

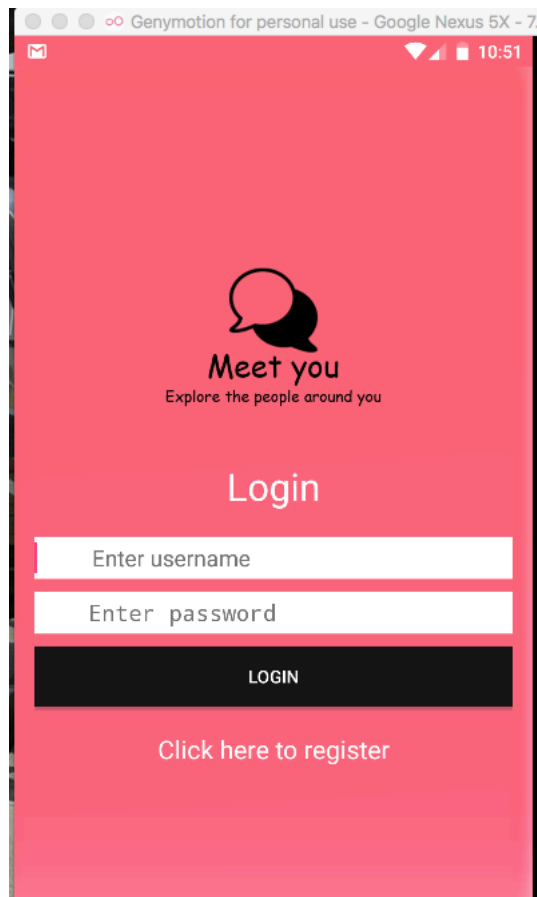
Android Final Project Report: MeetYou

Chien Chih Kao ckao01@syr.edu

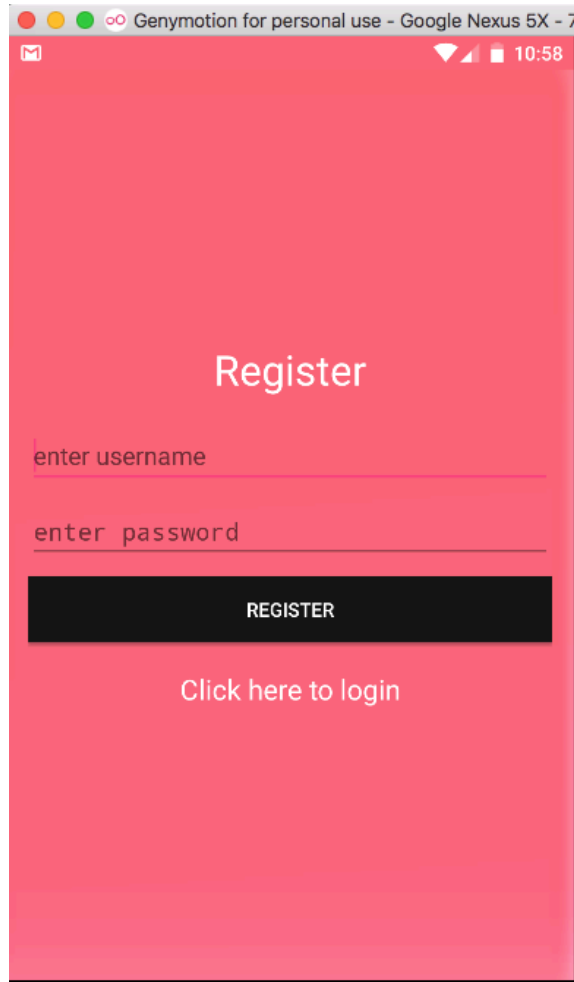
1.Login Page

1.1 UI

1.1.1 Login:



1.1.2 Register



1.2Description:

User can register a account and use the account to login. The information is stored in the firebase.

1.3Technical Feature:

Use firebase api to access the Firebase database.

