Assignment 3: Opposition Report

Group 8

Name	Email	Swedish Id
Khan, Muhammad Mohsin Qamar	muqm22@student.bth.se	19980625T659
Hassan, Syed Ali	syal22@student.bth.se	19830417T674
Shahzad,Ghilmaan	ghgh22@student.bth.se	19950206T758
Ayyaz,Muawaz	muay22@student.bth.se	19960513-T771

Contents

Introduction:

This is a Software architecture evaluation report done by group 8 of the software architectural design of Group 9. The evaluation technique used in the report is Architectural Trade-Off Analysis Method (ATAM). The method steps 4, 6, 8 and 9 are used in the analysis of the architecture of the system.

Architectural approaches:

The team has used Micro services architectural pattern to provide the solution.

Micro services:

- Increased Development Time
- Performance Decrease

Key Quality Attributes:

- Availability
- Performance
- Security
- Usability

Identification of Architectural Approach

1. Reference Architecture is a micro services architecture and presented in Layered based system. The different types of services are implemented and shown in the respective layers
2. Design pattern is Micro services based architecture (to justify Availability and Performance)
3. Divide functionality into multiple services hosted into multiple hosting points
4. Provide a security service (OTIS gateway) in order to protect data from malicious attacks
5. Get all Data from external systems and dump in 2 databases: User information, Report within 15 minutes and make prebuild data to show user information when needed.
6. Customer Middleware service to get data from external system and store them in multiple databases using the business layer

Tactics:

SR No	Tactic	QA tactics	Module/component plan to implement
1	Exception handling	Availability	Logger Service
2	Removal from service	Availability	OTIS gateway
3	Heart beat	Availability	Data Layer
4	Increase Available resources	Performance	Business services
5	Caching of Data	Performance	Business Layer

6	Detect attacks and resist	Security	OTIS gateway
7	Separate User Interface	Usability	User service

Risk and Non-risks

Risks:

Risk ID	Description
R1	No solution written for providing API endpoints for external systems is missing, which is a risk for Availability
R2	Services while communicating with one another is through end-points which is not encrypted/secured.
R3	No backups of databases are defined in the system which causes a risk in availability if database connection fails
R4	System needs to be deliver within 6 months. Considering OTIS portfolio, the implementation of micro services is not causes overhead for lower number of users.
R5	OTIS gateway is the main entry point for the user to access the system for the users. This makes the failure of this service a risk as users cannot access the system

Non-Risks:

Non-Risk Id	Description
NR1	Stack Overflow attacks and Code Injections are handled. Alter will be generated to the system Admins if store issue occurs.
NR2	Malicious logging attempts and Spam IP blocking is done which improves security. To many logging attempts will trigger an alert to the system admin and system block the malicious IP address
NR3	Data Caching approach will improve the performance and availability of the system. Through caching data which don't change frequently, can cached which reduce the amount of database queries
NR4	Providing logging service as separate service to provide availability to the system. System Admin can view the error, security and user logs if any service of the system occurs failure or attacked by hacker
NR5	System uses 3rd party services (Elastic search etc.) to save Important logs of the system in encrypted form to handle the security threats

NR6	Notification service is added into the system whose availability is critical in provides error, security and alter notification to respective stakeholders. By separating it, maintenance of the service will be easy

Risk Themes:

Team review the quality aspects and the identified risks, non-risks of the system. The following the risk themes identified from the inputs mentioned above:

- The implementation micro services architecture will be challenging for the current team and external hires to develop in such short amount of time. Service Oriented architecture is difficult and complex to implement and overall system testing is difficult.
- Adding additional security to the system will impact the usability and performance of the system. Dividing work into multiple service will yield additional latency which will impact performance.
- OTIS system is relatively small and having multiple databases will add additional time for querying data from multiple services. Different types of databases will add complexity to the system as different maintenance and learning overheads will occur

Sensitivity Points:

- No backup exits for databases of services for User information Database and report database
- Data Layer is sensitive point as all user and report information is stored in the system. Querying one database with other will impact the security aspect.
- No Databases backup tactic is provided which makes data layer as sensitive
- OTIS Gateway Service is critical for entire system. The service communicates with the external system will fetch the data and deliver the data to respective business objects

Trade Offs:

1. Additional micro services added to increase Availability will impact Performance due to increased communication overhead between system components

2. Added security will impact Usability from User perspective. Multiple checks and invalid login attempts from correct user could potentially impact bad user experience.

3. The importance of data and data change frequency determines which data would be beneficial to cache. As cache takes up additional storage in the system, micro services load could impact the extra overhead of space and caching computation every time external system provides data in the system.

4. Adding Security service in the system will impact the performance of the system.

Evaluation Opinion/Comments:

As the architect team has identified suited quality attributes which fulfils the system requirements but as the evaluation team we consider this system to be more suited monolithic pattern/architecture as concern of less cost and time constrain which is 6 months. Security is also considered to be one of the 3 main quality attributes whereas any sensitive information is not being present in the system as our team opinion. [2]

Micro services have issue with the testing and validation of the system and prolonged issue in the system is a concern of availability. Caching of the system is great approach for performance and availability but caching all system data will add extra computational overhead to update the cached data every time data changes occur.

Micro services are better in large scale applications. Our number of users and application is relatively small so additional cost of resources (Time, budget, human resources) are required to implement the system in this architecture. [1]

Questions:

- If user want to see the tracking progress of the order. Does your system provide the functionality for that?
- Considering Security as key QA, 1 hour response time for Malicious Attempt seems much prolonged, should be in a few minutes.
- Why Micro services? Does this system require the scalability of micro services as this system has less clients and more data to handle from external system.
- Usability is mentioned in the system what tactics are used by you as there is no tactic mentioned in the document.
- Restful services for external client not used why

- No Databases backup plan is incorporated with the OTIS system
- Report Update Service is unclear, how this will work and what kind of Report Update is intended by User Account Service?
- How much data will be cached in the system

References

[1] magello, "Microservices — Good or Bad?," magello, 13 Janurary 2018. [Online]. Available: https://magello.se/microservices-good-or-bad/.

[2] J. Nemer, "Advantages and Disadvantages of Microservices Architecture," 13 November 2019. [Online]. Available: https://cloudacademy.com/blog/microservices-architecture-challenge-advantage-drawback/.