**To Do List App**

-Ashley Smith, Braden King, Ramya Nagarajan, Shalaka Sharma

**Application Description-**

Our Android application is a to-do-list which can keep track of tasks that the user wants to complete, along with the date on which the task is to be done. The tasks can be added, deleted and viewed based on the status, which is either pending or completed. The main view shows the list of all tasks (pending and completed).

**Code**

Following is the code that drives the application.

This is the Main Activity.

**package** iu.i527.shalaka.todolistapp;  
  
  
**import** android.content.ContentValues;  
**import** android.content.Intent;  
**import** android.content.res.ColorStateList;  
**import** android.database.Cursor;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.graphics.Color;  
  
**import** android.graphics.Typeface;  
**import** android.media.MediaPlayer;  
**import** android.os.Bundle;  
**import** android.support.design.widget.FloatingActionButton;  
**import** android.support.design.widget.Snackbar;  
**import** android.view.Gravity;  
**import** android.view.View;  
**import** android.support.design.widget.NavigationView;  
**import** android.support.v4.view.GravityCompat;  
**import** android.support.v4.widget.DrawerLayout;  
**import** android.support.v7.app.ActionBarDrawerToggle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.Toolbar;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.view.ViewGroup;  
**import** android.widget.CheckBox;  
**import** android.widget.CompoundButton;  
**import** android.widget.EditText;  
**import** android.widget.ImageButton;  
  
**import** android.widget.TableLayout;  
**import** android.widget.TableRow;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** java.text.DateFormat;  
**import** java.text.SimpleDateFormat;  
**import** java.util.ArrayList;  
**import** java.util.Date;  
  
**public class** MainActivity **extends** AppCompatActivity  
 **implements** NavigationView.OnNavigationItemSelectedListener {  
  
 *//ArrayList<ToDoTask> toDoList;* String **lastNavSelected** = **"all"**;  
 TodoListDatabaseHelper **mDbHelper**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 **mDbHelper** = **new** TodoListDatabaseHelper(getApplicationContext());  
 setContentView(R.layout.***activity\_main***);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 setSupportActionBar(toolbar);  
 FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.***fab***);  
 fab.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Intent intent = **new** Intent(getApplicationContext(), ScrollingActivity.**class**);  
 startActivity(intent);  
 }  
 });  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 ActionBarDrawerToggle toggle = **new** ActionBarDrawerToggle(  
 **this**, drawer, toolbar, R.string.***navigation\_drawer\_open***, R.string.***navigation\_drawer\_close***);  
 drawer.setDrawerListener(toggle);  
 toggle.syncState();  
  
 NavigationView navigationView = (NavigationView) findViewById(R.id.***nav\_view***);  
 navigationView.setNavigationItemSelectedListener(**this**);  
  
 renderToDoList(**"all"**);  
 }  
  
 **public void** renderToDoList(String action) {  
 ArrayList<ToDoTask> toDoList = **mDbHelper**.fetchToDoListFromDB(**lastNavSelected**);  
 TableLayout tableLayout = (TableLayout) findViewById(R.id.***tableLayout***);  
 tableLayout.removeAllViews();  
 TextView textView = **new** TextView(getApplicationContext());  
 textView.setPadding(10, 60, 10, 60);  
 textView.setTextSize(24);  
 textView.setTextColor(Color.*parseColor*(**"#000000"**));  
 textView.setTypeface(textView.getTypeface(), Typeface.***BOLD***);  
 **if** (action == **"all"**) {  
 textView.setText(**"All Tasks"**);  
 } **else if** (action == **"completed"**) {  
 textView.setText(**"Completed Tasks"**);  
 } **else if** (action == **"pending"**) {  
 textView.setText(**"Pending Tasks"**);  
 }  
  
