Python with Django vs MERN

We need to weigh the pros and cons of using Python with Django vs MERN for our app development. Please share your opinion and insights on the following factors:

* Scalability: Is the language and technology capable of scaling the app to meet future demands?
* Maintenance: Is it easy to maintain the app over time?
* Development time: How long will it take to develop the app using this language and technology?
* Community support: Is there a large community of developers who use this language and technology and can provide support and resources?
* Cost: How much will it cost to develop and maintain the app using this language and technology?

Please leave your thoughts and any relevant information in the comments section below.

Note

If we are planning to develop an AI application using Python, it may be easier to use Python with Django instead of MERN, depending on the specific requirements of our project.

Django is a popular web framework for Python, which includes many built-in features and tools that can help simplify and streamline the development process. Additionally, Django has a large community of developers who contribute to the framework and provide support and resources, which can be helpful when developing a complex application like an AI system.

MERN, on the other hand, is a popular technology stack for developing web applications using MongoDB, Express, React, and Node.js. While MERN can also be used for developing AI applications, it may not have the same level of built-in features and community support as Django.

Ali Al-kinani

1. Scalability: Django has built-in support for scalability through its ability to handle large amounts of traffic and user data. MERN, on the other hand, requires additional tools and resources to handle scalability.
2. Maintenance: Django is a more established framework, which means that it has a larger community of developers who provide support and resources. MERN is a newer technology stack and may require more maintenance and updates to keep up with new technologies and security standards.
3. Development time: Django has a shorter development time due to its built-in features and ease of use. MERN may require more time for developers to learn and implement the different technologies.
4. Community support: Django has a larger community of developers who use the framework and provide support through forums, documentation, and open-source libraries. MERN has a growing community, but it may not have the same level of resources and support.
5. Cost: Django is a free and open-source framework, which means that it has no direct cost. However, some add-ons and third-party tools may have a cost associated with them. MERN requires more resources and tools to run, which may result in higher development and maintenance costs.

Nikhil Ranjan (Nik) -

|  |  |  |
| --- | --- | --- |
|  | **MEAN/MERN stack** | **Django** |
| **Security** | Great security, more hands-on development needed, more flexibility | Great Security – inbuilt security packages for almost everything. Only follow the framework for security |
| **Speed / performance** | Slower to build, faster – real time performance | Faster to build, Slower – Python is a high-level language |
| **Scalability** | Highly Scalable | - Scalable, less than MERN |
| **Support - community** | Great support and community | - Great support and community |
| **Complexity / time to build** | - More complex compared to Django – more work | - Easier to build, template like – maybe not enough work for everyone? |
| **Does it offer what we need?** | Yes | Yes |
| **Cost** | free | free |
| **Technology** | - Harder to integrate with python  - | - easily integrated with python projects  - Easy to integrate API’s into – potential future dev?  - visualisation with python libraries like matplotlib etc |

Links for reading   
<https://careerfoundry.com/en/blog/web-development/django-framework-guide/#what-is-the-django-framework>

<https://docs.djangoproject.com/en/4.1/topics/security/>

Evan -

|  |  |  |
| --- | --- | --- |
|  | **Django** | **MERN Stack** |
| Scalability | Still allows developer to take several scalability actions | Very adaptable and scalable |
| Maintenance | Large community can help with maintenance and providing common solutions | Require more maintenance to keep up with trends and updates |
| Development time | Rapid, easy and is suitable for speedy development because of its many built-in functionality | Slower and more complex to build |
| Community support | Strong community and quite supportive, beginner friendly | MERN is very popular including its community, and has been widely used many developers |
| Cost | Mostly free and open source, with optional add-ons | Free, but may require additional cost for certain elements to implement or function |

Mewantha Jayasekara -

* Scalability: MERN can scale the app to meet future demands and for the server-side programming, it uses NoSQL database and Node.js.
* Maintenance: MERN uses single programming language for its modular architecture. This helps us to maintain MERN easily over time.
* Development time: Unlike python with Django, MERN is able to create an app without consuming too much time.
* Community support: Many resources and support are available for MERN due to the considerable number of developers who use it.
* Cost: Due to the availability of many open-source tools, MERN is not costly to develop and maintain.

**Summary**:

* Django is easier to learn and implement – much faster, maybe not enough work for all people
* In-built security features from all sorts of issues
  + Cross site scripting - XSS
  + SQL injection protection
  + Cross site request forgery protection – CSRF
  + Clickjacking protection
  + SSL/HTTPS
  + etc
* Upsides include easy integration with ML, visualisation libraries in python can be used for graphs. API support for future endeavours
* As this web app does not need to perform necessarily fast, Django will work fine.
* For computer intensive tasks Django is better compared to MERN – Mern can cause bottlenecks potentially
  + MERN is single threaded, means requests are serialized and done one after another so until a captcha is done, the next might not execute.
  + Django would be able to do many in parallel