**Comparing the impact of database systems, data serialization format, and communication protocols on the performance of web applications.**

Mostly, a web application will have 3 main components:

* **The client app:** it runs on a browser or a mobile, and it allows the users to use the services that the app provides.
* **The server app:** It runs on a server and handles users’ requests and delivers required data from the database to the users.
* **The database system:** It could be a relational or non-relational database system and users can only access the data through the server app.

Nowadays, there are dozens of technologies to achieve the implementation and the communication between those components which were mentioned above, every technology has existed for a reason, it addresses one or many more particular problems, and has its own pros and cons.

In our project, we are not introducing a million-dollar idea, neither we are focusing on the app functionality itself, rather we are concerned about some critical phases when building web applications.

More precisely, we want to discuss, analyze, and compare the options available for the following stages:

* **Database selection:** we will test the application’s performance on both sql and no-sql databases, we will try many engines like my sql, sql server, MongoDB, CouchDB, and study the time of the writing and reading operations, and we will study how the schema design should look like in every case.
* **Data serialization format:** when the server wants to send the data back to the client, it serializes it to a certain format (e.g., JSON, XML, ProtoPuf, MessagePack, …etc.), and it is been deserialized on the client-side, we will see the impact of these different formats on the size of the transferred bytes across the network, and analyze the serialization and deserialization time for each format.
* **Communication protocol:** whether the communication is between a client and a server, or it is between two services in a microservices architecture, the communication protocol plays a crucial role in determining the speed and the efficiency, we will compare the use cases for the common protocols like HTTP, WebSockets, AMQP, …etc.

Since that we are not focusing on the application itself, we will create one or more fully working web applications that help to clarify the use case. Often a standard e-commerce app will fit.

Development Tech Stack: ASP.NET CORE, Laravel, Angular, Vue Js, Android with java, and Flutter.