肇慶學院

毕　业　论　文

论文题目　移动服务系统的分析与设计

——无线点菜系统客户端

系　　别　计算机科学系

专　　业　计算机科学与技术

年　　级　2003

学　　号　2B0324151138

学生姓名　樊海军

指导教师　杨帆

完成时间　2007　年　4　月

肇庆学院教务处制

目　　录

1 需求分析 1
1.1 应用背景 1
1.2 功能需求 1
2 系统功能设计 2
2.1 业务流程描述 2
2.2 基本功能 2
2.3 功能描述 3
3 系统环境设计 3
3.1 移动数据库技术简介 3
3.1.1 移动数据库SQL Server Mobile 3
3.1.2 SQL Server Mobile 体系结构 3
3.1.3 SQL Server Mobile 运行环境 4
3.1.4 SQL Server Mobile与 SQL Server 数据库交换数据的方法 4
3.2 基本硬件环境 4
3.3 基本软件环境 5
3.4 开发工具 5
3.5 环境搭建与配置 5
3.5.1 安装SQL Server 2000 5
3.5.2 安装配置IIS服务器 5
4 数据表设计 7
4.1 数据结构设计 7
4.2 数据表关系图 9
5 详细功能设计 10
5.1 桌台管理 10
5.1.1 预定 10
5.1.2 开台 11
5.1.3 清台 11
5.1.4 并台 12
5.1.5 分台 13
5.1.6 转台 14
5.2 菜品管理 14
5.2.1 点菜 14
5.2.2 未发送菜品 15
5.2.3 已点全部菜品 15
5.2.4 已上菜品 15
5.2.5 未上菜品 16
5.3 结账 16
6 测试和发布 16
6.1 程序测试 16
6.2 程序发布 20
6.2.1 为解决方案添加智能设备 CAB 项目 20

6.2.2 为 CAB 项目添加设备项目应用程序 20
6.2.3 生成 CAB 文件 21
6.2.4 将 CAB 文件部署到设备上 21
7 结束语 21
参考文献 21
致 谢 21
Analysis and design of the Mobile Servicing System 22

移动服务系统的分析与设计

——无线点菜系统客户端

樊海军　　指导老师：杨帆

摘　要　研究了基于移动智能客户端的无线点菜系统应用方案，目的就是通过最新的802.11b 无线信息技术和智能掌上电脑，为餐饮业带来崭新的服务手段与管理理念，优化业务流程，大幅提升餐厅的服务品质与企业形象，实现企业价值最大化同时又使自己的成本最低化，并最终提高其本身在餐饮行业的竞争优势。

关键词　移动服务；掌上电脑；餐饮；无线点菜

1 需求分析

1.1 应用背景

在互联网飞速发展的今天，各行各业都在积极地向信息科技靠拢。而餐饮业庞大的场地空间、繁琐而重复的点菜程序尤需改造。餐饮业传统的点菜方式纯属人工操作，因此繁重的工作量、高比例的人为错误是严重影响工作效率的主要原因，其中存在的几大弊病归纳起来如下：

人工传递单据浪费时间，效率低下；

经营大规模菜系时单据多、信息量大，而开单、传菜等环节越多越易出差错，直接影响服务质量和效率；

我们的无线点菜系统目的就是通过最新的802.11b 无线信息技术和智能掌上电脑，为餐饮业带来崭新的服务手段与管理理念，优化业务流程，大幅提升餐厅的服务品质与企业形象，实现企业价值最大化同时又使自己的成本最低化，并最终提高其本身在餐饮行业的竞争优势。

1.2 功能需求

[**无线点菜**]：通过具有无线功能的智能掌上电脑（PDA），服务员可以随时随地为顾客点菜，并即时的把数据传到后台数据库，厨房管理端将可以得到菜单信息，以进行配菜操作。

[**即时数据同步查询**]：无线掌上电脑与数据库实时同步，保证数据统一。让操作者在网内任何地点都能查询当前的桌台状态点菜单等。

[**无需布线**]：系统前台使用无线网络与掌上电脑技术，使前台使用者可以在营业大厅内随意走动，自由的使用系统为顾客服务，无需在大厅中布置任何网络线路而影响餐厅环境。

[**操作简便**]：使用智能的掌上电脑作为客户端，所有的操作都是笔触式或手触式，所以操作非常方便，不用经过复杂的培训就能熟练使用，特别适宜于计算机水平一般的服务员使用。

[**长距离传送**]：普通 AP 室内传送距离可达 150 米，室外传送距离可达 300 米。如果加放大器，传输距离可以达到 30 公里。

[**节约成本**]：简化后厨及传菜间的业务流程，在提高后厨工作效率的同时减少损耗。

[**提高效率、减少错误**]：大幅度加快上菜速度，提高餐饮业的工作效率和服务质

量；减轻收银台计算的工作量，在加快结账速度的同时减少收银结账错误的发生。

[树立品牌形象、提高顾客知名度]：无线智能掌上电脑具有时尚流行的外观以及绚丽的彩色屏幕，当顾客看见服务员都使用如此先进时尚的掌上电脑后，一定会觉得该餐馆具有很高的品位，可以提高酒店的档次，有力的宣扬用户的品牌，提高知名度。为了与实际的应用相接近，我们用“云飞”命名这个系统。

2 系统功能设计

2.1 业务流程描述

一般的餐饮点菜服务的基本工作流程是：

具体流程如下：客人进入餐厅后由服务员接待登记用餐位置，一般要分配给某一个固定的桌台号，此时接待处并不需要使用计算机进行开单的操作；客人安排好桌台后持台号牌进行点菜，此时手持掌上电脑的服务员进行开单、点菜等操作；客人在用餐过程中需要加菜、转台等服务员可以使用 PDA 随时进行操作；客人点菜单在服务员操作完的同时就被传到后厨；待厨房配菜完毕后通知传菜员上菜；服务员可以使用 PDA 随时查询当前桌台的各菜品状态，以方便告知顾客；用餐完毕后，服务员利用 PDA 查看当前账单，结账后可以清台。

