in collaboration with

Softwarica College of IT & E-commerce | Coventry University

Assignment Title:
Coursework CW1 Assignment

Coursework Type:
Individual

Module Name:
STW104KM Enterprise Information System

Intake:
MARCH 2020

Submitted By
CU ID: 10735039
College ID: 200011
Name: Ritika Dongol

Table of Contents

Introduction

In this project we are asked to develop a software solution to effectively manage and conduct different Seminar. Actors are specified first and then use cases for respective actors to build a suitable use case diagram for the application followed by use case documentation, for the involvement of audience in seminar. Then, different classes and operations are defined by performing Natural Language Analysis(NLA) to build the system and created class diagram. Similarly, activity diagram is created, representing different activities, in order to find highest voted question. Finally, different entities with their attributes are identified and then normalization is performed up to 3NF(third normal form) and entity relationship diagram is made.

Use case diagram

Here, in the use case diagram, the system is Seminar Assistant whereas the actors are Audience, Organizer and Admin. The use cases for respective actors are:

For Audience:

1. Sign up
2. Login
3. Select department
4. View upcoming seminars
5. View description
6. Enroll
7. Register
8. Get token
9. Join seminar live
10. Enter token
11. Chat
12. Ask questions
13. Vote in polls
14. View feedbacks

For Organizer:

1. Register
2. Login
3. Create seminar
4. Give title of event
5. Set date and time
6. Set location
7. Add name of presenters
8. Add sponsors
9. Add schedule

10. Create and send invitation
11. Generate token
12. Update information
13. Start Seminar live
14. Manage seminar
15. Add/Remove attendee
16. Chat
17. View questions
18. Generate polls
19. Give feedback in high voted question

For Admin:

1. Login
2. Manage organizer
3. Paid organizer
4. Unpaid organizer
5. Edit details
6. Notify

The use case diagram for different actors is made using above information which is given below:

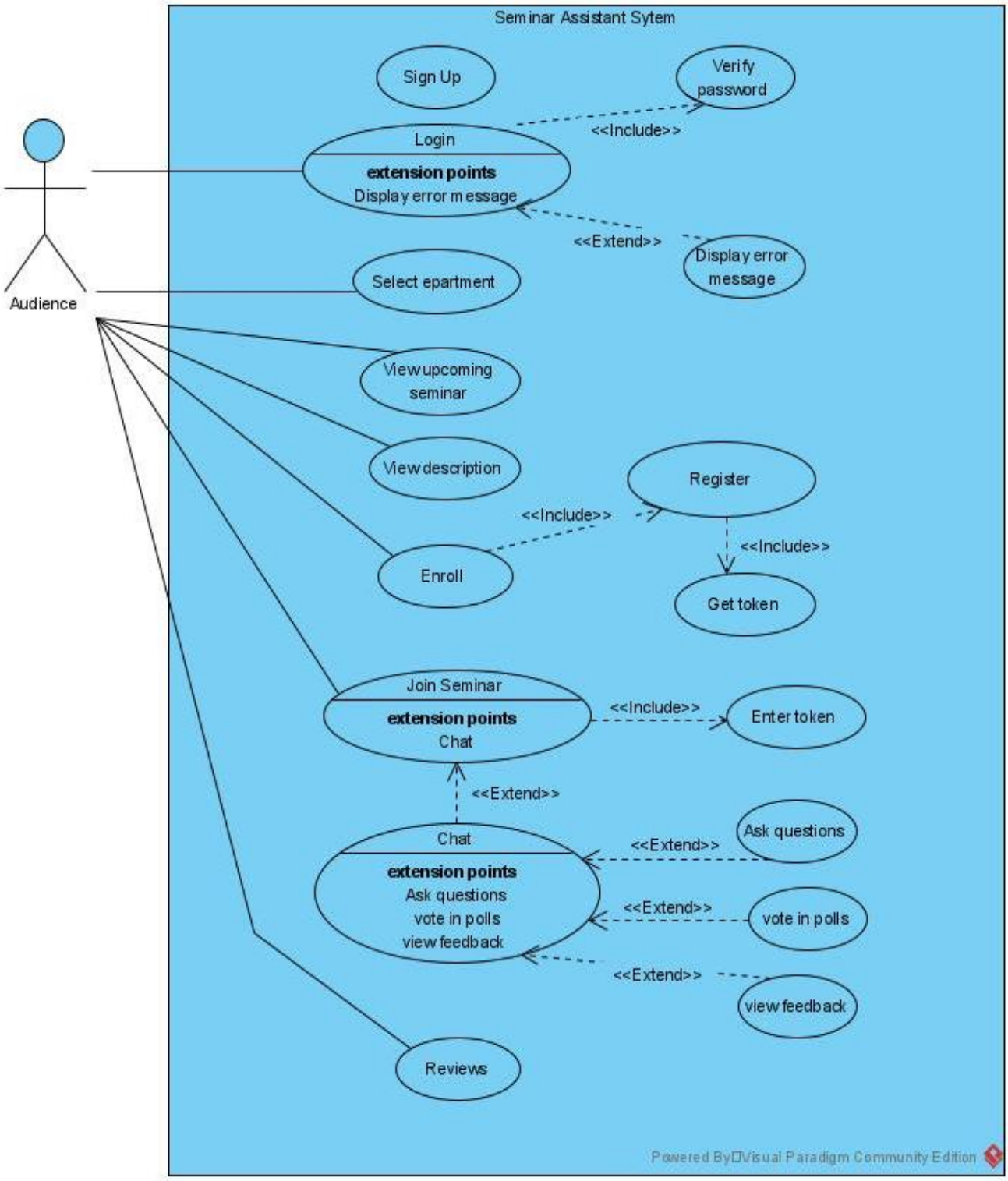

Figure 1: Use case diagram for audience

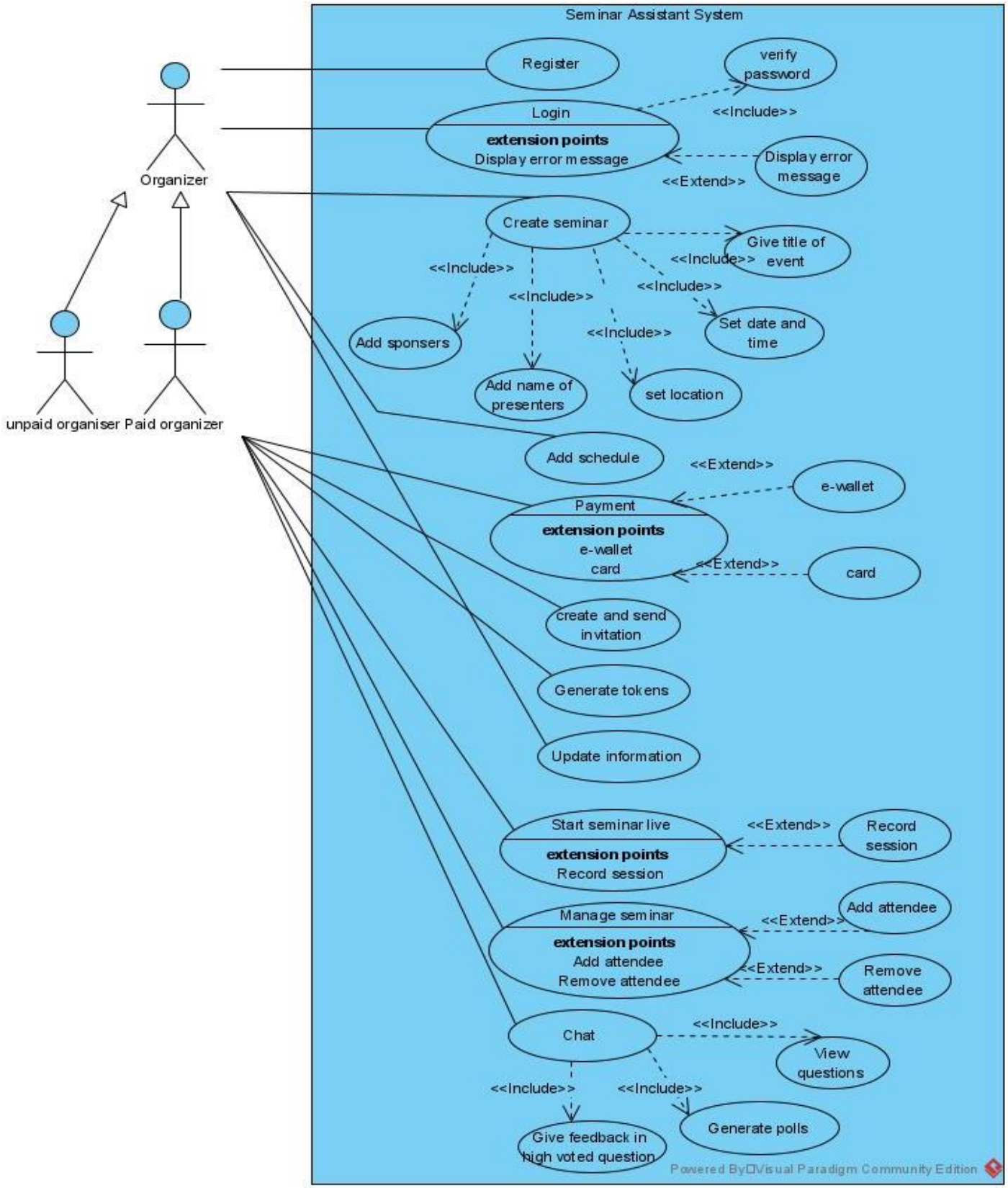

Figure 2: Use case diagram for organizer

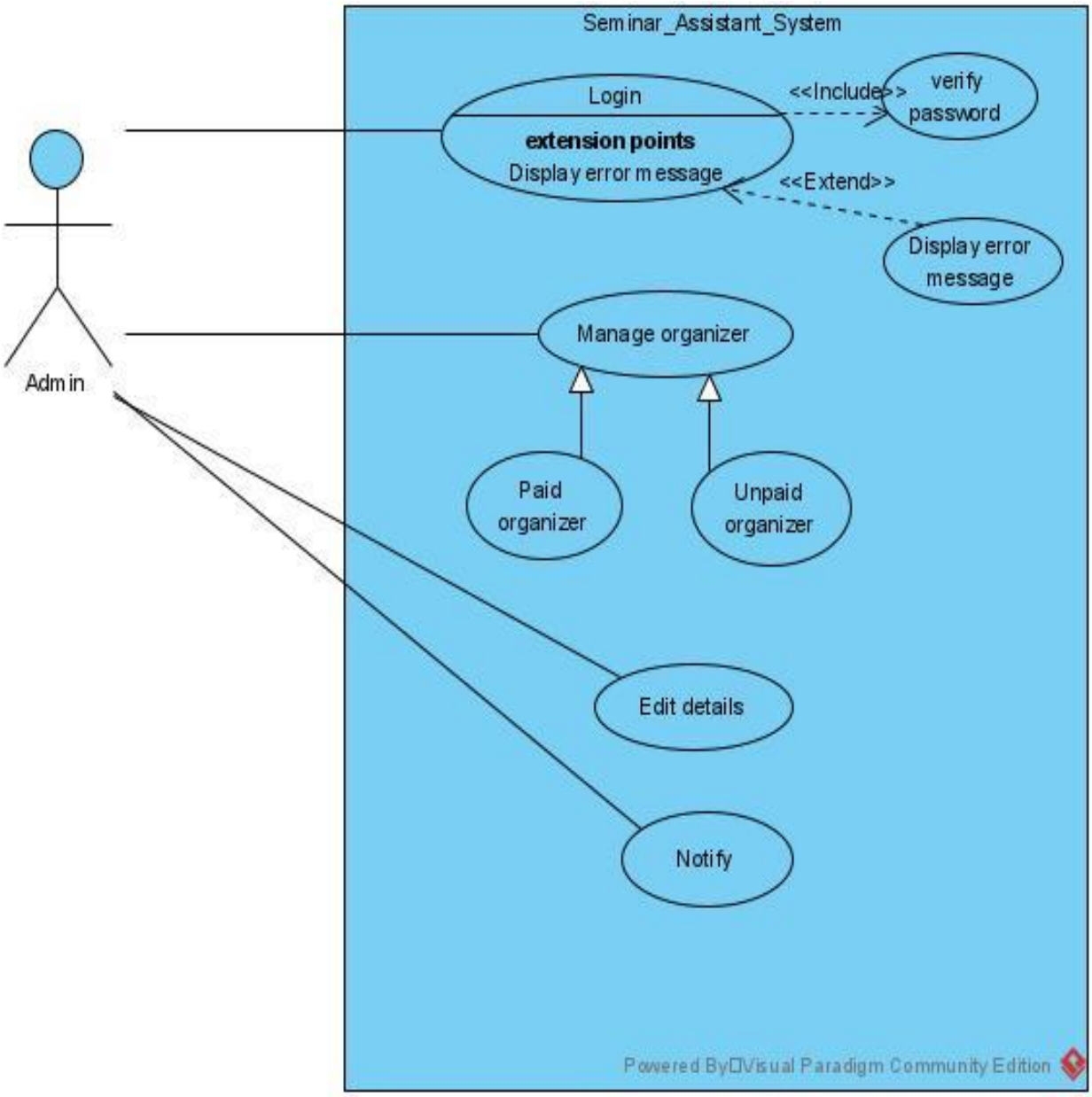

Figure 3: Use case diagram for admin

Use Case Documentation

Use case documentation for Audience module is given below:

Title	Audience Module
Brief description	Allows audience to participate in a live seminar
Actors	Primary actor: Audience Secondary actor: Server
Pre-Condition	1. User signs up into app as Audience 2. Audience logs into the system 3. Audience dashboard is displayed
Flow of event	1. Audience selects department 2. Audience views upcoming seminar 3. Audience views description 4. Audience selects 'Enroll' button 5. System