iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VI **Month of publication:** June 2022

DOI: https://doi.org/10.22214/ijraset.2022.44522

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

Certification and Attestation Genuinity Management System

Hari Krishnan D R[1], Dr. Tamilarasi K[2]
[1]*Student,* [2]*Professor, Department of Computer Science and Engineering,* Jeppiaar Institute of Technology

Abstract: CAGMS (Certificate and Attestation Management system) is an easy-to-use management system where any organization or individual can issue or receive digitally verifiable certificate for any type of documents / awards / rewards etc. with ease of complexity of system and technology. The system uses modern techs and high-interactive API which makes fit for all sort of organization.
Keywords: Digital Certificates, Serverless, GraphQL, AWS, Micro services, Certificate authority

I. INTRODUCTION

Certification and attestation have become a vital in modern era since many organizations use authorization in certificate and attestation to valuate a individual /organization identity, Long back organization used manually sealed and signed/facsimiled physical certificate /attestation which made it more vulnerable to fraudulent and forgery. On progress of modern IT growth, the certification got a new shade of digitalization, Digital certificates are created by using high-tech cryptographic algorithm and modern tech such aa blockchains which makes it more secure, provides end-to-end verification, provides unique identity etc. Along with all these features the capital of system, maintenance cost, human resource, orchestration, upfront also increases, thereby Cryptographic blockchain based certification system becomes less adaptable and accessible by small-cap organizations. CAGMS system focus mainly on providing high reliable, easily adaptable, feasible certificate management system which reduces capital.
Also project focus on delivering highly feasible, scalable, intergratable, and reliable api and SaaS based model to meet the current organizational business needs.

II. DESIRED FRAMWORKS

A. Serverless / Micro-services

In serverless architecture the entire project is broken down into several component /features which is linked by leveraging the cloud-services provided and maintained by PaaS providers such as gcp, aws or azure. Serverless architecture form a basis of layered structure where services are inter-connected through api. This type of architectures makes the project more agile with less upfront and maintenance cost, improved debugging ability and accessibly, reliable infrastructure, easy updates and rollout releases, improved monitoring and orchestration, reduced maintained and improved productivity etc., Since the services are provided by PaaS, micro-service are termed as serverless in cloud computing.

B. Cloud Infrastrcture

Indeed, using in-house resource for maintaining and developing a infrastructure, CAGMS uses cloud AWS infrastructure, AWS maintains and secures the infra with specialized monitoring and orchestration tools.

C. AWS – Amazon web services

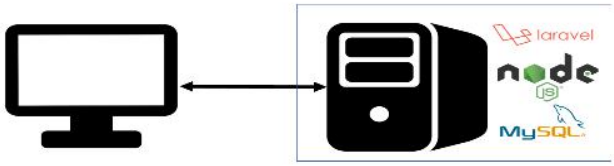
Without having to manage servers, AWS provides technology for running code, maintaining data, and integrating apps. To boost agility and save costs, serverless solutions include automatic scaling, built-in high availability, and a pay-per-use invoicing mechanism. These technologies also take care of infrastructure management responsibilities like capacity provisioning and patching, allowing to concentrate on building code.An event-driven compute solution natively linked with over 200 AWS services and software as a service (SaaS) applications, is the foundation for serverless applications.

III. MONOLITHICS VS SERVERLESS/MICRO SERVICES

Choice of cloud based serverless/micro-services architecture form the basis for CAGMS system since the project majorly focus on reduced cost, feasibility and reliability, hence choosing between monolithic and micro-service becomes vital.

A. Monolithic arthitecture

Monolith architecture is a traditional development framework where entire project is full-stacked and composed in a singular entity, i.e. for example considering a backend, all the endpoint, api's , features , modules will be in a single entity/server .



B. Merits of Monolithic arthitecture

1) Traditional development frameworks.
2) High availability of resources and tools .
3) Trustable since being used a long-back.
4) Reduced chunks of repository.

C. Demerits of Monolithic arthitecture

1) High upfront and production cost.
2) Less reliable and available.
3) Maintenance and monitoring is hectic.

D. Serverless / micro-services

In serverless architecture the entire project is broken to several components, i.e. for example considering a backend, all the endpoint, api's , features , modules is broken into several component and connected through api,

E. Merits of Serverless / micro-services

1) Easily maintenance and monitoring.
2) High reliable, available and reduced cost.
3) Easily scalable and highly feasible.

