ANNAMALAI UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY
Department of Computer Science & Engineering

MOBILE APP DEVELOPMENT
ASSIGNMENT -1

Routine

Your handy task manager for organizing & prioritizing your to-do list with ease.

- BARATHRAJ B (LE05) — Designer
- DEEPATH KUMAR S (LE10) — Documentation
- MOHAMED AKRAM M N (LE11) — Programmer
- PRADEEP G (LE19) — Programmer
- SABARIVASAN S (LE08) — Documentation
- SANJEEVIKUMAR (LE06) — Designer

Submitted to
Dr. K. T. Meena Abarna

Project ROUTINE:

Crafting a Productive To-Do Application with User-Centric Features

Overview:

This documentation will provide an in-depth exploration of the development process, from conceptualization to implementation. It will delve into the core features of "ROUTINE," including the ability to add tasks, set due dates, assign priorities, and mark tasks as completed. Throughout this journey, we will emphasize a user-centric approach, ensuring that every feature aligns with the needs and preferences of our target audience.

Teck Stack:

- **Development Environment:**
  - ❖ Android Studio: The primary integrated development environment (IDE) for Android app development.
- **Programming Language:**
  - ❖ Java: The core programming language used for developing the Android app.
- **Database Management:**
  - ❖ Room Persistence Library: A part of the Android Architecture Components, Room is used for local data persistence, making it easier to work with SQLite databases in Android.
- **User Interface (UI) Design:**
  - ❖ Android XML Layouts: For defining the app's user interface elements.
- **Version Control:**
  - ❖ Git: To manage version control and track changes in the project.
- **Documentation:**
  - ❖ Microsoft Word: For creating detailed documentation and reports for the assignment.
- **Testing:**
  - ❖ Android Emulator: For testing the app on different virtual Android devices.
- **Operating System:**
  - ❖ Windows and Linux

System Requirement:

- **Operating System:**
  - ❖ Android OS: The app is designed to run on Android devices, so it requires an Android operating system.
- **Minimum Android Version:**
  - ❖ Android 7.0 (Nougat): The app is compatible with Android devices running version 7.0 and above.
- **Device Screen Size:**
  - ❖ A minimum screen size of 4.5 inches is recommended for a comfortable user experience.
- **Storage Space:**
  - ❖ The app requires a minimum of 20MB of free storage space on the device for installation.
- **RAM (Memory):**
  - ❖ A device with at least 1GB of RAM is recommended for smooth app performance.
- **Internet Connectivity:**
  - ❖ The app can function without internet connection.
- **Permissions:**
  - ❖ The app requires permissions for accessing device storage to save data and reminders.

Source code:

The source code of the project is available on github (*https://github.com/Dark-WizZ/Routine*).

MainActivity.java:

```
package com.darkwizz.routine;

import android.os.Bundle;

import com.darkwizz.routine.adaptor.OnTodoClickListener;
import com.darkwizz.routine.adaptor.RecyclerViewAdaptor;
import com.darkwizz.routine.model.SharedViewModel;
import com.darkwizz.routine.model.Task;
import com.darkwizz.routine.model.TaskViewModel;
import com.google.android.material.bottomsheet.BottomSheetBehavior;
```

```
import
com.google.android.material.floatingactionbutton.FloatingActionButton;

import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.constraintlayout.widget.ConstraintLayout;
import androidx.lifecycle.ViewModelProvider;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.view.Menu;
import android.view.MenuItem;

public class MainActivity extends AppCompatActivity implements
OnTodoClickListener {
   private TaskViewModel taskViewModel;
   private static final String TAG = "ITEM";
   private RecyclerView recyclerView;
   private RecyclerViewAdaptor recyclerViewAdaptor;
   private int counter;
   BottomSheetFragment bottomSheetFragment;
   private SharedViewModel sharedViewModel;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      Toolbar toolbar = findViewById(R.id.toolbar);
      setSupportActionBar(toolbar);
      counter=0;

      bottomSheetFragment = new BottomSheetFragment();
      ConstraintLayout constraintLayout = findViewById(R.id.bottomSheet);
      BottomSheetBehavior<ConstraintLayout> bottomSheetBehavior =
BottomSheetBehavior.from(constraintLayout);
      bottomSheetBehavior.setPeekHeight(BottomSheetBehavior.STATE_HIDDE
N);

      recyclerView = findViewById(R.id.recycler_view);
      recyclerView.setHasFixedSize(true);
```

```
        recyclerView.setLayoutManager(new LinearLayoutManager(this));

        taskViewModel = new ViewModelProvider.AndroidViewModelFactory(
                MainActivity.this.getApplication())
                .create(TaskViewModel.class);

        sharedViewModel = new ViewModelProvider(this)
                .get(SharedViewModel.class);

        taskViewModel.getAllTasks().observe(this, tasks -> {
            recyclerViewAdaptor = new RecyclerViewAdaptor(tasks, this);
            recyclerView.setAdapter(recyclerViewAdaptor);
        });

        FloatingActionButton fab = findViewById(R.id.fab);
        fab.setOnClickListener(view -> {
//            Task task = new Task("Task" + counter++, Priority.MEDIUM,
Calendar.getInstance().getTime(),
//                    Calendar.getInstance().getTime(),false);
//            TaskViewModel.insert(task);
            showBottomSheetDialog();
        });
    }

    private void showBottomSheetDialog() {
        bottomSheetFragment.show(getSupportFragmentManager(),
bottomSheetFragment.getTag());
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
```

```
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
        if (id == R.id.action_settings) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onTodoClick(Task task) {
        sharedViewModel.selectItem(task);
        sharedViewModel.setIsEdit(true);
        showBottomSheetDialog();
    }

    @Override
    public void onTodoRadioButtonClick(Task task) {
        TaskViewModel.delete(task);
        recyclerViewAdaptor.notifyDataSetChanged();
    }
}
```

