Android Chatt Demo using Firebase

by MyAppTemplates

Package
com.chatt.demo

com.chatt.demo
Class Chat

```
java.lang.Object
    │
    +-FragmentActivity
        │
        +-com.chatt.demo.custom.CustomActivity
            │
            +-com.chatt.demo.Chat
```

public class **Chat**
extends CustomActivity

The Class Chat is the Activity class that holds main chat screen. It shows all the conversation messages between two users and also allows the user to send and receive messages.

Fields

convList

private java.util.ArrayList **convList**

The Conversation list.

adp

private com.chatt.demo.Chat.ChatAdapter **adp**

The chat adapter.

txt

private EditText **txt**

The Editext to compose the message.

buddy

private com.chatt.demo.model.ChatUser **buddy**

The user name of buddy.

lastMsgDate

private java.util.Date **lastMsgDate**

The date of last message in conversation.

Constructors

Chat

public **Chat**()

(continued on next page)

Methods

onCreate

`protected void `**`onCreate`**`(Bundle savedInstanceState)`

onResume

`protected void `**`onResume`**`()`

onPause

`protected void `**`onPause`**`()`

onClick

`public void `**`onClick`**`(View v)`

sendMessage

`private void `**`sendMessage`**`()`

Call this method to Send message to opponent. It does nothing if the text is empty otherwise it creates a Parse object for Chat message and send it to Parse server.

loadConversationList

`private void `**`loadConversationList`**`()`

Load the conversation list from Parse server and save the date of last message that will be used to load only recent new messages

onOptionsItemSelected

`public boolean `**`onOptionsItemSelected`**`(MenuItem item)`

com.chatt.demo
Class Chat.ChatAdapter

```
java.lang.Object
    │
    +-BaseAdapter
        │
        +-com.chatt.demo.Chat.ChatAdapter
```

private class **Chat.ChatAdapter**
extends BaseAdapter

The Class ChatAdapter is the adapter class for Chat ListView. This adapter shows the Sent or Receieved Chat message in each list item.

Constructors

ChatAdapter

```
private ChatAdapter()
```

Methods

getCount

```
public int getCount()
```

getItem

```
public Conversation getItem(int arg0)
```

getItemId

```
public long getItemId(int arg0)
```

getView

```
public View getView(int pos,
        View v,
        ViewGroup arg2)
```

com.chatt.demo
Class ChattApp

```
java.lang.Object
    │
    +-Application
        │
        +-com.chatt.demo.ChattApp
```

public class **ChattApp**
extends Application

The Class ChattApp is the Main Application class of this app. The onCreate method of this class initializes the Parse.

Fields

database

private FirebaseDatabase **database**

> The Firebase database

firebaseAuth

private FirebaseAuth **firebaseAuth**

> Firebase Authentication component

Constructors

ChattApp

public **ChattApp**()

Methods

onCreate

public void **onCreate**()

com.chatt.demo
Class Login

```
java.lang.Object
    │
    +-FragmentActivity
        │
        +-com.chatt.demo.custom.CustomActivity
            │
            +-com.chatt.demo.Login
```

public class **Login**
extends CustomActivity

The Class Login is an Activity class that shows the login screen to users. The current implementation simply includes the options for Login and button for Register. On login button click, it sends the Login details to Parse server to verify user.

Fields

user

private EditText **user**

> The username edittext.

pwd

private EditText **pwd**

> The password edittext.

loginProgressDlg

private ProgressDialog **loginProgressDlg**

> Login progress dialog

Constructors

Login

public **Login**()

Methods

onCreate

protected void **onCreate**(Bundle savedInstanceState)

onClick

```
public void onClick(View v)
```

onActivityResult

```
protected void onActivityResult(int requestCode,
        int resultCode,
        Intent data)
```

onClick

```
public void onClick(View v)
```

com.chatt.demo
Class Register

```
java.lang.Object
    │
    +-FragmentActivity
        │
        +-com.chatt.demo.custom.CustomActivity
            │
            +-com.chatt.demo.Register
```

public class **Register**
extends CustomActivity

The Class Register is the Activity class that shows user registration screen that allows user to register itself on Parse server for this Chat app.

