

A Sleep Tracking App for A Better Night's Rest

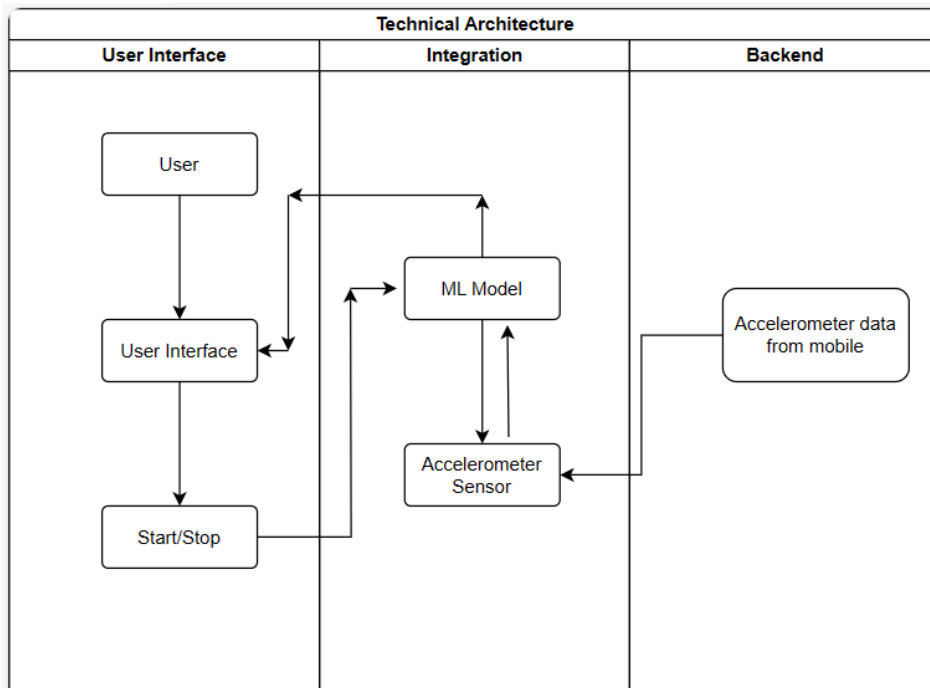
Introduction

Sleep tracking apps can be a helpful tool for improving your sleep quality. By tracking your sleep patterns over time, you can identify trends and patterns. This information can help you make changes to your sleep habits, such as going to bed and waking up at the same time each day, creating a relaxing bedtime routine, and avoiding caffeine and alcohol before bed.

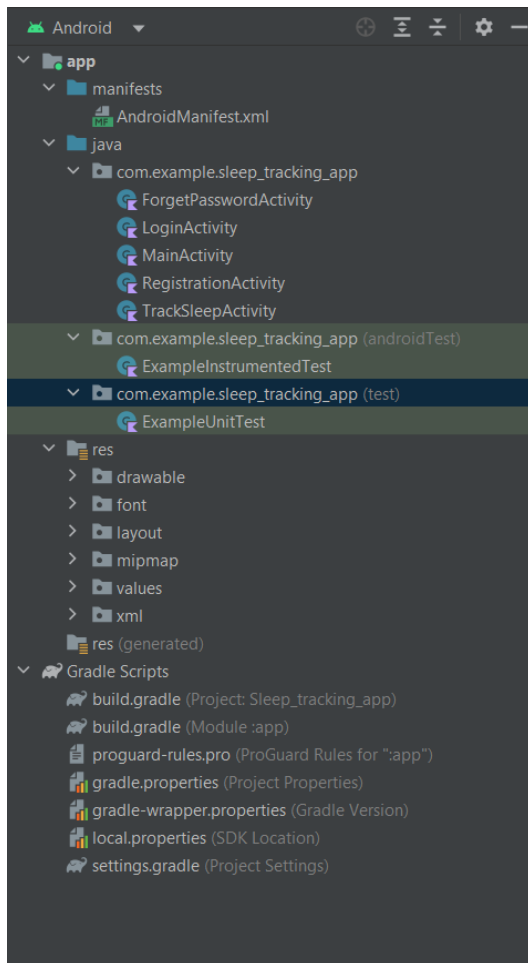
To use a sleep tracking app, simply place your smartphone near your bed while you sleep. The app will use the phone's sensors to track your movements and other physiological data. In the morning, you can open the app to view your sleep data and insights.

