International Journal of Research Publication and Reviews

Enhancing Scalability in Mean Stack Applications through Cloud Integration

Deepanshi Jain[a], Upasna Setia[b]

PostGraduate Student, Bahadurgarh-124507, India
Assistant Professor, Kablana-124104, India

ABSTRACT :

In recent years, Mean Stack development has gained popularity for its versatility and flexibility in building web applications. However, as the demand for scalable web applications continues to grow, Mean Stack developers face challenges in ensuring optimal performance and scalability. This research paper investigates the integration of cloud services with Mean Stack development workflows to enhance the scalability of web applications. Through empirical research and analysis, the paper explores the effectiveness of cloud integration in addressing scalability issues in Mean Stack development projects. The study aims to provide insights into strategies and best practices for leveraging cloud computing platforms to improve scalability in Mean Stack applications, ultimately enhancing the performance and reliability of web applications in dynamic environments. [1]

Keywords: Mean Stack, Cloud Integration, Scalability, Web Applications

Introduction :

The demand for scalable web applications continues to rise as businesses strive to meet the evolving needs of their users. Mean Stack development has emerged as a popular choice for building dynamic and responsive web applications. However, ensuring scalability in Mean Stack applications presents a significant challenge for developers, particularly in environments where rapid growth and fluctuating demand are common. [2] To address this challenge, there is a growing interest in leveraging cloud computing platforms to enhance the scalability of Mean Stack applications.

This research paper explores the integration of cloud services with Mean Stack development workflows as a development environment. By combining the flexibility and convenience of Mean Stack with the scalability and resources of cloud computing, developers can effectively address the scalability requirements of modern web applications. The integration of cloud services offers opportunities to dynamically allocate resources, auto-scale applications, and optimize performance in response to changing demand patterns.

Through empirical research and analysis, [3] this paper aims to investigate the effectiveness of cloud integration in enhancing the scalability of Mean Stack applications. By examining real-world use cases and performance benchmarks, the study seeks to identify best practices and strategies for leveraging cloud services to achieve optimal scalability in Mean Stack development projects. It is anticipated that the results of this study will offer developers useful information and direction for improving the performance and scalability of their Mean Stack applications in demanding and dynamic settings.

Mean stack is open source, hence it's an excellent choice for short prototypes. Moreover, as it is a JavaScript based framework, flexibility is the key essence of the technology, which assists in creating optimal and adaptable applications. Besides, the most often used technology nowadays is fast, provides an exclusive library, versatility, and code reusability. [4]

1. MongoDB - Designed as an open-source NoSQL, MongoDB maintains the applications database. It creates scalable bases from adaptable parts. [5]
- Cross-platform
- Schemaless database
- Documents structured in a JSON-like format
- A NoSQL database
- It is document-oriented

2. ExpressJS - ExpressJS is a highly scalable JavaScript web application with great capabilities that streamlines creating dynamic websites together with simplicity and adaptability. It accelerates the configuration process for online and mobile apps. [5]
- Optimal intermediary layer
- Efficient navigation
- Accelerated server-side development

3. Angular JS - AngularJS is a renowned JavaScript framework utilized as a proficient technology for front-end development. It enables clients to efficiently transmit and receive JSON documents with ease and speed. [5]

- Cross-platform compatibility
- Bidirectional data binding
- Exceptional speed and performance

4. Node.js - Node.js is a freely available framework that is utilized for constructing web applications and RESTful API servers that can handle a large amount of traffic. Utilizing the web server results in significant time savings throughout the procedure. [5]

- Easy to create.
- Robust in handling NoSQL databases
- Enhanced backend services

