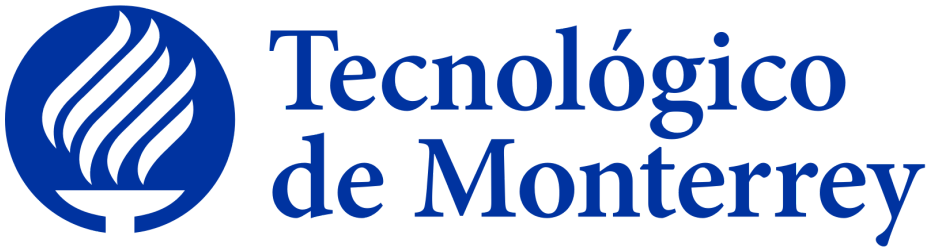
CHA Finding Explorer

First Deliverable



Semestre I

Information Security

Mahdi Zareei

Team 3:

Ari Valenzuela

Nataly Hernández

Emilio Castillo

Carolina Pérez

Risk Management Framework(RMF)

Understand the Business Context

CHA Finding Explorer is currently a real-time/ historical analysis tool that generates a JSON report for the Oracle RAC users. Oracle RAC (Real Application Cluster) is used by users that need a good way to create a cluster of servers or databases, in a way that allows availability and scalability.

Currently the information generated by CHA is only available in the JSON file it provides to its users, but a JSON file can be tedious to read through and it's hard to obtain meaningful information from it if you don't have a lot of time to analyze it properly.

Our main objective is to create a way in which the users are able to visualize the information from the JSON file in a clear, attractive and simple way, in which it's easy for them to find the information they need to take the corrective or preventive actions in their clusters.

A solution that will make this objective possible is the creation of a web application that will be locally used by each user, where the only thing they need to provide is the JSON file generated by CHA and the web application will create graphs, dashboards and different sections for the user to access the information which will be presented in an attractive way.

Identify the Business and Technical Risk

Business Risks

- Unexpected changes on the project requirements and scope: [Huge impact on the result of the project, not being able to achieve those changes, delays]
- Lack of communication with the stakeholders: [Requirements and expectations of the project may not be satisfied]
- Loss of project/material code: [Huge delay impact, unsatisfied stakeholders, code not testable nor readable, system failure]
- Project cancellation. [Waste of resources, bad reputation, lack of productivity]
- Failure to display the information in an attractive way [Not meeting the main objective of the project]

Technical Risks

- JSON file information processed incorrectly. [This could lead to unsatisfied stakeholders since they wouldn't be getting the information they need from the reports, failure in data integrity]
- Lack of validation while processing the JSON. [Someone could crash the application if they use a wrong JSON which affects the availability or the application may process it and display wrong information which affects the integrity]
- Lack of technical knowledge to properly manage Oracle Jet components. [This could lead to delays, bad reputation and decreased productivity]
- Not thinking about edge cases. [This could lead to application crashes which affect the availability of the application if the application isn't able to handle certain specific cases]
- High load times due to a bad or inefficient processing of the JSON [This could lead to unsatisfied stakeholders]

Synthesize and Rank the Risks

LEVELS OF IMPACT:
The impact for each risk is divided in 3 levels, which are:

8.- For the risks that could cause an impact that wouldn't allow us to meet a big amount of the requirements, some of the main requirements, the deadlines for the project or cause big deficits in the budget.

4.- For the risks that could cause big enough delays to be noticable, that could provoke a failure to meet a few requirements or a financial loss.

2.- For the risks that can be solved easily and would only cause minimal delays, a minimal financial loss or failure to meet one not important requirement.

LIKELIHOOD:
This represents the probability of the risks happening in the project, it is divided in 3 levels:

- 1 - The probability of the risk to occur during the project is really low.
- 2 - The probability of the risk to occur during the project is medium.
- 3 - The probability of the risk to occur during the projects is high.

RISK LEVEL:
Likelihood * Level of Impact = Risk level

Rank	Risk	Impact	Likelihood	Risk Level	Explanation
1	JSON file information processed incorrectly	8	3	24	This risk would cause huge problems overall because it's hard to identify which part of the code is not processing the JSON correctly and since the

					JSON is the main part of it, all the information would be wrong and the webapp wouldn't be working.
2	Unexpected changes on the project requirements and scope	8	2	16	This risk would cause not accomplishing all the requirements as expected.
3	Failure to display the information in an attractive way	8	2	16	This is basically one of the main objectives of the project. We need to pay close attention to the UI/UX features of the project.
4	Loss of project/material code	8	2	16	A loss of material is probably not going to happen, but in case it does, the project would not be testable and therefore, it would be delayed.
5	Lack of validation while processing the JSON	4	3	12	Validating inefficiently while processing the JSON would cause errors when generating the report.
6	Lack of communication with the stakeholders	4	3	12	It is possible that we have communication problems with stakeholders but the impact is not as high because it wouldn't completely stop the project
7	Not thinking about edge cases	4	3	12	These edge cases can cause the application to crash which would affect certain users.
8	High load times due to a bad or inefficient processing of the JSON	4	3	12	It would cause availability problems which won't affect the users too much but it may annoy them and cause complaints.
9	Lack of technical knowledge to properly manage Oracle Jet components	4	3	12	In this case it is very possible for us to get stuck when developing the app, but the impact is low because there's a lot of documentation.

10	Project cancellation	8	1	8	A project cancellation would have one of the highest impacts on the project, but it is somewhat unlikely to happen.

Applied Technologies Vulnerabilities

The main technology we are going to be using is Oracle JET which is a javascript framework. Based on the Oracle JET documentation the main attacks it can suffer are: Cross Scripting Attacks (XSS), Packet Sniffing, Clickjacking and Authentication vulnerabilities since it uses OAuth for authentication. These vulnerabilities shouldn't affect our application since it's going to be stored locally so someone would need to have access to the computer in which it's executed to perform these attacks and more.