Overall, sleep tracking apps can be a valuable tool for improving your sleep quality. If you're interested in trying a sleep tracking app, there are many different options available. Some apps are free, while others require a subscription. Be sure to read reviews and compare features before choosing an app.

Technical Architecture



Project Structure



Dependencies

```
import android.content.Intent
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
```

```
import android.content.Intent
import android.os.Bundle
import android.os.Handler
import android.os.Looper
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import com.google.firebase.FirebaseApp
import java.text.SimpleDateFormat
import java.util.Date
import java.util.Locale
```

```
import android.os.Bundle
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import java.text.SimpleDateFormat
import java.util.Date
import java.util.Locale
```

Manifest

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      package="com.example.sleep_tracking_app">
5
6      <application
7          android:allowBackup="true"
8          android:dataExtractionRules="@xml/data_extraction_rules"
9          android:fullBackupContent="@xml/backup_rules"
10         android:icon="@mipmap/ic_launcher"
11         android:label="@string/app_name"
12         android:roundIcon="@mipmap/ic_launcher_round"
13         android:supportRtl="true"
14         android:theme="@style/Theme.Sleep_tracking_app"
15         tools:targetApi="31">
16
17         <activity
18             android:name=".ForgetPasswordActivity"
19             android:exported="false" />
20
21         <activity
22             android:name=".TrackSleepActivity"
23             android:exported="false" />
24
25         <activity
26             android:name=".LoginActivity"
27             android:exported="true">
28             <intent-filter>
29                 <action android:name="android.intent.action.MAIN" />
30
31                 <category android:name="android.intent.category.LAUNCHER" />
32             </intent-filter>
33         </activity>
34
35         <activity
36             android:name=".RegistrationActivity"
37             android:exported="false" />
38
39         <activity
40             android:name=".MainActivity"
41             android:exported="false" />
42
43         <meta-data
44             android:name="preloaded_fonts"
45             android:resource="@array/preloaded_fonts" />
46     </application>
47
48 </manifest>
49
```

Fonts XML

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3      <array name="com_google_android_gms_fonts_certs">
4          <item>@array/com_google_android_gms_fonts_certs_dev</item>
5          <item>@array/com_google_android_gms_fonts_certs_prod</item>
6      </array>
7      <string-array name="com_google_android_gms_fonts_certs_dev">
8          <item>
9              MIIEQDCCA5CgAwIBAgIJANWFu6x90071MA0GCSqGSIb3DQEBAUAMIHQMswCQYDVQQGEwJVVzETMBEGA1UECBMKQ2FsaWZvcml5YTEwHBCGA1UEBxMMMTU
10             </item>
11          </string-array>
12      <string-array name="com_google_android_gms_fonts_certs_prod">
13          <item>
14              MIIQzCCAyugAwIBAgIJAMLgh0ZkSjCNMA0GCSqGSIb3DQEBAUAMIHQMswCzAJBgNVBAYTA1VTMRMwEQYDVQQIEwpDYWxpZm9ybmlhHRRYWFAYDVQQHEw1Nb3
15             </item>
16          </string-array>
17      </resources>
18