 TableRow header = **new** TableRow(getApplicationContext());  
 header.addView(textView);  
 tableLayout.addView(header);  
 **if**(toDoList.size() == 0) {  
 TextView textView2 = **new** TextView(getApplicationContext());  
 textView2.setText(**"\*\*No tasks to show\*\*"**);  
 textView2.setTypeface(textView.getTypeface(), Typeface.***ITALIC***);  
 textView2.setTextColor(Color.*parseColor*(**"#000000"**));  
 textView2.setTextSize(14);  
 textView2.setPadding(30, 30, 10, 30);  
 TableRow row = **new** TableRow(getApplicationContext());  
 row.addView(textView2);  
 tableLayout.addView(row);  
 }  
 **for** (ToDoTask task : toDoList) {  
 **if** (action.equals(**"completed"**) && !task.isStatus()) {  
 **continue**;  
 } **else if** (action.equals(**"pending"**) && task.isStatus()) {  
 **continue**;  
 }  
 TableRow tableRow = **new** TableRow(getApplicationContext());  
  
 CheckBox cb = **new** CheckBox(getApplicationContext());  
 cb.setText(task.getTask\_description());  
 cb.setChecked(task.isStatus());  
 cb.setTextColor(Color.*parseColor*(**"#000000"**));  
 cb.setTextSize(20);  
 ColorStateList colorStateList = **new** ColorStateList(  
 **new int**[][]{  
 **new int**[]{-android.R.attr.***state\_checked***}, *// unchecked* **new int**[]{android.R.attr.***state\_checked***} *// checked* },  
 **new int**[]{  
 Color.*parseColor*(**"#000000"**),  
 Color.*parseColor*(**"#FF1493"**)  
 }  
 );  
 cb.setButtonTintList(colorStateList);  
 cb.setPadding(10, 30, 10, 30);  
 **final** MediaPlayer mp = MediaPlayer.*create*(getApplicationContext(), R.raw.***todolist***);  
 cb.setOnCheckedChangeListener(**new** CompoundButton.OnCheckedChangeListener() {  
 @Override  
 **public void** onCheckedChanged(CompoundButton buttonView, **boolean** isChecked) {  
 mp.start();  
 CheckBox cb = (CheckBox) buttonView;  
 TableRow t = (TableRow) buttonView.getParent();  
 EditText editText1 = (EditText) t.getChildAt(1);  
 Date date = **new** Date(Long.*parseLong*(editText1.getText().toString()));  
 **if** (isChecked) {  
 **mDbHelper**.changeTaskStatusInDB(cb.getText().toString(), date, **true**);  
 Toast.*makeText*(getApplicationContext(), **"Task marked as completed"**, Toast.***LENGTH\_SHORT***).show();  
 } **else** {  
 **mDbHelper**.changeTaskStatusInDB(cb.getText().toString(), date, **false**);  
 Toast.*makeText*(getApplicationContext(), **"Task marked as pending"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 renderToDoList(**lastNavSelected**);  
  
 }  
 });  
  
 tableRow.addView(cb);  
 EditText editText = **new** EditText(getApplicationContext());  
 editText.setText(task.getDate().getTime() + **""**);  
 editText.setVisibility(View.***INVISIBLE***);  
 ViewGroup.LayoutParams lparams = **new** ViewGroup.LayoutParams(0, 0);  
 editText.setLayoutParams(lparams);  
 tableRow.addView(editText);  
  
 TextView textview = **new** TextView(getApplicationContext());  
 DateFormat formatter = **new** SimpleDateFormat(**"dd/MM/yyyy"**);  
 String dt = formatter.format(task.getDate());  
 textview.setText(dt);  
 textview.setTextColor(Color.*parseColor*(**"#000000"**));  
 textview.setTextSize(20);  
 tableRow.addView(textview);  
  
 ImageButton button = **new** ImageButton(getApplicationContext());  
 button.setImageResource(R.drawable.***btn\_close\_normal***);  
 button.setBackgroundColor(Color.***TRANSPARENT***);  
 button.setOnClickListener(**new** View.OnClickListener() {  
  
 @Override  
 **public void** onClick(View view) {  
 mp.start();  
 ImageButton b = (ImageButton) view;  
 TableRow t = (TableRow) b.getParent();  
 CheckBox cb = (CheckBox) t.getChildAt(0);  
 EditText editText1 = (EditText) t.getChildAt(1);  
  
