

GUILIN UNIVERSITY OF TECHNOLOGY

课程设计与实习报告

评 语:

2019 年 6 月 15 日

目录

- 1. 需求分析（撰写人：徐仕成）1
 - 1.1 项目所能达到的功能1
 - 1.2 测试数据1
 - 1.2.1 注册测试数据.....1
 - 1.2.2 登录测试数据.....3
 - 1.2.3 部分截图:3
- 2. 概要设计（撰写人：曾鑫）4
- 3. 详细设计（撰写人：刘笑枫、曾鑫）5
 - 3.1 ER 图.....5
 - 3.2 功能模块的关系图6
 - 3.3 流程图7
 - 3.4 核心代码8
- 4. 调试分析（撰写人：曾鑫、刘笑枫、徐仕成） 15
 - 4.1 遇到的问题及解决 15
 - 4.2 设计与实现的讨论分析..... 17
 - 4.3 核心功能测试截图 17
 - 4.4 经验体会 18
- 5. 用户使用说明（撰写人：徐仕成） 19
 - 5.1 操作步骤 19
- 6. 软件测试结果（撰写人：徐仕成） 20
 - 6.1 注册测试数据及结果..... 20
 - 6.2 登录测试数据及结果..... 22
 - 6.3 地点搜索测试数据及结果 23
 - 6.4 其他测试结果 24
- 7. 心得体会（撰写人：刘笑枫、曾鑫、徐仕成） 24

1. 需求分析（撰写人：徐仕成）

本软件（鸿鹄导航，英文：Swan，后面称鸿鹄导航）主要通过使用高德地图 API 完成基本的地图显示及放大缩小、实时定位、地址搜索、规划路线、导航、搜索历史记录等功能，并自定义实现注册登录功能。并在注册登录、搜索历史记录模块实现数据持久化，同时，路线规划提供驾车、公交、步行、跨城公交四种方式的路线规划，并且会计算出打车费用，以及驾车、公交、步行所需的时间，还有详细的路线规划详情，另外，还提供了全国各城市的离线地图下载。

1.1 项目所能达到的功能

- （1）地图的显示
- （2）地图的放大缩小
- （3）实时定位
- （4）地址搜索
- （5）路线规划
- （6）驾车导航
- （7）公交路线
- （8）步行导航
- （9）跨城公交路线
- （10）搜索的历史记录（数据持久化）
- （11）登录（数据持久化）
- （12）注册（数据持久化）
- （13）离线地图下载

1.2 测试数据

由于地址搜索功能使用高德地图 API，所以在搜索内容上没有过多要求，只要是存在的地点名均可搜索。下面将测试数据主要集中在自定义的登录注册模块。

说明：登录、注册模块的信息输入均做了非空验证，以下测试数据不讨论为空的情况。

1.2.1 注册测试数据

- （1）正确的输入输出

① 第一组测试数据

正确的输入：

账号: test1

密码: test1

确认密码: test1

验证码: cmgf

(注: 验证码为图形验证码, 需要输入图形验证码中的字符, 点击图形验证码可更新, 每次的验证码均为随机)

正确的输出:

点击注册按钮后: 提示注册成功, 并自动跳转到登录页面

② 第二组测试数据

正确的输入:

账号: test2

密码: test2

确认密码: test2

验证码: c2w2

(注: 验证码为图形验证码, 需要输入图形验证码中的字符, 点击图形验证码可更新, 每次的验证码均为随机)

正确的输出:

点击注册按钮后: 提示注册成功, 并自动跳转到登录页面

(2) 错误的输入及其输出

① 第一组测试数据

错误的输入:

账号: test3

密码: test3

确认密码: tets3

验证码: ae47

(注: 验证码为图形验证码, 需要输入图形验证码中的字符, 点击图形验证码可更新, 每次的验证码均为随机)

输出结果:

点击注册按钮后: 提示两次密码不同, 仍停留在注册页面

② 第二组测试数据

错误的输入:

账号: test1

密码: test1

确认密码: test1

验证码: 2dd2

输出结果：

点击注册按钮后：提示不可重复注册，仍停留在注册页面

1.2.2 登录测试数据

(1) 正确的输入输出

① 第一组测试数据

正确的输入：

账号：test1

密码：test1

正确的输出：

点击登录按钮后：直接跳转到个人中心页面

② 第二组测试数据

正确的输入：

账号：test2

密码：test2

正确的输出：

点击登录按钮后：直接跳转到个人中心页面

(2) 错误的输入及其输出

① 第一组测试数据

错误的输入：

账号：test1

密码：tes1

输出结果：

点击登录按钮后：提示账号密码不匹配，仍停留在登录页面

② 第二组测试数据

错误的输入：

账号：tes2

密码：test2

输出结果：

点击登录按钮后：提示不存在该用户，仍停留在登录页面

1.2.3 部分截图：



图 1-1 注册成功



图 1-2 重复注册



图 1-3 登录输入



图 1-4 个人中心



图 1-5 登录失败



图 1-6 登录失败

2. 概要设计（撰写人：曾鑫）

在这个系统中，主要采用了面对对象的原则，并且采取了一系列的设计原则来降低耦合度。如采用单例模式等等。将功能封装到相关的类中，然后接口让主线程进行调用。

相关功能主要参考高德地图 API 进行设计。

