

## Members :

Name	ID	Section	Name in GitHub
1. Mr. Chawapat Yongjittikornkul	56070501010	Sec.A	darkream
2. Miss Phenphitcha Patthanajitsilp	56070501033	Sec.A	phenphitcha555
3. Mr. Saweera Ngamkamolsuk	56070501050	Sec.B	Saweera2903
4. Mr. Saritrat Jirakulpornchai	56070501056	Sec.B	Sikiryl
5. Mr. Intouch Sangkrajang	56070501068	Sec.B	nongart1155

**Module :** ทำ add product to cart โดยจะมีหน้า product ที่แสดงสินค้าทั้งหมด เมื่อกดปุ่ม add to cart สินค้านั้นจะมาอยู่ในหน้า cart จากนั้นในหน้า cart จะคำนวณราคาทั้งหมดให้

## Design specification and functional specification

## 4.1. DESCRIPTION CART

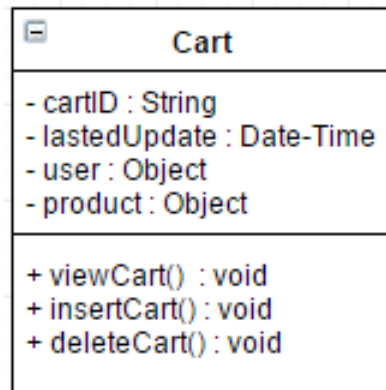
## 4.1.1. PROCESSING NARRATIVE (PSPEC) FOR CART

## Cart

ทำหน้าที่ในการเก็บสินค้าที่ลูกค้าคิดว่าจะเลือกซื้อ โดยจะเก็บได้แค่ 7 วัน จนกว่าลูกค้าจะชำระเงิน หากลูกค้าไม่ได้ชำระเงิน ระบบจะลบรายการนั้นทิ้ง ในการเก็บสินค้าจะแบ่งเป็นรายการสินค้า โดย 1 รายการสามารถมีจำนวนสินค้าได้หลายชิ้น และมีการคำนวณราคาทั้งหมด พร้อมกับหักโปรโมชั่นให้อัตโนมัติ

## 4.1.2. CART PROCESSING DETAIL

## 4.1.2.1. Design Class hierarchy for cart



#### 4.1.2.2. Restrictions/limitations for cart

หากกด Check Out แล้ว จะไม่สามารถแก้ไขรายการสินค้าได้

#### 4.1.2.3. Performance issues for component 4

ไม่มี

#### 4.1.2.4. Design constraints for component 4

ไม่มี

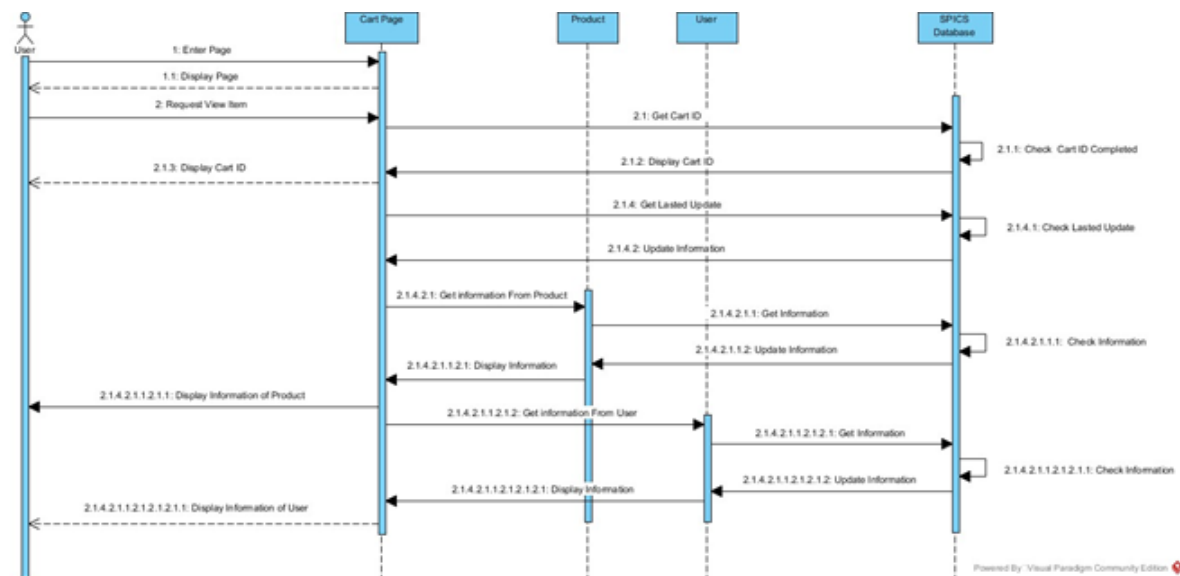
#### 4.1.2.5. Processing detail for each operation of component 4