 **final** String taskDescription = cb.getText().toString();  
 **final** Date date = **new** Date(Long.*parseLong*(editText1.getText().toString()));  
 **final boolean** status = cb.isChecked();  
 **mDbHelper**.deleteFromDatabase(taskDescription, date.getTime()+**""**);  
 renderToDoList(**lastNavSelected**);  
  
 Snackbar.*make*(findViewById(R.id.***tableLayout***), **"1 Item Removed"**, Snackbar.***LENGTH\_LONG***)  
 .setAction(**"UNDO"**,  
 **new** View.OnClickListener() {  
  
 @Override  
 **public void** onClick(View view) {  
 ToDoTask toDoTask = **new** ToDoTask(taskDescription, date, status);  
 **mDbHelper**.addTodoTaskToDB(toDoTask);  
 renderToDoList(**lastNavSelected**);  
 }  
 }).show();  
  
 }  
  
 });  
 tableRow.addView(button);  
 tableLayout.addView(tableRow);  
  
 }  
  
 }  
  
 **public** ArrayList<ToDoTask> getListItems() {  
 *// Gets the data repository in write mode* SQLiteDatabase db = **mDbHelper**.getWritableDatabase();  
  
 ArrayList<ToDoTask> toDoList = **new** ArrayList<ToDoTask>();  
 Date dt = **new** Date();  
 ToDoTask t1 = **new** ToDoTask(**"Call James"**, dt, **false**);  
 ToDoTask t2 = **new** ToDoTask(**"Call Mom"**, dt, **false**);  
 ToDoTask t3 = **new** ToDoTask(**"Buy apples"**, dt, **false**);  
 ToDoTask t4 = **new** ToDoTask(**"Water plants"**, dt, **false**);  
 toDoList.add(t1);  
 toDoList.add(t2);  
 toDoList.add(t3);  
 toDoList.add(t4);  
 **return** toDoList;  
 }  
  
 @Override  
 **public void** onBackPressed() {  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 **if** (drawer.isDrawerOpen(GravityCompat.***START***)) {  
 drawer.closeDrawer(GravityCompat.***START***);  
 } **else** {  
 **super**.onBackPressed();  
 }  
 }  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.***main***, menu);  
 **return true**;  
 }  
  
 @SuppressWarnings(**"StatementWithEmptyBody"**)  
 @Override  
 **public boolean** onNavigationItemSelected(MenuItem item) {  
 *// Handle navigation view item clicks here.* **int** id = item.getItemId();  
 **if** (id == R.id.***nav\_all\_tasks***) {  
 *// Handle the all tasks* **lastNavSelected** = **"all"**;  
 renderToDoList(**"all"**);  
 } **else if** (id == R.id.***nav\_pending***) {  
 **lastNavSelected** = **"pending"**;  
 renderToDoList(**"pending"**);  
 } **else if** (id == R.id.***nav\_completed***) {  
 **lastNavSelected** = **"completed"**;  
 renderToDoList(**"completed"**);  
 }  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 **return true**;  
 }  
  
 */\*public void deleteFromList(String taskDesc, Date date) {  
 DateFormat formatter = new SimpleDateFormat("dd/MM/yyyy");  
 String dtString = formatter.format(date);  
 int indexToDelete = -1;  
 for (int i = 0; i < toDoList.size(); i++) {  
 ToDoTask task = toDoList.get(i);  
 String dt = formatter.format(task.getDate());  
 if (task.getTask\_description().equals(taskDesc) && dtString.equals(dt)) {  
 indexToDelete = i;  
 break;  
 }  
 }  
 if (indexToDelete != -1) {  
 toDoList.remove(indexToDelete);  
 }  
 renderToDoList(toDoList, lastNavSelected);  
 }\*/  
  
