

Faculty of Electronic
Engineering,
Menofia University,
CSE Department.



KidSafe

Android Parental Control App

*A project submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Computer Engineering.*

By

Mahmoud Elsyaed Mohamed Mansour

Khaled Samir Goma Rizk

Supervised by

Dr. Marwa Abbas

7 2019

ACKNOWLEDGMENT

We would like to express our deepest appreciation to all those who provided us the possibility to complete this project. A special gratitude we give to our supervisor, Dr. Marwa Abbas, whose contribution in stimulating suggestions and encouragement, helped us to coordinate our project and inspired us to give our best. Dr. Marwa has shown a great interest and passion about the topic and it was rewarding to us.

Furthermore we would also like to acknowledge with much appreciation to all who helped and motivated us.

UNDERTAKING

This is to declare that the project entitled “KidSafe” is an original work done by undersigned, in partial fulfillment of the requirements for the degree “Bachelor of Computer Engineering” at Computer Science and Engineering Department, Faculty of Electronic Engineering, Menofia University.

All the analysis, design and system development have been accomplished by the undersigned. Moreover, this project has not been submitted to any other college or university.

Student 1

Student 2

ABSTRACT

Smart phones have become essential and this could be both a good or a bad thing. If we look at the positives, having a phone keeps us connected to our friends and family. For safety reasons, it is a boon as it helps us contact people in case we are stranded somewhere or need help. The disadvantage of having a mobile phone is the fact that so many people are so involved in the digital world, that they forget to live, contribute and participate in the present. We agree that smart phones have practically placed the world at our fingertips, but maintaining a balance is essential. Children today cannot even fathom a life pre-internet a life where school work involved visits to libraries and phone calls required you to stay in one spot, since the telephone was attached to the wall. Kids spend an inordinate amount of time on their smart phones, communicating with friends (and possibly strangers) via text, Twitter and Facebook, and work to keep up their Snapstreaks on Snapchat. Even toddlers are proficient in navigating their way around a wireless tablet these days. So we thought of something to provide the parents some control over their children's phones or tablets and gain access to their activity. That's why we developed KidSafe.

Contents

1. Introduction	9
2. Proposal	12
2. 1. Project title	12
2. 2. Project type	12
2. 3. Supervisor	12
2. 4. Project description	12
2. 5. Project features	12
2. 6. Project tools	13
2. 7. Participants	13
2. 8. Project life cycle	13
2. 9. Project source code	13
3. KidSafe in a nutshell	15
3. 1. Reasons to use KidSafe	15
3. 2. KidSafe's features	15
3. 3. License	16
4. Features in details	18
4. 1. Locking and unlocking the child's device	18
4. 2. Screen time control	18
4. 3. App blocking	19
4. 4. Location tracking	19
4. 5. Geo-fencing	19
4. 6. View their activity	20
4. 7. View, call and message their contacts	20
5. Reasons to use Google's firebase	22
5. 1. Firebase's NoSQL real time database	23
5. 2. Firebase's Authentication	26
5. 3. Firebase's Cloud Storage	28
6. KidSafe's design	31
6. 1. Wire-framing	31
6. 2. Signing up	33
6. 3. Login	39
6. 4. Phone locking	41
6. 5. App blocking	42
6. 6. Geo-fencing	43
6. 7. App installed receiver	44
6. 8. App Uninstalled receiver	45

6. 9.	Phone calls receiver	46
6. 10.	SMS receiver	47
6. 11.	Testing	48
7.	User manual	54
7. 1.	Signing up	54
7. 2.	Singing up as a parent	55
7. 3.	Verification	58
7. 4.	Signing up as a child	62
7. 5.	Granting required child permissions	64
7. 6.	Signing up as a child using Google	68
7. 7.	Locking child's device	72
7. 8.	Blocking apps	73
7. 9.	Location tracking	75
7. 10.	Geo-fencing	76
7. 11.	Activity log	77
7. 12.	Settings	80
7. 13.	Protecting KidSafe against uninstalls	88
7. 14.	Password recovery	90
8.	References	92

1

INTRODUCTION

One in four children under the age of six has a smart phone, a survey[1] has found. Despite parents insisting that 11 is the "ideal" age for children to have a phone, a poll found 25 per cent of children aged six and under already have their own mobile and nearly half of these are spending an average of 23 hours a week on smart phones and other gadgets - twice as much time as they spend conversing with their parents, polling suggests.

The survey[2] of 2,000 families with children below the age of 14 found that on average they were spending 3 hours 18 minutes a day on personal devices. By contrast, they were found to be spending 1 hour 43 minutes a day engaged in conversation with members of their family. The survey found four in five parents said they had tried to persuade their children to spend less time on their personal devices. And two in five admitted to giving children devices in order to keep them occupied. Overall, children were found to be spending an average of 23 hours a week isolated on their mobiles, tablets and games consoles at home, almost double the 12 hours they spend conversing with their parents.

Two thirds of parents said they wished they had more family time. The new guidance on screen time and social media follows a review of the evidence about the impact of screen-time on children's mental health and well being. It found that heavy use of social media was associated with a doubling in depressive symptoms.

UK's four chief medical officers suggests parents should take a "precautionary approach" and not allow children to spend more than two hours at a time on smart phones and other gadgets. Prof Dame Sally Davies, England's chief medical officer, said parents should talk to their children about the content of what they are watching, and look out for changes in behavior. She also recommends that parents try using tracking features which measure how much time they and - with permission - their children spend looking at screens, or on social media.

Families are urged to have “screen-free mealtimes” with “adults giving their full attention to children” away from televisions, tablets and smart phones.

The guidance, published by the chief medical officers, also says parents should never

assume a child is happy to have their photo published online and should talk to them about the fact that photographs can be manipulated.

The review found 38.1 per cent of teenage girls spending more than five hours a day on social media suffered from depressive symptoms, compared with rates of 18.1 per cent among those spending between one to three hours on such sites. Levels of depression among teenage boys rose from 6.8 per cent to 14.5 per cent, in the group spending longer on such sites.

From restricting the time they spend on the device, to keeping a close eye on what they are downloading, there are many steps parents can go through to limit usage. Other than making calls and sending messages, it also emerged that 38 per cent of children used their mobile phone to play games.

So we thought of something that can provide parents control over their children’s phones or tablets such as locking and unlocking apps or the device itself. Or set a daily screen time timer when the child exceeds it the device will be locked automatically. And to keep your children’s activity under your sight, we thought we should allow parents to get the real time location of their children. Furthermore, they should be able set a Geo-fence, which if the child exceeds, the parents should be notified. We also will allow parents to access their children’s SMS messages and phone calls.

PROPOSAL

2

The following proposal describes the basic idea of the project entitled “KidSafe”, its features, the tools used in the development, the participants and their roles.

2.1. Project Title: KidSafe – Android Parental Control

2.2. Project Type: Graduation Project

2.3. Supervisor: D. Marwa Abbas

2.4. Project Description:

Many parents believe their teens are too attached to their phones, maybe even addicted. This project aims to keep the child’s activity under control by his parents.

2.5. Project Features:

- Lock/Unlock the child’s smart phone or tablet
- Screen time control
- App blocking
- Location tracking
- Geo-fencing
- View the child’s activity (SMS messages and phone calls)
- View the child’s contacts as well as calling or messaging these contacts

2.6. Project Tools:

All the tools used in the development of KidSafe are free and some of them are open source. That's why we wanted to contribute back to this awesome community by making **KidSafe free and open source under the MIT License**.

- Programming Language: Java
- Back End: Google's Firebase
- Front End: HTML, CSS and JavaScript
- IDE: Android studio and Visual studio code
- Graphics: InkScape and GIMP

2.7. Participants:

1. Mahmoud Mansour - Mobile App Developer
2. Khaled Samir - Mobile App Developer

2.8. Project Development Life Cycle:

Visit this trello board[3] to check the progress of the project.

2.9. Project Source Code:

Visit this github repository[4] for the source code of the project.

3

KidSafe in a nutshell

KidSafe is an android parental control app which when installed on the child's device and linked with his parent's, will provide the parents remote access to that device, giving them the ability to control, block or manage certain features that can ensure your child's safety and digital well being.

3.1. Reasons to use KidSafe:

Parents whose children are spending a lot of time on their smart phones or tablets should use KidSafe to control their children screen time or activity. KidSafe makes it easy to set the boundaries for your child. You can manage their screen time, find out the installed applications, block addictive applications, locate them on real time or set a Geo-fence around them. You can also access their SMS messages, phone calls and view their contacts as well as calling or messaging these contacts.

3.2. KidSafe's Features:

- Lock/Unlock the child's device
- Screen time control
- App blocking
- Location tracking
- Geo-fencing
- View the child's activity (SMS and Phone calls)
- View the child's contacts as well as calling or messaging these contacts

3.3. License:

KidSafe is free and open source under MIT license. A copy of the license is attached below.

MIT License

Copyright (c) 2019 Mahmoud Mansour

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

4

FEATURES IN DETAILS

In the following section we will introduce all the features of KidSafe and how they can help the parents keep the children's activity under their sight and remotely control their devices.

4.1. Locking/Unlocking the child's device

KidSafe allows the parents to lock and unlock their children's smart phones or tablets remotely.



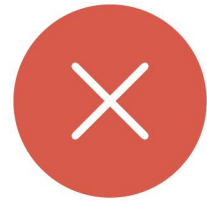
4.2. Screen time control

With screen time control, parents can set a daily usage period of time. If the child exceeded this period, the phone or the tablet locks automatically until the next day.



4.3. App blocking

If your child is kinda addictive to an app or a game, KidSafe allows you to lock and unlock his installed apps with your fingertips. Once you check the switch, if he is online the app will be locked immediately. If he is not online, once he get connected to internet changes will apply.



It is really useful if you want to block app usage during study or sleep time.

4.4. Location tracking

With KidSafe's real time location tracking, you can track your child's location in real time as long as he is connected to the internet. If he is not connected, you will see the most recently location he was at when he was last connected to the internet.



4.5. Geo-fencing

KidSafe allows the parents to set a Geo-fence around their children's current location or the parent's. Once the fence is active, the parent will be alerted when the child exceeds the limits of that fence.



4.6. View their activity

With KidSafe, parents will be able to access their children's SMS messages and phone calls data. Once the child receives a SMS or a phone call it will be forwarded to the parent.

The parent will be able to see the sender/receiver phone number, message body, and sending/receiving time if it is a SMS message.

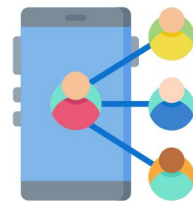


If it is a phone call, the parent will be able to see the call type (outgoing or incoming), caller or called phone number, contact name if the number is saved in your child's contacts the duration of the call, and the time when this call was received or made by the child.



4.7. View, call and message their contacts

Once kidsafe is installed on both the child's and the parent's phones. The parent will have access to all the child's contacts and he can message or call them with just a click of a button.



5

Reasons to use Google's Firebase

Google's firebase is a free **back end as a service** (BaaS) meaning there is no server infrastructure needed. This shortens development time and removes a layer of complexity for developers. The best thing about BaaS, though, is that it frees developers from the tedium of building out a back end. Instead, they can direct all of their focus to creating dynamic, user-oriented apps.

One of the Firebase's biggest draws is its **robust, well-tested** feature set. It has tools for nearly everything a developer could need. Some, like Googleanalytics, are built in free. Other can be incorporated as needed, such as:

- Real-time NoSQL database
- Authentication
- Hosting
- Push notifications
- Real-time messaging
- Cloud storage
- Performance monitoring
- Machine learning kit
- And many other features

KidSafe uses Google's firebase as a back end. We use firebase's real time database, firebase's Authentication and firebase's storage. We shall take about them in the following section.

5.1. Firebase's NoSQL Real-time database

KidSafe store and sync data with firebase's NoSQL cloud database. Data is synced across all clients in realtime, and remains available when KidSafe goes offline.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. All the clients share one Realtime Database instance and automatically receive updates with the newest data.

Key capabilities

Real-time: Instead of typical HTTP requests, the Firebase Realtime Database uses data synchronization—every time data changes, any connected device receives that update within milliseconds. Provide collaborative and immersive experiences without thinking about networking code.

Offline: Firebase apps remain responsive even when offline because the Firebase Realtime Database SDK persists your data to disk. Once connectivity is reestablished, the client device receives any changes it missed, synchronizing it with the current server state.

Accessible from client devices: The Firebase Realtime Database can be accessed directly from a mobile device or web browser; there's no need for an application server. Security and data validation are available through the Firebase Realtime Database Security Rules, expression-based rules that are executed when data is read or written.

Scale across multiple databases: With Firebase Realtime Database on the Blaze pricing plan, you can support your app's data needs at scale by splitting your data across multiple database instances in the same Firebase project. Streamline authentication with Firebase Authentication on your project and authenticate users across your database instances. Control access to the data in each database with custom Firebase Realtime Database Rules for each database instance.

How does it work?

The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code. Data is persisted locally, and even while offline, realtime events continue to fire, giving the end user a responsive experience. When the device regains connection, the Realtime Database synchronizes the local data changes with the remote updates that occurred while the client was offline, merging any conflicts automatically.

The Realtime Database provides a flexible, expression-based rules language, called Firebase Realtime Database Security Rules, to define how your data should be structured and when data can be read from or written to. When integrated with Firebase Authentication, developers can define who has access to what data, and how they can access it.

The Realtime Database is a NoSQL database and as such has different optimizations and functionality compared to a relational database. The Realtime Database API is designed to only allow operations that can be executed quickly. This enables you to build a great realtime experience that can serve millions of users without compromising on responsiveness. Because of this, it is important to think about how users need to access your data and then structure it accordingly.

5.2. Firebase's Authentication

Most apps need to know the identity of a user. Knowing a user's identity allows an app to securely save user data in the cloud and provide the same personalized experience across all of the user's devices.

Firebase Authentication provides backend services, easy-to-use SDKs, and readymade UI libraries to authenticate users to your app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more.

Firebase Authentication integrates tightly with other Firebase services, and it leverages industry standards like OAuth 2.0 and OpenID Connect, so it can be easily integrated with your custom backend.

Authentication Methodologies

Email and password based authentication: Authenticate users with their email addresses and passwords. The Firebase Authentication SDK provides methods to create and manage users that use their email addresses and passwords to sign in. Firebase Authentication also handles sending password reset emails.

Federated identity provider integration: Authenticate users by integrating with federated identity providers. The Firebase Authentication SDK provides methods that allow users to sign in with their Google, Facebook, Twitter, and GitHub accounts.

Phone number authentication: Authenticate users by sending SMS messages to their phones.

There are a lot of ways in which a user can authenticate using firebase authentication. This makes it easy-to-use for the user. However KidSafe allows both email and password and Google authentication.

How does it work?

To sign a user into your app, you first get authentication credentials from the user. These credentials can be the user's email address and password, or an OAuth token from a federated identity provider. Then, you pass these credentials to the Firebase Authentication SDK. Our backend services will then verify those credentials and return a response to the client.

After a successful sign in, you can access the user's basic profile information, and you can control the user's access to data stored in other Firebase products. You can also use the provided authentication token to verify the identity of users in your own backend services.

5.3. Firebase's Cloud Storage

Cloud Storage is built for app developers who need to store and serve user-generated content, such as photos or videos.

Cloud Storage for Firebase is a powerful, simple, and cost-effective object storage service built for Google scale. The Firebase SDKs for Cloud Storage add Google security to file uploads and downloads for your Firebase apps, regardless of network quality. You can use our SDKs to store images, audio, video, or other user-generated content. On the server, you can use Google Cloud Storage, to access the same files.

KidSafe uses cloud storage to upload the profile images of each authenticated user.

Key capabilities

Robust operations: Firebase SDKs for Cloud Storage perform uploads and downloads regardless of network quality. Uploads and downloads are robust, meaning they restart where they stopped, saving your users time and bandwidth.

Strong security: Firebase SDKs for Cloud Storage integrate with Firebase Authentication to provide simple and intuitive authentication for developers. You can use our declarative security model to allow access based on filename, size, content type, and other metadata.

High scalability: Cloud Storage for Firebase is built for exabyte scale when your app goes viral. Effortlessly grow from prototype to production using the same infrastructure that powers Spotify and Google Photos.

How does it work?

Developers use the Firebase SDKs for Cloud Storage to upload and download files directly from clients. If the network connection is poor, the client is able to retry the operation right where it left off, saving your users time and bandwidth.

Cloud Storage stores your files in a Google Cloud Storage bucket, making them accessible through both Firebase and Google Cloud. This allows you the flexibility to upload and download files from mobile clients via the Firebase SDKs, and do serverside processing such as image filtering or video transcoding using Google Cloud Platform. Cloud Storage scales automatically, meaning that there's no need to migrate to any other provider. Learn more about all the benefits of our integration with Google Cloud Platform.

The Firebase SDKs for Cloud Storage integrate seamlessly with Firebase Authentication to identify users, and we provide a declarative security language that lets you set access controls on individual files or groups of files, so you can make files as public or private as you want.

KidSafe's Design

6

This chapter will illustrate how most of the operations in KidSafe work. We will see how locking the child's phone, blocking apps, Viewing SMS & call logs, location tracking and Geo-fencing actually works

Before starting developing and designing the app some serious steps were taken like gathering the requirements and see what of them can actually be done and what can't be done.

6.1. Wire-framing

Wire-framing is one of these steps where we simply draw how our app should look like. At first we had 8 screens in mind and they are, login screen, parent signed in screen, apps screen, location screen, calls log screen, SMS log screen, contacts screen and the settings screen. And we imagined them as in Fig.6.0 starting from screen 1 to screen 8 accordingly.

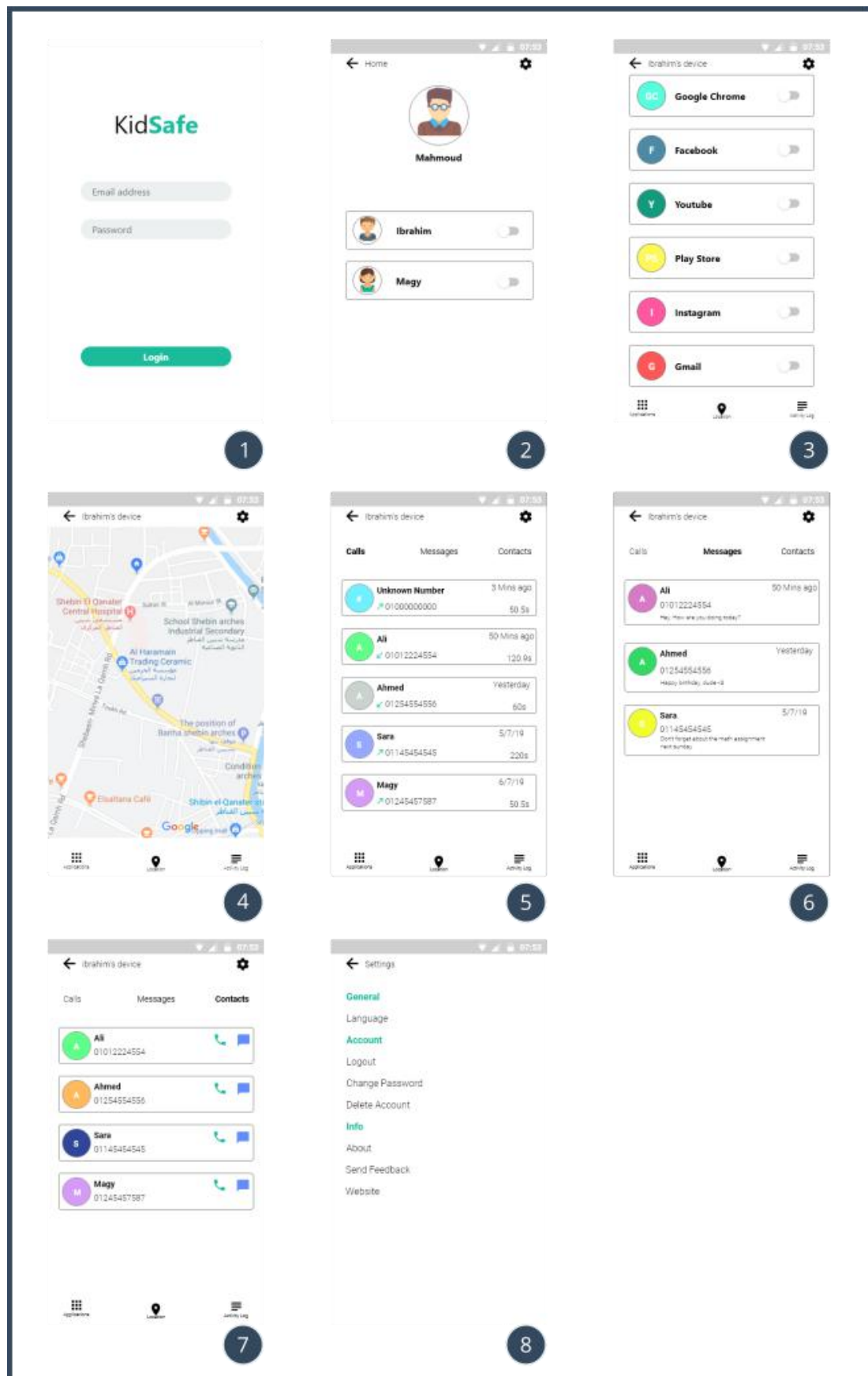


Fig.6.0
KidSafe's wireframes

6.2. Signing Up

We will discuss how you can sign up as a user later in chapter 6 “User Manual” but right now, lets know how signing up affects firebase’s database, storage and authentication.