##### 4.1.2.5.1. Processing narrative (PSPEC) for each operation

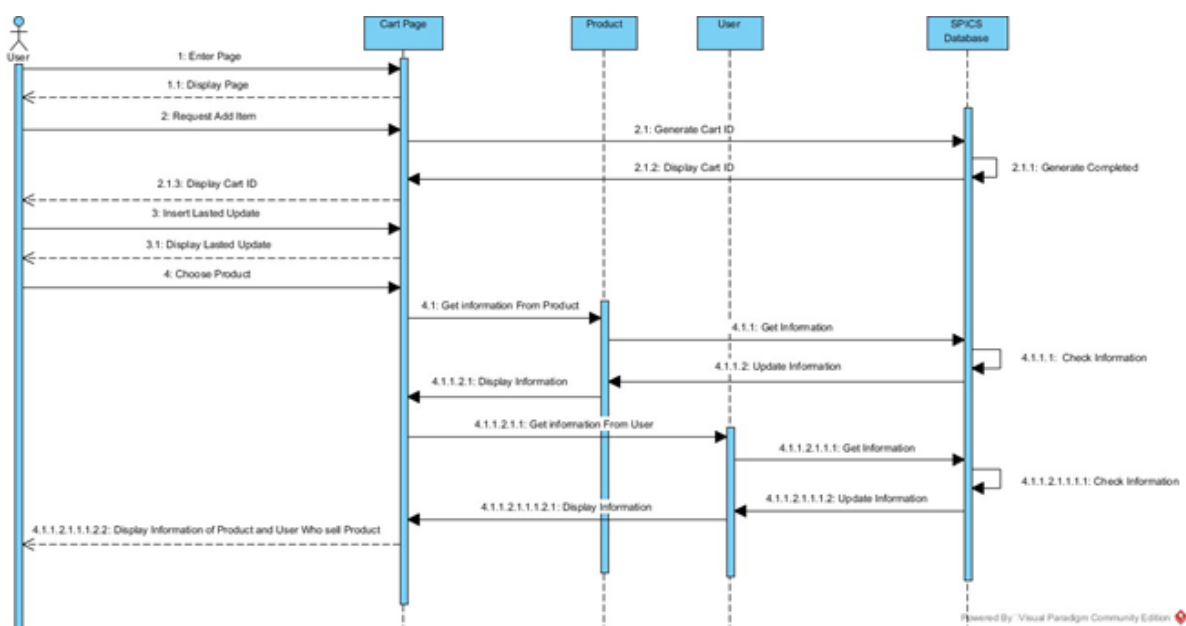
Operation	Explanation
+ viewCart()	ฟังก์ชันในการแสดง Cart ซึ่งภายในจะมี Operation get เพื่อทำการแสดงข้อมูลใน Attribute ต่างๆ ของ Class Cart ให้แสดงออกมาทางหน้า Cart page
+ insertCart()	ฟังก์ชันในการเพิ่ม Cart โดยภายในฟังก์ชันจะมี Operation get และ set เพื่อเก็บค่าต่างๆ ให้แก่ cartID, lastedUpdate, user และ product ซึ่งเมื่อมีการเรียกใช้ จะมีการเพิ่มรายการ Cart ขึ้น
+ deleteCart()	ฟังก์ชันในการลบ Cart ซึ่งภายในจะมี Operation delete เพื่อทำการลบข้อมูลใน Attribute ต่างๆ ของ Class Cart ก่อนที่จะชำระเงิน

## 4.1.2.5.2. Algorithmic model (e.g., PDL) for each operation

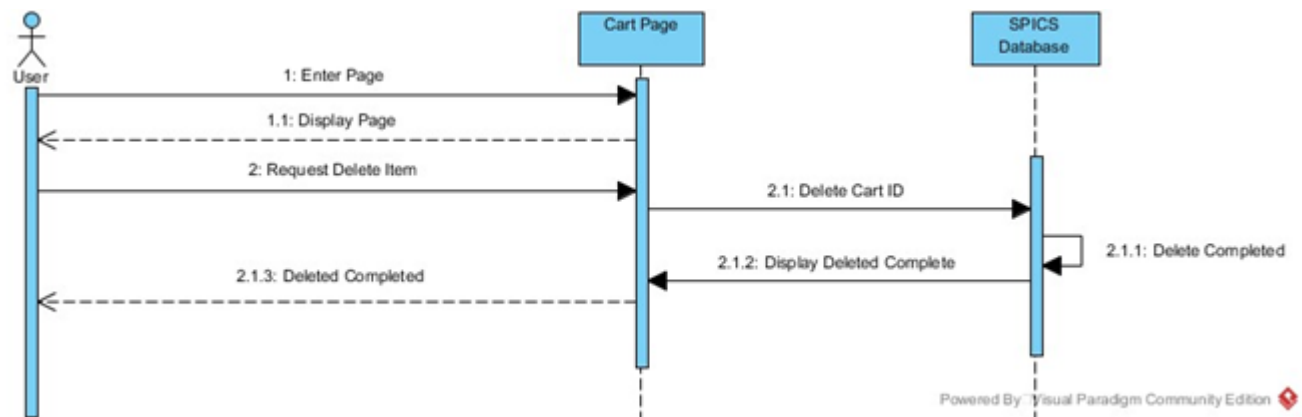
+ viewCart()



+ insertCart()