 /\*public void changeTaskStatus(String taskDesc, Date date, boolean isCompleted) {  
 DateFormat formatter = new SimpleDateFormat("dd/MM/yyyy");  
 String dtString = formatter.format(date);  
 for (int i = 0; i < toDoList.size(); i++) {  
 ToDoTask task = toDoList.get(i);  
 String dt = formatter.format(task.getDate());  
 if (task.getTask\_description().equals(taskDesc) && dtString.equals(dt)) {  
 task.setStatus(isCompleted);  
 break;  
 }  
 }  
 renderToDoList(toDoList, lastNavSelected);  
 }\*/* @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\_clear\_list***) {  
 **mDbHelper**.clearToDoListInDB();  
 renderToDoList(**lastNavSelected**);  
 **return true**;  
 }  
  
 **return super**.onOptionsItemSelected(item);  
 }  
  
 @Override  
 **protected void** onDestroy() {  
 **mDbHelper**.close();  
 **super**.onDestroy();  
 }  
}

This is the ToDoTask Activity.

**package** iu.i527.shalaka.todolistapp;  
  
**import** java.util.Date;  
  
*/\*\*  
 \* Created by Ramya on 10/14/2017.  
 \*/***import** java.io.Serializable;  
  
@SuppressWarnings(**"serial"**)  
**public class** ToDoTask **implements** Serializable {  
  
 **private long id**;  
 **private** String **task\_description**;  
 **private** Date **date**;  
 **private boolean status**;  
  
 **public** Date getDate() {  
 **return date**;  
 }  
  
 **public void** setDate(Date date) {  
 **this**.**date** = date;  
 }  
  
 **public** String getTask\_description() {  
 **return task\_description**;  
 }  
  
 **public void** setTask\_description(String task\_description) {  
 **this**.**task\_description** = task\_description;  
 }  
  
 **public boolean** isStatus() {  
 **return status**;  
 }  
  
 **public void** setStatus(**boolean** status) {  
 **this**.**status** = status;  
 }  
  
 **public** ToDoTask(String task\_description, Date date, **boolean** status) {  
 **this**.**task\_description** = task\_description;  
 **this**.**date** = date;  
 **this**.**status** = status;  
 }  
  
  
 **public** ToDoTask() {  
 }  
  
 **public** ToDoTask(**long** id, String task\_description, Date date, **boolean** status) {  
 **this**.**id** = id;  
 **this**.**task\_description** = task\_description;  
 **this**.**date** = date;  
 **this**.**status** = status;  
 }  
  
 **public long** getId() {  
 **return id**;  
 }  
  
 **public void** setId(**long** id) {  
 **this**.**id** = id;  
 }  
}

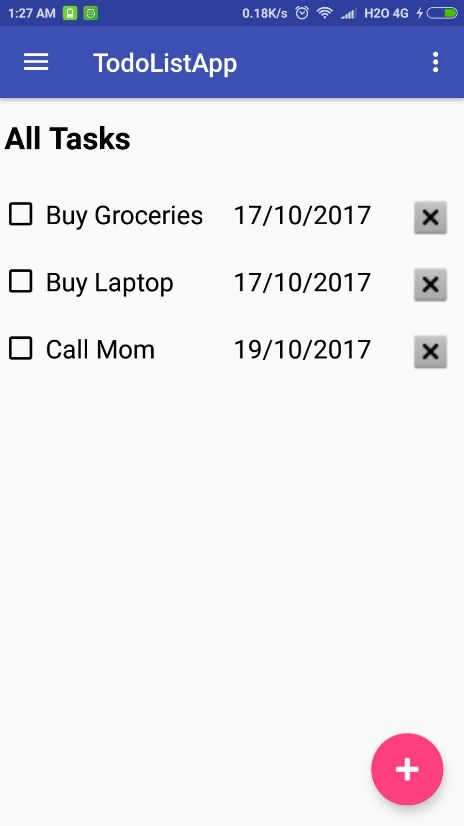
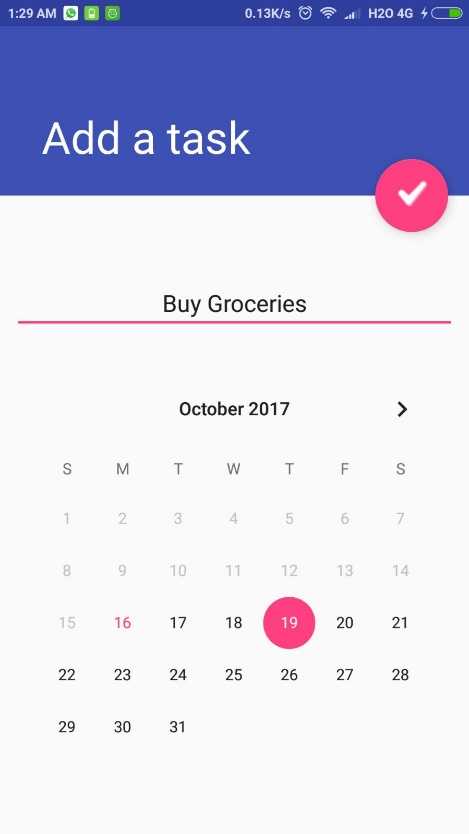
This is the Scrolling activity.

