Alarm Clock with Notifications in Android Studio using Java – Alarm Manager Easy 4 Steps



Table of Contents

* [Download Background Image:](https://androidknowledge.com/alarm-manager-notifications-android-studio/#download-background-image)
* [Output](https://androidknowledge.com/alarm-manager-notifications-android-studio/#output)
* [AK Bonus Points:](https://androidknowledge.com/alarm-manager-notifications-android-studio/#ak-bonus-points)

Alarm Manager

The Alarm Manager sets the system alarm as per your scheduled time. Even, if you are not currently running the application it will still throw a notification as per the scheduled time.

In this project, we have created an alarm clock where you can select a time as per your choice then click on set alarm and it will be set. We also provide a cancel alarm button.

Once you have selected the time and clicked on set alarm then a notification will appear on that particular time. To keep the project simple, we have not provided a user input notification so it will default as per the code. In further upcoming projects, I will create a fully functional alarm clock.

Step-by-Step Implementation

**Step 1: Open Android Studio, Click on New Project and Choose Empty Activity.**

**Step 2: Pre-requisites**:

**Gradle (module)**

buildFeatures{

viewBinding **true**

}

**colors.xml**

<**color** name="lavender">#8692f7</**color**>

**themes.xml**

<**resources** xmlns:tools="http://schemas.android.com/tools">

<!-- Base application theme. -->

<**style** name="Theme.AlarmClock" parent="Theme.MaterialComponents.DayNight.DarkActionBar">

<!-- Primary brand color. -->

<**item** name="colorPrimary">@color/lavender</**item**>

<**item** name="colorPrimaryVariant">@color/lavender</**item**>

<**item** name="colorOnPrimary">@color/white</**item**>

<!-- Secondary brand color. -->

<**item** name="colorSecondary">@color/teal\_200</**item**>

<**item** name="colorSecondaryVariant">@color/teal\_700</**item**>

<**item** name="colorOnSecondary">@color/black</**item**>

<!-- Status bar color. -->

<**item** name="android:statusBarColor">?attr/colorPrimaryVariant</**item**>

<!-- Customize your theme here. -->

</**style**>

</**resources**>

**Drawables**

Add vector assets – notifications icon.

**AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools">

<application

android:allowBackup="true"

android:dataExtractionRules="@xml/data\_extraction\_rules"

android:fullBackupContent="@xml/backup\_rules"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:roundIcon="@mipmap/ic\_launcher\_round"

android:supportsRtl="true"

android:theme="@style/Theme.AlarmClock"

tools:targetApi="31">

<activity

android:name=".NotificationActivity"

android:exported="false">

<meta-data

android:name="android.app.lib\_name"

android:value="" />

</activity>

<activity

android:name=".MainActivity"

android:exported="true">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

<meta-data

android:name="android.app.lib\_name"

android:value="" />

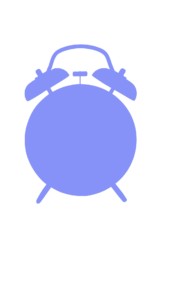
</activity>

<receiver android:name=".AlarmReceiver"/>

</application>

</manifest>

Download Background Image:



[Download](http://androidknowledge.com/wp-content/uploads/2022/12/alarm.png)

**Step 3: XML Layouts**:

**activity\_main.xml**

**<?xml version="1.0" encoding="utf-8"?>**

<**LinearLayout**

xmlns:android="http://schemas.android.com/apk/res/android"

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:orientation="vertical"

android:gravity="center\_vertical"

android:background="@drawable/alarm"

tools:context=".MainActivity">

<**TextView**

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Select \nTime"

android:id="@+id/selectTime"

android:textSize="40sp"

android:layout\_marginTop="280dp"

android:layout\_gravity="center\_horizontal"

android:textColor="@color/white"

android:textAlignment="center"/>

<**RelativeLayout**

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<**Button**

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:id="@+id/setAlarm"

android:text="Set \nAlarm"

android:textSize="20sp"

android:padding="8dp"

android:layout\_marginStart="40dp"

android:layout\_marginTop="250dp"/>

<**Button**

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:id="@+id/cancelAlarm"

android:text="Cancel \nAlarm"

android:textSize="20sp"

android:padding="8dp"

android:layout\_alignParentEnd="true"

android:layout\_marginTop="250dp"

android:layout\_marginEnd="40dp"/>

</**RelativeLayout**>

</**LinearLayout**>

**activity\_notifications.xml**

**<?xml version="1.0" encoding="utf-8"?>**

<**androidx.constraintlayout.widget.ConstraintLayout**

xmlns:android="http://schemas.android.com/apk/res/android"

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"

tools:context=".NotificationActivity">

<**TextView**

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Welcome to \nNotification Activity"

android:textSize="36sp"

android:textAlignment="center"

android:textColor="@color/lavender"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"/>

