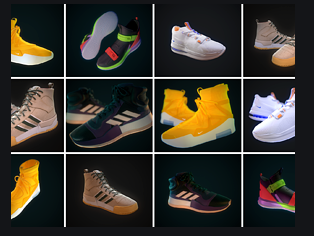
2020

Technical Specification



x19175108 Geneci Ui Fhatharta

x19176414 Katarzyna Casserly

x19141815 Ronan Behan

x19139306 Rory Cleary

Server Side Development

Shop@Web Application

Contents

[1 Session 1 - Project Overview 3](#_Toc57052108)

[1.1 Description of the Shoes@Web Application 3](#_Toc57052109)

[2 Session 2 – Technical Specification 3](#_Toc57052110)

[2.1 Model 3](#_Toc57052111)

[2.2 Key Users Flow 4](#_Toc57052112)

[2.3 Shop@Web Flow from visitor to User/Customer 4](#_Toc57052113)

[2.4 Functional Requirements for Non-registered User 5](#_Toc57052114)

[2.5 Functional Requirements for a Registered User 5](#_Toc57052115)

[2.6 Functional Requirements for an Administrator Manager 6](#_Toc57052116)

[2.7 Functional Requirements for an Administrator User 6](#_Toc57052117)

[2.8 Non-Functional Requirements 6](#_Toc57052118)

[2.9 Our Web Application will have the following pages available: 6](#_Toc57052119)

[2.10 Bibliography 8](#_Toc57052120)

# Session 1 - Project Overview

## Description of the Shoes@Web Application

The scope of this project is to design and implement an e-commerce website for Shoes@Web. Ruby On Rails will be the designated language used to create the application. Rails is suitable for large and small web apps, future proofed to expand easily. It is relatively quick to create a site due to its existing pre-defined configurations. There are already three main environments included: development, testing and production. Ruby On Rails is based on Model-View-Controller (MVC) architecture. It allows one to create a very responsive and flexible application.

The Shoes@Web application will be responsive, and it will automatically adjust to work on all screen resolutions and devices (desktop, laptop, tablet & phone). A relational database to store all data (products, customer, cart, order etc.) will need to be created.

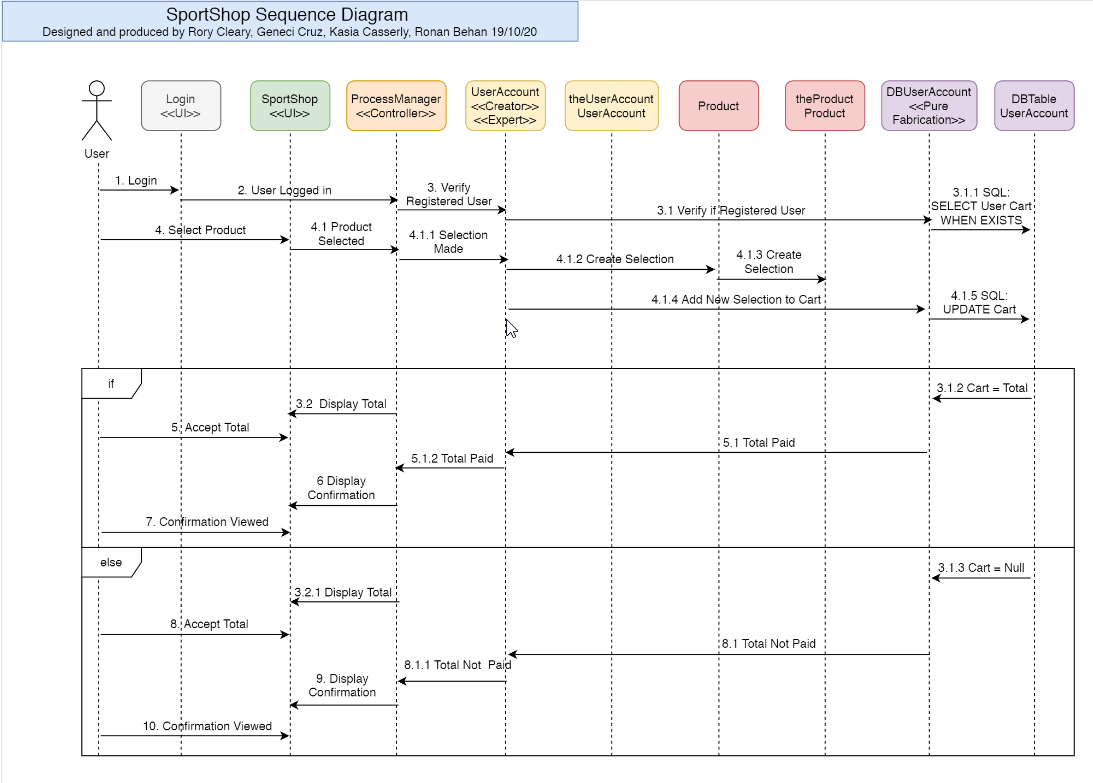
Shoes@Web will be an interactive application allowing response on request from the web browser. The primary purpose of the website will be selling products online. The application allows users to browse through the website in order to search, select and purchase products. If a user decides to purchase a product, they will first need to create an account. Shoes@Web will store customer details (address and order details). It will facilitate safe payment processing using major credit cards, debit cards or PayPal.