BottomSheetFragment.java:

```
package com.darkwizz.routine;

import android.os.Bundle;
import android.text.TextUtils;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.CalendarView;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.RadioButton;
import android.widget.RadioGroup;

import com.darkwizz.routine.model.Priority;
import com.darkwizz.routine.model.SharedViewModel;
import com.darkwizz.routine.model.Task;
import com.darkwizz.routine.model.TaskViewModel;
```

```
import com.darkwizz.routine.util.Utils;
import com.google.android.material.bottomsheet.BottomSheetDialogFragment;
import com.google.android.material.chip.Chip;
import com.google.android.material.snackbar.Snackbar;

import androidx.annotation.NonNull;
import androidx.constraintlayout.widget.Group;
import androidx.lifecycle.ViewModelProvider;

import java.util.Calendar;
import java.util.Date;

public class BottomSheetFragment extends BottomSheetDialogFragment implements
View.OnClickListener{

    private EditText enterTodo;
    private ImageButton calendarButton;
    private ImageButton priorityButton;
    private RadioGroup priorityRadioGroup;
    private RadioButton selectedRadioButton;
    private int selectedButtonId;
    private ImageButton saveButton;
    private CalendarView calendarView;
    private Group calendarGroup;
    private Date dueDate;
    Calendar calendar = Calendar.getInstance();
    private SharedViewModel sharedViewModel;
    private boolean isEdit;
    private Priority priority;

    public BottomSheetFragment(){

    }

    @Override
    public View onCreateView(
            LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState
    ) {
        // Inflate the layout for this fragment
        View view = inflater.inflate(R.layout.bottom_sheet, container, false);
        calendarGroup = view.findViewById(R.id.calendar_group);
        calendarView = view.findViewById(R.id.calendar_view);
        calendarButton = view.findViewById(R.id.today_calendar_button);
        enterTodo = view.findViewById(R.id.enter_todo_et);
        saveButton = view.findViewById(R.id.save_todo_button);
        priorityButton = view.findViewById(R.id.priority_todo_button);
        priorityRadioGroup = view.findViewById(R.id.radioGroup_priority);
```

```
        Chip todayChip = view.findViewById(R.id.today_chip);
        todayChip.setOnClickListener(this);
        Chip tommorowChip = view.findViewById(R.id.tomorrow_chip);
        tommorowChip.setOnClickListener(this);
        Chip nextWeekChip = view.findViewById(R.id.next_week_chip);
        nextWeekChip.setOnClickListener(this);

        return view;
    }

    @Override
    public void onResume() {
        super.onResume();

        if(sharedViewModel.getSelectedItem().getValue() != null){
            isEdit = sharedViewModel.getIsEdit();

            Task task = sharedViewModel.getSelectedItem().getValue();
            enterTodo.setText(task.getTask());
        }
    }

    public void onViewCreated(@NonNull View view, Bundle savedInstanceState) {
        super.onViewCreated(view, savedInstanceState);
        sharedViewModel = new ViewModelProvider(requireActivity())
                .get(SharedViewModel.class);

        calendarButton.setOnClickListener(view1 -> {
            calendarGroup.setVisibility(
                    calendarGroup.getVisibility() == View.GONE ? View.VISIBLE
: View.GONE
            );
            Utils.hideSoftKeyboard(view1);
        });
        calendarView.setOnDateChangeListener((calendarView, year, month,
dayOfMonth) ->{
            calendar.clear();
            calendar.set(year, month, dayOfMonth);
            dueDate = calendar.getTime();
        });
        priorityButton.setOnClickListener(view1 -> {
            Utils.hideSoftKeyboard(view1);
            priorityRadioGroup.setVisibility(
                    priorityRadioGroup.getVisibility() == View.GONE ?
View.VISIBLE : View.GONE
            );
```

```
            priorityRadioGroup.setOnCheckedChangeListener((radioGroup,
checkedId) -> {
                if (priorityRadioGroup.getVisibility() == View.VISIBLE){
                    selectedButtonId = checkedId;
                    selectedRadioButton = view.findViewById(selectedButtonId);
                    if(selectedRadioButton.getId() == R.id.radioButton_high){
                        priority = Priority.HIGH;
                    }else if(selectedRadioButton.getId() ==
R.id.radioButton_med){
                        priority = Priority.MEDIUM;
                    }else if(selectedRadioButton.getId() ==
R.id.radioButton_low){
                        priority = Priority.LOW;
                    }else{
                        priority = Priority.LOW;
                    }
                }else{
                    priority = Priority.LOW;
                }
            });
        });

        saveButton.setOnClickListener(view1 -> {
            String task = enterTodo.getText().toString().trim();
            if(!TextUtils.isEmpty(task) && dueDate != null && priority !=
null){
                Task myTask = new Task(task, priority,
                        dueDate, Calendar.getInstance().getTime(),
                        false);
                if(isEdit){
                    Task updateTask =
sharedViewModel.getSelectedItem().getValue();
                    updateTask.setTask(task);
                    updateTask.setDateCreated(Calendar.getInstance().getTime()
);
                    updateTask.setPriority(priority   );
                    updateTask.setDueDate(dueDate);
                    TaskViewModel.update(updateTask);
                    sharedViewModel.setIsEdit(false);
                }else {
                    TaskViewModel.insert(myTask);
                }
                enterTodo.setText("");
                if (this.isVisible()){
                    this.dismiss();
                }
            }else{
```

```
                Snackbar.make(saveButton, R.string.empty_field,
Snackbar.LENGTH_LONG)
                        .show();
            }
        });
    }

    @Override
    public void onClick(View view) {
        int id = view.getId();
        if (id == R.id.today_chip){
            calendar.add(Calendar.DAY_OF_YEAR,0);
            dueDate = calendar.getTime();
        }else if(id==R.id.tomorrow_chip){
            calendar.add(Calendar.DAY_OF_YEAR,1);
            dueDate = calendar.getTime();
        }else if(id==R.id.next_week_chip){
            calendar.add(Calendar.DAY_OF_YEAR,7);
            dueDate = calendar.getTime();
        }
    }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
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=".MainActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.Todoister.AppBarOverlay">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            app:popupTheme="@style/Theme.Todoister.PopupOverlay" />

    </com.google.android.material.appbar.AppBarLayout>
```

```
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fab"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end"
        android:layout_margin="@dimen/fab_margin"
        app:srcCompat="@android:drawable/ic_input_add"
        tools:ignore="ContentDescription" />


    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycler_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="@dimen/dimen_60"
        android:padding="@dimen/recycler_padding"
        android:visibility="visible">

    </androidx.recyclerview.widget.RecyclerView>

    <include layout="@layout/bottom_sheet"/>

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

bottom_sheet.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"
    android:id="@+id/bottomSheet"
    style="@style/Animation.Design.BottomSheetDialog"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#fff"
    app:behavior_hideable="false"
    app:behavior_peekHeight="@dimen/dimen_32"
    app:layout_behavior="com.google.android.material.bottomsheet.BottomSheetBe
havior">


    <EditText
        android:id="@+id/enter_todo_et"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_margin="@dimen/dimen_16"
        android:hint="@string/enter_todo_hint"
        android:inputType="textPersonName"
        android:padding="@dimen/dimen_16"
```