F. Demerits of Serverless / micro-services

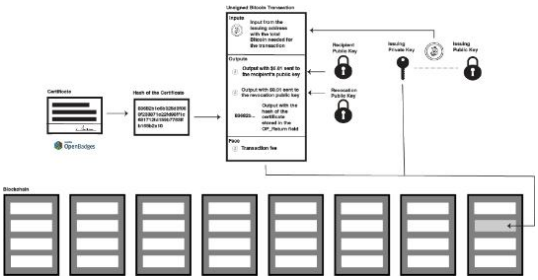
1) Less trust since very new technology.
2) Less developer and tool availability.

IV. CURRENT SYSTEM VS CAGMS

CAGMS differs with current digital certificate management system in not only architecture or framework also shows vast difference in algorithms, workflow function and business logics. And also current systems does not facilitate attestation management which is more crucial in many organization like colleges, govt official etc., CAGMS provides attestation management along certification system.

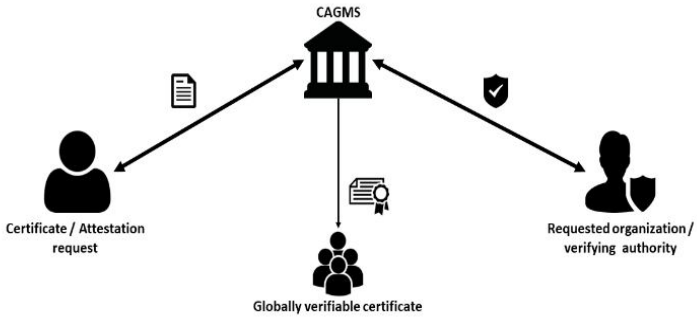
A. Blockchain -based certification system

The current system uses the advantage of web 3 technologies such as blockchain, Ethereum, Hyperledger which uses high reliable cryptographic algorithm techniques to sign a certificate and maintain the security and authenticity of the certificate.



B. CAGMS

CAGMS ease the use of cryptographic algorithm and use of advanced tools such as Ethereum, Hyperledger etc. to hold the system.
This system has its own CA(Certificate Authority) to management infrastructure and maintain the certificate Genuity ,authenticity , repository details.



CAGMS is facilitate easy to access environment where organization can generate their certificate/ issue certificate with ease. Rather than complex algorithm certificate can be easily generated in 2 step interactive process.
The email Id along with meta data serves the identification of a certificate or attestation.

V. TOOLS AND SERVICES IN CAGMS

A. Serverless Framwork

Serverless Framework is a templating tool used to template the entire cloud infrastructure in json format, which facilitate easy production update rollouts, easy debug of infra, update services faster, maintain infrastructure as a code etc. Since micro -service / serverless architecture is used Serverless Framework is useful in leveraging all the services easily.

B. GraphQL

GraphQL api is used instead of REST since it forms essential for serverless architecture.
GraphQL forms layer of interaction between client and server (other aws services) and rather than REST it has lot features which improves reliability and productivity.

C. AWS Lambda

Aws lambada function are the core functionality service which hold the core functional and business logics ie lambda function are those which process the request (crud operation, workflow logic etc).
Lambda function can be associated with any aws services and accessed using any backend language (Python, nodejs, ruby, java etc).

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

D. AWS Appsync

Usage of GraphQL in aws is bit complex directly hence appsync is a service in aws which facilitate the api for GraphQL.

E. AWS Cognito

In monolith architecture the account details/credential data are fully established and maintained in in-house, which is typically a critical module which is more vulnerable. Aws provides fully maintained api called Cognito where user credential such as name, password and other metadata acan be stored. Also aws maintains the issue, expire and rotation of JWT/JWK tokens.
CAGMS uses jwt token for all protected routes / features.

F. AWS DynamoDb

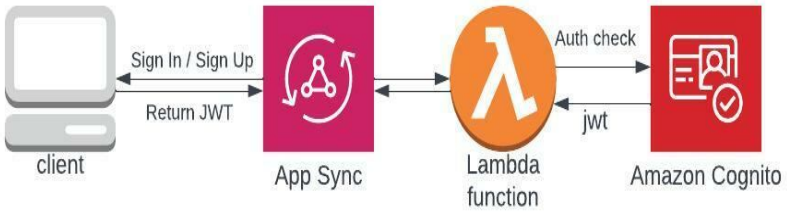
Dynamo db is a aws managed NoSql based database which has advantage high scalability and concurrency control.
Dynamo db is chosen rather than mangodb or firebase since leveraging within same provider service is easy and reliable.

