**EE4717/IM4717 Web Application Design - Project Report**

**Design project group number: F33-DG07**

**Team members: Chua You Liang, Neo Zhen Ting**

**Project Title: Beep and Geek Summary of Project**:

The objective of this project is to create a web portal for online purchase of consumer electronics and/or IT products. This web portal will provide a one-stop solution for all tech-savvy customers as well as tech-lovers out there. The web portal will be selling products such as phones, laptops, cables and many more. Customers will enjoy an easy, smooth and fuss-free experience from browsing products to selection to making payments. They will also be able to track the status of their orders via the web portal or email.

# Application Requirements and Specifications

* Allow customers to browse products based on categories
* See details of each product
* Make order + payment
* Receive confirmation email
* Check delivery status

# Functional Requirements and Specifications

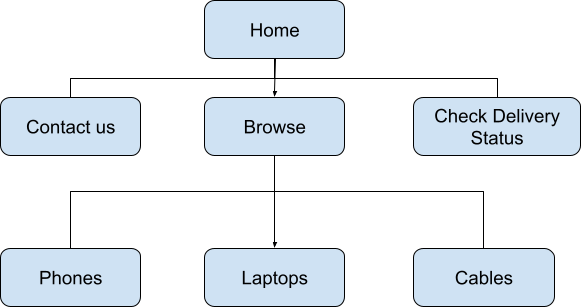
* Home page: Short paragraph describing the website, show featured items

(popular/promo items), button to “Go into store” which links to the catalogue

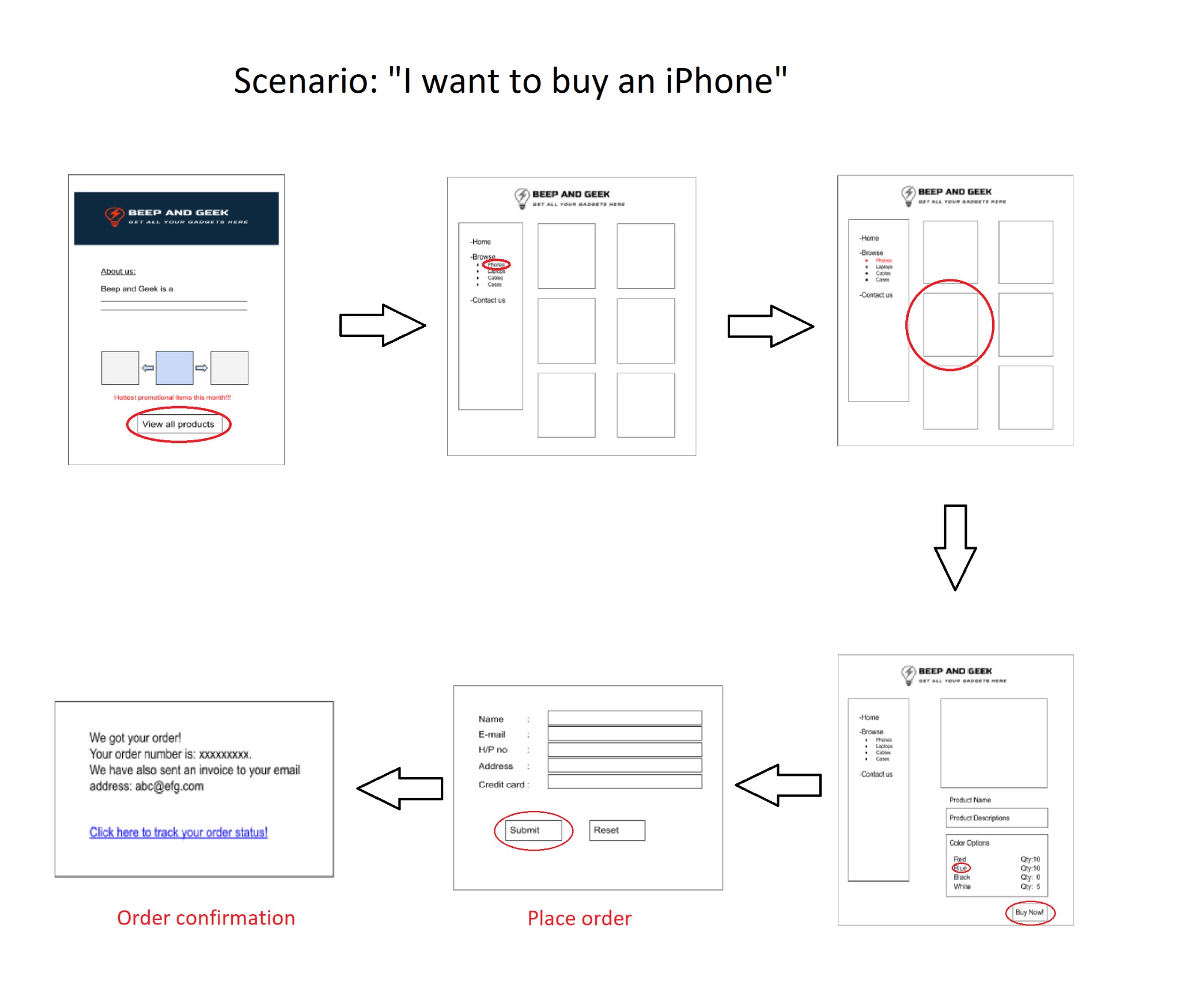
* Catalogue: allows user to browse all products (image + name)
* Browse by category: subset of catalogue - allows users to browse products in certain category (phone, laptops, smartwatch, earphones)
* Product page: Lists product image, name, description, stock available, color options, buy button. Buy button links to purchase form.
* Purchase form: Allows users to enter personal details, purchase amount, and payment details. Can implement as pop-up or separate page. Links to confirmation page upon submission. Confirmation page includes order no. for tracking purposes
* Check delivery status: Allows users to enter order no. to track delivery status (order received, ready for delivery, out for delivery, delivered).
* Side navigation bar: Includes link for Home, Catalogue, Browse by category, Contact us

