

行動裝置期末作業

BI0717032 楊滄潔 四外語三

Weight Record App

0989039516

檢查表

基本要求 (沒有扣 30)

- 切 Activity ☐
- Fragment Navigation 切 fragment ☐
- RecyclerView ☐

功能要求 (沒有扣 10)

- 要用到某種權限 (敏感權限) ☐
- 結合 Firebase Auth ☐
- 要存取資料庫 ☐
- Safe argument 做資料的拋轉 ☐
- 會做通知 ☐

進階選項 (三選一即可) (沒有扣 5)

- 使用 implicit intent 呼叫系統元件 ☐
- 接收 broadcast ☐
- 使用系統的服務 ☐

品質要求

- 可以跑 (沒有扣 100) ☐
- 有 bug (看情節輕重扣分) ☐
- 旋轉畫面不會有生命週期問題 (沒有扣 50) ☐

額外加分項目

- 到 PreferenceFragment (20) ☐
- 功能很有創意且有意義 (10+) ☐
- 整合測試 (10+) ☐

繳交要求

文件 (沒有扣 100)

- 功能介紹 ☐
- 檢查表 ☐

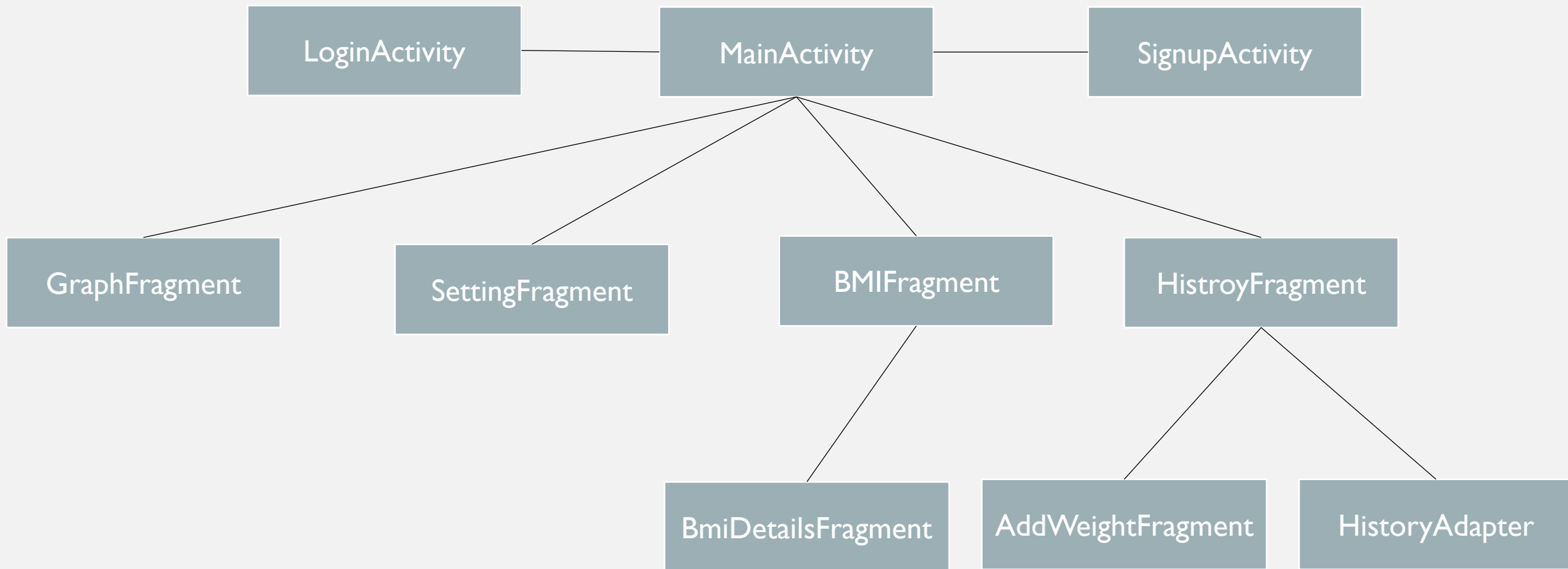
程式碼 (沒有扣 100)