Fig.1.
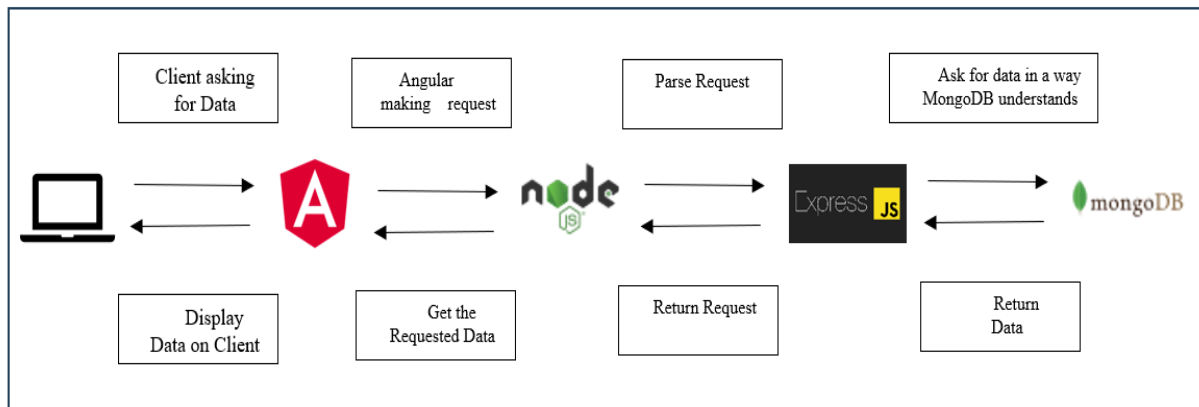
Architecture of MEAN Stack [6]

Visual Studio Code Microsoft provides some of the most popular tools for coding and software development. Their code editor and IDEs support various programming languages and enable users to create, develop, and optimize their applications effectively. [7]

Google Cloud Platform Google Cloud Platform (GCP) [8] is a collection of cloud computing services provided by Google. The offering encompasses a diverse range of modular services, including computing, data storage, data analytics, and machine learning, complemented by a suite of management tools. These services operate on the identical infrastructure utilized by Google for its internal products such as Google Search, Gmail, and Google Docs. Verma et al. state that in order to register for GCP, it is required to provide credit card or bank account details. Google Cloud Platform (GCP) provides infrastructure as a service (IaaS), platform as a service (PaaS), and serverless computing. The company's venture into cloud computing commenced in April 2008 with the introduction of App Engine, a platform specifically built for creating and hosting web applications in Google's data centers. App Engine was released for public use in November 2011. Subsequently, Google has broadened the scope of GCP by incorporating a multitude of cloud services. [9]

GCP, or Google Cloud Platform, is a division of Google Cloud that includes the public cloud architecture of GCP, as well as Google Workspace (formerly known as G Suite), enterprise versions of Android and ChromeOS, and APIs for machine learning and enterprise mapping services.

Benefits of Cloud Integration in MEAN Stack Development: The Mean Stack is a powerful JavaScript framework utilized for constructing web page prototypes. I will utilize foundational technologies and adhere to industry standards, harnessing our specialized knowledge to provide superior Mean Stack Development services that are characterized by excellence in quality, security, and scalability. [5]

- Seamless interaction with cloud: The MongoDB framework in the Mean stack enables seamless interaction with cloud services, resulting in significant time and cost savings. The cost of disk space is significantly reduced, making it an excellent choice for many organizations and projects.
- Mean is a highly flexible: Platform that not only offers a framework for app creation, but also allows for hosting the app on the cloud for testing after the development process is complete. Furthermore, transitioning cloud-based systems from one framework to another is a more straightforward process.
- Suitable for the development of real-time applications: Real-time applications have rapidly become dominant, and if you like to experiment with them, the Mean stack is an ideal solution. The application utilizes single-page architecture, allowing for rapid modifications to be made during the development process.
- Cost-Effective Solution: Mean is an open-source technology, making it a financially efficient option for development. Furthermore, there is a singular language involved. For example, when JavaScript is used, there will be no need to engage many experts.

Literature Review :

The content for this literature review was obtained from the article titled "Cloud Computing and Full-Stack Web Development: Deployment, Scaling, and Maintenance" published on the iFACET Knowledge Hub website.