</**androidx.constraintlayout.widget.ConstraintLayout**>

**Step 4: Java Files**

**AlarmReceiver.java**

**package** *com.example.alarmclock*;

**import** *android.app.PendingIntent*;

**import** *android.content.BroadcastReceiver*;

**import** *android.content.Context*;

**import** *android.content.Intent*;

**import** *androidx.core.app.NotificationCompat*;

**import** *androidx.core.app.NotificationManagerCompat*;

**public** **class** AlarmReceiver **extends** BroadcastReceiver {

@Override

**public** **void** onReceive(Context context, Intent intent) {

Intent nextActivity = new Intent(context, NotificationActivity.class);

intent.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK | Intent.FLAG\_ACTIVITY\_CLEAR\_TASK);

PendingIntent pendingIntent = PendingIntent.getActivity(context, 0, nextActivity, 0);

NotificationCompat.Builder builder = new NotificationCompat.Builder(context, "androidknowledge")

.setSmallIcon(R.drawable.ic\_baseline\_notifications\_24)

.setContentTitle("Reminder")

.setContentText("It's time to wake up")

.setAutoCancel(**true**)

.setDefaults(NotificationCompat.DEFAULT\_ALL)

.setPriority(NotificationCompat.PRIORITY\_HIGH)

.setContentIntent(pendingIntent);

NotificationManagerCompat notificationManagerCompat = NotificationManagerCompat.from(context);

notificationManagerCompat.notify(123,builder.build());

}

}

**MainActivity.java**

**package** *com.example.alarmclock*;

**import** *androidx.appcompat.app.AppCompatActivity*;

**import** *android.app.AlarmManager*;

**import** *android.app.NotificationChannel*;

**import** *android.app.NotificationManager*;

**import** *android.app.PendingIntent*;

**import** *android.content.Context*;

**import** *android.content.Intent*;

**import** *android.os.Build*;

**import** *android.os.Bundle*;

**import** *android.view.View*;

**import** *android.widget.Toast*;

**import** *com.example.alarmclock.databinding.ActivityMainBinding*;

**import** *com.google.android.material.timepicker.MaterialTimePicker*;

**import** *com.google.android.material.timepicker.TimeFormat*;

**import** *java.util.Calendar*;

**public** **class** MainActivity **extends** AppCompatActivity {

**private** ActivityMainBinding binding;

**private** MaterialTimePicker timePicker;

**private** Calendar calendar;

**private** AlarmManager alarmManager;

**private** PendingIntent pendingIntent;

@Override

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

**super**.onCreate(savedInstanceState);

binding = ActivityMainBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

createNotificationChannel();

binding.selectTime.setOnClickListener(new View.OnClickListener() {

@Override

**public** **void** onClick(View view) {

timePicker = new MaterialTimePicker.Builder()

.setTimeFormat(TimeFormat.CLOCK\_12H)

.setHour(12)

.setMinute(0)

.setTitleText("Select Alarm Time")

.build();

timePicker.show(getSupportFragmentManager(), "androidknowledge");

timePicker.addOnPositiveButtonClickListener(new View.OnClickListener() {

@Override

**public** **void** onClick(View view) {

**if** (timePicker.getHour() > 12){

binding.selectTime.setText(

**String**.format("%02d",(timePicker.getHour()-12)) +":"+ **String**.format("%02d", timePicker.getMinute())+"PM"

);

} **else** {

binding.selectTime.setText(timePicker.getHour()+":" + timePicker.getMinute()+ "AM");

}

calendar = Calendar.getInstance();

calendar.set(Calendar.HOUR\_OF\_DAY, timePicker.getHour());

calendar.set(Calendar.MINUTE, timePicker.getMinute());

calendar.set(Calendar.SECOND, 0);

calendar.set(Calendar.MILLISECOND, 0);

}

});

}

});

binding.setAlarm.setOnClickListener(new View.OnClickListener() {

@Override

**public** **void** onClick(View view) {

alarmManager = (AlarmManager) getSystemService(Context.ALARM\_SERVICE);

Intent intent = new Intent(MainActivity.this, AlarmReceiver.class);

pendingIntent = PendingIntent.getBroadcast(MainActivity.this, 0, intent, 0);

alarmManager.setInexactRepeating(AlarmManager.RTC\_WAKEUP, calendar.getTimeInMillis(), AlarmManager.INTERVAL\_DAY, pendingIntent);

Toast.makeText(MainActivity.this, "Alarm Set", Toast.LENGTH\_SHORT).show();

}

});

binding.cancelAlarm.setOnClickListener(new View.OnClickListener() {

@Override

**public** **void** onClick(View view) {

Intent intent = new Intent(MainActivity.this, AlarmReceiver.class);

pendingIntent = PendingIntent.getBroadcast(MainActivity.this, 0, intent, 0);

**if** (alarmManager == **null**){

alarmManager = (AlarmManager) getSystemService(Context.ALARM\_SERVICE);

}

alarmManager.cancel(pendingIntent);

Toast.makeText(MainActivity.this, "Alarm Canceled", Toast.LENGTH\_SHORT).show();

}

});

}

**private** **void** createNotificationChannel(){

**if** (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.O){

CharSequence name = "akchannel";

**String** desc = "Channel for Alarm Manager";

**int** imp = NotificationManager.IMPORTANCE\_HIGH;

NotificationChannel channel = new NotificationChannel("androidknowledge", name, imp);

channel.setDescription(desc);

NotificationManager notificationManager = getSystemService(NotificationManager.class);

notificationManager.createNotificationChannel(channel);

}

}

}

Output