☐
☐
☐

WEIGHT RECORD APP

目的：為了讓使用者紀錄自己的身高、體重和身型體態

架構圖



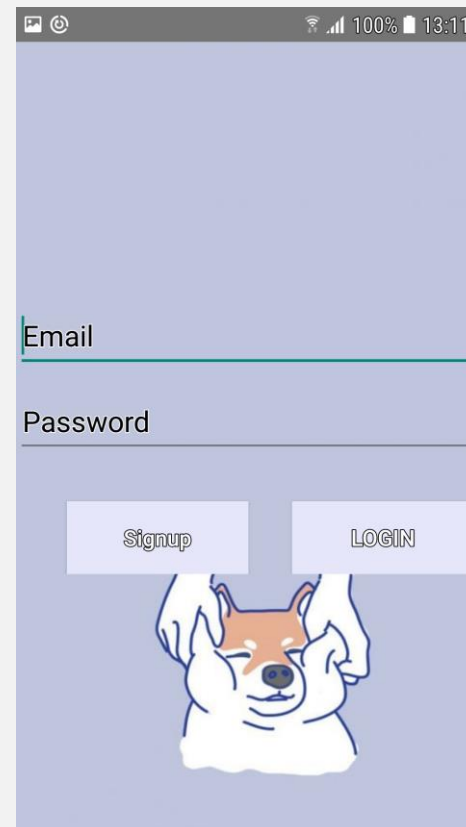
登入和註冊畫面

若已有註冊帳號了

直接按 “Go to login ”

利用 **Firebase** 用email登入

- LoginActivity
- Sign up Activity
- Main Activity



Mockup of a login screen with a light purple background. At the top, there is a status bar with icons for signal, Wi-Fi, 100% battery, and the time 13:11. Below the status bar, there are two input fields: "Email" and "Password", each with a teal underline. Under the "Email" field, there is a "Signup" button. Under the "Password" field, there is a "LOGIN" button. At the bottom of the screen, there is a cartoon illustration of a white dog with a brown patch on its face, looking up with its paws near its head.



Mockup of a registration screen with a light purple background. At the top, there is a status bar with icons for signal, Wi-Fi, 100% battery, and the time 13:11. Below the status bar, there are five input fields: "First Name", "Last Name", "Email", "Password", and "Phone", each with a teal underline. At the bottom of the screen, there are two buttons: "SIGNUP" and "Go to login". Below the buttons, there is a cartoon illustration of a white dog with a brown patch on its face, looking up with its paws near its head.

登入和註冊畫面

若已有註冊帳號了

直接按 “Go to login ”

