# YILDIZ TECHNICAL UNIVERTSITY DEPARTMENT OF COMPUTER ENGINEERING

# SEMESTER PROJECT

| | |
|---|---|
| **Course Name:** | Introduction to Mobile Programming |
| **Course Group:** | Group 1 |
| **Instructor Name:** | Ass. Prof. Dr. M. Amaç Güvensan |
| **Student ID:** | 11011027 |
| **Student Name and Surname:** | Mustafa Berkay Mutlu |
| **Delivery Date of the Assignment:** | 07.01.2016 |

# General Overview of the Application

Smart Phonebook application is an Android application that keeps user’s contacts, messages, and call logs in its separate database. It listens the incoming and outgoing calls and SMS messages and logs them to its own database. It also has Google Maps support, user can define a home location to each of his/her contacts.

# Activities

### MainActivity

MainActivity has a toolbar and some Menu operations for example “Add New Contact”, “Import from Default Phonebook” and “Search”. Search is done by SQLite search operations and the results are shown in a separate Activity (SearchResultsActivity) using SearchManager and SearchView.

MainActivity has tabbed layout, those tabs are Contacts, Speed Dial, Calls and SMS tabs. Tabs are created as Fragments and tab layout is created using TabLayout and ViewPager.

MainActivity also has a Floating Button, this button’s responsibility is to dial a new number or send a new SMS message. If the user is in one of the first three tabs (Contacts, Speed Dial or Calls) then Floating Button’s onClick() method will send the user to DialerActivity. If the user is in the last tab (SMS) then Floating Button will send the user to MessageActivity. You can find more details about DialerActivity and MessageActivity in the following subsections.

MainActivity is also responsible for keeping the UI update. It has to update the UI whenever a SMS or a call event is occurred and whenever a contact is updated. It does that via receiving custom (app specific) broadcasts. Those custom broadcasts are:

- SmsEventOccurred
- CallEventOccurred
- ContactUpdated
- ContactsListChanged
- SpeedDialUpdated

All custom broadcasts are defined in Config class. Receivers for those broadcasts are registered in the onCreate() method and unregistered in the onDestroy() method of the MainActivity.

### Contacts Fragment

This Fragment has only one ListView inside. It shows the contacts that are stored in Smart Phonebook’s database. MainActivity updates this ListView whenever it’s necessary.

Whenever user clicks to a contact, SingleContactActivity opens.

### SpeedDial Fragment

This Fragment has only one ListView inside. It shows the contacts that are added to the Speed Dial. MainActivity updates this ListView whenever it’s necessary.

Whenever user clicks to a contact, SingleContactActivity opens.

### Calls Fragment

This Fragment has only one ListView inside. It shows the call events (incoming, outgoing and missed). MainActivity updates this ListView whenever it’s necessary.

Whenever user clicks to a contact, a new call to that contact will be made.

### SMS Fragment

This Fragment has only one ListView inside. It shows the SMS message series (or message overviews) that are stored in Smart Phonebook’s database. Note that only the last message from/to a person is displayed in this ListView. If the phone number does not exists in the Smart Phonebook’s database then phone number itself (in formatted) displayed as remote phone number. If remote phone number exists in the database (as a contact) then contact’s name is displayed rather than his/her phone number. Whenever user clicks to an overview message, MessageActivity opens. MainActivity updates this ListView whenever it’s necessary.

## CreateOrEditContactActivity

This is the Activity where user can add a new contact or edit an existing contact. It has a Google Maps Fragment inside and user has to fill all the fields and select a location on the map in order to save the contact.

When a new contact is added or an existing contact edited, ContactUpdated and ContactsListChanged Intent’s are broadcasted. So that related Activity’s can update their UI.

### SingleContactActivity

Whenever user clicks a contact in the Contacts Fragment, SpeedDial Fragment or SearchResultsActivity, user will send to the SingleContactActivity.

User can display stored information about a contact in this Activity. Stored information are:

- Name and Surname
- Phone number
- E-mail address
- Latitude and Longitude of contact’s home location
- Total incoming and outgoing call duration with this contact
- Total missing calls from this contact
- Sent and received SMS message count with this contact
- The contact is in Speed Dial or not

Latitude and Longitude is of course displayed in a Google Map. Whenever user clicks on the Google Map, an Intent with action “Intent.ACTION_VIEW” is started. So that an application can pick up that Intent and show the route to contact’s home location from user’s current location. Note that in order to start that Intent, Smart Phonebook tries to get user’s current location. If it could not get the current location than Intent can’t started.

### MessageActivity

MessageActivity is the Activity used for messaging with a contact. It has a ListView that shows the messages that are sent or received to a specific contact. An EditText for SMS input, and a Button for sending.

