LOW-LEVEL DESIGNS DOCUMENT

Hubba - Authorization

Version 1.0
Prepared By: Development Hell
Class: CECS 491-04
Date: December 14, 2022

Github Repository**:**
https://github.com/DevelopmentHellaHell/SeniorProject

Team Leader
Kevin Dinh
Members
Garrett Tsumaki
Bryan Tran
Jett Sonoda
Tien Nguyen
Darius Koroni

Revision History

Version	Overview	Date
v.1.0	Initial LLD	12/4/2022

Low-Level Design

User story: As a user, I can access non-anonymous features, view, and data within my authorized permission context.

Preconditions:

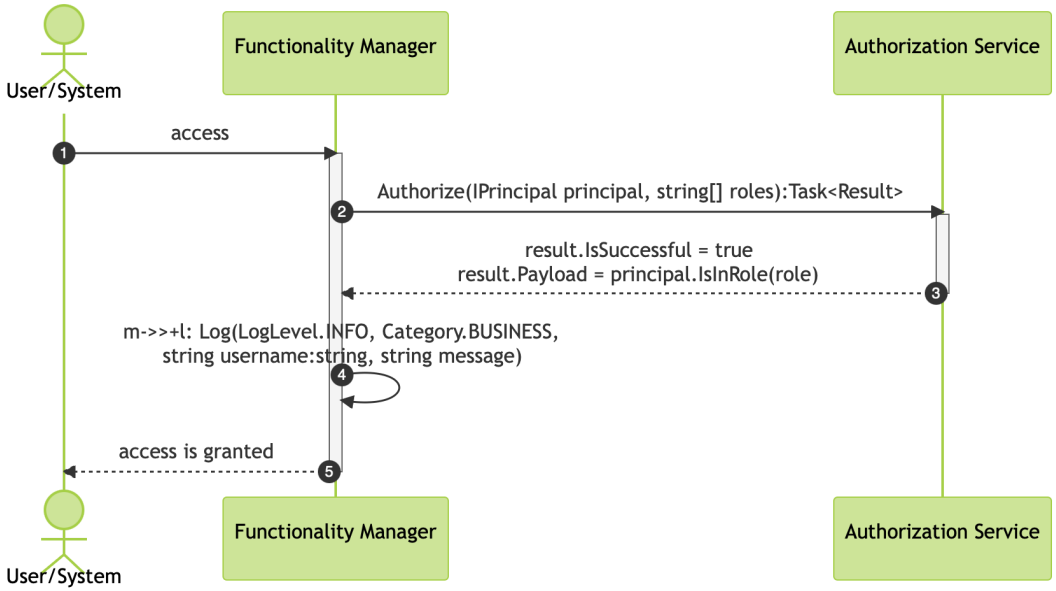
1. User must have an active authenticated session.
2. User's account must be enabled

Business Rules:

- Our program is using a role-based authorization access system. Each user will have a specific role tied to their thread and they will be granted full or no access to a component based on the role requested on component access. After reviewing their specific role and the required role(s) of the component, the user will be granted or denied access to the component.
- Unauthenticated users can only access system features which allow anonymous roles or do not require knowledge of the user
- A user attempting to access an unauthorized portion of the system will be logged and prevented from doing so
    - Unauthorized users attempting to access another user's protected system data
    - Unauthorized users attempting to execute another user's actions
    - Unauthorized users attempting to view another user's PII
    - Unauthorized users attempting to access a system view restricted from them
- Any changes to a user's access control will be active the next time the user authenticates their account

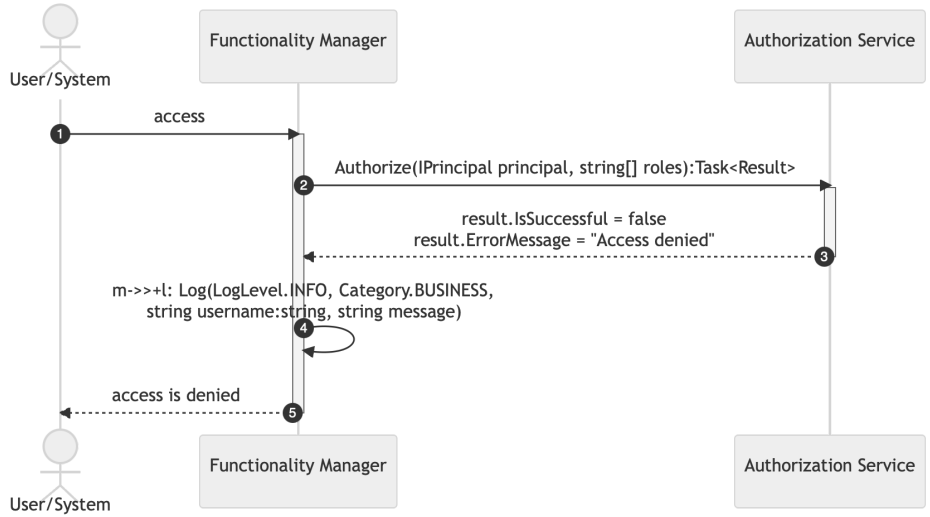
Successful Use Case(s)