利用 **Firestore** 用email登入

- LoginActivity
- Sign up Activity
- Main Activity

結合 **Firestore** Auth & 切Activity

```
progressBar.setVisibility(View.VISIBLE);

//authenticate user
auth.signInWithEmailAndPassword(email, password)
    .addOnCompleteListener( activity: LoginActivity.this, new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            // If sign in fails, display a message to the user. If sign in succeeds
            // the auth state listener will be notified and logic to handle the
            // signed in user can be handled in the listener.
            progressBar.setVisibility(View.GONE);
            if (!task.isSuccessful()) {
                Toast.makeText( context: LoginActivity.this, text: "Login Failed", Toast.LENGTH_LONG).show();
            } else {
                Intent intent = new Intent( packageContext: LoginActivity.this, MainActivity.class);
                startActivity(intent);
                finish();
            }
        }
    });
```

```
// [START initialize_auth]
// Initialize Firebase Auth
mAuth = FirebaseAuth.getInstance();
mainLabel = findViewById(R.id.tvHeaderName);
// [END initialize_auth]

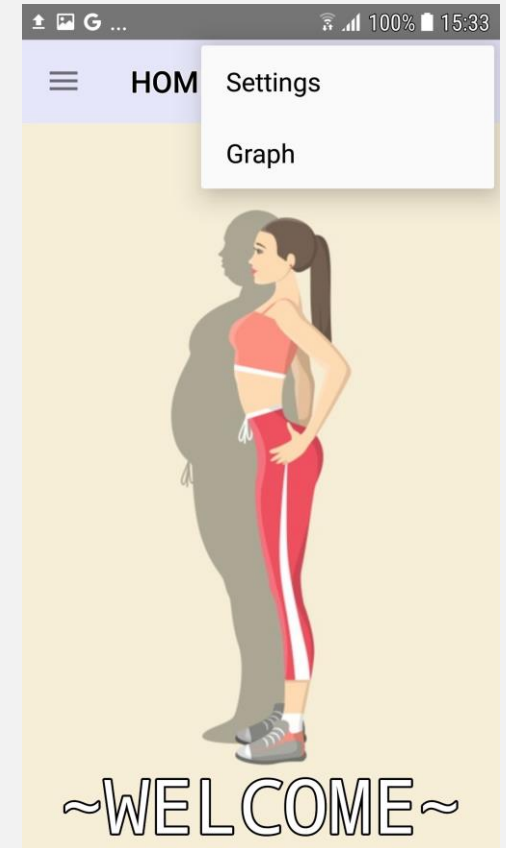
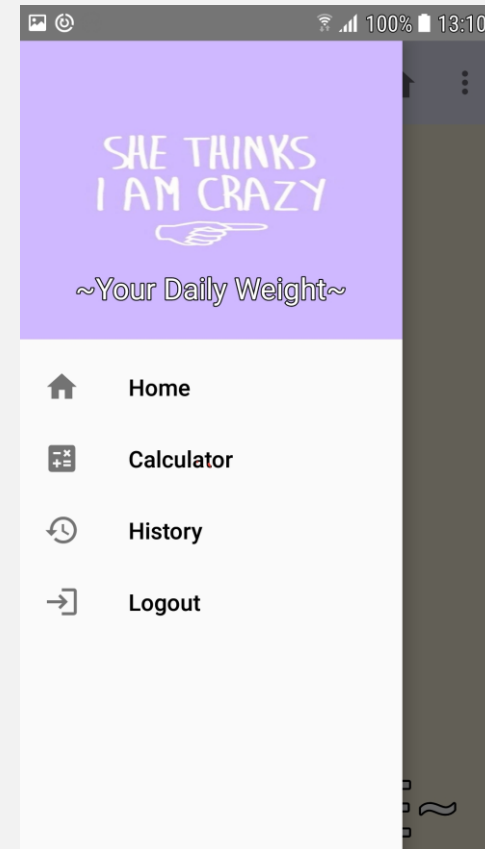
authListener = new FirebaseAuth.AuthStateListener() {
    @Override
    public void onAuthStateChanged(@NonNull FirebaseAuth firebaseAuth) {
        FirebaseUser user = firebaseAuth.getCurrentUser();
        if (user == null) {
            // user auth state is changed - user is null
            // Launch login activity
            startActivity(new Intent( packageContext: MainActivity.this, LoginActivity.class));
            finish();
        }
    }
};
```

HOME和MENU

利用Navigation Side Drawer menu

讓使用者選取功能，MenuBar讓使用者方便回到主畫面以及選取Setting和Graph。

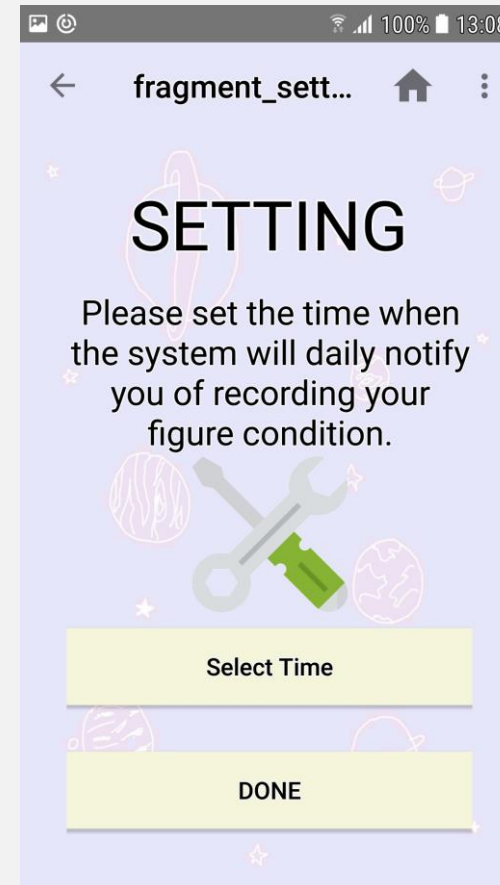
- Main Activity
- Main Fragment
- Setting Fragment
- Graph Fragment



