Use Case Examples

Brief:

Add software license. A request is received by the system. The security token is extracted from the request. The Authorization Server is called passing the security token. The Authorization Server credentials response is received. If the request is from a Manager, the a new license record is added to the repository with the data fields from the request.

Casual:

Add software license.
Main Success Scenario:
A request is received by the system. The security token is extracted from the request. The Authorization Server is called passing the security token. The Authorization Server credentials response is received. If the request is from a Manager, the a new license record is added to the repository with the data fields from the request. A License Added response is returned to the requester.

Alternate Scenarios:
If the Authorization Server is not reachable or times out, , an error response is returned to the requester.

If the security token is not valid or expired, an error response is returned to the requester.

If the requester is not a Manager, an error response is returned to the requester.

If the repository insertion is not successful, an error response is returned to the requester.

Full:

Use Case UC1: Add software license.
Scope: Software License Tracking Application
Level: User goal
Primary Actor: Manager

Stakeholders and Interests:
Manger – Wants the ability to add a new Software License.
Legal – Wants to track existing Software Licenses
DBA – Wants to ensure the Software License database maintains integrity

Preconditions: The Authorization Server is accessible. The database is accessible.
Postconditions: A new Software License has been added to the database. A response has been returned to the requester.

Main Success Scenario:

1. A request is received via the REST API with a POST value
2. The system validates the format of the request
3. The system extracts the security token from the request
4. The system creates a REST API call to the Authorization Server including the security token
5. The system sends the HTTP request the Authorization Server
6. The Authorization Server returns a response
7. The system extracts the title value from the Authorization Server response
8. The system build a new Software License record, including the details from the original request
9. The system calls the Database server to insert the new Software License record
10. The system create a response for the original incoming request
11. The system returns the HTTP response to the original requester

Extensions:

2.a If the incoming request is not valid
    1. an error message is generated
    2. returned to the requester

3.a If there is no security token in the incoming request
    1. an error message is generated
    2. returned to the requester

6.a If the response from the Authorization Server is an error
    1. an error message is generated
    2. returned to the requester

7.a If the request is not from a Manager
    1. an error message is generated
    2. returned to the requester
9.a If the Database server returns an error
    1. an error message is generated
    2. returned to the requester

Special Requirements:
The Authorization Server has a REST API accessible via an HTTPS connection
The Authorization Server has a sub three second response

Technology and Data Variations List:
The Authorization Server response is in the form of:
{'user_id':31,'name':'Jonathan Earl','department':'Sales',
'title':'Manager',’location’:’Dallas’,'exp':1760478912}
The Database record contains:

- the location assigned for static licenses such as a server license
- the employee assigned for individualize licenses
- the license type
- the date the license has been added to system
- the date the license expires

Frequency of Occurrence:
On demand

Open Issues:
Are the software names standard, is there a definite list
Does the department the Manger is a member of matter?