Fig.6.1 shows the flowchart of signing up as either a parent or a child. Notice when signing up as a parent, a name, an email and a password are required. But when you sign up as a child, a name, an email, a password and a parent email is required to link this child to him.

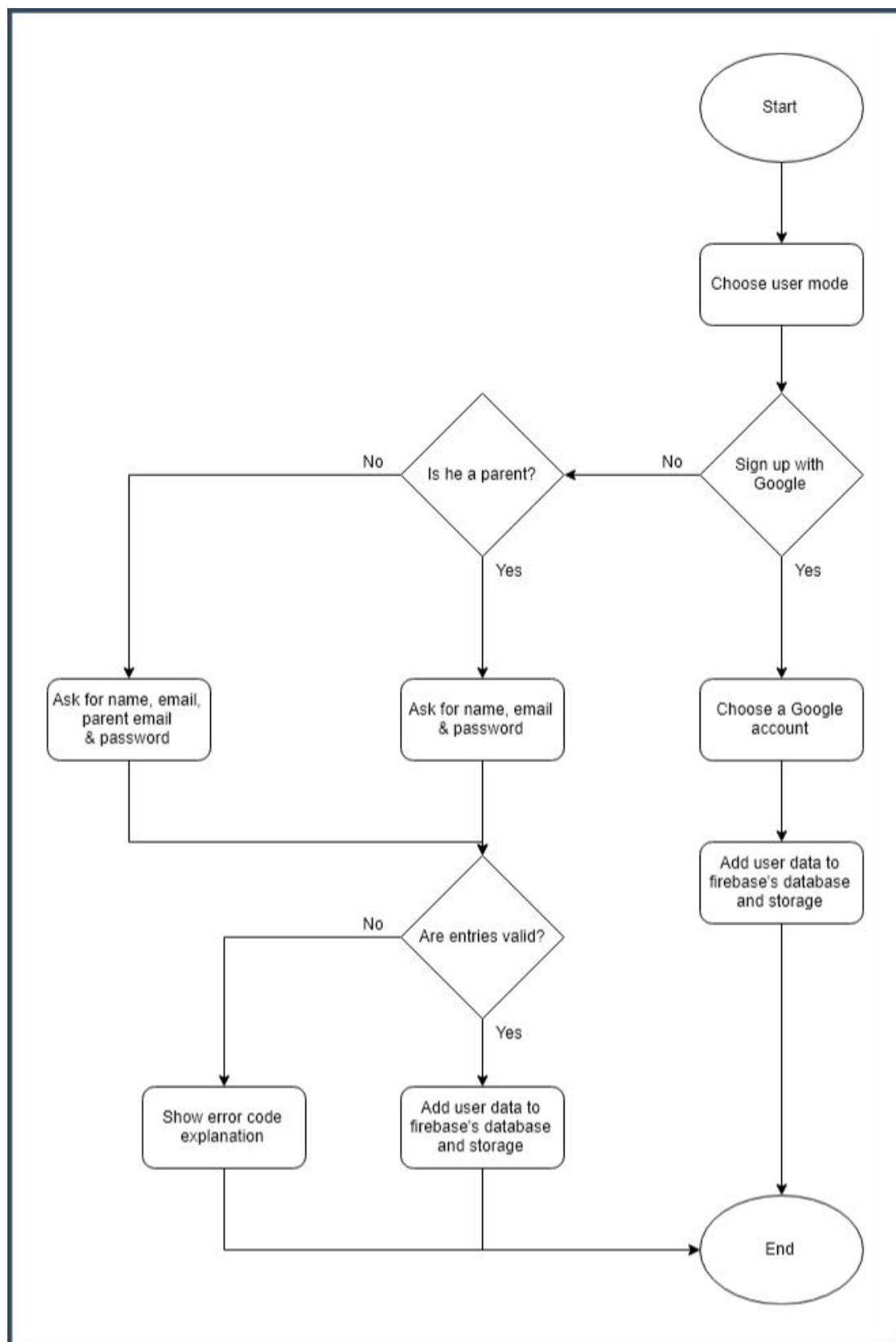


Fig.6.1
Singin up flowchart

Fig.6.2 Shows the real time database of the project entitled “KidSafe” with a kidsafe-3a667 as the firebase id.

It also shows that there is currently 3 users, 1 in the parents section with a unique id equal to **dxnXjv8ExleLd8hvJuDjh63JzN22** and 2 in the children section with ids of **loZEwaIKJvdjDFoDxBUCWrvgm6T2** and **sWqmfbYI0YdsZowmr9wAa0x6Sc63**.

The one in the parents section his email is mahmoud.kidsafe@gmail.com, his name is “Mahmoud”.

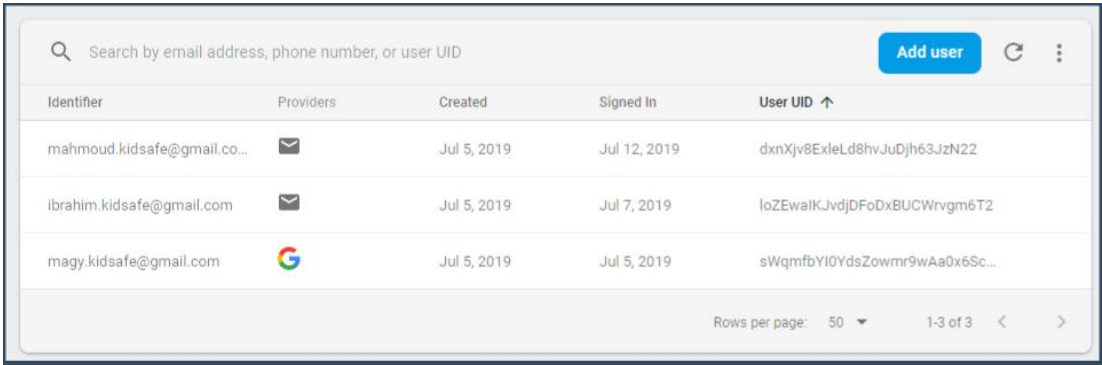
The other two in the childs section, one of them has an email equal to ibrahim.kidsafe@gmail.com and his name is “Ibrahim”. The other one has an email equal to magy.kidsafe@gmail.com and her name is “Magy”



Fig.6.2
How firebase's database changes on
Signing up parents or childs

Fig.6.3 shows the firebase's authentication page. We can clearly see that there are 3 user already signed up.

You can also see in the “Providers” column, that two of them were signed up using email and password and one of them signed up using Google.



The screenshot shows the Firebase Authentication 'Users' page. At the top, there is a search bar with the placeholder text 'Search by email address, phone number, or user UID'. To the right of the search bar are three buttons: 'Add user' (blue), a refresh icon, and a menu icon. Below the search bar is a table with the following columns: 'Identifier', 'Providers', 'Created', 'Signed In', and 'User UID' (with an upward arrow icon). The table contains three rows of user data. The first two rows show users signed up with email and password (indicated by an envelope icon in the 'Providers' column), and the third row shows a user signed up with Google (indicated by the Google 'G' logo). At the bottom right of the table, there is a pagination control showing 'Rows per page: 50' and '1-3 of 3'.

Identifier	Providers	Created	Signed In	User UID ↑
mahmoud.kidsafe@gmail.co...	✉	Jul 5, 2019	Jul 12, 2019	dxnXjv8ExleLd8hvJuDjh63JzN22
ibrahim.kidsafe@gmail.com	✉	Jul 5, 2019	Jul 7, 2019	loZEwaiKJvdjDFoDxBUCWrvgm6T2
magy.kidsafe@gmail.com	🌐	Jul 5, 2019	Jul 5, 2019	sWqmfY10YdsZowmr9wAa0x6Sc...

Fig.6.3
How firebase's authentication changes on
Signing up parents or childs

As shown in Fig.6.4, the profile images were uploaded to firebase’s storage and the link was added to each one in the database as we have seen earlier, so we can refer to it when needed.

We store the profile images in firebase’s storage. The images are stored in the following format `userId_profileImage`.

So the profile image of the parent “Mahmoud” has the following name `dxnXjv8ExleLd8hvJuDjh63JzN22_profileImage`, and since every user has a unique id that can never match with other users. The same applies to the profile images since they use the user id as their name.

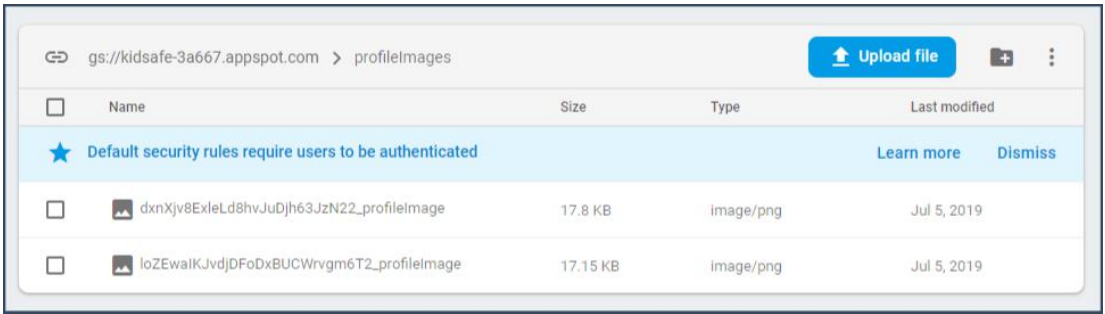


Fig.6.4
How firebase's storage changes on
Signing up parents or childs

6.3. Login

Fig.6.5 shows the flowchart of logging in, once a user is authorized his account is checked wither if he is a parent or a child and a specific behavior is done accordingly.

If he is a parent, we get all the linked children and there data and shown them in Fig.7.10 in the next chapter.

If he is a child, we start the foreground service to monitor the apps, location and register some receivers for SMS, calls and contacts...etc.

However if the user isn't authorized, we show an error with an explanation instead.

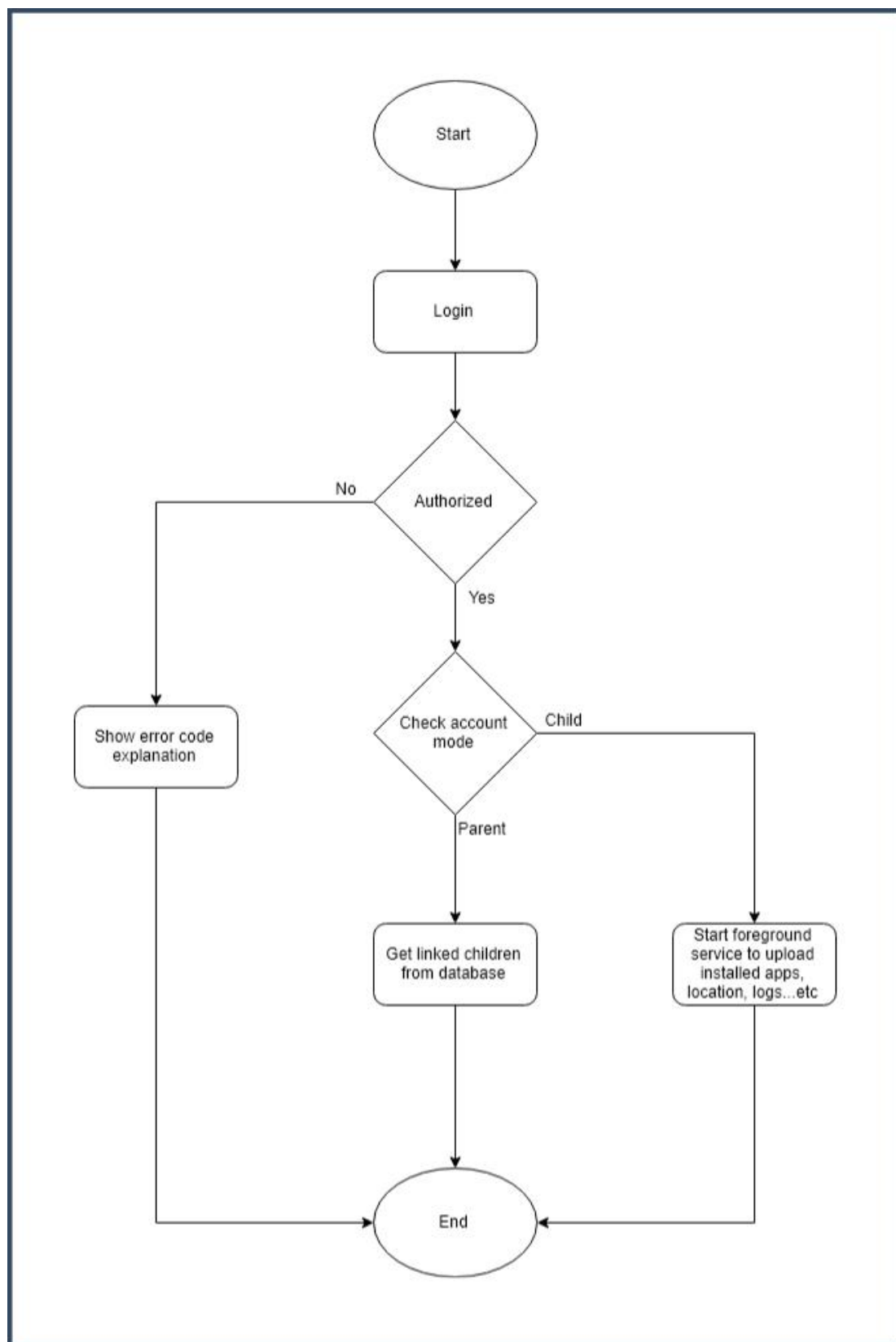


Fig.6.5
Login flowchart

6.4. Phone Locking

Fig.6.6 shows the flowchart of a parent locking the phone of a child, as you can see the parent interacts with ParentSignedInActivity object by clicking the lock phone switch next to the child's name, this shows a dialog asking if you want to lock it immediately or after a period. Whatever the return, we write it in the child's database reference as shown in Fig6.6.

This is explained to the user in Fig.7.11 in the next chapter.

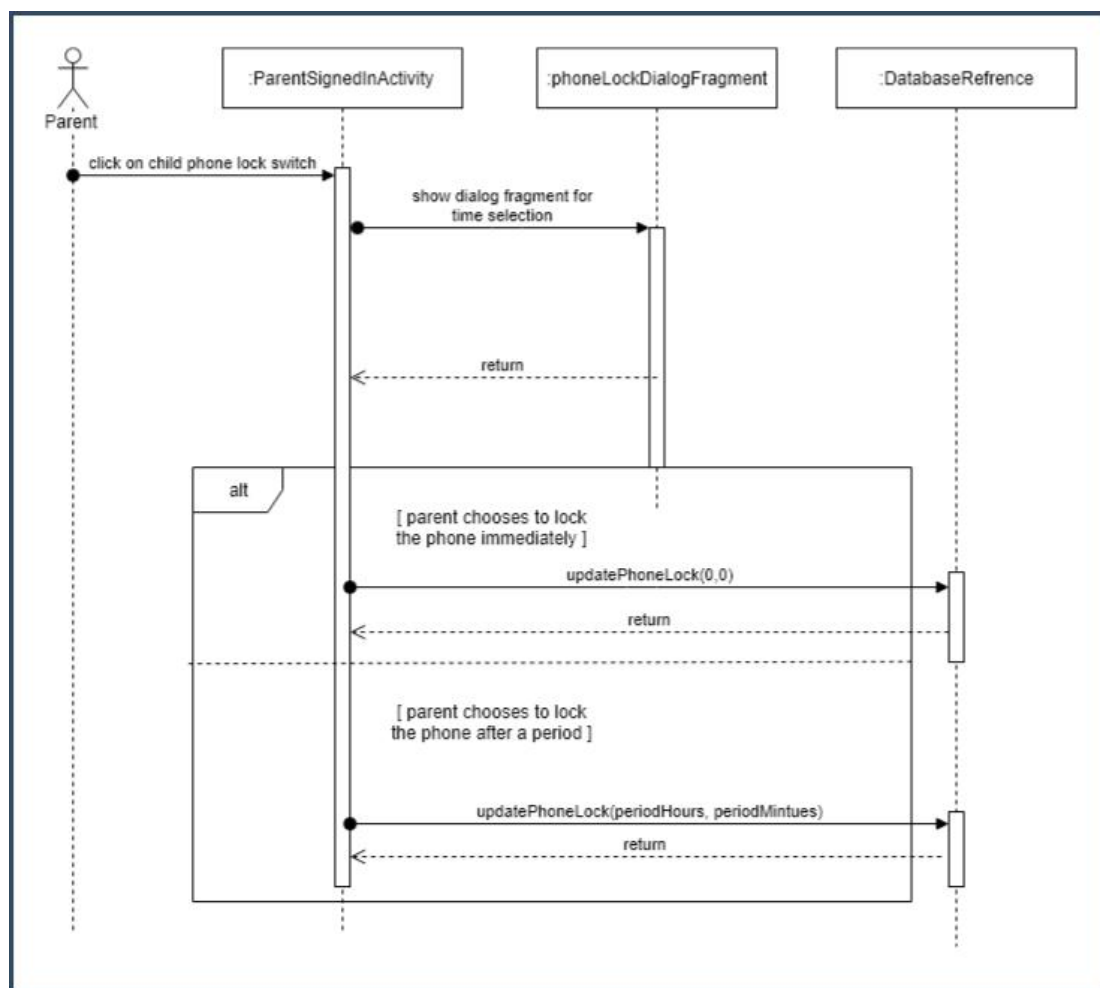


Fig.6.6
Phone locking sequence diagram

6.5. App Blocking

Fig.6.7 shows the flowchart of a parent blocking an app. As shown the parent interacts with the AppsFragment object by clicking on the block app switch next to the app name. If the switch turned to ON state, the app will be blocked. If it turned to the OFF state, the app will be unblocked.

This is explained to the user in Fig.7.12 and Fig.7.13 in the next chapter.