1. Access permission is granted based on user's role.
    a. User attempts to access a protected functionality within authorization scope. Access is granted to perform functionality.
    b. User attempts to access protected data within authorization scope. Access is granted to perform read operations.
    c. User attempts to modify protected data within authorization scope. Access is granted to perform write operations.
    d. User attempts to access protected views within authorization scope. Access is granted to the view. User is automatically navigated to view.

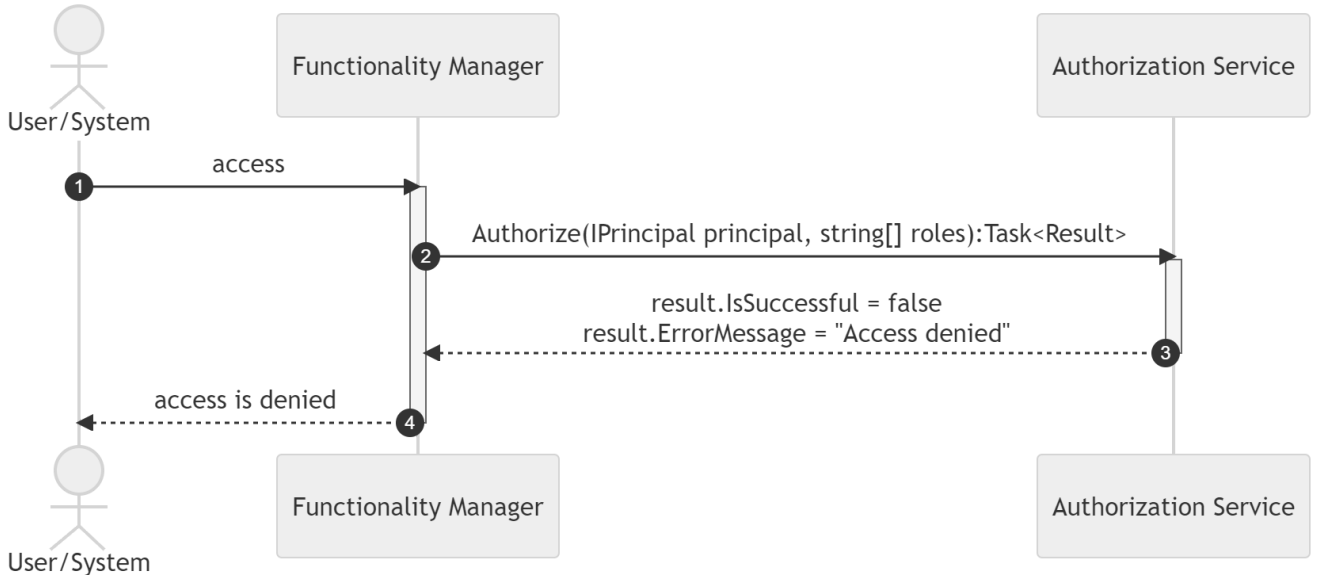


Failure Use Case(s)

1. Unauthorized access is not recorded by system when authorization fails. A system log of failure is attempted.



2. User attempts to access or modify a protected component or view outside of the authorization scope based on the component's required role(s). Access is denied and a system message displays "Access denied".



References

This document elaborates client's requirements[1] for one of the product's core components - Authorization.
The requirement is designed based on the system architecture provided in Business Requirements Document[2].
The diagrams included in this document utilized diagrams.net[3] and Mermaid.js[4].

[1] Client's email, 10/17/2022
[2] Business Requirements Document, URL:
https://github.com/DevelopmentHellaHell/SeniorProject/blob/b43182c4076d471ef03520052675f1f88371dafd/docs/HL%20Design/DevelopmentHell%20HLD%20v1.2.pdf
[3] https://www.diagrams.net/
[4] https://mermaid-js.github.io/mermaid/#/