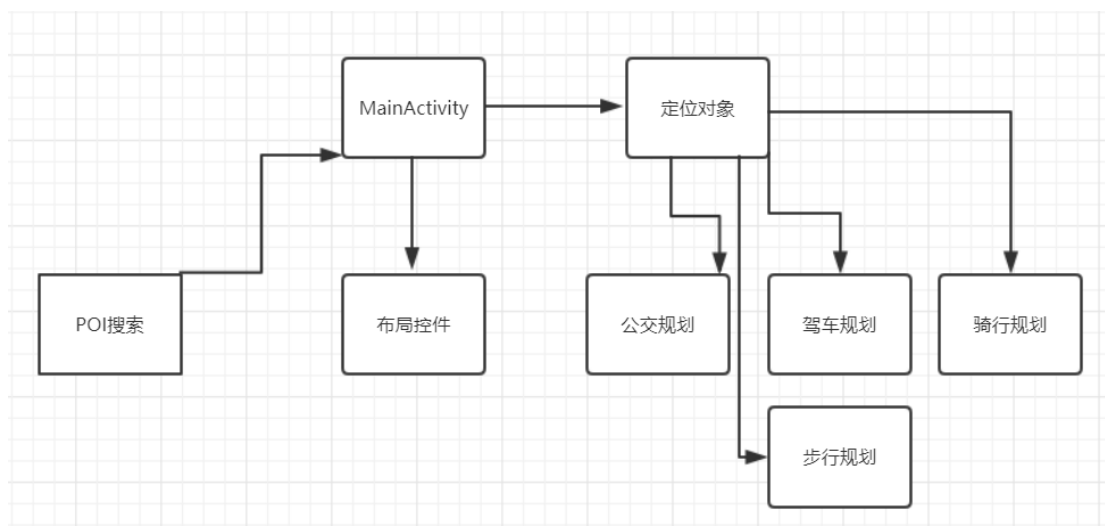


图 2-1 功能模块图

3. 详细设计（撰写人：刘笑枫、曾鑫）

3.1 ER 图

程序整体流程图：见图 3-1 整体流程图

界面流程图：见图 3-2 界面流程图

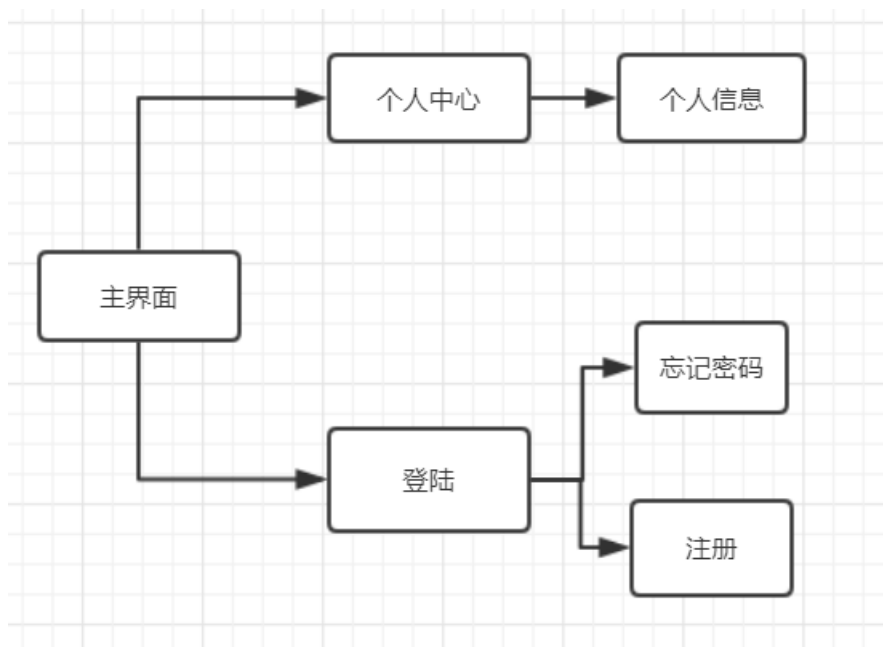


图 3-2 界面流程图

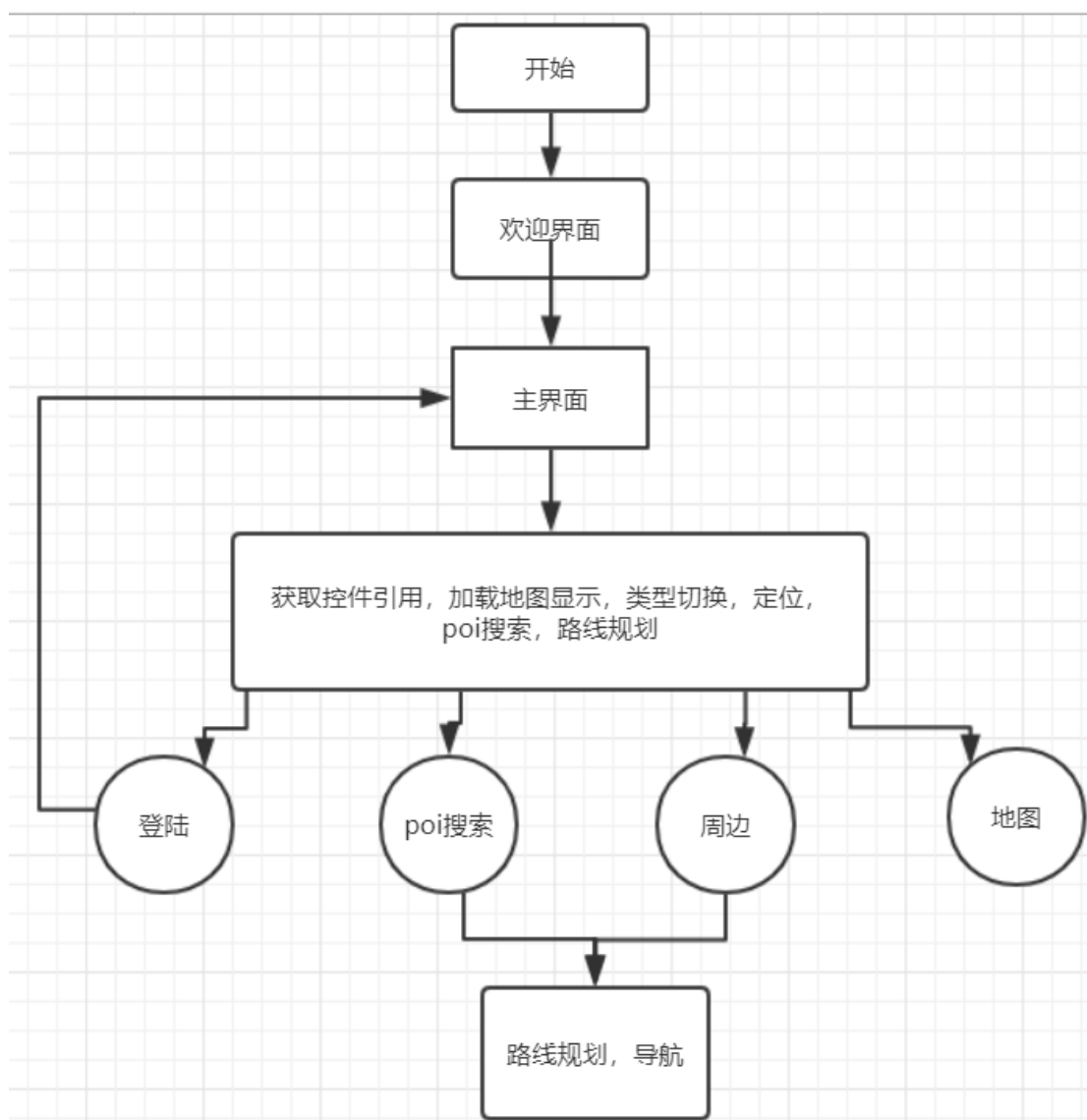


图 3-1 整体流程图

3.2 功能模块的关系图

见图 3-3 功能模块图

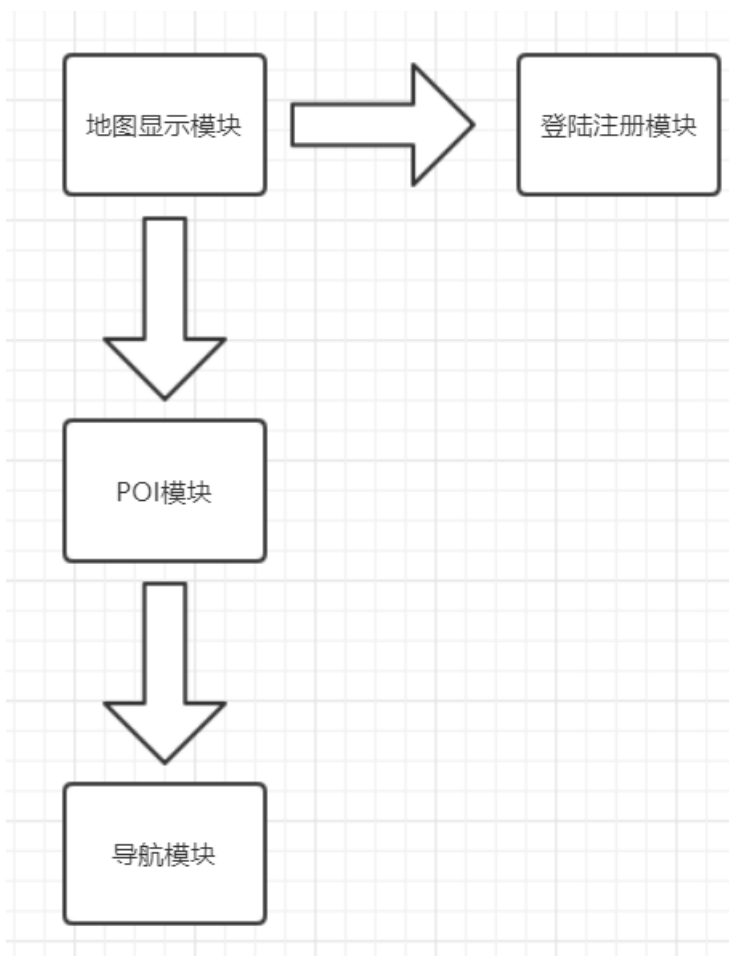


图 3-3 功能模块图

3.3 流程图

见图 3-4 程序整体流程图

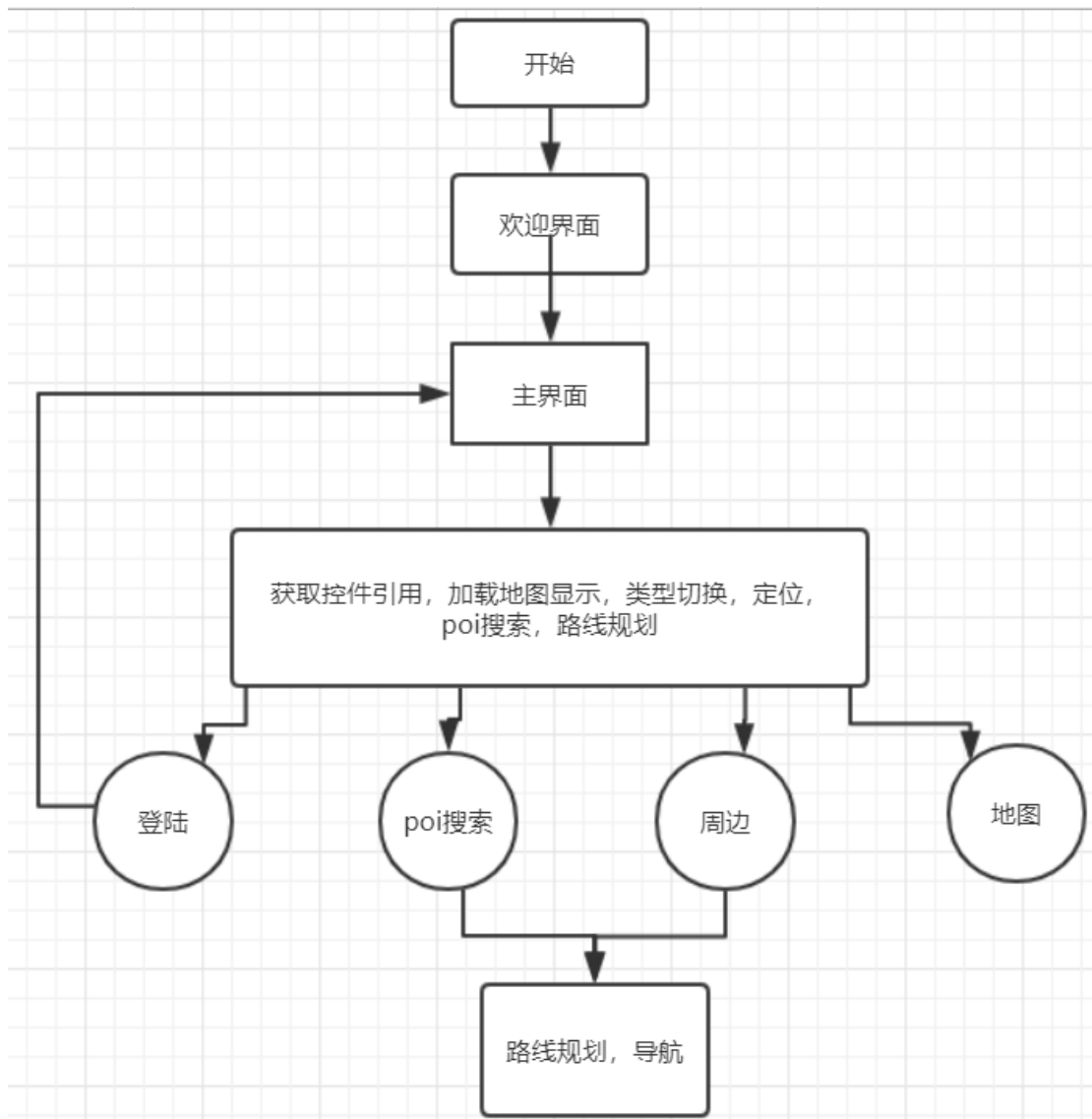


图 3-4 程序整体流程图

3.4 核心代码

//MainActivity.java 部分代码

@Override

protected void onCreate(Bundle savedInstanceState) {

 super.onCreate(savedInstanceState);

 setContentView(R.layout.activity_main);

 getSupportActionBar().hide();

 checkPermission();