1.4Code Snippet:

```

loginButton.setOnClickListener((v) -> {
    user = username.getText().toString();
    pass = password.getText().toString();

    if(user.equals("")){
        username.setError("can't be blank");
    }
    else if(pass.equals("")){
        password.setError("can't be blank");
    }
    else{
        String url = "https://mychatapp-dcb19.firebaseio.com/users.json";
        final ProgressDialog pd = new ProgressDialog(Login.this);
        pd.setMessage("Loading...");
        pd.show();

        StringRequest request = new StringRequest(Request.Method.GET, url, (s) -> {
            if(s.equals("null")){
                Toast.makeText(Login.this, "user not found", Toast.LENGTH_LONG).show();
            }
            else{
                try {
                    JSONObject obj = new JSONObject(s);

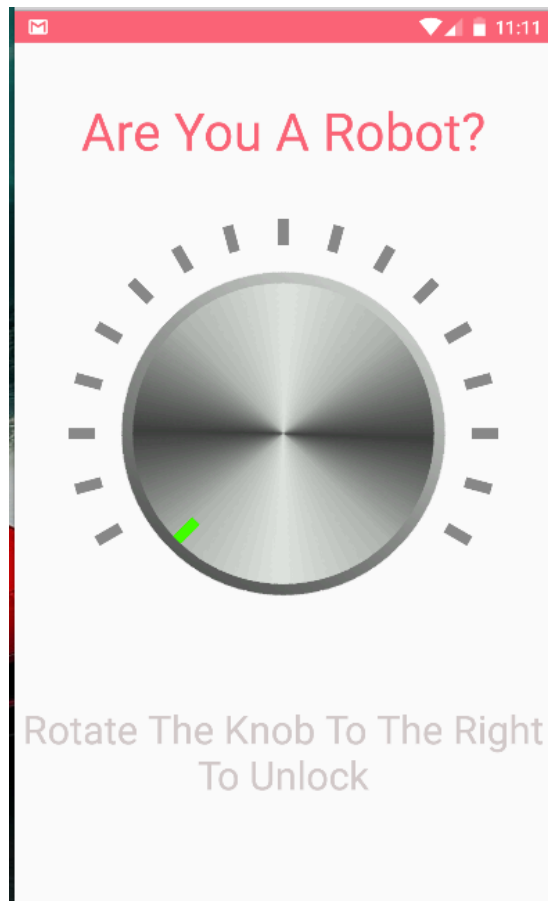
                    if(!obj.has(user)){
                        Toast.makeText(Login.this, "user not found", Toast.LENGTH_LONG).show();
                    }
                    else {
                        if (obj.getJSONObject(user).getString("password").equals(pass)) {
                            UserDetails.username = user;
                            UserDetails.password = pass;
                            //startActivity(new Intent(Login.this, Users.class));
                            //startActivity(new Intent(Login.this, Main_activity.class));
                            startActivity(new Intent(Login.this, Knob_Unlock_Activity.class));
                        } else {
                            Toast.makeText(Login.this, "incorrect password", Toast.LENGTH_LONG).show();
                        }
                    }
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        }, (volleyError) -> {
            System.out.println("" + volleyError);
            pd.dismiss();
        });

        RequestQueue rQueue = Volley.newRequestQueue(Login.this);
        rQueue.add(request);
    }
});

```

2. Unlock Page

2.1 UI



2.2Description:

User rotate the knob to the right to right to go to the main page.