```

Project UI

Login Page

```

1  package com.example.sleep_tracking_app
2
3  import android.content.Intent
4  import android.os.Bundle
5  import android.widget.Button
6  import android.widget.EditText
7  import android.widget.TextView
8  import android.widget.Toast
9  import androidx.appcompat.app.AppCompatActivity
10
11  class LoginActivity : AppCompatActivity() {
12
13      private lateinit var usernameEditText: EditText
14      private lateinit var passwordEditText: EditText
15      private lateinit var signInButton: Button
16      private lateinit var forgetPasswordTextView: TextView
17
18      override fun onCreate(savedInstanceState: Bundle?) {
19          super.onCreate(savedInstanceState)
20          setContentView(R.layout.activity_login)
21
22          usernameEditText = findViewById(R.id.usernameEditText)
23          passwordEditText = findViewById(R.id.passwordEditText)
24          signInButton = findViewById(R.id.signInButton)
25          forgetPasswordTextView = findViewById(R.id.forgetPasswordTextView)
26
27          signInButton.setOnClickListener { it: View!
28              val username = usernameEditText.text.toString()
29              val password = passwordEditText.text.toString()
30

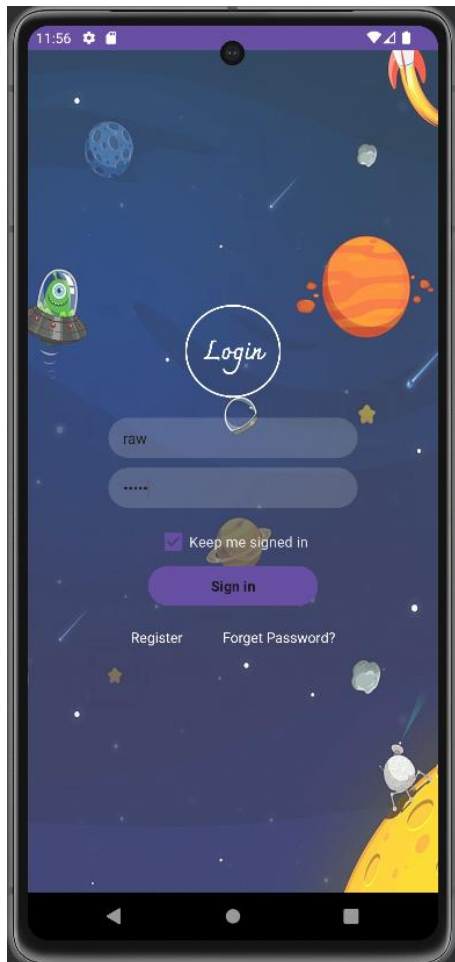
```

```

30
31
32     if (username.isNotEmpty() && password.isNotEmpty()) {
33         // Valid username and password, open RegistrationActivity
34         val intent = Intent( packageContext: this, RegistrationActivity::class.java)
35         startActivity(intent)
36     } else {
37         // Invalid username or password
38         Toast.makeText(baseContext, text: "Could not sign in. Please try again.", Toast.LENGTH_SHORT).show()
39     }
40
41     forgetPasswordTextView.setOnClickListener { it: View?
42         val intent = Intent( packageContext: this, ForgetPasswordActivity::class.java)
43         startActivity(intent)
44     }
45 }
46
47

