Iteration 3: Addressing Quality Attribute Scenario

Goal

To support quality attributes scenarios and address any additional concerns that might occur.

Step 2: Establish Iteration Goal by Selecting Drivers

In order to support our designs with our third iteration, our goal is have sufficient functionalities that will support our QA-5. Security will have our heavily encrypted server, store user information.

Step 3: Choose One or More Elements of the System to Refine

In this iteration, elements to be refined in this iteration are the elements that will be refined are The elements that support our QA scenarios. Which will be the Database Server and DataAccess.

Step 4: Choose One or More Design Concepts That Satisfy the Selected Drivers

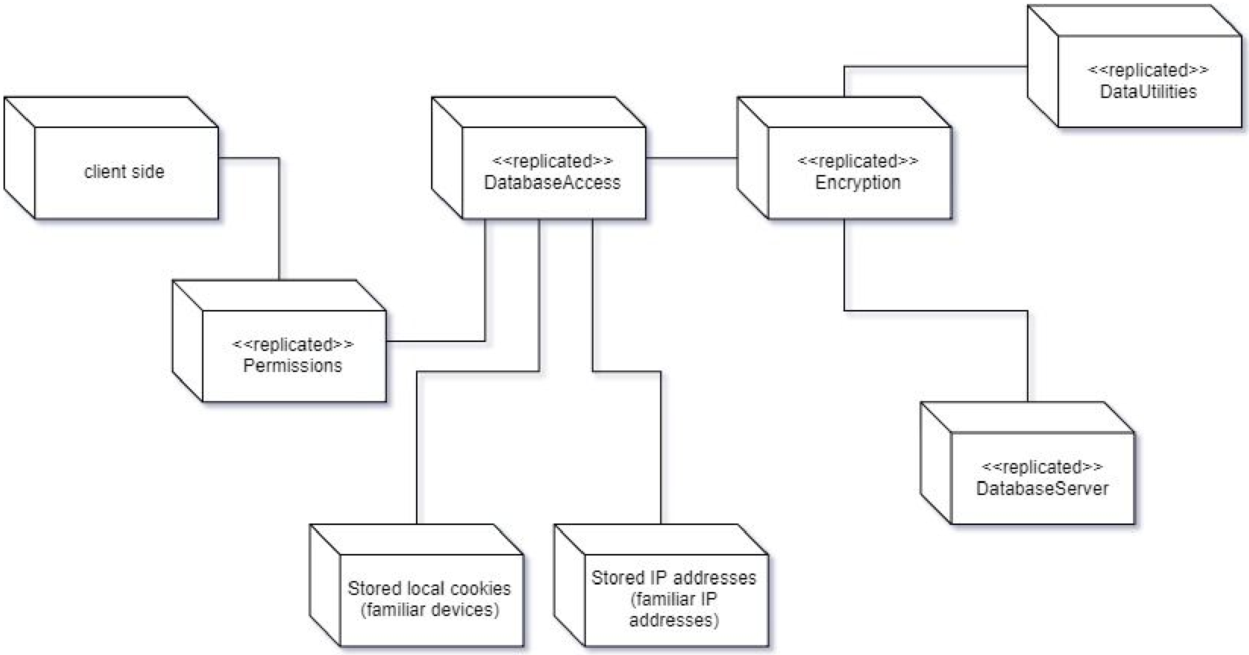
Design Decisions and Location	**Rationale and Assumptions**
Introduce the active redundancy tactic, done by replicating the database server and other components	We will replicate the critical elements of the system. This way, we can do all our tests on the replica without affecting the functionality of our original elements.
Introduce an element from the message queue	Emphasizing on the DataAccess that are placed in the message queue. The element will be such that whenever a sign in is done from another device (other than the usual, where the local cookies are stored), there will be a message sent to the person's communication (e-mail) to notify them of this unfamiliar log in.

Step 5: Instantiate Architectural Elements, Allocate Responsibilities, and Define Interfaces

Design Decisions and Location	Rationale and Assumptions
Deploy message queue on a separate node	Deploying the message notification on a separate node will ensure that no information is lost in case the application crashes. This node is duplicated using active redundancy as stated in step 4.
Use active redundancy and heavy encryption on the database access.	Since our database has a single point of vulnerability by having all user information stored in a database server. We will place heavy encryption on top of the server to prevent any malicious attacks. The active redundancy will also reduce our operating cost and automate our critical maintenance actions.
Implement redundancy using technology support	With the technological support available, we are able to implement active redundancy without having to develop an alternative solution that will be much harder.