2.3Technical Feature:

Custom View

2.4Code Snippet:

```
<com.androidchatapp.FancyRotaryKnobView
    android:layout_width="fill_parent"
    android:layout_height="0dp"
    android:layout_weight="4"
    android:gravity="center"
    android:id="@+id/fancyRotaryKnobView"/>
```

The custom view in the layout file

```

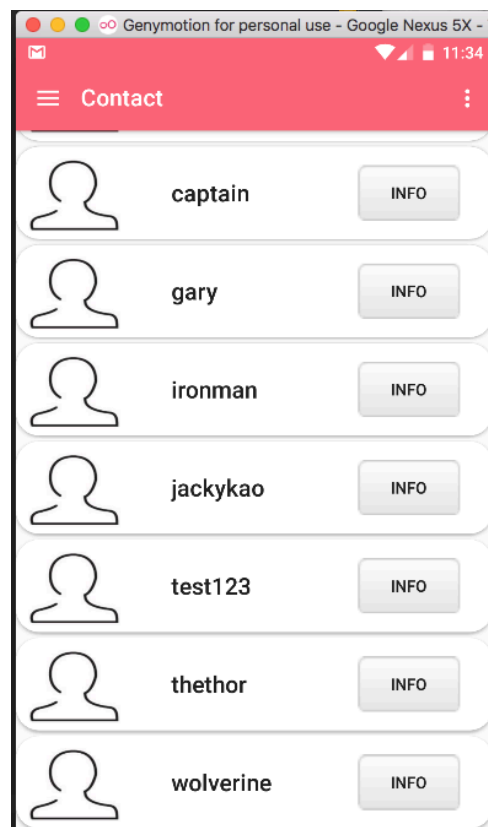
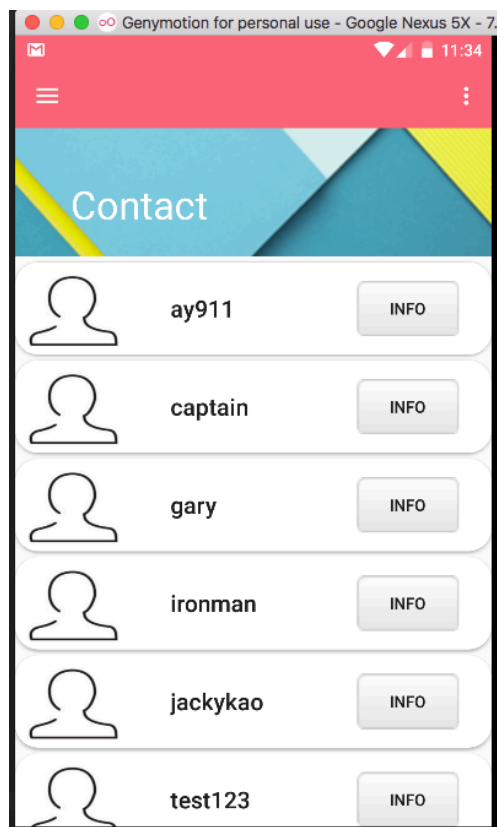
        (FancyRotaryKnobView) findViewById(R.id.fancy_rotary_knob_view);
        rotaryKnobView.setKnobListener(new FancyRotaryKnobView.RotaryKnobListener() {
            @Override
            public void onKnobChanged(int arg) {
                status.setText(Integer.toString(arg));
                if(arg==300) {
                    unlock.Unlock();
                    Toast.makeText(getApplicationContext(), "Unlock Successfully", Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}

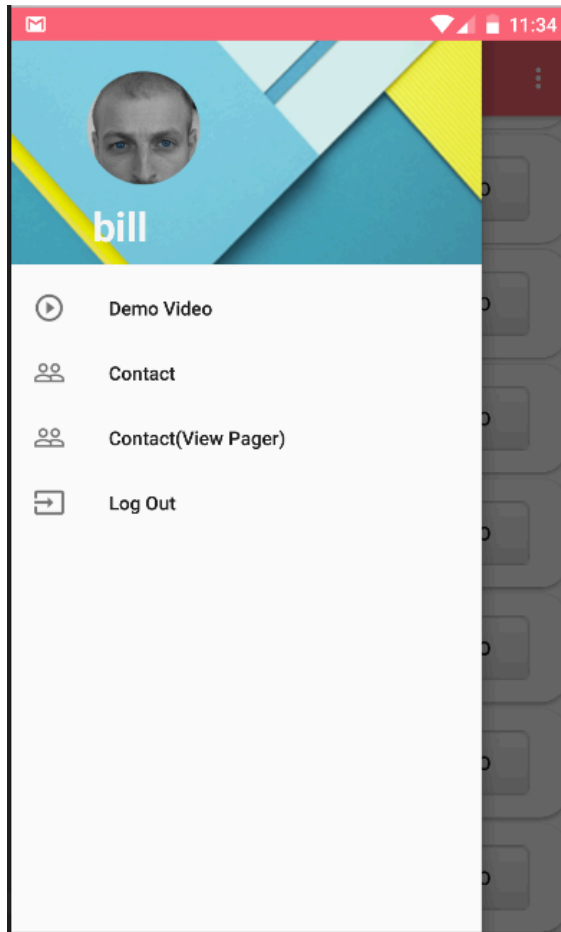
```

