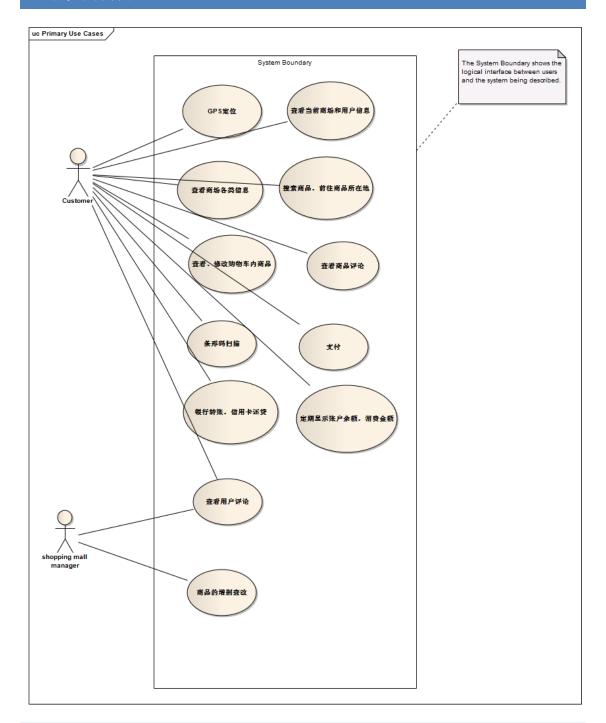


# 目录

1.	初步用例图3			
	1.1	1.1 角色		
	1.2	用例		3
2. 类图				4
	2.1	客户端类图		4
		2.1.1	总体框架图	4
		2.1.2	实体对象	6
		2.1.3	条形码扫描类	6
		2.1.4	扫描结果及处理	7
	2.2	2 服务端类图		9
		2.2.1	商场服务器	9
		2.2.2	银行服务器	9

# 1. 初步用例图



# 1.1 角色

- ▶ 消费者
- ▶ 商场管理员

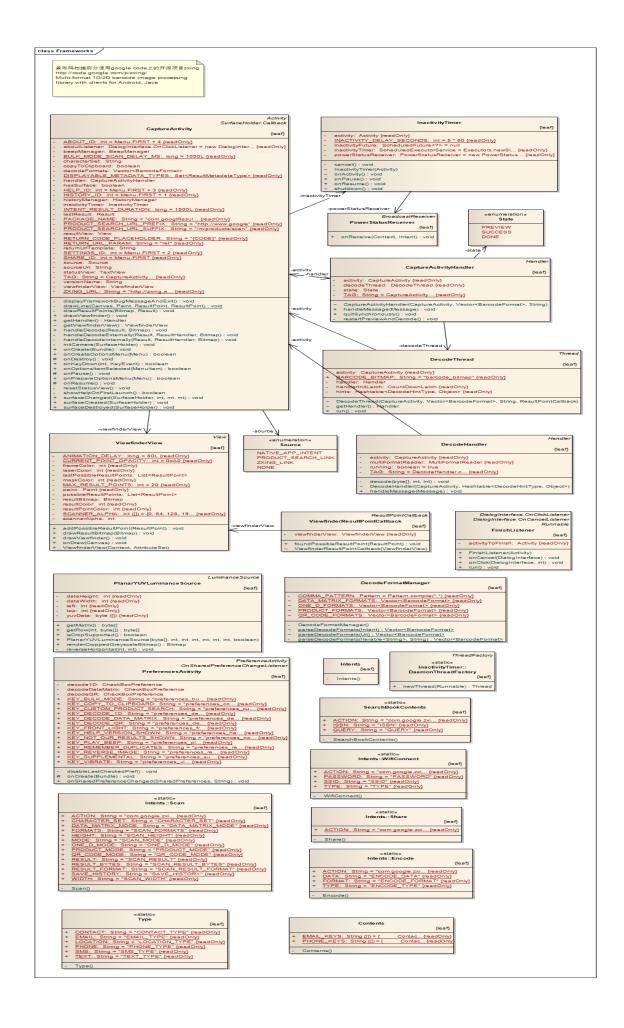
# 1.2 用例

- ▶ 消费者:
- 通过 GPS 定位查找商场
- 查看商场信息
- 搜索商品信息,摆放位置
- 查看商品评论
- 条形码扫描商品
- 加入,删除,更改购物车
- 支付
- 查看余额
- ▶ 商场管理员:
  - 产看消费者的评论
  - 商品的增删查改

