DEVELOPMENT AND IMPLEMENTATION OF A CMS-BASED WEBSITE FOR THE CSIR-FOOD RESEARCH INSTITUTE

BY

KWABENA ASIEDU BUGYEI
RAPHAEL KWAME KAVI
STEPHEN NKETIA

DECEMBER 31, 2014
CSIR-FOOD RESEARCH INSTITUTE
P.O.BOX M20, ACCRA

TABLE OF CONTENTS

Table of contents…….....i

List of figures……iii

List of Acronyms……...iv

Abstract……...v

1.0 Introduction.....1

1.1 Background of the CSIR-Food Research Institute.....1

1.1.2 Vision..…….....1

1.1.3 Mission…….....1

1.1.4 Core Mandate.....2

1.1.5 Core Values…….……...2

1.2 Introduction and Background to the Project……...…..3

1.2.1 Objectives of the Project……..…4

1.2.2 Project Duration……..……4

2.0 Content Management Systems.....5

2.1 Key features and components……..5

2.1.1 Content Management Application (CMA)……..…..6

2.1.2 Content Delivery Application (CDA)…….....7

2.1.3 Template engine…….....…7

3.0 Joomla.....8

3.1 Framework and extensions.....8

3.1.1 Components.....10

3.1.2 Modules.....10

3.1.3 Plugins.....11

3.1.4 Templates.....11

3.2 Security.....11

3.3 Technical Requirements.....12

4.0 Information Architecture…….....……..13

4.1 Methods and Techniques……..……14

4.1.1 Organization Scheme.....14

4.1.2 Organization Structures.....15

4.1.3 Content inventory.....16

4.1.4 Wireframing.....16

5.0 Project Implementation.....17

5.1 Requirements analysis and specification.....17

5.2 Preparing the development environment.....17

5.2.1 Why Joomla was chosen?.....17

5.2.2 Installing XAMP for Windows.....17

5.2.3 Installing Joomla on localhost.....18

5.2.4 Other Software Installations.....19

5.3 Designing the information architecture.....20

5.4 Designing the layout mock-ups.....21

5.5 Building the website with Joomla.....24

5.5.1 Choosing and customizing the template.....25

5.5.2 Implementing navigation system and additional modules.....26

5.5.3 Customizing the template with LESS.....27

5.6 Testing and deployment.....27

5.6.1 Deploying website to web server.....28

6.0 Conclusion and Recommendation…….……29

References.....30

Appendices.....33

Screenshots of the finished website…….34

Appendix 1: Screenshot of the Homepage mock-ups

Appendix 2: Screenshot of the Homepage of the finished website

Appendix 3: Screenshot of the About Us page

Appendix 4: Screenshot of the Products and Services page

Appendix 5: Screenshot of the Products page

Appendix 6: Screenshot of the Technical Services page

Appendix 7: Screenshot of the R & D page

Appendix 8: Screenshot of the R & D page

Appendix 9: Screenshot of the Management Team page

Appendix 10: Screenshot of the Contact page

Appendix 11: Screenshot of the News & Events page

Appendix 12: Screenshot of the Download page

List of figures

Fig. 1: Illustration of template engine's functionality

Fig. 2: Joomla web page visualizing the different extension types

Fig. 3: The three circles of information architecture according to Rosenfeld and Morville

Fig. 4: An example of 16 column grid system

Fig. 5: A screenshot of the Chimera template, front-end template

List of Acronyms

Apache – Open source HTTP server application

CMS – Content management system

cPanel – Control panel application for web hosting

CSS – Style sheet language for markup documents (Cascading Style Sheets)

CSIR – Council for Scientific and Industrial Research

FRI-Food Research Institute

HTML – Standardized web page markup language (HyperText Markup Language)

LESS – Dynamic style sheet language

MVC – Software design pattern (Model–view–controller)

MySQL – Open source relational database management system

PHP – Server-side scripting language (PHP: Hypertext Preprocessor)

phpMyAdmin – Open source web application for MySQL database management

SEO-Search Engine Optimization

SQL – Standardized query language for databases (Structured Query Language)

XAMPP – Cross-platform Apache distribution containing Apache, MySQL, PHP and Perl

WAMP – Web development stack (Windows, Apache, MySQL and PHP)

WYSIWYG – Text editor that mimics final appereance (What You See is What You Get)

Abstract

In today's modern world, organizations that want to reach global audiences uses the power of internet technology to achieve their goal. Websites are the most convenient way to present and disseminate information to the maximum number of people in the world. This report describes a live project carried out towards the design, development and implementation of an official website of the Council for Scientific and Industrial Research-Food Research Institute using Joomla!, an open source web Content Management System (CMS) software to design a dynamic website.

The purpose of this project was to build a dynamic and interactive CMS-based website for the CSIR-Food Research Institute.

The final result was an interactive, dynamic and user friendly website which provide information to its customers, clients, partners and the world at large. The website developed is secure, reliable, flexible and cost-effective.

1.0 Introduction

1.1 Background of the CSIR-Food Research Institute

The Food Research Institute (FRI) is one of the thirteen (13) Research Institutions of the Council for Scientific and Industrial Research, (CSIR) which operates as a Science and Technology Research Development Organisation. The Food Research Institute was established in October 1963, incorporated by L I No. 438 of 19th March 1965 and became an institute of CSIR in October, 1968 by NLC Decree 293.

CSIR-FRI is mandated to conduct market-oriented applied research, provide technical services and products to the food industry and assist in poverty alleviation through the creation of opportunities for income generation, thus contributing to food security and foreign exchange earnings. CSIR-Food Research Institute is the leading S&T Institute in the transformation of the food processing industry in Ghana.

The core research interests and programs of the FRI include:

I. Root and Tuber Products Programme
II. Meet, Fish, Poultry and Dairy Products Programme
III. Cereal and Grain Legume Products Programme
IV. Fruit and Vegetable Products Programme

1.1.2 Vision

The Institute`s vision is to play a key role in the transformation of the food processing industry and to be internationally competitive with particular reference to product safety, quality and preservation.

1.1.3 Mission

The Institute`s mission focusses on providing scientific and technological support to the growth of the

food and agricultural sectors of national economy in line with corporate priorisation and national objectives. Primarily, the Food Research Institute`s mission is to conduct market-oriented applied research and provide technical services and products profitably to the private sector and other stakeholders.

1.1.4 Core Mandate

The CSIR-FRI conduct applied research into problems of:

- Food processing and preservation
- Food safety and storage
- Food marketing, distribution and utilisation
- National food and nutritional security in support of the food industry
- Advice Government on its food policy
- To assist in poverty alleviation through creation of opportunities for generating and increasing income within the micro, small, medium and large-scale food industries
- In support of the food and agricultural sectors of the national economy

1.1.5 Core Values

CSIR-Food Research Institute believes and ascribes to the following values:

- Professionalism
- Team work
- Innovativeness
- Competitiveness
- Quality Delivery

1.2 Introduction and Background to the Project

In early 2000, websites were built using various languages such as XHTML, CSS, PHP, ASP, and JavaScript, which lead to several problems and limitations such as unmanageable code, increased risk of data loss, and making it harder to update and implement basic required Search Engine Optimization (SEO) techniques. For small frequent changes, these tasks required hiring web developers: an expensive labor. The complexities created the need for a system to mask all the technical details and to provide a user friendly platform that organizations could use to implement basic changes. Out of this need, Content Management Systems (CMSs) were invented.

CMS platforms are frameworks which build custom content applications based on an organization's needs. The content management system provides the ability to maintain and navigate the structure of the site to the users, and allows the developer to develop the website more easily.

The use of Content management systems by businesses, individuals, corporate and non-governmental organizations are recognized in the business and technology industries. Developers and companies are realizing the advantages in the concept of free development environment for Web contents.

Content Management systems have made content publishing on the internet fast and easy. Companies want to reach out to their customers, sell their products and expertise, maximize profits and increase efficiency. All these are great advantages provided by content management systems applications.

CSIR-Food Research Institute, a leading food research institution in Ghana intends to reach out to its online audiences and customers across the world to promote their products and services and R & D activities. The organization aims to increase its revenue by registering its presence on the internet.

Because of the many advantages offered by Content Management Systems and Joomla in particular, this project was developed and implemented using these tools. This report describes the processes involved

in developing and implementing a CMS-based website as an official website for the CSIR-FRI.

1.2.1 Objectives of the Project

The purpose of the project was to design, develop and implement a website for the CSIR-Food Research Institute. The main objective was to produce a dynamic and content rich website that would be easy to use and been built on top of a secure and robust content management system.

The specific objectives are as follows:

1. Analyze and specify the website's functional and non-functional requirements
2. Install and configure the development environment and the necessary software
3. Design the website's information architecture and layout mock-ups
4. Build the website with Joomla CMS
5. Test the usability and functionality of components and layout
6. Deploy the website to the web server (hosting provider’s web server)

1.2.2 Project Duration

The development task described in this report was carried out between June 2014 and December 2014 and the website was launched on 28th December 2014. This report was written along with the development tasks and was finished in December 2014.

2.0 Content Management Systems (CMS)

This chapter describes web based content management systems and their core features and components. According to US registered patent (US 6356903 B1), content management system (CMS) is an information delivery system for web based implementations that organize content of the information separately from the appearance of the presented information and by so allowing content creation and management to be done in a native format with familiar software tools (Baxter & Vogt 2002).

Simply put, CMSs are installable web applications that include a database and extensive set of tools dedicated for content creation, management and publication on the Internet. Their main purpose is to ease content management and publishing tasks on the web by providing tools that do not require extensive knowledge of the web markup languages or its protocols. (Shreves 2010, 3–4; TechTarget 2011a.)

Originally, CMSs were designed as tools for organizations to dispose static HTML websites and to simplify web publishing that required knowledge of HTML and other web standards. The traditional method was found impractical and expensive as publishing content required constant co-operation between the content contributors and web developers. A solution was achieved by developing a system that integrated all the elements of web publishing under single implementation. This innovation breached the barriers that had existed in traditional web publishing allowing it to become less technical and more streamlined. (Shereves 2010, 3–4.)

2.1 Key features and components

In software engineering, one will eventually come across with the terms front end and back end. Front end and back end are terms' which main purpose is to distinguish the tasks related to presentation layers and data access layers. The front end is more concerned about the presentation logics while the back end is more concerned about the data access and business logics. Technically speaking, front end development involves mostly client-side coding, while the back end development involves mostly server-

side coding.

When talked about CMSs, the terms front end and back end may refer to separate application areas as the CMS distributions basically contain two different applications. In this case, the front end refers to the publicly visible website that serves as the medium for regular users while the back end refers to the administration application that is used for managing content and application's preferences by the system administrators and editors. Nevertheless, both the public and administrator application contain front-end and back-end components as they both contain presentation and data access layers.

For the sake of clarity, this and the next chapters use the following naming conventions to distinguish these terms:

- Front end component refers to the presentation layer (Client-side code)
- Back end component refers to the data access layer (Server-side code)
- Administrator application refers to the back end application of CMSs (Administrative view)
- Public application refers to the front end application of CMSs (Public view)

2.1.1 Content Management Application (CMA)

Content Management Application (CMA) is the front end component of the administrator application. CMA is basically an administrative control panel with a graphical user-interface (GUI) that is used for managing content and assets, users, access rights, templates, extensions and other system preferences of websites. Back-end application is typically equipped with efficient and user-friendly tools such as WYSIWYG text editors that allow editing and publishing task to be done without having extensive technical knowledge of web standards and its protocols. Access to the CMA and its features are usually controlled with login and access control to ensure the integrity of content and security of the entire system. (TechTarget 2011b; Goodrich 2013.)

2.1.2 Content Delivery Application (CDA)

Content Delivery Application (CDA) is the back-end component of the administrator application. CDA is basically the component that handles all background operations while transacting data between the content repository and view by compiling user inputs made via CMA. (TechTarget 2011a; Goodrich 2013).

2.1.3 Template engine

Template engine is the CMS component dedicated for presenting information to users by attaching the content from data repository to a pre-defined layout template in order to generate the output for users. Basically, templates are layout definers and user-interface element placeholders that do not hold actual content, but rather provide framework for presenting it. With the help of CMA, editors with minimal technical knowledge are able to edit and manage templates to some extent but more advanced editing or building one from scratch requires advanced knowledge from HTML, CSS and various other web standards. (Shreves 2010. 497–498.)

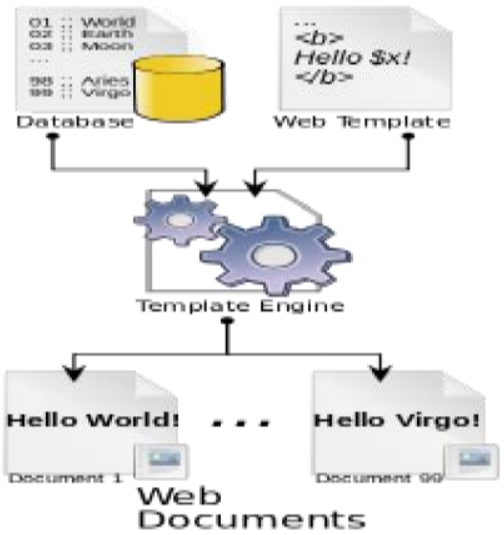


Figure 1: Illustration of template engine's functionality (Wikipedia 2006)

3.0 Joomla

This chapter introduces Joomla CMS and its core features and components. The chapter does not describe Joomla's technical implementation in high-detail or compare Joomla to other CMSs.

Joomla is a user-friendly open-source and community driven CMS that is built on top of robust model-view-controller (MVC) framework. It is developed and maintained by The Joomla Project Team, a global developer community with contributors all over the world. (Joomla 2014a; Joomla 2014b.)
It is considered one of the best and most sophisticated CMS's available and according to a survey of W3Techs (2014), it is currently ranked as the second most popular CMS platform in the world.

The benefits of Joomla to an organization are many, since Joomla is an Open Source Software meaning its free of charge, the organization can use Joomla to build a dynamic website without the cost of software license. The components associated with Joomla, PHP, MYSQL and Apache are Available at no cost to an organization and these components add much value to Joomla application. For example MYSQL component enables Joomla application to have a database were all web content could be stored and easy accessed too. Joomla has many appearing features that attract web developers to use this application, user-friendly interface, easy installation, professional standards, internationalization and customization is easy to do in Joomla. Many web hosting companies can host Joomla website without any reservation.

3.1 Framework and Extensions

As mentioned, Joomla has been built on top of MVC framework which can be used to build stand-alone applications. Besides the framework, Joomla CMS is composed of a set of extensions which each have differing functions. Some of the extensions are part of the CMS core and are essential for proper functionality and stability of the system and thus come along with the default installation. Moreover, the core can be extended with various other extensions which can be purchased or downloaded for free via Joomla Extension Directory which is a centralized distribution channel for all types of Joomla extensions.

(Joomla! Docs 2013a.)

Extensions are software packages that extend the default Joomla installation in some way. It is relevant to understand that extension term is generic and that it means all types of extensions such as components, modules, plugins and templates which each are meant for different purposes. (Joomla! Docs 2014a.)

Figure 2: Joomla web page visualizing the different extension types (Joomla 2013d).

3.1.1 Components

Components are the most sophisticated extensions that provide the main functionality to Joomla system. They can be referred as mini-applications that generate output for different parts such as the main content section of each web page. (Joomla! Docs 2014b; Shreves 2010, 547.)

Components are triggered by HTTP requests and they execute series of operations within the framework which ultimately leads to generating the output. For example, an article view component, com_content, performs all the actions required from fetching the data from database to rendering article's HTML document. The framework provides abstract classes for model (JModel), view (JView) and controller (JController) which the components extends to in order to have standardized architecture. (Joomla! Docs 2014b; Shreves 2010, 547–548.)

Joomla components are composed from site part (front-end application) and administrator part (back-end application). The site part is used for rendering page content for the front-end application while the administrator part provides a user-interface for the back-end application for configuring components' preferences. (Joomla! Docs 2014b; Shreves 2010, 547)

3.1.2 Modules

Modules are customizable page rendering blocks that are used for displaying specific information in a specific area. Modules can be associated to components when they inherit their functionality or they can be independent blocks that render content the way specified within the module. (Joomla! Docs 2014c.)

Modules are assigned to pages via navigation menu items so they can be shown or hidden depending on the page in question. The positioning of the module is entirely dependent on the used template's module placeholders but it is possible to position a module into any placeholder that template in question provides. Modules are also divided into front end and back end parts where the back-end part provides a

user-interface via Module Manager where they can be freely configured. (Joomla! Docs 2014c.)

3.1.3 Plugins

Plugins are extensions that serve as event handlers and helper applications. They are triggered by certain events and they respond to it with procedures that can be visible or esoteric. Plugins are mainly used for extending the functionality of components and they are built with observer design pattern where the dispatcher notifies all the associated plugins allowing them to be executed in sequence. (Joomla! Docs 2014d; Joomla! Docs 2013b; Shreves 2010, 543, 569.)

3.1.4 Template

Templates are extensions that control the overall layout and visual presentation of content. They define how various page elements should be structured and how the content should be rendered within them although some components and modules may contain their own styling rules. The purpose is to separate the visual presentation from the actual content allowing more efficient website maintenance with possibility to apply consistent layout throughout the website. (Joomla! Docs 2013c; Shreves 2010, 497).

Joomla provides separate templates for both front-end and back-end applications. Templates can be installed or built from scratch and they can also be configured via back-end application using the Template Manager. (Joomla! Docs 2014e; Joomla! Docs 2013c.)

3.2 Security

Joomla is considered to be a very secure CMS. Nevertheless, as being an open-source platform it is also a common target for various cyber-attacks and breaching attempts. (Joomla Security Info 2014; SiteGround 2014.)

“Keeping your site patched and up to date is one of the keys to maintaining your site’s integrity and protecting it against hackers” (Shreves 2010, 685).

Joomla documentation (Joomla! Docs 2014f; Joomla! Docs 2012) suggests few general guidelines on how to improve the security of Joomla installation. These guidelines include the following procedures:

- Back up early and often
- Install Joomla updates when they are released
- Use a trustworthy and secure hosting provider
- Use strong usernames and passwords for login
- Do not trust third party extensions
- Use an offline environment to test extensions before applying them to live site
- Do not use the default security settings
- Use the community to get help in security matters

In addition to above mentioned, it is considered as a good practice to use the SEF (Search Engine Friendly) component to rewrite Available ats, to use proper file permissions and ownerships on server side and to use some safe third party security extensions that protect the site from different types of attacks. (SiteGround 2014; Joomla! Docs 2014f.)

3.3 Technical requirements

By the time of writing this report, the latest stable release of Joomla is 3.3 and it requires PHP version 5.3 or higher, MySQL, SQL Server or Postgre SQL database and Apache 2.0, Nginx 1.0 or Microsoft IIS 7 as the webserver (Joomla.org 2014c).

4.0 Information Architecture

This chapter describes what is meant with an information architecture and what kind of methods can be applied at its design process.

According to Morville and Rosenfeld (2006, 4), information architecture is:

- A structural design of shared information environments.
- The combination of organization, labeling, search, and navigation systems within web sites and intranets.
- The art and science of shaping information products and experiences to support usability and findability.
- An emerging discipline and community of practice focused on bringing principles of design and architecture to the digital landscape.

Basically, information architecture (IA) is information system's structural solution for structuring, organizing and categorizing content in effective and sustainable way. The aim of it is to establish logical, clear and consistent structures and paths towards content so that the information can be found easily by any user. (Usability.gov 2014a.)

Information architecture is a broad concept including multiple aspects, tasks and purposes. In order to fully understand it, it is crucial to realize the overall structure in a big picture and how the pieces of information are connected to form a network of information and how they are related to each other. Designing information architecture is a constant balancing between the needs of independent users, diversity of content and context. In addition to users' needs, the content should also be structured for search engines and web crawlers making the task even more challenging. (Usability.gov 2014a & Morville 2012.)

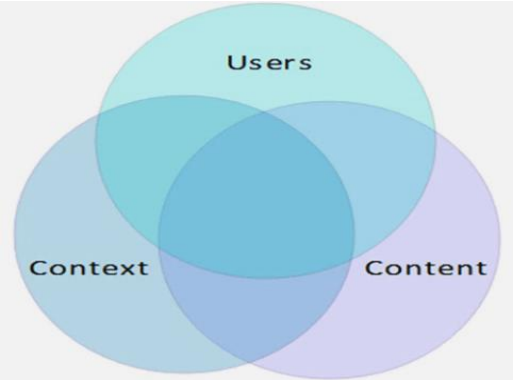


Figure 3. The three circles of information architecture according to Rosenfeld and Morville (Usability.gov 2014a).

4.1 Methods and techniques

Information architecture contains elements from various specialty areas. It involves many activities that are more or less related to topics such as usability, user experience (UX), layout and user-interface design. To be successful it requires a good understanding of industry and organizational standards as well as recognizing how users navigate and use information systems in general. (Usability.gov 2014a.)

Morville and Rosenfeld state that information architecture is composed from organization systems, navigation systems, search systems and labeling systems (Morville & Rosenfeld 2006. 43).

Organization systems describe how the content is organized and categorized. Usability.gov suggests that organization systems can be further divided into schemes and structures where schemes are used for categorizing content and structures for forming relationships and hierarchies between them. (Usability.gov 2014b.)