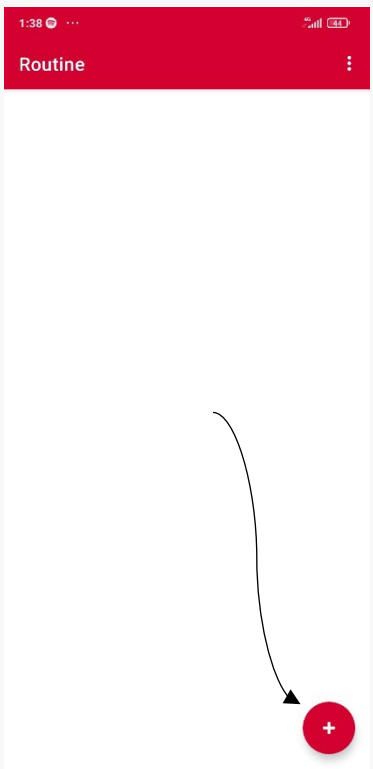
```
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ImageButton
        android:id="@+id/priority_todo_button"
        android:layout_width="77dp"
        android:layout_height="46dp"
        android:layout_marginStart="24dp"
        android:layout_marginTop="16dp"
        android:background="@android:color/transparent"
        app:layout_constraintStart_toEndOf="@+id/today_calendar_button"
        app:layout_constraintTop_toBottomOf="@+id/enter_todo_et"
        app:srcCompat="@drawable/ic_baseline_outlined_flag_24"
        android:contentDescription="@string/image_description" />

    <ImageButton
        android:id="@+id/save_todo_button"
        android:layout_width="70dp"
        android:layout_height="72dp"
        android:layout_marginTop="16dp"
        android:background="@android:color/transparent"
        android:scaleX="0.4"
        android:scaleY="0.4"
        android:contentDescription="TODO"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.843"
        app:layout_constraintStart_toEndOf="@+id/priority_todo_button"
        app:layout_constraintTop_toBottomOf="@+id/enter_todo_et"
        app:srcCompat="@drawable/ic_baseline_arrow_circle_up_24" />

    <ImageButton
        android:id="@+id/today_calendar_button"
        android:layout_width="74dp"
        android:layout_height="48dp"
        android:layout_marginStart="16dp"
        android:contentDescription="@string/image_description"
        android:layout_marginTop="16dp"
        android:background="@android:color/transparent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/enter_todo_et"
        app:srcCompat="@drawable/ic_baseline_calendar_today_24" />

    <com.google.android.material.chip.Chip
        android:id="@+id/tomorrow_chip"
        android:layout_width="269dp"
        android:layout_height="41dp"
        android:text="@string/tomorrow"
```