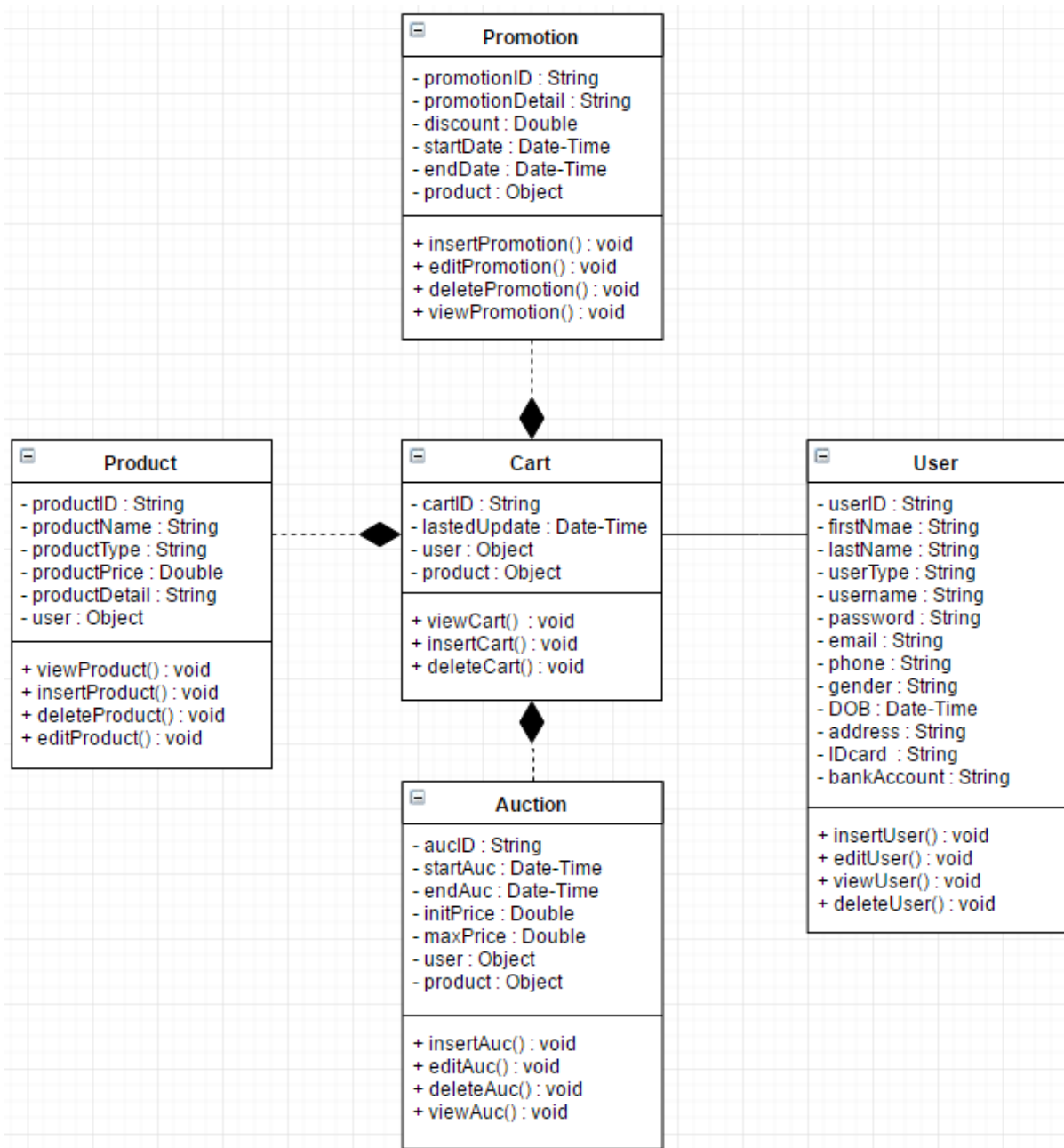
+ deleteCart()