 //获取地图控件引用

 mapView = findViewById(R.id.main_map);

 initTopBar();

 //在 activity 执行 onCreate 时执行 mapView.onCreate(savedInstanceState), 创建

地图

```
mapView.onCreate(savedInstanceState);
if (aMap == null) {
    aMap = mapView.getMap();
    setUpMap();
}
initMyLocation();
initSearch();
initRoute();
}
```

//SearchActivity.java 部分代码

```
public class SearchActivity extends AppCompatActivity {
    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        Intent it = getIntent();
        mCurrentCityName = it.getStringExtra("mCurrentCityName");
        latLonPoint = new
        LatLonPoint(it.getDoubleExtra("startPointLat", 25.063734), it.getDoubleExtra("startPointLon", 110.300496));
        setContentView(R.layout.activity_search);
        getSupportActionBar().hide();
        init();
    }
    private void init() {
        inputListView = findViewById(R.id.search_list);
        inputListView.setOnItemClickListener(itemClickListener);
        initTopBar();
        initSearchView();
    }
    private void initTopBar() {
        topBar = findViewById(R.id.search_top_bar);
        topBar.addLeftImageButton(R.drawable.ic_icon_back,
0).setOnClickListener(v -> finish());
        QMUIButton searchButton = new QMUIButton(this);
        topBar.addRightTextButton("搜索", searchButton.getId());
    }
    private void initSearchView() {
        searchView = findViewById(R.id.search_search_view);
        clearHistoryView = findViewById(R.id.search_list_clear);
        //修改 SearchView 的样式
        TextView searchText = (TextView) searchView.findViewById(searchView
            .getContext())
```

```

        .getResources()
        .getIdentifier("android:id/search_src_text", null, null));
searchText.setTextSize(TypedValue.COMPLEX_UNIT_SP, 14);
searchText.setTextColor(Color.parseColor("#365146"));
searchText.setHintTextColor(Color.parseColor("#9e9e9e"));
searchView.setOnQueryTextListener(queryTextListener);
//设置 SearchView 默认为展开显示
searchView.setIconified(false);
searchView.onActionViewExpanded();
searchView.setSubmitButtonEnabled(false);
initHistory();
}
private void initHistory() {
    LinkedList<String> stringList = AccountHelper.getHistorySearch(this);
    LinkedList<Tip> tempTipList = new LinkedList<Tip>();
    stringList.forEach(s -> {
        Tip tip=null;
        try {
            tip= JSON.parseObject(s,Tip.class);
        }catch (Exception e){
            //如果反序列化失败
            tip=new Tip();
            tip.setName(s);
            tip.setTypeCode("o(′ □ ′)o");
        }
        tempTipList.add(tip);
    });
    inputTipsAdapter = new InputTipsAdapter(
        getApplicationContext(),
        tempTipList);
    inputListView.setAdapter(inputTipsAdapter);
    inputTipsAdapter.notifyDataSetChanged();
    inputListView.setOnItemClickListener(itemClickListener);
    clearHistoryView.setOnClickListener(v -> {
        inputListView.setAdapter(null);
        AccountHelper.clearHistorySearch(this);
    });
}
    SearchView.OnQueryTextListener      queryTextListener      =      new
SearchView.OnQueryTextListener() {
    @Override
    public boolean onQueryTextSubmit(String query) {
        Intent intent = new Intent();
        intent.putExtra("KeyWord", query);

```

```

        setResult(MainActivity.RESULT_CODE_KEYWORDS, intent);
        AccountHelper.writeHistorySearch(SearchActivity.this, query);
        finish();
        return false;
    }

    @Override
    public boolean onQueryTextChanged(String newText) {
        Log.i("query 触发了", newText);
        if (!IsEmptyOrNullString(newText)) {
            InputtipsQuery inputQuery = new InputtipsQuery(newText,
mCurrentCityName);
            inputQuery.setLocation(latLonPoint);
            Inputtips inputTips = new Inputtips(SearchActivity.this,
Inputtips(SearchActivity.this, inputQuery);
            inputTips.setInputtipsListener(inputTipsListener);
            inputTips.requestInputtipsAsync();
        } else {
            if (inputTipsAdapter != null && currentTipList != null) {
                currentTipList.clear();
                inputTipsAdapter.notifyDataSetChanged();
            }
        }
        return false;
    }
};

Inputtips.InputtipsListener inputTipsListener = new
Inputtips.InputtipsListener() {
    @Override
    public void onGetInputtips(List<Tip> tipList, int rCode) {
        if (rCode == 1000) { // 正确返回
            currentTipList = tipList;
            List<String> listString = new ArrayList<String>();
            for (int i = 0; i < tipList.size(); i++) {
                listString.add(tipList.get(i).getName());
            }
            inputTipsAdapter = new InputTipsAdapter(
                getApplicationContext(),
                currentTipList);
            inputListView.setAdapter(inputTipsAdapter);
            inputTipsAdapter.notifyDataSetChanged();
        } else {
            ToastUtil.showerror(SearchActivity.this, rCode);
        }
    }
}

```

```

    };
    AdapterView.OnItemClickListener itemClickListener = new
AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> adapterView, View view, int
position, long id) {
        if (currentTipList != null) {
            Tip tip = (Tip) adapterView.getItemAtPosition(position);
            Intent intent = new Intent();
            intent.putExtra("ExtraTip", tip);
            AccountHelper.writeHistorySearch(SearchActivity.this,
JSON.toJSONString(tip));

            setResult(MainActivity.RESULT_CODE_INPUT_TIPS, intent);
            finish();
        } else {
            //在初始化历史纪录时
            Tip tip = (Tip) adapterView.getItemAtPosition(position);
            if (!tip.getTypeCode().equals("o(′ □ ′)o")) {
                Intent intent = new Intent();
                intent.putExtra("ExtraTip", tip);
                AccountHelper.writeHistorySearch(SearchActivity.this,
tip.getName());

                setResult(MainActivity.RESULT_CODE_INPUT_TIPS, intent);
                finish();
            } else {
                Intent intent = new Intent();
                intent.putExtra("KeyWord", tip.getName());
                setResult(MainActivity.RESULT_CODE_KEYWORDS, intent);
                AccountHelper.writeHistorySearch(SearchActivity.this,
tip.getName());

                finish();
            }
        }
    }
};
}