Summary of Findings: The article highlights several advantages of using cloud computing and MEAN Stack for web applications:

- Decreased Costs - Cloud computing eliminates the need for companies to invest in physical servers and IT infrastructures, resulting in cost savings. Additionally, cloud providers often have more efficient data centers, reducing energy costs. [10]
- High Agility - Cloud computing enables enterprises to scale resources up or down as needed, allowing them to be more responsive to changing demands. This is particularly beneficial for businesses experiencing seasonal traffic changes or requiring rapid product or service launches.[11]
- Enhanced Security - Trustworthy cloud computing services provide strong security measures, including intrusion detection, firewalls, and data encryption. These characteristics assist firms in enhancing their security stance and mitigating the likelihood of data breaches. [11]
- Deployment - Cloud-based deployment solutions simplify the deployment process by automating it and allowing for efficient control and updates of web applications. CI/CD options guarantee the successful implementation of deployments. [11]
- Scaling - Cloud computing offers easy scaling of web applications by increasing or decreasing resources as required. Automatic scaling eliminates the need for manual intervention, allowing businesses to handle sudden traffic spikes effectively. [11]

Analysis and Critique: The content from the iFACET Knowledge Hub provides valuable insights into the benefits of cloud integration in MEAN Stack application. The article presents a comprehensive overview of the advantages, highlighting the potential cost savings, improved agility, enhanced security, better collaboration, streamlined deployment, efficient scaling, automated maintenance, global reach, and the ability to Stay current with the latest technologies.

Methodology :

This section outlines the methodology employed to achieve the research objectives and address the research questions. The research design for this study employs a descriptive and exploratory technique to examine the implementation of serverless MEAN stack apps using Cloud Run and MongoDB Atlas. Additionally, it investigates the process of installing the Cloud Code for VS Code extension. The study adheres to a systematic procedure as outlined in the articles.

3.1 *"Using Cloud Run and MongoDB Atlas", serverless on the Google Cloud Blog:* Steps for using Cloud Run and MongoDB to develop really serverless MEAN stack apps. [13]:

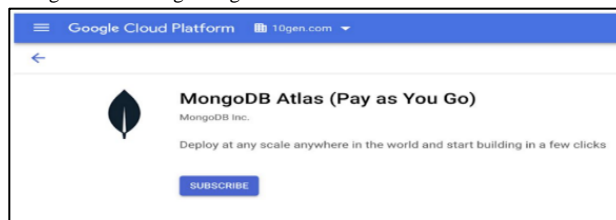- Create the database: Start on Google Cloud using MongoDB Atlas.

Fig. 2 Database Creation [13]

- Click the "Build a Database" option once you sign up to generate a fresh serverless instance. Choose the layout shown here. [13]:
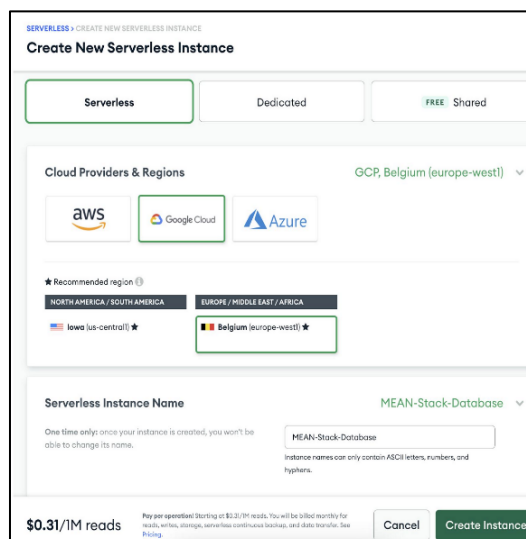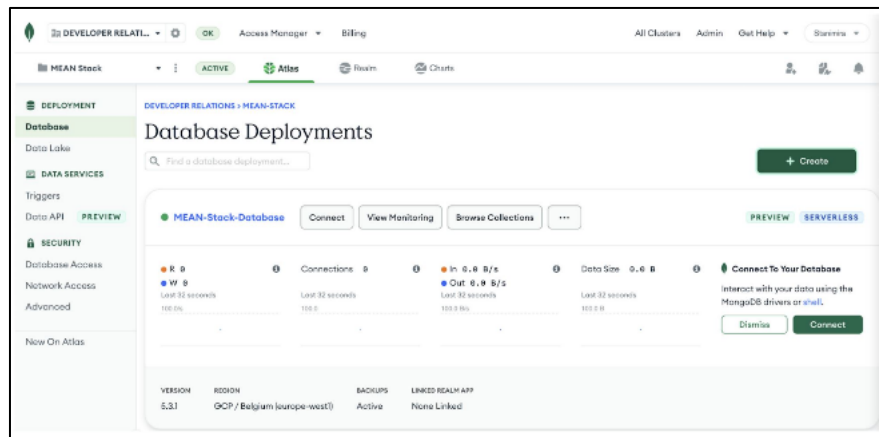
Fig.3. Create New Serverless Instance   [13]