G. AWS S3

S3 is a storage service where the certificate and attestation repos are stored. S3 is fully secured repo which is entirely maintained by aws and user policies.

VI. WORKING OF CAGMS – AUTH MODULE

Auth module (user account) in CAGMS is associated by leveraging layered micro-service aws appsync, lambda and Cognito.



A. Sign-Up

Create new account/signup is done using a graphQL mutation (graphQL can be integrated in react, angular or any frontend frameworks), where user prompt the basics details such as name, organization type, password and email, GraphQL forwards the request to lambda for further processing.

B. Sign-In

Sign-in / login is done using graphQL query where user pass email and password, lambda will prove the request for login and send response (jwt or error).

C. OTP-Verify

Once user request for sign-up, lambda check for concurrency and send the otp to the registered email, User can activate the account by inputting otp and email in OTP verify graphQL mutation.

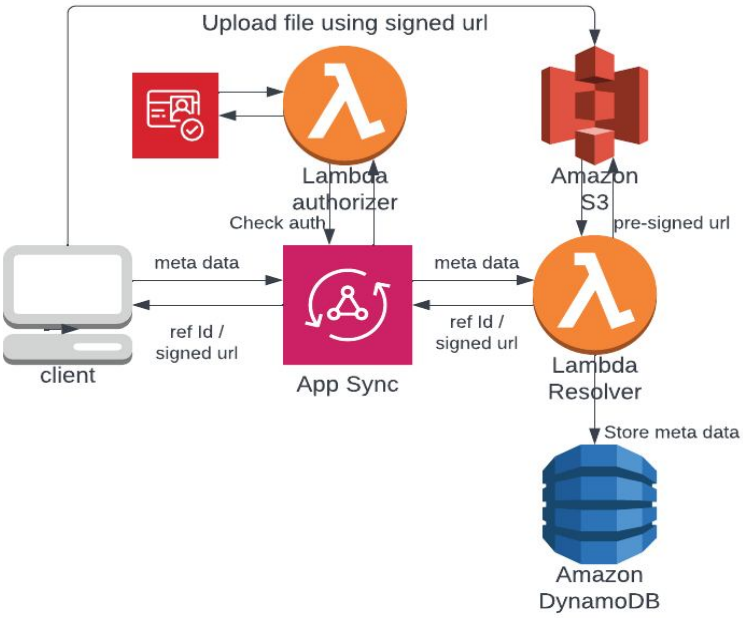
D. Dynamic inputs and error handlings

For future implementation CAGMS uses all generic input in order to add/make future updates, Also CAGMS handles all sorts of exceptions such as concurrent account registration, inactivated account exceptions and token timeouts (revoke a jwt token).

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

VII. WORKING OF CAGMS – CERTIFICTAE GENERATION (*WITH PDF*)

User can access the certificate issue function once the user logins. The certificate generation api works only with jwt provided. CAGMS provides two types of certificate generation i.e. With pdf and With Template. Certificate generation with pdf api facilitates user/ certificate issuer to provide/upload a pdf file of certificate and issue for a desired user(email) along with certificate metadata (name, subject etc.).



A. Create New Certificate

The user request new certificate with pdf using GraphQL mutation ,User prompts the certificates meta data such as respondent email and certificate mini details , after successful response user receives the uuid of the certificate and a pre-signed url(for file upload).

B. Lambda Authorizer

Authenticate the user identity by parsing the jwt token and claims the user identity from cognito.