```

//RouteActivity.java 部分代码

```

@Override
protected void onCreate(Bundle bundle) {
    super.onCreate(bundle);
    setContentView(R.layout.route_activity);
    mContext = this.getApplicationContext();
    mapView = (MapView) findViewById(R.id.route_map);
}

```



```

        mapView.onCreate(bundle); // 此方法必须重写
        init();
//        getIntentData();
        Intent it = getIntent();
        mStartPoint = new
        LatLonPoint(it.getDoubleExtra("startPointLat", 39.942295), it.getDoubleExtra("startPointLon", 116.335891));
        mEndPoint = new
        LatLonPoint(it.getDoubleExtra("endPointLat", 39.942295), it.getDoubleExtra("endPointLon", 116.335891));
        mCurrentCityName = it.getStringExtra("mCurrentCityName");
        mStartPoint_bus = mStartPoint;
        mEndPoint_bus = mEndPoint;
        setfromandtoMarker();
    }
    @Override
    public void onDriveRouteSearched(DriveRouteResult result, int errorCode) {
        dismissProgressDialog();
        aMap.clear(); // 清理地图上的所有覆盖物
        if (errorCode == AMapException.CODE_AMAP_SUCCESS) {
            if (result != null && result.getPaths() != null) {
                if (result.getPaths().size() > 0) {
                    mDriveRouteResult = result;
                    final DrivePath drivePath = mDriveRouteResult.getPaths().get(0);
                    if (drivePath == null) {
                        return;
                    }
                    DrivingRouteOverlay drivingRouteOverlay = new
                    DrivingRouteOverlay(
                        mContext, aMap, drivePath,
                        mDriveRouteResult.getStartPos(),
                        mDriveRouteResult.getTargetPos(), null);
                    drivingRouteOverlay.setNodeIconVisibility(false); // 设置节点
                    marker 是否显示
                    drivingRouteOverlay.setIsColorfulline(true); // 是否用颜色展
                    示交通拥堵情况, 默认 true
                    drivingRouteOverlay.removeFromMap();
                    drivingRouteOverlay.addToMap();
                    drivingRouteOverlay.zoomToSpan();
                    mBottomLayout.setVisibility(View.VISIBLE);
                    int dis = (int) drivePath.getDistance();
                    int dur = (int) drivePath.getDuration();
                    String des =

```

```

AMapUtil.getFriendlyTime(dur)+"("+AMapUtil.getFriendlyLength(dis)+")";
        mRotueTimeDes.setText(des);
        mRouteDetailDes.setVisibility(View.VISIBLE);
        int taxiCost = (int) mDriveRouteResult.getTaxiCost();
        mRouteDetailDes.setText("打车约"+taxiCost+"元");
        mBottomLayout.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(mContext,
                    DriveRouteDetailActivity.class);
                intent.putExtra("drive_path", drivePath);
                intent.putExtra("drive_result",
                    mDriveRouteResult);
                startActivity(intent);
            }
        });
    } else if (result != null && result.getPaths() == null) {
        Toast.makeText(RouteActivity.this, R.string.poi_no_result, Toast.LENGTH_SHORT)
            .show();
    }
    } else {
        Toast.makeText(RouteActivity.this, R.string.poi_no_result, Toast.LENGTH_SHORT)
            .show();
    }
    } else {
        ToastUtil.showerror(this.getApplicationContext(), errorCode);
    }
}

```

//HttpUtil.java 部分代码

```

public static Map<String, String> synGetKaptcha() {
    OkHttpClient client = new OkHttpClient();
    Request request = new Request.Builder().url(CommonConfig.URL +
"/request/kaptcha").build();
    Map<String, String> resParams = new HashMap<>();
    try {
        Response response = client.newCall(request).execute();
        if (response.isSuccessful() && response.body() != null) {
            try {
                Message message = JSON.parseObject(response.body().string(),
Message.class);
                if (message != null && message.isSuccess()) {
                    String kaptchaToken =
message.getParams().get("kaptcha_token");
                    String img = message.getParams().get("img");

```

```
        if (kaptchaToken != null && img != null) {
            resParams.put("kaptcha_token", kaptchaToken);
            resParams.put("img", img);
        } else {
            resParams = null;
        }
    }
} catch (Exception e) {
    e.printStackTrace();
    resParams = null;
}
} catch (IOException e) {
    e.printStackTrace();
    resParams = null;
}
return resParams;
}
```

4. 调试分析（撰写人：曾鑫、刘笑枫、徐仕成）

4.1 遇到的问题及解决

曾鑫：

（1）最开始由于没有 android 手机，想用虚拟机调试，结果弄了半天发现 as 自带的虚拟机启动不了，之后又去找了 genymotion 模拟器，调试了一下午，发现有一些兼容问题遂放弃，之后又想用夜神模拟器，又发现项目当初设置的 API 过高，而夜神只支持 5.0，遂放弃。只得接用真机调试。

（2）当有了真机后，发现定位时有时无，不知为何。。。之后看网上说是采用的卫星定位，必须要在室外才可以。。。开始发现好像是这样，但随后有时间又会失灵，最终发现是必须一人一 key，一旦 key 的安装包在第一个手机上装载后就会与手机码绑定。

（3）主线程比子线程提前返回，用异步办法解决了。

刘笑枫：

（1）为了增加程序的安全性，我使用了 https 协议，通过 nginx 进行反向代理，将指定域名配置到 8082 端口。在部署到远程服务器时，出现了无法访问应用服务器的情况，我检查了阿里云控制台是已经开放了 443 端口，是否允许当前 ip 访问。再通过 tcping 工具检测是否可以访问 443 端口，访问失败。再检查 ubuntu16.04 是否开放了 443 端口，通过 ufw status 命令，检测到允许 443 端口访问。后来通过 iptables -L 检查是没有允许 443 端口的相关链，手动添加规则 iptables -I IN_public_allow -p tcp --dport 443 -j ACCEPT 成功访