SETTING 通知服務

- 讓使用者設置時間，並會定期通知紀錄體重。點擊通知則會自動進入APP。按下Done則會自動跳出App。
- Setting Fragment
- Alarm Broadcast
- BootUp Receiver

Preference設置



SETTING 通知服務

- 讓使用者設置時間，並會定期通知紀錄體重。點擊通知則會自動進入APP。按下Done則會自動跳出App。
- Setting Fragment
- Alarm Broadcast
- BootUp Receiver

接收 broadcast & 通知

```
Intent intent1 = new Intent(context, NotificationMessage.class);
intent1.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
intent1.putExtra( name: "message", date);
//Notification Builder
PendingIntent pendingIntent = PendingIntent.getActivity(context, requestCode: 1, intent1, PendingIntent.FLAG_ONE_SHOT);
NotificationManager notificationManager = (NotificationManager) context.getSystemService(Context.NOTIFICATION_SERVICE);
NotificationCompat.Builder mBuilder = new NotificationCompat.Builder(context, channelId: "notify_001");

NotificationCompat.Builder notificationBuilder = new NotificationCompat.Builder(context,WATER_REMINDER_NOTIFICATION_CHANNEL_ID)
    .setColor(ContextCompat.getColor(context, R.color.brower_actions_bg_grey))
    .setSmallIcon(R.drawable.ic_baseline_history_edu_24)
    .setLargeIcon(LargeIcon(context))
    .setContentTitle("Record Your Weight Everyday - Keep fit and Stay Healthy...")
    .setContentText("Our growing softness, our increasing lack of physical f...")
    .setStyle(new NotificationCompat.BigTextStyle().bigText(
        "Our growing softness, our increasing lack of physical f..."))
    .setDefaults(Notification.DEFAULT_VIBRATE)
    .setContentIntent(contentIntent(context))
    .setAutoCancel(true);

if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
    String channelId = "channel_id";
    NotificationChannel channel = new NotificationChannel(channelId, name: "channel name", NotificationManager.IMPORTANCE_HIGH);
    channel.enableVibration(true);
    notificationManager.createNotificationChannel(channel);
    mBuilder.setChannelId(channelId);
}

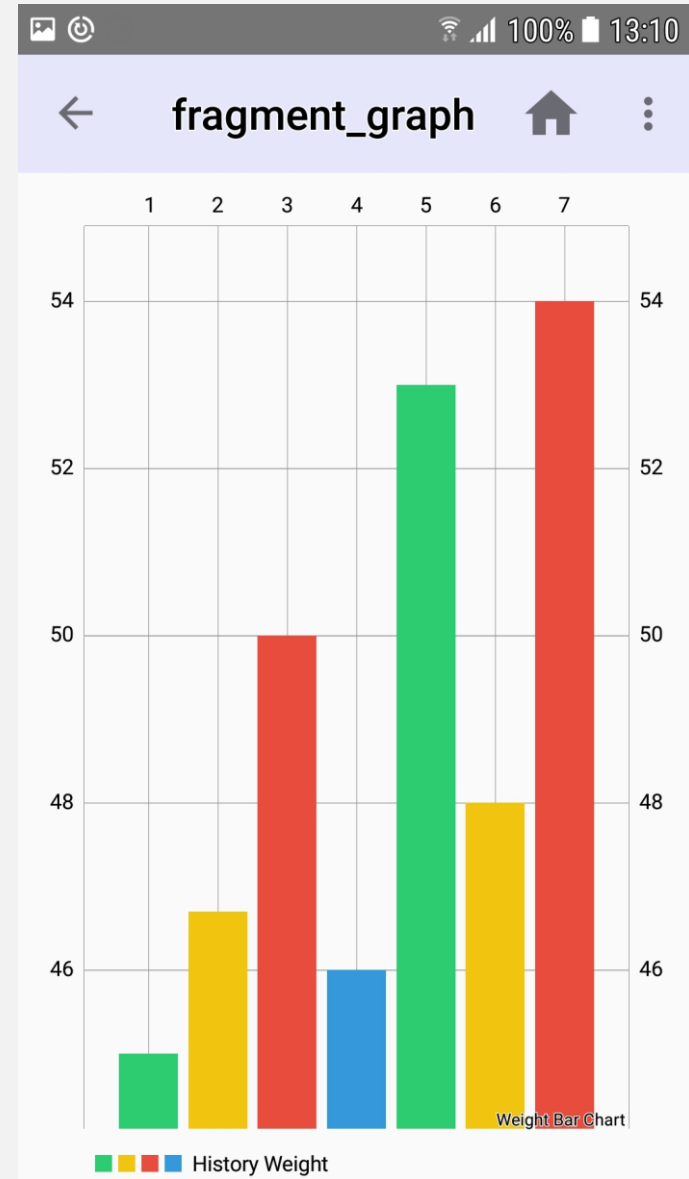
notificationManager.notify(WATER_REMINDER_NOTIFICATION_ID, notificationBuilder.build());
```

```
public class BootUpReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        Intent intent1 = new Intent(context, LoginActivity.class);
        intent1.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        context.startActivity(intent1);
    }
}
```