When rotate to the right side, go to the chat_activity

3. Contact list Page

3.1 UI





2.2Description:

The Page display all the other user stored in the database and user can click them to chat.

2.3Technical Feature:

NavDrawer, Recycler view, Coordinate layout

2.4Code Snippet:

Coordinate Layout:

```

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent">

    <android.support.design.widget.AppBarLayout
        android:id="@+id/appbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">
        <android.support.design.widget.CollapsingToolbarLayout
            android:id="@+id/collapsing_toolbar"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:fitsSystemWindows="true"
            app:expandedTitleMarginStart="48dp"
            app:expandedTitleMarginEnd="64dp"
            app:contentScrim="?attr/colorPrimary"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">
            <ImageView
                android:id="@+id/backdrop"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:scaleType="centerCrop"
                android:fitsSystemWindows="true"
                app:layout_collapseMode="parallax"
                android:src="@drawable/header"
            />
            <android.support.v7.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimaryDark"
                app:layout_collapseMode="pin"
                app:popupTheme="@style/ThemeOverlay.AppCompat.Light">

            </android.support.v7.widget.Toolbar>
        </android.support.design.widget.CollapsingToolbarLayout>
    </android.support.design.widget.AppBarLayout>
    <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/container"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MainActivity"
        android:layout_below="@+id/toolbar"
        app:layout_behavior="android.support.design.widget.AppBarLayout$Scroll
        tools:ignore="MergeRootFrame" />
</android.support.design.widget.CoordinatorLayout>

```

```

Toolbar toolbar= (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
CollapsingToolbarLayout collapsingToolbar = (CollapsingToolbarLayout) findViewById(R.id.collapsing_toolbar);
collapsingToolbar.setTitle("Contact");

```

NavDrawer:

```

NavigationView navigationView = (android.support.design.widget.NavigationView) findViewById(R.id.navigation_view);
View headerView = navigationView.getHeaderView(0);
TextView navUsername = (TextView) headerView.findViewById(R.id.header_name);
navUsername.setText(UserDetails.username);

navigationView.setNavigationItemSelectedListener(this);

drawerLayout = (DrawerLayout) findViewById(R.id.drawer);

ActionBarDrawerToggle toggler = new ActionBarDrawerToggle(this, drawerLayout, toolbar, "Open", "Close"){
    @Override
    public void onDrawerClosed(View drawerView) { super.onDrawerClosed(drawerView); }

    @Override
    public void onDrawerOpened(View drawerView) { super.onDrawerOpened(drawerView); }
};
drawerLayout.setDrawerListener(toggler);