C. Lambda Resolver

It's the core functionality which check and validate the request (auth details and certificate details) and store details in db and also request s3 for pre-signed URL(file upload)

D. S3 Storage

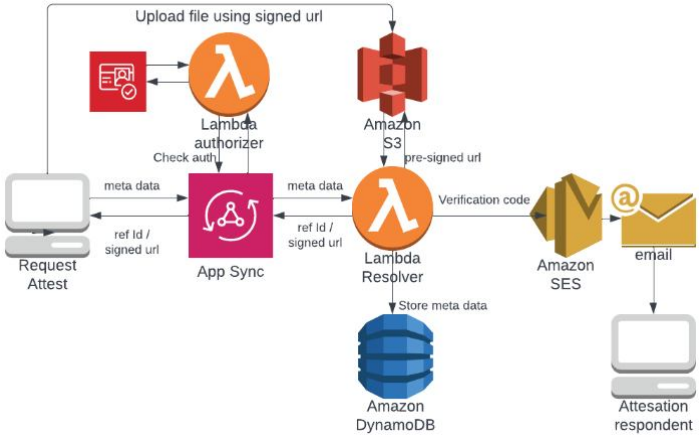
Uploading file directly in graphQL is not possible since graphQL have restriction of processing only json, Hence pre-signed technique is used. Pre-signed Url is a temporary url which is typically a REST api where user can perform PUT operation with file against the url and upload the file.

VIII.WORKING OF CAGMS – CERTIFICTAE GENERATION (*WITH TEMPLATE*)

When user is unaware / doesn't have the pdf/document to make digital certificate user can use this option.Certificate generation with template provides pre-defined instant templates where uses can prompt the certificate fillers such as type of certificate, body of certificate etc. along certificate metadata.This feature will be more useful for small cap organizations ,mini events and induvial user when design of certificate not available or when instant design required.In future more interactive feature such as bloat kit for certificate build can be added.Also, CAGMS is a api based system hence organization can customize their own features along the current availability.

IX. WORKING OF CAGMS – ATTESATION REQUEST

Along with Certificate management CAGMS also provides Attestation management system. Unlike certificates, attestation are usually done with third party respondent who holds powers to authenticate and attest a document for example government officials, Head of department, executives etc. Hence attestation module has two phases (Request for attestation and Authenticate document), Attesting a document is simple with 2 step interactive process in CAGMS.



A. *Create New Attest*

The user request new attesation using GraphQL mutation ,User prompts the certificates meta data such as attestation respondent email and attestation mini details , after successful response user receives the uuid of the attestation and a pre-signed url(for file upload).

B. *Lambda Authorizer*

Authenticate the user identity by parsing the jwt token and claims the user identity from cognito.

C. *Lambda Resolver*

It's the core functionality which check and validate the request (auth details and certificate details) and store details in db and also request s3 for pre-signed URL (file upload).

D. *Mailing Utility*

To maintain the integrity and Genuity of attestation document, Once all document all uploaded for attestation , CAGMS sends a verification code and verifying procedure to the attestation respondent where respondent need to accept and verify the attestation document.
Mail also includes the meta data for verification and validation purpose.
The status of the attestation (verified or not verified will change only on respondent action.

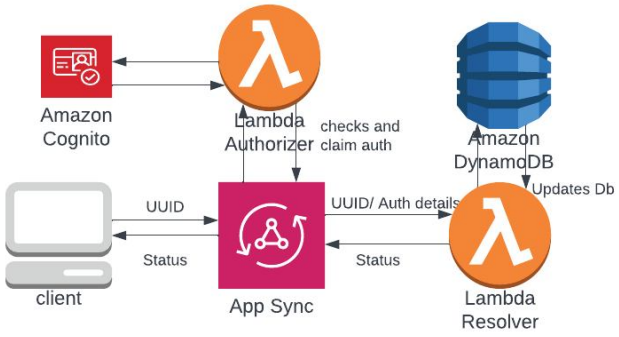
E. *S3 Storage*

Uploading file directly in graphQL is not possible since graphQL have restriction of processing only json, Hence pre-signed technique is used. Pre-signed Url is a temporary url which is typically a REST api where user can perform PUT operation with file against the url and upload the file.

X. WORKING OF CAGMS – ATTESATION VERIFICATION (WITH AUTH)

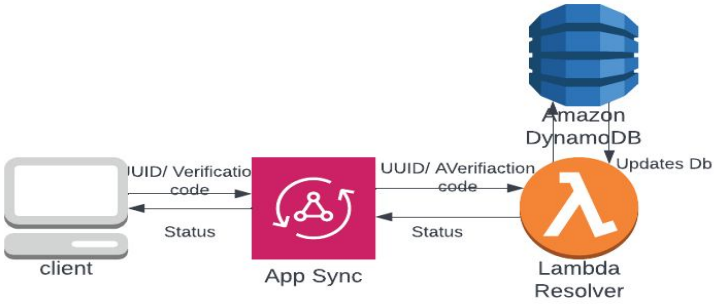
Once user request for a attestation, in the second phase the respondent need to verify the attestation to prove and authenticate the identity of document.
CAGMS provides two-way to verify a certificate (Using auth and using Code).

