**Cell Men**

**Nathan Silvester and Richard Olivera**

**Application Summary:**

For this project our team decided to make a website for Cell Phones which we are very passionate about. Although the company does not exist we set out to build something that would work, and look great if it were ever to be posted for the general public.

The idea behind the website was to create a “One Stop Cell Shop” where people could buy cell phones and tablets, sell cell phones and tablets, and purchase repair parts for their cell phones and tablets. Our main target audience are those people that want to upgrade their phones but are stuck in contracts or people who simply want to repair their broken phone. We envision people going to our website and receiving a reasonable offer for their current device and using that money to purchase another device from our inventory.

As for the functionality, it is safe to say that we wanted to go with a simplistic design that is easy to use. That is precisely what we got, our navigation is fairly intuitive and people can easily find what they are looking for. Apart from those users looking to sell, buy, or purchase repair parts, our intended users include admins. An admin can have the option to review the devices that are in inventory simply by using their login credentials.

**Application Guide:**

1. Master pages
2. .aspx pages
3. User controls
4. Databases
   1. CellDatabase.mdf
   2. ASPNETDB.MDF
5. Other supporting files (Themes & Skins, XML files, sitemap files, etc.)

|  |
| --- |
| **Master Pages**  1. *MasterPage.master*: The primary master page for the site. This page contains header and footer content Navigation Links. The master page is used by *About.aspx, Contact.aspx, Default.aspx, Error.aspx, faq.aspx, forgotpassword.aspx, Inventory.aspx, Login.aspx, Manufacturer - repair.aspx, Manufacturer - sell.aspx, Manufacturer.aspx, Model - Repair.aspx, Model Sell.aspx, Model.aspx, newuser.aspx, Phone - Repair.aspx, Phone – Sell.aspx, Phone.aspx, Type - Repair.aspx, Type - Sell.aspx, Type.aspx.*  **.aspx Pages**   1. *About.aspx* – This page displays information about the company and it has pictures of our team. 2. *Contact.aspx* – This Page displays the map of the location of the business along with a form that will email the given address. (Does not work because it needs an smtp server) 3. *Default.aspx* – This Page displays the home page of the website. This page includes a dynamic carrousel that shows the latest device that has been entered into our inventory if it is in perfect condition. Along with the carrousel it has services we offer and links to those pages. 4. *Error.aspx*– This Page displays when there is an error on the page. (Hopefully that won’t happen.) 5. *faq.aspx*– This Page displays a few answers to commonly asked questions 6. *forgotpassword.aspx* – This Page displays a control that can help you retrieve your password 7. *Inventory.aspx* – This Page displays the Device Inventory. This page is dynamic and is pulled from the database. 8. *Login.aspx* – This Page displays the login control so that admins along with users could login to their accounts. 9. *Manufacturer - repair.aspx*– This Page displays the manufacturer part of the repair path where users will go to repair their devices. It will display all the distinct manufacturers in the database. 10. *Manufacturer - sell.aspx* – This Page displays the manufacturer part of the sell path where users will go to sell their devices. It will display all the distinct manufacturers in the database. 11. *Manufacturer.aspx* – This page displaysthe manufacturer part of the buy path where users will go to buy devices. It will display all the distinct manufacturers in the database. 12. *Model - Repair.aspx –* This page displays the model of the phone in the repair path. It will display all distinct phone models in the database. 13. *Model - Sell.aspx -* This page displays the model of the phone in the repair path. It will display all distinct phone models in the database. 14. *Model.aspx -* This page displays the model of the phone in the repair path. It will display all distinct phone models in the database. 15. *newuser.aspx –* This page will aid in creating a new user and adding it to the Membership table. 16. *Phone - Repair.aspx –* This page will is the end of the path and will show the available parts for the selected phone. 17. *Phone – Sell.aspx –* This page is the end of the path and will display the options to sell the users phone to our company. 18. *Phone.aspx –*  This page is the end of the path for buying a device. It will display the selected device and the different conditions. 19. *Type - Repair.aspx -*  This will display either phone or tablet for the Repair path. 20. *Type - Sell.aspx -* This will display either phone or tablet for the sell path. 21. *Type.aspx -* This will display either phone or tablet for the buy path.   **User Controls**   1. *SellItem.ascx*: A user control that allows the user to select Device Condition and upload an image and insert in to the database.   **Databases**   1. *aspnetdb*: A MS SQL Server Express 2008 database. This database manages all membership, role and profile information for the application. 2. *CellDatabase*: A MS SQL Server Express 2008 database. This database stores devices, device types, inventory, manufacturers, and parts.   **Other Supporting Files**  *Web.SiteMap:* An XML file that contains a map of the site. Used by the navigation controls. |