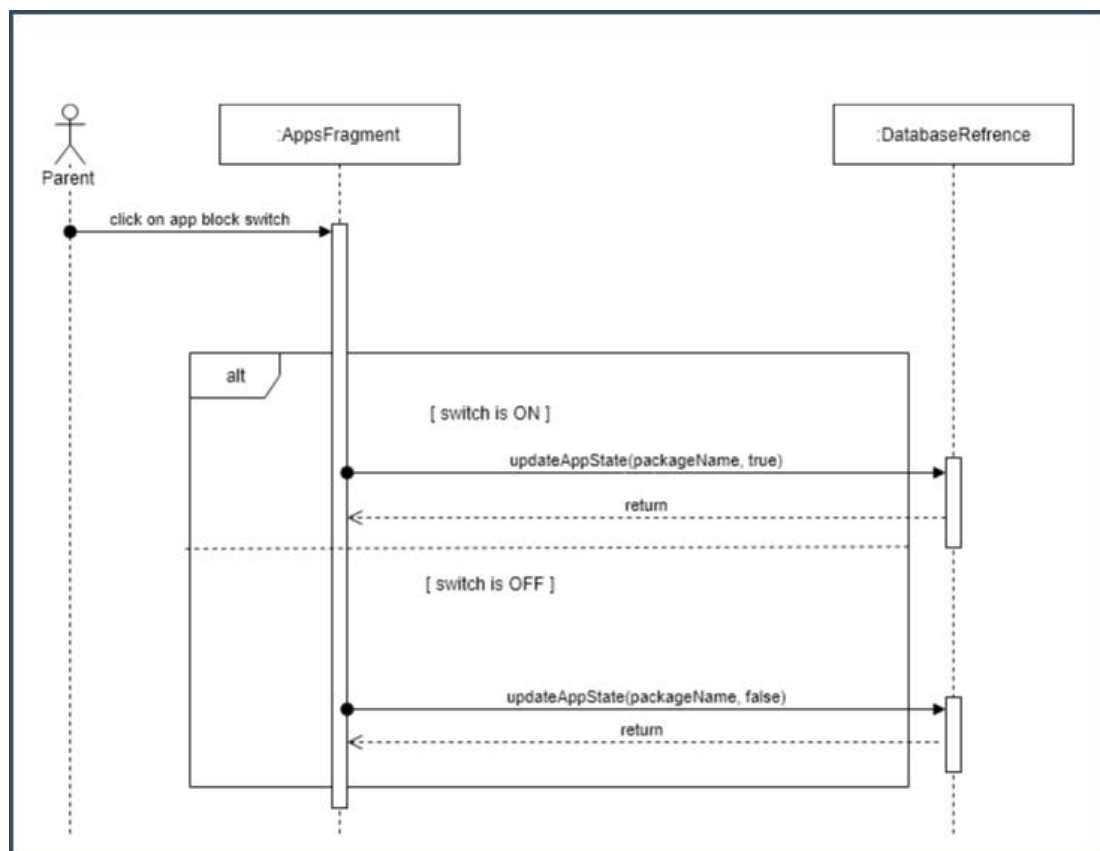


Fig.6.7
App blocking sequence diagram

6.6. Geo-fencing

Fig.6.8 shows the flowchart of a parent setting a Geo-fence on a child. As shown in the figure, the parent interacts with the LocationFragment object by clicking the floating Geo-fencing button, this shows a dialog asking for the fences center and diameter. Whatever the return, we write it in the child's database reference as shown in Fig6.8.

This is explained to the user in Fig.7.15 in the next chapter.

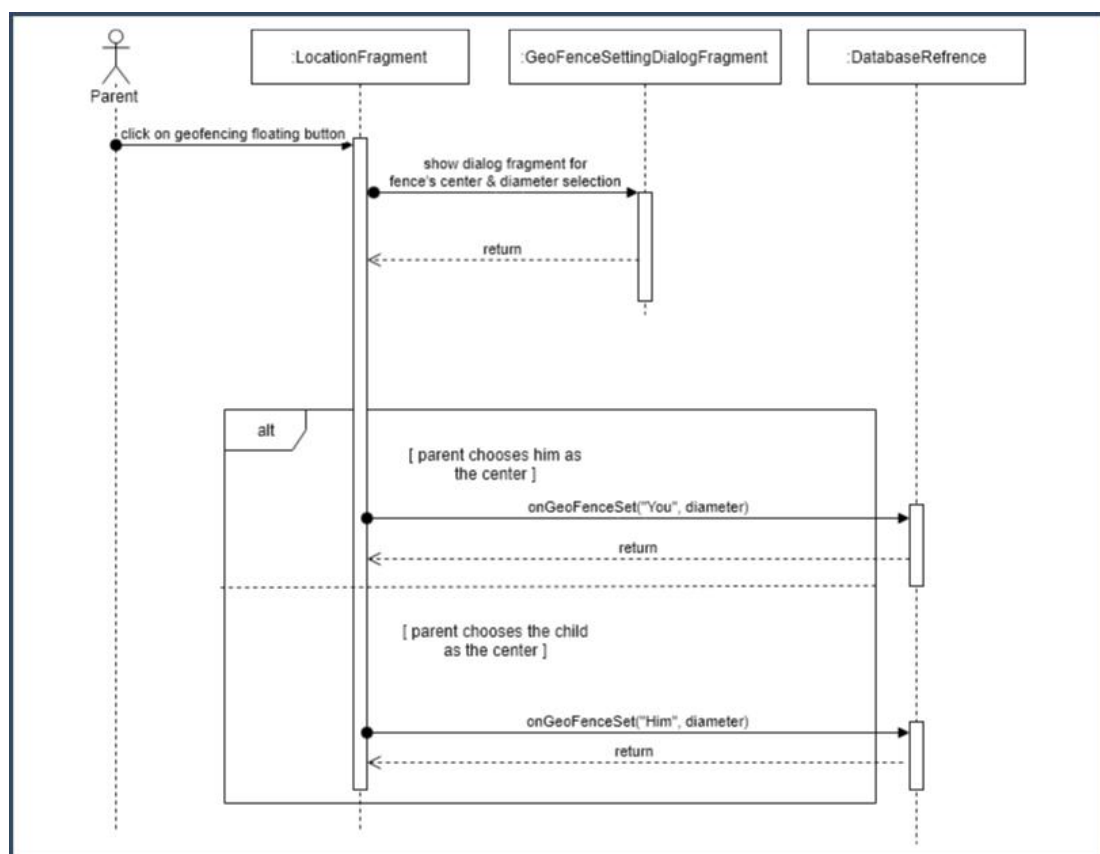


Fig.6.8
Geo-Fencing sequence diagram

6.7. App Installed Receiver

Fig.6.9 shows the flowchart of what happens when a new app is installed on the child's device. As shown in the figure, the child doesn't interact with the app, the MainForegroundService object does the job instead. Once the service starts it registers the AppInstalledReceiver. Which listens for new apps being installed. If there are any, they will be added to the database as shown in the figure below.

The MainForegroundService also registers a OnBootCompleteReceiver which restarts the service if a restart or a shutdown happens. This way we ensure our service and the receivers never stop.

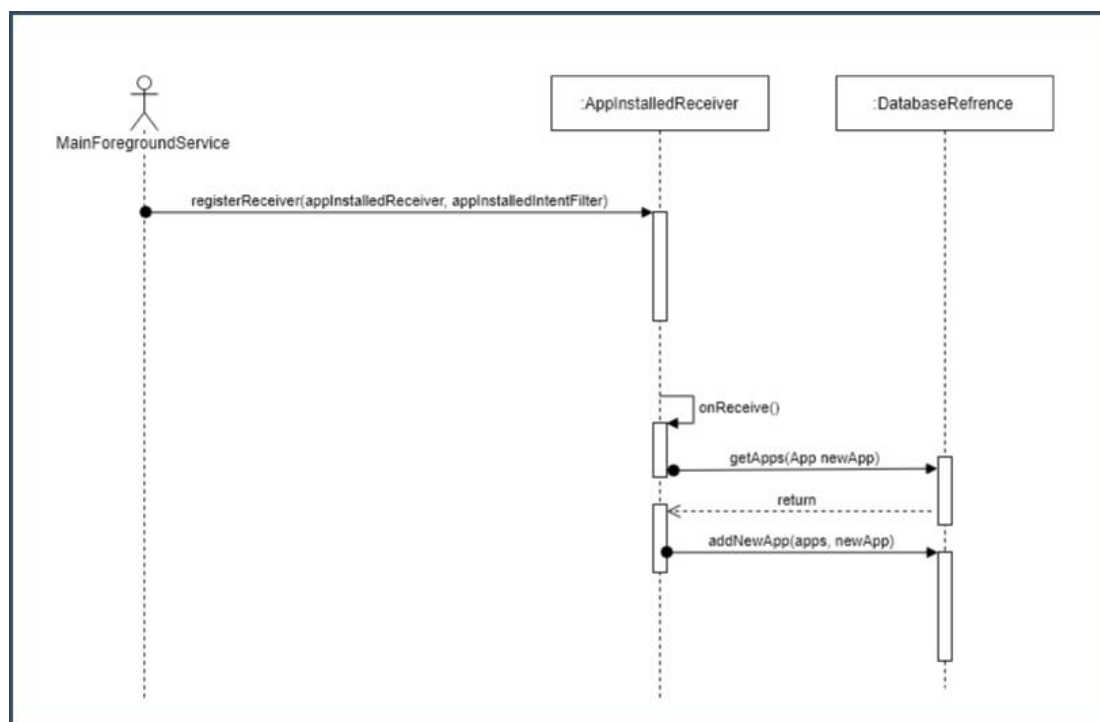


Fig.6.9
App installed sequence diagram

6.8. App Uninstalled Receiver

Fig.6.10 shows the flowchart of what happens when an app is uninstalled on the child's device. As shown in the figure, the AppRemovedReceiver is notified and it removes the app from the database as shown in the figure below.

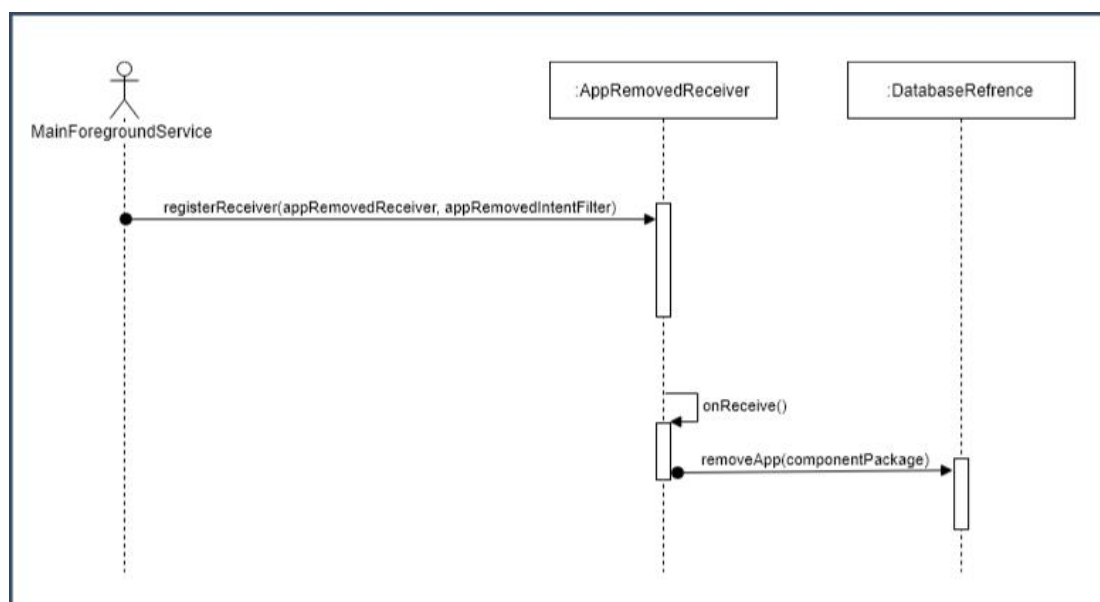


Fig.6.10
App uninstalled sequence diagram

6.9. Phone Calls Receiver

Fig.6.11 shows the flowchart of what happens when a child receives a call. When the PhoneStateReceiver object is notified, it first tries to find the contact name of the caller/reception number, if the number is unsaved, it returns Unknown number. Then it creates a Call object with the contact name, phone number, duration, date and the call type either incoming or outgoing. The receiver then push the call to the database reference as shown in the figure below.

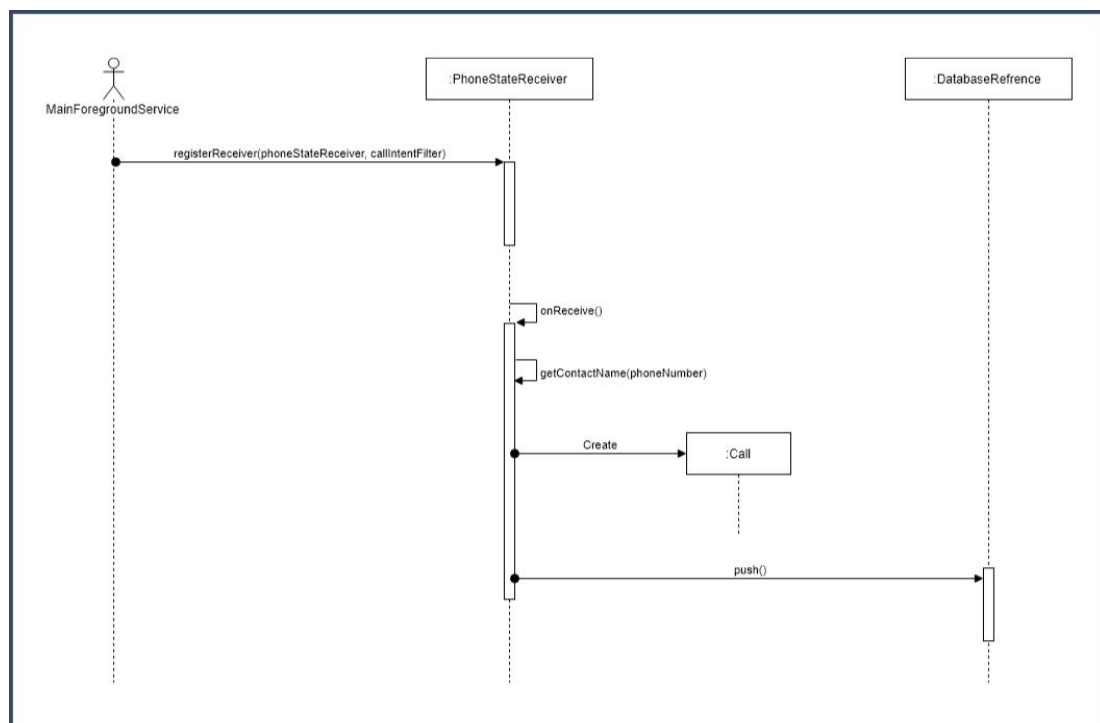


Fig.6.11
Phone calls received sequence diagram

6.10. SMS Receiver

Fig.6.12 shows the flowchart of what happens when a child receives a SMS. When the SmsReceiver object is notified, it first tries to find the contact name of the sender/reception number, if the number is unsaved, it returns Unknown number. Then it creates a Message object with the contact name, phone number, message body and the date. The receiver then push the message to the database reference as shown in the figure below.

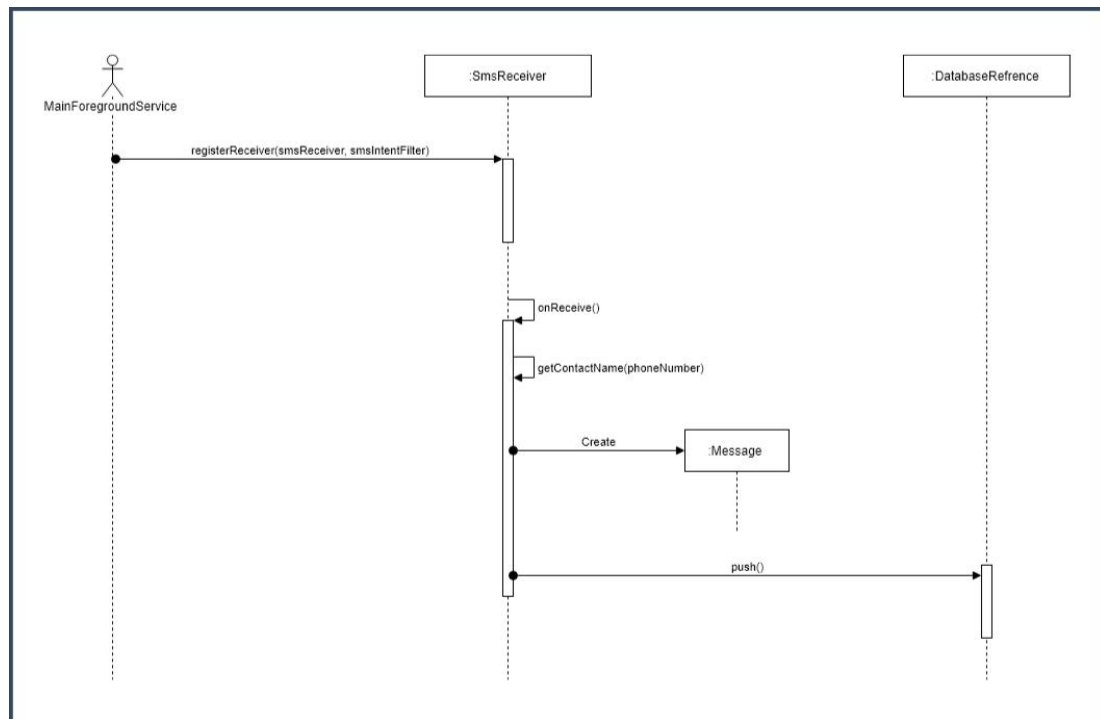


Fig.6.12
SMS received sequence diagram

6.11. Testing

Robo test is a test tool that is integrated with Firebase Test Lab. Robo test analyzes the structure of your app's UI and then explores it methodically, automatically simulating user activities. Unlike the UI/Application Exerciser Monkey test, Robo test always simulates the same user activities in the same order when you use it to test an app on a specific device configuration with the same settings. This lets you use Robo test to validate bug fixes and test for regressions in a way that isn't possible when testing with the UI/Application Exerciser Monkey test.

We used Robo test 4 times on different devices with different APIs, All of them were successful. As shown in the following figures.

If the below images weren't clear, you can find them here[5]



The screenshot shows the Firebase Test Lab interface for an app named 'KidSafe'. At the top right is a blue button labeled 'Run a test'. Below this is a table with the following columns: 'Test matrix', 'Test type', 'Started', 'Total executions', and 'Issues'. There are two rows of test results, both marked with a green checkmark in the 'Test matrix' column, indicating they passed.

Test matrix	Test type	Started	Total executions	Issues
matrix-1l8tdpu09sqgp	Robo	13 hours ago	3	—
matrix-1u1qmaqqdwi9	Robo	8 days ago	1	—

Fig.6.13
The 4 passed robo tests

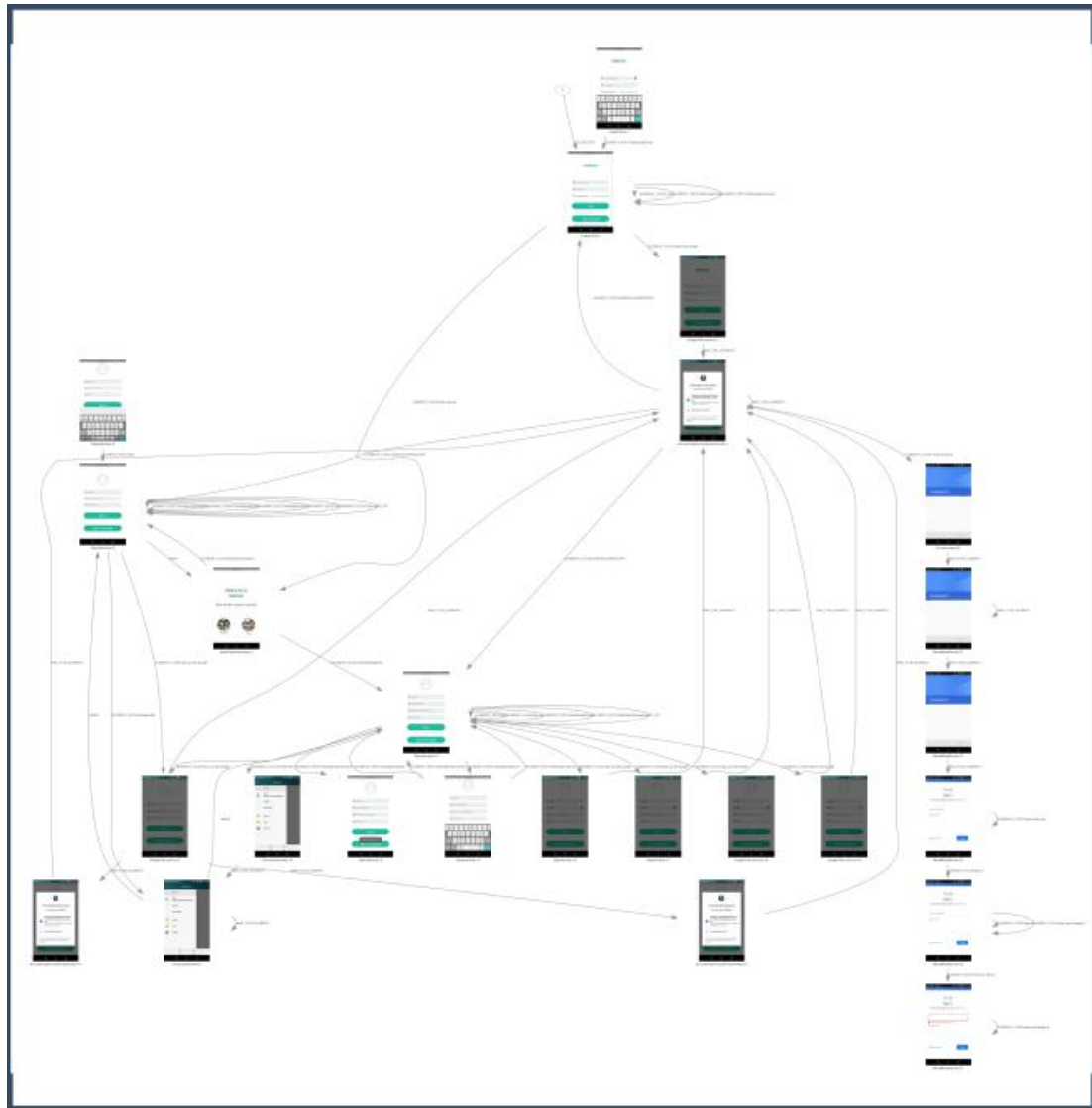


Fig.6.14
Robo test, Huawei P8 lite, API Level 21

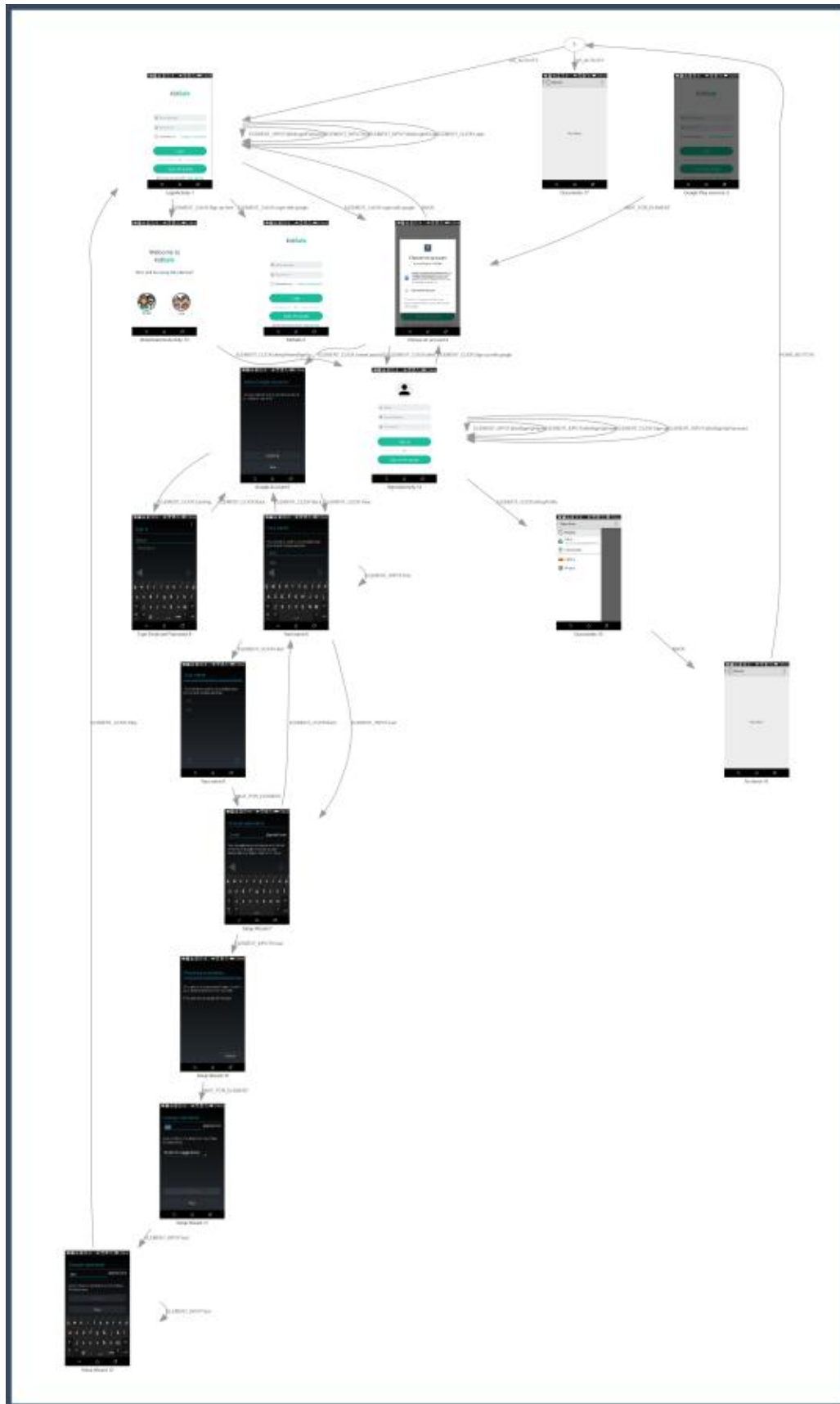


Fig.6.15
Robo test, HTC One (M8), API Level 19

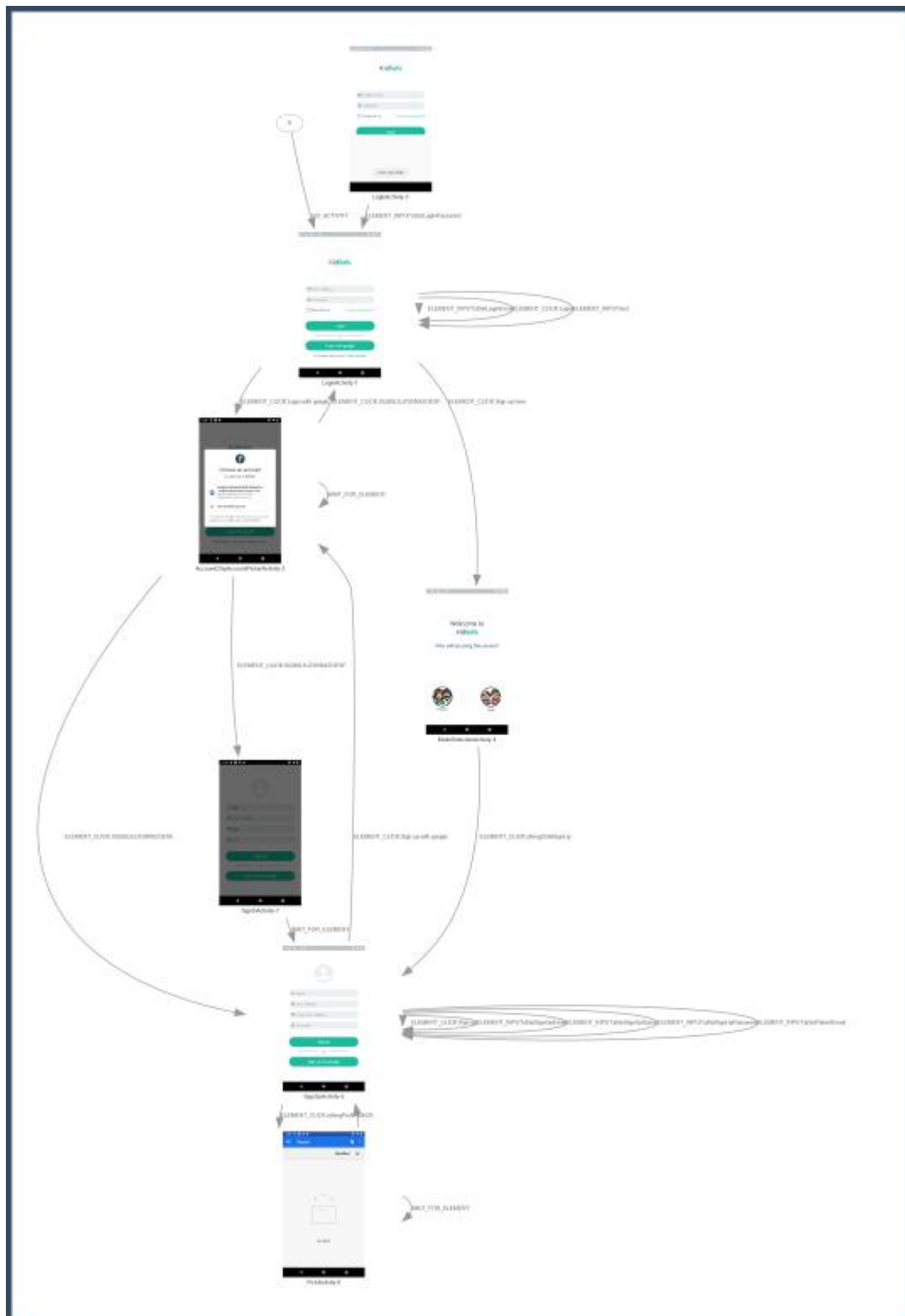


Fig.6.16
Robo test, Pixel 2, API Level 28

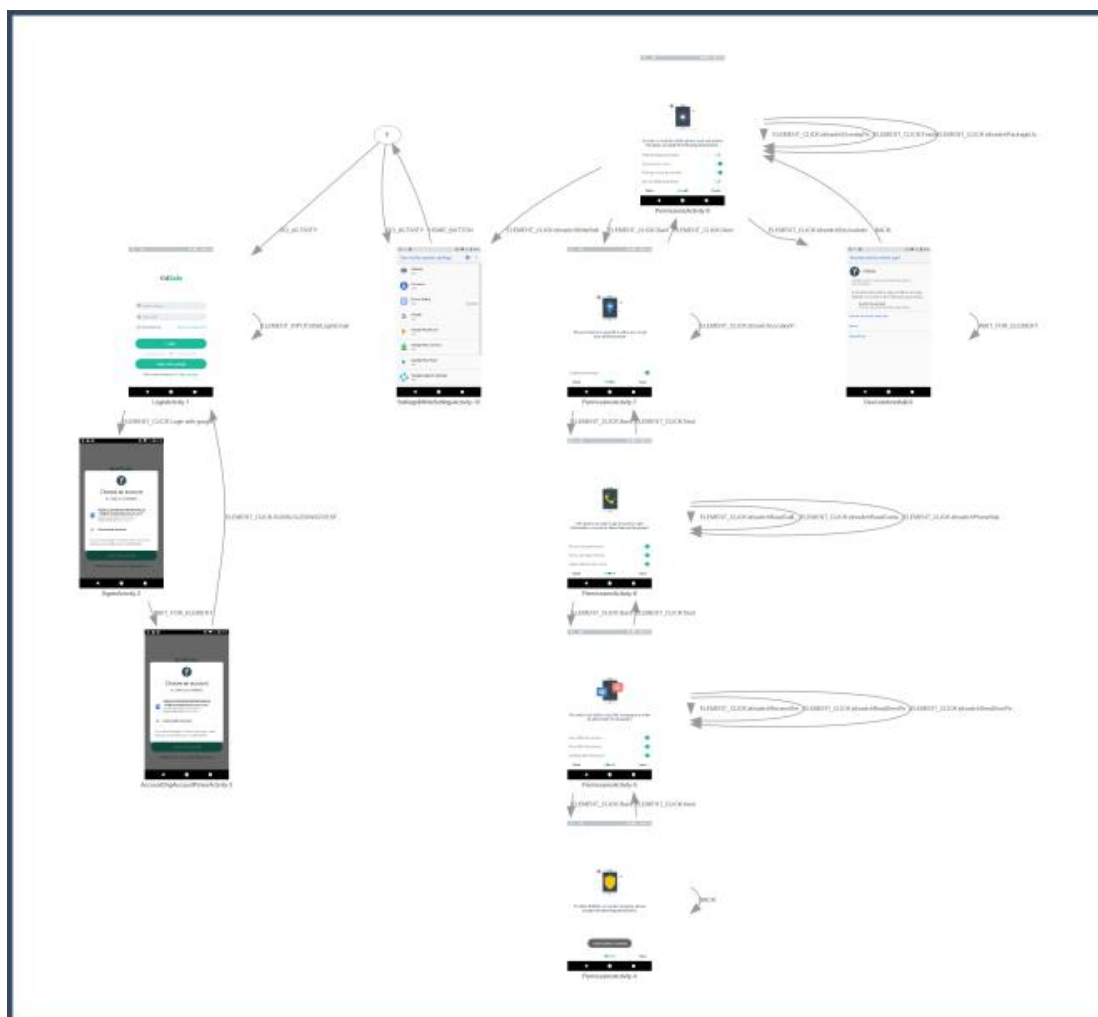


Fig.6.17
Robo test, Pixel, API Level 26

7

USER MANUAL

This chapter will illustrate how to get the most out of kidsafe as a parent. We will be talking about each feature individually and explain how to use them. But the first step of course is Signing up.

7.1. Signing up

To use kidsafe you must sign up and verify your account through the email that will be sent to the same email used in the signing up process. Signing up is essential to use kidsafe on both the parent's and the child's phones. To make it easier for the user, kidsafe offers two methods to sign up.

1. Using email and password
2. Using Google

Both the methods are working flawlessly without any problem and it is up to the user to decide which one to use.

7.2. Signing up as a parent

Lets go ahead and sign up the parent “Mahmoud” first.

Fig.1 shows how to sign up in KidSafe as a parent using an email and a password.

Screen 1 of Fig7.1 shows the first screen of KidSafe which is the login screen. If you already have an account then go ahead and login, but in our case we need to sign up first.

To sign up you should click on the “**Sign up here**”, it will take you to another screen used to determine who should be using this device. A parent or a child.

Screen 2 of Fig7.1 shows the screen in which you will choose what the user of this device will be.

This screen just determines the input fields that are needed in signing up later in the signing up screen.

If you choose a parent you will be asked about your name, email, password and maybe a profile image if you would love to, like in screen 3 of Fig7.1

Now lets register the parent “Mahmoud”

We will enter the name as Mahmoud, the email address as Mahmoud.KidSafe@gmail.com and the password as 123456 as shown in screen 4 of Fig.7.1.

If you didn't already add a profile image and try to click on the sign up button, you will get a confirmation dialog like in Fig.7.1 screen 5 asking you if you would like to add a profile image or not if you clicked Cancel, you will sign up with the default profile image. It is not very pretty to be honest.

That's why we will click OK and add profile image from the gallery as shown in Fig.7.1 screen 6 and we will choose the image in the middle to represent the parent “Mahmoud”. Click on it, and now you are ready to sign up as shown in Fig.7.1 screen 7.

Before this parent “Mahmoud” can start using his account he must verify his email, this is to avoid unverified users or robots that may cause any overload on our system.

Screen 8 of Fig.7.1 will appear when you login as long as you aren't verified. After clicking the Sign up button. It asks you to verify your account through the email used in the signing up process.

The button below in screen 8 of Fig.7.1 with a count down timer shows the time you will have to wait before asking again for another verification email. Again this is to avoid overload on the system.

After the count down timer finishes the button will be set enabled and clickable allowing you to resend verification email as shown in screen 9 of Fig.7.1.

This means you can ask for another verification email in case the first one didn't arrive.

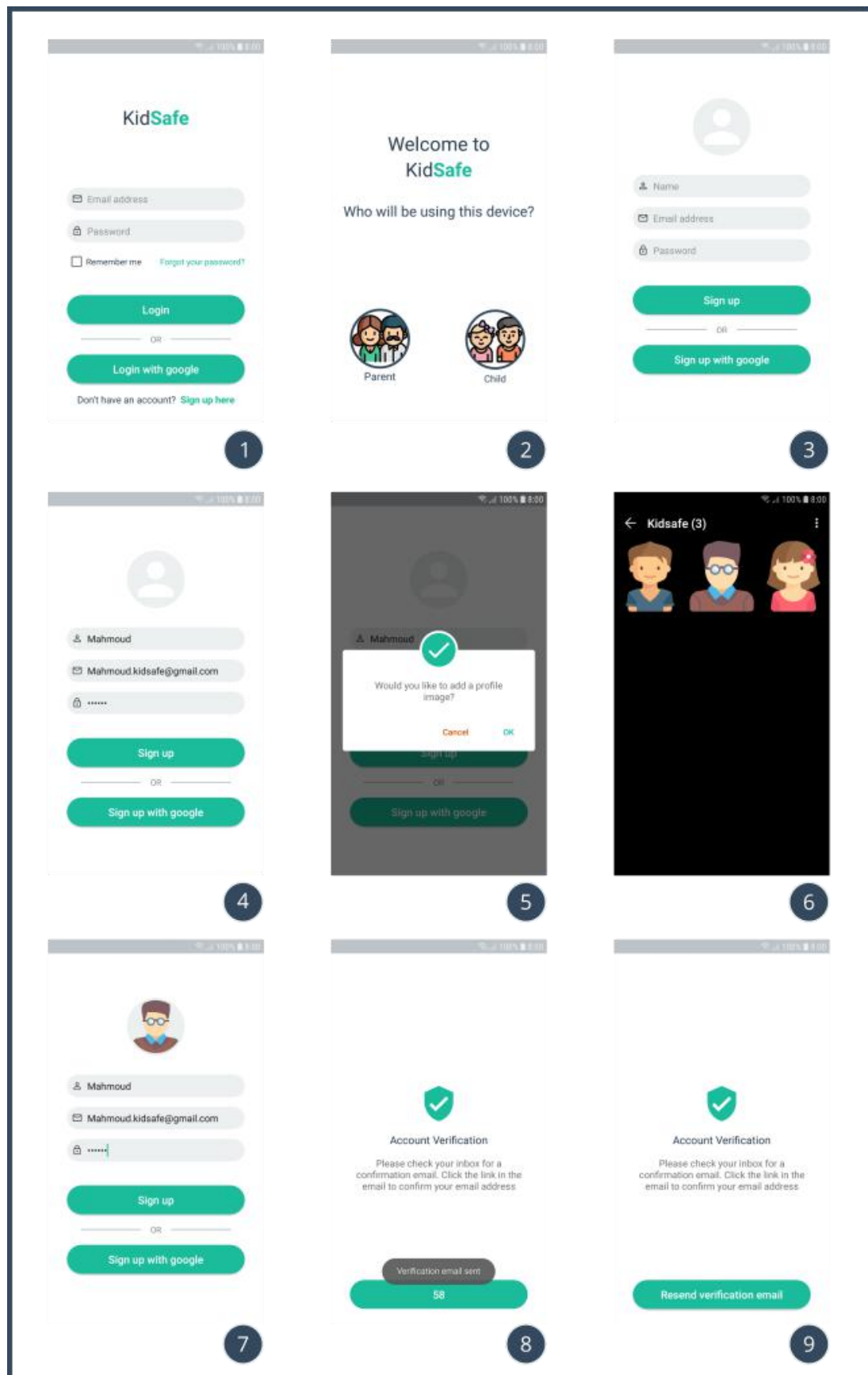


Fig.7.1

Signing up using email and password as a parent

7.3. Verification

To verify this parent account we will sign in to Google using Mahmoud.Kidsafe@gmail.com and take a look at our inbox as shown in screen 1 of Fig.7.2.

You will find the email shown in screen 2 of Fig.7.2 sent by noreply@kidsafe-3a667.firebaseio.com which is KidSafe's verification portal in firebase. It includes KidSafe's firebase id in it (kidsafe-3a667) and asking for no reply.

This email kindly asks you to verify your account by clicking on the attached hyper link.

When you click on the hyper link, you will be directed to a screen telling you that your account has been verified like as shown in screen 3 on Fig.7.2.

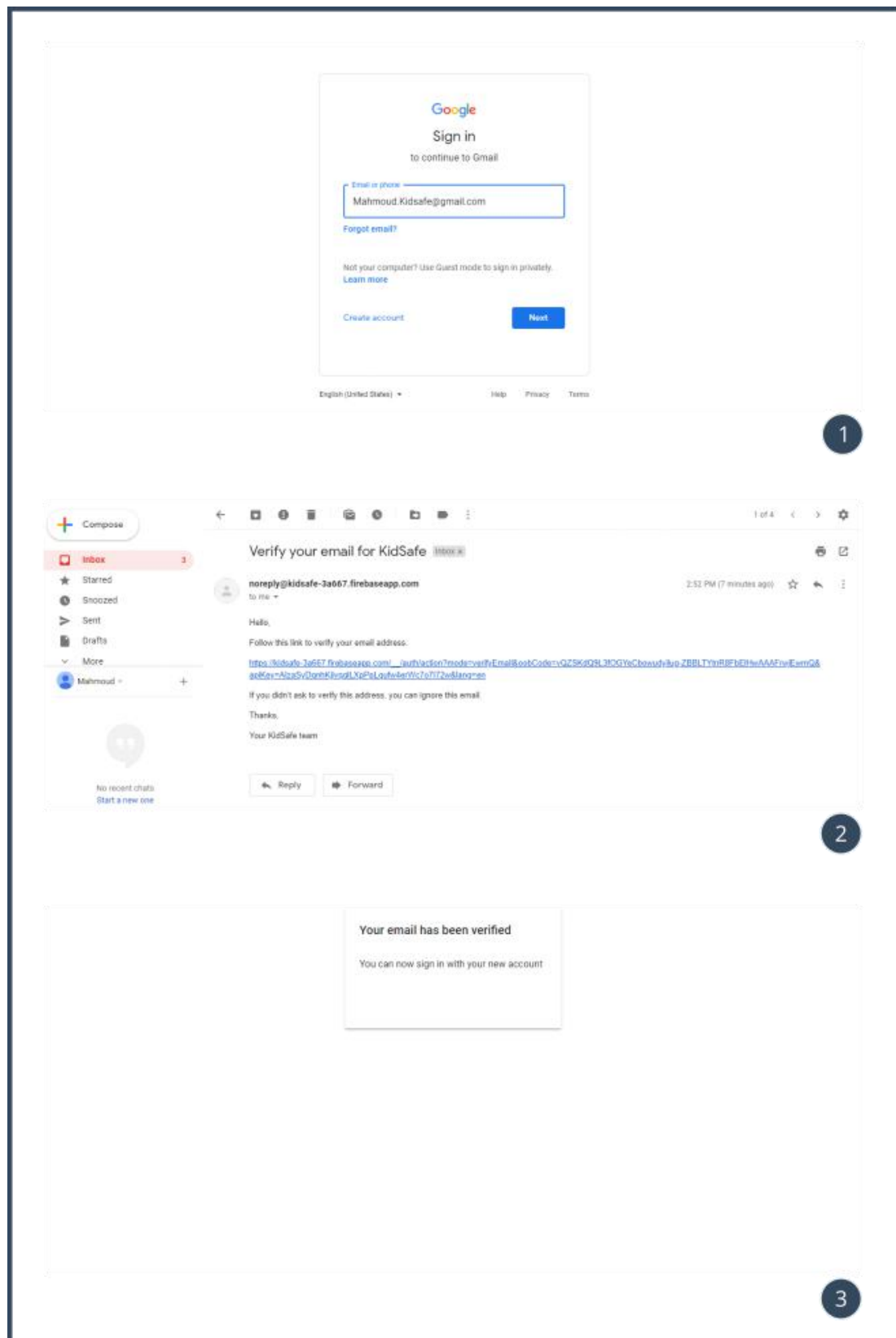


Fig.7.2
Verifying Mahmoud's account

Now this parent “Mahmoud” can use his account without any problems and the next time he tries to log in, he won’t be directed to the verification screen any more, instead he will be directed to another screen showing the kids linked with that parent as shown in Fig.7.3.

But as you can see it shows that there are “No kids added yet”.

This is true since we didn’t create any child account yet and linked it with this parent “Mahmoud”.



Fig.7.3
Parents signed in screen

Note

Make sure you enter a name less than 15 characters and a valid email eg. example@domain.com without spaces and a password of 6 characters at least.

If we try to invalidate these rules, for example lets enter 123 as a password and click the “Sign up” button and see how KidSafe will react to this.

As expected, it shows an error on the password field saying “Enter a valid password” as shown in screen 2 of Fig.7.4.

KidSafe wouldn't let you sign up with invalid data according to the steps used in the validation process inside the code.

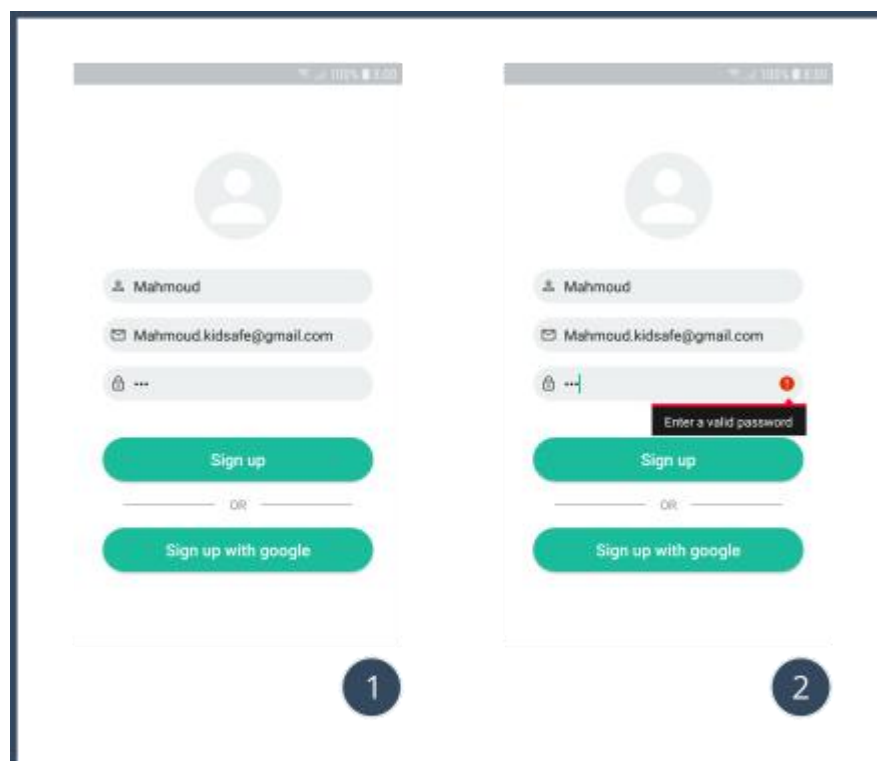


Fig.7.4

Trying to sign up with invalid password

7.4. Signing up as a child

Now, let's go ahead and link two kids to that parent by first creating their accounts and link them to the parent "Mahmoud"

Using the same signing up routine but this time we will sign up this device as a child in by clicking on the child image in screen 2 of Fig.7.5.

When we do so, the signing up screen which is screen 3 of Fig.7.5, will show another required field in the signing up screen asking for parent email to add and link this child to him.

The first child's name is going to be Ibrahim, Ibrahim.KidSafe@gmail.com as his email, Mahmoud.KidSafe@gmail.com as his parent's email and 123456 as his password.

Notice how we wrote this child's parent email in screen 4 of Fig.7.5 as Mahmoud.KidSafe@gmail.com which is already signed up and verified parent email. This way is how we link the two together. HENCE, WRITING A CORRECT AND VALID PARENT EMAIL IS ESSENTIAL.

Again, after writing correct data and clicking the Sign up button you will be directed to the verification screen which is screen 5 of Fig.7.5.

The verification method is the same for both kids and parents, so the same steps of verification which were shown in Fig.7.2 too by signing in to Google with Ibrahim.Kidsafe@gmail.com and checking his inbox for an email with a title of noreply@kidsafe-3a667.firebaseio.com and clicking on the hyper link included in the email.

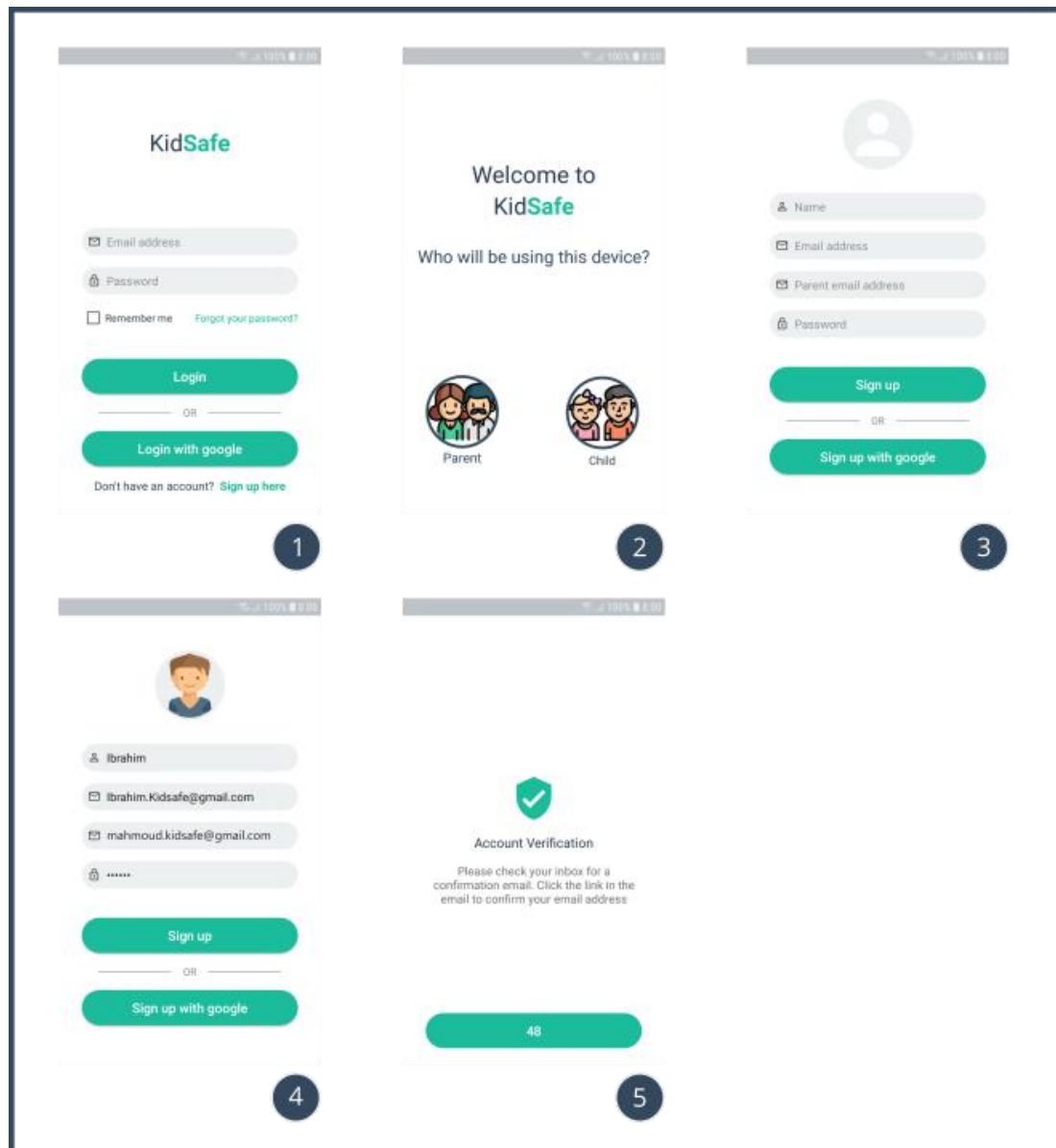


Fig.7.5
Signing up using email and password as a child

7.5. Granting required child permissions

If we try to login with that child “Ibrahim” after verification, you will be represented to the permissions screens as shown in Fig.7.6 asking you for some permissions to be able to gather the data from the child’s phone.

Our app does a lot of things eg. Phone locking, screen time control, app blocking, location tracking, Geo-fencing, sms & call logs and getting the contacts of the kid. It isn’t surprising how many permissions we ask for.

Screen 1 of Fig.7.6 shows the main permissions screen telling the user “To allow kidsafe to function properly, please accept the following permissions” and it is self explanatory.

Screen 2 of Fig.7.6 shows the first required permissions, which are the permissions needed to read the child’s sms messages logs. These permissions must be allowed as shown in screen 3 & screen 4 of Fig.7.6 to be able to continue and use kidsafe.

Screen 5 of Fig.7.6 shows the second required group of permissions, which are the permissions needed to get the child’s phone call logs. Again, all these permissions must be accepted as we have done before.

Screen 6 of Fig.7.6 shows the location permission, which asks if this app can use your location. Enabling location is a must to allow both location tracking and Geo-fencing to function properly at the parent’s side.

Screen 7 of Fig.7.6 shows the settings permissions. The write settings, overlay and package usage permissions are needed to block the child’s apps and they are accepted as in screens 8, 9 & 10 accordingly. Meanwhile device admin permission is needed to allow the parent to lock his child’s phone, and it is activated as shown in screen 11 of Fig.7.6. We also use device admin to identify if any of your kids have uninstalled KidSafe. We will cover this later on.

When all the permissions are granted as shown in screen 12 of Fig.7.6 you can now continue to use KidSafe without any problems as the child “Ibrahim”.

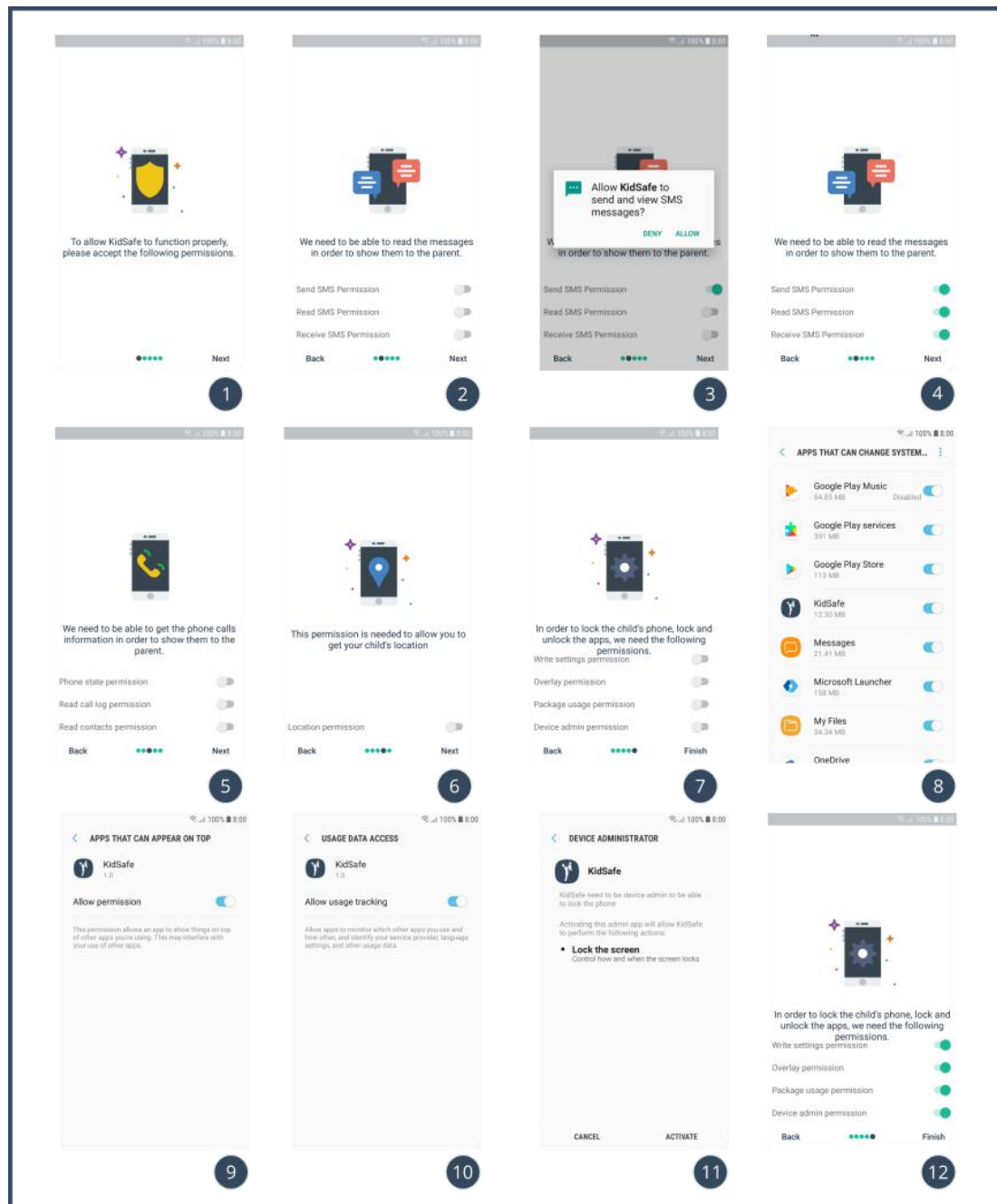


Fig.7.6
Permissions screen shown as we sign in as a child for the first time

The last time you click on the next button in screen 12 of Fig.7.6 you will be represented to the child signed in screen as shown in screen 3 of Fig.7.7. And a notification will appear on top as shown in screen 4 of Fig.7.7, this notification conforms that KidSafe is up and running. As long as this notification exists, your child data will be monitored in real time. And we made it as hard as possible to stop this notification that it is almost impossible to stop even after a restart or a shut down.

If your location isn't enabled, you will see that dialog shown in screen 1 of Fig.7.7 asking you to activate the location. When you click on OK, you will be able to activate the location as shown in screen 2 of Fig.7.7 by switching the ON switch to the right.

Now this child "Ibrahim" data is uploaded and the parent "Mahmoud" can access it.

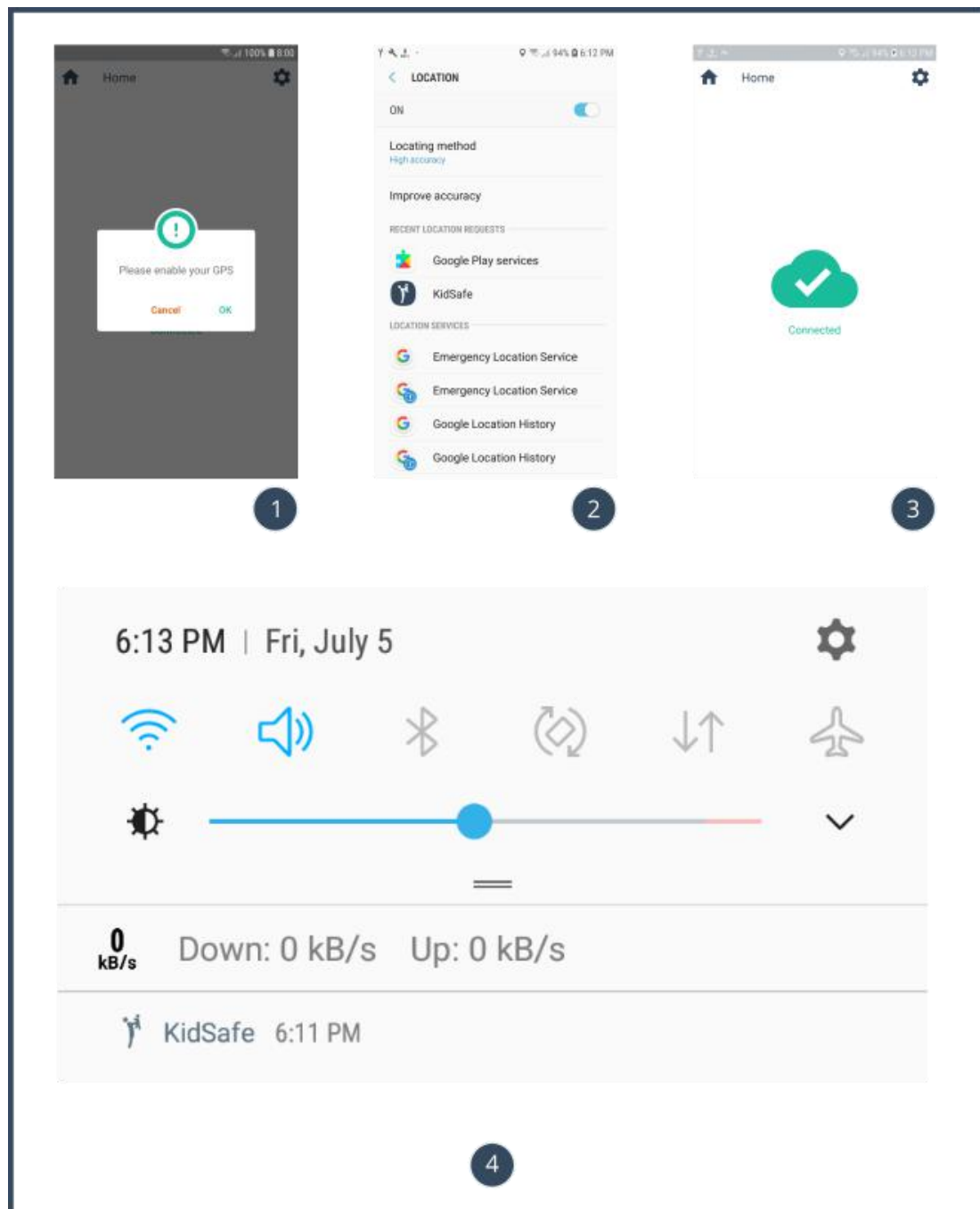


Fig.7.7
Logging in as a child

7.6. Signing up as a child using Google

Now let's try to login again as the parent "Mahmoud" to see what changes have happened.

