PROJECT 3 – Accommodations Web portal

Nowadays, people acquire most of the information online. They also want to do as many tasks as possible using the Web and Web applications. For this reason, providers are forced to present their products or services in an appropriate manner on the web.

The main purpose of the project is to create a Web application that will allow providers to present their holiday units on the one hand, and on the other hand allow searchers to review the offer and the reservation or rent (like Airbnb). The application should cover the offer in the territory of NSW.

There are quite a few websites with similar content for different territories, but they are mostly overloaded with information and, therefore, more or less opaque and useless. If we want the planned web portal to come to life, it is necessary to offer something new or different, that is, the opposite of existing websites.

This was the basic objective from which raised the basic starting points for the creation of the web portal:

- The layout of the portal should be easy
- The number of information on each page should not be too large
- Each page should only offer the information that the visitor requests

This project focuses on the creation of a generic, modular architecture for building web portals that can be used for accommodation or other similar services, which will be based on three core components:

1. Accommodation advertising module – for providers of accommodations to advertise properties and maintain their advertisements,
2. Visitor request module – for users to:
    a. put their requests and get a simple and useful feedback of the available accommodations,
    b. select and book preferred accommodation,
    c. (optional - publish a review about the accommodation)
3. An Application/UI component, which will enable to use the web portal for other similar services, for example, for selling and buying things or other types of exchanges.

A Python-based solution is preferred (but not required) using:

1) Modern development frameworks,

2) Cloud-based architectures, and

3) Building a solution, which leverages on existing services/APIs without creating a full dependency on the selected tools.

It is essential that the system requirements are refined with the stakeholders and that the engineering of the system is thought through properly before building the individual components.