

OAES
Technical Architecture Document

Version <1.0>

By
Sujindar Selvaraj (MT2019143)
Swati N Murthy (MT2019120)
Sravya M (MT2019114)

Table of Contents

Introduction	7
Purpose	7
Scope	7
Definitions, Acronyms, and Abbreviations	7
Architecture Principles[1]	7
Principle 1: Technology Independence	7
Principle 2: Ease-of-Use	7
Principle 3: Requirements-Based Change	8
Principle 4: Service Orientation	8
Principle 5: Data is an Asset	9
Principle 6: Data is Shared	9
Principle 7: Data is Accessible	10
Principle 8: Data Trustee	10
Principle 9: Common Vocabulary and Data Definitions	11
Principle 10: Data Security	11
Architectural Representation	11
Software Architecture	12
Architectural Goals and Constraints	18
Web Browser	18
Application Firewall	18
Authentication Server	18
User Portal	19
Student Portal	19
Local Server	19
Question Bank Module	20
Exam Control Module	20
Assessment Module	20
Evaluation Module	20
Reports Module	21
Use-Case View	21
Deployment View	22
Size and Performance[2]	23
Bibliography	23

Software Architecture Document

1. Introduction

OAES - Online Assessment Evaluation System is an online platform for conducting national level examinations like JEE, NEET, GATE, etc.. The system should be dynamic to conduct different types of exams and should be capable of many functionalities like user/student management, questions management, question paper generation, and conducting the exam. It should also be able to generate comprehensive reports.

Purpose

This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

Scope

This document elaborates details regarding use-case view, application and deployment architecture of the OAES that will provide designers with a concrete idea of the requirements.

Definitions, Acronyms, and Abbreviations

OAES - Online Assessment Examination System.

2. Architecture Principles[1]

Principle 1: Technology Independence

Statement:

Applications are independent of specific technology choices and therefore can operate on a variety of technology platforms.

Rationale:

Independence of applications from the underlying technology allows applications to be developed, upgraded, and operated in the most cost-effective and timely way. Otherwise technology, which is subject to continual obsolescence and vendor dependence, becomes the driver rather than the user requirements themselves.

Realizing that every decision made with respect to IT makes is dependent on that technology, the intent of this principle is to ensure that Application Software is not dependent on specific hardware and operating systems software.

Implications:

- This principle will require standards which support portability
- Middleware should be used to decouple applications from specific software solutions.

Principle 2: Ease-of-Use

Statement:

Applications are easy to use. The underlying technology is transparent to users, so they can concentrate on tasks at hand.

Rationale:

The more a user has to understand the underlying technology, the less productive that user is. Ease-of-use is a positive incentive for use of applications. It encourages users to work within the integrated information environment instead of developing isolated systems to accomplish the task outside of the enterprise's integrated information environment. Most of the knowledge required to operate one system will be similar to others. Training is kept to a minimum, and the risk of using a system improperly is low.

Using an application should be as intuitive as driving a different car.

Implications:

- Applications will be required to have a common "look-and-feel" and support ergonomic requirements; hence, the common look-and-feel standard must be designed and usability test criteria must be developed.
- Guidelines for user interfaces should not be constrained by narrow assumptions about user location, language, systems training, or physical capability.
Factors such as linguistics, customer physical infirmities (visual acuity, ability to use keyboard/mouse), and proficiency in the use of technology have broad ramifications in determining the ease-of-use of an application.

Principle 3: Requirements-Based Change**Statement:**

Only in response to business needs are changes to applications and technology made.

Rationale:

This principle will foster an atmosphere where the information environment changes in response to the needs of the business, rather than having the business change in response to IT changes. This is to ensure that the purpose of the information support - the transaction of business - is the basis for any proposed change.

Unintended effects on business due to IT changes will be minimized.

A change in technology may provide an opportunity to improve the business process and, hence, change business needs.

Implications:

- Changes in implementation will follow full examination of the proposed changes using architecture
- There is no funding for a technical improvement or system development unless a documented business need exists
- Change management processes conforming to this principle will be developed and implemented
- This principle may bump up against the responsive change principle
We must ensure the requirements documentation process does not hinder responsive change to meet legitimate business needs. The purpose of this principle is to keep the focus on business, not technology needs - responsive change is also a business need.

Principle 4: Service Orientation**Statement:**

The architecture is based on a design of services which mirror real-world business activities comprising

the enterprise (or inter-enterprise) business processes.

Rationale:

Service orientation delivers enterprise agility and Boundaryless Information Flow.

Implications:

- Service representation utilizes business descriptions to provide context (i.e., business process, goal, rule, policy, service interface, and service component) and implements services using service orchestration
- Service orientation places unique requirements on the infrastructure, and implementations should use open standards to realize interoperability and location transparency
- Implementations are environment-specific; they are constrained or enabled by context and must be described within that context.
- Strong governance of service representation and implementation is required.

Principle 5: Data is an Asset

Statement:

Data is an asset that has value and is managed accordingly.

Rationale:

Data is a valuable resource; it has real, measurable value. In simple terms, the purpose of data is to aid decision-making. Accurate, timely data is critical to accurate, timely decisions. Most assets are carefully managed, and data is no exception. Data is the foundation of our decision-making, so we must also carefully manage data to ensure that we know where it is, can rely upon its accuracy, and can obtain it when and where we need it.

Implications:

- Stewards must have the authority and means to manage the data for which they are accountable
- The role of data steward is critical because obsolete, incorrect, or inconsistent data could be passed to users and adversely affect decisions across the system.
- Part of the role of data steward, who manages the data, is to ensure data quality. Procedures must be developed and used to prevent and correct errors in the information and to improve those processes that produce flawed information. Data quality will need to be measured and steps taken to improve data quality - it is probable that policy and procedures will need to be developed for this as well.
- Since data is an asset of value, data stewards accountable for properly managing the data must be assigned at the module level

Principle 6: Data is Shared

Statement:

Users have access to the data necessary to perform their duties; therefore, data is shared across different modules.

Rationale:

Timely access to accurate data is essential to improving the quality and efficiency of decision-making. The speed of data collection, creation, transfer, and assimilation is driven by the ability of the system to efficiently share these islands of data across the modules.

Shared data will result in improved decisions since we will rely on fewer (ultimately one virtual) sources of more accurate and timely managed data for all of our decision-making. Electronically shared data will result in increased efficiency when existing data entities can be used, without re-keying, to create new entities.

Implications:

- To enable data sharing we must develop and abide by a common set of policies, procedures, and standards governing data management and access for both the short and the long term.
- We will also need to develop standard data models, data elements, and other metadata that defines this shared environment and develop a repository system for storing this metadata to make it accessible.
- For the long term, as legacy systems are replaced, we must adopt and enforce common data access policies and guidelines for new application developers to ensure that data in new applications remains available to the shared environment and that data in the shared environment can continue to be used by the new applications.
- For both the short term and the long term we must adopt common methods and tools for creating, maintaining, and accessing the data shared across the enterprise.
- This principle of data sharing will continually "bump up against" the principle of data security - under no circumstances will the data sharing principle cause confidential data to be compromised.
- Data made available for sharing will have to be relied upon by all users to execute their respective tasks. This will ensure that only the most accurate and timely data is relied upon for decision-making.

