HTML FRAMEWORKS

1. **What is a Framework?**

A framework in general is a **standardized** set of concepts, practices and criteria for dealing with a common type of problem. In web design, frameworks are a set of files and folders of standardized code used to support the development of websites. Most websites have a similar structure; frameworks, therefore provide this basic template on which developers will build upon instead of starting from scratch. They could be considered “cheat sheets” that make the developers’ work easier.

1. **Types of Frameworks**

There are essentially two types of frameworks, depending on the process is being carried out on the web i.e.:

* Front end - designer’s layer (the visual – what the user sees)
* Back end - the programmer’s layer (that which connects the database to the site)
  1. Frontend Frameworks

The components of this type of framework package are:

* CSS Source code to create a grid – to position stuff in
* Typography style definitions for HTML elements
* Solutions for cases of browser incompatibility so the site displays correctly in all browsers.
* Creation of standard CSS classes which can be used to style advanced components of the user interface.

These frameworks also facilitate the development of responsive websites. These are websites that that can adapt to various resolutions for different mobile and desktop devices.

Within CSS frameworks, a distinction has been made between simple and complete frameworks

Depending on their level of ‘complexity’. This distinction, however, is subjective.

Simple frameworks: Often called “grid systems”

They offer hey offer style sheets with column systems to facilitate the distribution of different elements according to the established design. Examples: 1140 CSS grid, Titan, Ingrid etc.

Complete Frameworks: They usually offer complete frameworks with configurable features like styled-typography, sets of forms, buttons, icons and other reusable components built to provide navigation, alerts, popovers, and more, images frames, HTML templates, custom settings, etc.

Examples: Bootstrap, skeleton, tuktuk etc.

1. **How to choose a framework**

Not simple for the following reasons:

1. Different sites require different characteristics.
2. There’s not that many differences between the frameworks themselves.

Some other things to consider while making a choice are:

* Speed of installation: some are very simple to install and start using. Others require more time to configure.
* Ease of understanding: some are a bit of a pain to get to grips with, complicated. Others, by contrast, are comparatively more straightforward.
* Options: some frameworks are more complex than others and offer more configuration options, widgets and interface options. These will allow you to do better things with your site.
* Integration with other systems.
* Best long-term support: Some digital projects lack continuity in time and updates and support services stop. It’s always better to opt for those that offer continued support guarantees. Many of them are supported by companies that offer other professional products on the market.

1. **Advantages and Disadvantages**

Advantages

1. Speeds up the mock-up process
2. Clean and tidy code
3. Solutions to common CSS problems
4. Browser compatibility
5. Learn good practices
6. Having a single procedure to resolve common problems makes maintaining various projects more straightforward.
7. Helpful in collaborative work

Disadvantages

1. Mixes content and presentation
2. Unused code leftover
3. Slower learning curve
4. You don’t learn to do it yourself
5. **To use or not to use?**

It totally depends on the user.