**Role Descriptions:**

**Admin:** Administrator role that can make changes to site

**Users:** General role for the people who want to sell a phone on the site

**Project Requirements:**

|  |  |
| --- | --- |
| **MIS 5050 Project Requirements** | **Project Component(s)** |
| 5 interlinked Web forms (.aspx pages) that display dynamically-generated content **per team member** | 1. Default.aspx 2. Type.aspx 3. Manufacturer.aspx 4. Model.aspx 5. Inventory.aspx 6. Phone.aspx 7. Type - Repair.aspx 8. Manufacturer - Repair.aspx 9. Model - Repair.aspx 10. Phone - Repair.aspx 11. Type - Sell.aspx 12. Manufacturer - Sell.aspx 13. Model - Sell.aspx 14. Phone - Sell.aspx |
| At **least** 5 database transactions (a transaction includes selecting, inserting, updating, or deleting data from the database) **per team member** | 1. Default.aspx 2. Type.aspx 3. Manufacturer.aspx 4. Model.aspx 5. Inventory.aspx 6. Phone.aspx 7. Type - Repair.aspx 8. Manufacturer - Repair.aspx 9. Model - Repair.aspx 10. Phone - Repair.aspx 11. Type - Sell.aspx 12. Manufacturer - Sell.aspx 13. Model - Sell.aspx 14. Phone - Sell.aspx |
| User/Role management for at least 2 roles | Roles Supported:   1. Admin 2. Users |
| Preservation of state (may include cookies, session state, application state, cross-page posting, or querystring) | When an employee logs on to the system, her employeeID is stored in a session variable. This variable is accessed on all other employee pages in the system. |
| 10 different ‘basic’ ASP.NET Web controls (those that map directly to an html tag) | Error.aspx:   1. Panel1 (Panel)   Something.aspx  Phone – Repair.aspx:   1. CheckBoxList1 (CheckBoxList1) 2. Image1 (Image) 3. Label1 (Label)   Model – Repair.aspx:   1. ImageButton1 (ImageButton) 2. LinkButton1 (LinkButton)   SellItem.ascx   1. DropDownList1 (DropDownList) 2. FileUpload1 (FileUpload) |
| 5 different ‘advanced’ ASP.NET Web controls (those that do not map  directly to an html tag) | Default.aspx:   1. SqlDataSource1 (SqlDataSource) 2. DataList1 (DataList)   Login.aspx:   1. Login1 (Login)   SellItem.ascx:   1. ConfirmButtonExtender1 (Ajax Toolkit ConfirmButtonExtender)   ChangePassword.aspx:   1. ChangePassword1 (ChangePassword)   ForgotPassword.aspx:   1. PasswordRecovery1 (PasswordRecovery) |
| Input validation for **all** free-form user entries | SellItem.ascx:   1. RequiredFieldValidator1 – Make sure they select a condition. 2. RequiredFieldValidator2 – Make sure they select an image to upload. |
| 1 user control | 1. SellItem.ascx: A user control that allows the user to select Device Condition and upload an image and insert in to the database. |
| 1 navigation control | 2. SiteMapPath1: A SiteMapPath control included on SiteMaster.Master |