2.2 基本功能

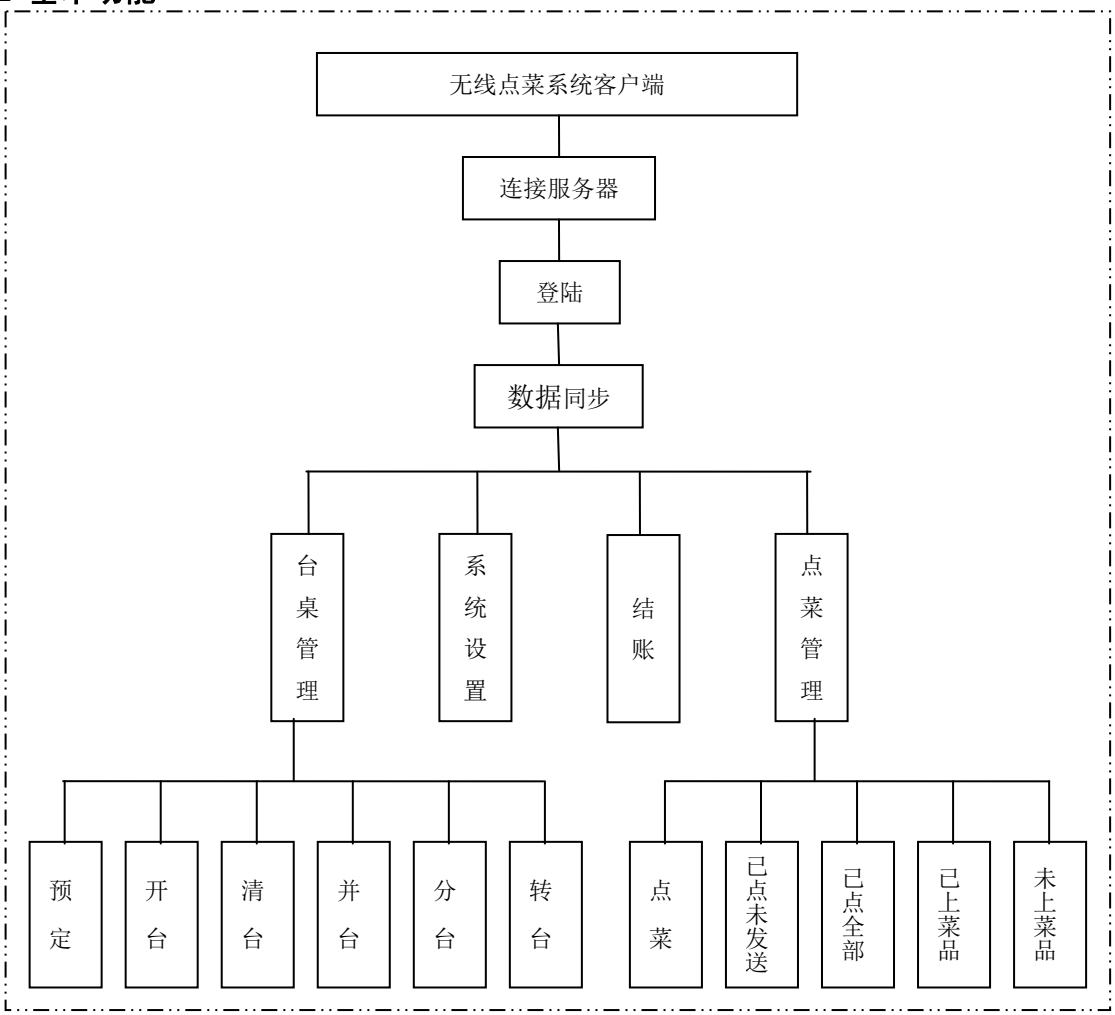

图 2-1 系统功能图

台桌管理：预定、开台、清台、并台、分台、转台等；

点菜管理：点菜（包括选择口味，做法等）、查看菜品状态，查看当前已点菜品总共价格；

系统设置：数据的同步系统参数的修改；

结账：查看账单，结账。

2.3 功能描述

1、无线开单

当客人来店后，服务员使用 PDA 可以即时下载桌台信息，点击开单，完成后系统自动将数据传送至数据库。

2、无线点菜

客人选好菜后，服务员用 PDA 将客人选好的菜加入系统内客人的点菜单中，系统内的其他站点也可以同时查询菜单添加菜单。

3、无线传菜

服务员点菜完成后，并没有传到数据库，此时可以进行修改，确认无误后点击发送按钮进入数据库。

4、转台分单

服务员可以通过 PDA 转台，将一台原由的信息转移到另外一个空台上，还可以将两张台合并成一台就餐，两台信息也将合并，并能进行分台操作，将一张台分成两张。

5、结账清台

当客人要结账时，可以通过 PDA 进行结账。结账完成后将允许服务员进行清台操作。

6、预定桌台

当客人需要预定某一个时间的桌台时，可以由服务员进行预定操作。

3 系统环境设计

3.1 移动数据库技术简介

3.1.1 移动数据库 SQL Server Mobile

因为移动服务系统需要下载桌台表及菜单等信息到移动设备上，并在移动设备上临时保存，所以在移动设备上需要有一个数据库系统。我们选择的移动数据库系统是 SQL Server Mobile。

Microsoft SQL Server 2005 Mobile Edition 是用于移动设备上的小型关系数据库。可以在 Visual Studio .NET 中通过 ADO.NET 访问 SQL Server Mobile 的数据库。本系统采用的是 Microsoft SQL Server 2005 Mobile Edition 3.0 (SQL Server Mobile) 。

