OOM GROUP PROJECT

TOPIC:-DOCUMENT STORAGE AND RETRIEVAL

GUIDED BY:-

PROF.O.P. VYAS SIR

PRESENTED BY:-

1.KUNAL PENDE IIT2019187

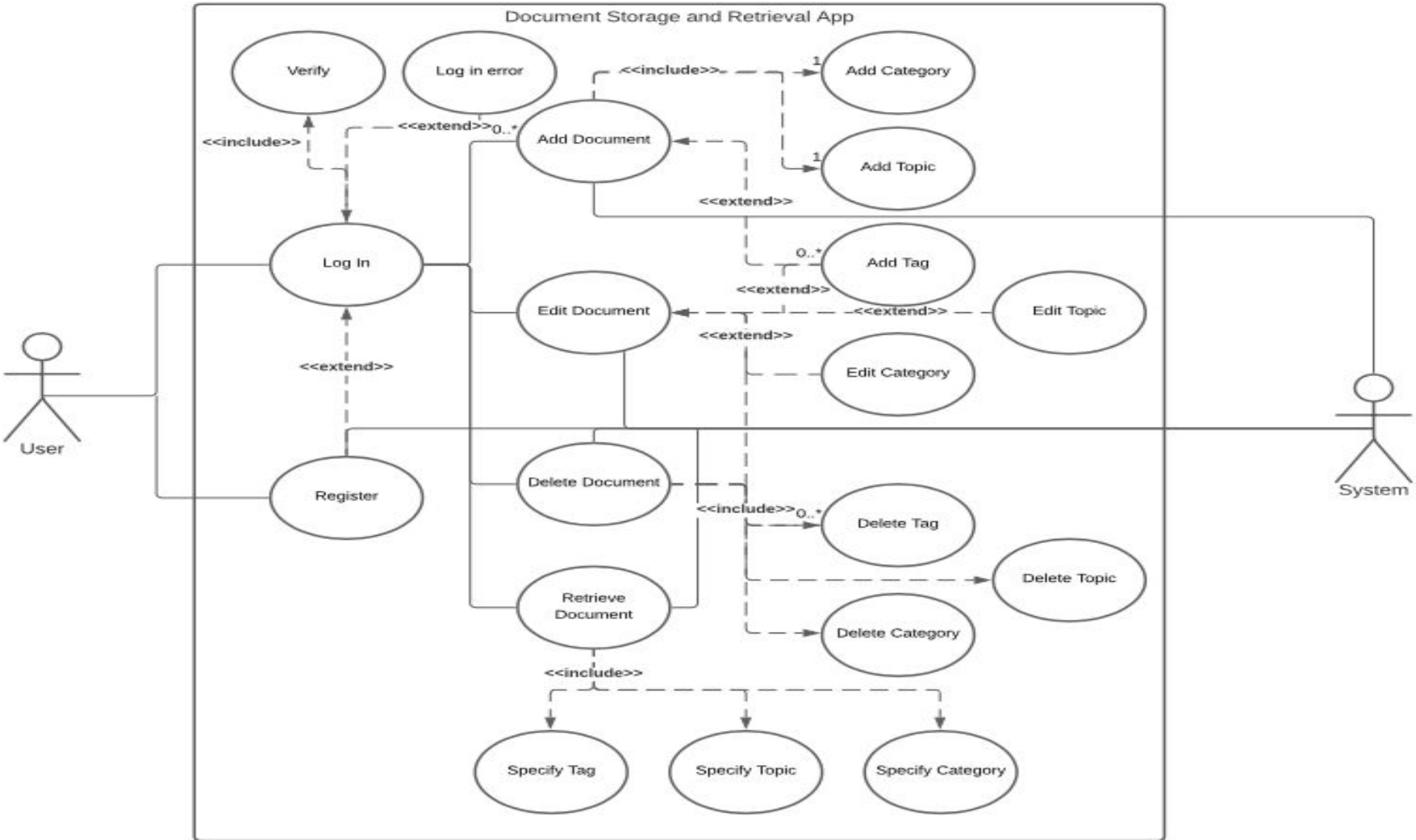
2.DEBASHISH DATT IIT2019188

3.RAHUL ROY IIT2019194

4.RITU MUMU IIB2019025

USE CASE DIAGRAM DEFINITION:-

A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.


Document Storage and Retrieval App
Verify
Log in error
<<include>>
1
Add Category
<<extend>>
0..*
Add Document
<<include>>
1
Add Topic
<<extend>>
Log In
0..*
Add Tag
<<extend>>
Edit Document
<<extend>>
Edit Topic
<<extend>>
<<extend>>
Edit Category
User
<<extend>>
System
Register
Delete Document
<<include>>
0..*
Delete Tag
Delete Topic
Retrieve Document
Delete Category
<<include>>
Specify Tag
Specify Topic
Specify Category