4.1.1 Organization Scheme

Organization schemes can be exact or subjective depending on the approach. Exact schemes mean organizing content objectively in a way that it is commonly known and can be easily sorted. Such means include alphabetical and chronological ordering. Subjective schemes mean the opposite as it focus on

organizing content into subjective categories defined by organizations. These categories can be for example a certain topic, a tag word or audience. The advantage of subjective mean is that the user is more likely to find related content of the subject while the exact schemes on the other hand are more predictable and so the user is likely to understand the logic of organization scheme more quickly. (Usability.gov 2014c.)

4.1.2 Organization Structures

Organization structures is a about defining relationships between contents and schemes. Well implemented structures are predictable and help users to clearly understand the overall structuring and categorization and so make navigation a lot easier. The organizational structures can be hierarchical, sequential or in a form of matrix. (Usability.gov 2014d.)

The purpose of hierarchical structures is to establish a clear hierarchy among content. The essence of hierarchical structure is in parent-child relationships where parents cover broader sections while the children provide more detailed and targeted content. The hierarchical structure is often visualized with a tree chart involving a root and numerous branches. (Usability.gov 2014d.)

Sequential structures are line like structures that include continuity of procedures until reaching the end. Sequential structures are typically used when providing step-by-step guidance such as filling registration form or conducting online shopping. Sequential structure assume that the next piece of content is somehow dependent on the previous content and so there is clear ordering among them. (Usability.gov 2014d.)

When designing organization structures, it is important to design them flexible. Information architecture is usually established with a long term perspective so it is likely that the amount of content will grow or the organizational needs change. The structures should neither be too broad nor confined as the content might get lost if there are too few or too many levels between the user and content. (Usability.gov 2014d.)

4.1.3 Content inventory

Content inventory is an inventory of all content that is included within the information system. Content inventories are tools for designers and information architects to understand the content in a larger scale and by so provide insights whether all relevant content is gathered and properly organized. In addition, content inventories give insights whether the essential metadata has been assigned to each content item. (Usability.gov 2014e.)

4.1.4 Wireframing

Wireframing is a technique that can be used for layout design as well as for information architecture. Wireframe is a two-dimensional mock-up of the layout and by so it visualizes the final implementation. The benefits of wireframing are that it demonstrates early on how the users will see the content and how the used information architecture fits in. It basically gives insights whether it is functional or not. Wireframe also demonstrates the space allocation and how the content can be positioned within the limits of displays providing valuable information for making the final design decisions. (Usability.gov 2014f.)

5.0 Project Implementation

This chapter describes the implementation process of the CSIR-Food Research Institute website by describing the work phases and methods used.

5.1 Requirements analysis and specification

The project was started with requirements analysis and specification. The purpose of this work phase was to define the functional and non-functional requirements and to decide on the most essential guidelines that should be used in the subsequent design and development work phases.

Requirements analysis was conducted by having discussions with the Director and the Scientific Secretary of the Institute about the content to be published on the website as well as the design and structure.

The website had a specific requirements for the technical or aesthetic implementation and so those requirements were clearly defined.

5.2 Preparing the development environment

5.2.1 Why Joomla was chosen

While specifying the requirements, it was agreed that the website should be built with a CMS. This was due to the fact that the website would ultimately require various content management and publishing tools which usage should not require extensive technical knowhow. Joomla CMS in particular was chosen among other potential candidates simply because the Institutes previous website had been built with it. This meant that the Information Management Team was already familiar with Joomla's features and functionality and so did not require further consultation.

5.2.2 Installing XAMPP for Windows

Since Joomla requires a web server that supports PHP and MySQL, one was required to be configured in order to start the development tasks. During the development phase, Joomla website and database was

hosted on local computer that was integrated with a WAMP (Windows, Apache, MySQL and PHP) development stack.

This environment was prepared by installing XAMPP, a cross-platform Apache distribution that contains all the components (Apache, MySQL, PHP and PERL) that are required to run a Joomla website. XAMPP was chosen, because it was suggested by the Joomla documentation and it was considered the fastest way to set up the environment for a Windows operating system. (Joomla! Docs 2013e)

The installation of XAMPP was rather easy and straight-forward process as it was done with a step-by-step installer wizard. After the installation, Apache's configuration took some effort, since the default port number required changing. By default XAMPP installation for Windows assigns Apache's port number to 80, which in this case was already reserved by some other application. The port number was changed to 81 via Apache's httpd.conf file that was residing beneath the "xampp/apache/conf/" directory path. Finally, the functionality of XAMPP installation was tested by accessing localhost via web browser.

In this implementation, XAMPP version 1.8.1 was used containing Apache 2.4.3, MySQL 5.5.27 and PHP 5.4.7. All installations were done for Windows 8 operating system.

5.2.3 Installing Joomla on localhost

After the localhost had been configured, the installation of Joomla was started. The installation was done in two phases. In the first phase, Joomla's installation package was downloaded via Joomla's official website and the package's contents were extracted into a new directory on localhost. In XAMPP setup, a proper directory path for websites is "xampp/htdocs/", where each website should be assigned to separate directory. All websites residing under "htdocs" can then be accessed via browser by entering the Available at address of the localhost or alternatively using the computer's IP address. The localhost's Available at is formatted as "http://localhost:81/website/", where "81" refers to Apache's web server's port number and "website" to the namespace of website's directory. A good convention is to use short

and easy to remember namespaces, because it is just temporary and it is typed often during the development.

In the next phase, the initialization of Joomla CMS was done with Joomla's browser based installation application, which was launched, when the website was accessed for the very first time. The installation was a step-by-step process, where the website's basic information and settings such as website's name, description and database preferences where chosen and configured. After the installation process was completed, Joomla deleted the contents of the installation application automatically due to security reasons and after that, Joomla CMS was set up and ready to be used.

In this implementation, the website was built for Joomla 3.3.6 CMS.

5.2.4 Other Software Installations

In addition to XAMPP and Joomla, the following software was also installed and/or used in order to be able to complete this project:

- Adobe Photoshop CC – Graphics design software that was used for designing the layout mock-ups and was used to optimize photos and web graphics to make it web ready.
- Microsoft Word 2013 – Word processor software that was used for all documentation tasks and preparation of the contents to be published.
- Microsoft Visio 2013 – Flowchart drawing software that was used for designing the information architecture and workflow documents
- Microsoft Project 2013 – Project management software that was used for project management activities
- MySQL Workbench 6.0 and phpMyAdmin 4.3.6 – MySQL database management system that was used for managing the Joomla's MySQL database
- NetBeans IDE 7.3 and Aptana Studio 3– Integrated development environment that was used for

debugging Joomla and various PHP, HTML and CSS coding activities

- Notepad++ 6.3 – Simple text editor that was used for small coding and editing tasks
- FileZilla 3.7 – FTP client application that was used for deploying the website to the hosting account web server

5.3 Designing the information architecture

After completing the installation of Joomla and other software on the localhost, designing of website's information architecture was started. The goal of this work phase was to produce consistent and logical categorization conventions for contents and to decide how the website's navigation structure should be organized. This work phase included the following tasks:

- Defining categorization schemas and naming conventions
- Organizing and categorizing articles and other content
- Organizing content in hierarchical structure and producing the sitemap
- Choosing the media content for the websites (photos, videos etc.)
- Choosing the navigation system for the website

Before any categorization could be made, it was crucial to get familiar on how Joomla organizes and stores its content. Especially important was to research which database tables are essential for categorization and which components use these tables for displaying content. This research was done by reading Joomla's documentation and by analyzing Joomla's database schema, table relations and their contents in practice.

In Joomla, content is organized into categories. Each category can then contain sub-categories, which establishes clear hierarchical structure between different categories. Organizing content into categories contains one problem though, since by default, one article can only belong to one category. For some implementations, this can make categorization with categories a bit inflexible. For this reason, in this

implementation content was categorized by using tag words, which could be defined as many as liked. (Joomla! Docs 2013g.)

Content's categorization was started by organizing the content inventory. In this implementation, the content was organized under three-level structure by using subjective categorization schemes. In the beginning, all content was organized under a main category, because in Joomla, all articles must belong to some category. After this, each article was categorized under three main groups, depending on which topic they dealt with. Because these groups were organized by using tag words, each article could be assigned to more than one group. These groups also formed the second layer in main navigation system. Finally, each main group was assigned with a number of relevant tag words as sub-groups, forming the third level in navigation system.

The implementation of this work phase required extensive awareness of the content since the categorization was based on subjective schemas. In addition, the convention of good collective and accurate categorization terms was found challenging.

5.4 Designing the layout mock-ups

Designing of the layout was done in order to build an early mock-up of the website that would illustrate the final presentation by excluding all underlying functionality. This was done in order to simulate website's visual presentation in an early stage and also help the process of building the website's template. In addition, building an early mock-up confirmed that the requirements regarding the layout would be considered in final design.

Instead of using the simplistic wireframing technique, mock-up method was chosen, because it could demonstrate the final presentation more accurately and aid the process of choosing between different design practices.

The website's layout and layout design requirements specified that:

- The website should have a home page for featured and latest articles, photo sliders and news flash.
- The website should have pages for an about us page, products and services, projects, R & D, Downloads, contact us, etc.
- The layout should have a static header block for displaying logo, navigation, social media links and search bar
- The layout should have a static footer block for displaying contact information, Google map location, directions to the Institutes premises, other useful links, search bar and copyright information.
- The layout should have a dynamic body blocks displaying dynamic content
- The layout should be stylish that emphasizes content
- The layout should contain familiar and commonly known user-interface elements
- The layout should contain aesthetic but easy-to-read font stack
- The layout should be responsive and it should function well on every display device
- The layout design should be made with modern design practices

The layout mock-up was built with Adobe Photoshop Creative Cloud, a graphics design software, which was chosen due to previous experience.

Before continuing into drawing the layout elements, a proper design document was setup. First of all, it is a good practice to use pixels as the measuring unit instead of millimeters or inches, when web design is in question. This is mainly because digital media uses pixels to define and measure contents at their screen displays. (Webopedia; Web Designer Depot 2010.)

The dimensions of the design document are not absolute in Adobe Photoshop CC since the document

borders do not restrict the ability to position layout elements beyond the borderlines. In addition, document borderlines can be re-sized at any point which makes them more as guidelines for designers that aid in visualizing what the users can see at a time. That is why it is a good practice to use screen display's resolution for the document's dimensions. In this implementation, the document was set to width of 1920 pixels and height of 1080 pixels. (Just Creative 2012; Microsoft Developer Network. 2011.)

Another good practice is to enable the grid guidelines. This helps alignment and positioning of layout elements as the grid allows elements to be snapped to their borders. Moreover, by defining 1 pixel as the document's nudge increment, which defines how much an element is moved by single stroke of an arrow key, can help in aligning elements even more precisely. This implementation was done by setting the grid size to 24 pixels, which was the initially planned as the default margin between layout elements.