3.1.2 SQL Server Mobile 体系结构

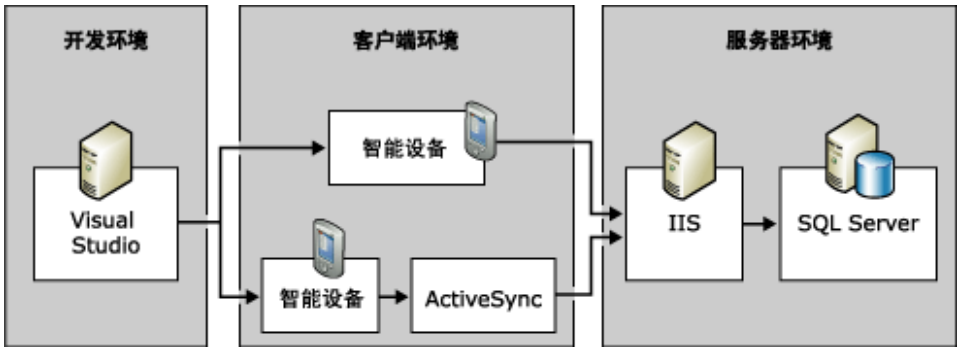

图 3-1 SQL Server Mobile 体系结构

3.1.3 SQL Server Mobile 运行环境

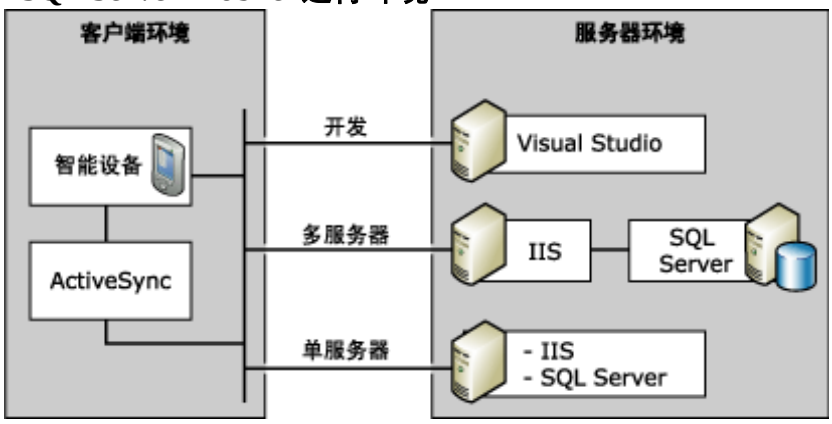

图 3-2 SQL Server Mobile 运行环境

3.1.4 SQL Server Mobile 与 SQL Server 数据库交换数据的方法

合并复制，提供一种强大且功能齐全的解决方案，该解决方案允许移动应用程序对复制的数据自行更改，然后再将这些更改与 Microsoft SQL Server 数据库进行合并，并在必要时解决冲突。

远程数据访问 (RDA)，使移动应用程序能够方便地将远程 Microsoft SQL Server 数据库表中的数据读取（Pull）到本地 SQL Server Mobile 数据库表中，或者将本地 SQL Server Mobile 数据库表中的数据发送（Push）到远程 Microsoft SQL Server 数据库表中。RDA 还可用于在运行 SQL Server 的服务器上发出 SQL 命令 (SubmitSql)。

3.2 基本硬件环境

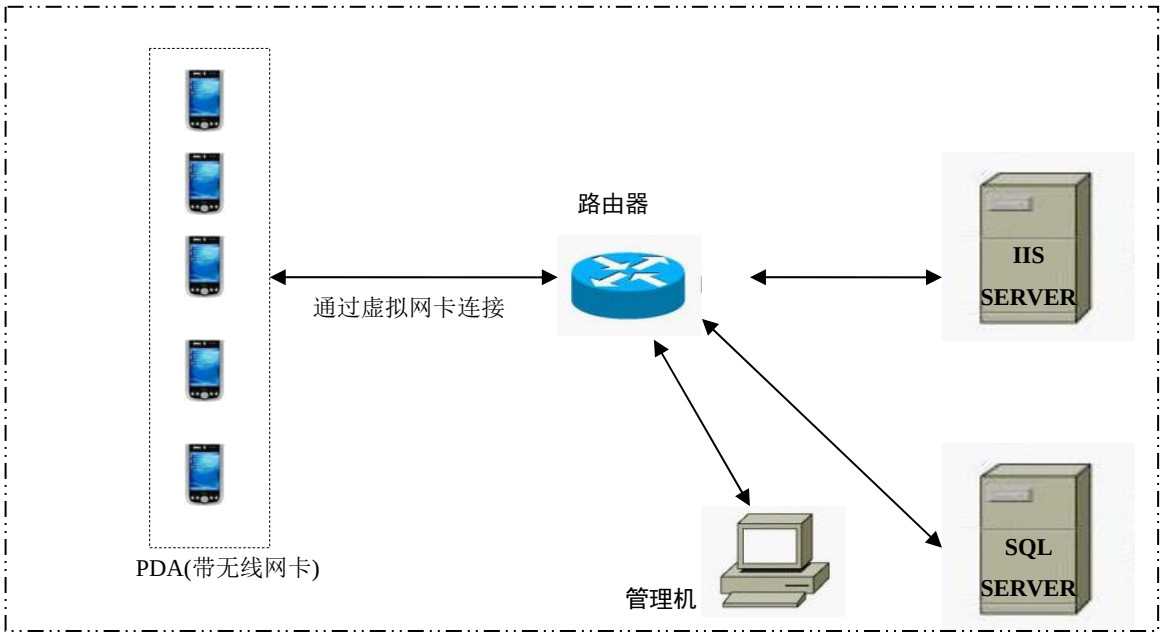

图 3-3 系统架构

3.3 基本软件环境

表 3-1 基本软件环境

	服务器端	PDA 客户端
操作系统	Windows XP SP2	Windows CE
数据库	SQL Server 2000 SP4	SQL Server 2005 Mobile Edition
IIS	IIS 5.1 以上	无