## 4.1.3. COMPONENT 4 TEST POINTS LIST AND DESCRIPTION

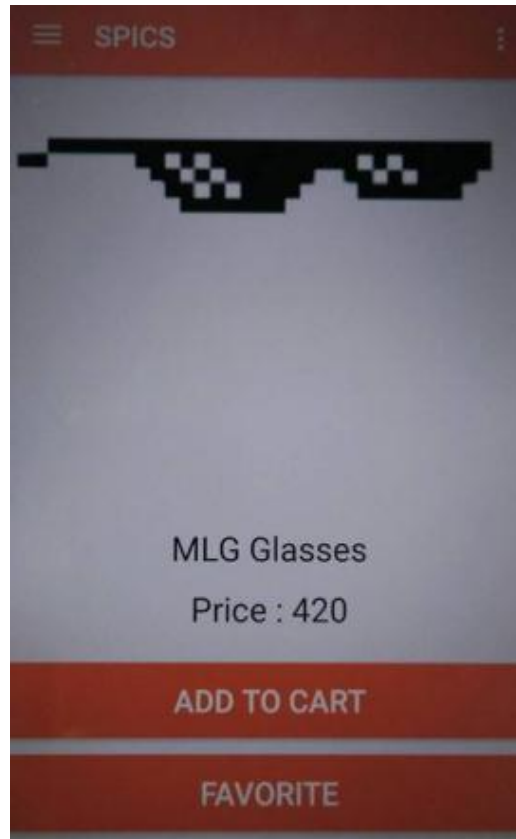
No.	Description Testing	Expected Result
1	กดปุ่ม Add to Cart ที่สินค้าชิ้น	รายการสินค้าชิ้น จะต้องแสดงในหน้า Cart
2	กดที่หน้า Cart เพื่อแสดงรายการสินค้าต่างๆ	แสดงสินค้าต่างๆ ที่ลูกค้าได้กดเพิ่มเข้ามา โดยดึงข้อมูลมาจาก Database
3	กดเพิ่มจำนวนสินค้า	ระบบจะต้องคำนวณราคาได้อย่างถูกต้อง
4	กดเพิ่มสินค้าที่ร่วม Promotion	ระบบจะต้องหัก Promotion และคำนวณราคาได้อย่างถูกต้อง
5	กดปุ่ม Delete เพื่อลบสินค้าชิ้นออก	ลบสินค้าชิ้นออกจากหน้า Cart
6	กดปุ่ม Check Out เพื่อยืนยันการสั่งซื้อ	ระบบต้องทำการยืนยันการสั่งซื้อ และนำไปสู่ขั้นตอนถัดไป

## 4.1.4. CART DYNAMIC BEHAVIOR



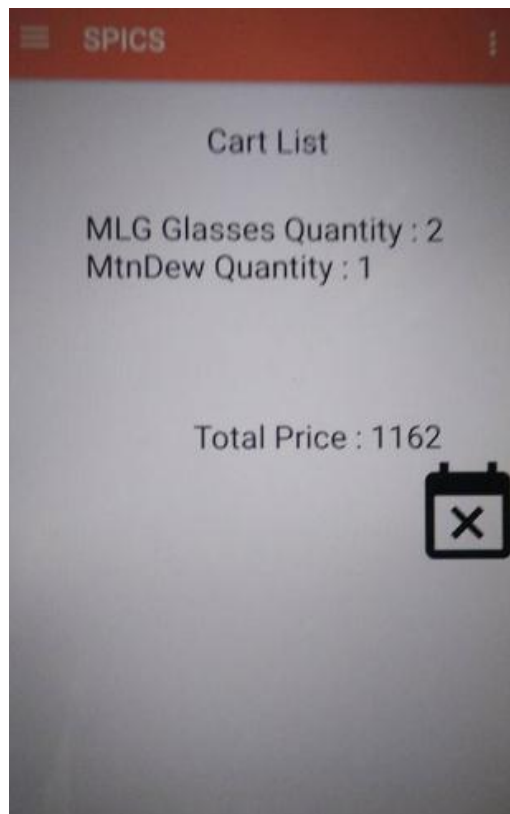
#### 4.1.1. CART INTERFACE

Shopping page



เมื่อกดปุ่ม add to cart ที่สินค้าแล้ว จะไปแสดงที่หน้า Cart

Cart page



หน้า Cart จะแสดงรายการที่ได้ทำการเลือกซื้อสินค้า และราคารวม

## Source code with code comments

Source code
<pre>package com.example.sikiryl.spics;  import android.os.Bundle; import android.support.design.widget.FloatingActionButton; import android.support.design.widget.Snackbar; 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.widget.TextView;  import java.util.List;  public class MainActivity extends AppCompatActivity     implements NavigationView.OnNavigationItemSelectedListener {      NavigationView navigationView = null;     Toolbar toolbar = null;      @Override     protected void onCreate(Bundle savedInstanceState) {         super.onCreate(savedInstanceState);         setContentView(R.layout.activity_main);         //Set the fragment initially         HomeFragment fragment = new HomeFragment();         android.support.v4.app.FragmentTransaction fragmentTransaction =</pre>



```
        getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.fragment_container, fragment);
        fragmentTransaction.commit();

        toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        /* Test Snack Bar */
        //      FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
        //      fab.setOnClickListener(new View.OnClickListener() {
        //          @Override
        //          public void onClick(View view) {
        //              Snackbar.make(view, "Replace with your own action",
        //              Snackbar.LENGTH_LONG)
        //                  .setAction("Action", null).show();
        //          }
        //      });

        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) findViewById(R.id.nav_view);

        //How to change elements in the header programatically
        View headerView = navigationView.getHeaderView(0);
        TextView emailText = (TextView) headerView.findViewById(R.id.email);
        emailText.setText("newemail@email.com");

        navigationView.setNavigationItemSelectedListener(this);
```

```
}

@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;
}

@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);
}
```

```
@SuppressWarnings("StatementWithEmptyBody")
@Override
public boolean onNavigationItemSelected(MenuItem item) {
    // Handle navigation view item clicks here.
    int id = item.getItemId();

    if (id == R.id.nav_home) {
        //Set the fragment initially
        HomeFragment fragment = new HomeFragment();
        android.support.v4.app.FragmentTransaction fragmentTransaction =
            getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.fragment_container, fragment);
        fragmentTransaction.commit();
        // Handle the camera action
    } else if (id == R.id.nav_shopping) {
        //Set the fragment initially
        ShoppingFragment fragment = new ShoppingFragment();
        android.support.v4.app.FragmentTransaction fragmentTransaction =
            getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.fragment_container, fragment);
        fragmentTransaction.commit();

    } else if (id == R.id.nav_cart) {
        //Set the fragment initially
        CartFragment fragment = new CartFragment();
        android.support.v4.app.FragmentTransaction fragmentTransaction =
            getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.fragment_container, fragment);
        fragmentTransaction.commit();
    }
    /* For extended app in the future */
    else if (id == R.id.nav_manage) {
```

```

        } else if (id == R.id.nav_share) {

        } else if (id == R.id.nav_send) {

        }

        DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
        drawer.closeDrawer(GravityCompat.START);
        return true;
    }
}