# Design of the web application

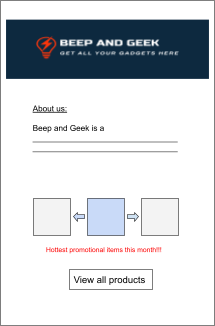
1. Site Map



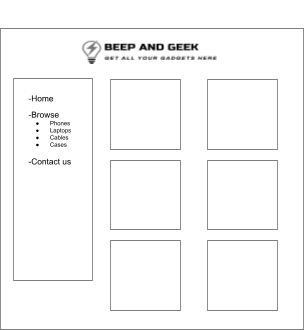
1. Storyboards:



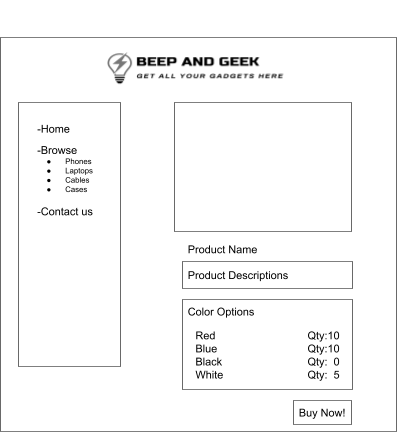
1. Wireframes:
   1. Home page:



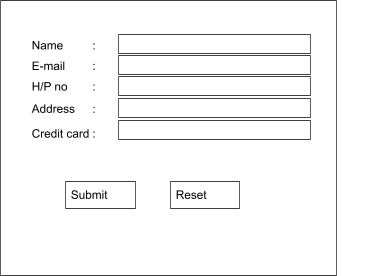
* 1. Catalogue:



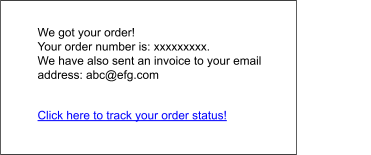
* 1. Product page:



* 1. Purchase form:



* 1. Confirmation page:



**Database Schema:**

Customers (CustomerID, Name, Address, Phone, Email)

Product\_Main (ProductName, Category, ProductImage)

* Image stores the path to the image of the particular product.

Products (ProductID, ProductName, VariantImage, Price, Stock, Color)

* Different colour options will have different ProductID, but same ProductName
* Eg: Iphone 13 Blue => ProductID = A01, ProductName = Iphone 13
* Iphone 13 Gold => ProductID = A02, ProductName = Iphone 13
* This allows different colour options to have different price and stock available, but we are still able to group all the different colour options under 1 product name.

Orders (OrderID, *CustomerID*, Amount, TransactionTime)

* OrderID will be given to customer for tracking purposes

Order\_Items (OrderID, *ProductID*, Quantity)

\*Order\_Reviews (ProductID, Reviews)

- \*Optional\*

Project MUST contain at least:

1. One (1) home page and four(4) but no more than ten(10) additional content pages  
o Every page must contain appropriate text and images to convey the desired  
messages to the target audience  
o Every page must contain appropriate page titles

* Pages:
  1. Index.html
  2. Category/all.php
  3. Category/phones.php
  4. Category/laptops.php
  5. Category/earphones.php
  6. Product.php (product page, takes ProductName as parameter, eg product.php?productname=iphone13 gives the product page for iphone 13)
  7. Tracking.html
  8. Contact.html

2. One (1) table display content effectively

* Color options on product pages use table. Alternatively, the two columns of product on the catalogue can be done using table.

3. One (1) form (four (4) fields minimum), together with server-side processing of form  
data and transactions with database.

* Purchase form containing 4 fields: Name, E-mail, H/P no., and Address.
* Database transactions:
  1. Insert to Customers table
  2. Insert to Order, Order\_Items table
  3. Update Products.Stock

4. Transactions with database must involve SQL commands such as Select, Insert,  
Update.

* See Criteria 3

5. One (1) server-side generated web page.

* Here is my suggestion:
  1. Catalogue (including browsing by each category) is generated server-side. Php script goes through each entry in the Products table and then renders them on the catalogue page.
  2. Every product pages are also server-side generated. PHP script queries the Products table to get the image, colour options, stock available, etc associated with a particular ProductName.
  3. Advantages:
     + Need to write less code. For catalogue and browse by category pages, everything should be the same, except for filtering the Products table by different Category.
     + Same for each product pages, where the only difference is the ProductName.

6. Data entries not validated by HTML5 should be validated by JavaScript except for  
those which can only be validated by server-side scripts together with the database.

* Purchase form:
  1. Validation done on Name, Email, H/P no, and Address field using Javascript/HTML.
  2. Validation done using PHP to ensure that items ordered have stock >= quantity ordered.

7. Data entries not validated on client-side must be validated on the server-side using  
PHP.

* See Criteria 6

8. Form must reside on project site (link to an external form is not allowed)

* Not sure what this means

9. One (1) external style sheet (CSS) file which contains a minimum of four (4) styles  
should be found in the web folders

* Catalogue, Browse by Category page, and Product pages can share a main.css style sheet including styling for the top banner, side navbar, and footer.