As per my project title Developing an Application on Software Architecture Patterns for Deploying and Managing Generative AI Models at Scale .

* We need to develop an application which we can deploy on software application or web – based application.
* So, the application we are developing contains all the data of the evaluations of all software architecture patterns in initial stages.
* We have to deploy the application on a software application or web – based application to check the generated data is same as the data from initial stages or higher than that.
* If it is higher, we have to compare the findings data and initial data to show that this particular architectural pattern works best for this on software application or web – based application

Proposal as written in DPP

* A fully functional proof-of-concept application demonstrating scalable AI deployment.
* Performance evaluation results comparing different architectural models.
* Best practices and recommendations for scalable AI model deployment.

Proposal as written in IPR

* Identification and Evaluation of Software Architecture Patterns
* Development of a Scalable Prototype
* Addressing Challenges in AI Model Deployment

So now this is my point of view

Step 1 – find a data where it contains all the evaluations of different software architectural patterns which are used universal for all the application developers to measure or scale the application.

Step 2 – create a prototype which we can run on a test application to generate data of the measure or scale of the test application.

Step 3 – now our gen AI model will run the same tests on the text model or the generated measured data and compare the data from the findings of step 1 and suggest the best software architecture.

Step 4 – these all should be in an application or a web based environment(GUI)

New updated Clarify

1. Scalability means the system’s ability to grow and maintain performance when faced with higher usage or larger amounts of data. So, we have to Make a code to check scalability of an application.

Metrics to measure or track :- Latency, Throughput, CPU & Memory Usage, Disk I/O & Network Bandwidth, Error Rate, Response Time, Concurrency & Load Handling

We can use tools like JMeter, Grafana, Prometheus, etc. We will get some metrics for the application

2. Now we will make our Gen AI app (main project Application ) in this we will show that different architectural patterns have different metrics. We will run this to get data of metrics, evaluation results of different patterns and we will show that is architectural pattern is best for this test application

Links [Kaggle](https://www.kaggle.com/datasets/qadeemkhan/dataset-of-software-architectural-styles), [Hugging Face](https://huggingface.co/datasets/epinnock/software-architecture-instructions-preference), [Google Datasets](https://datasetsearch.research.google.com/search?src=0&query=software%20architecture&docid=L2cvMTF4MTJ3emNiMg%3D%3D), [Scale a Web](https://bytebytego.com/courses/system-design-interview/scale-from-zero-to-millions-of-users)

we will get this report data from our Gen AI application in actual numbers

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Architecture Pattern** | **Throughput** | **Latency** | **Availability** | **Resource Utilization** | **Fault Tolerance** | **Elasticity** | **Cost Efficiency** | **Data Consistency** |
| **Monolithic Architecture** |  |  |  |  |  |  |  |  |
| **Microservices Architecture** |  |  |  |  |  |  |  |  |
| **Serverless Architecture** |  |  |  |  |  |  |  |  |
| **Event-Driven Architecture** |  |  |  |  |  |  |  |  |
| **Peer-to-Peer Architecture** |  |  |  |  |  |  |  |  |
| **Service-Oriented Architecture (SOA)** |  |  |  |  |  |  |  |  |

3. Now we will say that after comparison of patterns from above data THIS pattern is best practice for scaling an application

This is the final output example before scaling is step 1 and after scaling are the results of THIS pattern from step 3

|  |  |  |
| --- | --- | --- |
| **Metric** | **Before Scaling** | **After Scaling** |
| **Latency** | 1.5 sec | 0.4 sec |
| **Throughput** | 200 RPS | 2000 RPS |
| **CPU Utilization** | 90% | 50% |
| **Memory Usage** | 12 GB | 8 GB |
| **Error Rate** | 10% | 0.5% |

1. The application doesn’t choose the Right pattern just gives the analysis scale of the application
2. These are still pending - Event driven Architecture, Peer-to-Peer Architecture, Service-Oriented Architecture
3. The generated metrics are genuine or some random generated data.
4. Full report shows the data of AI, or any other Datasets provided
5. The AI used is on what bases or which AI’s API is being used, and quantity of plagiarism is there.
6. On what the prediction is based on like on which environment is the scalability prediction available?
7. On which platform did you run these tests to get the before metrics and after metrics.
8. Explain each and every metric used to get data from the application.
9. Where is the GEN AI that you described to do in the project.
10. Using Grok’s AI what are the ethical implications.
11. Create a custom test plan to compare all the 6 patterns in one click the result should just have metrics comparison as the table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Architecture Pattern** | **Throughput** | **Latency** | **Availability** | **Resource Utilization** | **Fault Tolerance** | **Elasticity** | **Cost Efficiency** | **Data Consistency** |
| **Monolithic Architecture** |  |  |  |  |  |  |  |  |
| **Microservices Architecture** |  |  |  |  |  |  |  |  |
| **Serverless Architecture** |  |  |  |  |  |  |  |  |
| **Event-Driven Architecture** |  |  |  |  |  |  |  |  |
| **Peer-to-Peer Architecture** |  |  |  |  |  |  |  |  |
| **Service-Oriented Architecture (SOA)** |  |  |  |  |  |  |  |  |

1. Make the graphs for these Latency Distribution, System performance, Performance Overview, latency vs throughput which compares in one for all the patterns because the single pattern test does not show much data and a conclusion saying that this pattern is the best approach for this application.
2. For AI Analysis, how the data generated and from where the data is extracted, is there any datasets or anything else?
3. Dose your Gen AI application works on any application? For example, run on https://www.herts.ac.uk/
4. Cite the every resource you took the data and for your own data write in footnote and make appendix for each source took from internet, ai, article, code snippets, video, image etc everything.