Next phase in preparing the design document was setting up the grid system. Grid system is commonly applied layout design technique that helps alignment of layout elements and content sections according to relatively positioned baseline columns. Grid systems can be static when each column is given a precise width in pixels, or they can be fluid, when the widths are assigned as relative percentages based on the full width of the template. On website's, the responsiveness of grid systems is usually handled with media queries and CSS classes, that handle floating, clearing and widths of columns elements. (Bootstrap; Sonspring 2008.)

One of the most common grid systems is a 12-column grid, where the template is divided into 12 equally wide columns. The width of columns is dependent on the overall width of the page and gutter margin that is left between each column, excluding the first and last column. This implementation was done by using a 12 column grid systems where each column was assigned to a width of 69 pixels with 24 pixel gutter margins. (Bootstrap.)

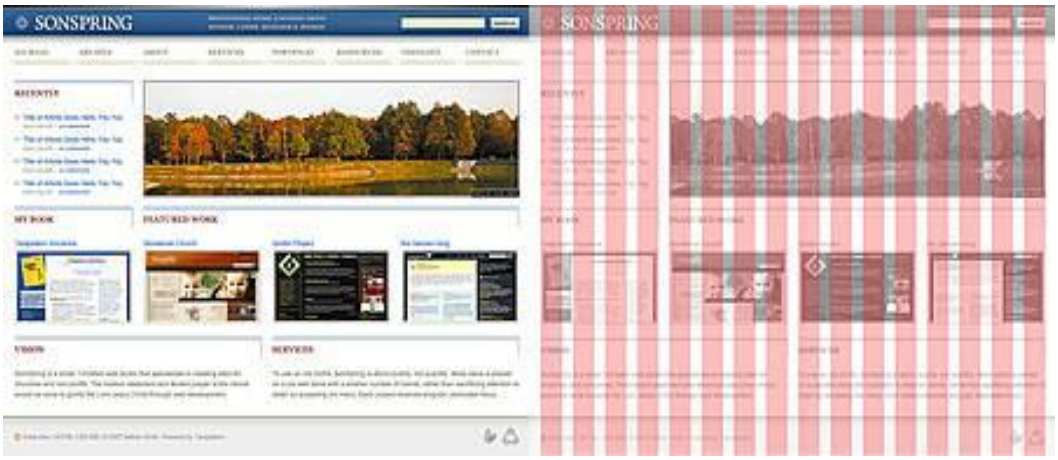


Figure 4: An example of 16 column grid system (Sonspring 2008).

After the design document and the grid system had been prepared, drawing of layout elements was started. This phase was done by first including the main content sections and components that had been specified in the requirements. Next, all the additional page modules were included along with dummy texts that served as temporary placeholders to visualize the use of space. Finally, all layout elements were organized and groomed to represent the finalized mock-up of a webpage.

The final outcome was achieved by improving the layout mock-up through numerous revisions and iterations. During these iterations, different page structures, font-stacks and element positions were tested, until the layout was considered to fulfill its requirements and be generally well balanced, where content and other layout elements were aligned in a good harmony.

5.5 Building the website with Joomla

After successfully designing the layout mock-ups, building of the website was started. The goal of this work phase was to construct the final website by following the specified requirements, information architecture and mock-up designs.

This phase included the following main tasks:

- Setting up a template that was similar to mock-up designs
- Setting up the navigation structure by creating menu items
- Assigning components to each menu item in order to provide page functionality
- Implementing additional modules and page elements
- Customizing the template according to layout mock-up
- Optimizing the layout for mobile devices

5.5.1 Choosing and customizing the templates

Before the implementation of any features, a proper template for the website needed to be set up. For this implementation, Chimera, a template developed by RocketTheme, was chosen, because it contained similar layout structuring and lots of similar features than the initially designed mock-up. By using a similar base template, less customization was required. Also, the fact that the template came along with a powerful Gantry framework that contained good customization features via administrator application, supported the decision making.

The template was installed via Joomla's back-end application by importing the template's zip file and configured via Template manager.

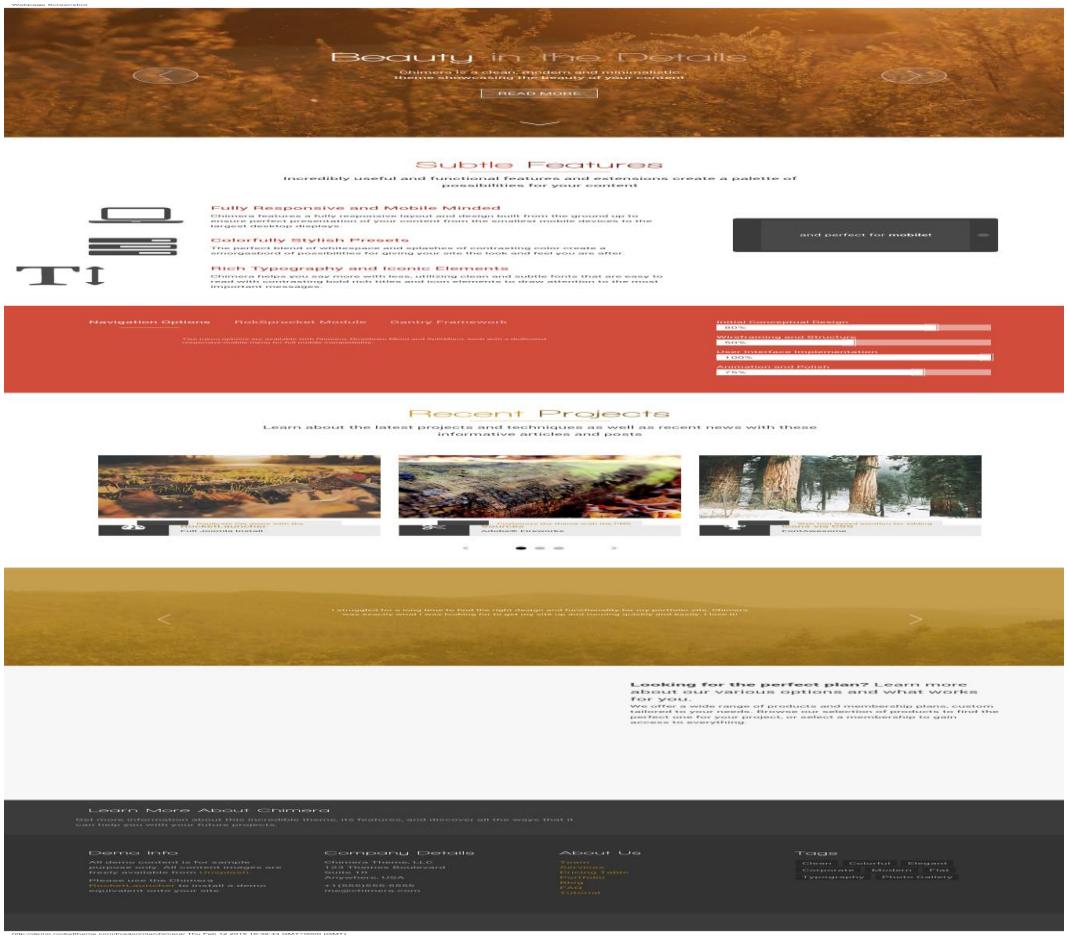


Figure 5: A screenshot of the Chimera template, front-end template (RocketTheme 2014)

5.5.2 Implementing navigation system and additional modules

The navigation system was created via administrator application's menu manager by creating new menu items in a hierarchical structure and assigning each item to a specific component. The components were chosen according to target page as what type of content were required to be rendered. Most of the menu items were assigned to a tag component, to fetch and display all articles that contain a specific tag word. (Joomla! Docs 2013g.)

Finally, the additional page modules were created via administrator application's module manager. Also, the modules were associated to specific component, although in some cases, the Custom HTML component was chosen, either included with some components functionality such as displaying a list of latest article releases or defined as customized HTML modules, when the content, functionality and

element structure where defined within the module.

5.5.3 Customizing the template with LESS

The customization of the template was done by editing template's LESS files. The editing was done by extending the template's default CSS class selectors and writing additional selectors for customized modules. Joomla 3 and Gantry framework support LESS, the dynamic CSS pre-processor language written in JavaScript, that extend the functionality of traditional CSS by adding support for variables, functions, nested rules and other features. These additions increase the efficiency and re-usability of CSS definitions and allow creation of more logical nested rules that mimic websites' DOM structures. (Joomla 2013; LESS)

In addition to editing the LESS files, the default views of Joomla components written in PHP and HTML were edited in order to have full control over contents' final presentation.

5.6 Testing and Deployment

Testing was done in order to confirm that the website was functioning well and fulfilled the specified requirements. Because the implementation did not include development of components or major coding tasks, testing was targeted mostly for website's usability and appearance.

During the testing, the following conditions were under a review:

- Pages and components function properly
- Back-end application's features function properly
- Layout functions properly on all display sizes
- Chosen font-stacks and font-sizes are good for readability
- All type of media content scales well on all display sizes
- All hyperlinks work well and points to the targeted page

- Recommended information security practices has been applied

Testing was done by running scenarios, where the functioning of the above mentioned test conditions were reviewed. The testing was conducted independently and they were run along with other implementation tasks.

After the basic test scenarios had been completed and the website functioning been confirmed, website was demonstrated to the Food Research Institute Internal Management Committee. During this demonstration session, the IMC was able to test the functionality of the website and suggested improvements to parts that required more polishing. After the demonstration session, the work was approved by the IMC and it was ready to be released.

5.6.1 Deploying website to the hosting provider's web server

After the website was approved by the IMC, it was released by deploying it to a hosting provider's web server. The website was deployed under the domain name *foodresearchgh.org*. The deployment process was completed in three phases, which all required the use of different application tools.

In the first phase, all project files residing on localhost's directory were compressed as a zip file,that was transferred to the web server with FileZilla, FTP client application. Finally, the contents of the zip file were extracted into a new directory on the web server.

In the second phase, new MySQL database for the website was created within the web server by using the hosting provider's CPanel application. Next, the localhost's MySQL database was exported as a database dump file by using localhost's phpMyAdmin database management application. Finally, the contents within the dump file were exported into web server's newly created database by executing the dump file's insert statements with server's phpMyAdmin application.

In the third and final phase, website's configurations were set by editing the new connection parameters

to a config file residing on the website's root directory on server's side. Finally, the deployment work phase was concluded when the website could be publicly accessed by entering the website's Available at address.

6.0 Conclusion and Recommendation

The Information Management Team of the CSIR-Food Research Institute has successfully developed and implemented an official website for the Food Research Institute of the Council for Scientific and Industrial Research using Joomla Content Management System.

Joomla CMS was a familiar platform from the Team's previous projects, so choosing it for this implementation was a safe choice.

This project implementation was quite laborious and challenging, but also useful and rewarding. The Team learned from each work phase something new. The final website ended up being functional and decent and it satisfied the Management of Food Research Institute at the end.