GRAPH體重圖表

Graph顯示使用者每天記錄體重的趨勢圖
barchart，利用MPAndroidchart呈現。

- Graph Fragment
- Database

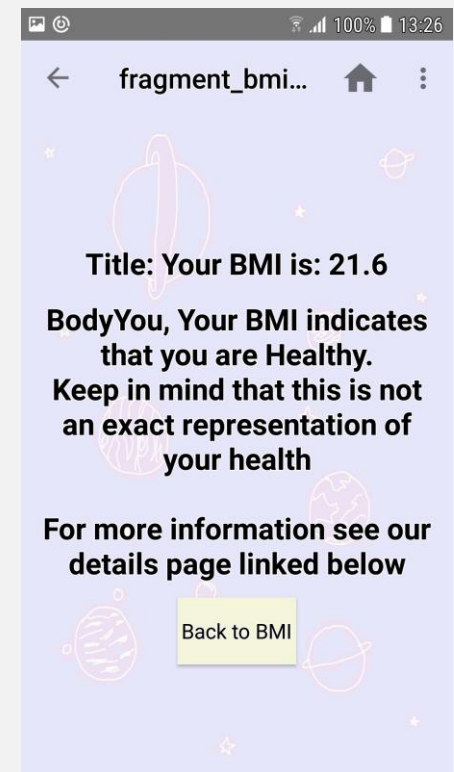
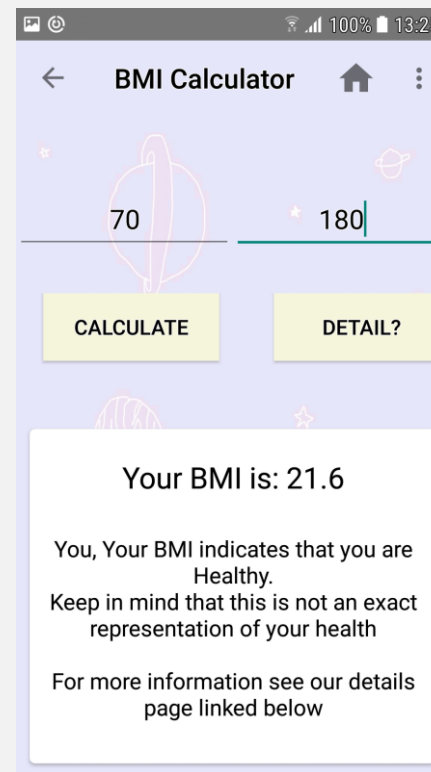