When MessageActivity is created, it looks for a remote phone number that may or may not put in the Intent.  If the remote phone number exists in the Intent then MessageActivity loads the past messages from database. If a remote phone number does not exists in the Intent then an EditText becomes visible on top of the Activity layout that is used to get phone number from user.

ListView that shows the messages is scrolled to the end whenever a message is sent or received. Also this ListView’s items are aligned to the left side or right side depending on the sender (if from remote then left, if it’s from user itself then to the right), just like in the modern messaging applications.

There is also a menu Button exists in MessageActivity that is used to delete all messages with current phone number.

### DialerActivity

When user wants to dial a number and call instead of pick one from contacts, then this Activity opens. There is an EditText on top of the layout and dialer buttons under it. Call button is at the end the layout.

The EditText that shows the phone number has a Button for deleting the numbers. If user clicks that Button only the last character is deleted. If user long clicks that Button than all characters are deleted.

Note that I would like make this EditText not focusable. So that software keyboard doesn't show up when user clicks to EditText. I tried few methods to accomplish this but unfortunately none of them gave any significant results.

### SearchResultsActivity

This Activity is used for showing the results of search. It displays the Contact results by setting a ContactsAdapter to the ListView.

## Adapters

There are 4 Adapters used for displayed information on ListViews. Those are:

1. CallsAdapter

Used in Calls Fragment.

2. ContactsAdapter

Used in Contacts and Speed Dial Fragments as well as SearchResultsActivity.

3. MessagesAdapter

Used in MessageActivity.

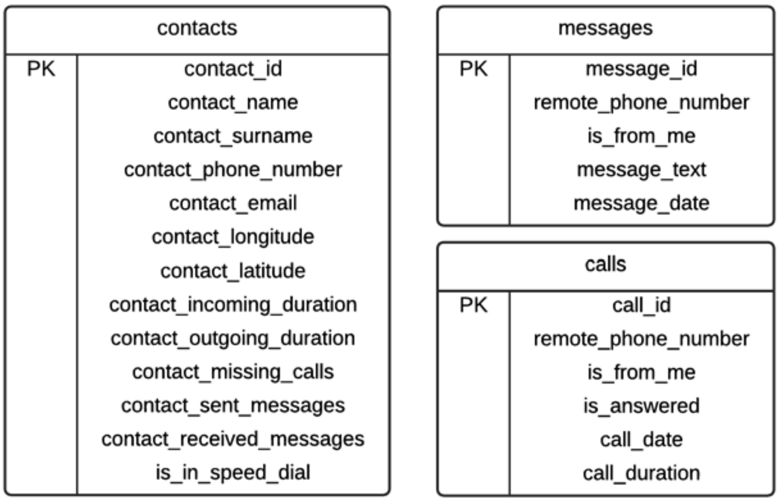
4. MessagesOverviewAdapter

Used in SMS Fragment.

All Adapters are used in regular ListViews (not in RecyclerViews). But in all of them, ViewHolder pattern is implemented for better performance in ListView.

## Database

SQLite database is used for storing contacts, call logs and messages.  Only 3 tables are used in database. You can see their contents in the ER diagram below.



| | contacts |
|---|---|
| PK | contact_id |
| | contact_name |
| | contact_surname |
| | contact_phone_number |
| | contact_email |
| | contact_longitude |
| | contact_latitude |
| | contact_incoming_duration |
| | contact_outgoing_duration |
| | contact_missing_calls |
| | contact_sent_messages |
| | contact_received_messages |
| | is_in_speed_dial |

| | messages |
|---|---|
| PK | message_id |
| | remote_phone_number |
| | is_from_me |
| | message_text |
| | message_date |

| | calls |
|---|---|
| PK | call_id |
| | remote_phone_number |
| | is_from_me |
| | is_answered |
| | call_date |
| | call_duration |

*Figure 1 ER Diagram of the Database*

Inserting and retrieving operations are widely used in the application. Therefore DatabaseOperations class made as Singleton. That way we could be able to access the database anywhere within the application where the accessor has a Context object, without creating a new DatabaseOperations object.

## Broadcast Receivers

Broadcasts have a very important place in the application. Incoming and outgoing calls and SMS messages are received using Broadcasts. Those receivers are defined statically in the AndroidManifest.xml.

There are also custom actions that are specific to the application itself. Those actions are broadcasted in order to keep UI update. Those custom actions are defined in the Config class. Whenever an event, that requires UI to update, occurs those custom actions are broadcasted.

1. SmsEventOccurred

Broadcasted when a SMS message is received or sent.  Received by MainActivity in order to update the SMS Fragment.

2. CallEventOccurred

Broadcasted when there is an incoming/outgoing/missed call. Received by MainActivity in order to update the Calls Fragment.