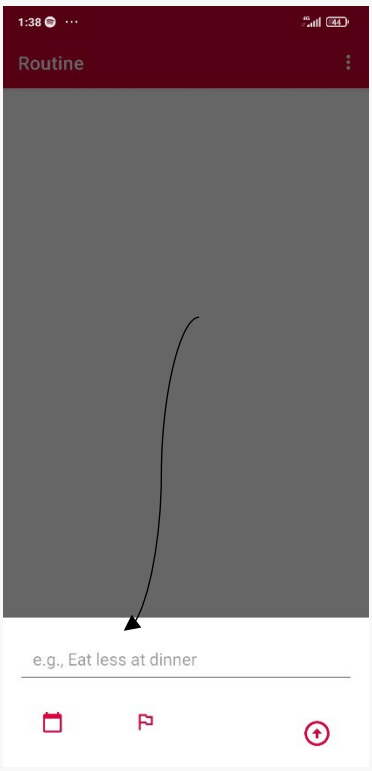
```
        app:chipIcon="@drawable/ic_baseline_wb_sunny_24"
        app:layout_constraintEnd_toEndOf="@+id/next_week_chip"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="@+id/next_week_chip"
        app:layout_constraintTop_toBottomOf="@+id/today_chip" />

    <com.google.android.material.chip.Chip
        android:id="@+id/next_week_chip"
        android:layout_width="272dp"
        android:layout_height="41dp"
        android:background="@android:color/transparent"
        android:text="@string/next_week"
        app:chipIcon="@drawable/ic_baseline_next_week_24"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tomorrow_chip" />

    <com.google.android.material.chip.Chip
        android:id="@+id/today_chip"
        android:layout_width="267dp"
        android:layout_height="41dp"
        android:layout_marginTop="2dp"
        android:text="@string/today"
        app:chipIcon="@drawable/ic_baseline_today_24"
        app:layout_constraintEnd_toEndOf="@+id/tomorrow_chip"
        app:layout_constraintStart_toStartOf="@+id/tomorrow_chip"
        app:layout_constraintTop_toBottomOf="@+id/textView" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="244dp"
        android:layout_height="33dp"
        android:layout_marginTop="48dp"
        android:gravity="center_horizontal"
        android:text="@string/due_date_text"
        android:textSize="18sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/priority_todo_button" />

    <CalendarView
        android:id="@+id/calendar_view"
        android:layout_width="318dp"
        android:layout_height="296dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
```