Principle 7: Data is Accessible

Statement:

Data is accessible for users to perform their functions.

Rationale:

Wide access to data leads to efficiency and effectiveness in decision-making, and affords a timely response to information requests and service delivery. Using information must be considered from a system perspective to allow access by a wide variety of users. Staff time is saved and consistency of data is improved.

Implications:

- Accessibility involves the ease with which users obtain information.
- The way information is accessed and displayed must be sufficiently adaptable to meet a wide range of users and their corresponding methods of access.
- Access to data does not constitute understanding of the data - personnel should take caution not to misinterpret information
- Access to data does not necessarily grant the user access rights to modify or disclose the data.

Principle 8: Data Trustee

Statement:

Each data element has a trustee accountable for data quality.

Rationale:

One of the benefits of an architected environment is the ability to share data, across the system. As the degree of data sharing grows and business units rely upon common information, it becomes essential that only the data trustee makes decisions about the content of data. Since data can lose its integrity when it is entered multiple times, the data trustee will have sole responsibility for data entry which eliminates redundant human effort and data storage resources.

Note: A trustee is different from a steward - a trustee is responsible for accuracy and currency of the data, while responsibilities of a steward may be broader and include data standardization and definition tasks.

Implications:

- The data trustee will be responsible for meeting quality requirements levied upon the data for which the trustee is accountable.
- It is essential that the trustee has the ability to provide user confidence in the data.

Principle 9: Common Vocabulary and Data Definitions

Statement:

Data is defined consistently throughout the architecture, and the definitions are understandable and available to all users.

Rationale:

The data that will be used in the development of applications must have a common definition to enable sharing of data. A common vocabulary will facilitate communications and enable dialog to be effective. In addition, it is required to interface systems and exchange data.

Implications:

- The enterprise must establish the initial common vocabulary for the business; the definitions will be used uniformly throughout the architecture.
- Whenever a new data definition is required, the definition effort will be coordinated and reconciled with the system "glossary" of data descriptions.
The data administrator will provide this coordination.
- Ambiguities resulting from multiple definitions of data must give way to accepted system-wide definitions and understanding.

Principle 10: Data Security

Statement:

Data is protected from unauthorized use and disclosure. Data is always sent and received after encryption.

Rationale:

Open sharing of information and the release of information must be balanced against the need to restrict the availability of classified, proprietary, and sensitive information.

Pre-decisional (work-in-progress, not yet authorized for release) information must be protected to avoid unwarranted speculation, misinterpretation, and inappropriate use.

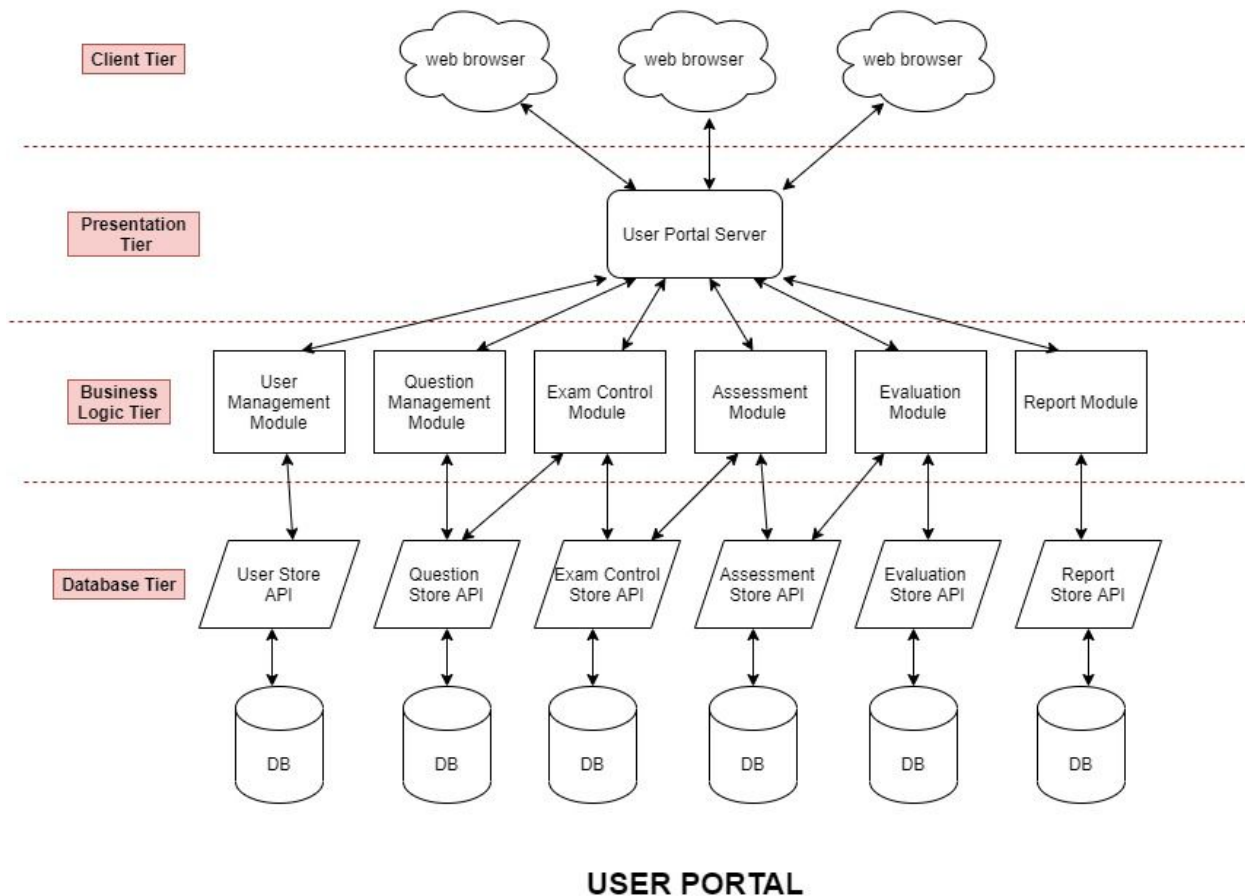
Implications:

- Aggregation of data, both classified and not, will create a large target requiring review and de-classification procedures to maintain appropriate control.
Access to information based on a need-to-know policy will force regular reviews of the body of information.
- Data security safeguards can be put in place to restrict access to "view only" or "never see".
- Security must be designed into data elements from the beginning; it cannot be added later
Systems, data, and technologies must be protected from unauthorized access and manipulation, inadvertent or unauthorized alteration, sabotage, disaster, or disclosure.