问。

(2) 在部署应用服务器的时候, 为了让应用服务器在后台持续运行, 编写 shell 脚本

```
rm -f /root/springbootProject/swan.log  
ps -aux |grep swan-server-0.0.1.jar |cut -c 9-15 |xargs kill 9  
nohup java -jar /root/springbootProject/swan-server-0.0.1.jar >swan.log 2>&1 &  
# rm /root/springbootProject/swan-server-0.0.1.jar
```

易于部署

徐仕成:

(1) **问题:** 页面的部分 TopBar 的样式调整

页面的部分 TopBar 使用了 QMUI 的组件, 但是 QMUI 的开发文档非常的简略, 在没有文档的情况下修改样式是比较麻烦的一件事, 有一些组件的样式调整起来较困难。

解决: 通过查看 QMUI 相关组件的源码并且查看相关技术博客, 逐步解决相关问题, 达到想要的样式效果。

(2) **问题:** 登录、注册页面的设计

登录、注册页面的设计与样式调整较为复杂和细致。登录、注册页面元素丰富, 但要实现样式简洁清晰美观, 需要付出较多的时间, 在一些样式和细节的调整上是较为困难的。

解决: 登录、注册页面参考 GitHub 的登录、注册页面进行设计, 充分利用线性布局的优势, 并结合 ImageView、TextView、EditText、Button 等组件的特点进行布局上的优化和美化。利用长达 10 多个小时的时间进行不断的调整, 终于实现了想要的效果。

(3) **问题:** 注册页面图形验证码的样式调整

由于图形验证码是由服务器端返回的位图, 在尺寸上不符合要求, 所以需要对返回的位图进行格式的調整, 这一部分较为困难。

解决: 通过查阅相关文档, 了解了 Android 中对位图样式的调整方法, 最终通过一些细节上的微调, 达到理想效果。

(4) **问题:** 搜索页面搜索框的样式调整

由于搜索框使用了 Android 本身提供的 SearchView 组件, 而该组件的样式又被写死了, 想要自定义修改是很麻烦的一件事, 查阅了很多的解决办法都无法理想的实现。

解决: 通过详细阅读关于 SearchView 组件的介绍文档, 以及尝试多种解决办法, 最后综合的进行参考设计, 综合使用 XML 文档中相关属性的设计调整和使用 java 代码的相关操作方法, 综合调整, 最终达到理想效果。

(5) **问题:** 历史记录的 ListView 展示。

由于历史记录的内容不固定, 当历史记录较多, 超过一个页面的内容时, 清空历史记录的按钮就被挤掉了, 无法正常显示。想要实现当历史记录较少时, 清空历史记录的按钮紧跟在历史记录的后面, 当历史记录较多时, 清空历史记录固定在页面底部, 清空历史记录的按钮随历史记录的内容进行变化是比较不好处理的。

解决: 将 ListView 与 TextView 放在线性布局中, 设置 LinearLayout 的 layout_height="wrap_content", orientation="vertical"。ListView 的

layout_height="0dp", layout_weight="1"。TextView 的 layout_height="wrap_content"。充分利用线性布局的 layout_weight 属性实现效果。

(6) **问题:** 其它部分细节的样式调整问题。

相关显示的页面还是比较多的，有很多细节上的样式需要慢慢的调整。需要耗费较多时间。

解决: 反复调整, 通过多次真机实测, 查看效果, 逐步调整。

4.2 设计与实现的讨论分析

(1) 模拟器上无法定位成功。经过讨论, 仍无法确定根本原因, 最终放弃使用模拟器进行测试, 改用真机测试。

(2) 定位功能只能在一人的手机上运行成功。经过讨论，发现是由于在使用高德地图 API 的时候，需要设置唯一的 key 值，在开发时 key 值需要每个人自己生成自己对用的 key 值，不能共用同一个 key，所以导致了后面一系列的问题。后面解决了 key 值的问题后，在功能上就基本上都可以正常使用了。

(3) 导航和路线的交互设计。由于一些原因，页面上并没有直接显示导航和路线的按钮，需要在选择具体地点后，点击地点信息后再出现导航、路线的按钮，然后点击相应按钮，进行导航和查看路线详情。

4.3 核心功能测试截图



图 4-1 主页



图 4-2 历史记录



图 4-3 功能选择

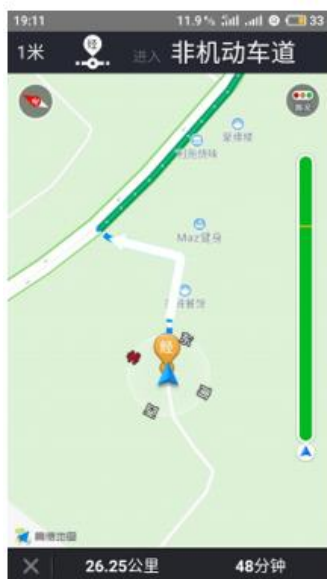


图 4-4 导航



图 4-5 方式选择



图 4-6 登录页面

4.4 经验体会

曾鑫:

经过这一次的实习，我明白了在调用 api 时应该怎么去分析怎么去看文档可以更合理的进行操作。同时在实习后期，给我的感觉就是有点忙不过来，压力有点大，同时由于是团队合作我们必须使用 github 来进行交互，由此也产生了一些小的问题，不过经过学习和沟通终于还是完美的解决了。同时，通过这次实习，我发现了自己的知识在一点一点的增长，对 android 的控件，布局，activity 等等知识有了更深的了解。同时也在整个过程中发现了很多不足，进行了查漏补缺。最后成功的完成这个项目使得我的信心大增。

刘笑枫:

经过这一次的实习，我懂得了如何去开发一个简单的安卓应用。一个安卓应用的工作量不仅仅只是 ui 界面，其系统内部的逻辑，内部的交互，服务端的开发，服务端的部署，以及客户端与服务端的通信对于开发者来说是必须要考虑的。

在开发过程中，由于使用了 git 工具进行团队的合作开发，在一些公共开发的模块经常会遇到代码冲突，因此保留或者丢弃部分代码成了我必须要考虑的问题。

徐仕成:

通过在实习过程中对各种问题的解决，提高了自己对 Android 页面设计的能力，同时也熟悉了 Android 开发的基本流程，对高德地图的 API 文档的学习使用更加的了解。在设计和调整样式的时候，出现了很多的问题，但是，当解决了这些问题后，自己也掌握了很多的技巧与方法，对 Android 页面的常用布局以及常用组件的控制调整也更加的熟练，掌握了更多的相关属性。在这个过程中也总结了很多的经验技巧，比如自定义 SearchView 的样式，对位图的样式调整，ListView 的使用等等。