If the respondent of the attestation request also has a CAGMS account, they can verify the document within their dashboard with graphQL api without any verification code.
The attestation respondent user need to use verify with auth graphQL api by providing the uuid of the attestation document along api call. The lambda authorizer checks for claim from jwt token and verify the attestation user.
Lambda resolver validates the attestation metadata against attestation respondent claims (from lambda authorizer) ,verifies the attestation as and updates the database status for attestation document .



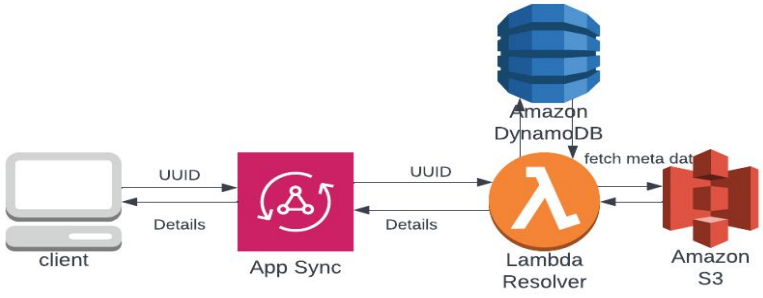
XI. WORKING OF CAGMS – ATTESATION VERIFICATION (WITH CODE)

If the user doesn't have / doesn't want to create account to attest a document, they can verify the attestation with verification code.
In this type of Verification, the respondent needs to prompt the verification code and uuid in a publicly accessible graphQL api.
The mail contains the verification code, uuid of the attestation document and api details.
In future a the api can be integrated with any frontend application to provide more interactivity.



XII. WORKING OF CAGMS – VERIFING CERTICATE AND ATTESATIONS

All Certificates and attestation documents can be verified publicly by anyone using the uuid provided while issuing a certificate or attestation.



Any user / even CAGMS non user can prompt the uuid in a graphQL api and get details of the certificate/attestation such as metadata, ownership details, respondent details, document file (link) etc.

A. Key takeaways in Cagms

1) CAGMS doesn't use cryptographic methodology to creating and maintaining certificate/attestation.
2) Uses email addresses as a core key for identities (users)
3) CAGMS uses open- to-world methodology where all meta data for certificate / attestation is accessible by anyone.
4) The open-to-word methodology is chosen since the project believes that certificate / attestation should be an access to all data.
5) Hence project majorly focus on Genuity of data(users) by end-end verifying the identity (email and code verification).
6) All the components of project are api and SaaS model based hence organization can add their own features along the prevailing.
7) The project adds more values if email with registered domain names used.
8) Attestation module is new initiative by CAMS which is mostly not reliably available in current system.
9) In future, advanced option such as KYC can be added to improve the reliability of data.