3. Architectural Representation

3.1 Software Architecture

3.1.1 User Portal

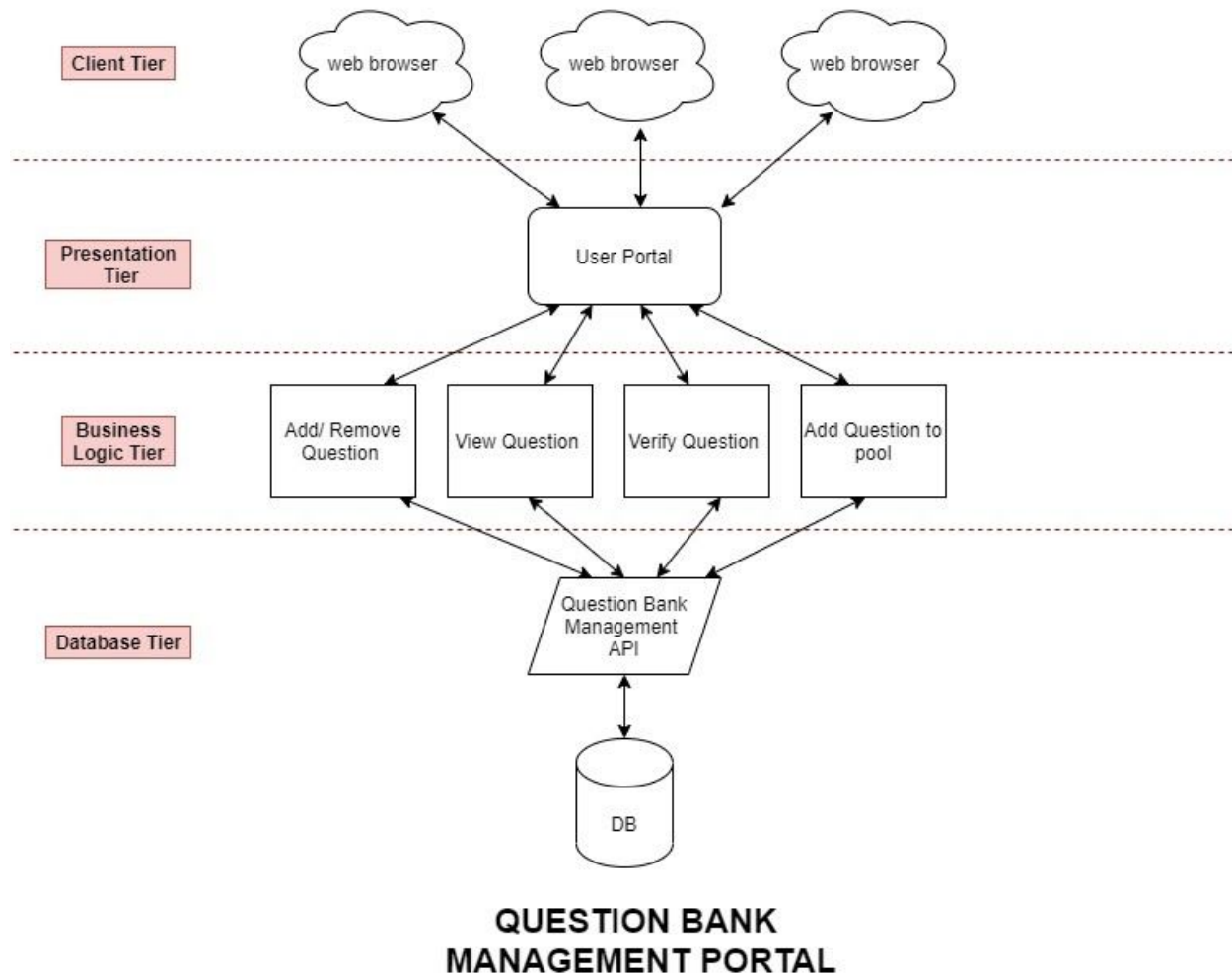


- N-tier architecture which allows the users to access the resources based on the access privilege.
- There are various modules in the user portal, which themselves have their own architecture which are explained below.

3.1.2 Question Bank Management Module

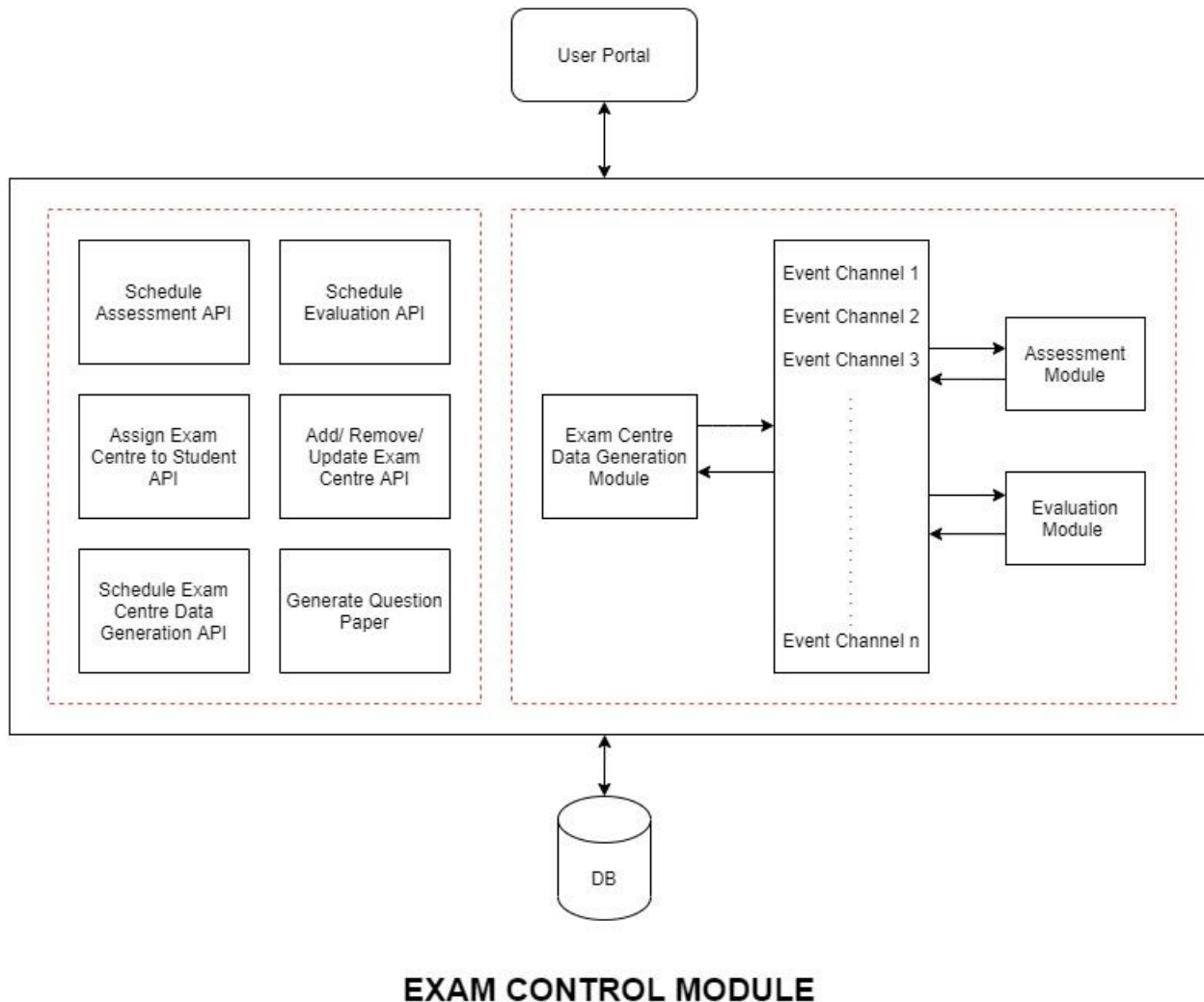
- This module contains an API for managing the questions. The users can add, modify and remove questions. The questions are publicly visible only after they are verified.