**package** iu.i527.shalaka.todolistapp;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.annotation.NonNull;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.Toolbar;  
**import** android.view.View;  
**import** android.widget.CalendarView;  
**import** android.widget.EditText;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** java.util.ArrayList;  
**import** java.util.Calendar;  
**import** java.util.Date;  
  
**public class** ScrollingActivity **extends** AppCompatActivity {  
  
 TodoListDatabaseHelper **mDbHelper**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 **mDbHelper** = **new** TodoListDatabaseHelper(getApplicationContext());  
 setContentView(R.layout.***activity\_scrolling***);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 CalendarView calendarView = (CalendarView) findViewById(R.id.***simpleCalendarView***);  
 calendarView.setMinDate(**new** Date().getTime());  
 Calendar calendar = Calendar.*getInstance*();  
 Date date = calendar.getTime();  
 TextView textview = (TextView) findViewById(R.id.***dateTextView***);  
 textview.setText(date.getTime()+**""**);  
 calendarView.setOnDateChangeListener(**new** CalendarView.OnDateChangeListener() {  
 @Override  
 **public void** onSelectedDayChange(@NonNull CalendarView view, **int** year, **int** month, **int** dayOfMonth) {  
 Calendar calendar = Calendar.*getInstance*();  
 calendar.set(year,month,dayOfMonth);  
 Date date = calendar.getTime();  
 TextView textview = (TextView) findViewById(R.id.***dateTextView***);  
 textview.setText(date.getTime()+**""**);  
 }  
 });  
  
 setSupportActionBar(toolbar);  
 }  
  
 **public void** onClickSaveButton(View view) {  
 TextView textview = (TextView) findViewById(R.id.***dateTextView***);  
 EditText editText = (EditText) findViewById(R.id.***editText***);  
 String taskDescription = editText.getText().toString();  
 Date date = **new** Date(Long.*parseLong*(textview.getText().toString()));  
 ToDoTask newTask = **new** ToDoTask(taskDescription, date, **false**);  
 **mDbHelper**.addTodoTaskToDB(newTask);  
 Toast.*makeText*(**this**, **"Task Added"**, Toast.***LENGTH\_SHORT***).show();  
 Intent intent = **new** Intent(getApplicationContext(), MainActivity.**class**);  
 startActivity(intent);  
 }  
  
 @Override  
 **protected void** onDestroy() {  
 **mDbHelper**.close();  
 **super**.onDestroy();  
 }  
}