3.4 开发工具

本系统采用的开发工具为：

1. Visual Studio 2005
2. Microsoft ActiveSync 4.0
3. Virtual PC Network Driver.
3. Windows Mobile 5.0 SDK for Pocket PC
5. Windows Mobile 5.0 Pocket PC 简体中文虚拟机

3.5 环境搭建与配置

3.5.1 安装 SQL Server 2000

首先在数据库服务器上安装 SQL Server 2000，安装完成后，还需要安装 SP4 补丁。

3.5.2 安装配置 IIS 服务器

3.5.2.1 安装与配置 IIS 服务器

在系统控制面板——添加或删除程序——添加/删除 Windows 组件中选择 IIS

3.5.2.2 配置 WEB 同步向导

1、在 IIS 服务器上安装 Microsoft SQL Server 2005 Mobile Edition

进入%SystemRoot%\Program Files\Microsoft Visual Studio 8\SmartDevices\SDK\SQL Server\Mobile\v3.0 目录，顺序运行 sql2kcnsp4.msi 和 Sqlce30setupcn.msi

2、运行 WEB 同步向导

如图所示

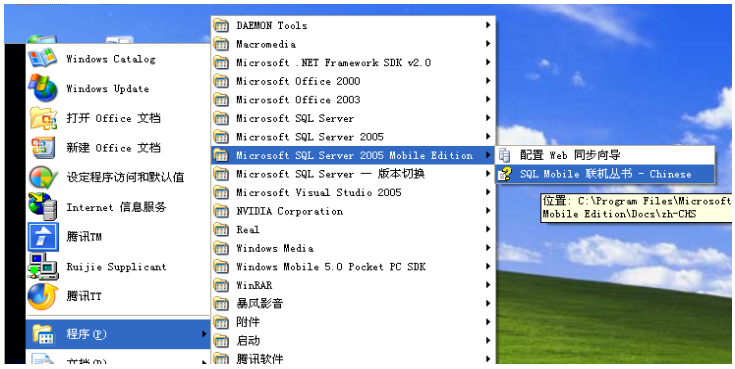

图 3-4 运行 WEB 同步向导

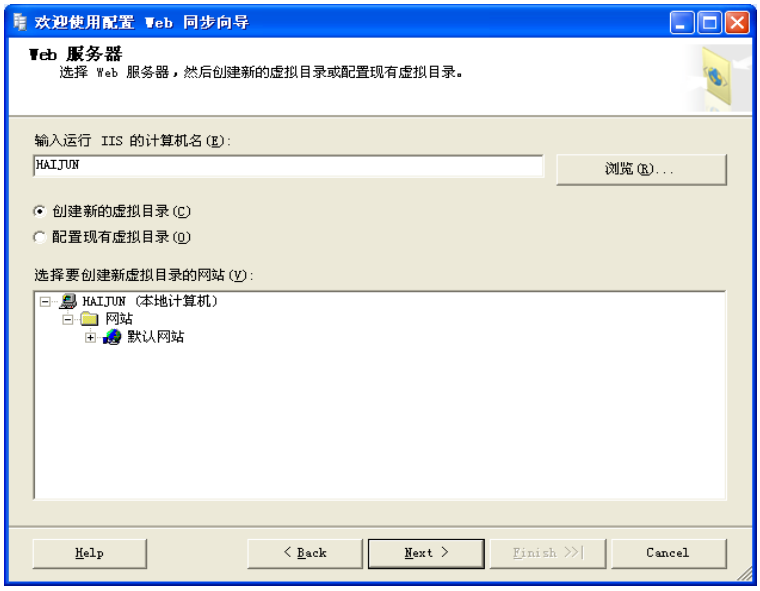

图 3-5 运行 WEB 同步向导

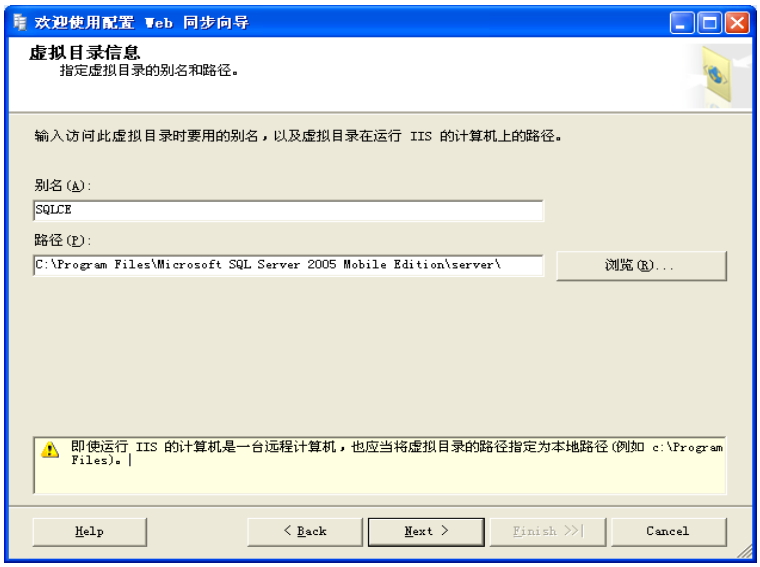

图 3-6 运行 WEB 同步向导

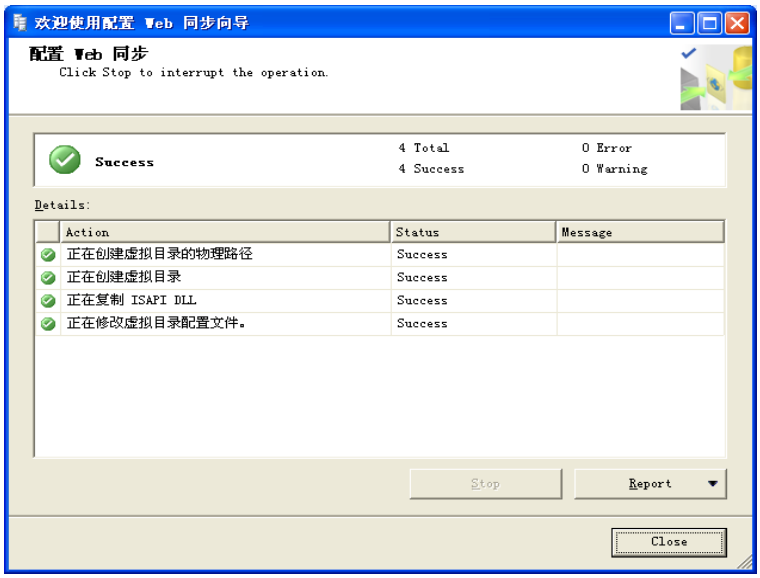

图 3-7 运行 WEB 同步向导

配置就绪后，我们可以检测 SQL Server CE Server Agent 工作是否正常，启动 IE，在地址栏中输入 Http://localhost/sqlce/sscesa30.dll，或者将 localhost 改为实际的 IIS 主机名称，浏览器返回“SQL Server Mobile Server Agent 3.0”说明 SQL Server SQL Server Mobile Server Agent 运行正常，IIS 配置正确。如图：

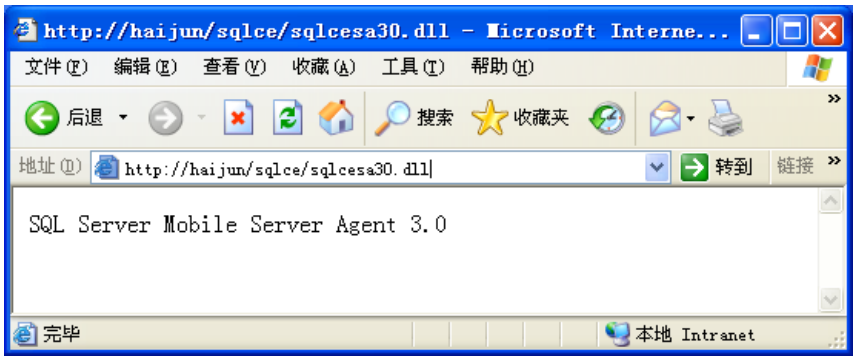

图 3-8 检测 SQL Server CE Server Agent

4 数据表设计

4.1 数据结构设计

系统数据库共有 9 张数据表，如下所示：

设计表“yunfei_class”，位置是“yunfei”中、“(local)”上

列名	数据类型	长度	允许空
class_id	int	4	
class_name	nvarchar	50	✓