3. ContactUpdated

Broadcasted when a contact is updated. Received by SingleContactActivity in order to update the UI. Note that a Contact is only editable from CreateOrEditContactActivity. CreateOrEditContactActivity is only accessible from SingleContactActivity. Therefore when a Contact is updated, SingleContactActivity needs to update the UI regardless which contact is updated. Because of that, broadcasted Intent does not have phone number information inside.

4. ContactsListChanged

Broadcasted when a new contact is added, removed or updated. Received by MainActivity in order to update the Contacts Fragment.

5. SpeedDialUpdated

Broadcasted when a contact is added to Speed Dial or removed from Speed Dial. Received by MainActivity in order to update the SpeedDial Fragment.

## Libraries Used

Several libraries are used in development of the project. Those libraries are:

- com.android.support:appcompat-v7:23.1.1
- com.android.support:design:23.1.1
- com.android.support:support-v4:23.1.1
- com.github.medyo:fancybuttons:1.5@aar
- com.google.android.gms:play-services:8.4.0
- com.googlecode.libphonenumber:libphonenumber:7.2.2
- com.github.rahatarmanahmed:circularprogressview:2.4.0

FancyButtons library is only used for creating rounded Buttons (or making corners rounded).

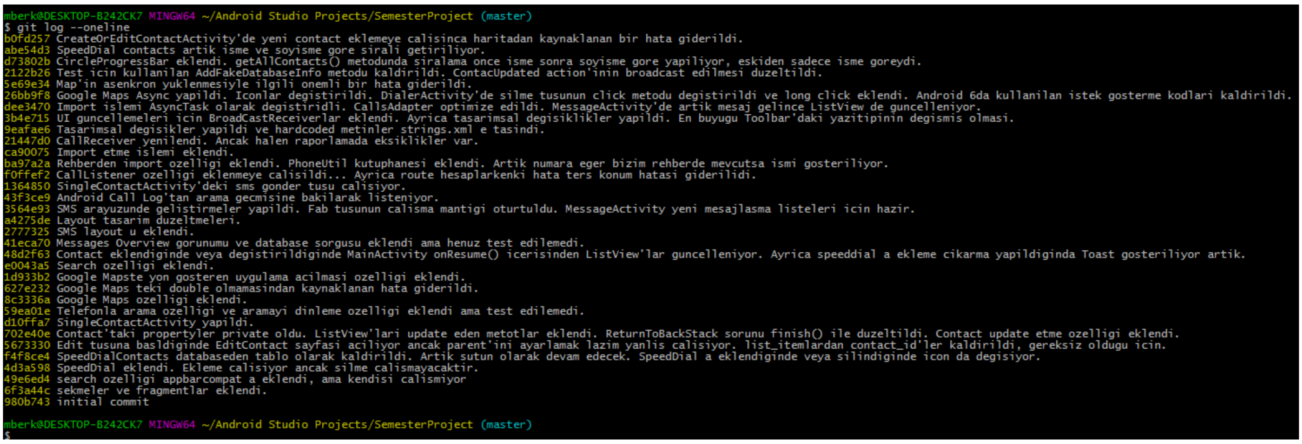
Google Play Services is a required library for Google Maps.

LibPhoneNumber library is used for formatting the phone numbers.

CircularProgressView is used for displaying a circular progress view while importing the contact list from phone's default phone book.