Although the website is now fully functional and operational, content is needed to keep it up to date. The Team therefore recommends that the Institute set up an editorial committee who will provide the Webmaster with the requisite content needed for regular update of the website.

References

Baxter, S & Vogt L.C. 2002. Google Patents. Content Management System. US 6356903 B1. Available at: http://www.google.com/patents/US6356903. Accessed: 5 Sep 2014.

Bootsrap. CSS. Available at: http://getbootstrap.com/css/. Accessed. 17 Oct 2014.

Goodrich, R. 2013. Business News Daily. CMS: Content Management Systems & Software. Available at: http://www.businessnewsdaily.com/5148-content-management-systems.html

Joomla. 2013. Joomla 3: Mobile R3ADY / US3R Friendly. Available at: http://www.joomla.org/3/en. Accessed: 14 Oct 2014.

Joomla! Docs. 2012. Top 10 Stupidest Administrator Tricks. Available at: http://docs.joomla.org/Top_10_Stupidest_Administrator_Tricks. Accessed: 5 Sep 2014.

Joomla! Docs. 2013a. Understand How Joomla Works. Available at: http://docs.joomla.org/Portal:Beginners/Understand_How_Joomla_Works. Accessed: 7 Sep 2014.

Joomla! Docs. 2013b. Administration of a Plugin in Joomla. Available at: http://docs.joomla.org/Administration_of_a_Plugin_in_Joomla. Accessed: 7 Sep 2014.

Joomla! Docs. 2013c. Getting Started with Templates. Available at: http://docs.joomla.org/J3.x:Getting_Started_with_Templates. Accessed: 7 Sep 2014.

Joomla! Docs. 2013d. Extension types (general definitions). Available at: http://docs.joomla.org/Extension_types_ (general_definitions). Accessed: 5 Sep 2014.

Joomla! Docs. 2013e. Beginners. Installing Joomla. Available at: http://docs.joomla.org/Beginners#Installing_Joomla.21. Accessed: 14 Oct 2014

Joomla! Docs. 2013f. Using nested sets. Available at: http://docs.joomla.org/Using_nested_sets. Accessed: 14 Oct 2014.

Joomla! Docs. 2013g. How To Use Content Tags in Joomla!. Available at: http://docs.joomla.org/J3.x:How_To_Use_Content_Tags_in_Joomla. Accessed: 16 Oct 2014.

Joomla. 2014a. About Joomla. Available at: http://www.joomla.org/about-joomla.html. Accessed: 7 Sep 2014.

Joomla. 2014b. About the Joomla Project. Available at: http://www.joomla.org/about-joomla/the-project.html. Accessed: 7 Sep 2014.

Joomla. 2014c. Technical Requirements. Available at: http://www.joomla.org/technical-requirements.html. Accessed: 5 Sep 2014.

Joomla! Docs. 2014a. Extension. Available at: http://docs.joomla.org/Extension. Accessed: 7 Sep 2014.

Joomla! Docs. 2014b. Component. Available at: http://docs.joomla.org/Component. Accessed: 7 Sep

2014.

Joomla! Docs. 2014c. Module. Available at: http://docs.joomla.org/Module. Accessed: 7 Sep 2014.

Joomla! Docs. 2014d. Plugin. Available at: http://docs.joomla.org/Plugin. Accessed: 7 Sep 2014.

Joomla! Docs. 2014e. Template. Available at: http://docs.joomla.org/Template. Accessed: 7 Sep 2014.

Joomla! Docs. 2014f. Security: Available at: http://docs.joomla.org/Security. Accessed: 5 Sep 2014.

Joomla Security Info. 2014. Available at: http://www.joomlasecurity.org/. Accessed: 5 Sep 2014.

RocketTheme. 2014. Templates. Chimera Joomla template. Available at: http://demo.rockettheme.com/live/joomla/chimera/ Accessed: 17 Oct 2014.

Just Creative. 2012. The Ultimate Web Design Workspace for Photoshop. Available at: http://justcreative.com/2012/04/17/web-design-workspace-photoshop/. Accessed: 17 Oct 2014.

LESS. Language Features. Available at: http://lesscss.org/features/. Accessed: 14 Oct 2014.

Morville, P. 2012. Understanding Information Architecture. Available at: http://prezi.com/aafmvya6bk7t/understanding-information-architecture/. Accessed: 6 Sep 2014.

Morville, P. & Rosenfeld L. 2006. Information Architecture for the World Wide Web. 3rd ed. O'Reilly. Sebastopol, CA.

Petersen, E. 2012. Nested Sets. Available at: http://www.evanpetersen.com/item/nested-sets.html. Accessed: 14 Oct 2014.

Shreves, R. 2010. Joomla Bible. Whiley Publishing Inc. Indianapolis, CA.

SiteGround. 20014. Joomla Security Tutorial. How to Secure Joomla 3 and Protect it Against Hacker Attacks. Available at: http://www.siteground.com/tutorials/joomla/joomla-security.htm. Accessed: 5 Sep 2014.

Sonspring. 2008. 960 Grid System. Available at: http://sonspring.com/journal/960-grid-system. Accessed: 17 Oct 2014.

TechTarget. 2011a. Content Management System (CMS). Available at: http://searchsoa.techtarget.com/definition/content-management-system. Accessed: 6 Sep 2014.

TechTarget. 2011b. Content Management Application (CMA). Available at: http://searchcontentmanagement.techtarget.com/definition/content-management-application-CMA. Accessed: 7 Sep 2014.

Usability.gov. 2014a. Information Architecture Basics. Available at: http://www.usability.gov/what-and-why/information-architecture.html. Accessed: 6 Sep 2014.

Usability.gov. 2014b. Information Architecture Methods. Available at: http://www.usability.gov/how-

to-and-tools/methods/information-architecture/index.html. Accessed: 6 Sep 2014.

Usability.gov 2014c. Organization Schemes. Available at: http://www.usability.gov/how-to-and-tools/methods/organization-schemes.html. Accessed: 6 Sep 2014.

Usability.gov 2014d. Organization Structures. Available at: http://www.usability.gov/how-to-and-tools/methods/organization-structures.html. Accessed: 6 Sep 2014.

Usability.gov 2014e. Content Inventory. Available at: http://www.usability.gov/how-to-and-tools/methods/content-inventory.html. Accessed: 6 Sep 2014.

Usability.gov 2014f. Wireframing. Available at: http://www.usability.gov/how-to-and-tools/methods/wireframing.html. Accessed: 6 Sep 2014.

Verens, K. 2010. CMS Design Using PHP and jQuery. Packt Publishing Ltd. Birmingham, UK.

W3Techs. 2014. Usage of content management systems for websites. Available at: http://w3techs.com/technologies/overview/content_management/all. Accessed: 7 Sep 2014.

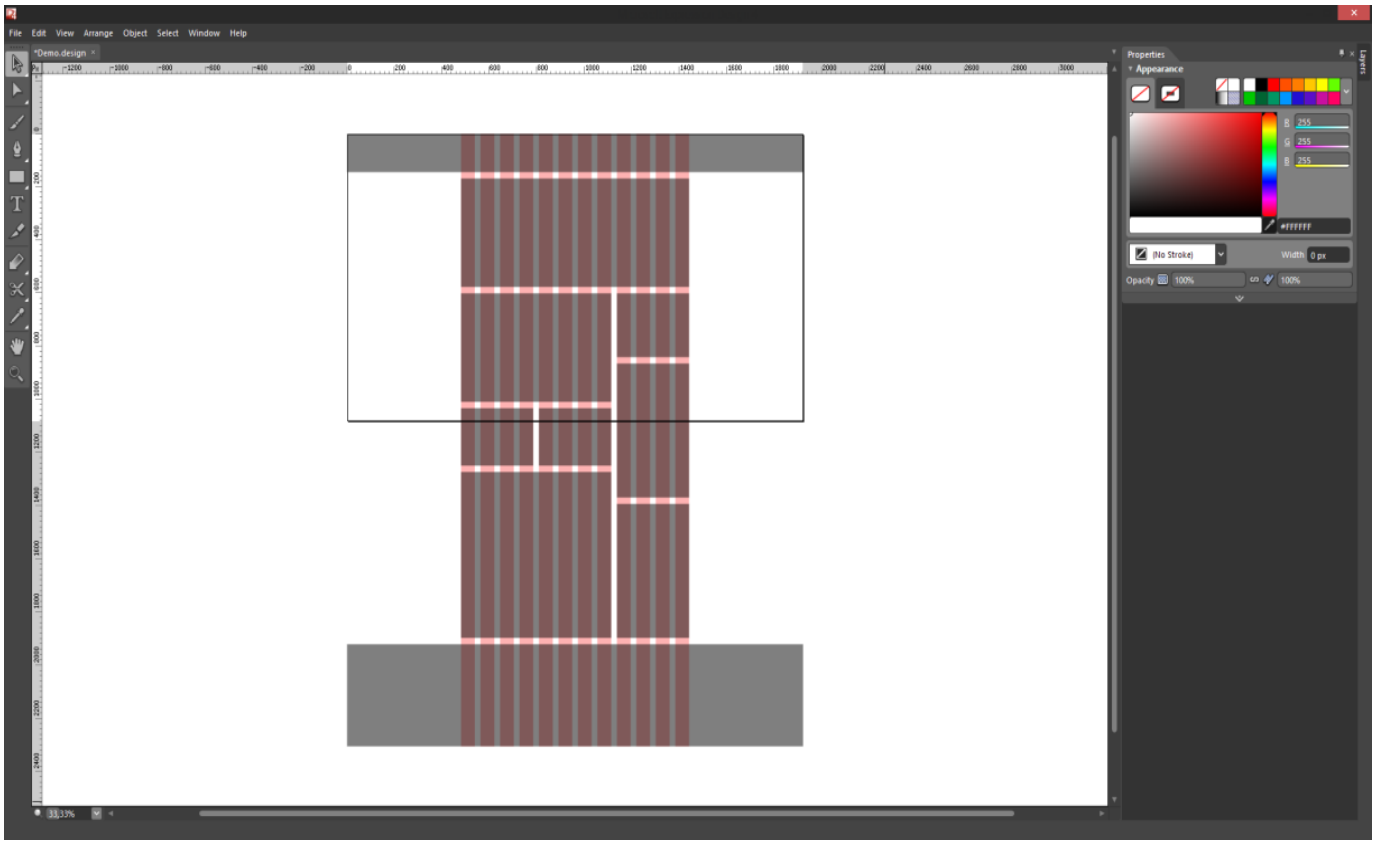
Web Designer Depot. 2010. The Myth of DPI. Available at: http://www.webdesignerdepot.com/2010/02/the-myth-of-dpi/. Accessed: 17 Oct 2014.

Webopedia. Pixel. Available at: http://www.webopedia.com/TERM/P/pixel.html. Accessed: 17 Oct 2014.

