

Using the same frameworks for JISF and why does it matter?

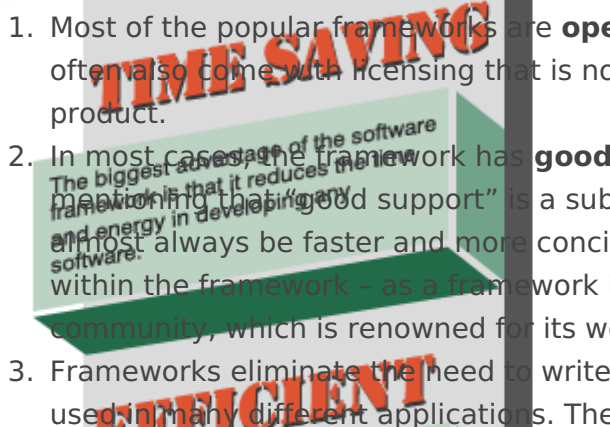
As all great endeavors start with a question. Every business or system, at some point, faces the same question, regardless of the industry niches they target or position that they occupy. How to get better at what we do? This remains a persistent need for the business of every size, regardless of a software company or a business or a government organization down the street that makes canoes. Once they start doing it correctly and they realize they need to manage more efficiently, make it speedy and ameliorate it with time.

This is the point there is a need for software process improvement (SPI) and stepping up for the same framework system for all software or application development. A business always focuses on delivery but a need to improve internal processes should not be an ancillary concern either.

Web application development has become an inevitable need for businesses of all types and sizes, regardless of their value proposition. That said, there are several ways to develop a web application, the most popular one being, 'the use of frameworks.' Web development frameworks have grown in popularity over the past several years as they provide a straightforward approach to developing custom web apps. Besides, developers get an increasingly responsive and intuitive coding interface that facilitates seamless programming in the supported languages. Above all, frameworks provide a simplified representation of the supported languages, enabling developers to code swiftly and get the job done faster. This whole process will be applicable to the JISF for better communication, robust service policy, and data security.

Figure: Framework in JISF

5 Major Usage of Frameworks:

- 
1. Most of the popular frameworks are **open-source** (or available to use for free). They often also come with licensing that is not restrictive and allows to build a commercial product.
 2. In most cases, the framework has **good documentation and support**. It is worth mentioning that "good support" is a subjective issue at times. Typically, paid support will almost always be faster and more concise, but this also depends on the level of activity within the framework – as a framework like Ruby on Rails demonstrates with a massive community, which is renowned for its welcoming nature and good support too.
 3. Frameworks eliminate the need to write a lot of repetitive code that you will find being used in many different applications. The advantage of **efficiency** will never be

underestimated. You can expect to build a project in much less time than would be achieved writing code without a framework.

4. As far as a framework usually developed and tested by many different developers it can gain a strong level of **security**. It is extremely likely that many security risks are addressed and tested when the framework is being built
5. **Integration**: for building almost any type of application (including a website) where you want to store some data, you will typically use a database. There also exist many other tools that link to web development. Many frameworks will thus make it easier to link to these tools and also communicate with them.

Revision #4

Created 25 September 2021 16:36:22 by Niton

Updated 28 September 2021 20:54:03 by Niton