```

#### comments

**MainActivity.java** เป็นโค้ดที่รวมทุกฟังก์ชันที่จะแสดงในแอป

#### Source code

```

package com.example.sikiryL.spics;

import android.database.Cursor;
import android.content.ContentValues;
import android.database.sqlite.SQLiteDatabase;
import android.net.Uri;
import android.os.Bundle;
import android.support.design.widget.Snackbar;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

```

```
/**
 * A simple {@link Fragment} subclass.
 */
public class ShoppingFragment extends Fragment {

    public ShoppingFragment() {
        // Required empty public constructor
    }

    private DBHelper helper;
    int[] productID = new int[5];
    int[] productPrice = new int[5];
    String[] productName = new String[5];
    String[] productImage = new String[5];

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        /* Pull Information From DB to GUI */
        // Inflate the layout for this fragment
        View rootView = inflater.inflate(R.layout.fragment_shopping, container, false);
        helper = new DBHelper(getActivity());
        try{
            Cursor cursor = getProduct();
            listProduct(cursor);

            TextView outProductName1 = (TextView) rootView.findViewById(R.id.txtName);
            outProductName1.setText(productName[0]);
            TextView outProductPrice1 = (TextView) rootView.findViewById(R.id.txtPrice);
            outProductPrice1.setText("Price : " + Integer.toString(productPrice[0]));
            ImageView outProductImage1 = (ImageView)
rootView.findViewById(R.id.img_shopping);
```

```
        outProductImage1.setImageURI(Uri.parse(productImage[0]));

        TextView outProductName2 = (TextView) rootView.findViewById(R.id.txtName2);
        outProductName2.setText(productName[1]);
        TextView outProductPrice2 = (TextView) rootView.findViewById(R.id.txtPrice2);
        outProductPrice2.setText("Price : " + Integer.toString(productPrice[1]));
        ImageView outProductImage2 = (ImageView)
rootView.findViewById(R.id.img_shopping2);
        outProductImage2.setImageURI(Uri.parse(productImage[1]));

        TextView outProductName3 = (TextView) rootView.findViewById(R.id.txtName3);
        outProductName3.setText(productName[2]);
        TextView outProductPrice3 = (TextView) rootView.findViewById(R.id.txtPrice3);
        outProductPrice3.setText("Price : " + Integer.toString(productPrice[2]));
        ImageView outProductImage3 = (ImageView)
rootView.findViewById(R.id.img_shopping3);
        outProductImage3.setImageURI(Uri.parse(productImage[2]));

        TextView outProductName4 = (TextView) rootView.findViewById(R.id.txtName4);
        outProductName4.setText(productName[3]);
        TextView outProductPrice4 = (TextView) rootView.findViewById(R.id.txtPrice4);
        outProductPrice4.setText("Price : " + Integer.toString(productPrice[3]));
        ImageView outProductImage4 = (ImageView)
rootView.findViewById(R.id.img_shopping4);
        outProductImage4.setImageURI(Uri.parse(productImage[3]));

        TextView outProductName5 = (TextView) rootView.findViewById(R.id.txtName5);
        outProductName5.setText(productName[4]);
        TextView outProductPrice5 = (TextView) rootView.findViewById(R.id.txtPrice5);
        outProductPrice5.setText("Price : " + Integer.toString(productPrice[4]));
        ImageView outProductImage5 = (ImageView)
rootView.findViewById(R.id.img_shopping5);
        outProductImage5.setImageURI(Uri.parse(productImage[4]));
```

```
}  
finally{  
    helper.close();  
}  
  
Button addProduct1 = (Button) rootView.findViewById(R.id.btn_add);  
addProduct1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        try{  
            addToCart(productID[0]);  
            Snackbar.make(v, "Cart added", Snackbar.LENGTH_LONG)  
                .setAction("Action", null).show();  
        }  
        finally{  
            helper.close();  
        }  
    }  
});  
  
Button addProduct2 = (Button) rootView.findViewById(R.id.btn_add2);  
addProduct2.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        try{  
            addToCart(productID[1]);  
            Snackbar.make(v, "Cart added", Snackbar.LENGTH_LONG)  
                .setAction("Action", null).show();  
        }  
        finally{  
            helper.close();  
        }  
    }  
});  
  
Button addProduct3 = (Button) rootView.findViewById(R.id.btn_add3);
```

```
addProduct3.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        try{  
            addToCart(productID[2]);  
            Snackbar.make(v, "Cart added", Snackbar.LENGTH_LONG)  
                .setAction("Action", null).show();  
        }  
        finally{  
            helper.close();  
        }  
    }  
});  
Button addProduct4 = (Button) rootView.findViewById(R.id.btn_add4);  
addProduct4.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        try{  
            addToCart(productID[3]);  
            Snackbar.make(v, "Cart added", Snackbar.LENGTH_LONG)  
                .setAction("Action", null).show();  
        }  
        finally{  
            helper.close();  
        }  
    }  
});  
Button addProduct5 = (Button) rootView.findViewById(R.id.btn_add5);  
addProduct5.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        try{  
            addToCart(productID[4]);
```



```
        Snackbar.make(v, "Cart added", Snackbar.LENGTH_LONG)
            .setAction("Action", null).show();
    }
    finally{
        helper.close();
    }
}
});
return rootView;
}

/* Add to Cart Method*/
private void addToCart(int pID) {
    SQLiteDatabase db = helper.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("productID",Integer.toString(pID));
    db.insertOrThrow("cart",null,values);
}

/* Get Product */
private Cursor getProduct(){
    SQLiteDatabase db = helper.getReadableDatabase();
    Cursor cursor = db.rawQuery("SELECT * FROM product ORDER BY productID;",null);
    return cursor;
}

/* Get Product List */
private void listProduct(Cursor cursor){
    int i=0;
    while (cursor.moveToNext()){
        productID[i] = cursor.getInt(0);
        productName[i] = cursor.getString(1);
        productPrice[i] = cursor.getInt(2);
    }
}
```

```

        productImage[i] = cursor.getString(3);
        i++;
    }
}
}