**Screenshots**

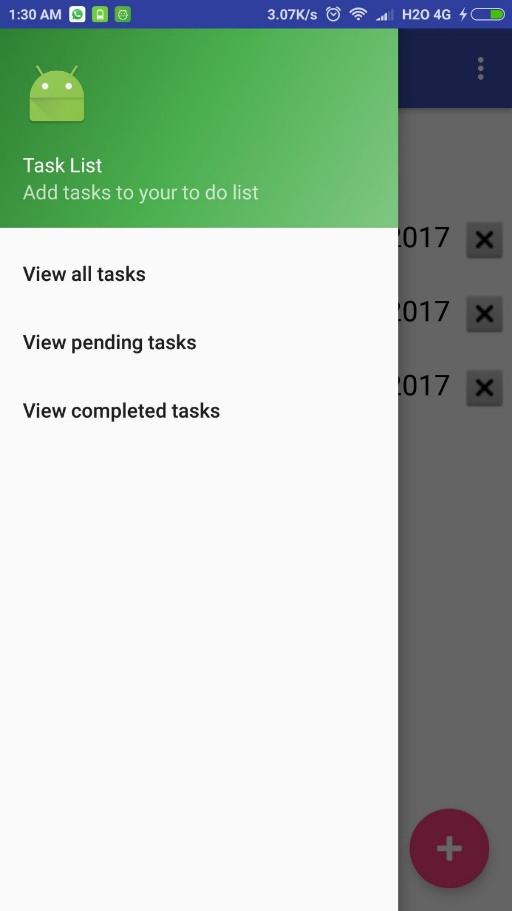
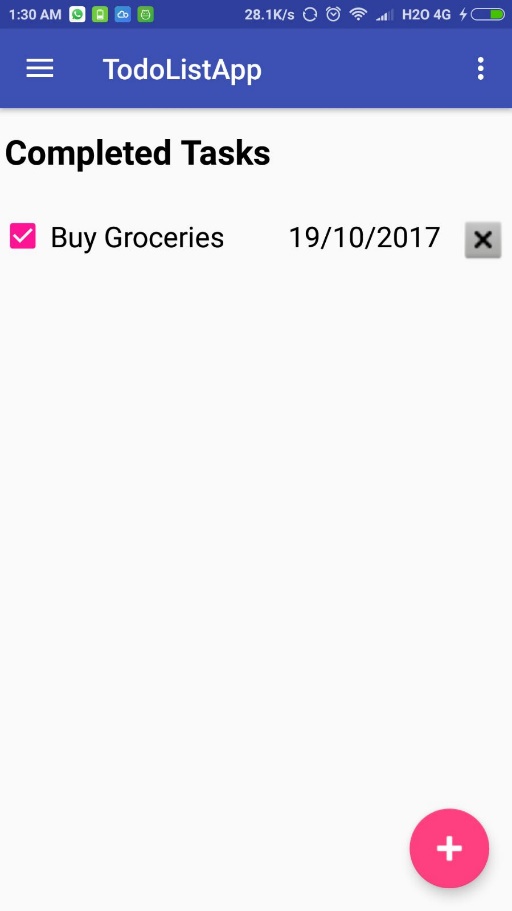
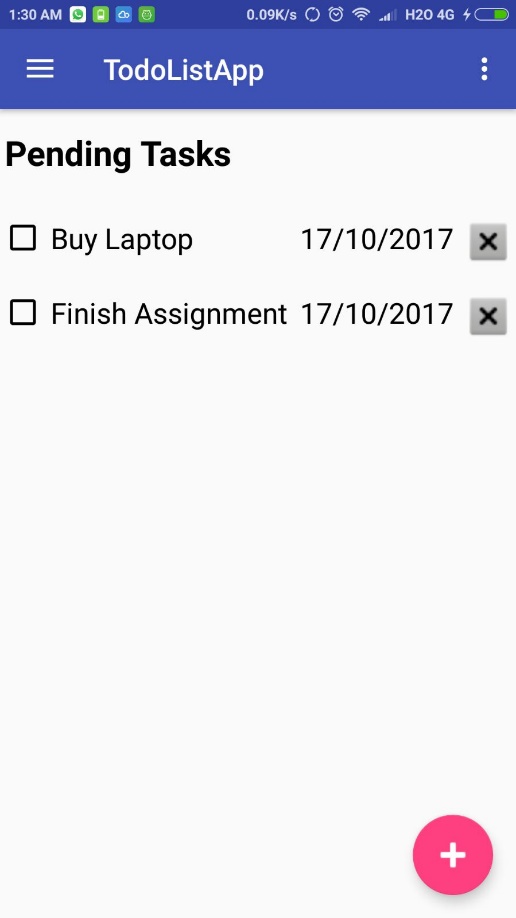
Following is the screenshot of all the screens.