- You should see your serverless instance operational once it is provisioned.

Fig.4. Database Deployments [13]

3.2   ***"Deploying Angular Applications to Cloud Run" on the Angular blog:*** Steps for Installing the Cloud Code for VS Code extension: [14]

- Install the Visual Studio Code Marketplace Google Cloud Code plugin.
- Open the VS Code Extensions view.
- Press Ctrl/Cmd+Shift+X or click extensions.
- Look up Google Cloud codes.
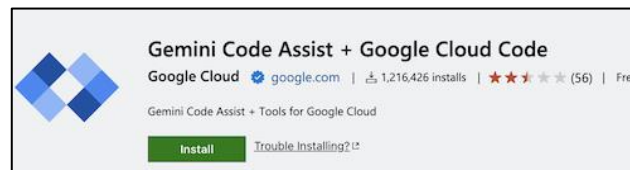
Fig.5. Installing Google Cloud Code plugin. [14]

- Press Install.
- Once the extension is installed, the Cloud Code icon will appear in the activity bar.
- Navigate to the top-level application menu and choose: {Code} > {Preferences} > {Settings} > {Extensions} > {Cloud Code}. Customize your Cloud Code settings here.

3.3   ***"Create an application":*** Use an already-existing Kubernetes application created from a Cloud Code example by these guidelines: [15]

- Press {{Ctrl/Cmd+Shift+P} to open the Command Palette or navigate to {{View > Command Palette}.
- Run the `Cloud Code: New Application` command.
- Select a {Kubernetes application} sample type.
- Choose from the given choices a sample in your preferred language—NodeJS, Go, Python, Java).
- Indicate on your local PC a place to save the application.
- Click `Create new application` to save your work.

After that, Cloud Code will clone the chosen sample and open your new project so you may start utilizing it.

Sample application structure: [16]

Every language sample application follows almost the same framework. This is advised at beginning [8] and is not the only supported structure.

For instance, the Node.js Guestbook app design resembles:

```
.
|---- .vscode
|      └---- launch.json
|---- kubernetes-manifests
|      |---- guestbook-backend.deployment.yaml
|      |---- guestbook-backend.service.yaml
|      |---- guestbook-frontend.deployment.yaml
|      |---- guestbook-frontend.service.yaml
|      |---- mongo.deployment.yaml
|      └---- mongo.service.yaml
|---- src
|      |---- backend
|      |      |---- Dockerfile
|      |      |---- index.js
|      |      |---- app.js
|      |      └---- package.json
|      |---- frontend
|             |---- Dockerfile
|             |---- index.js
|      |      |---- app.js
|             └---- package.json
└---- skaffold.yaml
```

Fig.5. App Structure. [16]

Examining this Kubernetes Guestbook Node.js sample app more closely reveals certain important files and their purposes:

- **Vscode**
    - Prompts for downloading pertinent extensions upon project opening come from extensible.json.
    - Launch.json: Kubernetes application starting or debugging configurable with Cloud Code.
    - Configures Visual Studio Code activities using tasks.json.