As shown in screen 2 of Fig.7.8, you will see one child added to your list.

From this screen, you can lock Ibrahim's phone by clicking on the switch next to his name. We will cover this later, but right now we are going to add one more child "Magy" to the parent "Mahmoud".

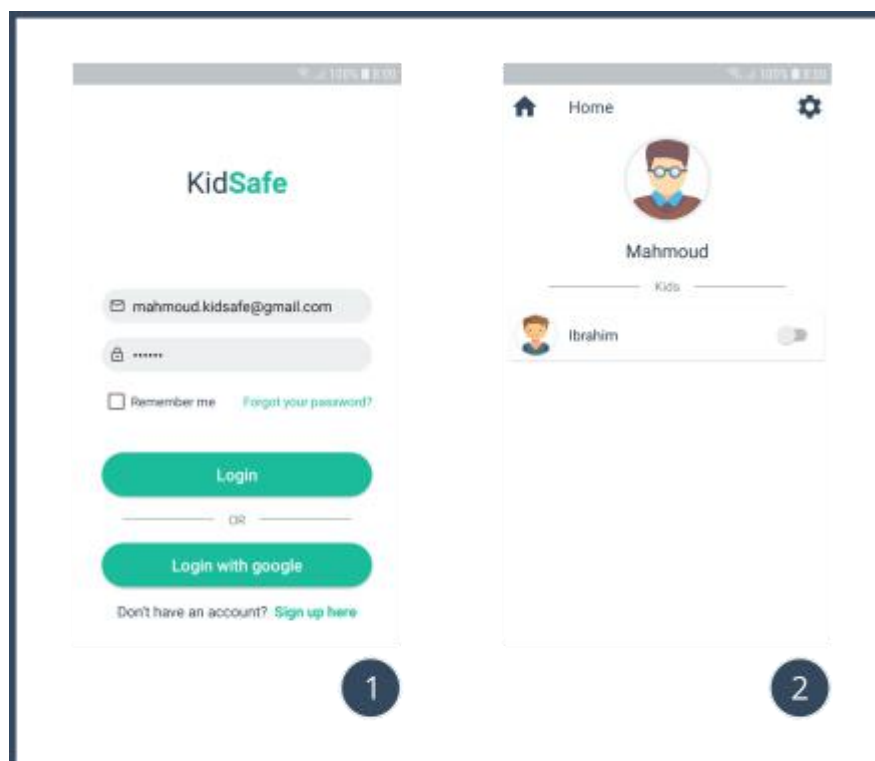


Fig.7.8

Logging in as the parent "Mahmoud"
after adding the child "Ibrahim" to him

To sign up a child “Magy” using Google we will go through the conventional steps of signing up a child as stated before until we reach the signing up screen which is screen 3 of Fig.7.9.

From screen 3 of Fig.7.9 we will click on Sign up with google button, we will get a dialog as shown in screen 4 of Fig.7.9 asking you which to either choose an already signed in Google account or “Use another account”. We have magy.kidsafe@gmail.com signed in already as shown in screen 4 of Fig.7.9, so we will choose it.

When you click on magy.kidsafe@gmail.com you will see another dialog as shown in screen 5 of Fig.7.9 asking you to enter your parent email. In our case we will enter mahmoud.kidsafe@gmail.com to add and connect the child “Magy” to the parent “Mahmoud”.

When you are done with connecting the child “Magy” to the parent “Mahmoud”, you will be represented to the same permissions screens in Fig.7.6 asking you for some permissions to be able to collect magy’s data to shown it to the parent “Mahmoud”.

When all is set you will see the same as screen 6 of Fig.7.9 telling you that you are connected.

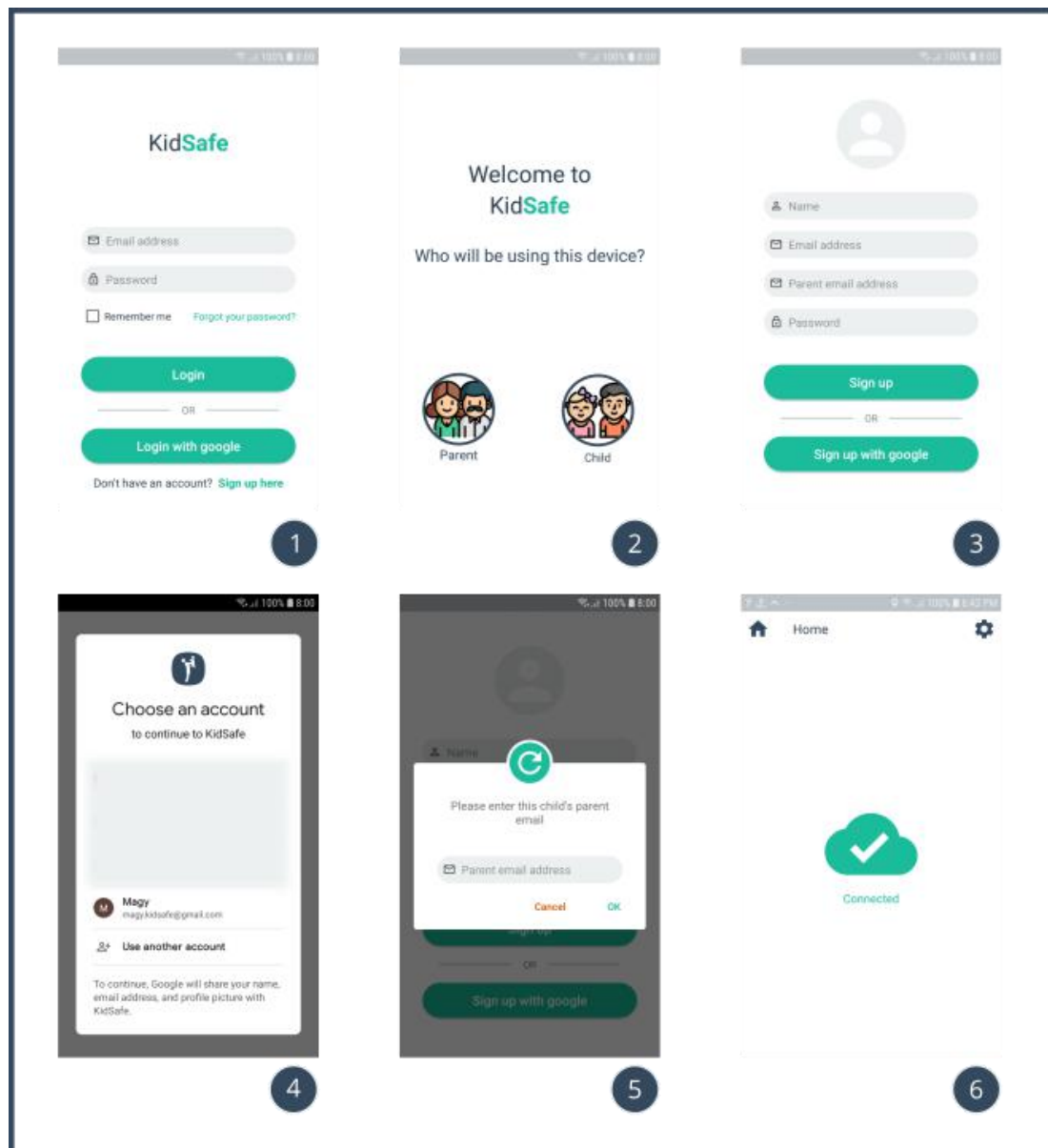


Fig.7.9

Signing up a child "Magy" using Google and connecting it to the parent "Mahmoud"

The next time you login as the parent “Mahmoud” you will see the child “Magy” is added to your list.

Note

Signing up with Google fetches your Google’s account image and sets it as your KidSafe’s account profile image as the case with the child “Magy”.

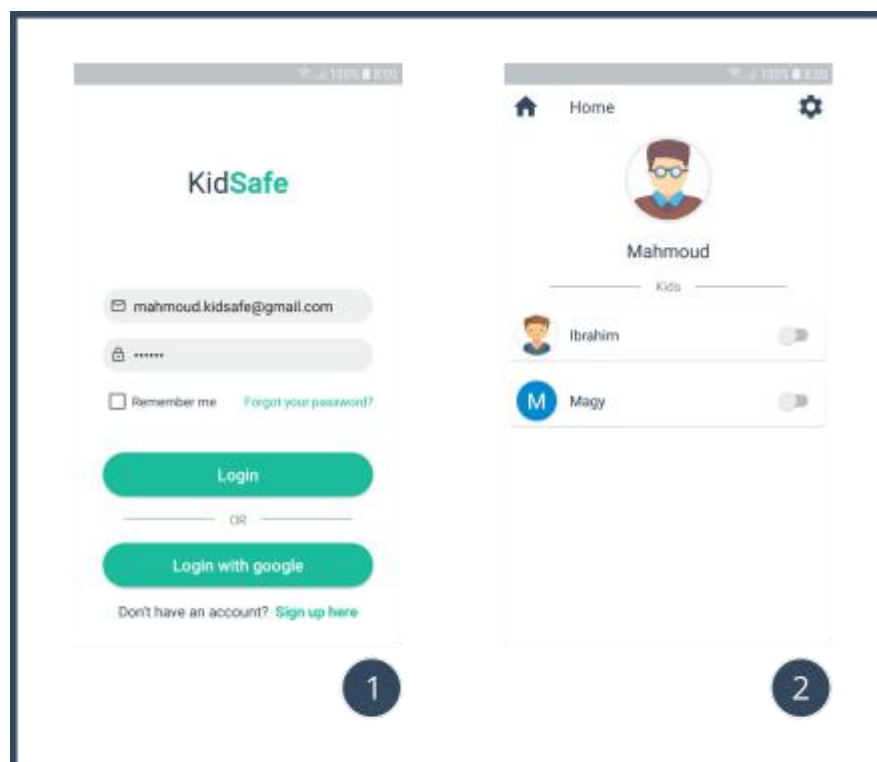


Fig.7.10

Logging in as the parent "Mahmoud"
after adding the child "Magy" to him

7.7. Locking child's device

Now let's talk about our features. The first one is locking the child's phone or setting a daily screen time.

When you press on the switch, you will see the dialog in Fig.7.11 asking you if you want to lock his phone now "Immediately" as shown in screen 1 of Fig.7.11 or set a daily screen time "Specify a period" as shown in screen 2 of Fig.7.11, enter your preferred value in hours and minutes and when the child exceeds this amount, his phone will shut down until the next day as the timer resets daily.

Whatever you choose, when you click OK, the switch next to the child's name will be turned on as shown in screen 3 of Fig.7.11.

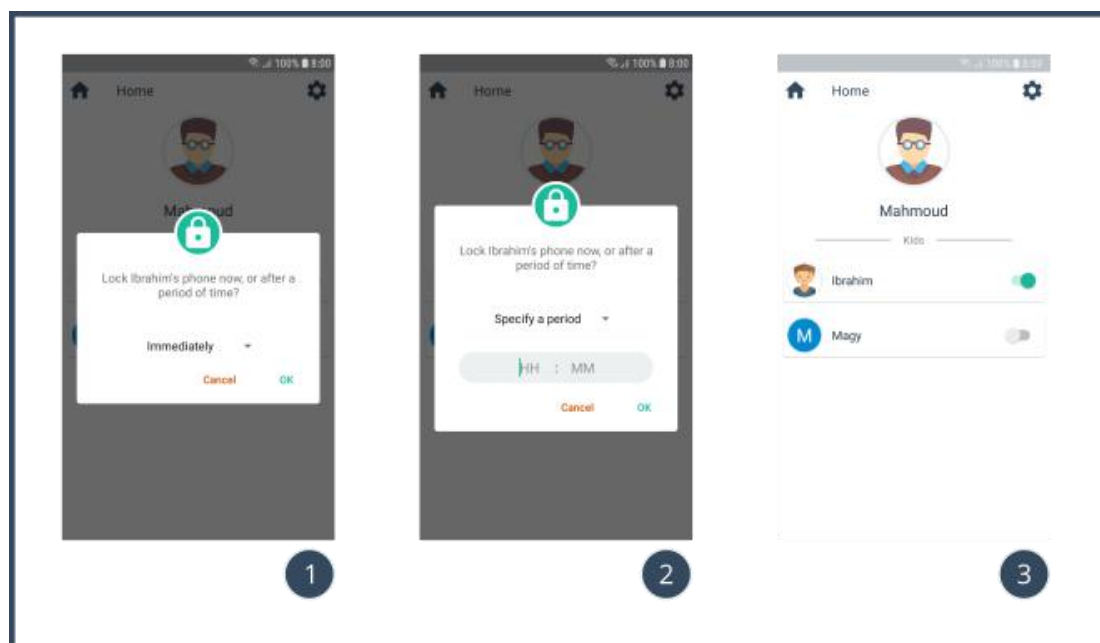


Fig.7.11
Locking Ibrahim's phone

7.8. Blocking apps

If your child uses an app a lot, that he became addictive to it. You can block it using KidSafe with a click of a button.

As shown in Fig.7.12, when clicking on any of the children in the parent “Mahmoud” children list shown in screen 1 of Fig.7.12, you will be represented to the child details screen in both screen 2 & 3 of Fig.7.12.

As you can see in screen 2 & 3 of Fig.7.12, there are three icons with the titles apps, location and activity log respectively. Right now we are in the apps part of the whole screen, which shows all the installed apps on the child’s phone. As shown in screen 2 & 3 of Fig.7.12 Ibrahim has a list of apps, let’s try to block the “Root Browser” app.

Blocking apps in KidSafe is very easy, just turn the switch next to the app name to the on state as shown in screen 3 of Fig.7.12 and that’s it. And if you want to unblock the app, just uncheck the switch back to the off state.

This app “Root Browser” is now blocked.

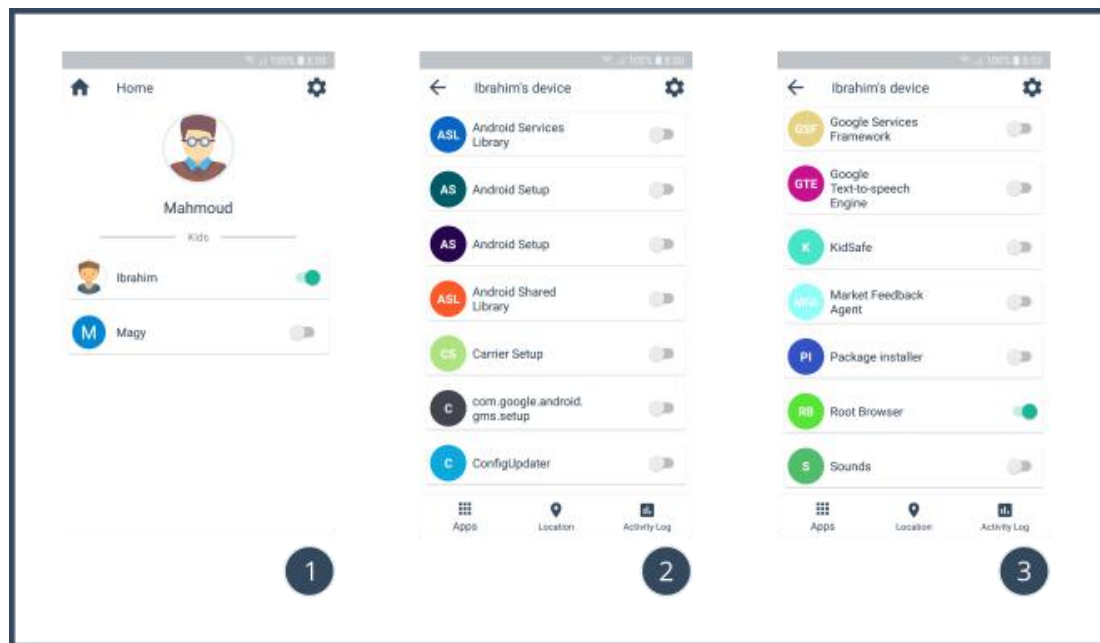


Fig.7.12
Blocking the child's apps.

When Ibrahim tries to open Root Browser in his phone he won't be able to access it like in Fig.7.13, and he will be represented with that screen instead of the app he opened.

This screen tells him that the app “Root Browser” is blocked by your parents.

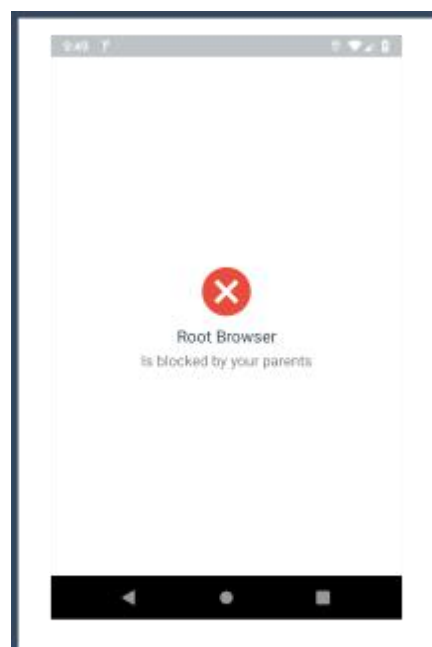


Fig.7.13
Blocked apps screen

7.9. Location tracking

Location tracking allows the parent to view his child's real time location just by clicking on the location icon in the bottom of screen 1 of Fig.7.14. When you click on it you will see a map with a green location pointer shows the child's location. If you click on that icon, you will see the child's name as shown in screen 2 of Fig.7.14.

As Ibrahim moves, this location icon moves with him, showing the parent his child's precise real time location.

If you click on the green round button on the right bottom of both screens 1 & 2 of Fig.7.14, you can set up a Geo-fence as shown in Fig.7.15.

