**Capabilities Covered:**

* Implement MVC framework using .Net MVC 5
* Use Entity framework to implement persistence layer of an application

Gadget Store Application

# Gadget Store Application

Design and implement a system that manages and organizes your favourite gadgets. It enables users to store into and search for his/her wish list from the store.

* Home page contains a promo video and carousel of latest gadgets
* The home page contains the following links
  + Add a gadget
  + Create and Search Wish-List

# System Features

|  |  |
| --- | --- |
| **Use Case #1** | **Add a Gadget** |
| Trigger | User clicks on “Add a gadget” link on the home page |
| Pre-Conditions | - |
| Post-Conditions | The complete information about the gadget is stored in the application database |
| UI Screen Details | The input form contains the following fields   |  |  |  | | --- | --- | --- | | Gadget ID | AlphaNumeric |  | | Gadget Title / Name | Text value |  | | Company | Text Value |  | | Image | Image |  | | Category | Text Value |  | | Quantity | Number |  | | Price | Money |  | |
| Main Flow | The user provides the required details in the form and Clicks “Save” button. On successfully saving the gadget details, user is forwarded to the home page along with the message “Gadget details have been saved successfully” |
| Data Validations | * Input data has to be validated for numeric and date data fields. * All fields are mandatory. |
| Business Rules | ID = first 3 letters of category and 4 digits auto increment number |
| Exception Flows | * In case of any failure in Data Validation or Business Rules, user should be brought back to the screen along with the error message “The value provided for the field …. Is incorrect or incomplete” |
| Alternate Flows | * When the user selects “Cancel” button, user should be taken back to the home page |

|  |  |
| --- | --- |
| **Use Case #2** | **Create and Search Wish-List** |
| Trigger | User clicks on “Create and Search Wish-List ” link on the home page |
| Pre-Conditions | Gadget details are in the database |
| Post-Conditions | Details of gadget **Wish-List** are stored in database |
| UI Screen Details | * ‘**Gadget list**’ table needs to be displayed. This displays the list of **all** the gadgets available in the database. Each gadget display will have a button called “Add to Wish-List” beside it. * ‘**Wish-List’** form needs to be displayed the gadgets would be selected from the ‘Gadget list’ table.   + The form will have delete button. delete button, on click, will validate user entry and delete the details from wish-list |
| Main Flow | When the page is loaded the ‘Gadget List’ form and the ‘Wish-List’ table are displayed one below the other.  The ‘Wish-List’ table is initially loaded with details in Database. Once the data is entered through the ‘Add to Wish-List’ button and stored successfully, report with Wish-List also could be displayed |
| Data Validations | * Input data has to be validated for numeric and image data fields. |
| Business Rules | * ID should be unique in the system |
| Exception Flows | In case of any failure in Data Validation or Business Rules, user should be brought back to the same screen along with a relevant error message |
| Alternate Flows | * When the user selects “Close” button, user should be taken back to the home page |