Shoes@Web exists to increase sales, work as a marketing tool to improve customer service and to obtain data regarding general product demand.

# Session 2 – Technical Specification

## Model

## Key Users Flow



## Shop@Web Flow from visitor to User/Customer

The Shoes@Web application main objective is to allow a registered user to process online sport shoes products shopping. To allow it the following functionalities must be developed.

* A fully functional Shoes@Web application
* Define access control to different users
* A top menu bar with home, products pages, contact us pages contained in it.
* A side navigation bar with the product pages by categories
* A database to hold information inputted into the website.
* An easy to use website.
* An easy to use menu bar.
* Flash message informing user: “Item successfully added to cart”.
* Flash message informing user: “Please agree with our term conditions”.
* Registration page – The Shoes@Web application must provide a registration page to define a username and a password to a user.
* The web service provides a search functionality to the inventory
* Inventory – any user can visit different pages of the web application
  + Men AND Women AND Kids products pages
* Navigation can be done by:
  + Product name
  + Product model
  + Product brands
  + Product sizes

## Functional Requirements for Non-registered User

* Non-registered user can open the website URL.
* Non-registered user can navigate between products pages.
* Non-registered user cannot add items to a Cart, therefore, cannot make a purchase.

## Functional Requirements for a Registered User

* Login page – To allow user to log into the web service at any time.
* The web service must allow user to log out.
* Registered user can make purchase
* Registered user can add item to the cart
* Registered user should be able to remove item from the cart
* A link to the checkout should be available all the time at all products pages
* The web application must have a Payment page
* The payment page must provide a summary of what user has in their cart
* The payment page provides a total of the purchase being made
* The payment page collects user personal details
* The payment page collects user card details
* The payment page collects user deliver address OR / AND billing address details
* User can access the website in many different device (Desktop, Laptop, Mobile and Tablets)
* Registered users have access to his/her historical orders
* Registered user can unregister from account

## Functional Requirements for an Administrator Manager

* Has Read/Write/Modify permission
* Add, edit, and delete products.
* Change prices of created products.
* Deal with site security

## Functional Requirements for an Administrator User

* Has no write permission to the customer table
* Has write permission to edit images
* Has read permission to orders and addresses of customers

## Non-Functional Requirements

* Ensure there is a security measure to only allow admin user access to change the content of the website pages.
* Ensure that only registered users can make purchase on the website
* The website should be compatible with all browsers and viewable on most technologies, for example desktops, laptops, and mobile devices.
* The website should be easy to use with easy navigation through the pages and consistent design and page layouts.
* The website needs to have a fast response time and processing time for example in accessing the sign-up forms to register.
* The website will be live for twenty-four hours a day, seven days a week and will be free of charge for users to use.

## Our Web Application will have the following pages available:

The pages are divided into statics and dynamics pages.