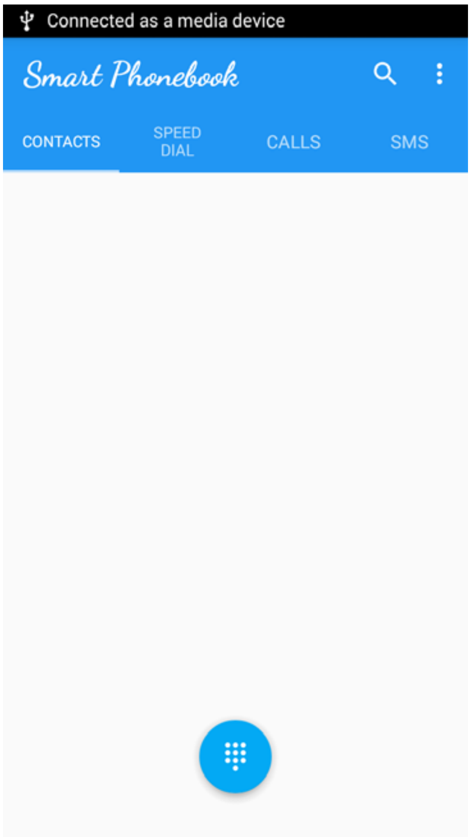
## Comments

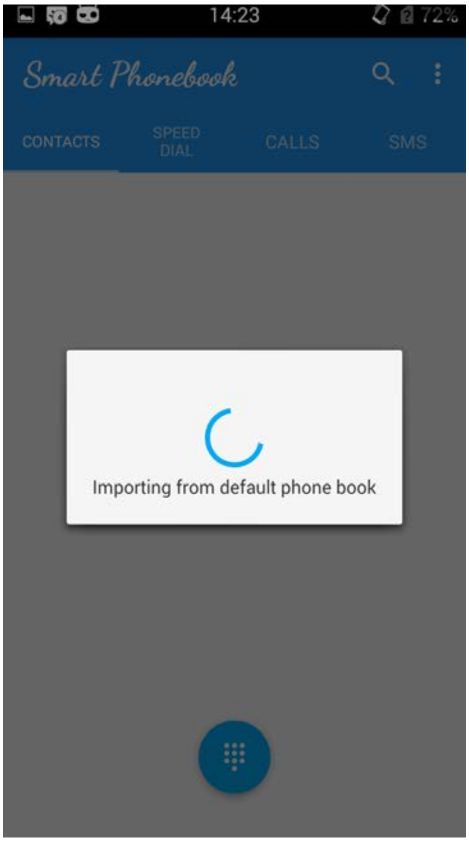
Git version control software is used through the development process. You can see the commits listed below:



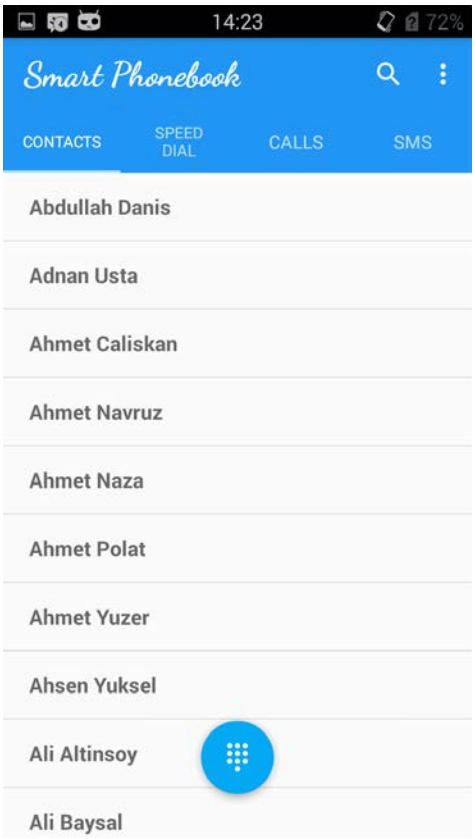
```
mberk@DESKTOP-B242CK7 MINGW64 ~/Android Studio Projects/SemesterProject (master)
$ git log --oneline
b0fd257 CreateOrEditContactActivity'de yeni contact eklemeye calisinca haritadan kaynaklanan bir hata giderildi.
abe54d3 SpeedDial contacts artik isme ve soyisme gore sirali getiriliyor.
d73802b CircleProgressBar eklendi. getAllContacts() metodunda siralama once isme sonra soyisme gore yapiliyor, eskiden sadece isme goreydi.
2122b26 Test icin kullanilan AddFakeDatabaseInfo metodu kaldirildi. ContacUpdated action'inin broadcast edilmesi duzeltildi.
5e69e34 Map'in asenkron yuklenmesiyle ilgili onemli bir hata giderildi.
26bb9f8 Google Maps Async yapildi. Iconlar degistirildi. DialerActivity'de silme tusunun click metodu degistirildi ve long click eklendi. Android 6da kullanilan istek gosterme kodlari kaldirildi.
dee3470 Import islemi AsyncTask olarak degistiridli. CallsAdapter optimize edildi. MessageActivity'de artik mesaj gelince ListView de guncelleniyor.
3b4e715 UI guncellemeleri icin BroadCastReceiverlar eklendi. Ayrica tasarimsal degisiklikler yapildi. En buyugu Toolbar'daki yazitipinin degismis olmasi.
9eafae6 Tasarimsal degisikler yapildi ve hardcoded metinler strings.xml e tasindi.
21447d0 CallReceiver yenilendi. Ancak halen raporlamada eksiklikler var.
ca90075 Import etme islemi eklendi.
ba97a2a Rehberden import ozelligi eklendi. PhoneUtil kutuphanesi eklendi. Artik numara eger bizim rehberde mevcutsa ismi gosteriliyor.
f0ffef2 CallListener ozelligi eklenmeye calisildi... Ayrica route hesaplarkenki hata ters konum hatasi giderilidi.
1364850 SingleContactActivity'deki sms gonder tusu calisiyor.
43f3ce9 Android Call Log'tan arama gecmisine bakilarak listeniyor.
3564e93 SMS arayuzunde gelistirmeler yapildi. Fab tusunun calisma mantigi oturtuldu. MessageActivity yeni mesajlasma listeleri icin hazir.
a4275de Layout tasarim duzeltmeleri.
2777325 SMS layout u eklendi.
41eca70 Messages Overview gorunumu ve database sorgusu eklendi ama henuz test edilemedi.
48d2f63 Contact eklendiginde veya degistirildiginde MainActivity onResume() icerisinden ListView'lar guncelleniyor. Ayrica speeddial a ekleme cikarma yapildiginda Toast gosteriliyor artik.
e0043a5 Search ozelligi eklendi.
1d933b2 Google Mapste yon gosteren uygulama acilmasi ozelligi eklendi.
627e232 Google Maps teki double olmamasindan kaynaklanan hata giderildi.
8c3336a Google Maps ozelligi eklendi.
59ea01e Telefonla arama ozelligi ve aramayi dinleme ozelligi eklendi ama test edilemedi.
d10ffa7 SingleContactActivity yapildi.
702e40e Contact'taki propertyler private oldu. ListView'lari update eden metotlar eklendi. ReturnToBackStack sorunu finish() ile duzeltildi. Contact update etme ozelligi eklendi.
5673330 Edit tusuna basldiginde EditContact sayfasi aciliyor ancak parent'ini ayarlamak lazim yanlis calisiyor. list_itemlardan contact_id'ler kaldirildi, gereksiz oldugu icin.
f4f8ce4 SpeedDialContacts databaseden tablo olarak kaldirildi. Artik sutun olarak devam edecek. SpeedDial a eklendiginde veya silindiginde icon da degisiyor.
4d3a598 SpeedDial eklendi. Ekleme calisiyor ancak silme calismayacaktir.
49e6ed4 search ozelligi appbarcompat a eklendi, ama kendisi calismiyor
6f3a44c sekmeler ve fragmentlar eklendi.
980b743 initial commit

mberk@DESKTOP-B242CK7 MINGW64 ~/Android Studio Projects/SemesterProject (master)
$
```