requests user to register 6. Audience registers into the seminar 7. Audience gets token 8. Audience opens the seminar 9. Audience enters token 10. Audience joins the seminar 11. Audience asks question 12. Audience votes in poll
Post-Conditions	1. Audience views feedbacks 2. Audience gives reviews 3. Audience logs out
Alternative Scenarios	1. Audience is not able to login into system 2. Audience is not able to join seminar 3. Error in the information of seminar 4. Incorrect token

Table 1: Use case Documentation

Class Diagram

To identify the classes, attributes and operation of system, NLA(Natural Language Analysis) is performed. According to NLA, nouns can be considered as potential class names, verbs and adjective as attributes and methods respectively. The verbs and nouns present in the system is given below:

Nouns	**Verbs**
Seminar Audience Organizer Admin Seminar Assistant Poll Token Software company Presenter Name Seminar title, description Result Review	Create seminar Start seminar Join Edit seminar details Manage organizer Generate token Enter token Record seminar Ask question Generate poll, delete poll Vote Give feedback Count vote Edit seminar details Delete seminar, add seminar Edit profile Add/Remove attendee Manage seminar Enroll, register Give reviews

Table 2: Nouns and verbs

These nouns and verbs are then filtered removing synonyms, out of scope and ambiguous. Finally, the class diagram is created from the details analysed above, which is given below:

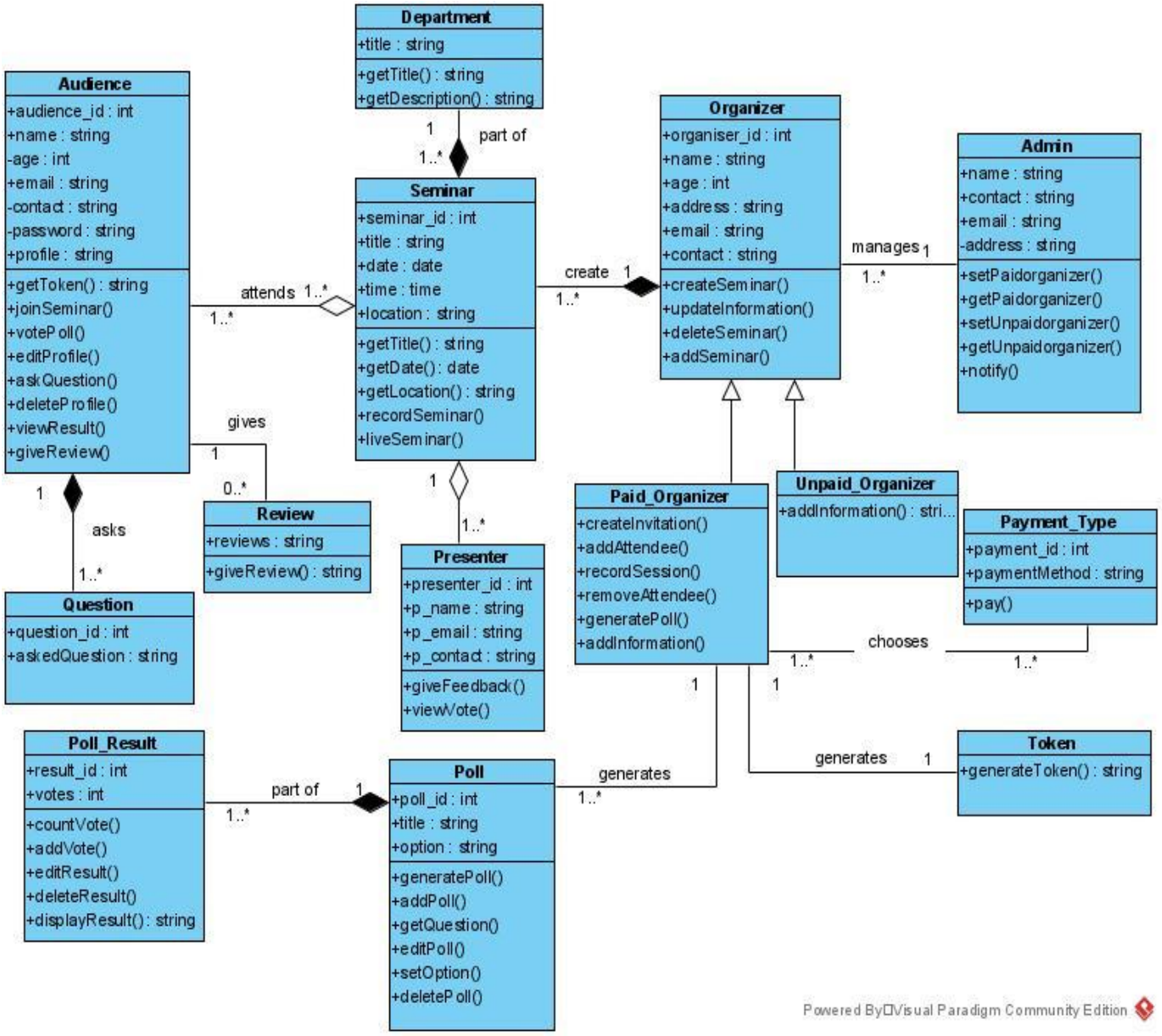

Figure 4: Class diagram

Activity Diagram

Activity diagram is created illustrating how the system finds highest voted question during feedback session in live seminar. The process to find highest voted question is given in below pseudocode:

Step 1: Start
Step 2: Start live seminar
Step 3: Audience ask questions in chat
Step 4: System generate polls of provided questions
Step 5: Audience gives vote in poll
Step 6: If audience has already vote then system informs user that they cannot vote otherwise go to step 7.
Step 7: System counts vote
Step 8: Wait for few minutes
Step 9: Check if vote is highest. If no, go to step 8 otherwise go to step 10.
Step 10: Presenter gives feedback
Step 11: If audience has still query then go to step 3 otherwise go to step 12
Step 12: Audience gives reviews
Step 13: Audience exits
Step 14: Finish

The activity diagram is created representing above pseudocode which is given below:

