

软件设计文档

选型理由

选择 **Java** 作为安卓 APP 开发语言。

- 目前安卓平台开发官方语言为 **Java**，官方支持度大
- 社区庞大，第三方资源多
- **Java** 适合团队开发，软件工程可以做到相对规范

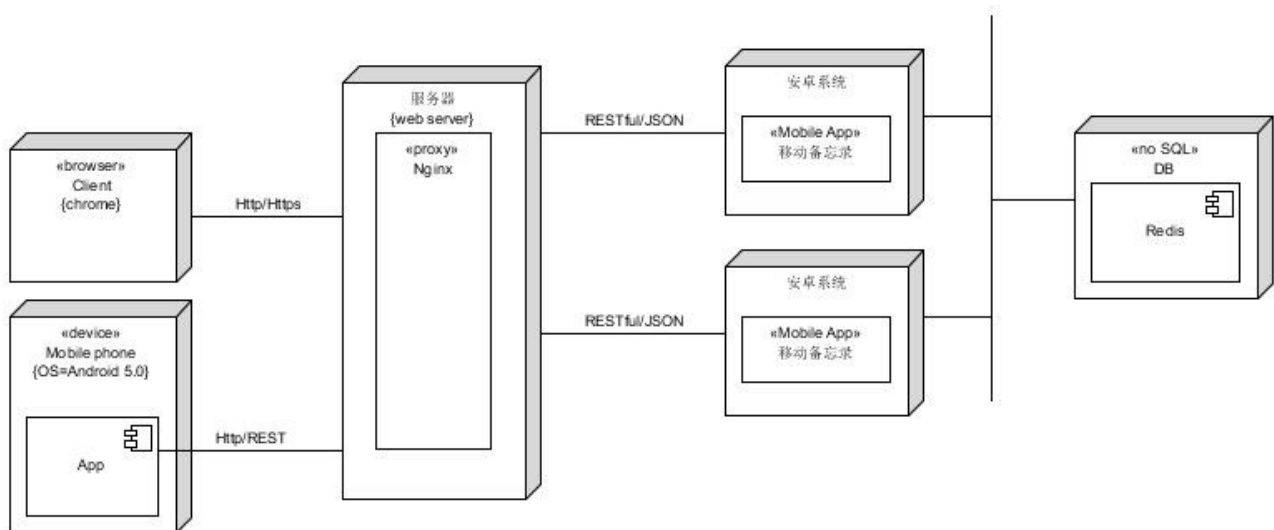
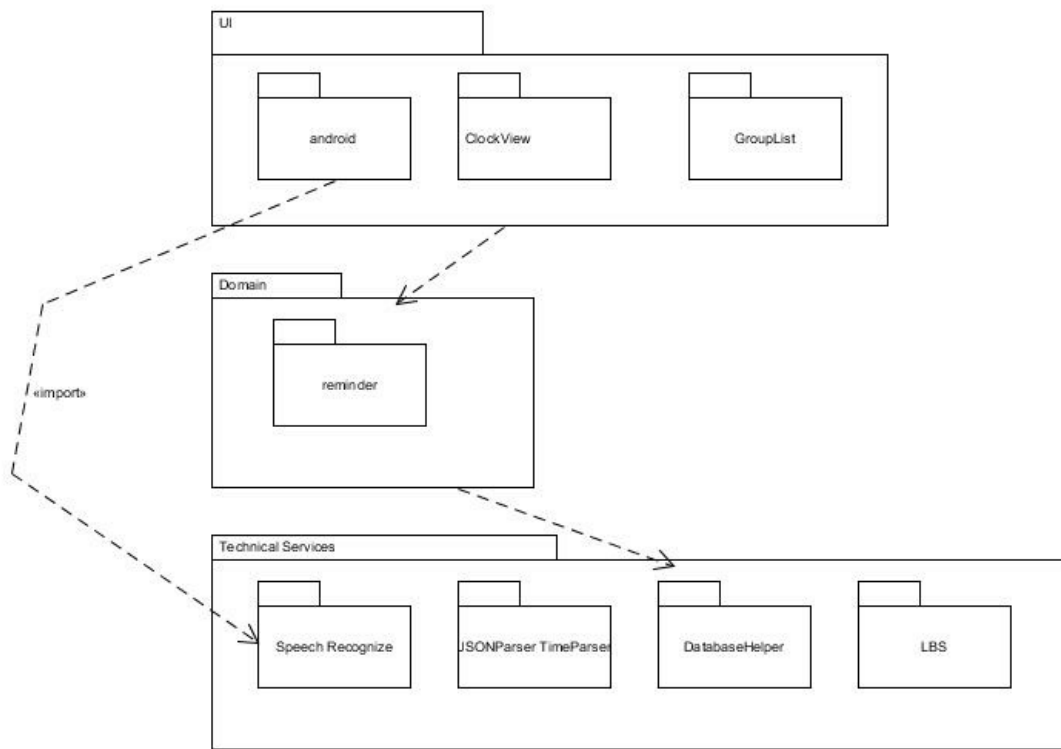
使用 **OrmLite** 作为数据库开发框架

- 原生的 **SQLite** 直接拼接 SQL 语句低效且不安全，
- **OrmLite** 基于对象关系映射，轻量、易学、安全。

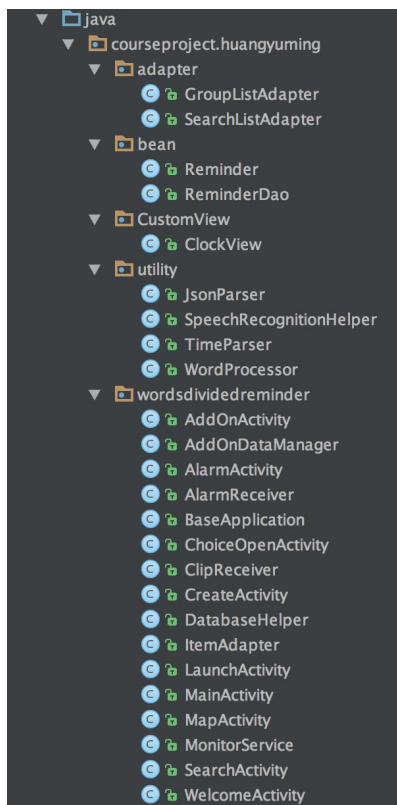
使用 **百度地图** 作为地图组件

- SDK 易上手
- 覆盖地域广

架构设计



模块划分



软件设计技术

- **Structure Programming**，一种编程典范。它采用子程序、程式码区块、for循环以及while循环等结构，来取代传统的 goto。希望借此来改善计算机程序的明晰性、品质以及开发时间，并且避免写出面条式代码。

```
searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {  
    @Override  
    public boolean onQueryTextSubmit(String query) { return false; }  
  
    @Override  
    public boolean onQueryTextChange(String newText) {  
        filteredList.clear();  
        for (Reminder item : rawList) {  
            if (item.getTime().contains(newText) || item.getPosition().contains(newText) || item.getTasks().contains(newText)) {  
                filteredList.add(item);  
            }  
        }  
        searchListAdapter.notifyDataSetChanged();  
        return false;  
    }  
});
```

- **Object-Oriented Programming**。Java 是一门面向对象的语言，因此面向对象涉及项目的方方面面。

```

public class ReminderDao {
    private Context context;
    private DatabaseHelper helper;

    public ReminderDao(Context context) {
        this.context = context;
        try {
            helper = DatabaseHelper.getHelper(context);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private Dao<Reminder, Integer> getReminders() throws SQLException {
        return helper.getRemindersDao();
    }
}

```

- **Component-Oriented Programming**。面向构件编程，即将一个大的程序拆分成一个个的小组件，再拼合而成。

```

public class ClockView extends View {
    public ClockView(Context context) {
        super(context);
        init();
    }

    public ClockView(Context context, AttributeSet attrs) {
        super(context, attrs);
        init();
    }

    public ClockView(Context context, AttributeSet attrs, int defStyleAttr) {
        super(context, attrs, defStyleAttr);
        init();
    }

    public ClockView(Context context, AttributeSet attrs, int defStyleAttr, int defStyleRes) {
        super(context, attrs, defStyleAttr, defStyleRes);
        init();
    }
}

```

- **Singleton pattern**，单例模式

```

public class SpeechRecognitionHelper {

    // 静态方法
    public static SpeechRecognitionHelper instance;
    public static SpeechRecognitionHelper getInstance() {
        if (instance == null) {
            instance = new SpeechRecognitionHelper();
        }
        return instance;
    }
}

```