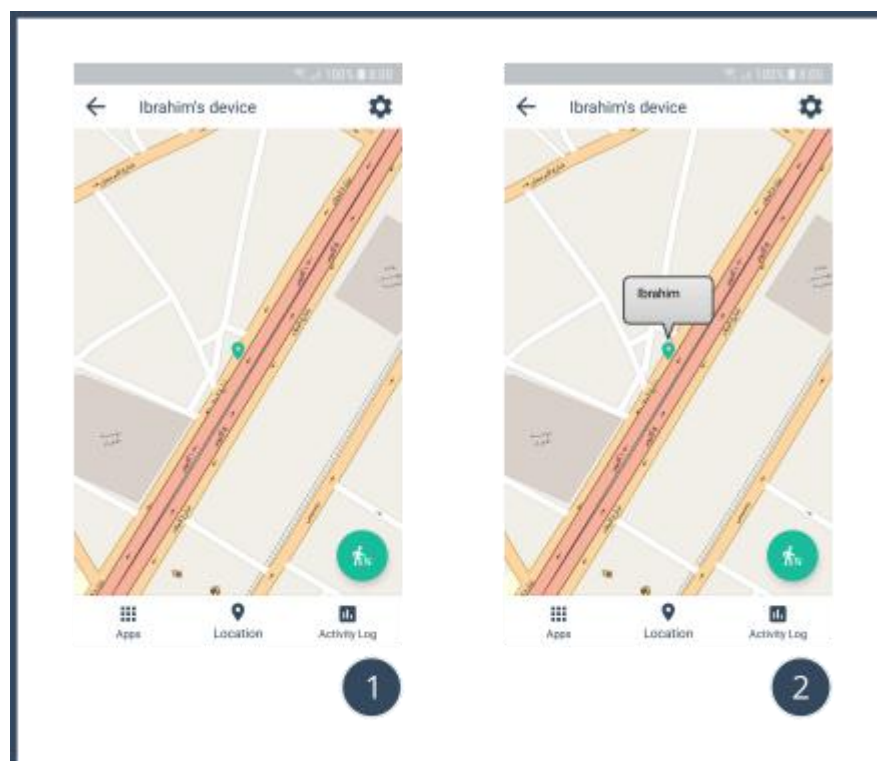


Fig.7.14
Child location tracking

7.10. Geo-fencing

When you click on the green round button on the right bottom of both screens 1 & 2 of Fig.7.14, the dialog in Fig.7.15 will appear asking if you want to setup a Geo-fence on Ibrahim. If so, who should be the center of the fence, you or him? And what is the fence's diameter beyond which you should be alerted?

Geo-fencing is very simple, image if we simply draw a circle with a center of either you or him and a diameter in meters as you wish. When Ibrahim leaves this circle, the parent “Mahmoud” will be notified and alerted.

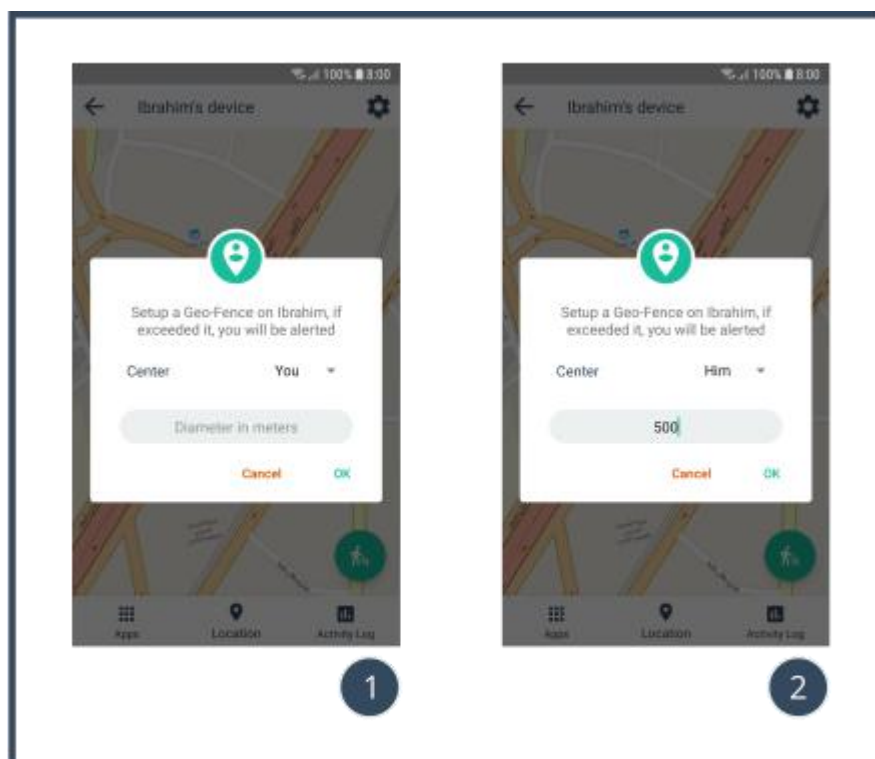


Fig.7.15

Setting up a 500m geo-fence around Ibrahim

7.11. Activity Log

Now let's talk about the activity log part, which can be accessed by clicking on the activity log icon on the bottom of screen 1 of Fig.7.16.

The activity log, contains the child's SMS logs, call logs and contacts.

If you click on the first (default) CALLS tab at the top of the activity log in screen 2 of Fig.7.16. you will see the child's call log, if there are no records already, you will see screen 1 of Fig.7.16 telling you that there are no calls received yet.

However if there are some calls, the parent will be able to see a list containing these calls, in our case in screen 2 of Fig.7.16, the child "Ibrahim" has only 1 call in his log.

A call has 5 details.

1. Contact name which is Unknown Number.
2. When the call was received which is 3 Mins ago.
3. The call type which is represented by an arrow, if the arrow is pointing up then it is an outgoing call, if it is pointing down then it is an incoming call, in our case it is outgoing call.
4. The recipient/caller -if it is incoming call- number which in our case is 01000000000.
5. The duration of the call which in our case is 10.811 seconds.

If you click on the second MESSAGES tab. You will see the child's message log, if there are no records already, you will see a screen like screen 1 of Fig.7.16 telling you that there are no messages received yet.

However if there are some messages, the parent will be able to see a list containing these messages, in our case in screen 3 of Fig.7.16, the child "Ibrahim" has only 1 message in his log.

A message has 5 details.

1. Contact name which is Test Number.
2. When the message was received which is 1 Mins ago.
3. The recipient/sender number which in our case is 01080000000.
4. The message body which in our case is "THIS IS A TEST MESSAGE"

If you click on the third CONTACTS tab. You will see the child's contacts, if there are no records already, you will see a screen like screen 1 of Fig.7.16 telling you that there are no contacts added yet.

However if there are some contacts, the parent will be able to see a list containing these contacts, in our case in screen 4 of Fig.7.16, the child "Ibrahim" has only 1 contact.

A contact has 2 details.

1. Contact name which is Test Number.
2. Contact number which in our case is 01080000000.

There are also two buttons at the right end of each contact as shown in screen 4 of Fig.7.16. The first button - with the phone icon- can let you call this contact and when you press on it you will be directed to the dialer as shown in screen 5 of Fig.7.16. The second button - with the message icon- allows you to send a SMS to that contact as shown in screen 6 of Fig.7.16.

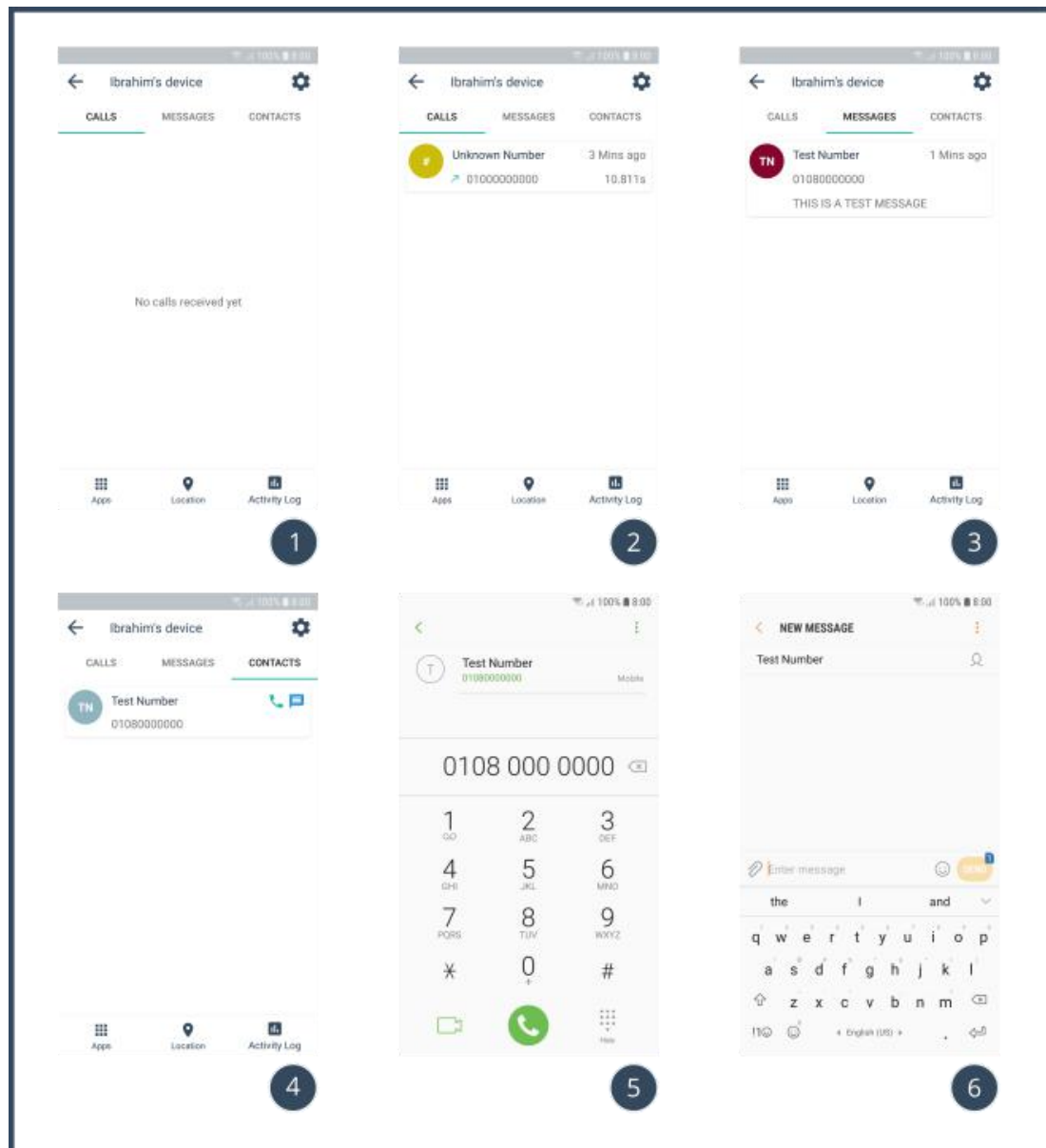


Fig.7.16
Viewing Ibrahim's sms logs, call logs & contacts

7.12. Settings

With the features covered, lets take a look at our settings screen.

You can access the settings screen from any other screen by clicking on the gear button on the right top of the screen as shown in screen 1 of Fig.7.17.

You have 3 main sections.

- General
- Account
- Info.

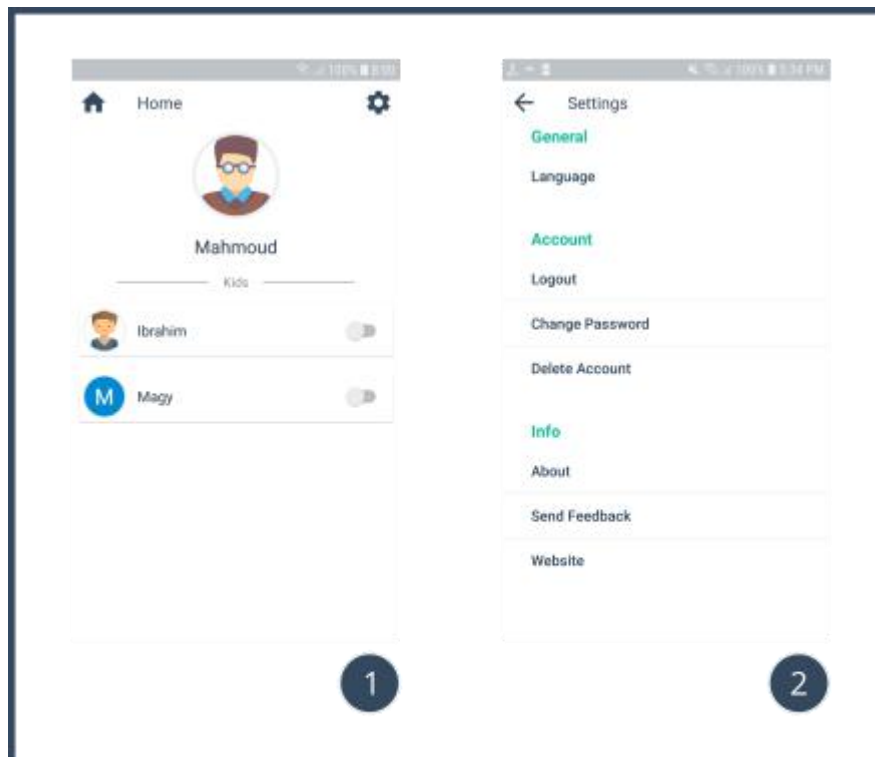


Fig.7.17
KidSafe's Settings

General section of the settings screen shown in screen 1 of Fig.7.18 contains the Language where you can select between two languages, English and Arabic as shown in screen 3 of Fig.7.18.

When you select a different language, KidSafe will restart and go back to the login screen as shown in screen 4 of Fig.7.18 but it is now in Arabic.

Note

The parent and child names can't be translated as they are signed up in English.

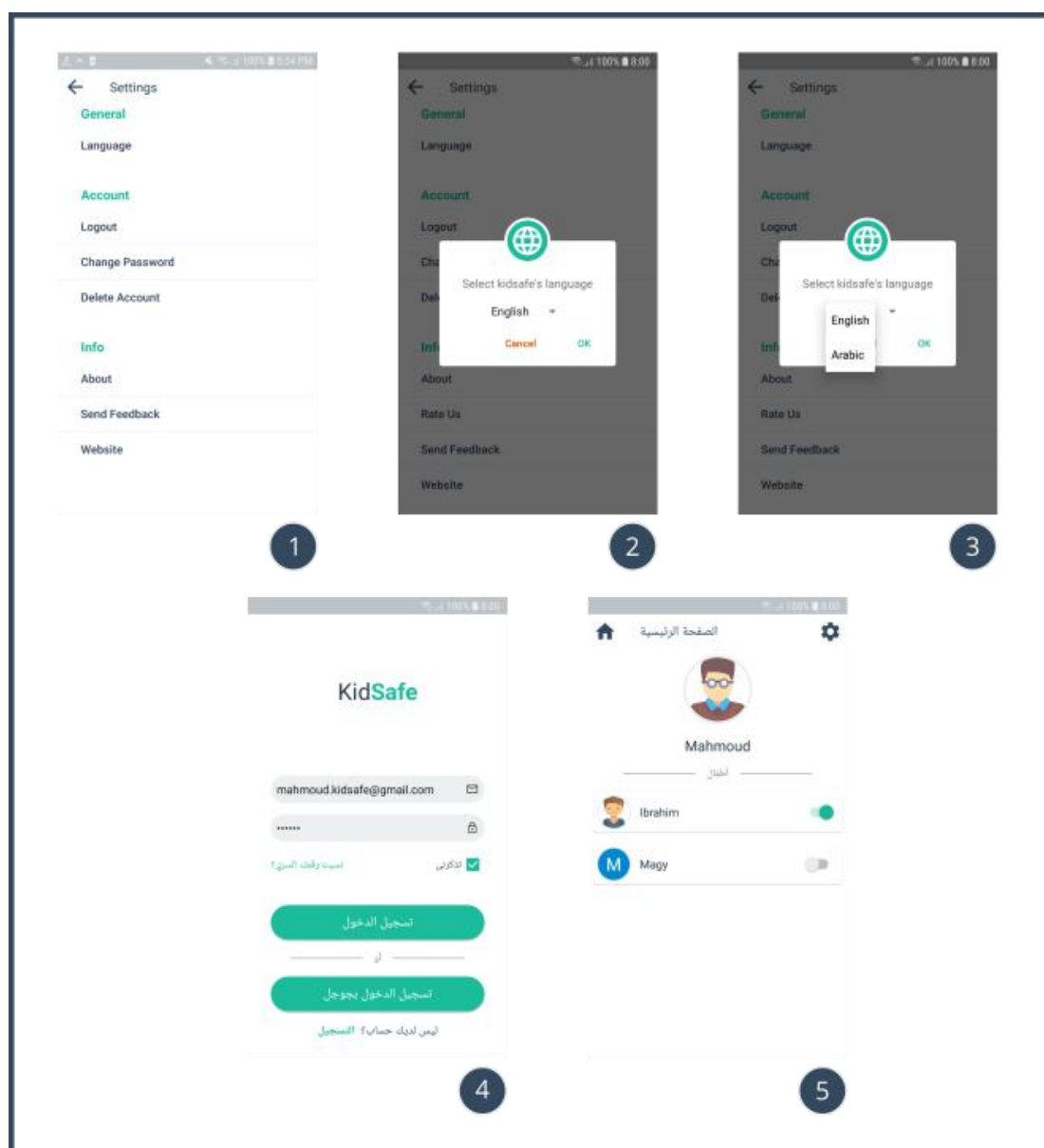


Fig.7.18
Changing KidSafe's language

Account section of the settings screen shown in screen 1 of Fig.7.19 contains account specific details like, logout, change password and delete account.

When you click on logout in screen 1 of Fig.7.19 you will see the dialog in screen 2 of Fig.7.19 asking you, if you want to logout or not. If you click on OK, it will take you back to the login screen even if the “Remember me” check box was checked.

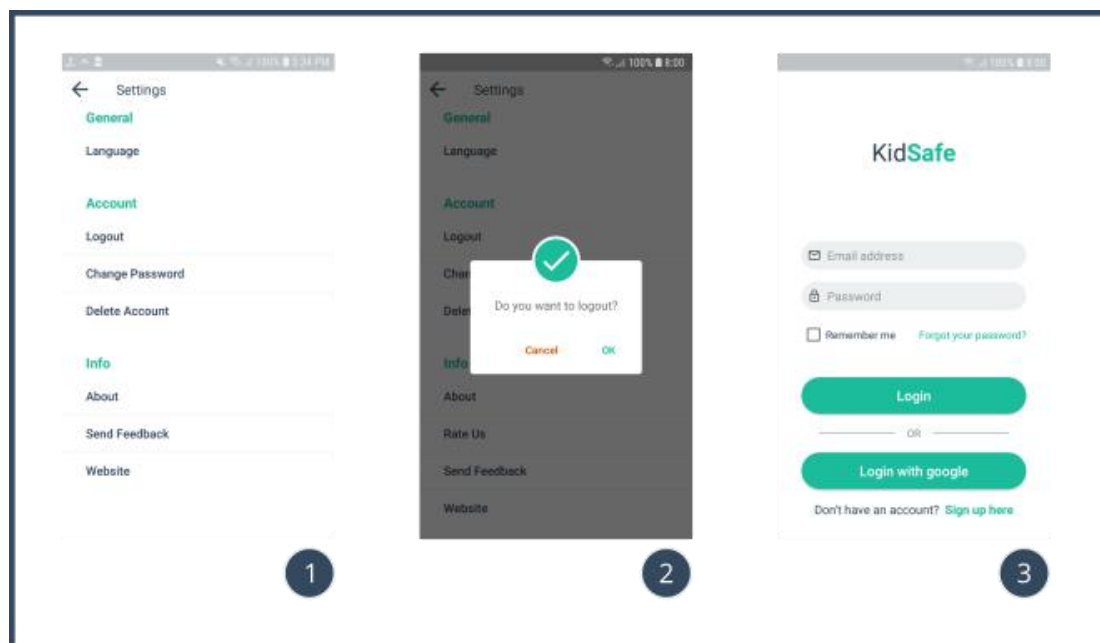


Fig.7.19
Logging out of KidSafe

If you want to change your password just click on the change password in the settings screen shown in screen 1 of Fig.7.20.

When you click on change password in screen 1 of Fig.7.20 you will see the dialog in screen 2 of Fig.7.20 asking you, about your old password, the new password and a confirmation to the new password.

After you click OK, KidSafe will restart and go back to the login screen as shown in screen 3 of Fig.7.20.

Note

The next time you login, make sure you login with the new password to avoid “Wrong password” error as shown in screen 3 of Fig.7.20.

