



地铁管家说明文档

软工 141 - 王家梁

一、App 名称：地铁管家

二、自评分数：95

三、工作量：

代码行数：没有准确估计过 预计百行到千行吧反正很多

源代码文件数量：Java 主要文件 24 个 layout 布局文件 22 个 其余控制布局文件若干。

四、系统功能描述：

本系统一共实现了如下功能：

- 1、用户登录和注册功能，用户自动登录服务器验证。
- 2、地铁选站购票功能，目前只支持同线路站点，地铁选站支持开头字母大写分类
- 3、用户支持查询购票记录，包括未取票的记录和历史记录，所有记录均支持下拉刷新功能。
- 4、用户支持自定义修改设置功能。
- 5、用 laravel 框架定制的后台服务端，实现与客户端的完整交互。
- 6、对用户认证功能设计了基于 OAUTH2 协议的认证办法。

后台路由 API 列表：

```
Route::get('/user', function (Request $request) {
    return $request->user();
})->middleware('auth:api');

////login old api routes
//Route::post('/login', "LoginController@login")->middleware('auth:api');

Route::post('/register', "RegisterController@register");

//get station list api
Route::get('/stations', "StationController@getStationList");

//get faremap list
Route::post('/faremap', "FaremapController@getFaremapList");

//insert new order to order history
Route::post('/submitOrder', "HistoryController@submitOrder")->middleware('auth:api');

//get order
Route::get('/getNoTravelOrder', "HistoryController@getNoTravelOrder")->middleware('auth:api');
Route::get('/getTotalOrder', "HistoryController@getTotalOrder")->middleware('auth:api');

//edit user
Route::post('/editUser', "EditUserController@editUser")->middleware('auth:api');
```

四、系统运行截图：

运行环境：android 5.1 模拟 最低要求 SDK4.4

App 开启运行后，登录界面和注册界面如图所示：



地铁管家

 请输入邮箱地址



 请输入密码

[忘记密码?](#)

登录 >





 请输入邮箱地址



 请输入昵称



 请输入密码



 再次输入密码



 请输入手机号码

注册 >

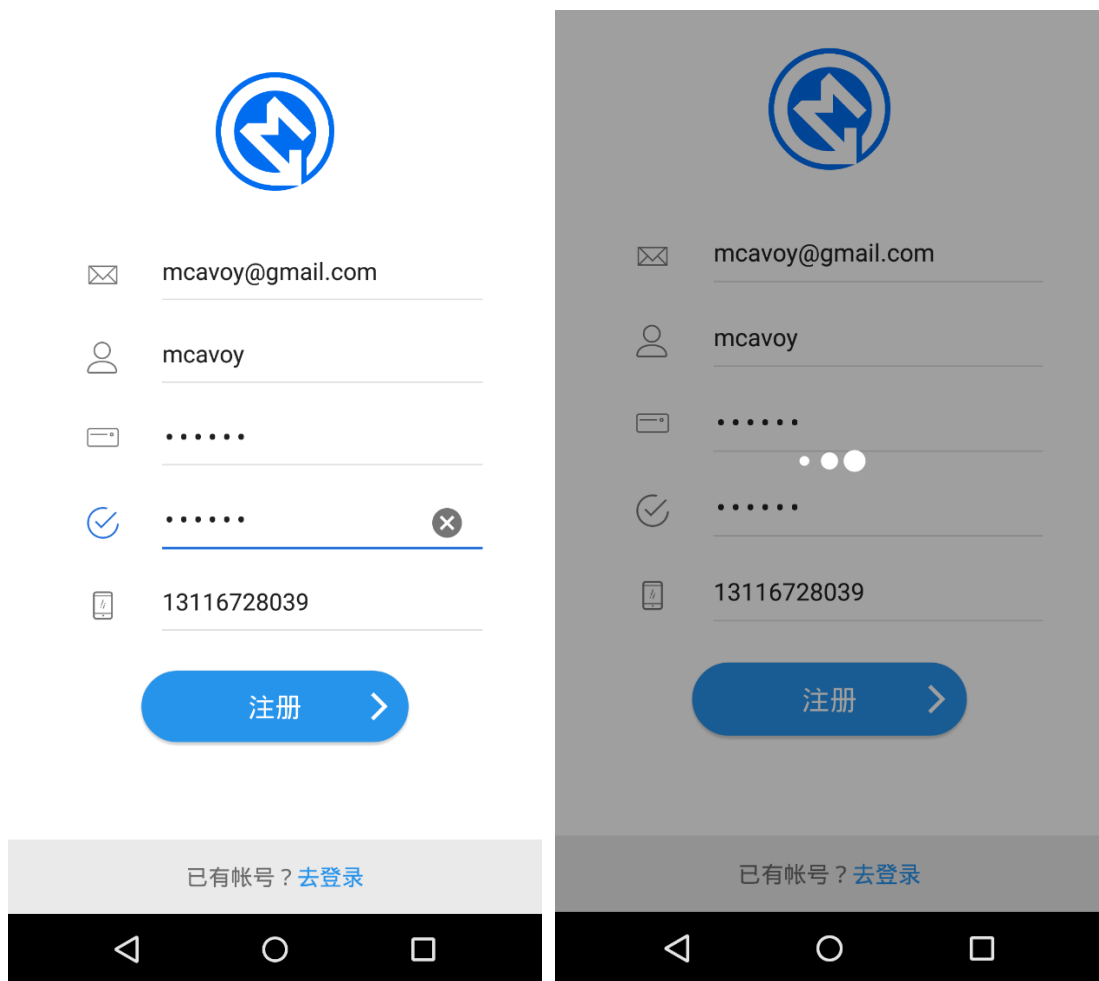
没有账户？[现在创建](#)