```

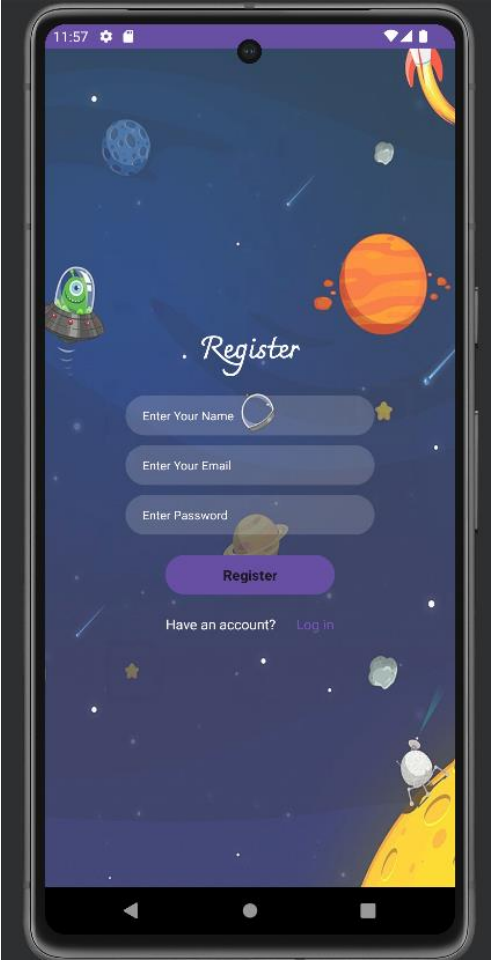
Result



Registration Page

```
1 package com.example.sleep_tracking_app
2
3 import ...
4
5
6
7
8
9
10 class RegistrationActivity : AppCompatActivity() {
11
12     private lateinit var nameEditText: EditText
13     private lateinit var emailEditText: EditText
14     private lateinit var passwordEditText: EditText
15     private lateinit var registerButton: Button
16
17     override fun onCreate(savedInstanceState: Bundle?) {
18         super.onCreate(savedInstanceState)
19         setContentView(R.layout.activity_registration)
20
21         nameEditText = findViewById(R.id.nameEditText)
22         emailEditText = findViewById(R.id.emailEditText)
23         passwordEditText = findViewById(R.id.passwordEditText)
24         registerButton = findViewById(R.id.registerButton)
25
26         registerButton.setOnClickListener { it: View?
27             val name = nameEditText.text.toString()
28             val email = emailEditText.text.toString()
29             val password = passwordEditText.text.toString()
30
31             if (name.isNotEmpty() && email.isNotEmpty() && password.isNotEmpty()) {
32                 // Valid input, open MainActivity
33                 val intent = Intent( packageContext: this, MainActivity::class.java)
34                 startActivity(intent)
35                 finish()
36
37             if (name.isNotEmpty() && email.isNotEmpty() && password.isNotEmpty()) {
38                 // Valid input, open MainActivity
39                 val intent = Intent( packageContext: this, MainActivity::class.java)
40                 startActivity(intent)
41                 finish()
42             } else {
43                 // Invalid input
44                 Toast.makeText(baseContext, "Please fill in all fields.", Toast.LENGTH_SHORT).show()
45             }
46         }
47     }
48 }
```