Step 6: Sketch Views and Record Design Decisions

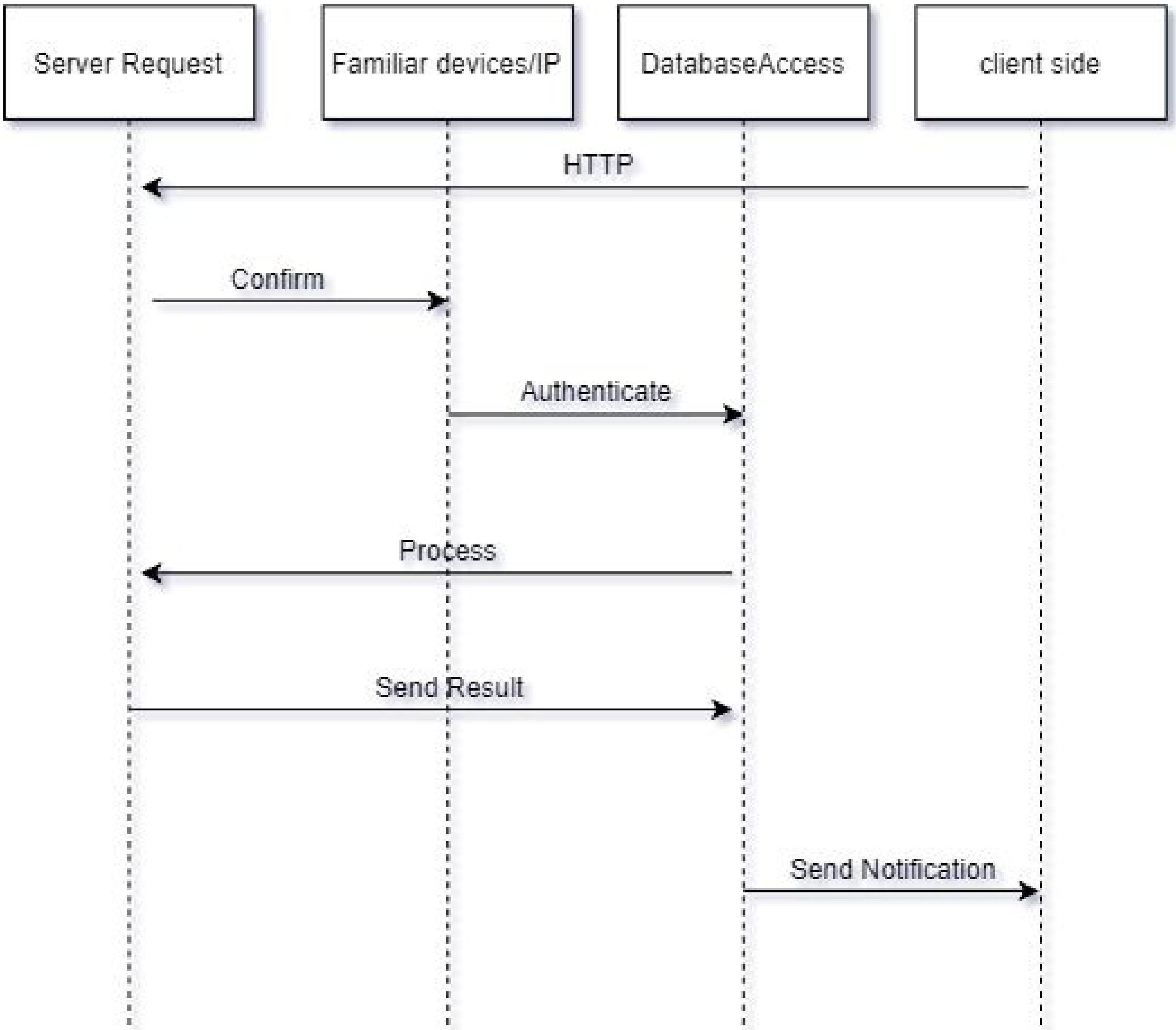
Deployment Diagram


client side
<<replicated>> Permissions
<<replicated>> DatabaseAccess
<<replicated>> Encryption
<<replicated>> DataUtilities
<<replicated>> DatabaseServer
Stored local cookies (familiar devices)
Stored IP addresses (familiar IP addresses)

For Elements that haven’t been introduced in Iteration 1 and 2

Element	**Responsibility**
Encryption	Stores user information into a secret code that has multiple layers of security level.
Stored local cookies	The database checks for the cookies that are stored in the user’s browser. This will let the database know if this browser (essentially the same device) is a first time log in with the specified account or not.
Stored IP addresses	The database checks for the IP addresses that have been used in the past recently with the specified account. This will let the database know if this is a familiar IP address. This with the familiar cookies, is another level of security that will enable an automatic message notification to the user.

UML Sequence diagram below shows how the stored familiar devices/IP affect the flow of logging into the system. Instead of directly going into the DatabaseAccess, it will have to go through multiple layers of checking and authentication to detect any unfamilar devices/IP. This mainly focuses on QA-5 and QA-1. While doing this will certainly improve on our main focus for this iteration, there will be added work that will be related to CON-3. The purpose of this diagram displays the workflow of how a message notification is processed and sent to the user.



Step 7: Perform Analysis of Current Design and Review Iteration

Not Addressed	Partially Addressed	Addressed	Designs Decisions Made During the Iteration
		UC-1	No relevant decisions made
		UC-2	No relevant decisions made
		UC-5	User Security and Permissions requirements were fully identified in step 6.
		QA-2	No relevant decisions made
		QA-5	The domain model allows security at every tier of the system. Specific implementation were also conceptualized in step 6. Further implementations were done in the third iteration.
		QA-6	No relevant decisions made
	CON-1		No relevant decisions made
		CON-2	No relevant decisions made
		CON-3	Additional functions of relational database were mapped and implemented.
		CNR-1	No relevant decisions made
CNR-2			No relevant decisions made
		CNR-3	EJS (Embedded Javascript)-complimented HTML and CSS were chosen for our front-end presentation, both of which are highly compatible with our team’s familiarity with Javascript. Further improvements were done. Additional designs were done for the third iteration.