

**Coding Academy by Orange™**

**2nd Project - Me and a tree**

**2019-2020**

Yaser Muhammed Jaser.

Baraa Daraghmeh.

A project submitted in partial fulfillment of the requirements of the Coding Academy.

Abstract:

Earth has been suffering unstoppably global warming and pollution over the years, rendering humans with new challenges to face such as poverty and lack of healthy life resources.

Me and a tree, a website non-profitable charity-based project initiated to stand up against the harshness of today’s life. An act of heart toward our mother nature, a word to speak against every impurity that harms our race and Planet.

Content:

Contents

[Problem Statement: 4](#_Toc15168154)

[Objectives: 4](#_Toc15168155)

[Development Methodology: 5](#_Toc15168156)

[Requirement analysis: 6](#_Toc15168157)

[Analysis model: 9](#_Toc15168158)

[Use case diagram: 9](#_Toc15168159)

[Design phase: 10](#_Toc15168160)

[Chosen development technologies: 10](#_Toc15168161)

[Data Flow Diagram (DFD): 11](#_Toc15168162)

[State Diagram: 12](#_Toc15168163)

[Implementation Strategy: 13](#_Toc15168164)

[Test Methodology: 14](#_Toc15168165)

Chapter one

Planning

# Problem Statement:

The project’s benefit is to ease the process of gathering resources, donators and plants to cover up desertification, improve local productivity and increase charities’ ability to provide.

The system consists of:

1. Components and needs:   
   -Me and a tree website.

-Domain.

-Internet connectivity.

-Device to explore on.

1. First look:  
   -The landing page consists of a pure message delivered to the client through animated figures and words.
2. Exploring:  
   - The client can explore responsively using any internet connected device such as mobile or computers using keyboard and mouse or touch screens.

# Objectives:

* Re-greening the planet.
* Fighting poverty and harsh life conditions.
* Donations to charity.
* Improving local productivity and industry.
* Enjoying a healthier life

# Development Methodology:

The project is divided into a Proof-of-principle prototype, visual prototype, working prototype, User experience prototype and a functional prototype that allow us to divide the project into steps and sequences that can be reverted thus, improving time and recourse management.

**When to use the Prototype model:**

* Users are actively involved in the development
* Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.
* Errors can be detected much earlier.
* Quicker user feedback is available leading to better solutions.
* Missing functionality can be identified easily.

Chapter two.

Analysis

# Requirement analysis:

User requirements:

* The user can navigate through a Navbar.
* The user can simply become a sponsor by creating an account.
* The user can see details of each work and plant provided.
* The user can exchange pages easily.
* The user can choose between the application’s supported languages.

Functional requirements:

* Users can navigate the website responsively using any device.
* Users can follow simple steps to sign up.
* Allow the user to being restricted to the shown paths.
* Show details about trees and recent work.

**Non-Functional requirements**

**External Non-functional Requirements:**

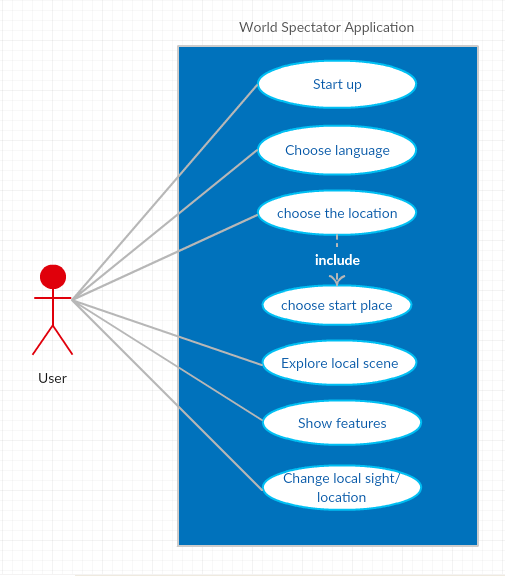
* **Usability:**  
  The system is simple enough for both new and experienced users as it only needs the basics of interacting with a website and the ability to set an internet connection for use.
* **Reliability:**  
  The system provides a smooth experience up to the whole site with details.
* **Appearance:**  
  The system’s appearance is appealing to the eye as it offers a simple interface and a modified experience though simple movement methods and menus.
* **Privacy:**The system does not collect any personal information and has no undesired access to the device’s applications and hardware such as cameras or mics.
* **Scalability and Adaptability:**The system is set to interact and view different devices in the same way and with a verity number of users.
* **Usefulness:**  
  The system meets the relevant needs and experience of a charity planning without the usual consummation of resources such as time, cost and travelling.
* **Delightfulness:**  
  The system’s experience goes beyond expectations of a usual visit to a traditional campus due to the information and the ease of viewing thus, creating an unforgettable experience.