Fields

pwd

private EditText **pwd**

> The password EditText.

email

private EditText **email**

> The email EditText.

displayName

private EditText **displayName**

> The displayName EditText.

registerProgressDlg

private ProgressDialog **registerProgressDlg**

> Register progress dialog

Constructors

Register

public **Register**()

Methods

(continued from last page)

onCreate

```
protected void onCreate(Bundle savedInstanceState)
```

onClick

```
public void onClick(View v)
```

onCreate

```
protected void onCreate(Bundle savedInstanceState)
```

com.chatt.demo
Class UserList

```
java.lang.Object
    │
    +-FragmentActivity
        │
        +-com.chatt.demo.custom.CustomActivity
            │
            +-com.chatt.demo.UserList
```

public class **UserList**
extends CustomActivity

The Class UserList is the Activity class. It shows a list of all users of this app. It also shows the Offline/Online status of users.

Fields

database

 DatabaseReference **database**

> Users database reference

uList

private java.util.ArrayList **uList**

> The Chat list.

user

public static com.chatt.demo.model.ChatUser **user**

> The user.

Constructors

UserList

public **UserList**()

Methods

onCreate

protected void **onCreate**(Bundle savedInstanceState)

onDestroy

```
protected void onDestroy()
```

onResume

```
protected void onResume()
```

updateUserStatus

```
private void updateUserStatus(boolean online)
```

Update user status.

Parameters:
online - true if user is online

loadUserList

```
private void loadUserList()
```

Load list of users.

com.chatt.demo
Class UserList.UserAdapter

```
java.lang.Object
    │
    +-BaseAdapter
        │
        +-com.chatt.demo.UserList.UserAdapter
```

private class **UserList.UserAdapter**
extends BaseAdapter

The Class UserAdapter is the adapter class for User ListView. This adapter shows the user name and it's only online status for each item.

Constructors

UserAdapter

```
private UserAdapter()
```

Methods

getCount

```
public int getCount()
```

getItem

```
public ChatUser getItem(int arg0)
```

getItemId

```
public long getItemId(int arg0)
```

getView

```
public View getView(int pos,
        View v,
        ViewGroup arg2)
```

Package
com.chatt.demo.custom

com.chatt.demo.custom
Class CustomActivity

```
java.lang.Object
    |
    +-FragmentActivity
        |
        +-com.chatt.demo.custom.CustomActivity
```

Direct Known Subclasses:
      Chat, Login, Register, UserList

---

public class **CustomActivity**
extends FragmentActivity

This is a common activity that all other activities of the app can extend to inherit the common behaviors like implementing a common interface that can be used in all child activities.

Fields

TOUCH

```
public static final com.chatt.demo.utils.TouchEffect TOUCH
```

Apply this Constant as touch listener for views to provide alpha touch effect. The view must have a Non-Transparent background.

Constructors

CustomActivity

```
public CustomActivity()
```

Methods

setContentView

```
public void setContentView(int layoutResID)
```

setupActionBar

```
protected void setupActionBar()
```

This method will setup the top title bar (Action bar) content and display values. It will also setup the custom background theme for ActionBar. You can override this method to change the behavior of ActionBar for particular Activity

setTouchNClick

```
public View setTouchNClick(int id)
```

Sets the touch and click listener for a view with given id.

Parameters:
id - the id

Returns:
the view on which listeners applied

setClick

```
public View setClick(int id)
```

Sets the click listener for a view with given id.

Parameters:
id - the id

Returns:
the view on which listener is applied

onClick

```
public void onClick(View v)
```

Package

com.chatt.demo.model

com.chatt.demo.model
Class ChatUser

```
java.lang.Object
    |
    +-com.chatt.demo.model.ChatUser
```

All Implemented Interfaces:
> java.io.Serializable

---

public class **ChatUser**
extends java.lang.Object
implements java.io.Serializable

The Class ChatUse is a Java Bean class that represents a single user.