1. Home page - The homepage will be the initial focus and it will feature a navigation menu that will direct users to the part of the site they wish to visit. The purpose of this page is to introduce the visitor to the site and give an overview of the products on offer, and to display a variety of products with an objective to keep the user interested enough so that they will purchase from our application.
2. About Us page – Detailing a short history of our company and what our goals are.
3. Registration page – Enabling users to register so that they can purchase our available products.
4. Profile page - details on user, such as phone number and email address, which can be edited by user. Purchase history of user will also be provided on this page.
5. Product page - displaying inventory which can be browsed in a variety of ways and can also be searched. The user can click on some of the sample products displayed on homepage or can choose from major product categories on the main menu. User also can use search function to search site for specific item(s). The ability to choose between Men’s, Women’s and Children products will be available.
6. Contact Us page – a facility that will enable users to contact our support team through filling out a form and will have the ability to contact us with queries.
7. Check out page – displaying what the user has placed in their cart, how many items they are purchasing and the total amount due. They can also edit their shopping cart from here by adding or deleting items. When they are ready to finalise their order, they will be redirected to a payment page to complete the process.
8. Men’s product page –
9. Women’s product page –
10. Kids product page –
11. Login page –
12. Cart page -
13. Thank you, page, – after a purchase is made a page displaying a thank you message, and order number will be visible on the user’s screen.

## Use of Third-Party Gems

* Devise gem used to generate authenticated users.
* PayPal gem – To allow us to receive payment via PayPal system.

## External interfaces

* The Shop@Web application will have a link to socio media such as: Facebook, Instagram, Snapshot, Twitter, and LinkedIn.

## Extra functionalities

To make the website more user friendly and fast to access it some functionalities can be added.

* Allow user to login to the Shop@Web via Facebook or Google account.
* Provide users with order history.
* Add some useful javascript
* Make the application adjusted to any device at any size.
* Instruct user about navigated page, so user can always locate themselves on the website.

## Bibliography

Roth, R.M., Dennis, A. and Barbara Haley Wixom (2013). *System analysis and design.* Hoboken, N.J.: Wiley.

Lecturer McCabe, L (2020) ‘Introduction to Server-Side Development’ [PowerPoint presentation]. [HDSDEVSEPOL\_YR2](https://mymoodle.ncirl.ie/course/view.php?id=813)Server-Side Development*.* Available at: <https://mymoodle.ncirl.ie/course/view.php?id=813> (Accessed: 11 October 2020).

Lecturer McCabe, L (2020) ‘Ruby on Rails’ [PowerPoint presentation]. [HDSDEVSEPOL\_YR2](https://mymoodle.ncirl.ie/course/view.php?id=813)Server-Side Development*.* Available at: <https://mymoodle.ncirl.ie/course/view.php?id=813> (Accessed: 12 October 2020).

Lecturer McCabe, L (2020) ‘Routes, Relationships and Active Record’ [PowerPoint presentation]. [HDSDEVSEPOL\_YR2](https://mymoodle.ncirl.ie/course/view.php?id=813)Server-Side Development*.* Available at: <https://mymoodle.ncirl.ie/course/view.php?id=813> (Accessed: 12 October 2020).

‌Dr. Muhammad, I (2020) ‘SQL Programming DML (Part I)’ [PowerPoint presentation]. *HDCOMP\_877117964 Introduction to Databases.* Available at: <https://moodle.ncirl.ie/mod/resource/view.php?id=50820> (Accessed: 13 October 2020).

Dr. Muhammad, I (2020) ‘SQL Programming DML (Part II)’ [PowerPoint presentation]. *HDCOMP\_877117964 Introduction to Databases.* Available at: <https://moodle.ncirl.ie/mod/resource/view.php?id=50820> (Accessed: 13 October 2020).

MySQL:: MySQL Tutorial. 2020. Date Calculations. [ONLINE] Available at: https://dev.mysql.com/doc/mysql-tutorial-excerpt/5.7/en/date-calculations.html. [Accessed 13 October 2020].

Learn the Unified Modelling Language V2 – Up2UML within distinct Software Development Processes [ONLINE] Available at: <https://moodle.ncirl.ie/pluginfile.php/505443/mod_resource/content/1/alpha1191248764%20%281%29.pdf>. [Accessed 19 October 2020].

Schramel, M (2020) ‘POST Case study part II Analysis Phase’ [PowerPoint presentation]. [HDSDEVSEPOL\_YR2](https://mymoodle.ncirl.ie/course/view.php?id=813) *Object Oriented Software Engineering.* Available at: <https://moodle2019.ncirl.ie/course/view.php?id=420> (Accessed: 13 October 2020).