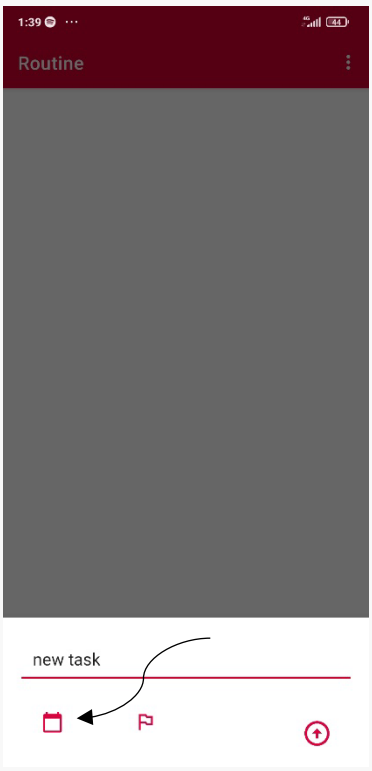
```
        app:layout_constraintHorizontal_bias="0.494"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/next_week_chip"
        app:layout_constraintVertical_bias="0.16000003" />

    <RadioGroup
        android:id="@+id/radioGroup_priority"
        android:layout_width="0dp"
        android:layout_height="51dp"
        android:layout_marginTop="8dp"
        android:orientation="horizontal"
        android:padding="@dimen/dimen_10"
        android:visibility="gone"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/priority_todo_button">

        <RadioButton
            android:id="@+id/radioButton_high"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="#C91517"
            android:text="@string/radio_high" />

        <RadioButton
            android:id="@+id/radioButton_med"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="#FFB300"
            android:text="@string/radio_med" />
    </RadioGroup>

    <androidx.constraintlayout.widget.Group
