



Intaliq

SE322: Software Design and Architecture

Section: 1340

Instructor

Dr. Suad Ramouni

Prepared by:

YARA FARIS FAIHAN ALBUGAMI

ROSE SAEED RAKAN AL RAKAN

SHOUG FAWAZ ABDULLAH ALOMRAN

FAJR ALGHOFAILI

MOZOUN ALMUQRIN

MUNIRA YAZEED ABDULAZIZ ALFARIS

Table Of Contents

1. Quality Attributes.....	3
1.1 Quality Attribute: Security.....	3
1.2 Quality Attribute: Availability.....	3
1.3 Quality Attribute: Performance.....	3
1.4 Quality Attribute: Usability.....	4
2. Quality Attribute Scenarios.....	4
2.1 Scenario for Quality Attribute: Security.....	4
2.2 Scenario for Quality Attribute: Availability.....	4
2.3 Scenario for Quality: Performance.....	5
2.4 Scenario for Quality Attribute: Usability.....	5
3. Architectural Style.....	5
3.1 Selected Architectural Style.....	5
3.2 Justification.....	6
3.3 How We Will Implement It in Intaliq.....	6
1. Client: Intaliq Mobile Application.....	7
2. Application Server: Intaliq Backend.....	7
3. Database: Intaliq Database.....	7
4. External AI Service.....	7

1. Quality Attributes

1.1 Quality Attribute: Security

One of the key quality attributes in Intaliq is *security*, as the application contains private data such as user account data, logs, and personal fitness goals. The system must ensure that the data is protected from any unauthorized access and the users have to be authenticated before accessing their accounts.

The authentication and authorization mechanisms in Intaliq must ensure that both coaches and regular users are restricted to execute only those functions for which they are authorized.

1.2 Quality Attribute: Availability

Availability is essential as users will have to access Intaliq in order to monitor their activities, define their fitness objectives, find partners, and make appointments for group classes.

Intaliq should be available throughout regular operation; however, in case any of its services becomes unavailable, the application should warn users and ensure that incomplete procedures do not get considered as completed.

Moreover, the Intaliq architecture features the Offline Sync functionality which holds actions till the time the device gets online again.

1.3 Quality Attribute: Performance

Why *performance* matters: Users of the Intaliq software must be able to use the program in an efficient manner. This means that tasks such as checking available sessions, logging, and recording fitness goals should be done in a timely manner without any significant delay.

Efficiency in processing user input through the mobile app and the back end server is necessary. Good performance will ensure that users can use the application efficiently.

1.4 Quality Attribute: Usability

Usability is necessary since the purpose of the program is to motivate the citizens of Saudi Arabia to engage in outdoor fitness exercises.

The program must ensure the ease of use through the provision of a user-friendly mobile interface whereby users can access all the elements of the application without going through complex steps.

Moreover, the program must have an alert system for successful completion of actions.

2. Quality Attribute Scenarios

2.1 Scenario for Quality Attribute: Security

Element	Description
Source	An unauthorized user
Stimulus	Attempts to access another user's fitness information without authentication
Environment	Normal system operation
Artifact	Authentication and Authorization component
Response	The system verifies the user's authentication and permissions and denies unauthorized access
Response Measure	100% of unauthorized access attempts in the defined security test cases are rejected

2.2 Scenario for Quality Attribute: Availability

Element	Description
Source	A user experiencing an internet connection failure
Stimulus	The user attempts to log an activity while the internet connection is unavailable
Environment	Mobile application operating without internet access
Artifact	Intaliq Mobile Application and Offline Sync component

Response	The system temporarily stores the activity locally and synchronizes it when the internet connection is restored
Response Measure	100% of successfully queued test activities synchronize after reconnection without data loss

2.3 Scenario for Quality: Performance

Element	Description
Source	An authenticated user
Stimulus	Requests to view available group sessions
Environment	Normal system operation with an active internet connection
Artifact	Intaliq Mobile Application, Backend API, and Database
Response	The system retrieves the available sessions and displays their details
Response Measure	95% of session-list requests complete within 3 seconds under the agreed normal-load test conditions

2.4 Scenario for Quality Attribute: Usability

Element	Description
Source	A new Intaliq user
Stimulus	Attempts to create a personal fitness goal
Environment	Normal mobile application operation
Artifact	Intaliq Mobile Application user interface
Response	The application guides the user through selecting an activity, entering a goal, and saving it with a confirmation message
Response Measure	At least 90% of first-time test users successfully create a fitness goal without assistance

3. Architectural Style

3.1 Selected Architectural Style

In Intaliq, Client Server architecture is implemented, wherein the mobile application will be the client communicating with a centralized backend server.

The client will provide the user interface for users to interact with the application's features such as activity monitoring, goal setting, session booking, partner matching, and AI Chatbox.

The application server will take care of user interaction with the system and processes requests; handles authentication and authorization; application level operations; and communicates with the database and other external services.

The database will be used to store information about users, coaches, activities, goals, sessions, and partners. The backend will communicate with an external AI service to enable the features of AI Chatbox.

3.2 Justification

Client-Server Architecture was chosen as a design strategy due to its clear division of mobile application and system's backend processes.

Here are some factors supporting the use of Client-Server Architecture for Intaliq:

- Data Centralization: User information, user actions, fitness goals and sessions can be controlled via the backend and the database.
- Security: Authentication and authorization processes can be performed on the server to prevent access to user information.
- Maintability: Backend and mobile applications can be maintained independently from each other.
- Performance: Data processing and database interactions will be done by the backend while the mobile application will interact with users.
- Integration: The architecture allows performing data exchange with third-party services such as AI Chatbox.

In general, Client-Server Architecture allows organizing the Intaliq system in a logical way.

3.3 How We Will Implement It in Intaliq

Implementation of Client-Server Architecture in Intaliq will be based on four key components:

1. Client: Intaliq Mobile Application

This is the application that will enable interaction of users with the system by performing the actions of logging their activities, setting goals, booking sessions, finding partners and communication with the AI Chatbox.

2. Application Server: Intaliq Backend

Backend receives requests for operations performed by the mobile application via HTTPS protocol. These operations include:

- Authentication and Authorization
- User Management
- Activity Management
- Goal Management
- Session Management
- Partner Matching
- Integration with the AI Chatbox

3. Database: Intaliq Database

This database stores the information needed for the application operations including:

- User Information
- Coach Information
- Activity Information
- Goal Information
- Session Information
- Partner Information

4. External AI Service

The back end interacts with the external AI service through an API connection that utilizes the HTTPS protocol. The AI service provides support for the AI Chatbox and enables the user to ask questions and get fitness tips.

Communication Process

1. User executes an action via the Intaliq Mobile Application.
2. The application passes on the request to the back end server through HTTPS.
3. The back end authenticates the request and processes it as necessary.

4. The back end communicates with the database or the external AI service when required.
5. The back end responds to the application with the outcome of the process.
6. The application shows the outcome to the user.

The above design is in line with the component and deployment diagrams discussed in Intaliq's Software Architecture Design document.