5. 用户使用说明（撰写人：徐仕成）

5.1 操作步骤

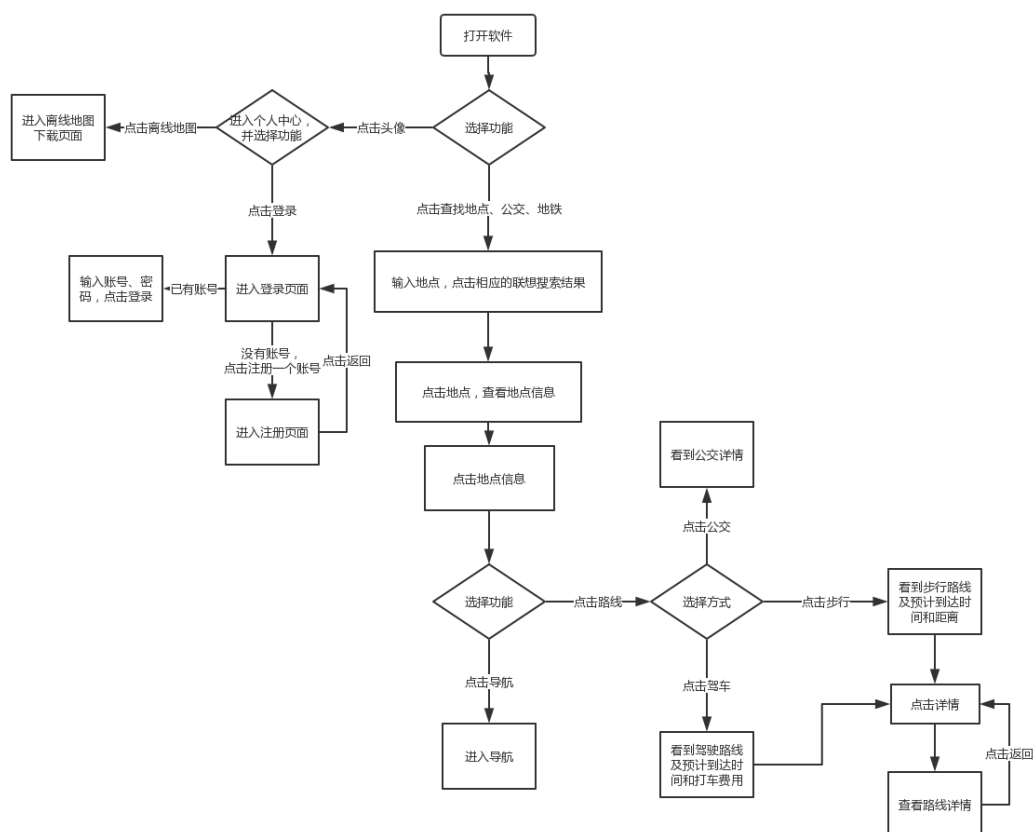


图 5-1 简易流程图

（1）进入：

打开 APP，点击同意获取相关权限。

（2）地点搜索、路线规划、导航的使用：

- 点击查找地点、公交、地铁，输入信息。
- 根据联想搜索的结果，选择对应信息点击。
- 然后点击具体点，查看点的具体信息。
- 再点击具体信息。
- 选择导航或路线。
- 选择导航后进入导航。
- 选择路线后，再选择驾车或公交或步行或跨城公交。

- 选择驾车或步行后可以点击查看路线详情。
- 选择公交或跨城公交后可以选择不同的公交路线。
- 点击相应公交路线查看公交路线详情

(3) 登录功能使用:

- 点击头像。
- 点击登录。
- 新用户点击注册一个账号。
- 已有账号, 输入账号和密码, 点击登录。

6. 软件测试结果 (撰写人: 徐仕成)

6.1 注册测试数据及结果

(1) 正确的输入输出

① 第一组测试数据

正确的输入:

账号: xushicheng

密码: 654123

确认密码: 654123

验证码: n18m

(注: 验证码为图形验证码, 需要输入图形验证码中的字符, 点击图形验证码可更新, 每次的验证码均为随机)

正确的输出:

点击注册按钮后: 提示注册成功, 并自动跳转到登录页面

② 第二组测试数据

正确的输入:

账号: ceshizhanghao

密码: ceshimima

确认密码: ceshimima

验证码: 2pw7

(注: 验证码为图形验证码, 需要输入图形验证码中的字符, 点击图形验证码可更新, 每次的验证码均为随机)

正确的输出:

点击注册按钮后: 提示注册成功, 并自动跳转到登录页面

③ 第三组测试数据

正确的输入：

账号：测试

密码：测试

确认密码：测试

验证码：3px3

（注：验证码为图形验证码，需要输入图形验证码中的字符，点击图形验证码可更新，每次的验证码均为随机）

正确的输出：

点击注册按钮后：提示注册成功，并自动跳转到登录页面

（2）错误的输入及其输出**① 第一组测试数据****错误的输入：**

账号：test3

密码：test3

确认密码：tets3

验证码：ae47

（注：验证码为图形验证码，需要输入图形验证码中的字符，点击图形验证码可更新，每次的验证码均为随机）

输出结果：

点击注册按钮后：提示两次密码不同，仍停留在注册页面

② 第二组测试数据**错误的输入：**

账号：test1

密码：test1

确认密码：test1

验证码：2dd2

输出结果：

点击注册按钮后：提示不可重复注册，仍停留在注册页面

③ 第三组测试数据**错误的输入：**

账号：test1

密码：test1

确认密码：test1

验证码：2dd

输出结果：

点击注册按钮后：提示 Kaptcha auth error!。表示验证码输入错误，仍停留在注册页面。

6.2 登录测试数据及结果

(1) 正确的输入输出

① 第一组测试数据

正确的输入：

账号：test1

密码：test1

正确的输出：

点击登录按钮后：直接跳转到个人中心页面

② 第二组测试数据

正确的输入：

账号：test2

密码：test2

正确的输出：

点击登录按钮后：直接跳转到个人中心页面

③ 第三组测试数据

正确的输入：

账号：测试

密码：测试

正确的输出：

点击登录按钮后：直接跳转到个人中心页面

(2) 错误的输入及其输出

① 第一组测试数据

错误的输入：

账号：test1

密码：tes1

输出结果：

点击登录按钮后：提示账号密码不匹配，仍停留在登录页面

② 第二组测试数据

错误的输入：

账号：tes2

密码: test2

输出结果:

