Architecture Decision Record for Python Django Framework

Decision Date: 2021-07-15

Status: Accepted

Context

Our organization is planning to develop a web application that manages customer data. We have chosen Python as the programming language and are considering Django as the web framework for the development of the application.

Decision

We have decided to use the Django web framework for the development of the web application. Django provides a robust set of tools and features for building web applications quickly and efficiently.

Factors

Some of the factors that influenced our decision include:

1. Object-Relational Mapping (ORM): Django has a built-in ORM that allows us to interact with the database without writing SQL queries. This makes it easier to develop the application and maintain it in the long run.

2. MVC framework: Django follows a Model-View-Controller (MVC) architecture, making it easier to separate the business logic and presentation layers of the application.

3. Scalability: Django is known for its scalability capabilities, making it an excellent choice for developing large-scale applications.

4. Security: Django has built-in security features, such as protection against common web attacks like cross-site scripting (XSS) and SQL injection.

5. Community Support: Django has a large and active community that provides support and contributes to the development of the framework.

Alternatives Considered

We considered other web frameworks such as Flask and Pyramid. However, we found that Django is a more mature and well-established framework with a robust set of features.

We also discussed developing the application without a web framework and using libraries like SQLAlchemy and Flask-RESTful. However, we found that Django offers broader functionality, making it a better choice for a complete web application.

Consequences

The adoption of Django will lead to the following consequences:

1. Easier to develop and maintain the application due to Django’s built-in tools and features.
2. Separation of business logic and presentation layer, leading to more organized and easier to maintain code.
3. Scalability and robustness of the application.
4. Built-in security features that help protect the application against common web attacks.
5. Access to a large and active community for support.

We understand that Django has a steeper learning curve than other frameworks, but we find that it is worth the investment for the long-term benefits it provides.

Conclusion

Based on the factors considered, we have decided to use the Django web framework for the development of the web application. We believe that Django’s features, community support, and scalability capabilities make it the best choice for building a complete web application. We will train our developers to use Django to ensure that the framework is used effectively and efficiently.