已有帐号？[去登录](#)



填入注册信息后 点击注册，出现载入界面，服务器验证成功好，提示注册成功 返回到登录界面。



输入刚才我们注册的信息，即可登录到主界面：

The image shows a mobile application login screen. It features the same blue circular logo at the top. Below the logo are two input fields: an email field containing 'mcavoy@gmail.com' and a password field with masked characters '.....' and a small 'x' icon on the right. Below the password field is a link '忘记密码?' (Forgot password?). At the bottom of the form is a blue rounded button labeled '登录' (Login) with a right-pointing arrow.

主界面如图所示：



此时我们可以看到，立刻购票的按钮是禁止点击的，因为我们还没有选择车次信息。这个时候我们点击始发站和终点站按钮，选择需要购票的站台信息。

、

此时进入的是选择站点界面，最右边的字母导航栏可以实现快速定位我们想要选择的车

站，每个车站选项的右边也都有几号线的提示。



这里我们点击九和路，然后在终点站处也同样选择，比如选择城站

出发站

九和路



到达站

城站

然后选择需要购票的数量，这个时候系统会异步从服务器更新票价

数量

⊖ 1 ⊕

应付金额

5.00 元

更改数量也会异步更新金额

数量

⊖ 3 ⊕

应付金额

15.00 元

此时我们点击立即购票 会出现支付对话框

3:17

首页

出发站

九和路

到达站

城站

×

支付详情

订单名称

杭州地铁单程票

起点

九和路

终点

城站

需付款

15.00元

确认付款

点击确认付款以后，因为这里省略了用支付宝支付接口，所以直接会提示付款完成。

×

支付详情

订单名称

杭州地铁单程票

起点

九和路

终点

城站

需付款

15.00元

购票成功

然后回到我们可以在历史订单中查询到未出行的订单信息

3:20

☰ 购票记录

未取票 | 历史订单

九和路 - 城站

3张 | 15.00元

未取票 >

点击未取票的订单信息，我们可以查看该订单的详细信息。二维码自动生成，用于地铁口取票之用，由于调试关系，故没有做扫描二维码的验证。



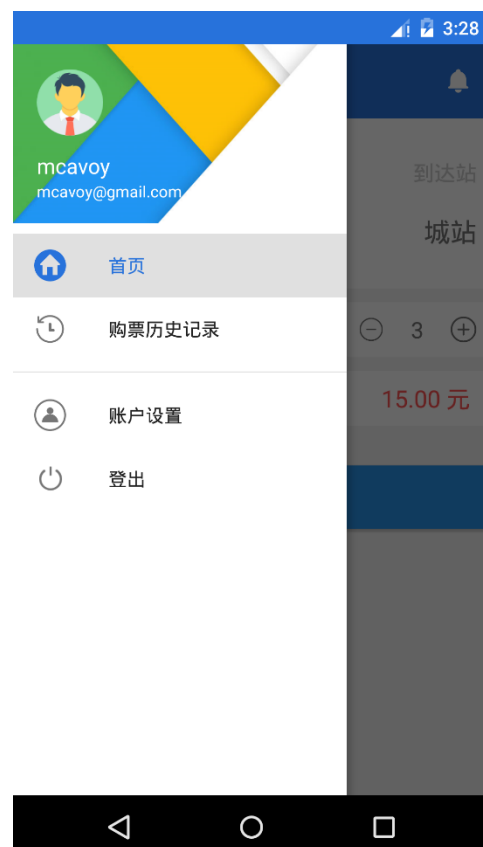
将前面的界面划到右边，可以查看已经出行的历史订单信息，已经取票的订单将不再支持查看详细信息。