点击登录按钮后: 提示不存在该用户, 仍停留在登录页面

③ 第三组测试数据

错误的输入:

账号: tes2

密码: test2

输出结果:

点击登录按钮后: 提示不存在该用户, 仍停留在登录页面

6.3 地点搜索测试数据及结果

① 第一组测试数据

输入: 桂林理工大学

显示: 如 图 6-1 搜索地点

② 第二组测试数据

输入: 银行

显示: 如 图 6-1 搜索银行

③ 第三组测试数据

输入: 景区

显示: 如 图 6-3 搜索景区



图 6-1 搜索地点

图 6-2 搜索银行

图 6-3 搜索景区

6.4 其他测试结果

部分测试结果见图 1-1、图 1-2、图 1-3、图 1-4、图 1-5、图 1-6、图 4-1、图 4-2、图 4-3、图 4-4、图 4-5。

补充部分测试截图：



图 6-4 驾车路线



图 6-5 驾车路线详情



图 6-7 公交路线详情

7. 心得体会（撰写人：刘笑枫、曾鑫、徐仕成）

刘笑枫：

经过为期 3 周的 Android 开发实习,我体验到了团队合作的重要性,对于安卓开发,每一个组员都有自身擅长的模块,充分利用自身的优点,扬长避短,在程序开发中尤为重要。

我所做的搜索模块,其最大的特点是他的联想搜索功能,即输入一个字符,系统会自动提示与该字符有关的选项,这极大地提高了用户的满意度。在开发过程中,与后台的网络通信是最难调试的,由于是异步非阻塞的请求,通过打断点进行调试很难取得较好的调试效果,因此,只能通过写日志的方式来发现并解决问题。

本次安卓项目的开发,其系统的可靠性与可维护性都比较差,用户不能得到一个满意的体验效果。其原因一方面是个人的能力较差,另一方面,3 周的时间和 3 个人远不能开发出像高德地图之类的健壮的系统,如果之后有时间,可以进行优化和添加新的功能。

曾鑫：

时间过得真快,实习一转眼间实习时间就过去了。在这段实习时间里,我学到了很多在

有用的东西，也认识到了自己很多的不足，感觉收益非浅。

在这几个周的实习过程中，实践让我的技能不断增长，工作能力不断加强。让我知道了一个人的知识和能力只有在实践中才能发挥作用，才能得到丰富、完善和发展。大学生成长，就要勤于实践，将所学的理论知识与实践相结合一起，在实践中继续学习，不断总结，逐步完善，有所创新，并在实践中提高自己由知识、能力、智慧等因素融合成的综合素质和能力，为自己事业的成功打下良好的基础。

在这过程中我在专业知识上得到了很多的进步。我学到了很多关于软件行业的知识，学到了 Android 编程技术。

学习 Android 有一段时间了，让我知道学习 Android 的时候 1、要有一份永不放弃的决心，尤其是刚开始的时候，面对全英文的黑白操作界面，所以，持之以恒是最重要的！2、自己创造一个 Android 学习的环境！由于 Android 有大量的操作需要记忆，很显然，靠死记硬背是不科学的！最好的方法就是实践，多操作，联系的多了，自然而然的就记住了，而且还是记得最牢靠的！3、细微处着眼。Android 是很严肃的，也许一个命令大小写的错误就可能导出错，甚至多一个少一个空格，或者小符号，它都会出错，这就要求我们在操作 Android 虚拟机的时候非常的细心，有足够的耐心！只有这样才能让我们学好 Android。

在实习工作中，我了解到了与人沟通合作是多么的主要。此次的实习任务比较艰巨由于小组成员较少，我们的工作量比较大，所以团队合作非常的重要。软件开发不是有一个厉害的人就可以很好地完成，同时也并不是人多就可以做好。必须是作为一个团队大家一起努力，按照进度有计划地进行。只要团队之中有任何一个人没有完成任务，那就会导致整个项目无法正常进行。

扎实的专业知识是提高工作水平的坚实基础。在学校学习专业知识时，可能感觉枯燥无味，但当工作以后，才会发现专业知识是多么的重要，因为在日常工作中要处处用到。要想提高我的工作效率，工作质量，很多知识只是知道是远远不够的，而是要精通。作为一个 IT 人员必须要有过硬的专业知识，要拥有较强的学习能力。时代在不断的发展，不断地进步。而信息行业的变化更是无比迅速。所以我们每时每刻都不能停下学习的脚步，我们只有通过不断的学习来充实自己，完善自己，才能在工作中更好地发挥自己的水平。

徐仕成：

经过为期 3 周的 Android 开发实习，使我在技能方面、代码阅读、代码改错、规范检查等方面有了一定的提高，训练了自己编写程序的熟练度和规范性，积累了一定的项目开发经验，通过完成项目案例和阶段项目实战，增加了自己对实际软件项目开发的体验，同时，项目组角色分配和技术研讨，培养我养成良好的职业习惯，逐步达到熟练掌握移动应用程序的开发。

在实习期间，也是遇到了各种各样的问题，感觉像是每天都在写 Bug，写完 Bug 又改 Bug。由于对知识掌握得不系统、不熟练、不扎实，在写的过程中就不停的在出问题，但是，通过不断的去努力想办法解决这些问题，自己也有了很大的提高。一些问题的解决过程很痛苦，但是，当解决了的时候也会感到非常的高兴。在这一次的实习中，我主要是负责页面的设计与美化，在 Android 开发中，页面的呈现主要是通过.xml 文档来进行设计，在 xml 文件中，利用 Android 本身提供的相关 View 组件和一些 ViewGroup 及相关布局，结合特定的属

性，来实现想要的效果。这个过程是有趣但繁琐的，同时也存在着很多的技术难点，比如自定义 SearchView 的样式，较好的展示出 ListView 的效果等等，通过刻苦的钻研和理解以及不断的尝试和实践，最终也都克服了相关的困难，使页面能够达到理想的效果，友好的展示出来。

总之，解决了这些问题后，自己也掌握了很多的技巧与方法，对 Android 页面的常用布局以及常用组件的控制调整也更加的熟练，掌握了更多的相关属性。在这个过程中也总结了很多的经验技巧，比如自定义 SearchView 的样式，对位图的样式调整，ListView 的使用等等。Android 开发是一个漫长且艰辛的学习过程，后续还需要不断的学习和实践，多学多练才能更好的掌握 Android 开发的技能。