B. Merits in Cagms

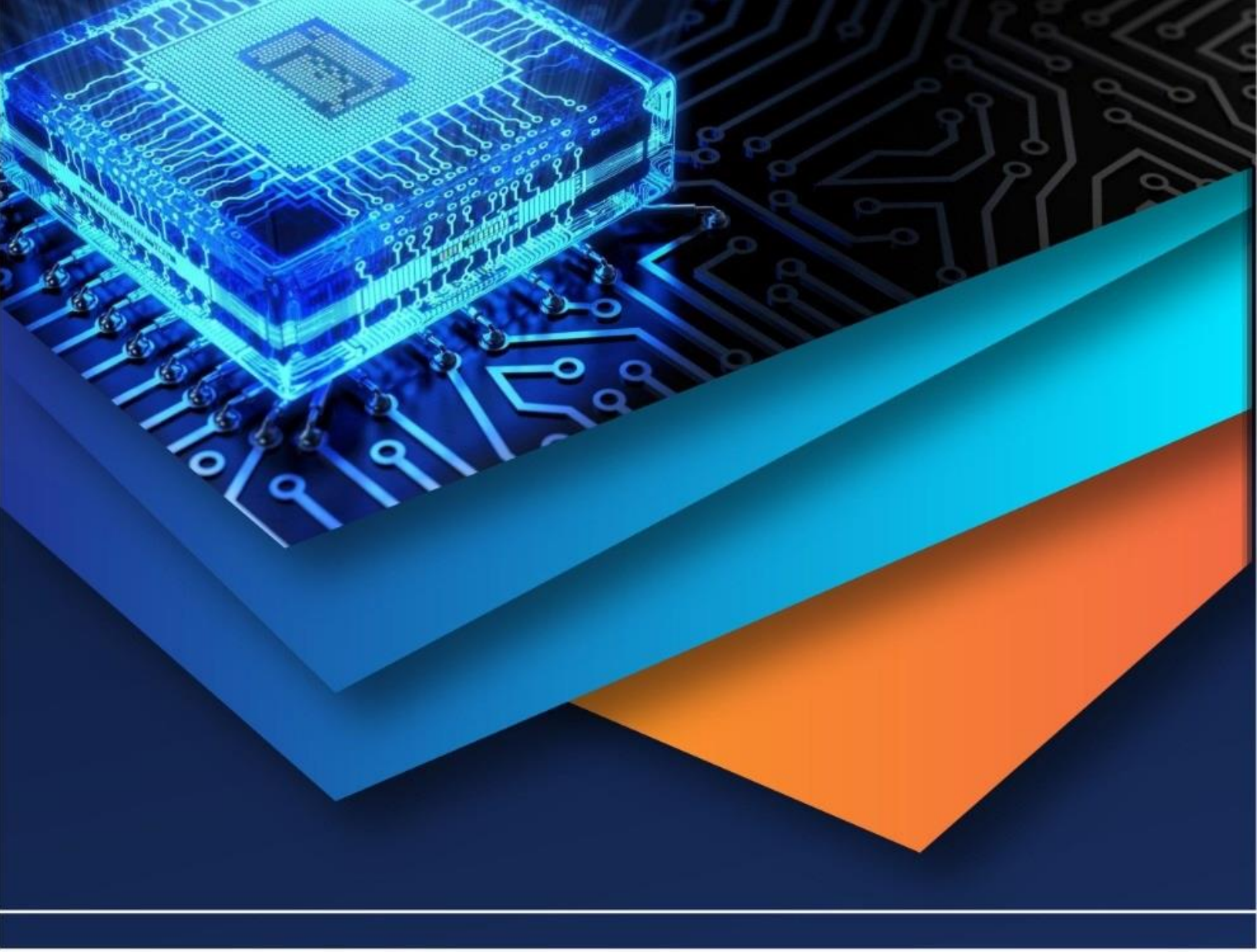
1) Highly available and reliable since services (db, backend etc) are maintained by PaaS.
2) Ease of maintenance and monitoring.
3) Any induvial user can access CAGMS.
4) Highly integrable api .
5) Developed with modern architectures.
6) Cheap since pay-as-go model.
7) Integrable by any organization since has SaaS model architecture.
8) Improved user experience
9) More Interactive, mostly automated workflows and less manual works.
10) Improved security.

C. Demerits in Cagms

1) Upfront cost is high a bit.
2) Less developer and resources(tools) available since modern and very new tech stacks.
3) Might be trust issue since very new tech stacks in market.

REFERENCES

[1] M. Abadi, A. Birrell, I. Mironov, T. Wobber and Y. Xie, "Global authentication in an untrustworthy world", Proc. 14th USENIX Conf. Hot Topics Operating Syst., 2013..
[2] S. Hendrickson, S. Sturdevant, T. Harter, V. Venkataramani, A.C. Arpaci-Dusseau and R.H. Arpaci-Dusseau, "Serverless computation with openlambda", Procedings of the 8th USENIX Conference on Hot Topics in Cloud Computing (Hot Cloud '16), pp. 7, June 2016.
[3] I. Baldini, P. Castro, K. Chang, P. Cheng, S. Fink, V. Ishakian, et al., Serverless Computing: Current Trends and Open Problems, pp. 20, June 2017.
[4] R. Hunt, "PKI and Digital Certification Infrastructure", Department of Computer Science, University of Canterbury, New Zealand, IEEE, 2001
[5] .[Serverless Framework] https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/what-is-sam.html.
[6] [Blockchain CA] https://medium.com/mit-media-lab/what-we-learned-from-designing-an-academic-certificates-system-on-the-blockchain-34ba5874f196



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY