applicable due to the nature of our prototype and the lack of physical components.