BMI計算和解說

使用者輸入身高體重，則會自動判斷BMI是否超標，點擊Detail則會顯示更多內容。

利用Safe argument 做資料的拋轉。

- BMI Fragment
- BMIDetail Fragment
- Calc



BMI計算和解說

使用者輸入身高體重，則會自動判斷BMI是否超標，點擊Detail則會顯示更多內容。

利用Safe argument 做資料的拋轉。

- BMIDetail Frag
- BMI Fragment

Safe argument 做資料的拋轉 Fragment Navigation 切 fragment

```
@Override
public void onClick(View v) { BMI(view); }
});

BmiDetailsbutton.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        NavController navController = Navigation.findNavController(view);

        BMIFragmentDirections.ActionBMIFragmentToBmiDetailsFragment action =
            actionBMIFragmentToBmiDetailsFragment( bmititle: "Title: " + titleText, bmibody: "Body"+bodyText);
        navController.navigate(action);
    }
});
```

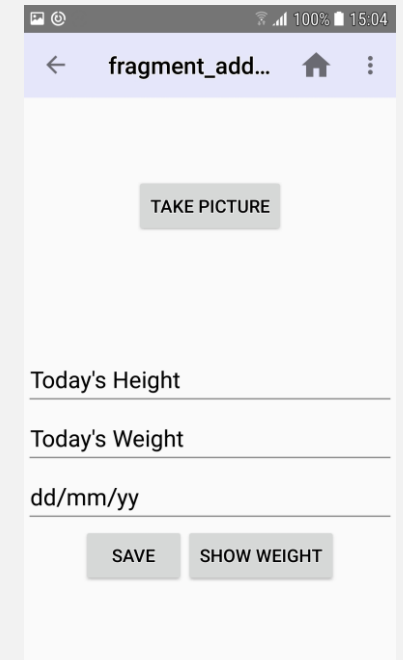
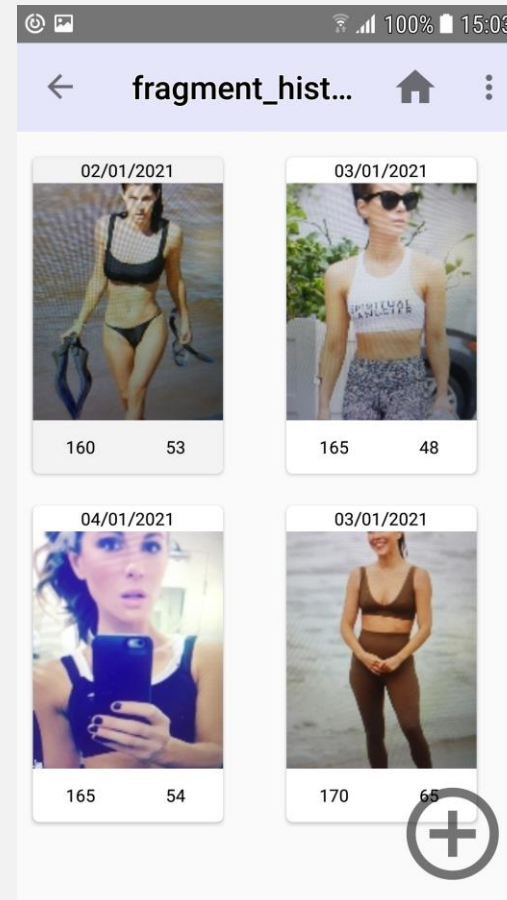
```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
    view= inflater.inflate(R.layout.fragment_bmi_details, container, attachToRoot: false);
    Toolbar toolbar=view.findViewById(R.id.toolbar);
    toolbar.setTitle("BMI Detail");

    todaytitle=view.findViewById(R.id.bmititle);
    todaybody=view.findViewById(R.id.bmibody);
    backbmi=view.findViewById(R.id.back);
    backbmi.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            NavController navController = Navigation.findNavController(view);
            navController.navigate(R.id.action_bmiDetailsFragment_to_BMIFragment);
        }
    });
    return view;
}
```

HISTORY & ADD WEIGHT

使用者進入History畫面會看到歷史體重和資料，點擊右下方按鈕，則會進入到新增體重的畫面，會先取得拍照權限，接下來輸入資料按下Save則會儲存資料成功。按下Show Weight則會回到History畫面。

- History Fragment
- Add Weight Fragment



HISTORY & ADD WEIGHT

使用者進入History畫面會看到歷史體重和資料，點擊右下方按鈕，則會進入到新增體重的畫面，會先取得拍照權限，接下來輸入資料按下Save則會儲存資料成功。按下Show Weight則會回到History畫面。