```

#### comments

**ShoppingFragment.java** เป็นโค้ดที่จะแสดงสินค้าทั้งหมด ซึ่งจะมีให้กดปุ่ม add to cart

#### Source code

```

package com.example.sikiryl.spics;

import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.net.Uri;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentTransaction;
import android.support.v7.widget.ThemedSpinnerAdapter;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;

/**
 * A simple {@link Fragment} subclass.
 * Activities that contain this fragment must implement the
 * {@link CartFragment.OnFragmentInteractionListener} interface
 * to handle interaction events.
 * Use the {@link CartFragment#newInstance} factory method to
 * create an instance of this fragment.

```

```
*/  
public class CartFragment extends Fragment {  
    /* // TODO: Rename parameter arguments, choose names that match  
        // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER  
        private static final String ARG_PARAM1 = "param1";  
        private static final String ARG_PARAM2 = "param2";  
        // TODO: Rename and change types of parameters  
        private String mParam1;  
        private String mParam2;  
        private OnFragmentInteractionListener mListener;*/  
        private DBHelper helper;  
        private int totalPrice = 0;  
  
    public CartFragment() {  
        // Required empty public constructor  
    }  
  
    /**  
     * Use this factory method to create a new instance of  
     * this fragment using the provided parameters.  
     *  
     * @param param1 Parameter 1.  
     * @param param2 Parameter 2.  
     * @return A new instance of fragment CartFragment.  
     */  
    // TODO: Rename and change types and number of parameters  
    /* public static CartFragment newInstance(String param1, String param2) {  
        CartFragment fragment = new CartFragment();  
        Bundle args = new Bundle();  
        args.putString(ARG_PARAM1, param1);  
        args.putString(ARG_PARAM2, param2);  
        fragment.setArguments(args);  
        return fragment;  
    }  
}
```

```
    }*/

    /* @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }*/

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {

        /* Get Information From DB and Calculate Total Price */
        // Inflate the layout for this fragment
        final View rootView = inflater.inflate(R.layout.fragment_cart, container, false);
        helper = new DBHelper(getActivity());
        totalPrice = 0;
        try{
            Cursor cursor = getCart();
            StringBuilder fullList = showCartAndCalculate(cursor);
            TextView outputFullList = (TextView) rootView.findViewById(R.id.txtFullList);
            outputFullList.setText(fullList);
            TextView outputTotalPrice = (TextView) rootView.findViewById(R.id.txtTotalPrice);
            outputTotalPrice.setText("Total Price : "+Integer.toString(totalPrice));
        }
        finally{
            helper.close();
        }

        Button btn = (Button) rootView.findViewById(R.id.flushNow);
        btn.setOnClickListener(new View.OnClickListener() {

            @Override
```

```
        public void onClick(View v) {
            SQLiteDatabase db = helper.getReadableDatabase();
            db.execSQL("DELETE FROM cart;");
            TextView outputTotalPrice = (TextView)
rootView.findViewById(R.id.txtTotalPrice);
            Fragment frg = null;
            frg = getFragmentManager().findFragmentById(R.id.fragment_container);
            final FragmentTransaction ft = getFragmentManager().beginTransaction();
            ft.detach(frg);
            ft.attach(frg);
            ft.commit();
        }
    });
    return rootView;
}

private Cursor getCart(){
    SQLiteDatabase db = helper.getReadableDatabase();
    Cursor cursor = db.rawQuery("SELECT productID,COUNT(productID) AS quantity FROM
cart GROUP BY productID ORDER BY productID;",null);
    return cursor;
}

private StringBuilder showCartAndCalculate(Cursor cursor){
    StringBuilder builder = new StringBuilder();
    SQLiteDatabase db = helper.getReadableDatabase();
    while (cursor.moveToNext()) {
        int productID = cursor.getInt(0);
        int quantity = cursor.getInt(1);
        Cursor cursor2 = db.rawQuery("SELECT productName,productPrice FROM product
WHERE productID = " + Integer.toString(productID) +";", null);
        cursor2.moveToNext();
        String productName = cursor2.getString(0);
        int productPrice = cursor2.getInt(1);
        totalPrice = totalPrice + (productPrice * quantity);
    }
}
```

```

        builder.append(productName).append(" Quantity :
    ").append(quantity).append("\n");
    }
    return builder;
}

// TODO: Rename method, update argument and hook method into UI event
/* public void onPressed(Uri uri) {
    if (mListener != null) {
        mListener.onFragmentInteraction(uri);
    }
}

@Override
public void onAttach(Context context) {
    super.onAttach(context);
    if (context instanceof OnFragmentInteractionListener) {
        mListener = (OnFragmentInteractionListener) context;
    } else {
        throw new RuntimeException(context.toString()
            + " must implement OnFragmentInteractionListener");
    }
}

@Override
public void onDetach() {
    super.onDetach();
    mListener = null;
}

*/
/**
 * This interface must be implemented by activities that contain this
 * fragment to allow an interaction in this fragment to be communicated
 * to the activity and potentially other fragments contained in that
 * activity.
 * <p/>