**Internal non-functional requirements**

* **Configuration:**The system needs a device to the run the application, a connection to the internet.
* **Environment:**  
  The system basically runs in any environment such as houses and companies.
* **Portability:**  
  The system is considered portable as it only needs a device running the website.
* **Manageability:**  
  The system is easy to manage through basic menus and basic traditional website’s methods of use.
* **Internationalization:**  
  The system can be used by the whole globe as it is supports foreign planning, donating and testing.
* **Installation:**  
  There is no need to install the website as it only needs an internet connection and a browser

# Analysis model:

# Use case diagram:



Chapter three.

Design.

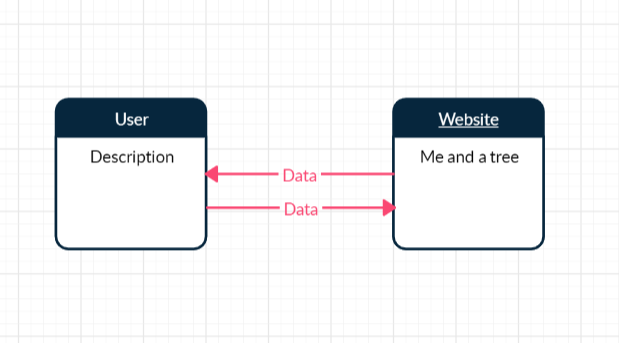
# Design phase:

CRC (Class Responsibility Collaboration) cards have been made depending on the main scenario to determine what functions and variables we will be using [See Appendix]. We designed several blue prints and designs of interfaces that were shown to a random sample of people to decide the best one. Several opinions were taken and several tests have been done to decide the best development language to be used in the creation of our Website.

# Chosen development technologies:

* HTML5.
* CSS3.
* Bootstrap 4.
* JavaScript.
* jQuery.
* Font Awesome.
* Google Fonts.
* Injections.

# Data Flow Diagram (DFD):

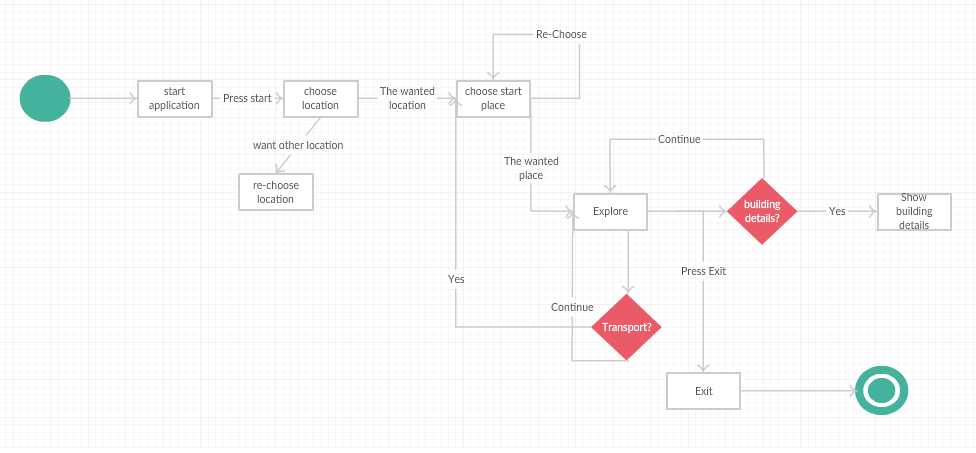


LVL (0) Context Diagram.

Chapter four.

Implementation.

# State Diagram:



# Implementation Strategy:

Dynamic programming:

One disadvantage of using Divide-and-Conquer is that the process of recursively solving separate sub-instances can result in the same computations being performed repeatedly since identical sub-instances may arise.

The idea behind dynamic programming is to avoid this pathology by obviating the requirement to calculate the same quantity twice.

The method usually accomplishes this by maintaining a table of sub-instance results.

Dynamic Programming is a Bottom-Up Technique in which the smallest sub-instances are explicitly solved first and the results of these used to construct solutions to progressively larger sub-instances.

In contrast, Divide-and-Conquer is a Top-Down Technique which logically progresses from the initial instance down to the smallest sub-instance via intermediate sub-instances.

# Test Methodology:

For our testing phase we used Beta testing, which is making some people who are intended to use the website to try it and check if there is any problem that could face the user before releasing it.

Their feedback can be written as points, which are:

1. The Navbar is not responsive.
2. The black background color must be changed.
3. The real pictures Carousel is too slow.
4. Too much words and paragraphs.

Solutions provided to their problems:

1. Use Bootstrap based Navbar.
2. Background color combination became a linear gradient between white and lemon green.
3. jQuery based Carousel was used with a better time interval.
4. Typography and icons were added to ease the delivery of ideas.