- History Fragment
- Add Weight Fragment

拍照權限

```
public void getPermission(){
    if(ActivityCompat.checkSelfPermission(getActivity(), Manifest.permission.CAMERA)
        != PackageManager.PERMISSION_GRANTED){
    }
    ActivityCompat.requestPermissions(getActivity(),new String[]{Manifest.permission.CAMERA}, requestCode: 1);
    new AlertDialog.Builder(getActivity())
        .setCancelable(false)
        .setTitle("需要相機權限")
        .setMessage("你是不是傻？不給我權限我要怎麼拍照？")
        .setPositiveButton( text: "知道了拉", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {
                ActivityCompat.requestPermissions(getActivity(),new String[]{Manifest.permission.CAMERA}, requestCode: 1);
            }
        })
        .show();

    if (ActivityCompat.shouldShowRequestPermissionRationale(getActivity(),
        Manifest.permission.CAMERA)) {

    }else{
        ActivityCompat.requestPermissions(getActivity(),new String[]{Manifest.permission.CAMERA}, requestCode: 1);
    }
}
```

```
// Camera exists? Then proceed...
Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);

// Ensure that there's a camera activity to handle the intent
MainActivity activity = (MainActivity)getActivity();
if (takePictureIntent.resolveActivity(activity.getPackageManager()) != null) {

    try {

        startActivityForResult(takePictureIntent, CAMERA_INTENT);

    } catch (ActivityNotFoundException e) {
        // display error state to the user
    }

}

//
}

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == CAMERA_INTENT) {
        if (resultCode == RESULT_OK) {
            bmpImage = (Bitmap) data.getExtras().get("data");
            imageView.setImageBitmap(bmpImage);
        } else if (resultCode == RESULT_CANCELED) {
            Toast.makeText(getActivity(), text: "Cancelled", Toast.LENGTH_LONG).show();
        }
    }
}
```

HISTORY & ADD WEIGHT

使用者進入History畫面會看到歷史體重和資料，點擊右下方按鈕，則會進入到新增體重的畫面，會先取得拍照權限，接下來輸入資料按下Save則會儲存資料成功。按下Show Weight則會回到History畫面。

- History Fragment
- Add Weight Fragment

RecyclerView

```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                          Bundle savedInstanceState) {
    view=inflater.inflate(R.layout.fragment_histroy, container, attachToRoot: false);
    ToAddWeightFragment=(ImageView)view.findViewById(R.id.toaddweight);
    mRecyclerView=view.findViewById(R.id.recyclerViewWeights);
    mRecyclerView.setLayoutManager(new LinearLayoutManager(getActivity()));
    mHistoryAdapter=new HistoryAdapter(getActivity(), mItemClickListener: this);
    LinearLayoutManager layoutManager = new LinearLayoutManager(getActivity());
    mRecyclerView.setLayoutManager(new GridLayoutManager(getActivity(), spanCount: 2));
    mRecyclerView.setAdapter(mHistoryAdapter);
    mDb=AppDatabase.getInstance(getActivity()).getApplicationContext();
}
```

```
@NonNull
@Override
public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    View view;
    LayoutInflater mInflater=LayoutInflater.from(mContext);
    view=mInflater.inflate(R.layout.weight_list_layout,parent, attachToRoot: false);
    return new ViewHolder(view);
}

@Override
public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
    Weight weight=dailyWeight.get(position);

    holder.daily_date.setText(dateFormat.format(weight.getDate()));
    holder.daily_figure_picture.setImageBitmap(DataConverter.convertByteArray2Image(weight.getImage()));
    holder.daily_height.setText(weight.getHeight());
    holder.daily_weight.setText(weight.getWeight());
}
```

HISTORY & ADD WEIGHT

使用者進入History畫面會看到歷史體重和資料，點擊右下方按鈕，則會進入到新增體重的畫面，會先取得拍照權限，接下來輸入資料按下Save則會儲存資料成功。按下Show Weight則會回到History畫面。

- History Fragment
- Add Weight Fragment

Room資料庫應用