## Screenshots





Importing from default phone book



Abdullah Danis

Adnan Usta

Ahmet Caliskan

Ahmet Navruz

Ahmet Naza

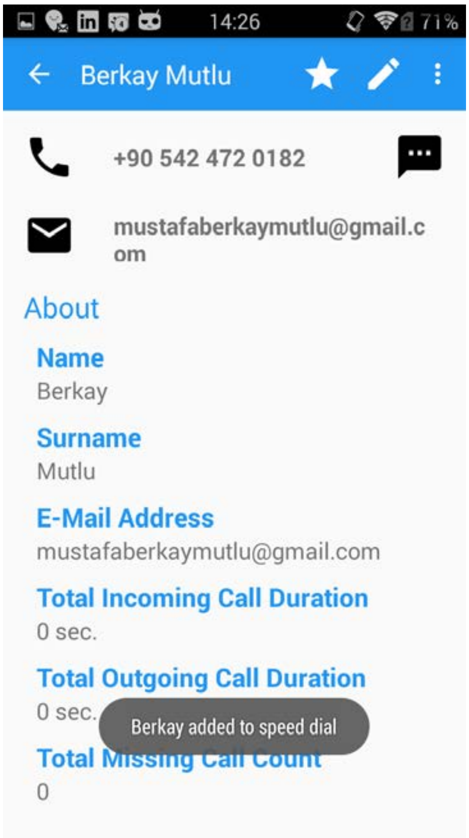
Ahmet Polat

Ahmet Yuzer

Ahsen Yuksel

Ali Altinsoy

Ali Baysal



Berkay Mutlu

+90 542 472 0182

mustafaberkaymutlu@gmail.com

About

Name
Berkay

Surname
Mutlu

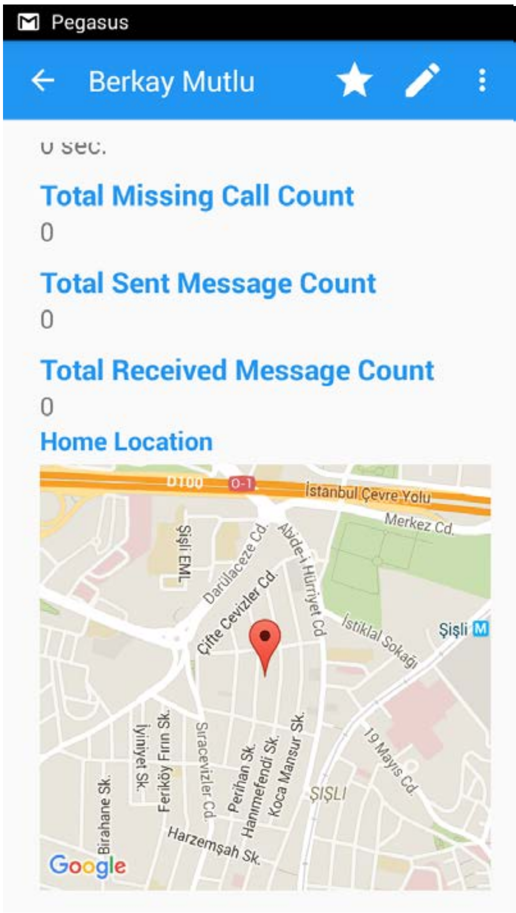
E-Mail Address
mustafaberkaymutlu@gmail.com

Total Incoming Call Duration
0 sec.