```

Recycler View

MyRecyclerView.java

```
public void onBindViewHolder(ViewHolder holder, int position) {
    String contactname=contact.get(position);
    //Map<String,?> movie = movieData.get(position);
    //holder.bindMovieData(movie);
    holder.contactName.setText(contactname);
}

@Override
public int getItemCount() { return contact.size(); }

public class ViewHolder extends RecyclerView.ViewHolder{

    public TextView contactName;

    public ViewHolder(View itemView) {
        super(itemView);

        contactName=(TextView)itemView.findViewById(R.id.contactNmae);

        itemView.setOnClickListener((v) -> {
            if(mItemClickListener!=null)
                mItemClickListener.onItemClick(v,getPosition());
        });
    }
}
```

Fragment.java


```

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, final Bundle savedInstanceState) {
    //ArrayList<String> al = new ArrayList<>();
    pd = new ProgressDialog(getContext());
    pd.setMessage("Loading...");
    pd.show();
    String url = "https://mychatapp-dcb19.firebaseio.com/users.json";

    StringRequest request = new StringRequest(Request.Method.GET, url, new Response.Listener<String>(){
        @Override
        public void onResponse(String s) {
            doOnSuccess(s);
        }
    }, (volleyError) -> { System.out.println(" " + volleyError); });

    RequestQueue rQueue = Volley.newRequestQueue(getContext());
    rQueue.add(request);

    System.out.println(" " + al.size());
    final View rootView = inflater.inflate(R.layout.recyclerview_layout, container, false);
    recyclerView = (RecyclerView) rootView.findViewById(R.id.cardList);
    //recyclerView.setHasFixedSize(true);
    layoutManager = new LinearLayoutManager(getActivity());
    recyclerView.setLayoutManager(layoutManager);
    return rootView;
}

public void doOnSuccess(String s){
    try {
        JSONObject obj = new JSONObject(s);

        Iterator i = obj.keys();
        String key = "";

        while(i.hasNext()){
            key = i.next().toString();

            if(!key.equals(UserDetails.username)) {
                al.add(key);
            }

            totalUsers++;
        }

    } catch (JSONException e) {
        e.printStackTrace();
    }

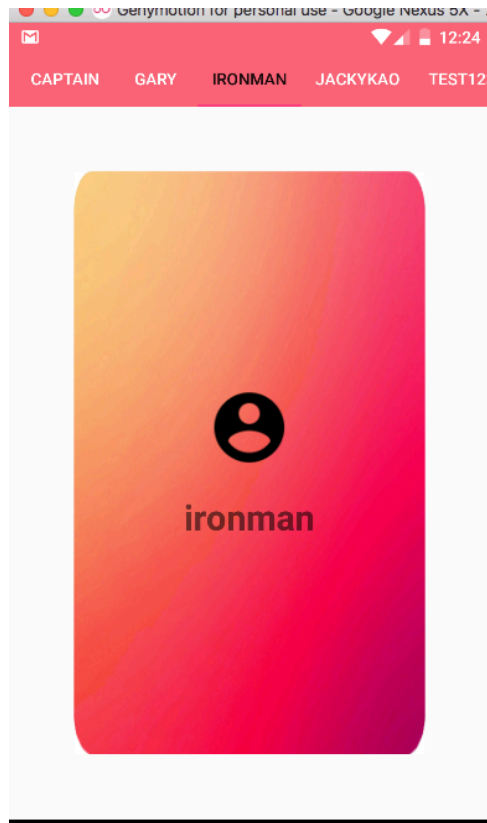
    recyclerViewAdapter = new MyRecyclerViewAdapter(getActivity(), al);
    recyclerView.setAdapter(recyclerViewAdapter);

    pd.dismiss();
    recyclerViewAdapter.setOnItemClickListener((view, position) -> {
        onCardItemClickedListener.onCardItemClicked(al.get(position));
    });
}
}

```

4. Contact list Page(View pager)

4.1 UI



2.2Description:

The ViewPager Page displays all the other user stored in the database and user can click them to chat.

