BUAN 6335.502

ORGANIZING FOR BUSINESS ANALYTICS PLATFORMS

GROUP PROJECT REPORT

GROUP 6

ANANYA CANAKAPALLI - AXC230004

MANISHA VARMA - MXV220022

ROHITH ANIL NAIR - RXN220027

SAI KALYAN MAMILLAPALLI - SXM220112

SAURABH DAS NAIR - SXN220009

YASHASWINI BORE GOWDA - YXB220019

PROJECT TEAM ROLES & RESPONSIBILITIES

Team Lead: Ananya

Project Management: Develop and execute project plan, manage budget, track progress, identify and mitigate risks.

Communication: Facilitate communication between team members and stakeholders, provide updates on progress.

Coordination: Ensure everyone is working towards common goals, resolve conflicts, maintain team morale.

Oversight: Monitor overall project performance, identify and address any issues.

Data Architect: Saurabh Nair

Data Platform Architecture: Design and implement the data platform architecture, including data ingestion, storage, processing, and analytics.

Data Integration Strategy: outline strategy for integrating data from various sources into the data platform.

Data Security: outline security measures to protect sensitive data.

Data Engineer: Rohit Nair

Data Platform Development: Develop and maintain the data platform infrastructure.

Data Ingestion and Processing: Ingest data from various sources and process it for analysis.

Data Storage: Implement and manage data storage solutions.

Data Analyst: Manisha

Data Analysis and Modeling: Analyze data to identify trends and patterns, and develop models for predicting future outcomes.

Report Writing: Write reports that summarize the findings of the data analysis.

Data Visualization: Create visualizations that communicate the results of the data analysis effectively.

Security Expert: Yashaswini

Security Assessment and Risk Management: Assess the security risks associated with the data platform and develop mitigation strategies.

Data Access Controls: Implement data access controls to protect sensitive information.

Technical Writer: Sai

Writing and Formatting the Project Report: Write the project conclusion and report in a clear and concise manner, ensuring that it meets all formatting requirements.

Ensuring Clarity and Accuracy: Proofread and edit the project report to ensure that it is free of errors and easy to understand.

EXECUTIVE SUMMARY

PROBLEM STATEMENT

The current data architecture landscape at the university is facing challenges due to archaic infrastructure, varying data sources, and a lack of standardization. As the institution expands its offerings in courses, including professional development, certificates, and online courses, the complexities of managing data have intensified. The hybrid data architecture, comprising both on-premise and cloud solutions, has led to difficulties in reconciling data, impeding the university's ability to establish a single source of truth for reporting and advanced analytics. Various vendor applications contribute to this complexity by using different data storage solutions. The data center's challenges in managing data are exacerbated by the institution's hyper-growth stage post-pandemic.

The existing data warehouse system is overwhelmed, resulting in high maintenance costs, limited flexibility, and challenges in introducing new products rapidly. Moreover, the university's inability to process unstructured or semi-structured data restricts its decision-making capabilities. The Chancellor's office aims to address these issues and leverage predictive analytics methods and AI to identify and support students in need. The lack of a unified data approach and data standardization further compounds the complexity, hindering the achievement of a comprehensive and accurate view of student, staff, and curriculum data.

UNIVERSITY GOALS

The Chancellor's Office aims for a transformative approach leveraging predictive analytics and AI to support students effectively. Key objectives include eliminating data reporting duplication, fostering data standardization across colleges for a comprehensive view of achievements and challenges, and implementing a common data platform. Emphasizing the importance of data security, the university seeks to scale data acquisition and processing. It aims to modernize its data architecture to overcome existing challenges and achieve the following goals:

Single Source of Truth: Establish a common data management platform to ensure seamless data integration, providing a golden copy of data with robust data governance controls, including data quality and security standards.

Unified Data Store for Analytics: Implement a unified data store for traditional and advanced analytics, incorporating data analytics governance controls, analytical data models, and support for predictive analytics and AI use cases.

Data Standardization and Reporting Tools: Facilitate data standardization and introduce reporting tools to enhance data visibility and accessibility.

Cost-Efficient Scalability: Design a scalable data architecture that optimizes costs while ensuring efficiency in data ingestion, extraction, and storage.

Predictive Analytics and AI: Enable the Chancellor's office to leverage predictive analytics and AI across critical data sets, identifying and supporting at-risk students, monitoring compliance, and providing insights for various internal and external teams.

Data Governance Processes and KPIs: Develop robust data governance processes and key performance indicators (KPIs) to ensure data integrity, security, and compliance.

SOLUTION

In summary, the university seeks a modern data solution that addresses the current challenges, supports scalability, and enables advanced analytics for informed decision-making. The proposed design should encompass unified data access and storage, streamlined data preparation, robust data security, seamless integration, cost efficiency, and scalability for future growth.

DATA PLATFORM ARCHITECTURE

As the Data Architect for this project, I am responsible for designing and implementing the data platform architecture that will support UT Dallas's data-driven initiatives. This report focuses on the technical aspects of the platform, specifically the architecture, data integration strategy, and security considerations.

BUSINESS NEEDS AND OBJECTIVES

The data platform must address the following key business needs:

Improved Decision-Making: Provide data-driven insights for strategic decision-making in areas like student engagement, resource allocation, and academic program development.

Enhanced Operational Efficiency: Streamline data management processes to optimize operational efficiency, reduce costs, and improve resource utilization.

Advanced Research and Analytics: Enable researchers and analysts to conduct complex analyses, develop predictive models, and extract deeper insights from diverse data sources.

ARCHITECTURE

The proposed data platform architecture prioritizes the following principles:

Scalability and Flexibility: The architecture must be capable of handling the university's growing data volume and diverse data types while allowing for future expansion and adaptation.

Security and Compliance: Robust security measures are essential to ensure data protection and adhere to relevant regulations like FERPA and HIPAA.

Integration and Interoperability: Seamless integration with existing systems and external data sources facilitates comprehensive data analysis and utilization.

Cost-Effectiveness: Utilizing cloud-based solutions and open-source technologies ensures cost-efficient data storage and processing.

TECHNOLOGY STACK

Based on the aforementioned principles, the following technology stack is recommended for the data platform:

Cloud-based Storage: Cloud storage solutions like Amazon S3 or Azure Data Lake Storage Gen2 offer scalability, cost-efficiency, and data accessibility.

Data Lakehouse Approach: A data lakehouse platform like Delta Lake or Apache Iceberg allows for efficient storage and management of both structured and unstructured data.

Open-source Tools: Utilizing open-source tools like Spark and Apache Kafka for data processing and streaming ensures flexibility, cost-effectiveness, and a large community of support.

Industry-Standard Security Protocols: Implementing industry-standard security protocols like encryption and IAM ensures data protection and compliance with regulations.

DATA INTEGRATION STRATEGY

The strategy for integrating data from diverse sources includes:

Data Standardization: Ensuring data consistency and interoperability across different datasets.

Data Cleansing and Validation: Implementing data quality checks to ensure the accuracy and integrity of the data.

Data Transformation: Transforming raw data into a format suitable for analysis.

Data Pipelines: Designing and implementing ETL pipelines to automate data integration processes.

DATA SECURITY

The data platform will incorporate the following security measures:

Access Controls: Granular access controls will ensure data is only accessible to authorized users based on their roles and permissions.

Data Encryption: Data will be encrypted at rest and in transit using industry-standard algorithms.

Auditing and Logging: Comprehensive audit logs will track all user activity and data access attempts.

Identity and Access Management (IAM): Secure IAM systems will enforce user authentication and multi-factor authentication.

Data Loss Prevention (DLP): DLP solutions will prevent sensitive data leaks.

Vulnerability Management: Regular vulnerability assessments and penetration tests will identify and address security vulnerabilities.

Network Security: Secure network configurations and intrusion detection/prevention systems will protect against unauthorized access.

Data Backup and Recovery: A robust backup and recovery plan will ensure data availability in case of incidents.

Security Awareness and Training: Regular training programs will educate personnel on data security best practices.

BENEFITS OF THE PROPOSED ARCHITECTURE

Implementing this data platform architecture will provide UT Dallas with the following benefits:

- Improved data management efficiency and accessibility.
- Enhanced data insights for more informed decision-making.
- Increased operational efficiency and reduced costs.
- Improved security and compliance with data privacy regulations.
- Support for advanced research and analytics initiatives.

CHALLENGES AND CONSIDERATIONS

While the proposed architecture offers significant benefits, several challenges exist in implementation:

Data Integration: Integrating data from diverse sources can be complex and require specialized skills.

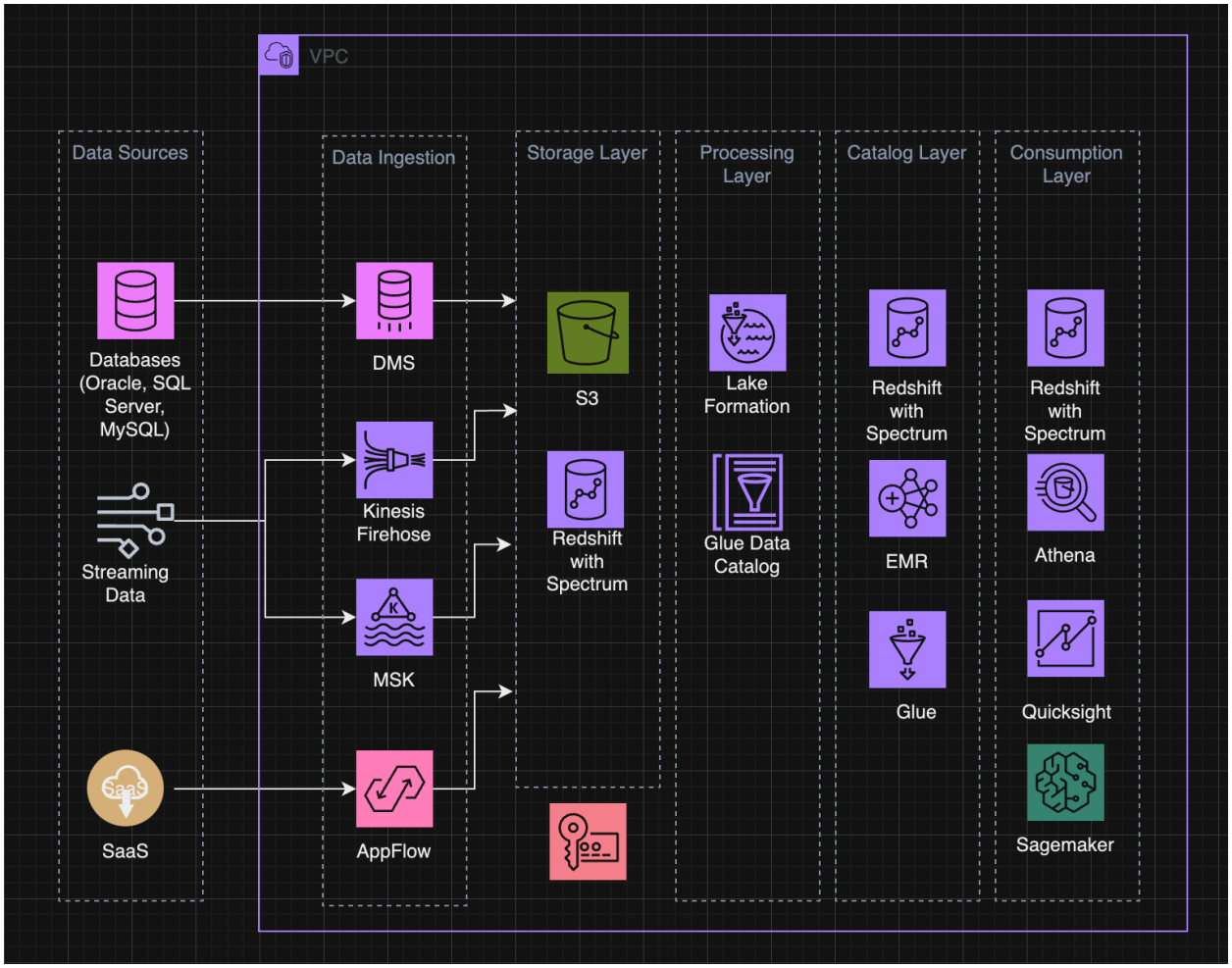
Data Security: Ensuring continuous data security requires ongoing vigilance and adaptation to evolving threats.

User Adoption: Encouraging widespread adoption of the platform among stakeholders requires user-friendly interfaces and effective training programs.

Technical Expertise: Building and maintaining the platform necessitates a team with diverse technical skills and expertise.

The data platform architecture outlined in this report provides a comprehensive and flexible solution for UT Dallas's data-driven initiatives. By prioritizing scalability, security, and cost-effectiveness, this architecture will enable the university to unlock the full potential of its data assets and achieve its strategic objectives.

DATA LAKEHOUSE ARCHITECTURE



DATA INGESTION LAYER

The system designed for bringing data into the Lake House system uses several AWS services specifically intended for ingesting data from many kinds of sources. These ingestion services transfer the data directly into the Lake House's storage layer, which contains both the data lake and data warehouse.

The architecture uses AWS Data Migration Service in the ingestion layer to connect many types of Operational RDBMS and NoSQL Databases and bring their data into Amazon S3 buckets within the data lake or Amazon Redshift staging tables. The Data Migration Service takes care of the change management by continuous replication of ongoing changes to source data.

The ingestion layer utilizes Amazon AppFlow to smoothly bring in data from SaaS applications into the data lake by setting up scheduled data transfers from the SaaS apps, or configuring the transfers to be triggered by certain events taking place within those applications.

Amazon Kinesis Data Firehose is used to take in streaming data from multiple sources and route it into the Lake House storage. Kinesis Data Firehose enables setting up an API endpoint that sources can transmit streaming data to, including clickstream information, application and infrastructure logs and monitoring metrics, and IoT telemetry and sensor data.

Amazon Managed Streaming for Apache Kafka is used to handle streaming data from applications that use Apache Kafka. Amazon MSK eliminates the operational overhead, including the provisioning, configuration, and maintenance of highly available Apache Kafka and Kafka Connect clusters.

DATA STORAGE LAYER

Amazon Redshift and Amazon S3 provide a unified, natively integrated storage. Amazon Redshift stores highly curated, conformed, trusted data that is structured into standard dimensional schemas, whereas Amazon S3 provides exabyte scale data lake storage for structured, semi-structured, and unstructured data. S3 intelligent tiering is able to move the data in the data lake into the most cost-effective tier based on usage patterns, hence aligning with performance & cost-effectiveness. Amazon Redshift Spectrum enables querying on combined datasets hosted in the data lake as well as data warehouse storage.

DATA CATALOG LAYER

Lake Formation and AWS Glue help to break down data silos and combine different types of structured and unstructured data into a centralized repository with a central catalog to store metadata for all datasets hosted in the Lake House. With AWS Lake Formation, we have a central place to manage permissions, providing Unified Governance.

DATA PROCESSING LAYER

The processing layer in the Lake House architecture has several components built for specific types of data processing needs. Powerful ELT pipelines can be built to transform structured data in the Lake House. The ELT pipelines can be used to transform data delivered by AWS DMS or Amazon AppFlow directly into Amazon Redshift staging tables. Amazon EMR makes it easy to set up, operate, and scale your big data environments by automating time-consuming tasks like provisioning capacity and tuning clusters.

DATA CONSUMPTION LAYER

The Lake House architecture makes data available for many types of users through specialized AWS services tailored to different analytics use cases like SQL queries, business intelligence, and machine learning. Data can be explored using Interactive SQL with Redshift using Redshift Spectrum and Athena. Amazon Quicksight provides serverless capability to easily create and publish rich interactive BI dashboards. Business analysts can use the Athena or Amazon Redshift interactive SQL interface to power QuickSight dashboards with data in Lake House storage.

SECURITY

Principle of Least Privilege: Grant only the permissions required to perform a task

Dealing with PII or other sensitive data

1. **Data Masking**: Masking obfuscates data and anonymization- For example, masking all but the last 4 digits of a credit card or social security number, Masking passwords, Supported in Glue DataBrew and Redshift

   Example: Use DataBrew/Redshift data masking to anonymize columns with SSN

2. **Macie** is a fully managed data security and data privacy service that uses machine learning and pattern matching to discover and protect your sensitive data in S3 buckets. It helps identify and alert you to sensitive data, such as personally identifiable information (PII)

   Example: Implement row/column level security in Redshift to restrict access to protected health info

   Set up monitors with Cloud Trail and Cloud Watch.

IAM ROLES

- Create IAM Roles for services that need to perform actions on behalf of the client.
- Assign permissions to AWS services with IAM roles

  Example: lambda functions, cloud formation, EMR services, Auto scaling, API gateway, EC2 auto-scaling, Cloud Trail, Data Pipeline, Data Brew, Data Sync, Kinesis, Lake Formation, Macie

- After creating roles attach a policy. ex: read-only access, allow or deny

  Example: Grant limited permissions to analytics tools like QuickSight, and Athena using dedicated roles

ENCRYPTION

Key Management Service (KMS): AWS manages encryption for the university, it is the easy way to control access to the data, KMS is fully integrated with IAM for authorization. By using KMS, the university can audit KMS key usage using Cloud trail. KMS can be seamlessly integrated into most AWS services. Example: To encrypt data (student profiles, transcripts) at rest in S3, enable KMS integration.

CloudTrail: CloudTrail enabled for governance, compliance, and audit of AWS account activities. Logs are stored in CloudWatch Logs or S3 for further analysis and CloudTrail can be investigated in case of resource deletion or security incidents.

Security - Amazon S3:

- Implements user-based security through IAM policies, specifying allowed API calls for specific users.
- Utilizes resource-based security, including bucket policies and Access Control Lists (ACLs), to define rules at both bucket and object levels.
- Ensures encryption at rest (SSE) and in-flight (HTTPS) for data stored in S3.
- Considers versioning and MFA delete for added security.
- Uses VPC Endpoint for secure access to S3 through a gateway from a private subnet.

Security - Glue:

- Applies IAM policies to control access to Glue service
- Configures Glue to only access databases with JDBC through SSL to enable encryption while connections from glue to databases are initiated.
- Ensures encryption of the Data Catalog using KMS and use resource policies to protect data catalog resources

Security - Athena, Redshift:

- Implements IAM policies, bucket policies, and ACLs to control access to the service
- Enforces encryption standards for data stored in S3 using SSE-KMS
- Secures data in transit using TLS between Athena/Redshift and S3 as well as JDBC connections.
- Fine-grained access using the AWS Glue Catalog

Security - Kinesis:

Kinesis is comprised of Kinesis Data Streams with an SSL endpoint.

a. Kinesis Data Streams: It uses SSL endpoints using the HTTPS protocol to do encryption in flight which means sending the data to Kinesis securely. There's also KMS integration to provide service. It implements server-side encryption using AWS KMS for encryption at rest and leverages supported Interface VPC Endpoints for private access.

b. Kinesis Data Firehose: Attaches IAM roles so it can deliver to S3 / Redshift. It can encrypt the delivery stream with KMS [Server side encryption] and supports Interface VPC Endpoints / Private Link – access privately.

c. Kinesis Data Analytics: Attaches IAM role so it can read from Kinesis Data Streams and reference sources and write to an output destination (example Kinesis Data stream/ Firehose).

Security - EMR:

- Attaches IAM roles to EC2 instances provisioned by EMR for proper S3 access and for EMRFS requests to S3(to write data back into S3)
- Two EC2 Security Groups - One for the master node and another one for the cluster node (core node or task node) to allow for node-to-node communication (In-transit encryption for node-to-node communication using TLS)

Security - SQS:

- Enables encryption in flight using the HTTPS endpoint so data is transferred to SQS securely.
- Implements server-side encryption using KMS.
- Defines IAM policies and SQS queue access policies for secure usage.
- Utilizes VPC Endpoint for private access.

Security - Lambda:

- IAM roles attached to each Lambda function will help decide what the Lambda function can do in terms of sources and targets.
- Leverages KMS for encryption of secrets.

Security - Quicksight:

- MFA enabled, Encryption at Rest.
- Row-level security and column-level security enabled to restrict access to specific rows/columns in the dataset.

By adhering to these security principles and leveraging AWS services with strong security features, the university can establish a robust and secure data platform. Regular monitoring, auditing, and updates to security policies will ensure ongoing protection of sensitive data and compliance with regulatory requirements.

DATA ANALYSIS AND VISUALIZATION

DATA ANALYSIS

The adoption of cloud-native data lakehouse architecture on Amazon Web Services (AWS) provides a powerful platform for efficient data analysis, exploration, and modeling. This segment explores how various AWS tools can be strategically employed in the different stages of data analytics to accommodate the requirements of the university.

AWS Redshift serves as the cornerstone for SQL analytics. Its distributed nature allows analysts and data scientists to execute complex queries on large datasets generated by the university. Redshift Spectrum extends this capability by enabling ad-hoc analysis directly on data stored in Amazon S3, supporting the exploration of transformed and raw data. AWS Athena, a serverless query service, further facilitates interactive querying without the need for upfront infrastructure provisioning.

Data Modeling: Redshift's analytical data modeling capabilities will allow data analysts to create tables and define relationships, supporting the development of sophisticated models for decision-making. For machine learning and AI, AWS SageMaker integrates seamlessly with Redshift, allowing the building, training, and deployment of predictive models based on historical data.

To support data-driven decision-making at operational levels, it is imperative to build analytical data models on Redshift. These models serve as a structured framework for understanding and interpreting data, fostering a more strategic approach to decision-making. Beyond mere descriptive analytics, the incorporation of machine learning and AI techniques enables the development of predictive models. These models identify patterns and trends in enrollment, grading, and tuition data, providing valuable foresight that aids in proactive decision-making. The synergy of traditional data modeling and advanced analytics ensures a holistic understanding of the educational landscape.

Business Intelligence Tools: Tools like Tableau, Power BI, or Amazon QuickSight can be connected directly to Redshift, enabling the creation of interactive dashboards for visualization. Looker, a modern BI platform, can be leveraged to enhance collaborative analysis by integrating with Redshift and providing a modeling layer for defining the university’s business metrics and dimensions.

For predictive modeling especially in areas like predicting grading trends, enrolment, and tuition trends, tools such as SageMaker simplify the development of machine learning models with built-in algorithms. AWS Lambda facilitates real-time integration by preprocessing data before feeding it into predictive models triggered by events such as data streaming through Kinesis.

Amazon QuickSight supports embedded analytics within applications, ensuring seamless data visualization. Its mobile compatibility caters to users accessing visualizations on the go, crucial for students and faculty. Customized dashboards cater to different user groups - students, faculty, and administrators. For example, an Overview Dashboard for students provides a visual representation of enrolled courses, grades, and GPA trends, while a Teaching Dashboard for faculty offers real-time class performance metrics and student engagement.
Interactive charts and graphs, geographical maps for regional analysis, and drill-down functionality enhance the visual representation of data. These techniques provide a comprehensive understanding of trends and patterns.

DATA VISUALIZATION

Amazon QuickSight for Embedded Analytics:

- QuickSight can be embedded within applications or portals for seamless data visualization.
- It supports a variety of visualization types, including charts, graphs, and dashboards.
- QuickSight dashboards can be customized based on user roles and preferences.

AWS Quicksight for Mobile Access:

- QuickSight's mobile compatibility ensures that users can access visualizations on the go.
- Mobile-friendly dashboards provide a responsive and intuitive user experience.
- This is crucial for students and faculty who may need to access data from mobile devices.

Detailed Dashboards:

For Students:

Overview Dashboard:

- Visual representation of enrolled courses, grades, and GPA trends.
- Personalized notifications for upcoming exams and assignments.
- Drill-down options for detailed course analytics.

For Faculty:

Teaching Dashboard:

- Real-time class performance metrics, student engagement, and grading analytics.
- Visualizations on course popularity, student performance trends, and class feedback.
- Tools for submitting grades and providing feedback.

For Admin:

Enrollment Management Dashboard:

- Enrollment trends and projections.
- Revenue breakdown and forecasts.
- Resource allocation metrics and efficiency indicators.

Cross-User Dashboards:

Analytics Dashboard:

- Overall institutional performance metrics.
- Predictive models for enrollment, grading, and tuition.
- Data quality and governance metrics.

Visualization Techniques:

Interactive Charts and Graphs:

- Line charts for trend analysis.
- Bar graphs for comparative analysis.
- Heatmaps for identifying patterns and anomalies.

Geographical Maps:

- Geographic visualizations for regional analysis of enrollment and performance.
- Maps can display the distribution of students and faculty across different locations.

Drill-Down Functionality:

- Interactive dashboards with drill-down capabilities for detailed exploration.
- Users can click on specific data points to view underlying details and trends.

Intuitive user interfaces with filter options, automated alerts for critical events, and export functionality for detailed analysis enhance user interaction. These features contribute to a user-friendly and efficient experience.

User Interaction:

Intuitive User Interface:

- User-friendly interfaces with intuitive navigation.
- Filter options for date ranges, courses, and demographics.

Alerts and Notifications:

- Automated alerts for critical events, such as grade submissions or enrollment deadlines.
- Notifications based on personalized preferences.

Export Functionality:

- Export options for detailed analysis or reporting.
- Data can be exported in various formats (CSV, Excel) for external use.

Overall, the integration of various AWS tools within a cloud-native data lakehouse architecture can help the university with a robust and scalable platform for data analytics. From exploration and modeling to visualization and governance, AWS provides a comprehensive ecosystem to unlock valuable insights, facilitating informed decision-making and driving academic excellence.

CONCLUSION

The blog post concludes with a comprehensive overview of the Lake House architecture on AWS, highlighting the myriad benefits and the strategic use of purpose-built services across different layers. The authors underscore the agility and scalability afforded by AWS services, allowing organizations to seamlessly integrate and process data from various sources.

The flexibility of the architecture is emphasized, with AWS providing a suite of tools such as Amazon Redshift, AWS Glue, Amazon Kinesis, Amazon Athena, Amazon SageMaker, and Amazon QuickSight, each serving a specific purpose in the data lifecycle. The authors stress the importance of these services in efficiently managing and transforming data, enabling powerful analytics, and facilitating machine learning endeavors.

A key takeaway is the Lake House architecture's ability to democratize data consumption, making it accessible to different user personas. This is achieved by providing unified interfaces to access data stored in Amazon S3, Amazon Redshift, and the Lake Formation catalog. Whether it's interactive SQL queries, business intelligence dashboards, or machine learning model development, the Lake House architecture is positioned as an all-encompassing solution.

The blog encourages readers to delve into detailed architectural patterns, walkthroughs, and sample code provided in additional resources. These resources are aimed at guiding users in building and optimizing each layer of the Lake House Architecture, offering practical insights for implementation.

In essence, the Lake House architecture on AWS is portrayed as a versatile and future-ready solution, capable of accommodating diverse data types, processing massive volumes of information, and supporting a range of analytics use cases. The authors believe that adopting this architecture will empower organizations to derive meaningful insights from their data and position themselves for future advancements in analytics and technology.