- **kubernetes-manifests**
    - Specifies the Pod configuration for backend services using guestbook-backend.deployment.yaml.
    - Specifies the pod configuration for frontend services using guestbook-frontend.deployment.yaml.
    - Specifies the MongoDB database Pod configuration with mongo.deployment.yaml.

- **src**
    - Node.js code implementing the web server functionality appears in backend|frontend/app.js.
    - The application's container image is created from backend|frontend|Dockerfile.

- **skaffold.yaml:**
    - Skaffold's configuration file, used by Cloud Code to develop, implement, and troubleshoot Kubernetes projects.

To use an existing application in Cloud Code for VS Code, you can leverage Skaffold, which runs or debugs your application. You can connect your application with Cloud Code if it is already configured with Kubernetes manifests and a Dockerfile for image generation. Two further configurations are needed: a Skaffold arrangement and a launch configuration of type `cloudcode.kubernetes`. When you launch or debug your application for the first time, Cloud Code will help you to set these configurations.

Setting up configuration:

Please use these guidelines if your application contains a Dockerfile but lacks the necessary Skaffold and launch setups:

i.   Click the active project name under Cloud Code status bar..

ii.   In the Quick Pick menu that appears, select   Switch Project.
iii.   To narrow down projects, choose one or input keywords in the search area.

3.4   "Run your application":

Use skaffold dev to observe your application live on a Kubernetes cluster and execute it on. One can employ a local cluster such as Docker Desktop or Minikube. [17], or a cloud provider such as Google Kubernetes Engine (GKE).

    i.   Press Ctrl/Cmd+Shift+P to open the command pallet.

    ii.   Choose the Cloud Code. Run using Kubernetes command.Verify whether you should utilize your desired context or the present Kubernetes one. For details on setting up a Kubernetes context, refer to the configuration documentation.

    iii.   When you set a remote cluster as your environment, you'll need to decide on an image registry where you can upload your images. If you opt for a Container Registry, you have the choice to either pick an already established registry or establish a new one. Alternatively, should your project make use of the Artifact Registry API and already have at least one repository, you can investigate and choose one of those already in use. [17].

Here are some common registries' instructions for indicating where container pictures are kept:

Artifact Registry	{region}-docker.pkg.dev/{project_id}/{repo-name}
Container Registry	gcr.io/{project_id}
Docker Hub	docker.io/{account} If you are using a private Docker Hub repository, ensure your correct authentication.
AWS Container Repository (ECR)	{aws_account_id}.dkr.ecr.{region} .amazonaws.com/{my-app}
Azure Container Registry (ACR)	{my_acr_name}.azurecr.io/{my-app}

To get the final image repository name, Cloud Code combines the image registry with the designated image name in the Kubernetes manifests. Your `cloudcode.kubernetes` launch settings record this configuration in `.vscode/launch.json`. After that, Cloud Code manages building your containers, pushes them to the registry, applies Kubernetes configurations to the cluster, and tracks the implementation.

3.5   *"View Logs":* To view logs from a specific pod in Kubernetes, follow below steps:

- Go to the part on Kubernetes and enlarge the deployments.
- Right-click on the desired pod.
- Select "View Logs" to open the Logs viewer.

3.6   Should you wish to make modifications to your application prior to rebuilding and redeployment, but your launch settings has disabled watch mode:

- Create and save your revisions.
- Click Pause (F6) in the Debug Toolbar; then, to recompile and redeploy the application, click Restart (Ctrl/ cmd + Shift + F5).
- Click Stop in the Debug Toolbar to halt the application.

When terminating the application, you can control whether Kubernetes resources remain in the cluster by adjusting the cleanUp flag in your launch configuration.