2.3Technical Feature:

ViewPager, Advanced Animation

2.4 Code Snippet:

ViewPager.java

```

public class Contact_ViewPager extends AppCompatActivity implements Fragment_DetailView.OnContactImageClickListener {
    MyFragmentManager myFragmentManager;
    MyFragmentManagerAdapter myPagerAdapter;
    ViewPager mViewPager;
    ArrayList<String> al = new ArrayList<>();
    int totalUsers = 0;
    ProgressDialog pd;
    @Override
    protected void onCreate(Bundle savedInstanceState){
        super.onCreate(savedInstanceState);
        setContentView(R.layout.viewerpager);
        //pd = new ProgressDialog(this);
        //pd.setMessage("Loading...");
        //pd.show();
        String url = "https://mychatapp-dcb19.firebaseio.com/users.json";

        String request = new StringRequest(Request.Method.GET, url, (s) -> {
            doOnSuccess(s);
        }, (volleyError) -> { System.out.println(" " + volleyError); });
        RequestQueue rQueue = Volley.newRequestQueue(this);
        rQueue.add(request);
    }
    public void doOnSuccess(String s){
        try {
            JSONObject obj = new JSONObject(s);

            Iterator i = obj.keys();
            String key = "";

            while(i.hasNext()){
                key = i.next().toString();

                if(!key.equals(UserDetails.username)) {
                    al.add(key);
                }
                totalUsers++;
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }
        myPagerAdapter=new MyFragmentManagerAdapter(getSupportFragmentManager(),al);
        mViewPager=(ViewPager) findViewById(R.id.pager);
        mViewPager.setAdapter(myPagerAdapter);

        mViewPager.setCurrentItem(2);
        mViewPager.setPageTransformer(true, new FlipHorizontalTransformer());
        TabLayout tabLayout=(TabLayout)findViewById(R.id.tabs);
        tabLayout.setupWithViewPager(mViewPager);
        //pd.dismiss();
    }
    @Override
    public void OnContactImageClick(String contact)
    {
        UserDetails.chatWith = contact;
        startActivity(new Intent(this, Chat_Activity.class));
    }
}