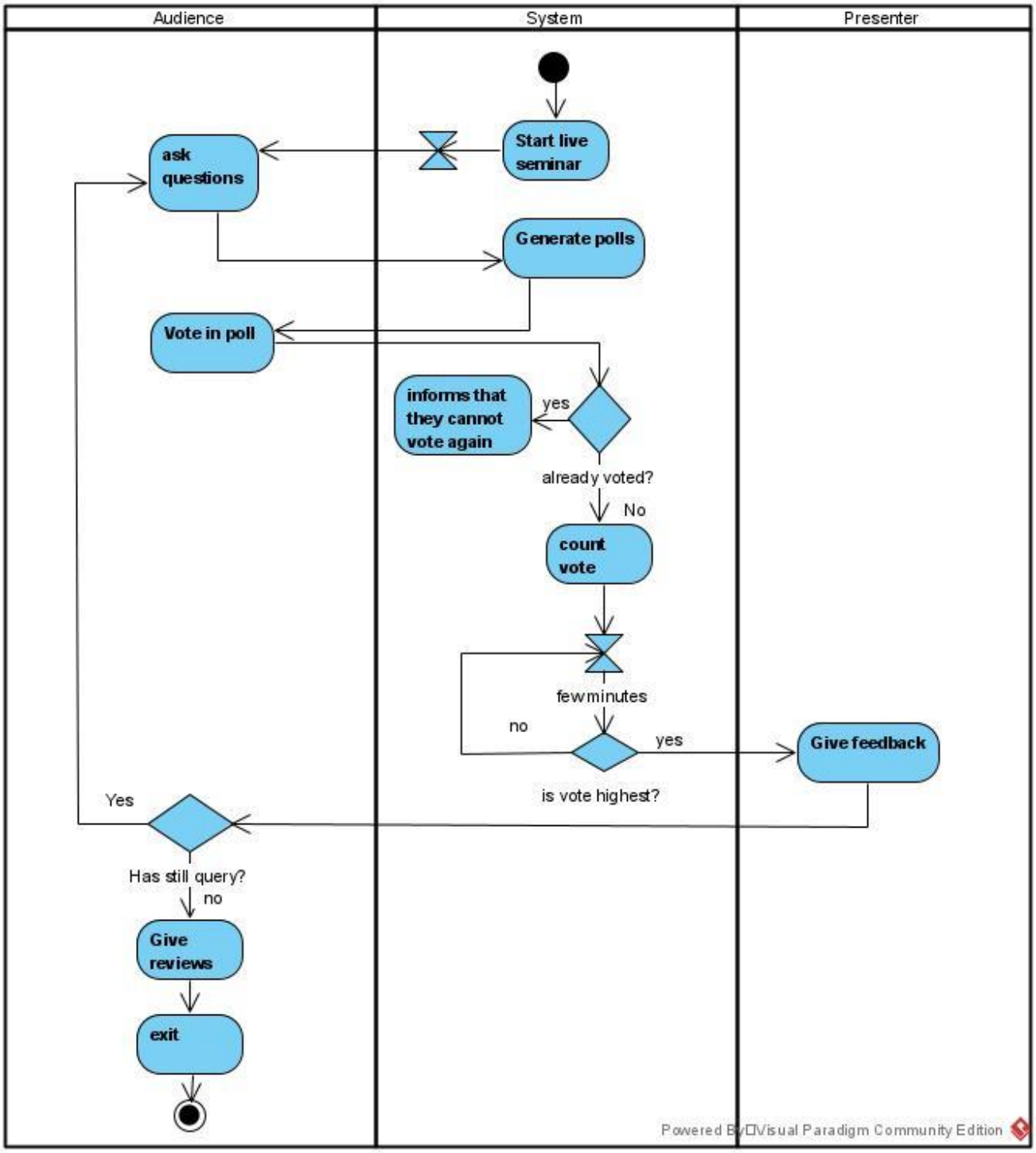

Figure 3: Activity diagram

Entity Relationship Diagram

ER diagram is a logical data model and is created using top-down approach in this system. It shows the relationship between different entity and the database structure. The entity and their attributes are defined first and normalized up to 3NF(3rd Normal Form) to remove anomalies. Then, entity relationship diagram is created illustrating relationship between them.

In first normal form(1NF), repeated attributes are removed from unnormalized data to remove possible redundancy, for example, separate entity is created for organizer. Similarly, atomic value is created in audience entity by breaking the name which is multi-value attribute. In second normal form(2NF), partial dependency is removed after the relation is made first normal form. New entity is formed between audience and seminar, organizer and payment method where primary key from their entity are made composite primary key. For third normal form(3NF), transitive dependency of the relation which is in second normal form, is removed. Separate entity for question is made.

Finally, the entity relationship diagram which is normalized up to 3NF is built which is given below:

