**在支付宝标准SDK的基础上再做了一层封装， 即C#版本demo中F2FPayDll.dll**

* 对外数据模型：



* 接口

public interface IAlipayTradeService

{

**//当面付条码支付**

**AlipayF2FPayResult tradePay(AlipayTradePayContentBuilder builder);**

**// 当面付2.0支付查询**

**AlipayF2FQueryResult tradeQuery(string outTradeNo);**

**// 当面付2.0支付退货**

**AlipayF2FRefundResult tradeRefund(AlipayTradeRefundContentBuilder builder);**

**// 当面付2.0预下单**

**AlipayF2FPrecreateResult tradePrecreate(AlipayTradePrecreateContentBuilder builder);**

**// 当面付2.0预下单，需要监听异步通知的模式**

**AlipayF2FPrecreateResult tradePrecreate(AlipayTradePrecreateContentBuilder builder, string notify\_url);**

}

1. **条码支付：**

提供：支付、查询、退款 （条码支付），封装了支付宝SDK，提供给收银系统界面直接调用。

1. 支付接口流程：



示例代码：

AlipayTradePayContentBuilder builder = BuildPayContent();

string out\_trade\_no = builder.out\_trade\_no;

AlipayF2FPayResult payResult = serviceClient.tradePay(builder);

switch (payResult.Status)

{

case ResultEnum.SUCCESS:

DoSuccessProcess(payResult);

break;

case ResultEnum.FAILED:

DoFailedProcess(payResult);

break;

case ResultEnum.UNKNOWN:

result = "网络异常，请检查网络配置后，更换外部订单号重试";

break;

}

1. 查询接口流程



示例代码：

//商户订单号

string out\_trade\_no = WIDout\_trade\_no.Text.Trim();

//商户网站订单系统中唯一订单号，必填

AlipayF2FQueryResult queryResult = serviceClient.tradeQuery(out\_trade\_no);

//请在这里加上商户的业务逻辑程序代码

//——请根据您的业务逻辑来编写程序（以下代码仅作参考）——

switch (queryResult.Status)

{

case ResultEnum.SUCCESS:

DoSuccessProcess(queryResult);

break;

case ResultEnum.FAILED:

DoFailedProcess(queryResult);

break;

case ResultEnum.UNKNOWN:

if (queryResult.response == null)

{

//result = "网络异常，请检查网络配置";

result = "配置或网络异常，请检查";

}

else

{

result = "系统异常，请重试";

}

break;

}

1. 退款接口流程：

示例代码：

AlipayTradeRefundContentBuilder builder = BuildContent();

AlipayF2FRefundResult refundResult = serviceClient.tradeRefund(builder);

//请在这里加上商户的业务逻辑程序代码

//——请根据您的业务逻辑来编写程序（以下代码仅作参考）——

switch (refundResult.Status)

{

case ResultEnum.SUCCESS:

DoSuccessProcess(refundResult);

break;

case ResultEnum.FAILED:

DoFailedProcess(refundResult);

break;

case ResultEnum.UNKNOWN:

if (refundResult.response == null)

{

result = "配置或网络异常，请检查";

}

else

{

result = "系统异常，请走人工退款流程";

}

break;

}

**2. 扫码支付（不含异步通知）**

预下单（扫码支付）接口

示例代码：

AlipayTradePrecreateContentBuilder builder = BuildPrecreateContent();

string out\_trade\_no = builder.out\_trade\_no;

AlipayF2FPrecreateResult precreateResult = serviceClient.tradePrecreate(builder);

//如果需要接收扫码支付异步通知，请调用另外一个方法

//string notify\_url = "http://10.5.21.14/notify\_url.aspx"; //商户接收异步通知的地址

//AlipayF2FPrecreateResult precreateResult = serviceClient.tradePrecreate(builder, notify\_url);

//以下返回结果的处理供参考。

//payResponse.QrCode即二维码对于的链接

//将链接用二维码工具生成二维码打印出来，顾客可以用支付宝钱包扫码支付。

string result = "";

switch (precreateResult.Status)

{

case ResultEnum.SUCCESS:

DoWaitProcess(precreateResult);

break;

case ResultEnum.FAILED:

result = precreateResult.response.Body;

Response.Redirect("result.aspx?Text=" + result);

break;

case ResultEnum.UNKNOWN:

if (precreateResult.response == null)

{

result = "配置或网络异常，请检查后重试";

}

else

{

result = "系统异常，请更新外部订单后重新发起请求";

}

Response.Redirect("result.aspx?Text=" + result);

break;

}

**扫码支付获得支付结果：**

方案一： 采用上面的查询接口。

方案二： 采用轮询逻辑：