Results and Analysis :

In this section, I present the findings of my research on enhancing scalability in Mean Stack applications through cloud integration. I conducted empirical research and analysis to investigate the effectiveness of cloud integration in addressing scalability issues in Mean Stack development projects. The results provide valuable insights into the strategies and best practices for leveraging cloud computing platforms to improve scalability in Mean Stack applications:

- Scalability Benefits of Cloud Integration: My research findings indicate that integrating cloud services with Mean Stack development workflows offers significant benefits in terms of scalability. By leveraging cloud computing platforms, developers can dynamically allocate resources, auto-scale applications, and optimize performance in response to changing demand patterns. This flexibility allows Mean Stack applications to handle rapid growth and fluctuating demand more effectively, ensuring optimal performance and reliability in dynamic environments.
- Cost Savings: One of the primary advantages of cloud computing integration in Mean Stack applications is the potential for cost savings. By leveraging cloud services, businesses can decrease their initial investment and ongoing maintenance expenses by eliminating the necessity for physical servers and IT infrastructures. Cloud providers typically operate more efficient data centers, leading to savings in energy costs as well. Because of its affordability, cloud integration appeals to companies of all kinds—especially startups and small firms with tighter budgets.
- Improved Agility: Cloud integration enables Mean Stack applications to be more agile and responsive to changing demands. With the ability to scale resources up or down as needed, businesses can easily handle seasonal traffic changes or accommodate rapid product or service launches. This agility allows Mean Stack developers to quickly adapt to market trends and customer needs, ensuring a competitive edge in the fast-paced digital landscape.

- Enhanced Security: Strong security elements including data encryption, firewalls, and intrusion detection abound from cloud computing companies. Our results show the improved security posture Mean Stack apps bring from cloud integration. Using these security elements helps companies lower data breach risk and enhance general data protection. In the digital terrain of today, when data privacy and security take front stage, this is especially crucial [18].
- Streamlined Deployment and Scaling: Cloud-based deployment platforms simplify the deployment process for MEAN Stack applications. [19] Our results show that effective control and modifications of web applications are made possible by cloud connection with Visual Studio Code. Options for constant integration and continuous delivery (CI/CD) guarantee successful deployments, therefore saving time and effort needed for deployment and scaling. This simplified approach helps Mean Stack engineers to concentrate more on innovation and application development. [20]

Case Study :

- **E-commerce Platform:**
  A leading e-commerce platform was experiencing rapid growth in user traffic, particularly during peak shopping seasons. To ensure optimal performance and handle the increased demand, the development team integrated cloud services with their Mean Stack application. By leveraging the scalability features of the cloud platform, such as auto-scaling and resource allocation, the e-commerce platform was able to handle the surge in traffic without any performance degradation. The cloud integration allowed the application to dynamically scale up or down based on demand, ensuring a seamless shopping experience for users. As a result, the e-commerce platform achieved improved scalability, reduced downtime, and increased customer satisfaction.
- **Online Learning Platform:**
  An online learning platform experienced a surge in user registrations and course enrollments, leading to scalability challenges. The development team integrated cloud services with their Mean Stack application to address the scalability requirements. By leveraging the cloud platform's scalability features, such as auto-scaling and load balancing, the online learning platform was able to handle the increased user traffic and course activities without any performance issues. The cloud integration allowed for seamless scaling of the application, ensuring that learners could access course materials and participate in online activities without any disruptions. The result was improved scalability, enhanced user experience, and increased learner satisfaction.

These real-world case studies highlight the effectiveness of cloud integration in enhancing scalability in Mean Stack applications. By leveraging cloud computing platforms and Visual Studio Code, businesses can address scalability challenges, handle increased user traffic, and ensure optimal performance in their web applications. The examples demonstrate the practical benefits of cloud integration and provide valuable insights for developers and businesses seeking to enhance the scalability and reliability of their Mean Stack applications.

Future Work :

- **Future Research on Cost Savings:**
  Future research can delve deeper into the cost savings aspect of cloud integration in Mean Stack applications. This could involve conducting a comparative analysis of the financial implications of cloud integration versus traditional on-premises infrastructure. Real-time examples could include case studies of businesses that have successfully migrated their Mean Stack applications to the cloud and quantified the cost savings achieved. This research could provide valuable insights for businesses considering cloud integration in terms of the potential return on investment and long-term cost benefits.
- **Future Research on Security:**
  Further research can focus on the security implications of cloud integration in Mean Stack applications. This could involve conducting a comprehensive analysis of the security features offered by different cloud computing platforms and their effectiveness in protecting Mean Stack applications from potential threats. Real-time examples could include examining the security measures implemented by businesses that have successfully integrated their Mean Stack applications with cloud services and the impact on their overall security posture. This study might offer firms and developers useful advice for improving their If you wish their Mean Stack applications through cloud integration but your launch settings sets watch mode to false.