        android:id="@+id/calendar_group"
        android:layout_width="280dp"
        android:layout_height="0dp"
        android:visibility="gone"
        app:constraint_referenced_ids="calendar_view,next_week_chip,tomorrow_c
hip,today_chip,textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />


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

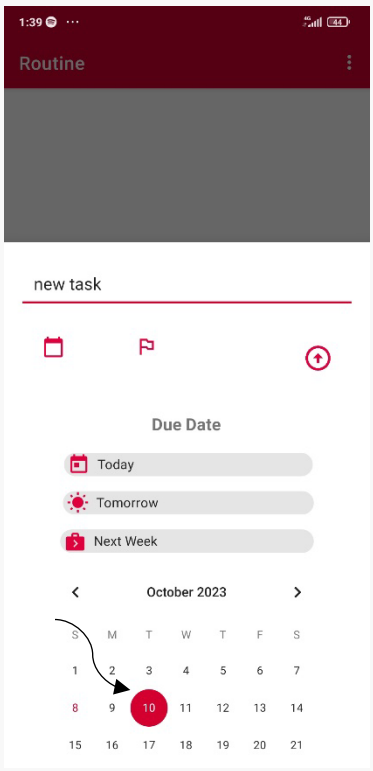
Demo:

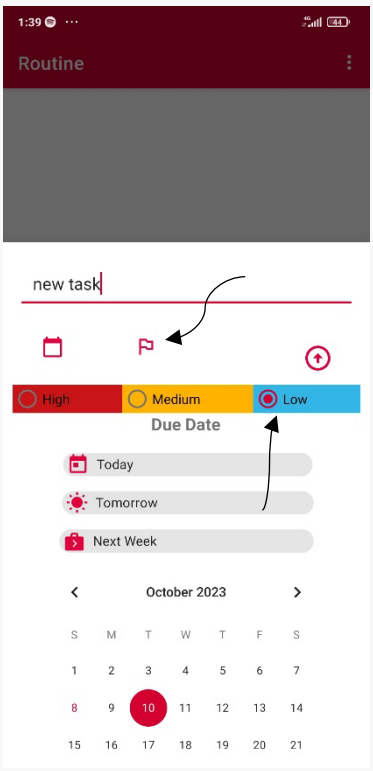
Creating a task:

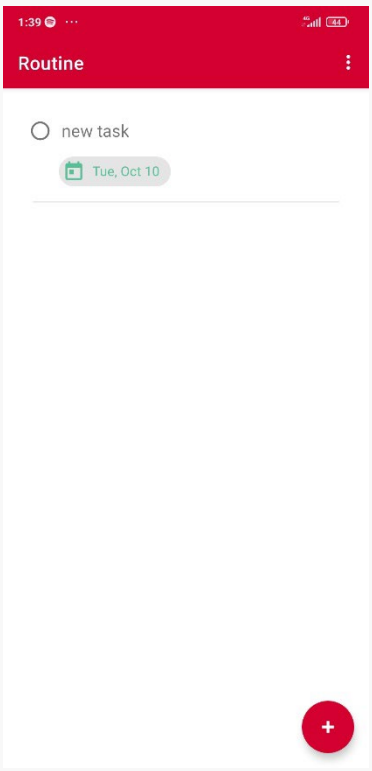




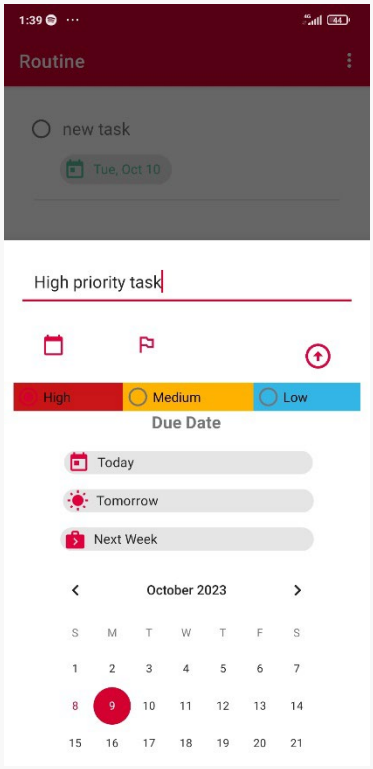


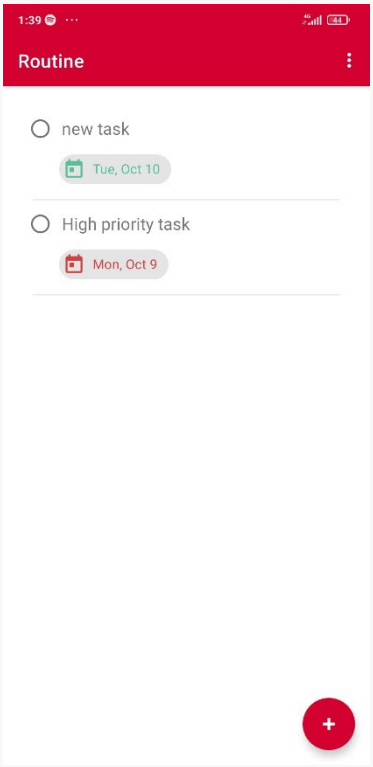


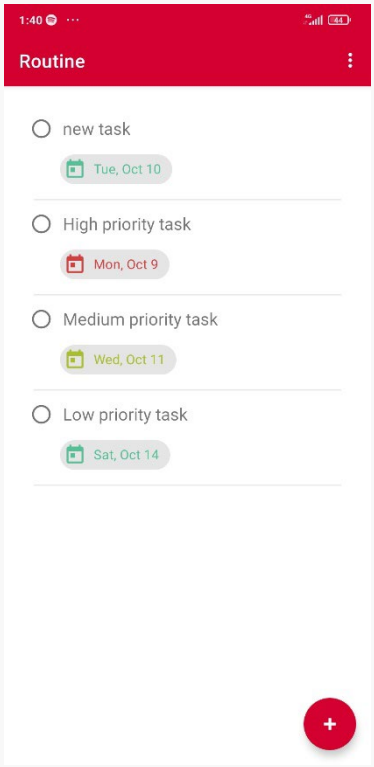




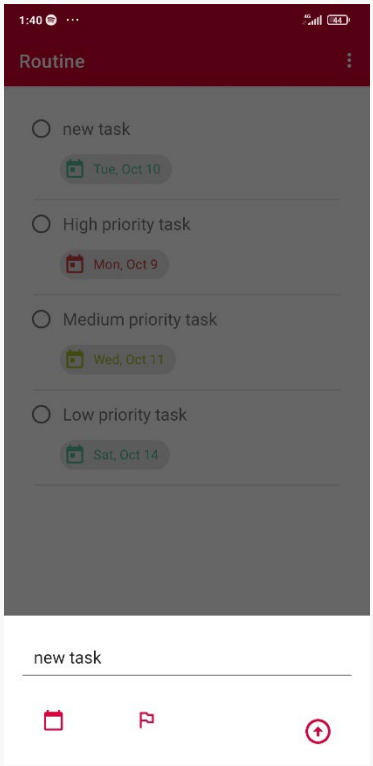
Prioritizing task:

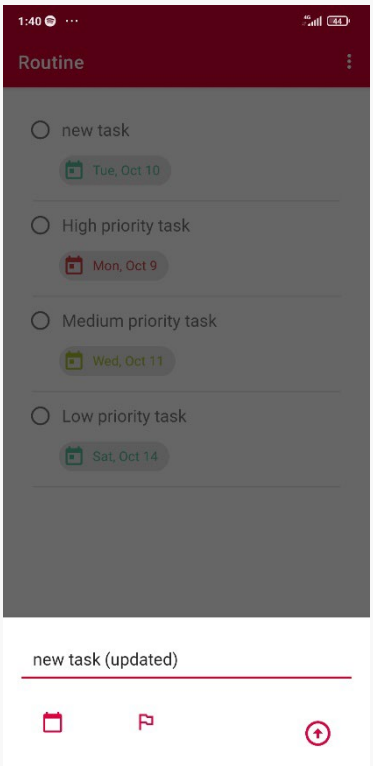


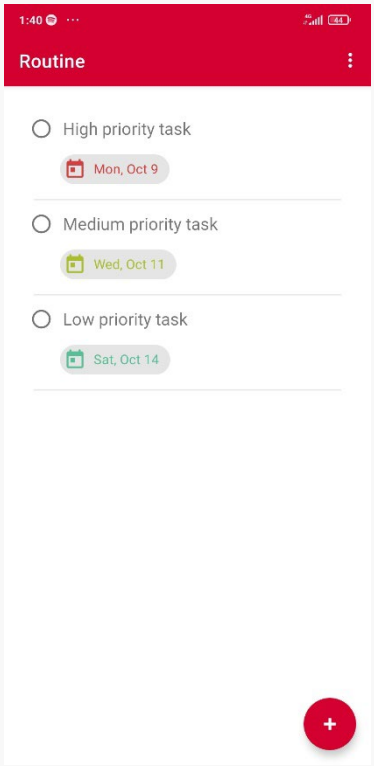




Updating task:







End Note:

In the journey of creating "ROUTINE," the to-do app for efficient task management, we learned valuable lessons in software development, user-centric design, and project management. This assignment allowed me to explore the intricacies of Android app development using Java and Room Persistence Library, fostering a deeper understanding of mobile application architecture.

As we crafted each feature and meticulously designed the user interface, I gained insights into the importance of simplicity and user-friendliness in app design. The emphasis on user-centric features such as task prioritization and due date management reflects a commitment to delivering a product that genuinely enhances productivity and organization.

We want to express our gratitude to my instructor for this assignment, which provided an opportunity to apply theoretical knowledge to real-world development. The process of creating "ROUTINE" has been both challenging and rewarding, and I look forward to further honing my skills as a developer.

Lastly, we want to acknowledge the support and encouragement of our friends and fellow students who provided valuable feedback and insights throughout this project.

Thank you