- The questions can be added to different question pools. The question pool might have their own setting for every question such as difficulty, marks, etc.
- The difference between question and question pool is that the question contains information such as the question itself, its type, subject, etc. The question pool contains information such as difficulty, marks, etc. This separation allows the same question to be in different pools for different contexts. The question paper can be generated in the exam control module using single or multiple question pools



3.1.3 Exam Control Module

- The module contains api to schedule assessment, evaluation, exam centre data generation.
- It also contains an API to manage the Exam Centre, and to allot exam centres to students based on their preference, and generate question paper from single or multiple question pools.
- User Portal has an event channel waiting for multiple events such as assessments, evaluation and exam centre data generation. Since all these events are independent of each other, broker topology is used for listening to the events. The assessment module, evaluation module themselves have their own architecture, which is described below.
- Exam Centre data generation module is used to precompute the data that needs to be sent to each exam centre. The data may contain different question paper, student details of that particular exam centre, etc.
- Exam Centre Module is a combination of N-tier architecture and Event driven architecture.

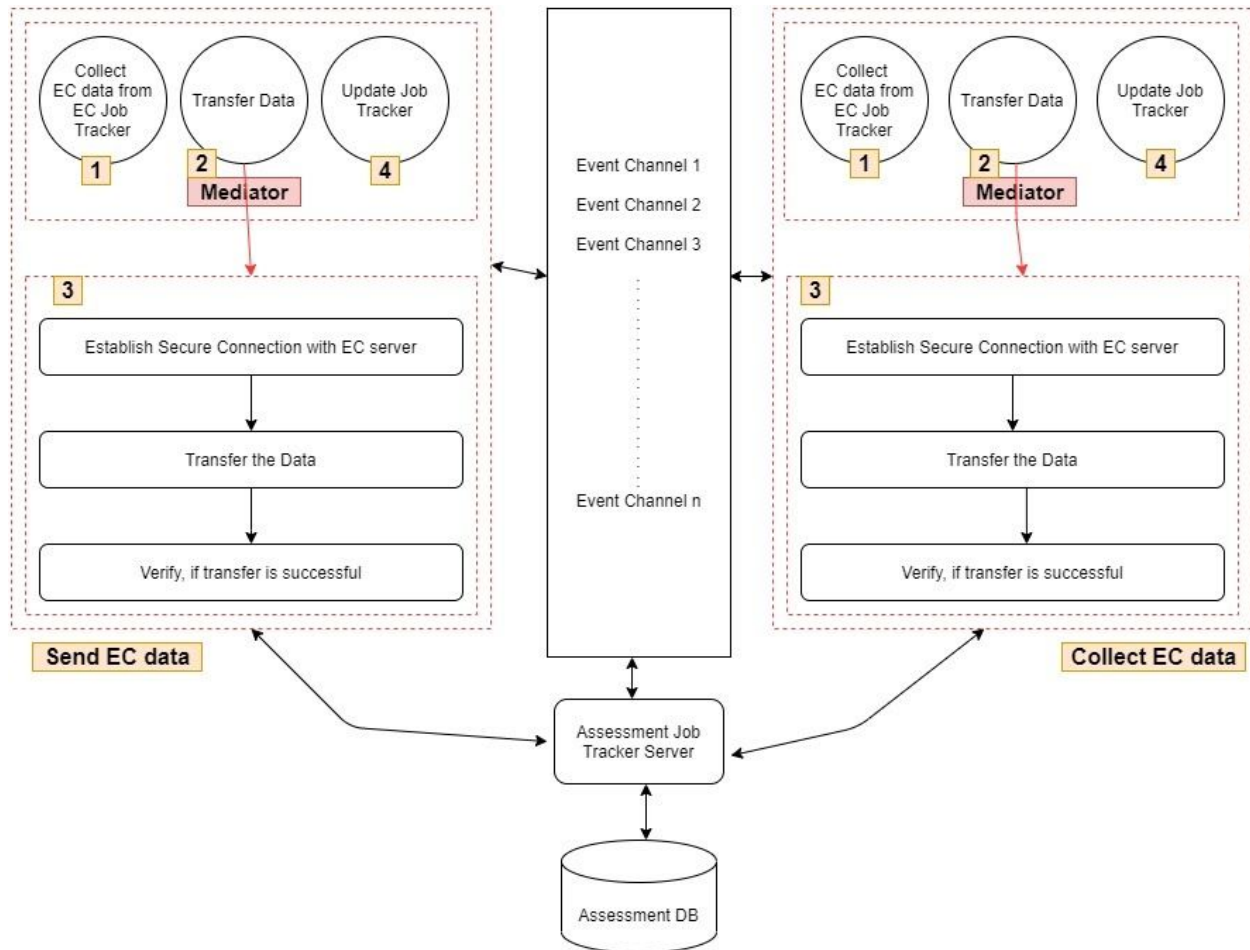


3.1.4 Assessment Module

- Assessment Module is designed as a event driven architecture. There are two main functions of the assessment module. To send the exam centre data to the local server and to collect the exam centre data from the local server. Both are independent events, and are listened to by a broker topology.
- The job tracker contains what data needs to be sent to each exam centre. This is precomputed at the exam control module.
- To send exam centre data, it queries the job tracker to get all details about the exam centre. It then establishes a secure connection with the exam centre local server and transfers the data. Establishing connection, transferring the data, verifying if transferred correctly are all implemented as layered architecture.
- Since this module is independent, it's highly scalable and the data can be sent to multiple exam centers simultaneously by querying the job tracker.
- Similarly, collecting the exam centre data from the exam centre local server, the module the queries the job tracker, from where it has to collect the data. It then establishes a secure connection to the local server, transfers the data and validates it before updating the job tracker. This is also implemented as layered architecture. This module is also independent and highly

scalable, the data can be collected from multiple exam centre local servers simultaneously by querying the job tracker.