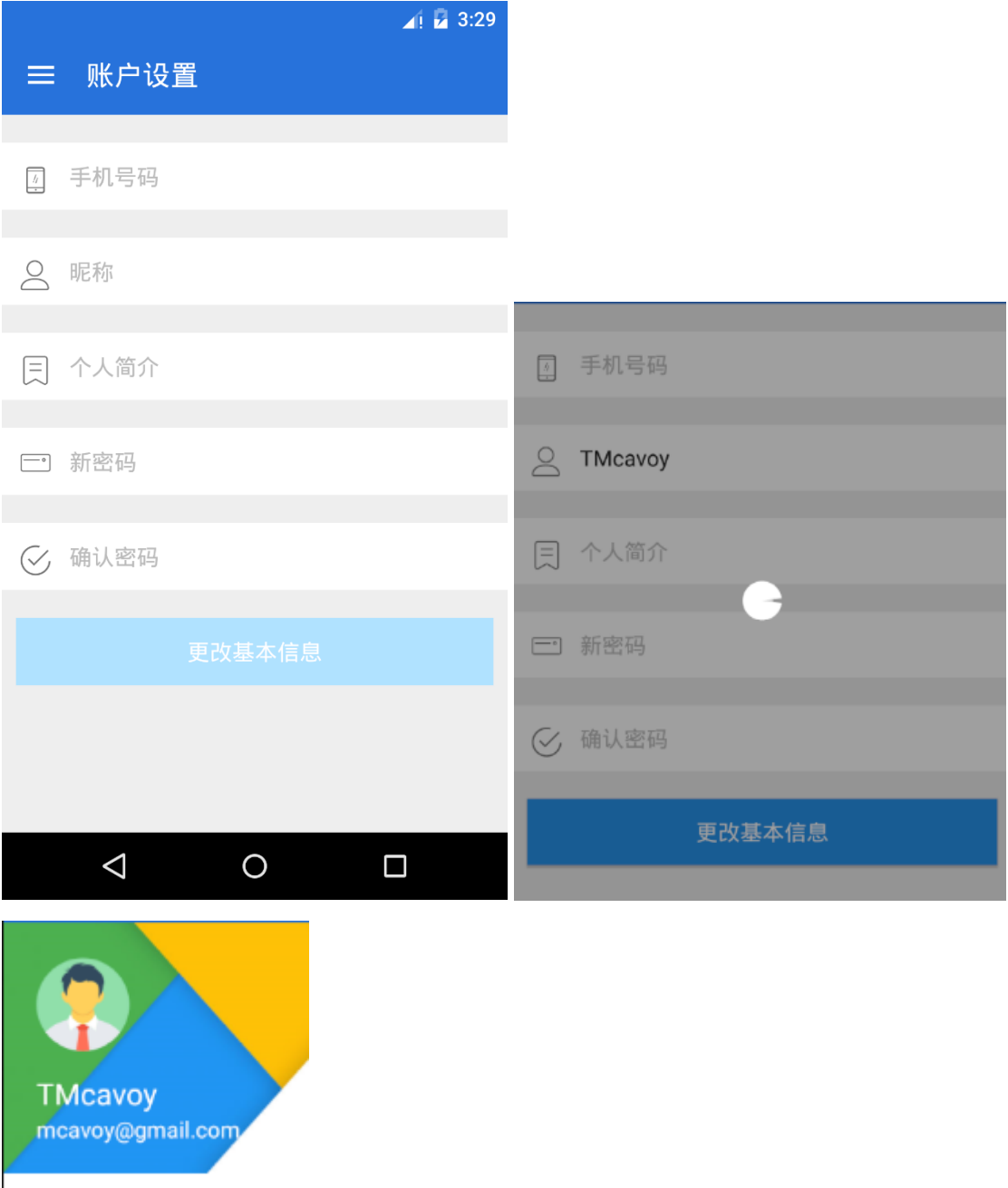
历史记录界面已经实现了下拉刷新, 为了优化加载我在这个历史记录界面均采用手动下拉刷新功能, 这样提高了 App 界面切换的流畅度。



首页和部分页面向左滑动可以唤出侧边栏：



点击账户设置，可以对当前账户进行设置，功能因为时间问题还不完善，但是如图所示的几个选项的修改都已经实现，此时我们将昵称改成 TMcavoy，如下图所示修改成功。



App 本身实现了自动登录功能，通过和服务器交换 token 值，基于 OAUTH2 协议，结合本地的账户信息存储，可以判断用户登录是否过期。如果本地有用户信息，会和服务器交互看看是否登录成功。为了保证最新的 用户数据在加载完成后才会被写入 App 中存储，我加入了加载动画界面作为交互。