```

```
* See the Android Training lesson <a href=
* "http://developer.android.com/training/basics/fragments/communicating.html"
* >Communicating with Other Fragments</a> for more information.
*/
/*public interface OnFragmentInteractionListener {
    // TODO: Update argument type and name
    void onFragmentInteraction(Uri uri);
}*/
}
```

#### comments

**CartFragment.java** เป็นโค้ดที่จะแสดงสินค้าที่อยู่ในหน้า cart ซึ่งจะทำการคำนวณ total price และลบรายการสินค้าทั้งหมดได้

## Test cases and results

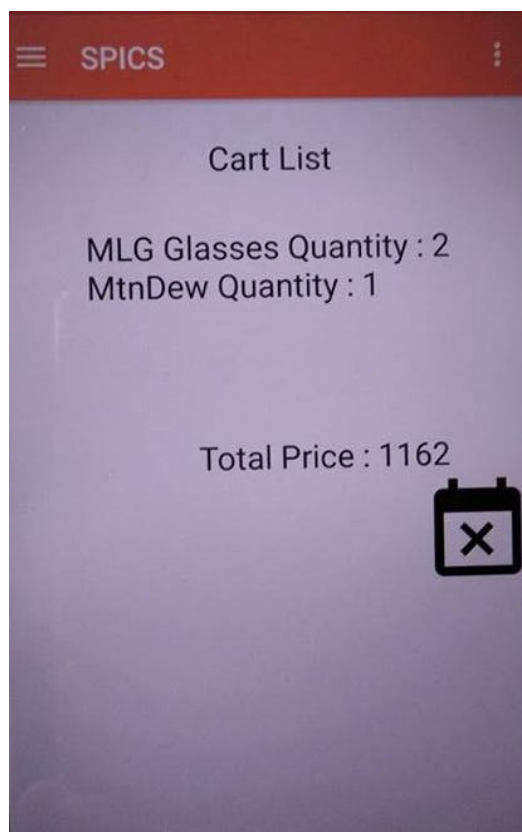
Test ID	Test description	Step	Pre-condition	Post-condition	Expected result	Actual result	Status
1	เลือกซื้อสินค้า และสามารถแสดงราคารวม	1. กดเลือกเมนู ( ≡ ) 2. กดเลือก Shopping 3. กดเลือก MLG Glasses ที่ปุ่ม ADD TO CART 1 ครั้ง 4. กดเลือก MtnDew ที่ปุ่ม ADD TO CART 1 ครั้ง 5. กดเลือก MLG Glasses ที่ปุ่ม ADD TO CART 1 ครั้ง 6. กดเลือกเมนู ( ≡ ) 7. กดเลือก Cart	เปิดแอปพลิเคชัน SPICS	-	ในหน้า Cart จะแสดง Cart List โดยจะแสดงรายการที่สั่งซื้อ จำนวนสินค้า และราคารวม	ในหน้า Cart จะแสดง Cart List โดยจะแสดงรายการที่สั่งซื้อ จำนวนสินค้า และราคารวม	ผ่าน
2	เลือกซื้อสินค้า และสามารถลบข้อมูลที่สั่งซื้อล่าสุดได้	1. กดเลือกเมนู ( ≡ ) 2. กดเลือก Shopping 3. กดเลือก Air Horn ที่ปุ่ม ADD TO CART 1 ครั้ง 4. กดเลือก Doritos ที่ปุ่ม	เปิดแอปพลิเคชัน SPICS	-	รายการสินค้า ถูกลบออก ราคารวม จะต้องแสดง เป็น 0 บาท	รายการสินค้า ถูกลบออก ราคารวม จะต้องแสดง เป็น 0 บาท	ผ่าน



