DP2 2023-2024

Report on WIS architecture Knowledge

Acme Software Factory

Repository: https://github.com/rafcasceb/Acme-SF-D02

Members:

- Castillo Cebolla, Rafael rafcasceb@alum.us.es
- Flores de Francisco, Daniel danflode@alum.us.es
- Heras Pérez, Raúl rauherper@alum.us.es
- Mellado Díaz, Luis luimeldia@alum.us.es
- Vento Conesa, Adriana adrvencon@alum.us.es

GROUP C1.049

Version 1.0

15-02-24

Contenido

Executive Summary

Web Information System (WIS) architecture refers to the structural design of a system that manages, processes, and delivers information over the internet. This report aims to provide an overview of our current understanding and knowledge of WIS architecture. The aforementioned initial evaluation will act as a reference point to measure our progress and learning outcomes as we progress through the course.

Revision Table

Date	Version	Description of the changes	Sprint
15/02/2024	v1.0	• Executive summary • Introduction • Content: Layers and considerations • Conclusion • Bibliography	1
17/02/2024	V1.1	• Content: entity, repository, service and controller concepts.	1

Introduction

An information system is a set of components working together to collect, process and store data to support decision-making and organizational processes. It includes hardware, software, data, procedures, and people.

A Web Information System (WIS) is a type of information system that operates over the internet. It allows users to access, interact with, and retrieve information through web browsers. Clear examples of WIS are websites, web applications, or online databases.

WIS architecture refers to the structural design of a Web Information System, more specifically, the organization and interaction of its components and layers.

Firstly, we will explore the fundamental concepts of WIS architecture, including its significance in modern web applications and its role in supporting information exchange. Next, we will talk about the different components or layers that constitute WIS architecture.

Finally, we will discuss considerations for scalability, performance optimization, and security measures within a WIS.

Contents

WIS architecture plays a key role in determining the system's performance, scalability, and reliability. Architecture considerations are essential for designing robust and efficient web applications that can handle varying levels of traffic and workload while safeguarding sensitive information from potential threats. In the following section, we will explain the main components or layers that constitute the architecture of a WIS.

Layers

Presentation Layer

The presentation layer is the user interface of the WIS, responsible for displaying information to users and collecting users' input. Those are: web pages, forms, multimedia content, and other elements that enable interaction with the system.

In the context of web applications, **controllers** handle user input and manage the flow of data between the presentation layer and the application layer. They interpret user actions and invoke corresponding actions in the application layer.

Application Layer

The application layer contains the business logic and functionality of the WIS. It processes user requests, performs operations over data, and applies business rules. This layer serves as an intermediary between the data layer and the presentation layer.

Services encapsulate business logic. They typically contain methods for performing specific operations or implementing business rules.

Data Layer

The data access layer is responsible for interacting with the data storage systems, such as relational databases, NoSQL databases, or external APIs. It handles tasks such as data retrieval, storage, and manipulation, ensuring data integrity and security. Object-relational mapping (ORM) frameworks are commonly used to abstract the database interactions and simplify data access code.

Entities represent the business objects or concepts being modelled in the system. They typically correspond to tables in a database in a relational database management system.

Repositories provide an abstraction layer for accessing and managing entities in the database. They encapsulate the logic for querying and persisting data, providing a clean interface for the application layer to interact with the database.

Considerations

Integration

Web integration allows to consume or provide services from/to other websites or applications. Facilitates communication and data exchange between different components of

the WIS and external systems. It may involve implementing APIs or middleware to enable integration with third-party systems, or other applications within the business.

Security

Nowadays a robust security is fundamental in WIS architecture to protect against unauthorized access, data breaches, and other cyber attacks. The security layer encompasses measures such as authentication, authorization, encryption, and input validation to safeguard sensitive information and ensure compliance with regulations and data protection laws.

Scalability and Performance

Scalability and performance are two factors to keep in mind. Scalability oversees the increasing traffic and growth of the WIS while performance is related to how fast the site's content is loaded and how well it responds to user interaction. These two considerations contribute to the overall quality of WIS service.

Conclusions

The architecture of a Web Information System (WIS) is a complex framework composed of multiple layers and components that work together to deliver functionality, reliability, and security. By studying WIS architecture, developers can design consistent systems that meet the evolving needs of users and organizations.

Bibliography

1. "Web Application Architecture: Types, Components & Models." Star Knowledge. https://star-knowledge.com/blog/web-application-architecture-types-components-models/#:~:text=Web%20application%20architecture%20is%20the,the%20data%20to%20the%20user.
2. "Web Information Systems" TU Delft. https://www.wis.ewi.tudelft.nl/
3. "Information System." Britannica. https://www.britannica.com/topic/information-system
4. "Business Logic Layer." ScienceDirect. https://www.sciencedirect.com/topics/computer-science/business-logic-layer#:~:text=The%20business%20logic%20layer%20contains,in%20the%20business%20logic%20layer.
5. "Website Integration." ActiveCampaign Glossary. https://www.activecampaign.com/glossary/website-integration
6. How to Build a Scalable Website." Nexcess. https://www.nexcess.net/blog/how-to-build-a-scalable-website/
7. "About web performance" Mozilla Developer Network. https://developer.mozilla.org/en-US/docs/Learn/Performance/What_is_web_performance