Total Outgoing Call Duration
0 sec.

Total Missing Call Count
0

Berkay added to speed dial



Pegasus

Berkay Mutlu

0 sec.

Total Missing Call Count

0

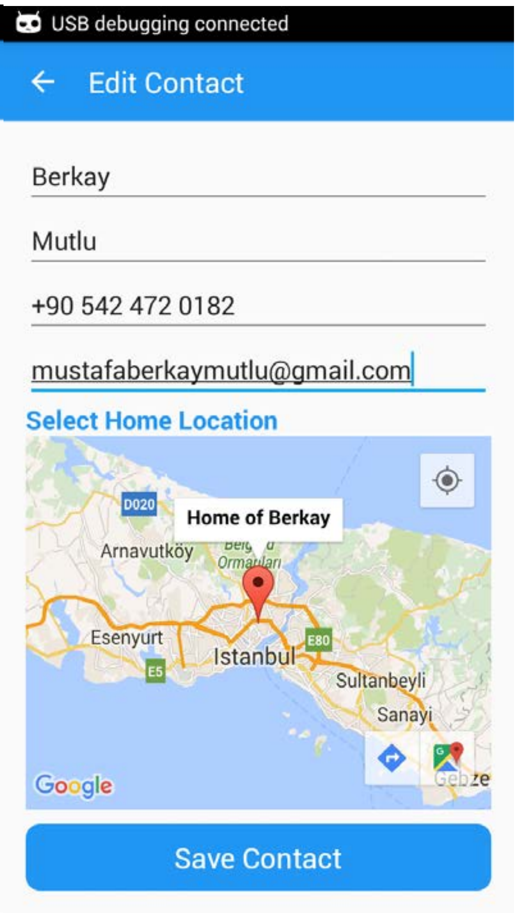
Total Sent Message Count

0

Total Received Message Count

0

Home Location



USB debugging connected

Edit Contact

Berkay

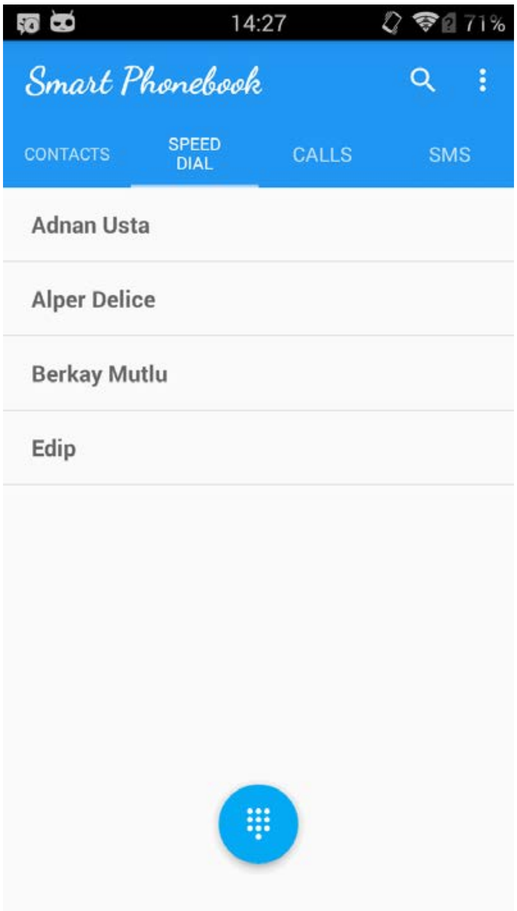
Mutlu

+90 542 472 0182

mustafaberkaymutlu@gmail.com