```

Animation:

```

public class FlipHorizontalTransformer extends BaseTransformer {

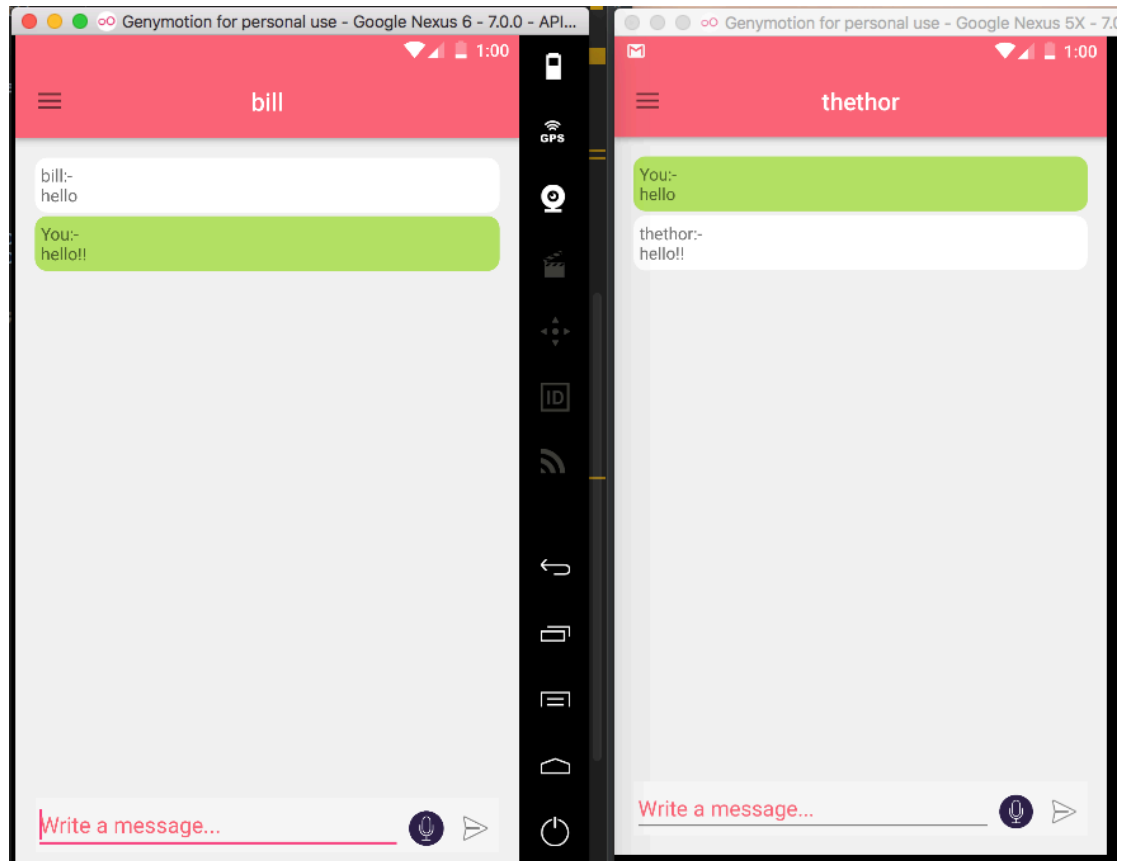
    @Override
    protected void onTransform(View view, float position) {
        final float rotation = 180f * position;

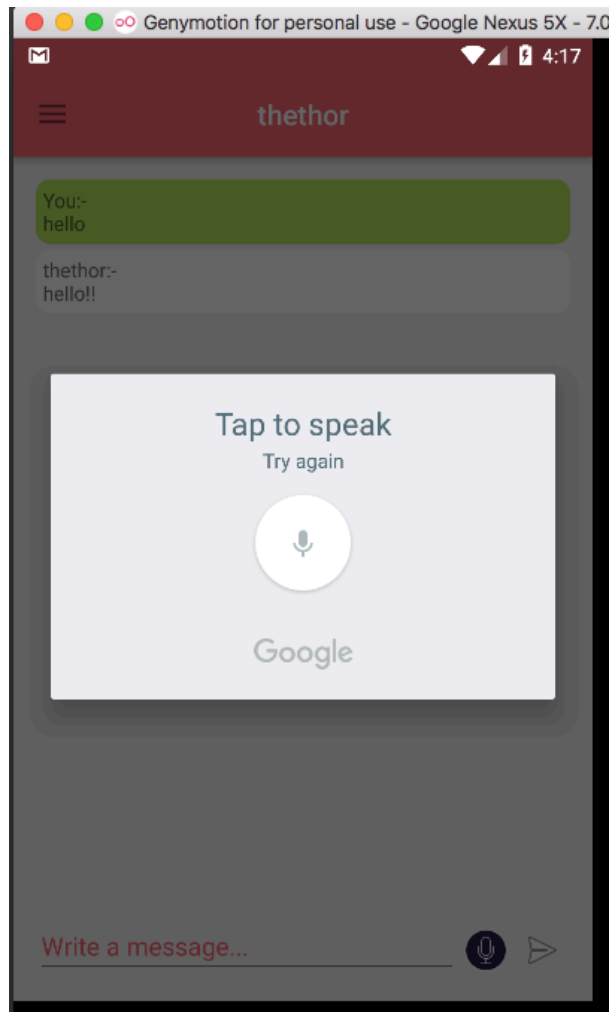
        view.setAlpha(rotation > 90f || rotation < -90f ? 0 : 1);
        view.setPivotX(view.getWidth() * 0.5f);
        view.setPivotY(view.getHeight() * 0.5f);
        view.setRotationY(rotation);
    }
}

```

5. Chat Page

5.1 UI





5.2Description:

User can use the app to do the real time chat and it also supply the voice to text functionality