- It's designed as event driven architecture because all the exams begin and ends at a particular time except few because of unexpected events which the module should be able to handle.

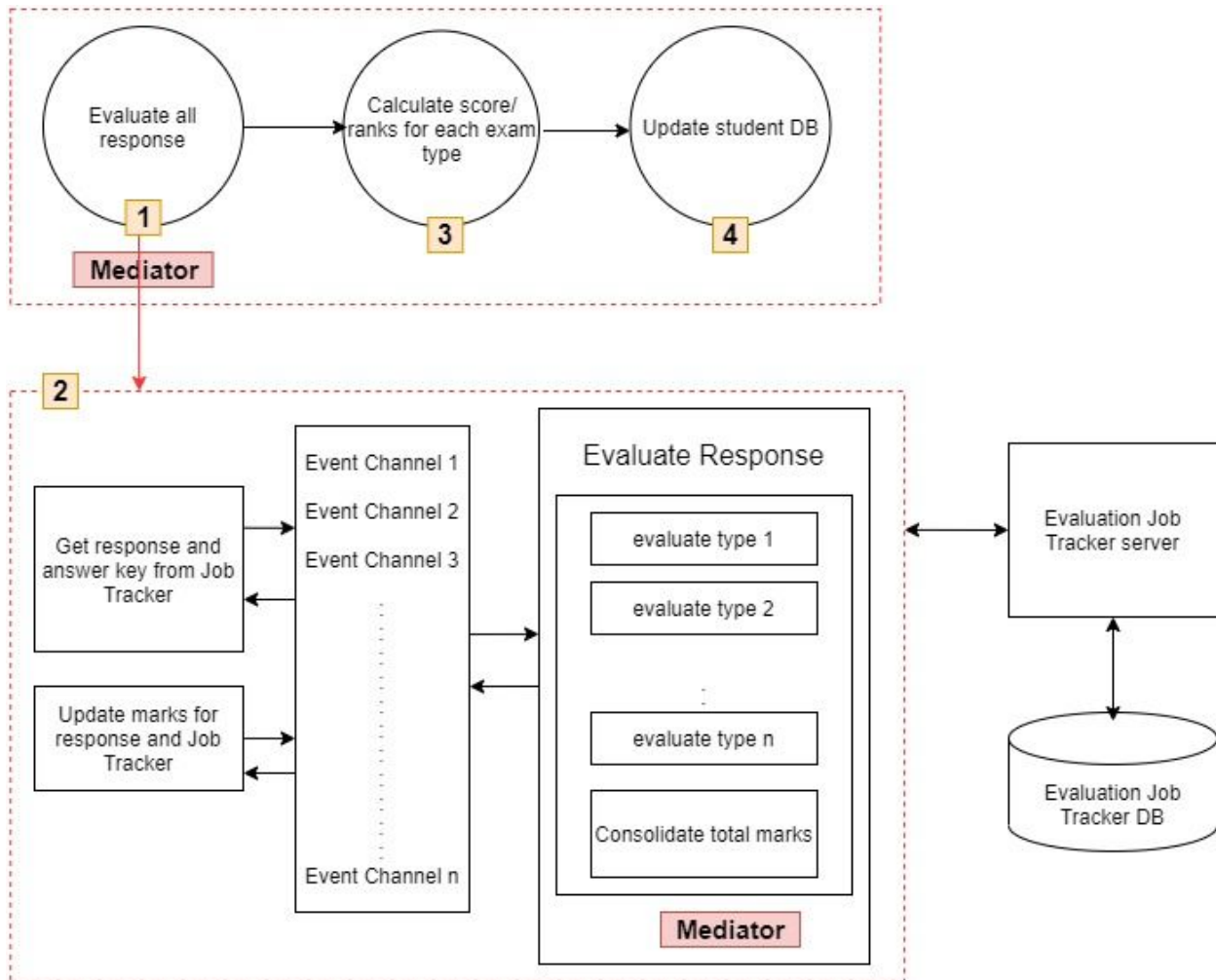


ASSESSMENT MODULE

3.1.5 Evaluation Module

- Evaluation Module is designed as an event driven architecture. The main job of the evaluation module is to evaluate all responses, compute the marks and final result as per that particular exam. It's a combination of both broker topology and Mediator topology.
- Broker topology is used for a series of events to take place. First evaluating all the responses finding the marks, then computing the final results as per the exam, updating the student DB, the result for each student.
- Mediator topology is used for evaluating all the responses. There are three events, to get a single response and corresponding answer key, to evaluate the response, to update the marks and job tracker.
- Evaluate response is subscribed to get response and answer key. Every time a response and answer key is published, it evaluates that particular response and through various evaluation types which in turn is monitored by Mediator topology. It then publishes the mark of that particular response.

- The update marks for response and job tracker are subscribed to evaluate response and update the mark in for the response in the database and update the job tracker.
- All these are independent of each other and are microservices, hence can be scaled easily. Since evaluating response is much more time consuming than other two events, more instances of this can be deployed. Can also be scaled based on the constraints such as number of responses and available time to announce the result, etc.