图 4-1 菜品分类表

设计表“yunfei_dish”，位置是“yunfei”中、“(local)”上

列名	数据类型	长度	允许空
dish_id	int	4	
dish_name	nvarchar	50	✓
dish_unit_id	int	4	✓
dish_class_id	int	4	✓
dish_price	int	4	✓

图 4-2 系统菜品表

设计表“yunfei_menu”，位置是“yunfei”中、“(local)”上

列名	数据类型	长度	允许空
mennu_id_sys	int	4	
menu_id	nvarchar	50	
menu_table_id	int	4	✓
menu_dish_id	int	4	✓
menu_num	int	4	✓
menu_memo	nvarchar	50	✓
menu_state_id	smallint	2	✓

图 4-3 客人菜单表

设计表“yunfei_menu_state”，位置是“yunfei”中、“(local)

列名	数据类型	长度	允许空
state_id	int	4	
state_name	nvarchar	50	✓

图 4-4 菜单状态表

设计表“yunfei_preorder”，位置是“yunfei”中、“(local)”上

列名	数据类型	长度	允许空
preorder_id	int	4	
preorder_table_id	int	4	✓
preorder_date	nvarchar	50	✓
preorder_person_num	smallint	2	✓
preorder_person_name	nvarchar	50	✓
preorder_person_tel	nvarchar	50	✓
preorder_precost	int	4	✓
preorder_time	nvarchar	50	✓

图 4-5 预定表

设计表“yunfei_table”，位置是“yunfei”中、“(local)”上

列名	数据类型	长度	允许空
table_id	int	4	
table_name	nvarchar	50	✓
table_zone_id	int	4	✓
table_person_num	smallint	2	✓

图 4-6 桌台信息表

设计表“yunfei_unit”，位置是“yunfei”中、“(local)”上

	列名	数据类型	长度	允许空
🔑	unit_id	int	4	
	unit_name	nvarchar	50	✓

图 4-7 菜品单位表

设计表“yunfei_user”，位置是“yunfei”中、“(local)”上

	列名	数据类型	长度	允许空
🔑	user_id	int	4	
	user_name	nvarchar	50	✓
	user_pass	nvarchar	50	✓
	user_level	nvarchar	50	✓
	user_login	datetime	8	✓