2.3Technical Feature:

firebase Real time, voice to text

2.4 Code Snippet:

chat.java

```

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, final Bundle savedInstanceState) {
    View rootView = inflater.inflate(R.layout.activity_chat, container, false);
    layout = (LinearLayout) rootView.findViewById(R.id.layout1);
    sendButton = (ImageView) rootView.findViewById(R.id.sendButton);
    messageArea = (EditText) rootView.findViewById(R.id.messageArea);
    scrollView = (ScrollView) rootView.findViewById(R.id.scrollView);
    final ImageView microphone = (ImageView) rootView.findViewById(R.id.microphone);
    Firebase.setAndroidContext(getContext());
    reference1 = new Firebase("https://mychatapp-dcb19.firebaseio.com/messages/" + UserDetails.username + "_" + UserDetails.chatWith);
    reference2 = new Firebase("https://mychatapp-dcb19.firebaseio.com/messages/" + UserDetails.chatWith + "_" + UserDetails.username);
    microphone.setOnClickListener((v) -> {
        mMicClick.speakToText();
        //messageArea.setText(UserDetails.microinput, TextView.BufferType.EDITABLE);
    });
    sendButton.setOnClickListener((v) -> {
        String messageText = messageArea.getText().toString();
        messageArea.setText("");
        if(!messageText.equals("")){
            Map<String, String> map = new HashMap<>();
            map.put("message", messageText);
            map.put("user", UserDetails.username);
            reference1.push().setValue(map);
            reference2.push().setValue(map);
        }
    });

    reference1.addChildEventListener(new ChildEventListener() {
        @Override
        public void onChildAdded(DataSnapshot dataSnapshot, String s) {
            Map map = dataSnapshot.getValue(Map.class);
            String message = map.get("message").toString();
            String userName = map.get("user").toString();

            if(userName.equals(UserDetails.username)){
                addMessageBox("You:-\n" + message, 1);
            }
            else{
                addMessageBox(UserDetails.chatWith + ":-\n" + message, 2);
            }
        }
    });
}

```

voice to text:

```

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, final Bundle savedInstanceState) {
    View rootView = inflater.inflate(R.layout.activity_chat, container, false);
    layout = (LinearLayout) rootView.findViewById(R.id.layout1);
    sendButton = (ImageView) rootView.findViewById(R.id.sendButton);
    messageArea = (EditText) rootView.findViewById(R.id.messageArea);
    scrollView = (ScrollView) rootView.findViewById(R.id.scrollView);
    final ImageView microphone = (ImageView) rootView.findViewById(R.id.microphone);
    Firebase.setAndroidContext(getContext());
    reference1 = new Firebase("https://mychatapp-dcb19.firebaseio.com/messages/" + UserDetails.username + "_" + UserDetails.chatWith);
    reference2 = new Firebase("https://mychatapp-dcb19.firebaseio.com/messages/" + UserDetails.chatWith + "_" + UserDetails.username);
    microphone.setOnClickListener((v) -> {
        mMicClick.speakToText();
        //messageArea.setText(UserDetails.microinput, TextView.BufferType.EDITABLE);
    });
    sendButton.setOnClickListener((v) -> {
        String messageText = messageArea.getText().toString();
        messageArea.setText("");
        if(!messageText.equals("")){
            Map<String, String> map = new HashMap<>();
            map.put("message", messageText);
            map.put("user", UserDetails.username);
            reference1.push().setValue(map);
            reference2.push().setValue(map);
        }
    });

    reference1.addChildEventListener(new ChildEventListener() {
        @Override
        public void onChildAdded(DataSnapshot dataSnapshot, String s) {
            Map map = dataSnapshot.getValue(Map.class);
            String message = map.get("message").toString();
            String userName = map.get("user").toString();

            if(userName.equals(UserDetails.username)){
                addMessageBox("You:-\n" + message, 1);
            }
            else{
                addMessageBox(UserDetails.chatWith + ":-\n" + message, 2);
            }
        }
    });
}

```