**Web Applications Development. Final Project Guidelines and Proposal.**

1. **Project description**
   1. **Decide why you need a web application.**
   2. **Scope**
   3. **Vision**
   4. **Objective**
   5. **Define the primary content of your web application**
      1. **Ajax, services, html, css, session, login, etc.**
      2. **Database, programming languages involved**
   6. **Define your secondary content of your web application**
      1. **Plugins**
      2. **Open source styles**
2. **Requirements**
   1. **Functionality and usability**
   2. **Style and layout**
   3. **Functional requirements**
      1. **User Requirements**
      2. **System Requirements**
   4. **Non-functional requirements**
      1. **User Requirements**
      2. **System Requirements**
3. **Specifications**
   1. **Functional Specifications**
   2. **Design Specifications**
   3. **Technical Specifications**
4. **System Architecture**
   1. **Software architectural pattern**
   2. **Development platform**
   3. **Components**
      1. **Front-end**
      2. **Back-end**

**CV Maker**

**I.a. Decide why you need a web application.**

CV Maker searches to implement a simple and intuitive tool for people to easily make a CV, they will have several sections to choose from and even the ability to add your own personalized items. This tool is targeted for people struggling to make their CV, even if it’s not their first time, it gives a clean classic look so no time is wasted on the form and they can focus on the content.

**I.b Scope**

This web application is to be Delivered on May 2, 2019, the typical user will be people looking to make or update their CV, with this project I hope to prove the knowledge of the topics learnt in class as there is no real final client, content used on this project will be mostly from stock material and credited accordingly.

This is a good example of the desired project <https://cvmkr.com/>, with our approach we are looking for something intuitive and easy to use.

**I.c Vision**

A project’s vision is the desired state or ultimate condition that the project is working to achieve.

**I.d Objective**

The project objective describes the project’s outcomes: intended and direct, short- and medium-term effects on the target group. The project objective must lie within the scope of the project, and one must be able to directly attribute the effects to the project.

**I.e Define the primary content of your web application**

The primary content of your site is the information that a visitor will need to find the products or services they need and contact you. The list of your key products or services.

For example:

**Home page**: An overview of what your web application does

**Product 1:**Details of product range 1 including prices, technical details, illustration etc

**Product 2:** Product range 2 etc.

**Contact us:** A page with full contact details and maybe a map.

**I.f Define your secondary content of your web application**

A site with primary content alone could be a bit dry. If you only show your products then your potential customers may not appreciate your expertise. Your site will be more effective if visitors can see there is a human behind it rather than a computer.

For example:

**Case studies:** Short articles describing how your services will benefit clients

**About us:** A page with details about your web application, perhaps including a short history.

**II.a Functionality and usability**

Decide what you find most usable and write it in to your specification

**II.b Style and layout**

Explain your designing approach. How will your page will be organized visually? Templating. Open source layouts and styles.

**II.c Functional Requirements**

Functional requirements describe what the software or web site is supposed to do by defining functions and high-level logic.

**II.d Non-functional Requirements**

Non-functional requirements specify criteria that can be used to judge the operation of a system, rather than specific behaviors.

**III.a Functional Specifications**

Functional specifications describe the necessary functions at the level of units and components; these specifications are typically used to build the system exclusive of the user interface.

**III.b Design Specifications**

The design specifications address the “look and feel” of the interface, with rules for the display of global and particular elements.

**III.c Technical Specifications**

Technical specifications describe how developers will implement the project. The developers work from the functional specifications, and translate the functions into their actual coding practices and methodologies.

**IV System Architecture**

A system architecture illustrates the way the system hardware and software must be configured, and the way the database tables should be defined and laid out.

**IV. a Architectural Pattern**

Set of principles—a coarse grained pattern that provides an abstract framework for a family of systems. An architectural pattern improves partitioning and promotes design reuse by providing solutions to frequently recurring problems.