EVALUATION MODULE

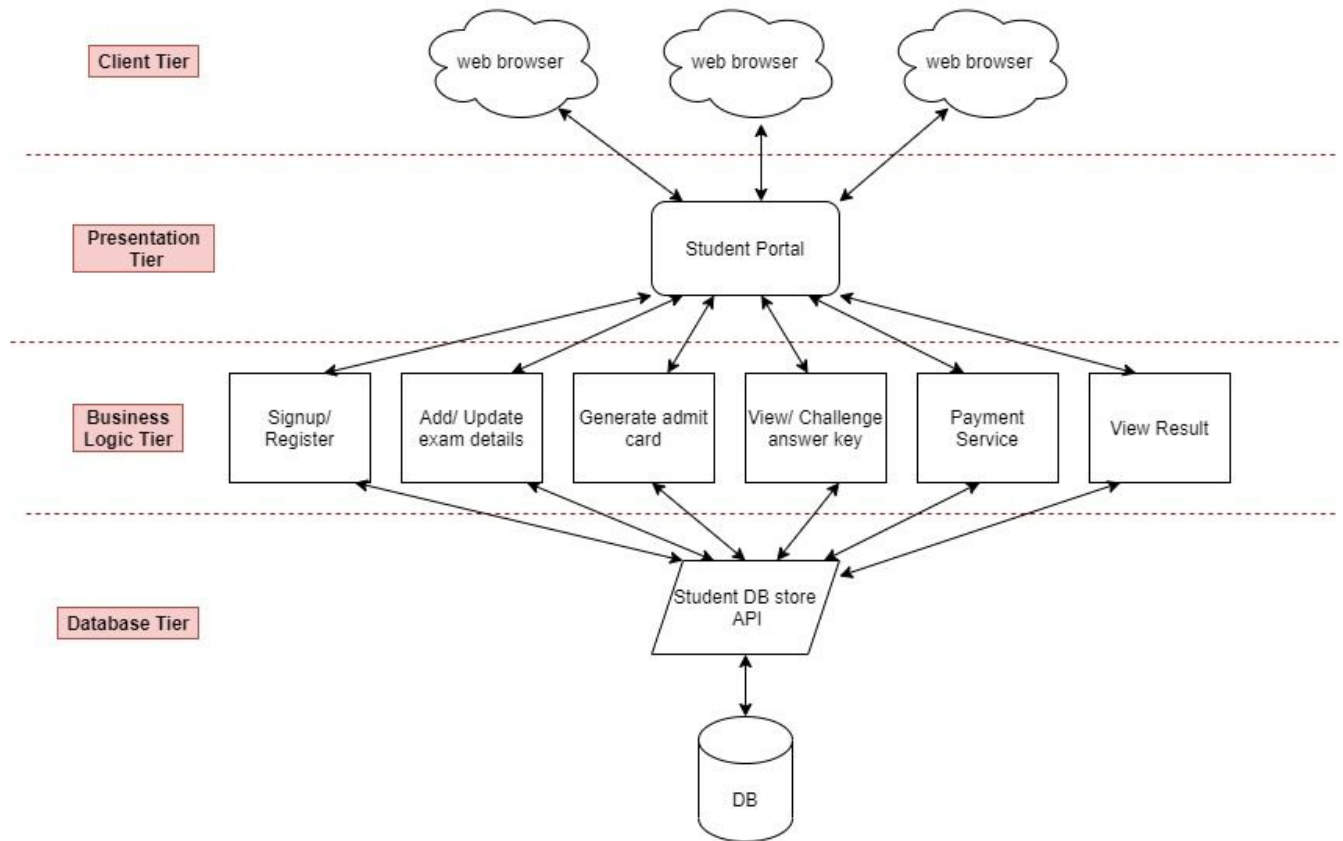
3.1.6 Student Portal

- N-tier architecture which contains API for the student to register, enroll in the exam, generate admit card before exam, view answer key, make payment for application and view his/her results.
- They are implemented as independent microservices and should be highly scalable. For example, the view result module should have more instances during announcement of results.

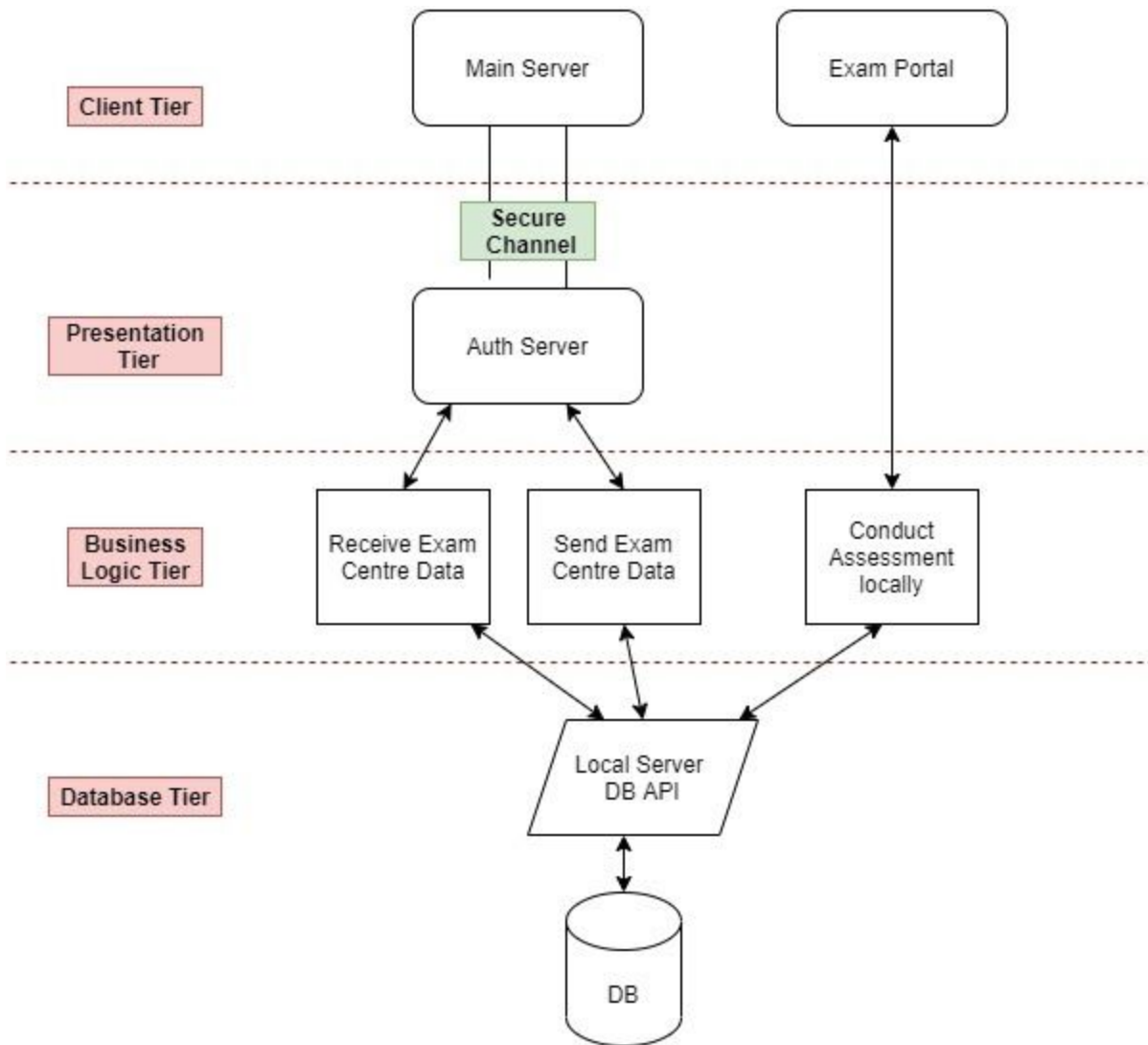
3.1.7 Local Exam Centre Module

- The main function of the local server is to collect the exam centre data in a secure channel, conduct the exam locally without Internet and send the response and other exam details back to the main server.

- It also contains a local authentication server, to establish connection with the main server and to securely serve the exam portal locally.
- It is an N-tier architecture.



STUDENT PORTAL



LOCAL EXAM CENTRE SERVER

4. Architectural Goals and Constraints

4.1 Web Browser

1. All latest browsers should support OAES applications.
2. Applications should be accessed via browser either on laptop, mobile or tablet. The application should be a responsive one.

4.2 Application Firewall

1. Firewall should monitor incoming traffic.
2. Firewall should block access from VPNs
3. Firewall should protect the system against attacks like DoS, DDoS, etc..
4. Firewall chosen should be strong and should have incorporated the latest, most secure open source algorithm.
5. All requests must be made via a secure channel.