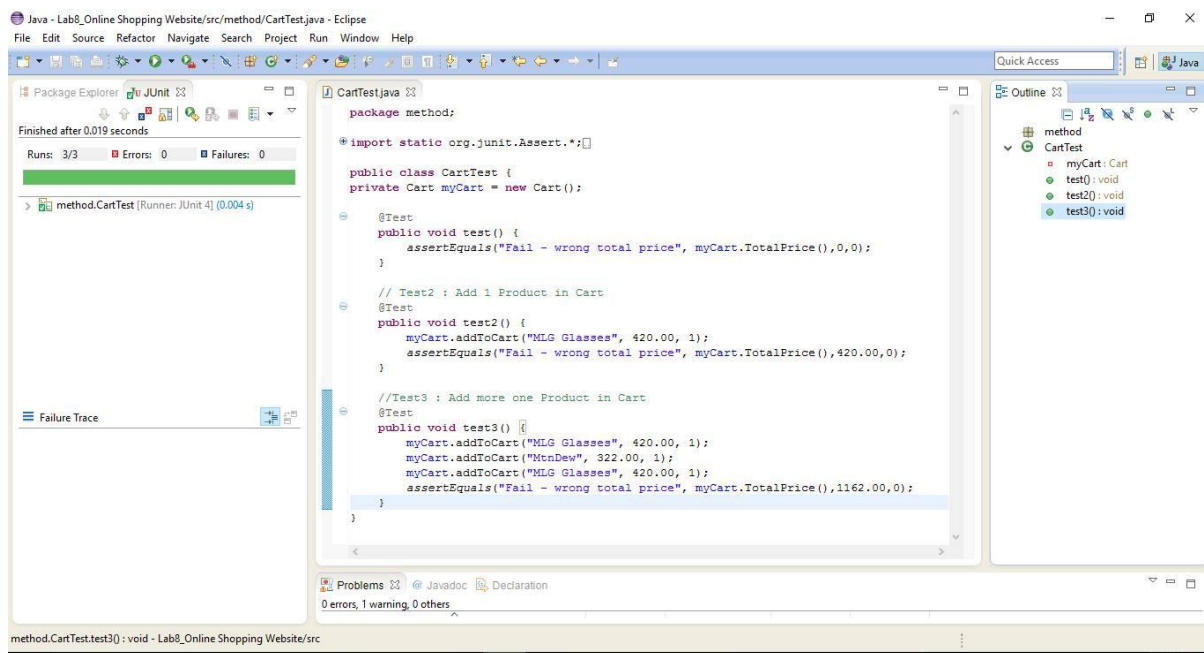
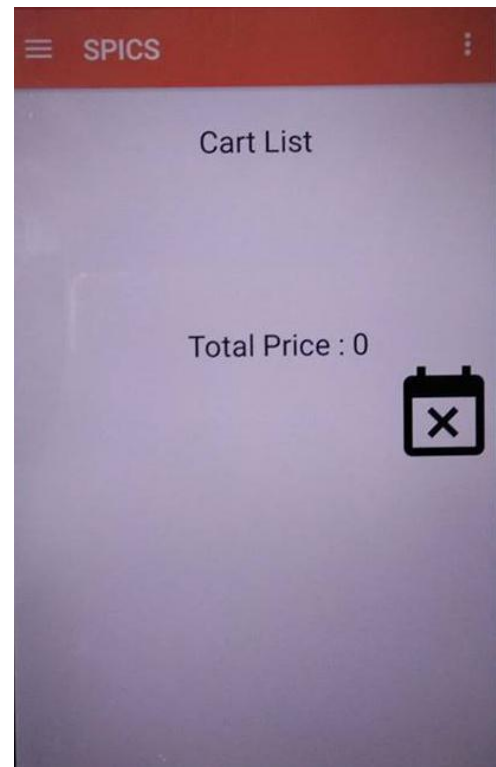
Test ID	Test description	Step	Pre-condition	Post-condition	Expected result	Actual result	Status
		ADD TO CART 1 ครั้ง 5. กดเลือก Salt ที่ปุ่ม ADD TO CART 1 ครั้ง 6. กดเลือกเมนู ( ≡ ) 7. กดเลือก Cart 8. กด X					

**Result:**

Test ID 1 :



Test ID 2 :



### Instructions to run your code.

- สามารถ run code ได้ โดยการ build แอปพลิเคชัน SPICS ผ่านโปรแกรม Android Studio ลงบนโทรศัพท์มือถือระบบปฏิบัติการ android
- function ที่สามารถใช้งานได้คือ Shopping และ Cart ซึ่งสามารถกดเลือกได้จากเมนูด้านข้าง
  - Shopping : หน้าเลือกซื้อสินค้า
  - Cart : หน้าแสดงรายการสินค้าที่เลือกซื้อ และราคารวม

### The link of GitHub repository.

- <https://github.com/Sikiry/SPICS-Imp>