Select Home Location

Save Contact



14:27 71%

Smart Phonebook

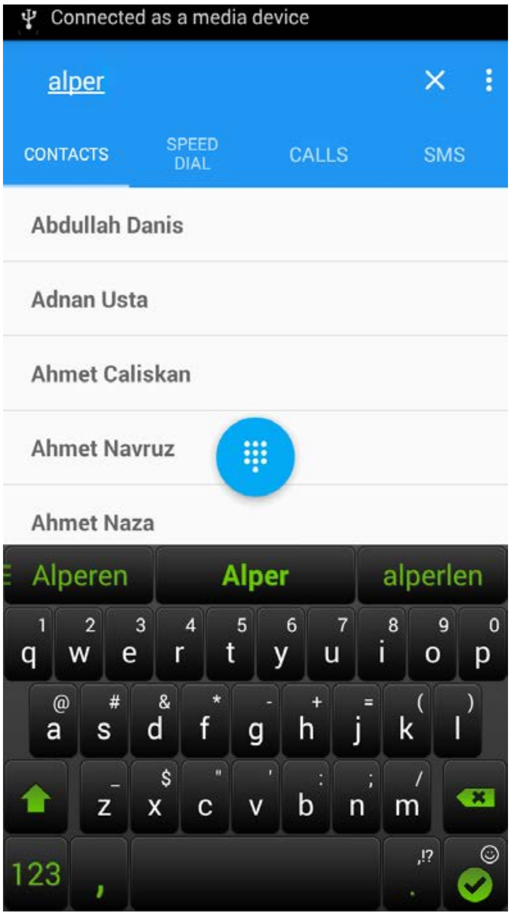
CONTACTS | SPEED DIAL | CALLS | SMS

Adnan Usta

Alper Delice

Berkay Mutlu

Edip



Connected as a media device

alper

CONTACTS | SPEED DIAL | CALLS | SMS

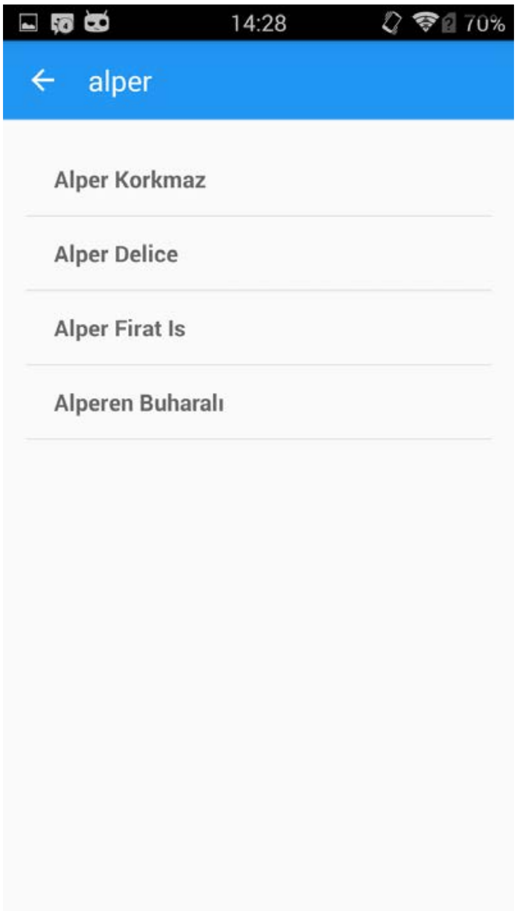
Abdullah Danis

Adnan Usta

Ahmet Caliskan

Ahmet Navruz

Ahmet Naza

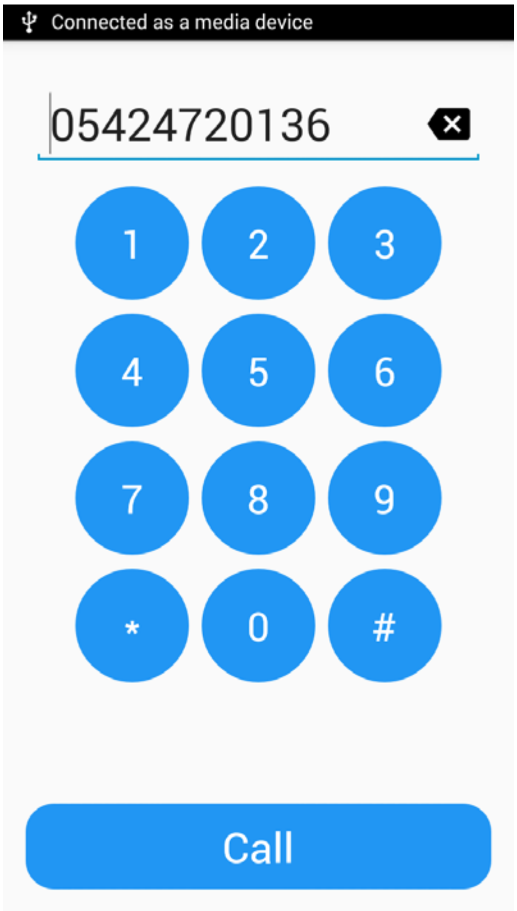