图 4-8 用户表

设计表“yunfei_zone”，位置是“yunfei”中、“(local)”上

	列名	数据类型	长度	允许空
🔑	zone_id	int	4	
	zone_name	nvarchar	50	✓

图 4-9 分区位置表

4.2 数据表关系图

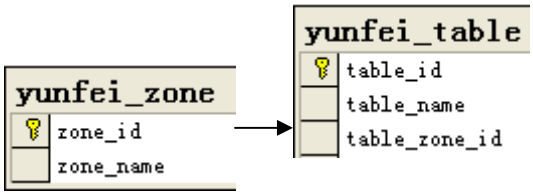

图 4-10 区号表—台桌表关系

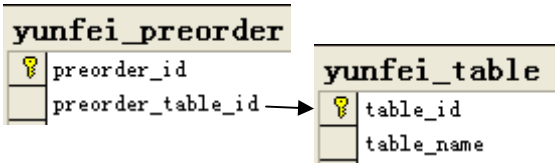

图 4-11 预定表—台桌表关系

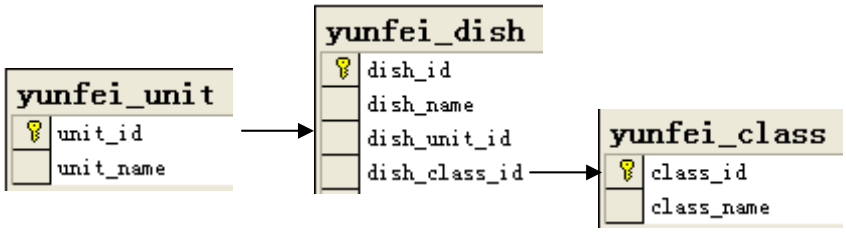

图 4-12 单位表—菜单表—类别表关系

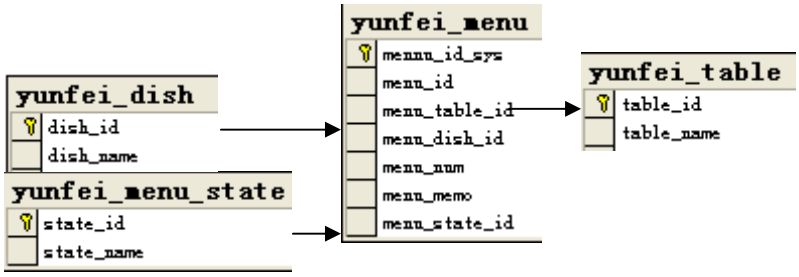

图 4-13 菜品表—状态表—菜单表—台桌表关系

5 详细功能设计

5.1 桌台管理

5.1.1 预定

选择某台后，点击“预定”，可以对该台进行预定操作，当该台所预定的时间段已经被预定，将提示预定失败，也可以查看该桌在指定日期的预定情况。

相关代码：

```
private void button1_Click(object sender, EventArgs e)
{
    SQLServerCEWrapper SSCEWrapper = new SQLServerCEWrapper();
    WaitCursor.Show(true);
    SSCEWrapper.Pull(false, "yunfei_preorder", "select * from yunfei_preorder where
preorder_table_id=" + table_id);
    string sqlCountStr = "select count(*)  from yunfei_preorder Where
preorder_table_id='" + table_id + "' and preorder_date='" + table_data.Text + "' and preorder_time='"
+ table_time.Text + "'";
    int count = SSCEWrapper.CountResult(sqlCountStr);
    if (count == 0)
    {
        string SQL = "insert into yunfei_preorder
(preorder_table_id,preorder_date,preorder_person_num,preorder_person_name,preorder_person_tel,p
reorder_precost,preorder_time)"
            + "values('" + table_id + "','" + table_data.Text + "','" +
table_num.Value.ToString() + "','" + table_person.Text + "','" + table_tel.Text + "','" +
table_cost.Text + "','" + table_time.Text + "')";
        SSCEWrapper.Submit(SQL);
```

```
                WaitCursor.Show(false);
                 MessageBox.Show("预定成功！");
            }
            else
            {
                WaitCursor.Show(false);
                 MessageBox.Show("此桌该时间已经被预定！");
            }
        }
```

5.1.2 开台

选择某台后，点击“开台”，若该台为空，则允许进一步操作：选择客人数量，完成开台操作。

相关代码：

```
        private void button1_Click(object sender, EventArgs e)
        {
            WaitCursor.Show(true);
            SQLServerCEWrapper SSCEWrapper = new SQLServerCEWrapper();
            string SQL = "update yunfei_table set table_person_num=" +
table_person_num.Value.ToString() + " where table_id=" + table_id;
            SSCEWrapper.Submit(SQL);
            return_value = int.Parse(table_person_num.Value.ToString());
           WaitCursor.Show(false);
            Close();
        }
```

5.1.3 清台

选择某台后，点击“清台”，若该台不为空，则允许进一步操作：若该台尚未结账，则提示应先结账，否则进行清空信息操作，此台随后可以继续进行开台操作。

相关代码：

```
        private void menuItem4_Click(object sender, EventArgs e)
        {
            string SQL;
            int table_id = 0;
            if (lvwTablelist.FocusedItem != null)
            {
                int table_person_num =
int.Parse(lvwTablelist.FocusedItem.SubItems[3].Text.ToString());
                if (table_person_num == 0)
                {
                    MessageBox.Show("此桌为空桌，请另外选择！");
                    return;
                }
                else
                {
```

```
                        table_id =
int.Parse(lvwTablelist.FocusedItem.SubItems[1].Text.ToString());
                        SSCEWrapper.Pull(false, "yunfei_menu", "select * from yunfei_menu where
menu_table_id=" + table_id);

                        string clearSQL = "select count(*) from yunfei_menu  where
menu_state_id<>0 and menu_table_id=" + table_id;
                        int count = SSCEWrapper.CountResult(clearSQL);//得到菜单状态
                        if (count != 0)
                        {
                            MessageBox.Show("请先结账，再清台！");
                            return;
                        }
                        else
                        {
                            DialogResult ResultDlg = MessageBox.Show("确定清台？", "清台确认",
MessageBoxButtons.OKCancel, MessageBoxIcon.Question, MessageBoxDefaultButton.Button2);
                            if (ResultDlg == DialogResult.OK)
                            {
                                WaitCursor.Show(true);
                                SQL = "update yunfei_table set table_person_num=0 where table_id="
+ table_id;
                                SSCEWrapper.Submit(SQL);
                                SQL = "delete from yunfei_menu  where menu_table_id=" + table_id;
                                SSCEWrapper.Submit(SQL);
                                SSCEWrapper.Pull(false, "yunfei_table", "select * from
yunfei_table");
                                UpdateComoBox();
                                UpdateListView();
                                WaitCursor.Show(false);
                                MessageBox.Show("清台成功！");
                                table_id = 0;
                            }
                        }
                    }
                }
            }
```