Fields

id

```
public java.lang.String id
```

---

username

```
public java.lang.String username
```

---

email

```
public java.lang.String email
```

---

online

```
public java.lang.Boolean online
```

---

room

```
public java.util.ArrayList room
```

Constructors

ChatUser

```
public ChatUser()
```

ChatUser

```
public ChatUser(java.lang.String id,
                java.lang.String username,
                java.lang.String email,
                java.lang.Boolean online,
                java.util.ArrayList room)
```

Methods

getUsername

```
public java.lang.String getUsername()
```

getId

```
public java.lang.String getId()
```

getEmail

```
public java.lang.String getEmail()
```

isOnline

```
public java.lang.Boolean isOnline()
```

getRoom

```
public java.util.ArrayList getRoom()
```

setId

```
public void setId(java.lang.String id)
```

setUsername

```
public void setUsername(java.lang.String username)
```

setEmail

```
public void setEmail(java.lang.String email)
```

setOnline

```
public void setOnline(java.lang.Boolean online)
```

setRoom

```
public void setRoom(java.util.ArrayList room)
```

```
public void setEmail(java.lang.String email)
```

com.chatt.demo.model
Class Conversation

```
java.lang.Object
    │
    +-com.chatt.demo.model.Conversation
```

public class **Conversation**
extends java.lang.Object

The Class Conversation is a Java Bean class that represents a single chat conversation message.

Fields

STATUS_SENDING

public static final int **STATUS_SENDING**

The Constant STATUS_SENDING.
Constant value: **0**

STATUS_SENT

public static final int **STATUS_SENT**

The Constant STATUS_SENT.
Constant value: **1**

STATUS_FAILED

public static final int **STATUS_FAILED**

The Constant STATUS_FAILED.
Constant value: **2**

msg

private java.lang.String **msg**

The msg.

status

private int **status**

The status.

date

private java.util.Date **date**

The date.

sender

```
private java.lang.String sender
```

The sender.

receiver

```
private java.lang.String receiver
```

The receiver

photoUrl

```
private java.lang.String photoUrl
```

The photo url.

Constructors

Conversation

```
public Conversation(java.lang.String msg,
                    java.util.Date date,
                    java.lang.String sender,
                    java.lang.String receiver,
                    java.lang.String photoUrl)
```

Instantiates a new conversation.

Parameters:
    msg - the msg
    date - the date
    sender - the sender
    receiver - the receiver
    photoUrl - the photo url

Methods

getMsg

```
public java.lang.String getMsg()
```

Gets the msg.

Returns:
    the msg

setMsg

```
public void setMsg(java.lang.String msg)
```

Sets the msg.

Parameters:
    msg - the new msg

isSent

```
public boolean isSent()
```

Checks if is sent.

Returns:
true, if is sent

getDate

```
public java.util.Date getDate()
```

Gets the date.

Returns:
the date

setDate

```
public void setDate(java.util.Date date)
```

Sets the date.

Parameters:
date - the new date

getReceiver

```
public java.lang.String getReceiver()
```

Gets the sender.

Returns:
the sender

setReceiver

```
public void setReceiver(java.lang.String receiver)
```

Sets the sender.

Parameters:
receiver - the new sender

getSender

```
public java.lang.String getSender()
```

Gets the sender.

Returns:
the sender

setSender

```
public void setSender(java.lang.String sender)
```

Sets the sender.

Parameters:
    `sender` - the new sender

getStatus

`public int `**`getStatus`**`()`

Gets the status.

Returns:
    the status

setStatus

`public void `**`setStatus`**`(int status)`

Sets the status.

Parameters:
    `status` - the new status

setPhotoUrl

`public void `**`setPhotoUrl`**`(java.lang.String photoUrl)`

Set the photo URL

Parameters:
    `photoUrl`

getPhotoUrl

`public java.lang.String `**`getPhotoUrl`**`()`

Get the photo URL

Returns:
    photoUrl

Package
com.chatt.demo.utils

com.chatt.demo.utils
Class Const

```
java.lang.Object
    │
    +-com.chatt.demo.utils.Const
```

public class **Const**
extends java.lang.Object

The Class Const is a single common class to hold all the app Constants.