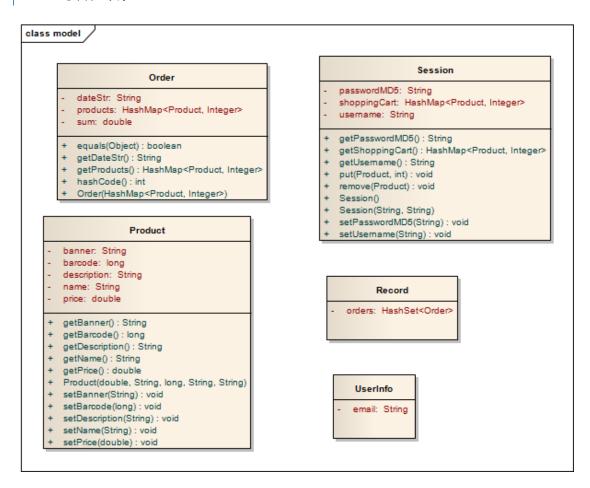
# 2. 类图

# 2.1 客户端类图

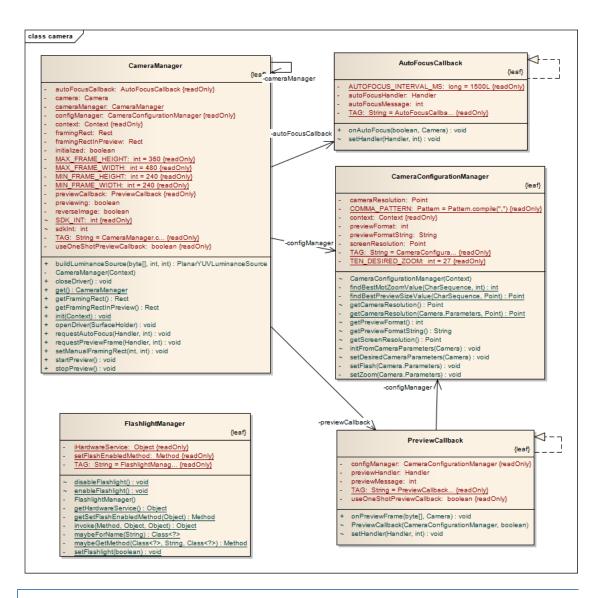
# 2.1.1 总体框架图



## 2.1.2 实体对象



# 2.1.3 条形码扫描类



### 2.1.4 扫描结果及处理

# class result / ResultHandler activity: Activity (readOnly) customProductSearch: String {readOnly} DATE\_FORMAT: DateFormat {readOnly} DATE\_TIME\_FORMAT: DateFormat (readOnly) DATE\_TIME\_FORMAT: DateFormat = new SimpleDateF.... {readOnly} GOOGLE\_SHOPPER\_ACTIVITY: String = GOOGLE\_SHOPPER\_.... {readOnly} GOOGLE\_SHOPPER\_PACKAGE: String = "com.google.end.... {readOnly} MARKET\_REFERRER\_SUFFIX: String = "areferer=utm .... {readOnly} MARKET\_URI\_PREFIX: String = "market://searo.... {readOnly} MAX\_BUTTON\_COUNT: int = 4 {readOnly} rawResult: Result {readOnly} result: ParsedResult {readOnly} addCalendarEvent(String, String, String, String, String): void ${\it addContact(String[], String[], String[], String, String, String, String, String): void areContentsSecure(): boolean\\$ calculateMilliseconds(String) : long dislPhone(String) : void dislPhoneFromUri(String) : void fillInCustomSearchURL(String): String getButtonCount(): int getButtonText(int) : int getDirections(double, double) : void getDisplayContents() : CharSequence getDisplayTitle() : int getResult(): ParsedResult getType(): ParsedResultType handleButtonPress(int): void hasCustomProductSearch(): boolean launchIntent(Intent): void openBookSearch(String): void openGoogleShopper(String): void openMap(String) : void openProductSearch(String): void openURL(String) : void parseCustomSearchURL() : String putExtra(Intent, String, String): void ResultHandler(Activity, ParsedResult) ResultHandler(Activity, ParsedResult, Result) searchBookContents(String): void searchMap(String, String): void sendEmail(String, String, String): void sendEmailFromUri(String, String, String, String): void sendMMS(String, String, String): void ${\sf sendMMSFromUri}({\sf String},\,{\sf String},\,{\sf String}): {\sf void}$ sendSMS(String, String) : void sendSMSFromUri(String, String): void shareByEmail(String) : void shareBySMS(String): void showGoogleShopperButton(View.OnClickListener): void showNotOurResults(int, AlertDialog.OnClickListener): void

#### ResultHandlerFactory

(leaf)

- makeResultHandler(Activity, Result) : ResultHandler parseResult(Result) : ParsedResult ResultHandlerFactory()

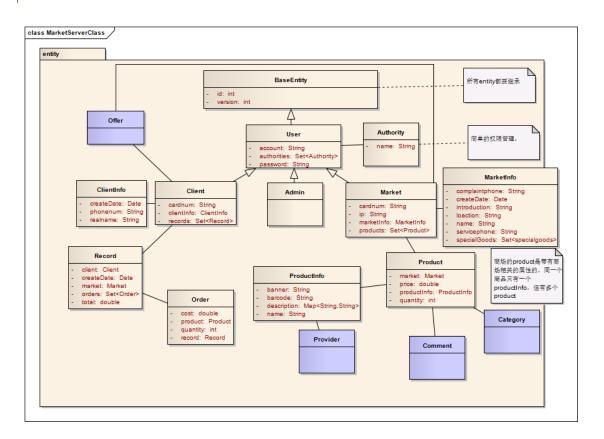
## class encode

webSearch(String) : void wifiConnect(WifiParsedResult) : void

#### Activity QRCodeEncoder EncodeActivity (leaf) {leaf} activity: Activity (readOnly) BLACK: int = 0xFF000000 {readOnly} contents: String MAX\_BARCODE\_FILENAME\_LENGTH: int = 24 {readOnly} qrCodeEncoder: QRCodeEncoder dimension: int {readOnly} TAG: String = EncodeActivity.... {readOnly} displayContents: String $\underline{\mathsf{makeBarcodeFileName}}(\mathsf{CharSequence}): \mathsf{CharSequence}$ format: BarcodeFormat TAG: String = QRCodeEncoder.c... {readOnly} onCreate(Bundle) : void onCreateOptionsMenu(Menu) : boolean -qrCodeEncoder title: String WHITE: int = 0xFFFFFFFF (readOnly) onOptionsItemSelected(MenuItem) : boolean onResume(): void showErrorMessage(int) : void encodeAsBitmap() : Bitmap encodeContentsFromShareIntent(Intent): boolean encodeContentsFromShareIntentDefault(Intent): boolean encodeContentsFromShareIntentPlainText(Intent): boolean encodeContentsFromZXingIntent(Intent): boolean encodeQRCodeContents(Intent, String): void encodeQRCodeContents(AddressBookParsedResult): boolean escapeMECARD(String): String getContents(): String getDisplayContents(): String getState(): String guessAppropriateEncoding(CharSequence): String QRCodeEncoder(Activity, Intent, int) trim(String): String

# 2.2 服务端类图

### 2.2.1 商场服务器



## 2.2.2 银行服务器