5.1.4 并台

选择某台后，点击“并台”，若该台不为空，则允许进一步操作：选择要并入的台号，原台清空，并入的台更新信息，菜单合并。

相关代码：

```
            private void button1_Click(object sender, EventArgs e)
            {
```

```
                WaitCursor.Show(true);
                int table_person_num_new = 0;
                //得到选择的桌台的当前人数
                for (int i = 0; i < ds.Tables["yunfei_table"].Rows.Count; i++)
                {
                    if (ds.Tables["yunfei_table"].Rows[i][0].ToString() ==
this.combo_table.SelectedValue.ToString())
                    {
                       table_person_num_new =
int.Parse(ds.Tables["yunfei_table"].Rows[i][3].ToString());
                        break;
                    }
                }
                int ands = table_person_num_new + int.Parse(table_person_num.ToString());
                String SQL = "update yunfei_table set table_person_num=" + ands + " where table_id="
+ this.combo_table.SelectedValue;
               //两台人数相加
                SSCEWrapper.Submit(SQL);
                SQL = "update yunfei_table set table_person_num=0 where table_id=" + table_id;
                SSCEWrapper.Submit(SQL);
                SQL = "update yunfei_menu set menu_table_id=" + this.combo_table.SelectedValue +
" where menu_table_id=" + table_id;
                //置菜单
                SSCEWrapper.Submit(SQL);
                return_id = int.Parse(combo_table.SelectedValue.ToString());
                WaitCursor.Show(false);
                this.Close();
            }
```

5.1.5 分台

选择某台后，点击“分台”，若该台不为空，则允许进一步操作：选择要分出的台号和要分出的人数，原台数据更新，菜单表不变，新台亦更新信息，无菜单信息。

相关代码：

```
            private void button1_Click(object sender, EventArgs e)
            {
                WaitCursor.Show(true);
                String SQL = "update yunfei_table set table_person_num=" + num_div.Value + " where
table_id=" + this.combo_table.SelectedValue; //新台人数
                SSCEWrapper.Submit(SQL);
                int divs_old =table_person_num- (int)num_div.Value;//剩余的人数要赋给原台
                SQL = "update yunfei_table set table_person_num="+divs_old+" where table_id=" +
table_id;
                SSCEWrapper.Submit(SQL);
                //不转菜单
```

```
            return_id = int.Parse(combo_table.SelectedValue.ToString());
            WaitCursor.Show(false);
            this.Close();
        }
```

5.1.6 转台

选择某台后，点击“转台”，若该台不为空，则允许进一步操作：选择要转入的台号，原台信息转到新台上。

相关代码：

```
        private void button1_Click(object sender, EventArgs e)
        {
            WaitCursor.Show(true);
            String SQL = "update yunfei_table set
table_person_num="+table_person_num.ToString()+" where
table_id="+this.combo_table.SelectedValue;
            SSCEWrapper.Submit(SQL);
            SQL = "update yunfei_table set table_person_num=0 where table_id=" + table_id;
            SSCEWrapper.Submit(SQL);
            SQL = "update yunfei_menu set menu_table_id="+this.combo_table.SelectedValue+"
where menu_table_id=" + table_id;
            SSCEWrapper.Submit(SQL);
            return_id = int.Parse(combo_table.SelectedValue.ToString());
            WaitCursor.Show(false);
            this.Close();
        }
```

5.2 菜品管理

5.2.1 点菜

选择某台后，点击“点菜”，若该台不为空，则允许进一步操作：系统显示当前菜品列表，选择某项，确定后会出现具体的菜品口味及做法选择。确定后将出现成功提示。接着返回菜品列表，可以重复以上操作，直到顾客满意后，可以点击“已点菜品”按钮进入已点未发送菜品列表，可以删除不满意的菜品，点击“全部发送”可以将菜单发送至后台数据库。

相关代码：

```
        private void button1_Click(object sender, EventArgs e)
        {
            string memo1 = null;
            string memo2 = null;
            string memo = null;
            for (int i = 0; i < listView1.Items.Count; i++)
            {
               if (listView1.Items[i].Checked)
                { memo1 +=listView1.Items[i].Text+","; }
            }
            for (int i = 0; i < listView2.Items.Count; i++)
```

```
                {
                    if (listView2.Items[i].Checked)
                    { memo2 +=listView2.Items[i].Text + ","; }
                }
                memo = memo1 + memo2;
                WaitCursor.Show(true);
                SQLServerCEWrapper SSCEWrapper = new SQLServerCEWrapper();
                string SQL = "insert into yunfei_menu
(menu_id,menu_table_id,menu_dish_id,menu_num,menu_memo,menu_state_id)"
                    + "values('"+gen_menu_id(table_id)+"','" + table_id + "','" + dish_id + "','"
+num_menu.Value.ToString() + "','" + memo + "',0)";
                SSCEWrapper.Submit(SQL);
                WaitCursor.Show(false);
                MessageBox.Show("点菜成功！");
                this.Dispose();
            }
```

5.2.2 未发送菜品

选择某台后，点击“点菜”，若该台不为空，则显示当前桌台的未发送菜品，可以删除不满意的菜品，点击“全部发送”可以将菜单发送至后台数据库。

相关代码：

```
            private void button1_Click(object sender, EventArgs e)
            {
                WaitCursor.Show(true);
                String SQL = "update yunfei_menu set menu_state_id=1  where menu_table_id=" +
table_id; //menu_state=1 表示已发送
                SSCEWrapper.Submit(SQL);
                dg_unserver.DataSource = null;
                //SSCEWrapper.Pull(false, "yunfei_menu", "select * from yunfei_menu where
menu_state_id=0 and  menu_table_id=" + table_id);
                UpdateDataGrid();
               WaitCursor.Show(false);
                MessageBox.Show("发送成功！");
            }
            //menu_state 状态说明：
            //0   已点
            //1   已发送
            //2   已确认
            //3   传菜中
            //4   已上
```

