



# UTT

UNIVERSIDAD TECNOLÓGICA DE TIJUANA

**GOBIERNO DE BAJA CALIFORNIA**

**TEMA:**

Framework Selection

**PRESENTADO POR:**

Paredes Nevarez Alexis Omar

**GRUPO:**

10B

**MATERIA:**

Software Development Process Management

**PROFESOR:**

Ray Brunett Parra Galaviz

**FECHA:**

**07/10/2025**

## **Framework Selection: A Critical Step in Software Development**

The selection of a software framework is a fundamental decision that can significantly impact the trajectory of a project. Frameworks provide pre-built modules, libraries, and tools that streamline development, improve efficiency, and ensure consistency across applications. However, choosing the wrong framework can result in inefficiencies, technical debt, and limitations in scalability. To make an informed decision, it's essential to evaluate frameworks based on several critical factors.

### **Key Factors to Consider**

#### **1. Popularity and Community Support**

Frameworks with a large and active community are often the most reliable. A robust community ensures that developers have access to extensive resources such as tutorials, plugins, and libraries. Additionally, popular frameworks tend to have more frequent updates and patches, addressing security vulnerabilities and keeping the framework aligned with current technologies. Examples include React and Django, which have vibrant ecosystems and strong developer support.

#### **2. Alignment with Project Requirements**

Different frameworks are designed for specific use cases. For instance, front-end frameworks like Angular or Vue.js are ideal for building interactive user interfaces, while back-end frameworks like Express.js or Flask are suited for server-side logic. It's crucial to analyze the project's requirements, including performance needs, scalability, and functionality, to select a framework that meets these demands.

#### **3. Sustainability and Long-Term Viability**

Frameworks evolve over time, and some may become obsolete. Choosing a framework with active maintenance and a history of updates ensures that your application remains functional and secure in the long term. Open-source frameworks supported by large organizations, like Google's Angular or Facebook's React, often provide this reliability.

#### 4. **Security Features**

Security is a non-negotiable aspect of software development. Frameworks should provide built-in mechanisms to address common vulnerabilities, such as Cross-Site Scripting (XSS) and SQL Injection. Additionally, they should integrate well with third-party security tools to enhance the overall security posture of the application.

#### **Best Practices for Framework Selection**

- **Prototype and Test:** Before committing to a framework, build a small prototype to evaluate its features and limitations.
- **Gather Team Feedback:** Involve developers in the decision-making process to ensure the framework aligns with their expertise and preferences.
- **Plan for Scalability:** Select a framework that can handle future growth in users, features, and data.
- **Evaluate Alternatives:** Compare multiple frameworks to identify the one that offers the best balance of features, performance, and cost.