Wikipedia. 2006. Template Engine. Available at: http://en.wikipedia.org/wiki/File:TempEngWeb016.svg. Accessed: 6 Sep 2014.

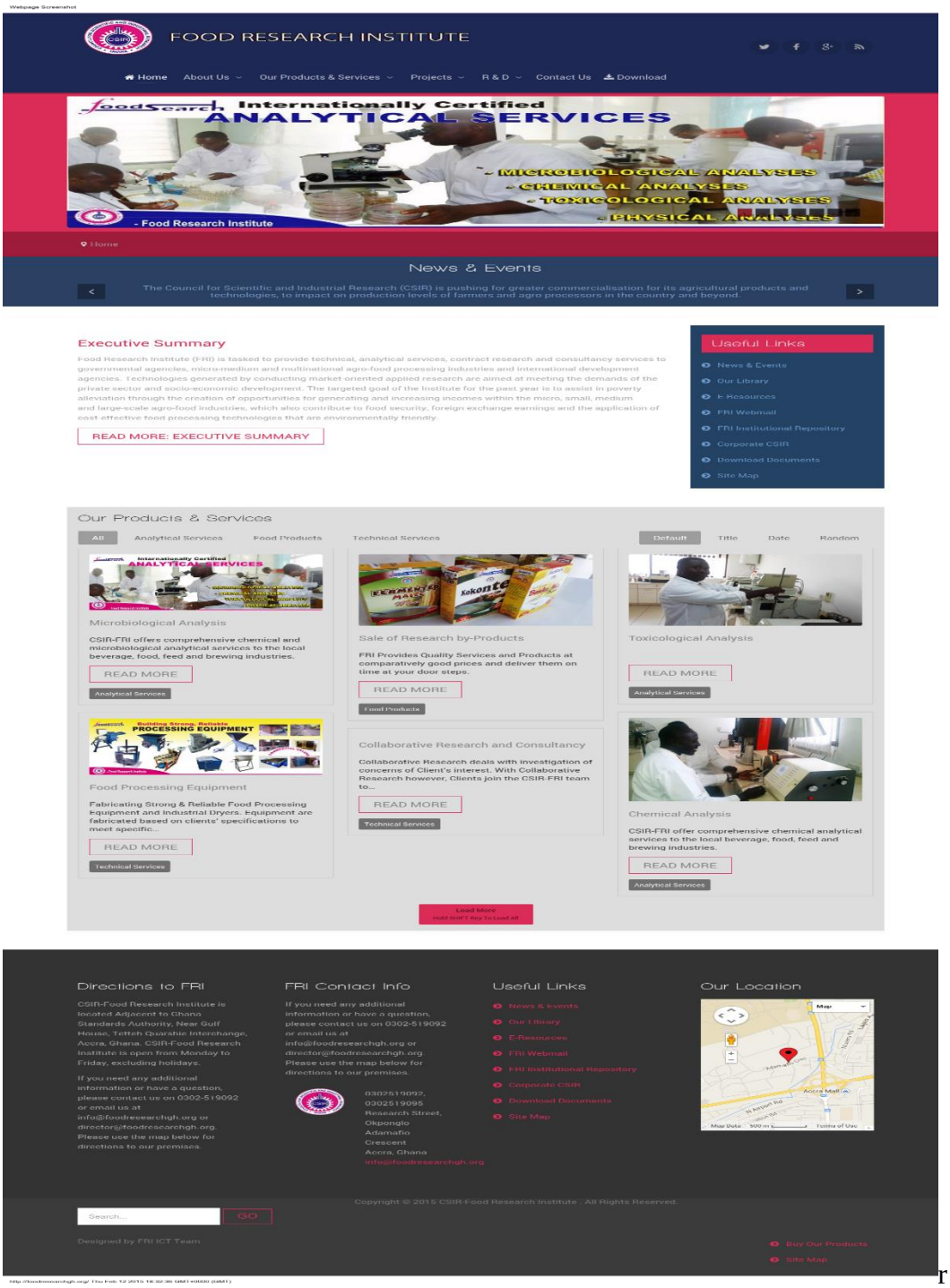
Appendices

Appendix 1: Screenshots of the homepage mock-ups



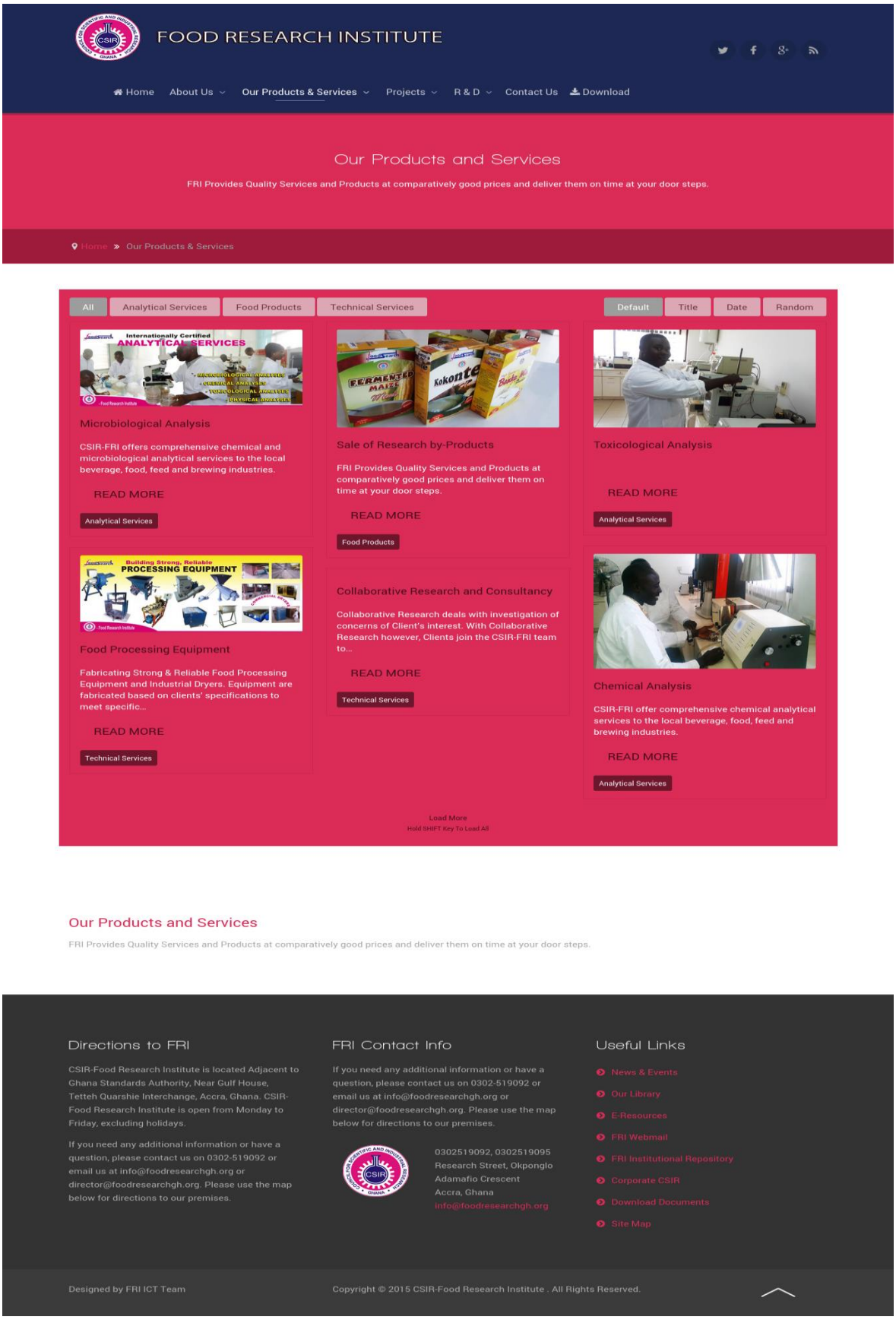
Screenshots of the finished website.