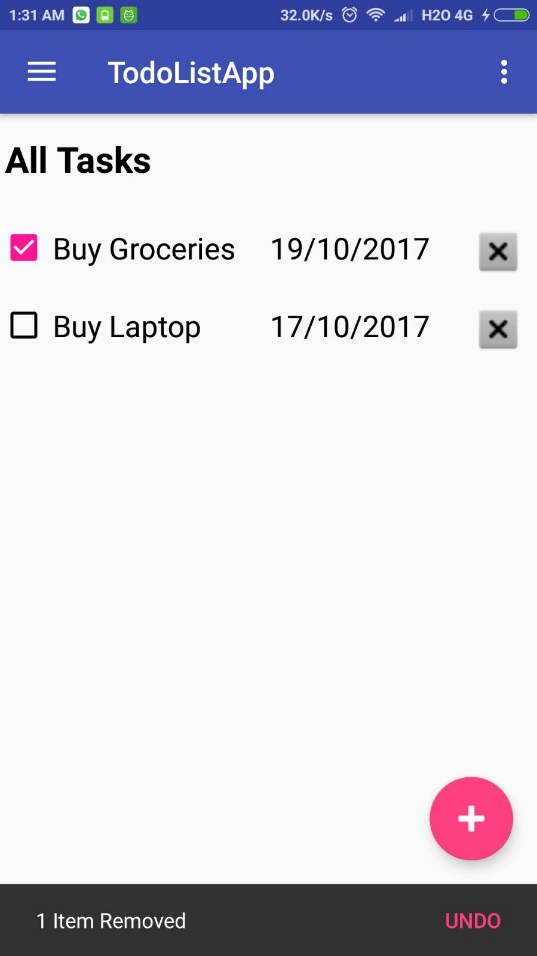
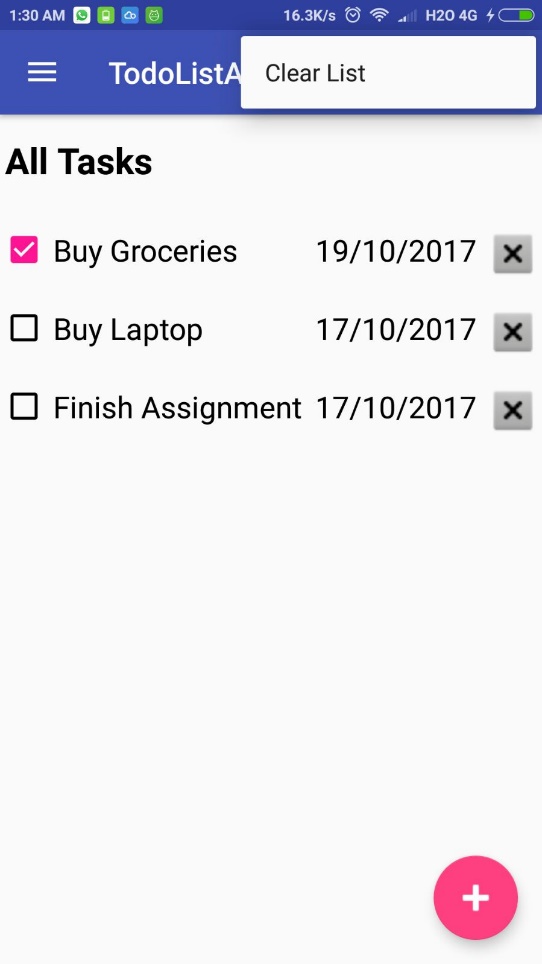
This is the first screen when the app starts.

On clicking +

The navigation drawer The completed tasks view The pending tasks view

After a task has been removed from list The all tasks view with the clear list option  

The All Tasks List, once checked, a tone plays An empty task list when all tasks are completed