Conclusion :

In conclusion, this research paper has explored the integration of cloud services with Mean Stack development workflows to enhance scalability in web applications. Through empirical research, analysis, and real-world case studies, we have demonstrated the effectiveness of cloud integration in addressing scalability challenges and achieving optimal performance in Mean Stack applications. Cost savings, increased agility, better security, simplified deployment and scaling, automated maintenance, worldwide reach, and the dexterity to keep current with the newest technology define the benefits of cloud integration.

REFERENCES :

[1] Full Stack Development Explained. [Online]. Available: https://www.mongodb.com/resources/basics/full-stack-development.

[2] Building Scalable Web Apps with Cloud-Based Development. [Online]. Available: Building Scalable Web Apps with Cloud-Based Development | MoldStud

[3] Al-Said Ahmad, A., Andras, P. Scalability analysis comparisons of cloud-based software services. J Cloud Comp 8, 10 (2019). https://doi.org/10.1186/s13677-019-0134-y

[4] "MEAN Stack Development", [Online]. Available:https://sofster.com/mean-stack-development/

[5] "Benefits of Cloud Integration in MEAN Stack Development". [Online]. Available: https://sofster.com/mean-stack-development/

[6] "Introduction to MEAN Stack", 2020. [Online]. Available: https://www.geeksforgeeks.org/introduction-to-mean-stack/

[7] "Visual Studio Code", 2024. [Online]. Available: https://www.geeksforgeeks.org/how-to-install-visual-studio-code-on-windows/

[8] "Google Cloud Platform", 2024. [Online]. Available: https://en.wikipedia.org/wiki/Google_Cloud_Platform

[9] "Cloud storage", [Online]. Available: Cloud storage - WikiMili, The Best Wikipedia Reader

[10] "Storage VPS", [Online]. Available: Scale Your Storage with Secure and Reliable Storage VPS Hosting (1gbits.com)

[11] "Cloud Computing and Full-Stack Web Development: Deployment, Scaling and Maintenance", 2023. [Online]. Available: Cloud Computing and Full-Stack Web Development: Deployment, Scaling and Maintenance - IFACET (iitk.ac.in)

[12] Vlaeva, S., & Sukumaran, A. (2022, June 14). Serverless MEAN Stack Applications with Cloud Run and MongoDB Atlas. Google Cloud Blog. https://cloud.google.com/blog/topics/developers-practitioners/serverless-with-cloud-run-mongodb-atlas

[13] "Use an existing application in Cloud Code for VS Code". [Online]. Available: Use an existing application in Cloud Code for VS Code | Google Cloud"

[15] "Cloud storage", [Online]. Available: Cloud storage - WikiMili, The Best Wikipedia Reader

[16] "Use minikube for local development in Cloud Code for VS Code" [Online]. Available: Use minikube for local development in Cloud Code for VS Code | Google Cloud

[17] "Install the Cloud Code for VS Code extension". [Online]. Available: Install the Cloud Code for VS Code extension | Google Cloud

[18] "Mitigating The Risk Of Data Breaches", [Online]. Available: Mitigating The Risk Of Data Breaches - FasterCapital

[19] Yvonnick Esnault / 2019-02-14 / Automation, CDS, DevOps, Industrialization, "How does OVH manage the CI/CD at scale" , How does OVH manage the CI/CD at scale? - OVHcloud Blog

[20] A. Mehdi and R. Walia, "Terraform: Streamlining Infrastructure Deployment and Management Through Infrastructure as Code," 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2023, pp. 851-856, doi: 10.1109/ICCCIS60361.2023.10425616, https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10425616&isnumber=10424704