Appendix 2: Homepage of the finished website

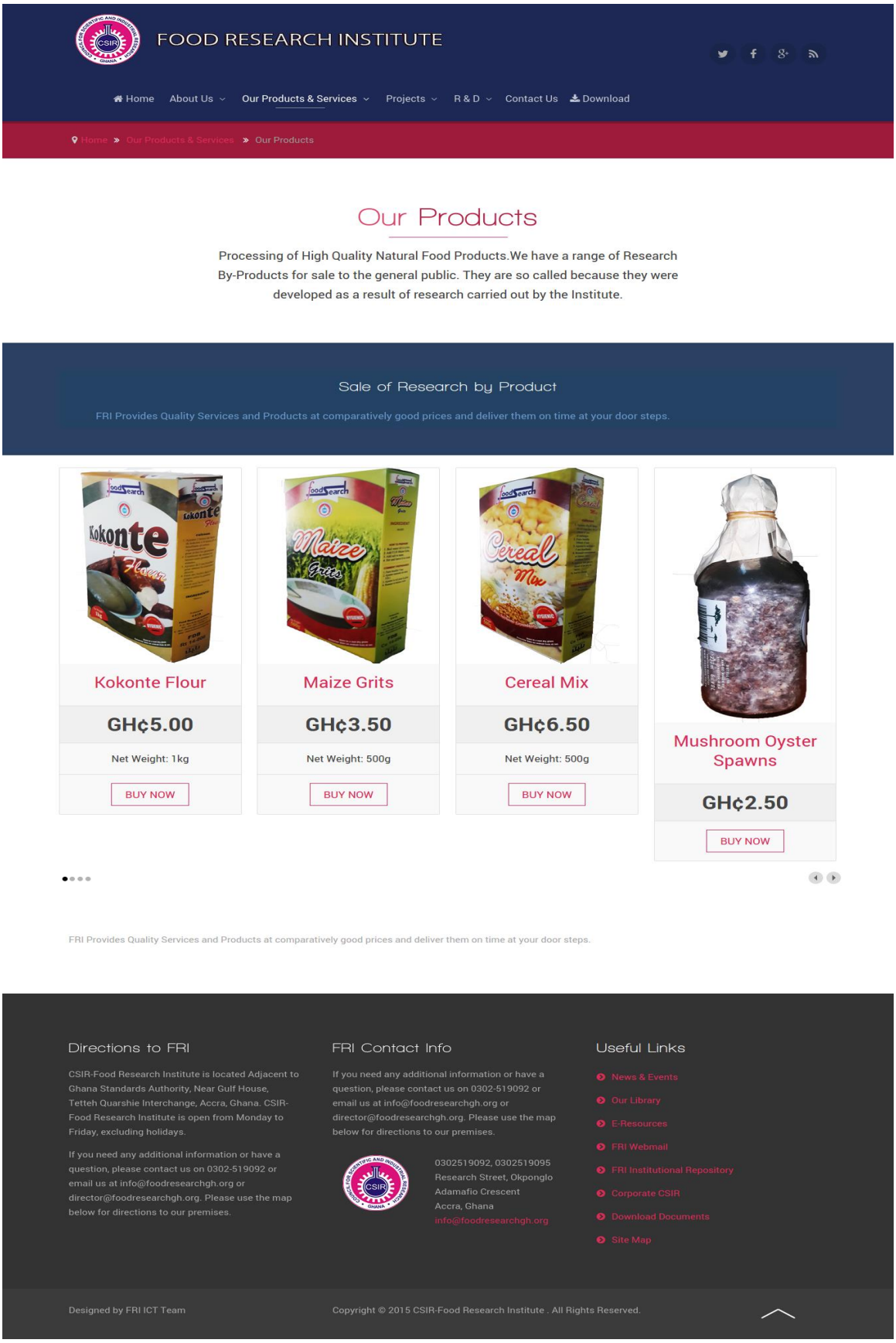


Appendix 3: About page

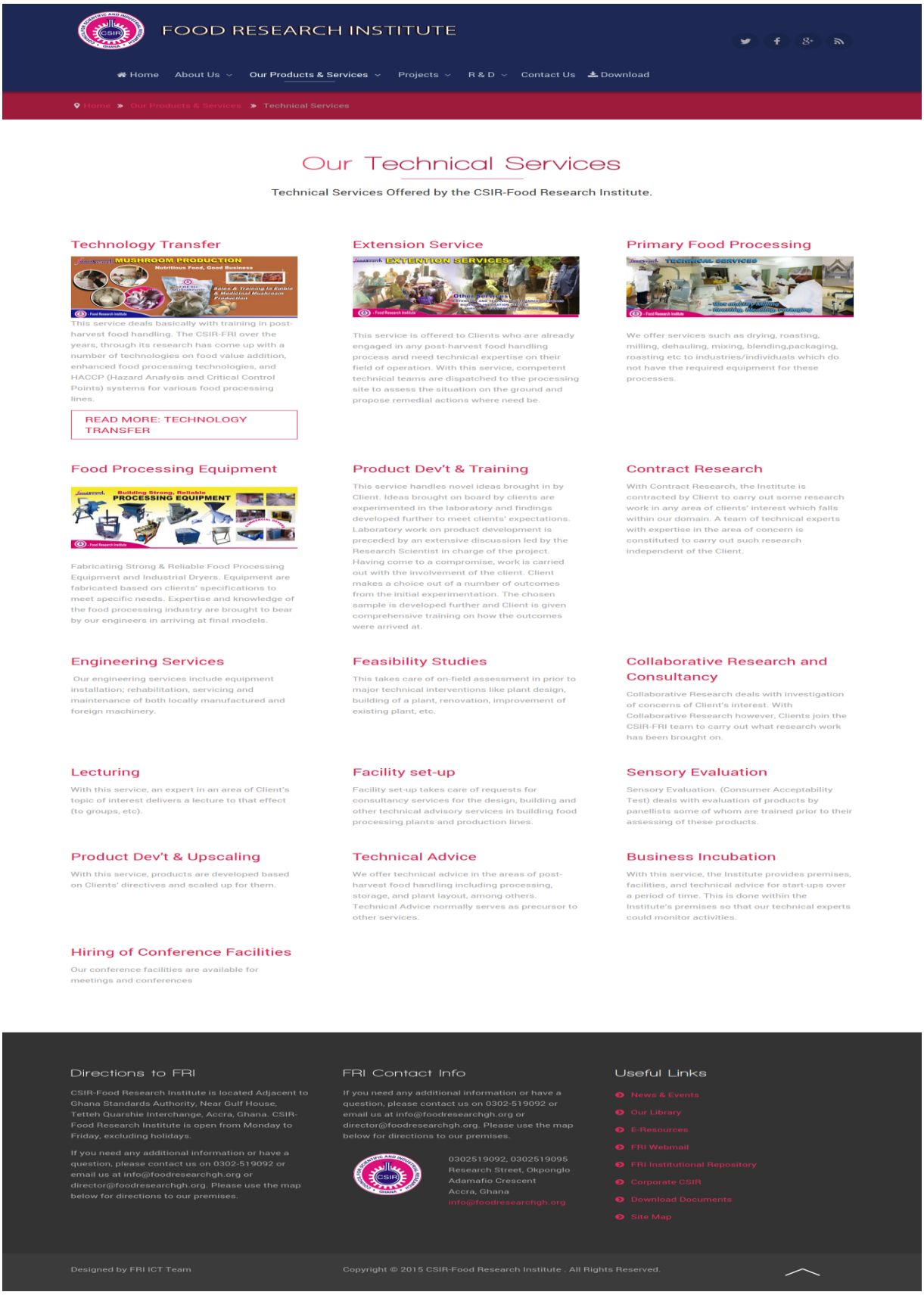
Appendix 4: Our products and services page



Appendix 5: Products page



Appendix 6: Technical services page



Appendix 7: R & D page

FOOD RESEARCH INSTITUTE

Home About Us Our Products & Services Projects R & D Contact Us Download

R & D

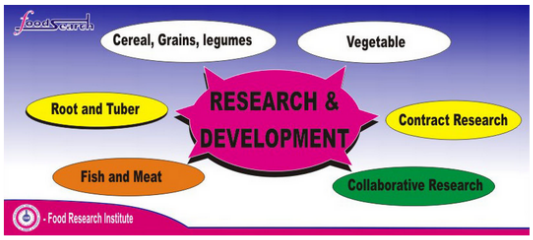
Our Research and Development Programmes

Home » R & D

Research & Development

Per our mandate in the food value chain, we play various roles from the farm gate to the consumer – post-harvest food handling. Sometimes however, for optimum results which so require that our expertise be brought to bear prior to harvesting / processing (value addition), we sometimes collaborate with the stakeholders at that level of the value chain.

We mainly research and come up with technologies for value addition, and solutions to food safety and security issues in the value chain.
Our competent research scientists and technical staff (technicians and technologists), based on their areas of expertise, have been grouped into sections to handle various food programmes.
Our expertise is in the areas of:



- Root and Tuber Products Programme
- Meet, Fish, Poultry and Dairy Products Programme
- Cereal and Grain Legume Products Programme
- Fruits and Vegetable Products Programme

Directions to FRI

CSIR-Food Research Institute is located Adjacent to Ghana Standards Authority, Near Gulf House, Tetteh Quarshie Interchange, Accra, Ghana. CSIR-Food Research Institute is open from Monday to Friday, excluding holidays.

If you need any additional information or have a question, please contact us on 0302-519092 or email us at info@foodresearchgh.org or director@foodresearchgh.org. Please use the map below for directions to our premises.

FRI Contact Info

If you need any additional information or have a question, please contact us on 0302-519092 or email us at info@foodresearchgh.org or director@foodresearchgh.org. Please use the map below for directions to our premises.

0302519092, 0302519095
Research Street, Okponglo
Adamafio Crescent
Accra, Ghana
info@foodresearchgh.org

Useful Links

- News & Events
- Our Library
- E-Resources
- FRI Webmail
- FRI Institutional Repository
- Corporate CSIR
- Download Documents
- Site Map

Designed by FRI ICT Team

Appendix 8: R & D page

FOOD RESEARCH INSTITUTE

Home About Us Our Products & Services Projects R & D Contact Us Download

Home » R & D » Root and Tuber Products Program

Root and Tuber Products Program



Root and Tuber Products Program

Directions to FRI

CSIR-Food Research Institute is located Adjacent to Ghana Standards Authority, Near Gulf House, Tetteh Quarshie Interchange, Accra, Ghana. CSIR-Food Research Institute is open from Monday to Friday, excluding holidays.

If you need any additional information or have a question, please contact us on 0302-519092 or email us at info@foodresearchgh.org or director@foodresearchgh.org. Please use the map below for directions to our premises.

FRI Contact Info

If you need any additional information or have a question, please contact us on 0302-519092 or email us at info@foodresearchgh.org or director@foodresearchgh.org. Please use the map below for directions to our premises.

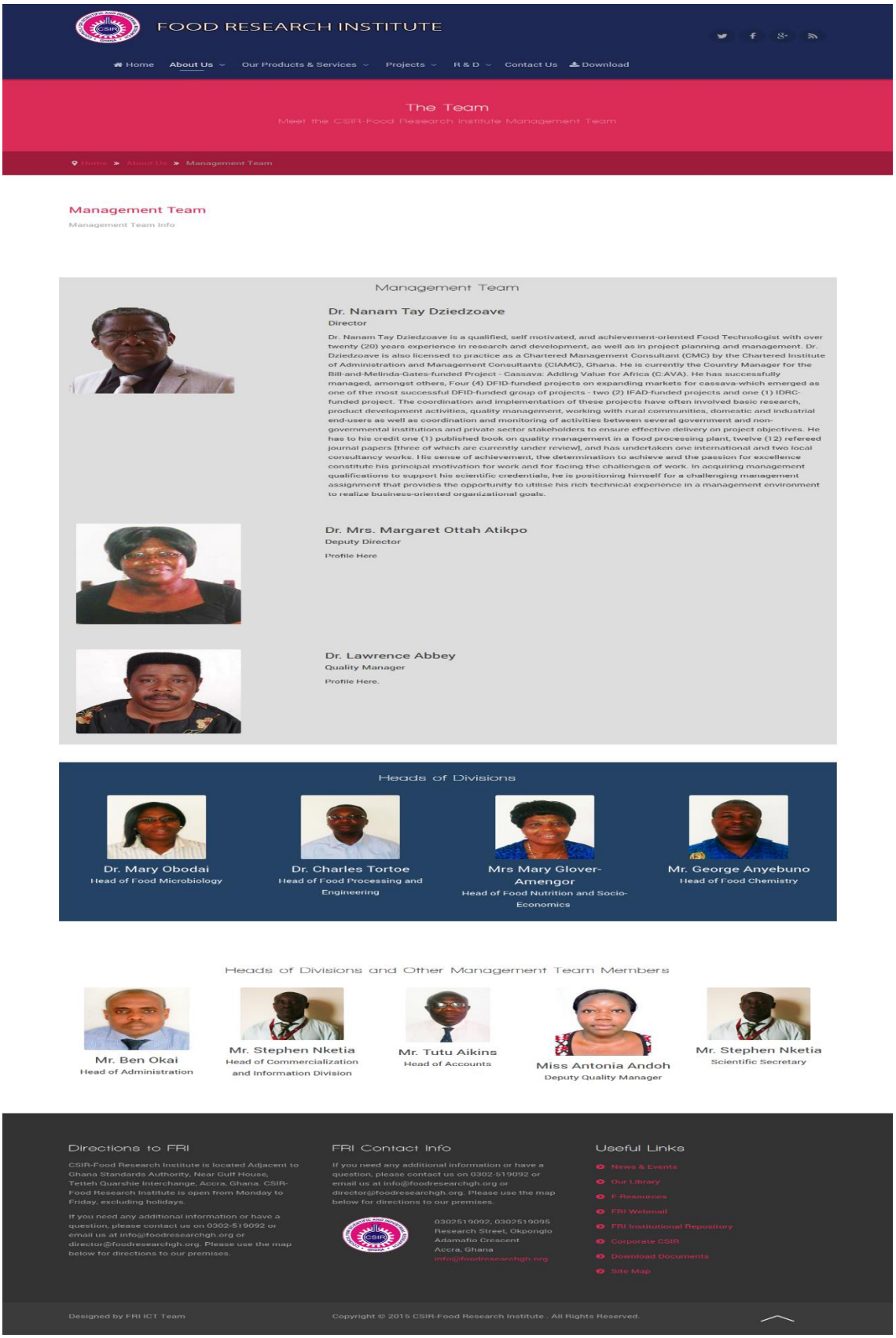
0302519092, 0302519095
Research Street, Okponglo
Adamafio Crescent
Accra, Ghana
info@foodresearchgh.org