Result

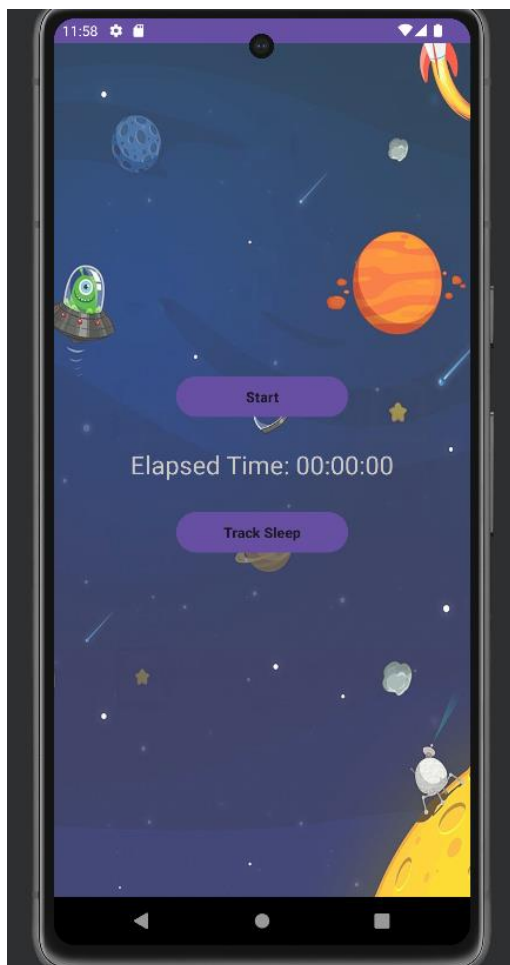


Dashboard

```
1 package com.example.sleep_tracking_app
2
3 import ...
4
14
15 class MainActivity : AppCompatActivity() {
16
17     private lateinit var startSleepTrackingButton: Button
18     private lateinit var elapsedTimeTextView: TextView
19     private lateinit var trackSleepButton: Button
20
21     private val handler = Handler(Looper.getMainLooper())
22     private var startTimeMillis = 0L
23
24     override fun onCreate(savedInstanceState: Bundle?) {
25         super.onCreate(savedInstanceState)
26
27         FirebaseApp.initializeApp(context: this)
28
29         setContentView(R.layout.activity_main)
30
31         startSleepTrackingButton = findViewById(R.id.startSleepTrackingButton)
32         elapsedTimeTextView = findViewById(R.id.elapsedTimeTextView)
33         trackSleepButton = findViewById(R.id.trackSleepButton)
34
35         startSleepTrackingButton.setOnClickListener { it: View!
36             startTimeMillis = System.currentTimeMillis()
37             handler.postDelayed(updateElapsedTime, delayMillis: 1000)
38         }
39
40         trackSleepButton.setOnClickListener { it: View!
41             val intent = Intent( packageContext: this, TrackSleepActivity::class.java)
42
43             // Set start and end times as extras
44             val currentTimeMillis = System.currentTimeMillis()
45             val endTimeMillis = currentTimeMillis + (8 * 60 * 60 * 1000) // Assuming 8 hours of sleep
46
47             intent.putExtra( name: "startTime", currentTimeMillis)
48             intent.putExtra( name: "endTime", endTimeMillis)
49
50             startActivity(intent)
51         }
52     }
53
54     private val updateElapsedTime = object : Runnable {
55         override fun run() {
56             val currentTimeMillis = System.currentTimeMillis()
57             val elapsedTimeMillis = currentTimeMillis - startTimeMillis
58
59             val hours = (elapsedTimeMillis / (1000 * 60 * 60)) % 24
60             val minutes = (elapsedTimeMillis / (1000 * 60)) % 60
61             val seconds = (elapsedTimeMillis / 1000) % 60
62
63             val formattedTime = String.format("%02d:%02d:%02d", hours, minutes, seconds)
64
65             elapsedTimeTextView.text = "Elapsed Time: $formattedTime"
66
67             handler.postDelayed( this, delayMillis: 1000)
68         }
69     }
70 }
```

```
71  override fun onDestroy() {  
72      super.onDestroy()  
73      handler.removeCallbacks(updateElapsedTime)  
74  }  
75  }  
76
```

Result



Sleep Track

```
1 package com.example.sleep_tracking_app
2
3 import ...
4
5
6
7
8
9
10 class TrackSleepActivity : AppCompatActivity() {
11
12     private lateinit var startTimeTextView: TextView
13     private lateinit var endTimeTextView: TextView
14
15     override fun onCreate(savedInstanceState: Bundle?) {
16         super.onCreate(savedInstanceState)
17         setContentView(R.layout.activity_track_sleep)
18
19         startTimeTextView = findViewById(R.id.startTimeTextView)
20         endTimeTextView = findViewById(R.id.endTimeTextView)
21
22         val startTimeMillis = intent.getLongExtra( name: "startTime", defaultValue: 0)
23         val endTimeMillis = intent.getLongExtra( name: "endTime", defaultValue: 0)
24
25         val startTime = formatTime(startTimeMillis)
26         val endTime = formatTime(endTimeMillis)
27
28         startTimeTextView.text = "Start Time: $startTime"
29         endTimeTextView.text = "End Time: $endTime"
30     }
31
32     private fun formatTime(timeMillis: Long): String {
33         val dateFormat = SimpleDateFormat( pattern: "dd-MM-yyyy", Locale.getDefault())
34         val timeFormat = SimpleDateFormat( pattern: "HH:mm:ss", Locale.getDefault())
35
36         val date = Date(timeMillis)
37         val formattedDate = dateFormat.format(date)
38         val formattedTime = timeFormat.format(date)
39
40         return "$formattedDate $formattedTime"
41     }
42 }
43
```

Result



Summary

Sleep tracking apps can be a valuable tool for improving your sleep quality. By tracking your sleep patterns over time, you can identify trends and patterns. This information can help you make changes to your sleep habits, such as going to bed and waking up at the same time each day, creating a relaxing bedtime routine, and avoiding caffeine and alcohol before bed.

Sleep tracking apps can also help you identify and address potential sleep problems, such as insomnia, sleep apnea, and restless leg syndrome. This information can be helpful for discussing your sleep concerns with your doctor.

Overall, sleep tracking apps can be a helpful tool for improving your sleep quality and overall health and well-being.

Here are some tips for using a sleep tracking app effectively:

- Place your smartphone near your bed while you sleep.
- Use the app to set sleep goals.
- Track your sleep data over time.
- Identify trends and patterns in your sleep data.
- Make changes to your sleep habits based on your sleep data.
- Discuss your sleep data with your doctor if you have any concerns.