5.2.3 已点全部菜品

选择某台后，点击“已点全部菜品”，若该台不为空，则显示当前桌台的全部已点菜品，可以显示各个菜品的相关信息，如价格、当前状态及总价等。

5.2.4 已上菜品

选择某台后，点击“已上菜品”，若该台不为空，则显示当前桌台的全部已上菜品，可以显示各个菜品的价格及总价。

5.2.5 未上菜品

选择某台后，点击“未上菜品”，若该台不为空，则显示当前桌台的全部未上菜品，可以显示各个菜品的价格及总价。

5.3 结账

选择某台后，若该台不为空，则显示当前桌台的全部菜品信息及总共价格，点击“结账”可进行结账操作，完成后可以进行“清台”操作。

相关代码：

```
private void button1_Click(object sender, EventArgs e)
{
    DialogResult ResultDlg = MessageBox.Show("确定结账？", "结账确认",
MessageBoxButtons.OKCancel, MessageBoxIcon.Question, MessageBoxDefaultButton.Button2);
    if (ResultDlg == DialogResult.OK)
    {
        WaitCursor.Show(true);
        string SQL = "delete from yunfei_menu  where menu_table_id=" + table_id;
        SSCEWrapper.Submit(SQL);
        UpdateDataGrid();
        WaitCursor.Show(false);
        MessageBox.Show("结账完成！");
        Close();
    }
}
```

6 测试和发布

6.1 程序测试

打开 SQL SERVER2000 数据库服务，启用 IIS 服务器，打开 Pocket PC2003 仿真程序，运行所编译的程序，设定网络参数，同步数据，登录系统，经测试，系统功能符合设计要求。

程序截图：

图 6-1 登录界面

图 6-2 网络设置界面

图 6-3 数据同步界面

图 6-4 台桌列表界面

图 6-5 桌台菜单界面

图 6-6 预定界面

图 6-7 开台界面

图 6-8 并台界面

图 6-9 分台界面

图 6-10 转台界面

图 6-11 点菜界面

图 6-12 已点菜品界面

图 6-13 结账界面

图 6-14 清台界面

6.2 程序发布

6.2.1 为解决方案添加智能设备 CAB 项目

打开现有的智能设备项目，并确保“解决方案资源管理器”可见。

在“文件”菜单上指向“添加”，然后单击“新建项目”。

出现“添加新项目”对话框。

在左侧的“项目类型”窗格中展开“其他项目类型”节点，再单击“安装和部署”。

在右侧的“模板”窗格下选择“智能设备 CAB 项目”。

这是唯一可用于智能设备的 CAB 项目类型。其他项目类型仅用于桌面解决方案。

在“名称”框中，键入 Yunfei_ClientSetup，然后单击“确定”。

此 CAB 项目即会添加到的解决方案，并显示在“解决方案资源管理器”中。现在显示“文件系统编辑器”的两个窗格。

6.2.2 为 CAB 项目添加设备项目应用程序

在“文件系统编辑器”的左窗格中，选择“应用程序文件夹”节点，以指定下列步骤中选择的文件将安装到目标设备上的此文件夹中。

如果“文件系统”编辑器不可见，请在“解决方案资源管理器”中右击 CAB 项目名称，选择“视图”，再单击“文件系统”。

在 Visual Studio 中的“操作”菜单上指向“添加”，然后单击“项目输出”。

在“添加项目输出组”对话框中，从“项目”下拉列表中选择”Yunfei_Client”。

从输出列表中选择“主输出”，然后单击“确定”。

6.2.3 生成 CAB 文件

在“解决方案资源管理器”中右击“Yunfei_ClientSetup”，再单击“生成”。然后在“文件”菜单上单击“全部保存”。

6.2.4 将 CAB 文件部署到设备上

在“Windows 资源管理器”中定位到存储此解决方案的文件夹。可以在解决方案的“Yunfei_ClientSetup\Release”文件夹中找到此 CAB 文件。

将 CAB 文件复制到与 ActiveSync 4.0 或更高版本连接的设备上。

此 CAP 必须部署在已经安装了.NET Compact Framework version 2.0 的目标机器上。

7 结束语

至此，已经完成本次毕业设计的全部工作，达到了选题的要求，但仍存在许多不足。

通过这段时间的努力，对我这 5 年来的学习有了系统的总结，对我自己感兴趣的知识也有了一些进步。网络这个时代的主流已经发展的越来越快了，各行各业都在积极地向信息科技靠拢，移动服务的应用也必将越来越广泛。希望此次毕业设计能为自己今后所从事的职业奠定一个基础。

参考文献：

[1] 颜友宁. NET compact FrameWork 移动开发指南 [M].北京: 清华大学出版社, 2006. 115－138.

[2] 精英科技. Windows CE 权威指南 [M].北京: 中国电力出版社, 2001. 25－48.

[3] [美]Douglas Boling. MICROSOFT WINDOWS CE 程序设计 [M].北京: 北京大学出版社, 1999. 96－120.

致　谢

感谢肇庆学院五年来对我的辛勤培育、感谢计算机科学与技术系为我提供了良好的学习环境、感谢领导、老师们几年来对我无微不至的关怀和指导，让我得以在这几年中学到很多有用的知识。在此，我还要感谢在班里同学和朋友，感谢你们能在我遇到困难的时候伸出援助之手，正是由于你们的支持和鼓励，我才有更大的动力，本论文才能顺利完成。

本论文是在我的指导老师杨帆老师的亲切关怀和悉心指导下完成的。从课题的选择到项目的最终完成，杨帆老师都始终给予我细心的指导和不懈的支持。在此谨向杨老师致以诚挚的谢意和崇高的敬意。谢谢！

Analysis and design of the Mobile Servicing System

——Wireless Client Ordering System

Fan Haijun

Abstract: Based on the study of intelligent mobile wireless client Ordering System applications, The purpose is, by the latest information technology 802.11b wireless PDA and smart, Food industry to bring a new service means and management philosophy, business process optimization, greatly increase the restaurant's service quality and corporate image, maximize the value of enterprises while enabling their lowest cost, and ultimately improve their own food and drink industry's competitive edge.

Key words: Mobile Service; PDA; Dining; Wireless food ordering