4.3 *Authentication Server*

1. Registration is a prerequisite to login.
2. The authentication server shall be functioning 24*7..
3. Every incoming request must pass through the authentication server
4. Login is mandatory for all users to access the portal.
5. Only verified and authorized users are allowed access features of the portal like adding questions, etc..
6. No user shall be able to access any page without authorization.
7. A user or student shall be allowed to login only from one device at a time.
8. Rate limiting shall be incorporated wherever required.

4.4 *User Portal*

1. Users (Professors, Exam controller) from various institutes shall be allowed to register.
2. All the registered users shall provide necessary documents to ensure verification of users, the accounts shall be verified and only then, activated.
3. Students should not be able to access User Portal.
4. User portal should support users with multiple roles like admin, professor, exam controller. The portal should support role based restrictive access to the features upon login.
5. 2 factor authentication must be supported.
6. The password mentioned shall be a strong password. It shall contain a minimum 8 characters. 1 uppercase, 1 lowercase, 1 numeric, 1 special character is mandatory.
7. Should have two 9's of availability.
8. Should support 250 simultaneous access.
9. The client portion shall require less than 20 MB disk space and 32 MB RAM

4.5 *Student Portal*

1. Students shall be allowed to register on the portal.
2. Students should be able to login into the portal and see the details they had filled during registration and see status of application.
3. 2 factor authentication must be supported.
4. All the functions of the student portal should be available after login only.
5. Before the exam, students should be able to view and download Admit Card.
6. After the exam, Student should be able to check his/her response sheet.
7. Once the results are announced, the result score card should be visible and available for download here.
8. The password mentioned shall be a strong password. It shall contain a minimum 8 characters. 1 uppercase, 1 lowercase, 1 numeric, 1 special character is mandatory.
9. Should have two 9's of availability.
10. Should support 1K simultaneous access.
11. The client portion shall require less than 20 MB disk space and 32 MB RAM

4.6 *Local Server*

1. Local servers are present in each of the examination centers.
2. Local servers should receive the exam center data(question paper, student info..) from the exam control server via a secure channel in advance.
3. Local servers should use Intranet/LAN facilities to send question paper to each system in the examination center. Internet facility must not be required at this stage.

4. The local server should be ON and available throughout the duration of the exam and until the data is safely sent to the evaluation module.
5. The local server presents the question paper to students upon authentication.
6. The local server stores each and every response of students.
7. In no case, the student response can be lost. Responses must be saved after each entry and also at regular intervals.
8. Data is backed up into the local server.
9. Once the exam is over, the local server should collect response sheets of each and every student and send it to the evaluation module servers via a secure channel.
10. Should have 6 9's of availability.
11. The client portion shall require less than 20 MB disk space and 32 MB RAM

4.7 Question Bank Module

1. The questions shall have the following details <Subject, Question type, Knowledge category(Bloom's taxonomy)...>
2. Only Professors shall be allowed to add questions. Professor can delete questions that only s/he posted.
3. Students are never allowed to access this portal.
4. Questions input shall be verified by a random teacher.
5. Verified questions shall be input into the Question pool.
6. Should have two 9's of availability.
7. Should support 250 simultaneous access.
8. The client portion shall require less than 20 MB disk space and 32 MB RAM

4.8 Exam Control Module

1. Exam controller should be able to access this module after authentication.
2. Professors, Students etc must never be able to access this module.
3. Taking in the preference from students the exam control module should allocate the exam center to each and every eligible student.
4. Exam control module shall have the capability to add/remove exam centers as needed.
5. Exam control module should take questions from QuestionBank module & also takes in configuration details and produces a question paper.
6. Exam control module should have capability to schedule assessment and evaluation of exams.
7. Should have three 9's of availability.
8. The client portion shall require less than 20 MB disk space and 32 MB RAM

4.9 Assessment Module

1. Assessment module should send the exam center data to each of the local servers in exam centers via a secure channel at a scheduled time.
2. Assessment module should collect the exam center data from each of the local servers in exam centers via a secure channel at a scheduled time.
3. Assessment module must ensure that the data sent and received are never tampered, it must have a defined mechanism to verify the same.
4. Assessment module should be highly scalable to transfer and receive exam center data to and from the local server in a stipulated time.
5. Should have five 9's of availability.
6. Should support 250 simultaneous access.

4.10 Evaluation Module

1. The evaluation module receives the input from the Assessment module.

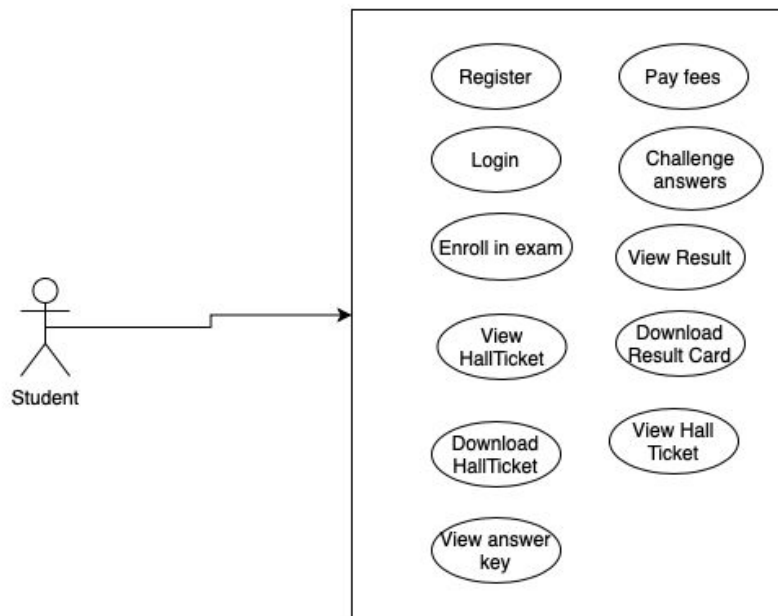
2. The evaluation module shall take a response sheet, evaluate the same and store the score for each answer.
3. The evaluation module should be highly scalable to produce the results of the exam in a stipulated time.
4. The evaluation module should be capable of taking in required configurations and produce the results in required fashion (Percentage, Percentile, Score out of 1000.. etc)
5. The results of the evaluation module shall be stored into the database.
6. Data regarding the results should never be tampered with.
7. The results must be visible on student portal and must be available for viewing and downloading until the mentioned date and time after which it can be deleted.
8. Should have two 9's of availability.
9. Should support 250 simultaneous access

4.11 Reports Module

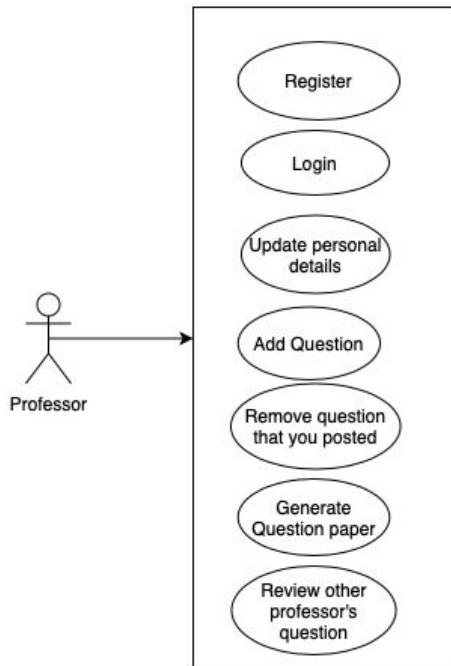
1. Reports module shall be flexible to generate different reports as per the requirements.
2. Reports shall be saved for a stipulated time and used for analysis.
3. Reports should be available for download in various formats.

5. Use-Case View

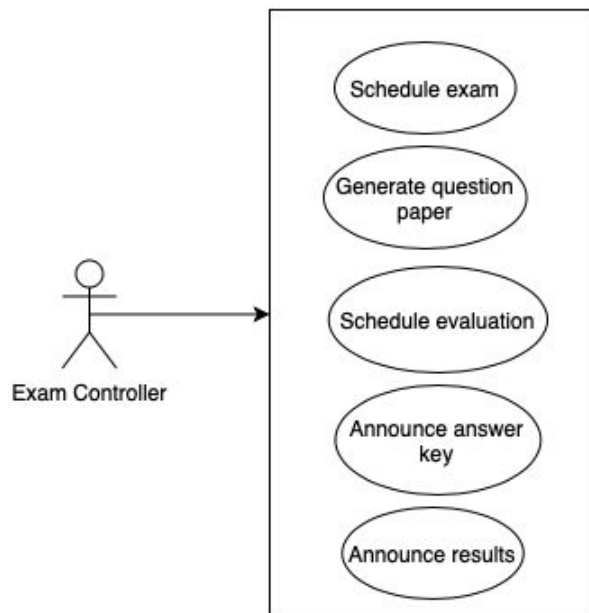
5.1 Student Use Case View



5.2 Professor Use Case View



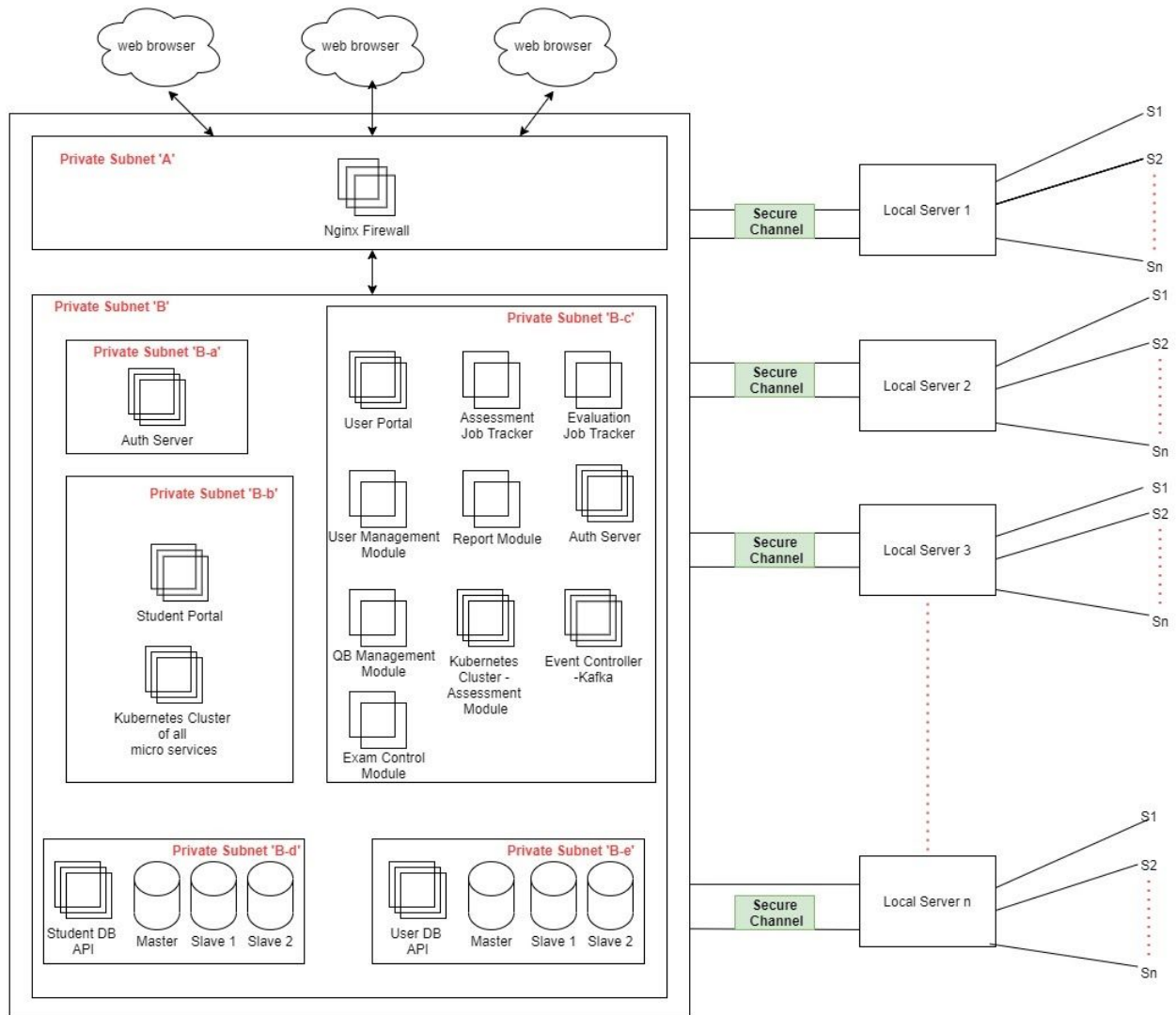
5.3 Exam Controller



6. Deployment View

- The system has a firewall which is the single entry point to the system. It filters all the incoming requests and protects it from different network attacks.
- Authentication server, modules related to users, modules related to students and databases are all deployed as private subnets which are not directly accessible publicly.
- All the modules are deployed in multiple instances and load balanced and hence are highly available.

- Evaluation Module and assessment module are deployed as separate kubernetes clusters. These are two components that require high compute, reliability and hence are orchestrated by an orchestration engine such as kubernetes.
- The below is the overall deployment architecture of OAES.



DEPLOYMENT ARCHITECTURE

7. Size and Performance[2]

- The system must be capable of supporting 30L students and 5K professors.
- The system must support simultaneous 10K students access and simultaneous 250 professors access.
- The platform must support at least 12000 exam centres.
- The platform should have three 9's of availability.
- The client portion shall require less than 20 MB disk space and 32 MB RAM

8. Bibliography

[1] The TOGAF standard architecture principles. ([Link](#))

[2] "How to Draw Useful Technical Architecture Diagrams" by Jimmy Soh ([Link](#))