CRC DIAGRAM DEFINITION:-

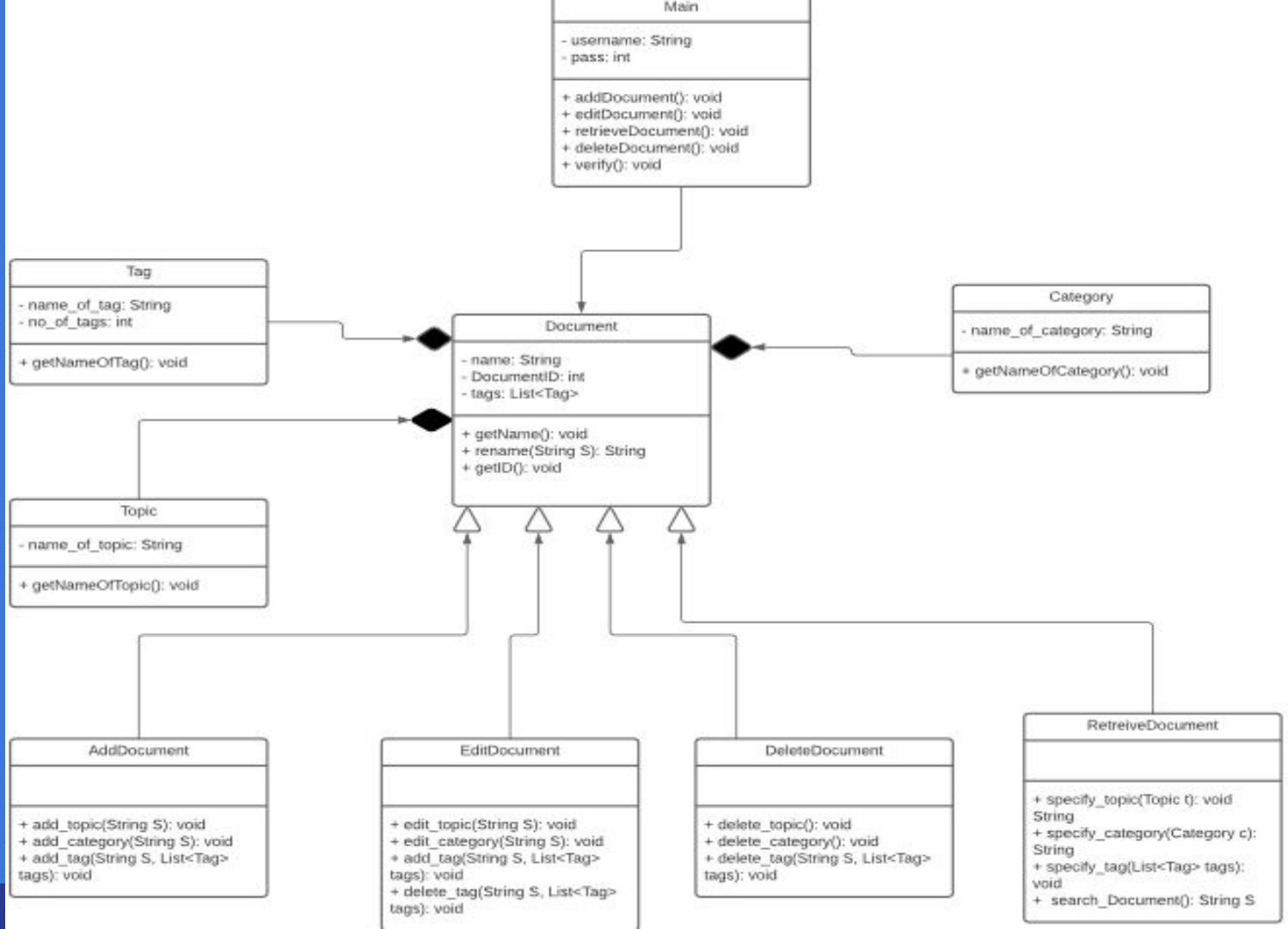
Class-responsibility-collaboration (CRC) cards are a brainstorming tool used in the design of object-oriented software. To create a CRC card, you can begin by writing out a scenario which identifies the major actors and actions which the actors do. Only write out actions and actors specific to that particular scenario. Nouns should turn into the classes of the card, verbs typically turn into the responsibilities of the card, and collaborators are the other cards with which the card will be interacting.

Main	
- Responsible for verification of user (login) - Responsible for choosing between options of actions on Document (add/edit/delete)	- Document - AddDocument - Edit Document - DeleteDocument - RetreiveDocument
Document	
- Responsible for holding data - Responsible for having tags, topic and category - Responsible for actions on documents	- Tag - Category - Topic
Tag	
- Responsible for attaching tags to a document - Responsible for holding record of each tag for each document	- Document
Category	
- Responsible for defining the category of each Document	- Document

Topic	
- Responsible for attaching a topic to each document	- Document
AddDocument	
- Responsible for adding topic,category and tags for each document added	- Document - Topic - Category - Tag
EditDocument	
- Responsible for updating and changing topic, category and tags for each document	- Document - Topic - Category - Tag
DeleteDocument	
- Responsible for deleting all contents of a document	- Document - Topic - Category - Tag
RetreiveDocument	
- Responsible for retrieving a document by specifying it's topic, category and tags	- Document - Topic - Category - Tag

CLASS DIAGRAM DEFINITION:-

The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.


Main
- username: String
- pass: int
+ addDocument(): void
+ editDocument(): void
+ retrieveDocument(): void
+ deleteDocument(): void
+ verify(): void
Tag
- name_of_tag: String
- no_of_tags: int
+ getNameOfTag(): void
Document
- name: String
- DocumentID: int
- tags: List<Tag>
+ getName(): void
+ rename(String S): String
+ getID(): void
Category
- name_of_category: String
+ getNameOfCategory(): void
Topic
- name_of_topic: String
+ getNameOfTopic(): void
AddDocument
+ add_topic(String S): void
+ add_category(String S): void
+ add_tag(String S, List<Tag> tags): void
EditDocument
+ edit_topic(String S): void
+ edit_category(String S): void
+ add_tag(String S, List<Tag> tags): void
+ delete_tag(String S, List<Tag> tags): void
DeleteDocument
+ delete_topic(): void
+ delete_category(): void
+ delete_tag(String S, List<Tag> tags): void
RetreiveDocument
+ specify_topic(Topic t): void
String
+ specify_category(Category c): String
+ specify_tag(List<Tag> tags): void
+ search_Document(): String S

THANK YOU:)