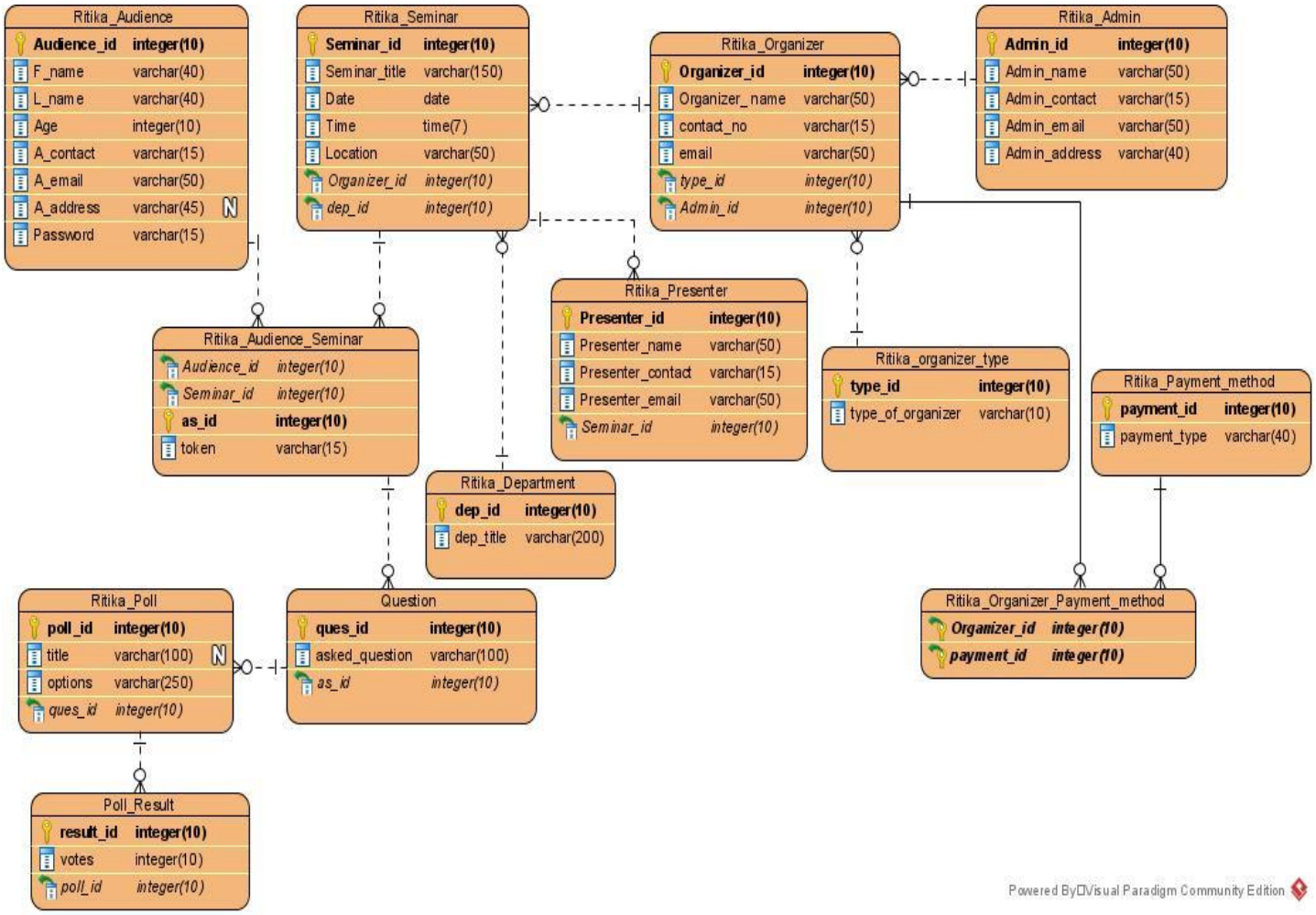

Figure 3: ER diagram

Conclusion

In conclusion, use case diagram for different actors involved in Seminar Assistant is created which represents behavior model. Similarly, use case documentation is made for audience module showing their stepwise events in participating seminar and class diagram is created performing NLA and showing their relationships and multiplicity of classes. Then, activity diagram is made illustrating how system finds highest voted question and ER diagram, showing how data is stored in database by normalizing up to third normal form.

References

1. *UML Association Vs Aggregation Vs Composition With EXAMPLE* (2021) available from <https://www.guru99.com/association-aggregation-composition-difference.html> [21 January 2021]

2. Kaula, R. (2007) *Normalizing With Entity Relationship Diagramming* [online] available from <https://tdan.com/normalizing-with-entity-relationship-diagramming/4583#:~:text=One%20of%20the%20ways%20an,tenets%20in%20relational%20model%20design.&text=Normalization%20utilizes%20association%20among%20attributes,table%20to%20accomplish%20its%20objective.> [21 January 2021]

3. *Swim Lane Diagram - Learn Everything About Swimlane Diagrams* (2021) available from <https://www.smartdraw.com/swim-lane-diagram/> [21 January 2021]

4. *Entity Relationship Diagram (ERD) - What Is An ER Diagram?* (2021) available from <https://www.smartdraw.com/entity-relationship-diagram/> [21 January 2021]

5. *Class Diagram Relationships In UML Explained With Examples* (2020) available from <https://creately.com/blog/diagrams/class-diagram-relationships/> [21 January 2021]