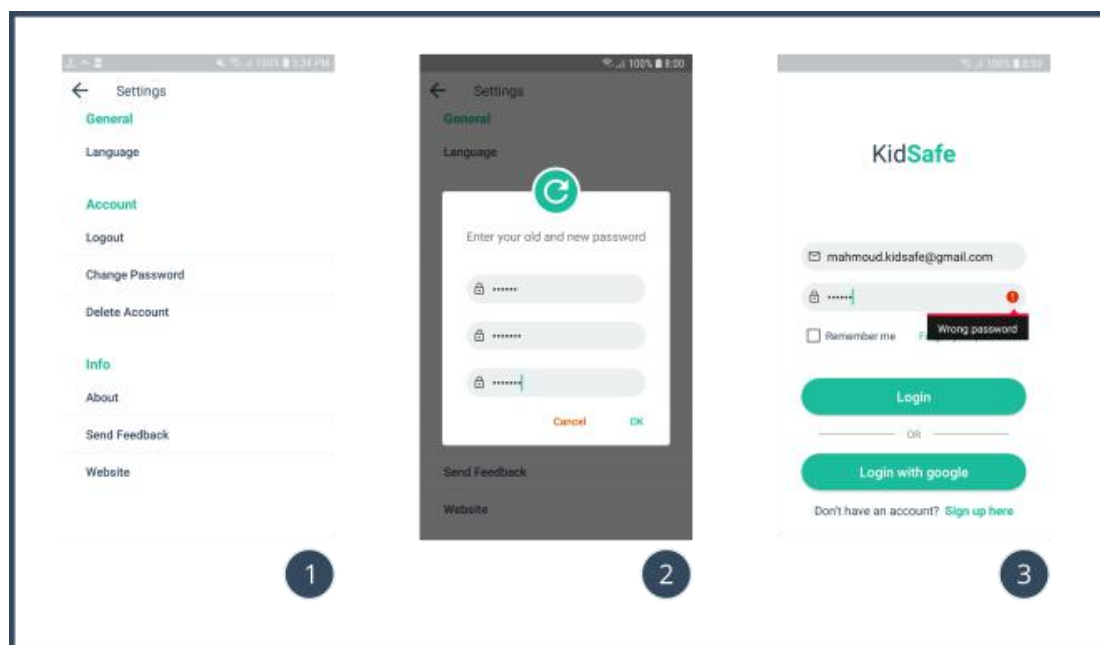


Fig.7.20
Changing your password

If you want delete your account just click on the delete account in the settings screen shown in screen 1 of Fig.7.21.

When you click on delete account in screen 1 of Fig.7.21 you will see the dialog in screen 2 of Fig.7.21 asking you to enter your password. If you entered it correctly. You will be signed out and you will never be able to login again and all your data is gone.

Note

Once you delete your account, there is no going back.

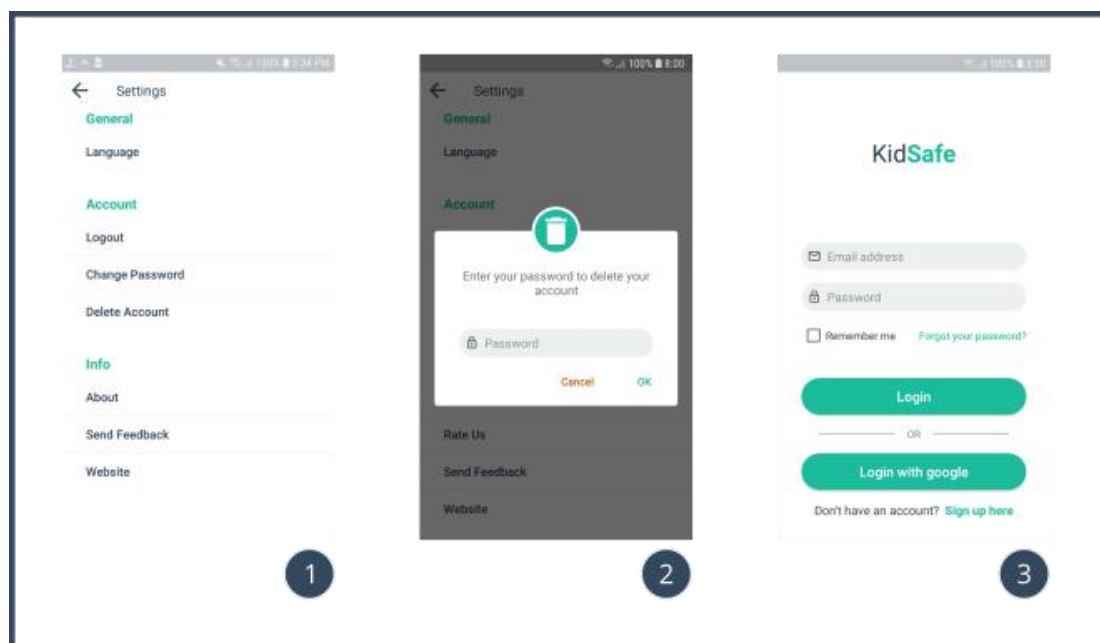


Fig.7.21
Deleting your account

Info section of the settings screen shown in screen 1 of Fig.7.22 contains about, send feedback and website.

About takes you to the about screen shown in screen 2 of Fig.7.22 which tells you some information about the app, the developers, the supervisor and some honorable mentions.

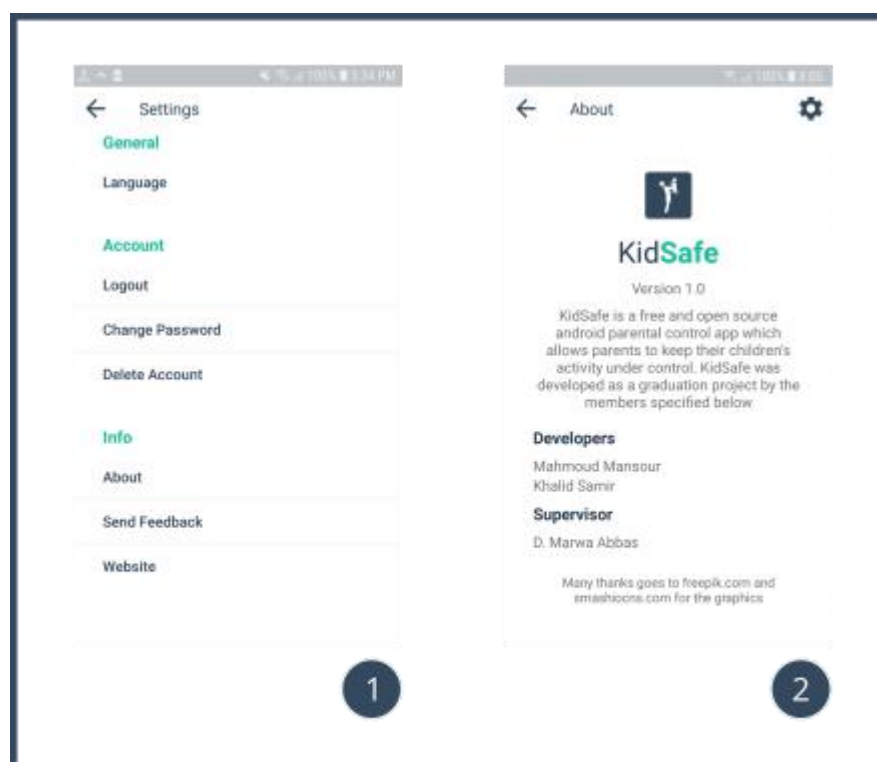


Fig.7.22
About screen

Send feedback in screen 1 of Fig.7.23 will open a dialog as shown in screen 2 of Fig.7.23 asking you for an email client, from which you can send an email as a feedback as shown in screen 3 of Fig.7.23.

The attached data with the headline “Please don’t remove this information” contains app and phone specific information that is important to the developer, to help him fix the issue.

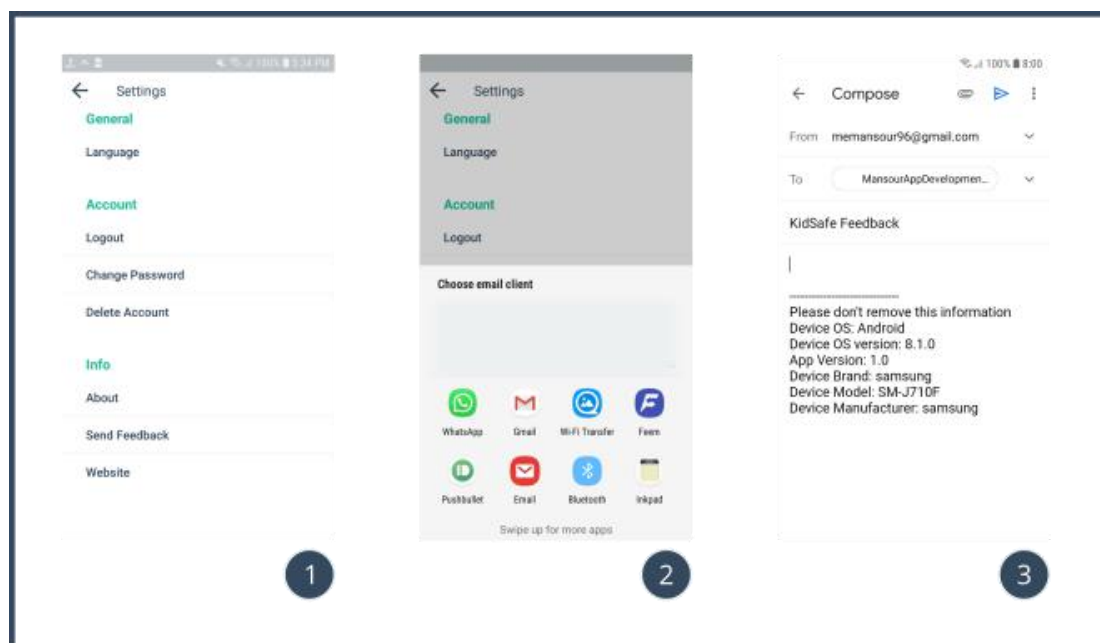


Fig.7.23
Sending feedback

Website in screen 1 of Fig.7.24 will take you to our website[6] as shown in screen 2 & 3 of Fig.7.24.

You can download the latest version of KidSafe through this website by clicking on the “DOWNLOAD APP” button as shown in screen 3 of Fig.7.24.

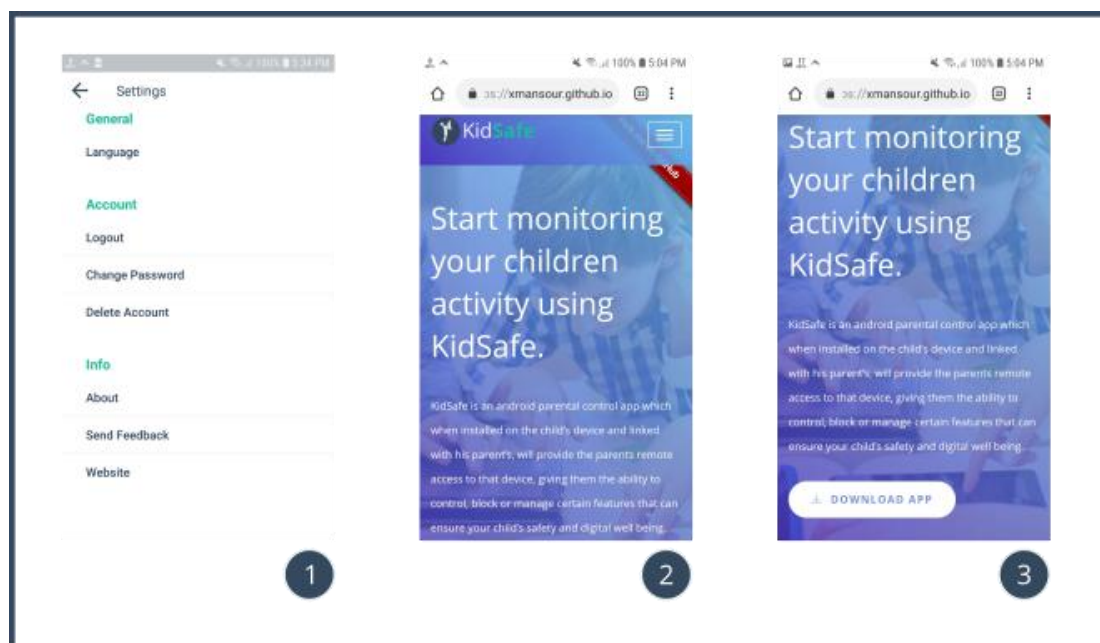


Fig.7.24
Visiting the website

7.13. Protecting KidSafe against uninstalls

What if the child uninstalled KidSafe? Well, it isn't impossible to uninstall KidSafe. But we have made sure you this will be as hard as possible.

We ensured that a child can't enter his settings screen without entering his password as shown in screen 2 and 3 of Fig.7.25. Since the parent is the one who creates the child's account and he is the only one knowing the password of his child's account. The child can never access the settings screen.

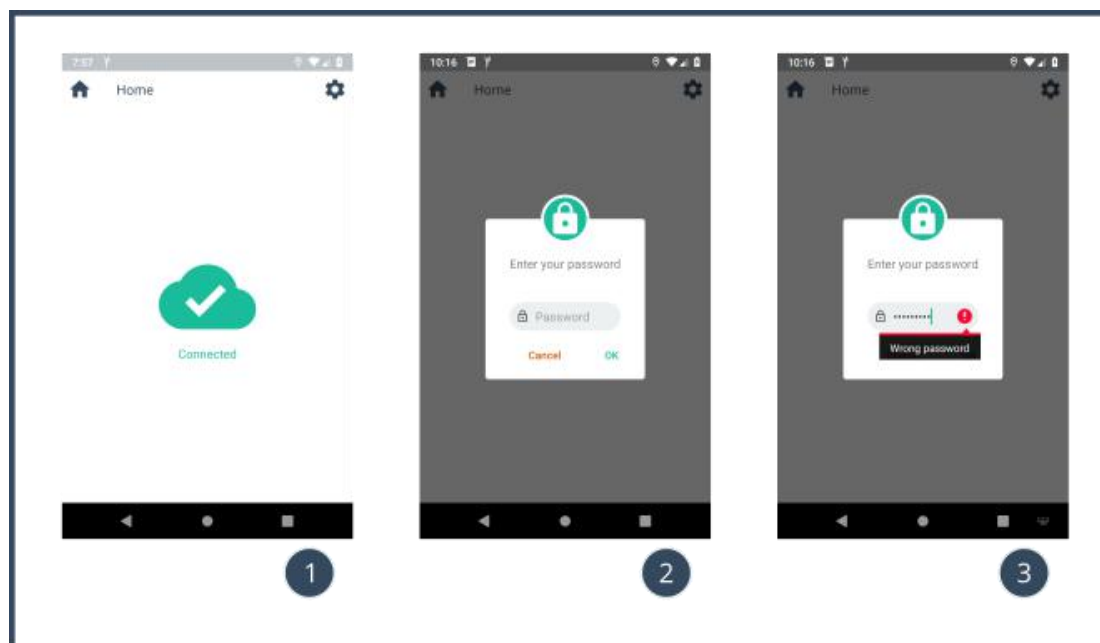


Fig.7.25

Accessing the settings screen as a child requires password

If he is smart enough and he knew that KidSafe is the app causing him all the trouble he is facing and decided to uninstall it. This will only be possible by deactivating the device admin permission. And if he deactivated it as shown in screen 1 of Fig.7.26, the parent will see “child’s name has uninstalled the app.” under the child’s image and name, in our cause child’s name is Ibrahim so it displays “Ibrahim has uninstalled the app.”

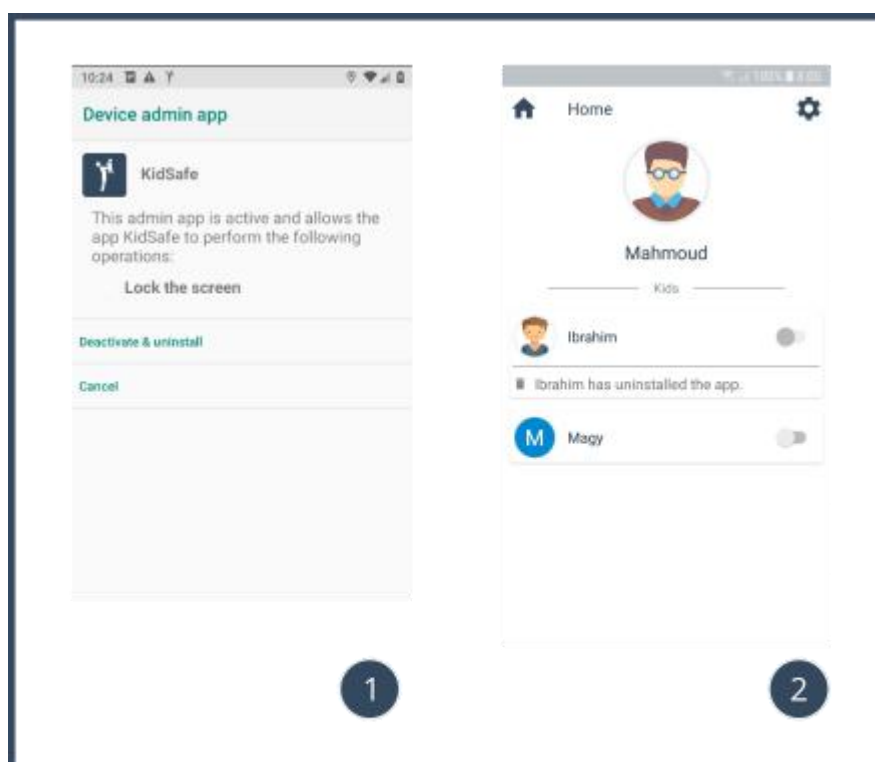


Fig.7.26

When a child uninstalls KidSafe, the parent is notified

7.14. Password recovery

Forgot your password? No problem. From the login screen click on the “Forgot your password?” as shown in screen 1 of Fig.7.27, this will show a dialog requesting your email to send a password recovery to that email from which you can reset your password to a new one.

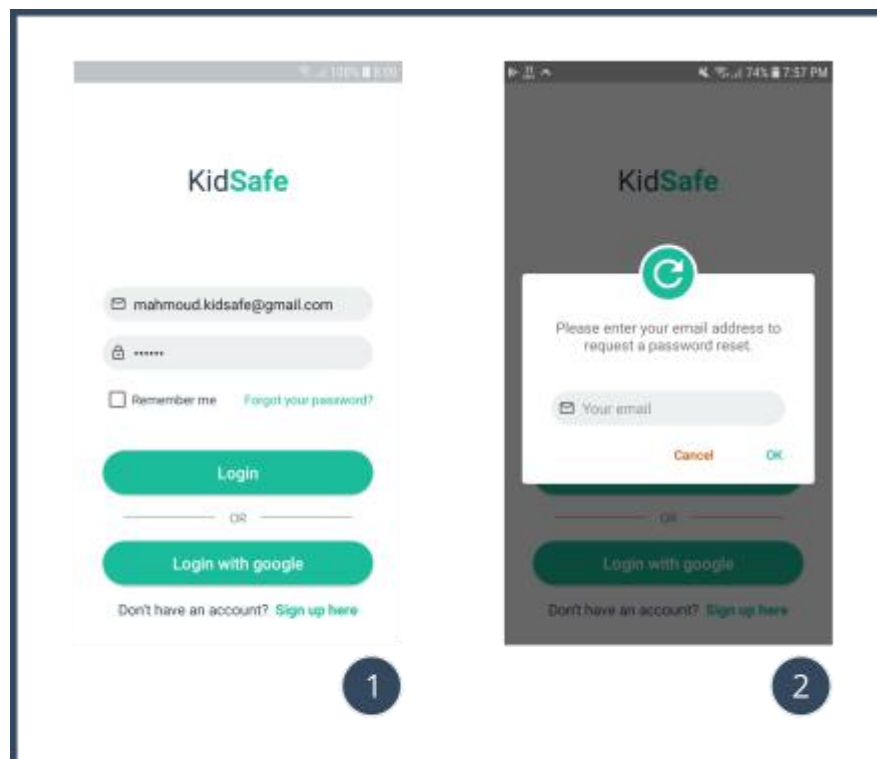


Fig.7.27
Password recovery

References

1. <https://www.independent.co.uk/life-style/gadgets-and-tech/children-smartphone-ideal-age-social-media-snapchat-youtube-a8294701.html>
2. <https://www.telegraph.co.uk/news/2019/02/07/children-spend-twice-long-smartphones-talking-parents/>
3. <https://trello.com/b/23AGFhoj/kidsafe-android-parental-control>
4. <https://github.com/xMansour/KidSafe>
5. <https://github.com/xMansour/KidSafe/tree/master/graphics/test>
6. <https://xmansour.github.io/KidSafe/>