Fields

EXTRA_DATA

```
public static final java.lang.String EXTRA_DATA
```

> The Constant EXTRA_DATA.
> Constant value: **extraData**

Constructors

Const

```
public Const()
```

com.chatt.demo.utils
Class TouchEffect

```
java.lang.Object
    |
    +-com.chatt.demo.utils.TouchEffect
```

public class **TouchEffect**
extends java.lang.Object

The Class TouchEffect is the implementation of OnTouchListener interface. You can apply this to views mostly Buttons to provide Touch effect and that view must have a valid background. The current implementation simply set Alpha value of View background.

Constructors

TouchEffect

public **TouchEffect**()

Methods

onTouch

```
public boolean onTouch(View v,
        MotionEvent event)
```

com.chatt.demo.utils
Class Utils

```
java.lang.Object
    │
    +-com.chatt.demo.utils.Utils
```

public class **Utils**
extends java.lang.Object

The Class Utils is a common class that hold many kind of different useful utility methods.

Constructors

Utils

public **Utils**()

Methods

showDialog

```
public static AlertDialog showDialog(Context ctx,
        java.lang.String msg,
        java.lang.String btn1,
        java.lang.String btn2,
        DialogInterface.OnClickListener listener1,
        DialogInterface.OnClickListener listener2)
```

Show dialog.

Parameters:
ctx - the ctx
msg - the msg
btn1 - the btn1
btn2 - the btn2
listener1 - the listener1
listener2 - the listener2

Returns:
the alert dialog

showDialog

```
public static AlertDialog showDialog(Context ctx,
        int msg,
        int btn1,
        int btn2,
        DialogInterface.OnClickListener listener1,
        DialogInterface.OnClickListener listener2)
```

Show dialog.

Parameters:

ctx - the ctx
msg - the msg
btn1 - the btn1
btn2 - the btn2
listener1 - the listener1
listener2 - the listener2

Returns:
the alert dialog

showDialog

```
public static AlertDialog showDialog(Context ctx,
        java.lang.String msg,
        java.lang.String btn1,
        java.lang.String btn2,
        DialogInterface.OnClickListener listener)
```

Show dialog.

Parameters:
ctx - the ctx
msg - the msg
btn1 - the btn1
btn2 - the btn2
listener - the listener

Returns:
the alert dialog

showDialog

```
public static AlertDialog showDialog(Context ctx,
        int msg,
        int btn1,
        int btn2,
        DialogInterface.OnClickListener listener)
```

Show dialog.

Parameters:
ctx - the ctx
msg - the msg
btn1 - the btn1
btn2 - the btn2
listener - the listener

Returns:
the alert dialog

showDialog

```
public static AlertDialog showDialog(Context ctx,
        java.lang.String msg,
        DialogInterface.OnClickListener listener)
```

Show dialog.

Parameters:
ctx - the ctx

> `msg` - the msg
> `listener` - the listener

> **Returns:**
>> the alert dialog

showDialog

```
public static AlertDialog showDialog(Context ctx,
        int msg,
        DialogInterface.OnClickListener listener)
```

> Show dialog.

> **Parameters:**
>> `ctx` - the ctx
>> `msg` - the msg
>> `listener` - the listener

> **Returns:**
>> the alert dialog

showDialog

```
public static AlertDialog showDialog(Context ctx,
        java.lang.String msg)
```

> Show dialog.

> **Parameters:**
>> `ctx` - the ctx
>> `msg` - the msg

> **Returns:**
>> the alert dialog

showDialog

```
public static AlertDialog showDialog(Context ctx,
        int msg)
```

> Show dialog.

> **Parameters:**
>> `ctx` - the ctx
>> `msg` - the msg

> **Returns:**
>> the alert dialog

showDialog

```
public static void showDialog(Context ctx,
        int title,
        int msg,
        DialogInterface.OnClickListener listener)
```

> Show dialog.

> **Parameters:**

        `ctx` - the ctx
        `title` - the title
        `msg` - the msg
        `listener` - the listener

hideKeyboard

```
public final static void hideKeyboard(Activity ctx)
```

Hide keyboard.

Parameters:
        `ctx` - the ctx

hideKeyboard

```
public final static void hideKeyboard(Activity ctx,
        View v)
```

Hide keyboard.

Parameters:
        `ctx` - the ctx
        `v` - the v

sizeFromIterable

```
public static int sizeFromIterable(java.lang.Iterable iterable)
```

Index