Alper Korkmaz

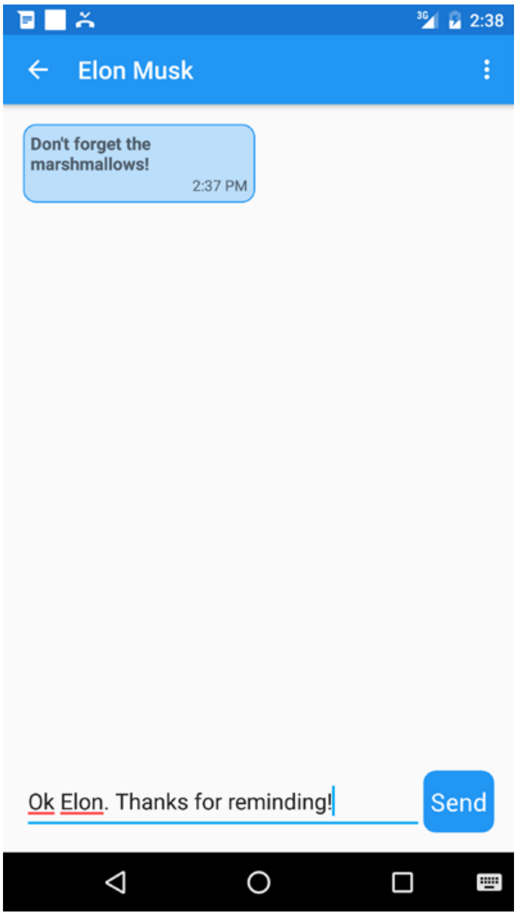
Alper Delice

Alper Firat Is

Alperen Buharalı

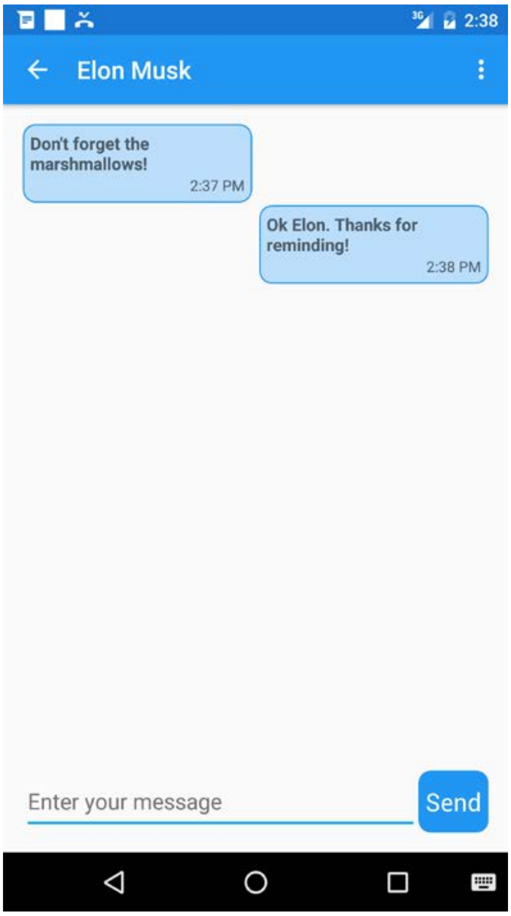


05424720136



Don't forget the marshmallows!
2:37 PM

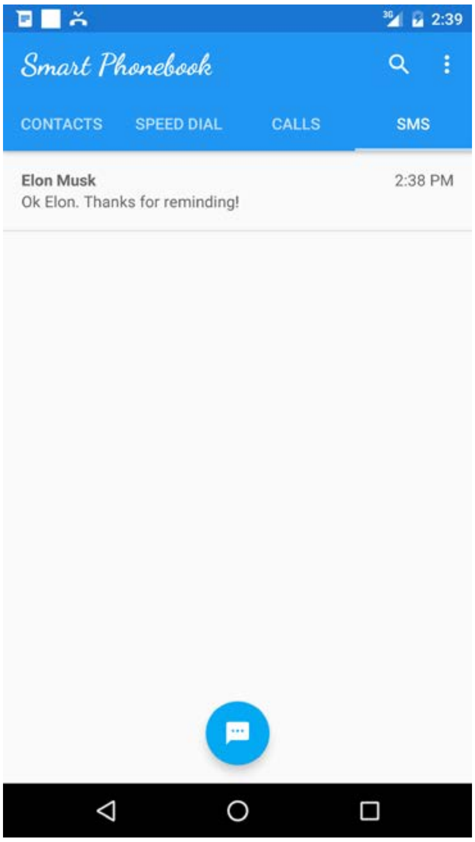
Ok Elon. Thanks for reminding!



Don't forget the marshmallows!
2:37 PM

Ok Elon. Thanks for reminding!
2:38 PM

Enter your message



Elon Musk 2:38 PM
Ok Elon. Thanks for reminding!