Useful Links

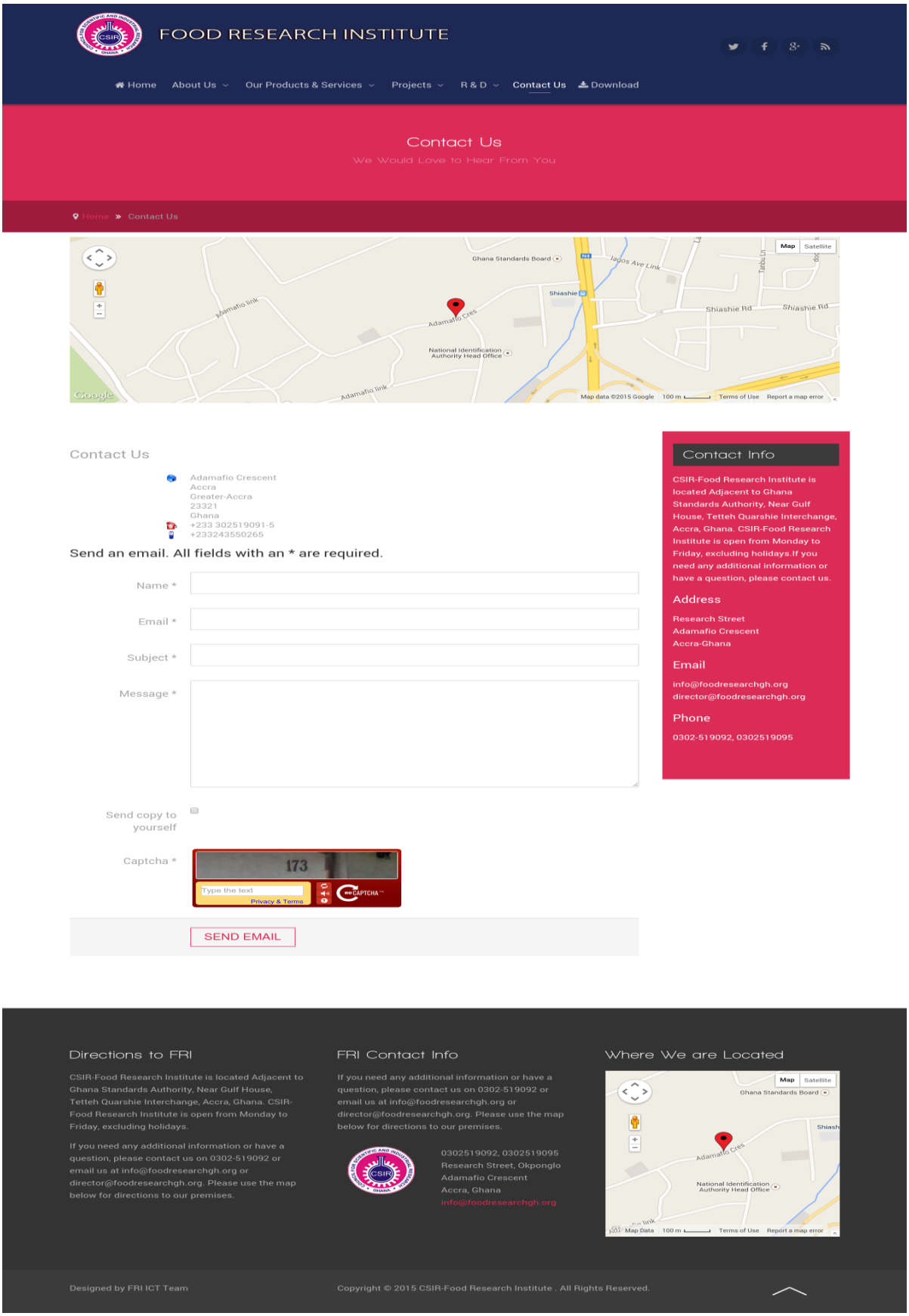
- News & Events
- Our Library
- E-Resources
- FRI Webmail
- FRI Institutional Repository
- Corporate CSIR
- Download Documents
- Site Map

Designed by FRI ICT Team

Appendix 9: Management Team page



Appendix 10: Contact page



Appendix 11: News & Events page

FOOD RESEARCH INSTITUTE

Home | About Us | Our Products & Services | Projects | R & D | Contact Us | Download

Home » News & Events

News & Events

Access all the Latest News & Events at the CSIR-Food Research Instititute, Ghana.

CSIR-FRI welcomes new Director General

The new Director –General (in black suit) of the Council for Scientific and Industrial research (CSIR), Dr. Victor Kwame Agyeman has paid glowing tribute to Dr. Abdulai Baba Salifu, out-going Director General who he said has transformed and achieved a lot for the CSIR for the past six years. He said "we need to recognise people who have worked for CSIR and acknowledge their services" . He said this after he was introduced to the staff of CSIR-Food Research Institute on Friday 16th January 2015.

CSIR to address challenges in food security and environmental management

Dr. George Owusu Essegbey, Director of CSIR Science and Technology Policy Research Institute (STEPRI), has said that the institute is concerned with the application of Science and Technology to address development challenges in Agriculture, Industry and the Environment.

READ MORE: CSIR TO ADDRESS CHALLENGES IN FOOD SECURITY AND ENVIRONMENTAL MANAGEMENT

CSIR pushes for commercialisation of its agricultural technologies

The Council for Scientific and Industrial Research (CSIR) is pushing for greater commercialisation for its agricultural products and technologies, to impact on production levels of farmers and agro processors in the country and beyond.

READ MORE: CSIR PUSHES FOR COMMERCIALISATION OF ITS AGRICULTURAL TECHNOLOGIES

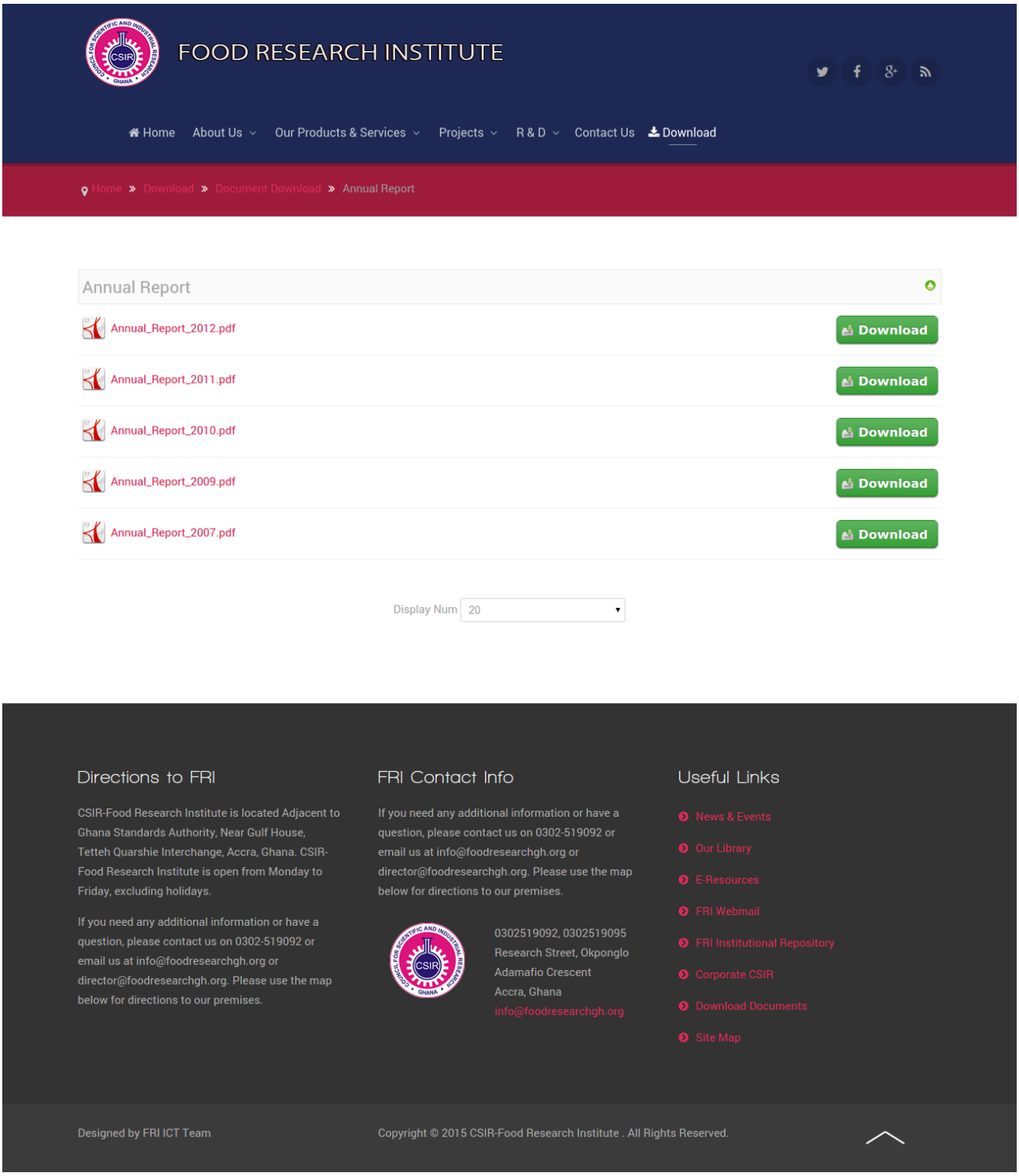
News & Events

CSIR to address challenges in food security and environmental management

Dr. George Owusu Essegbey, Director of CSIR Science and Technology Policy Research Institute (STEPRI), has said that the institute is...

READ MORE

Appendix 12: Download page



EDITORIAL COMMITTEE OF CSIR-FOOD RESEARCH INSTITUTE