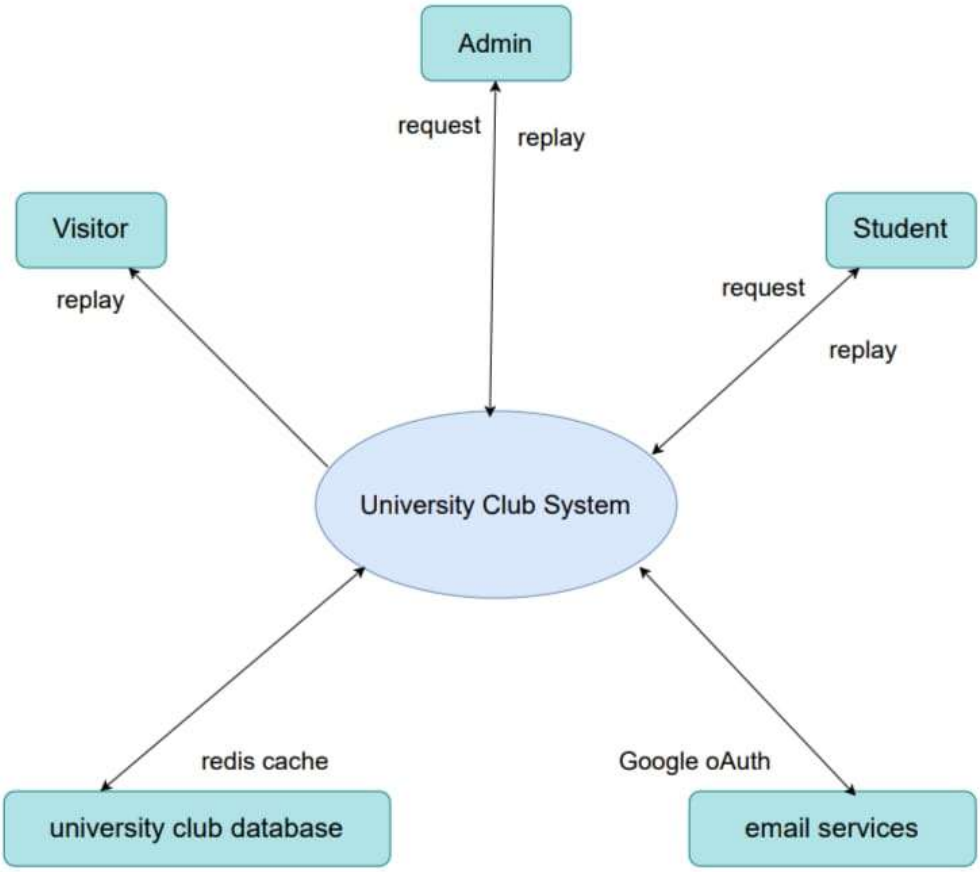
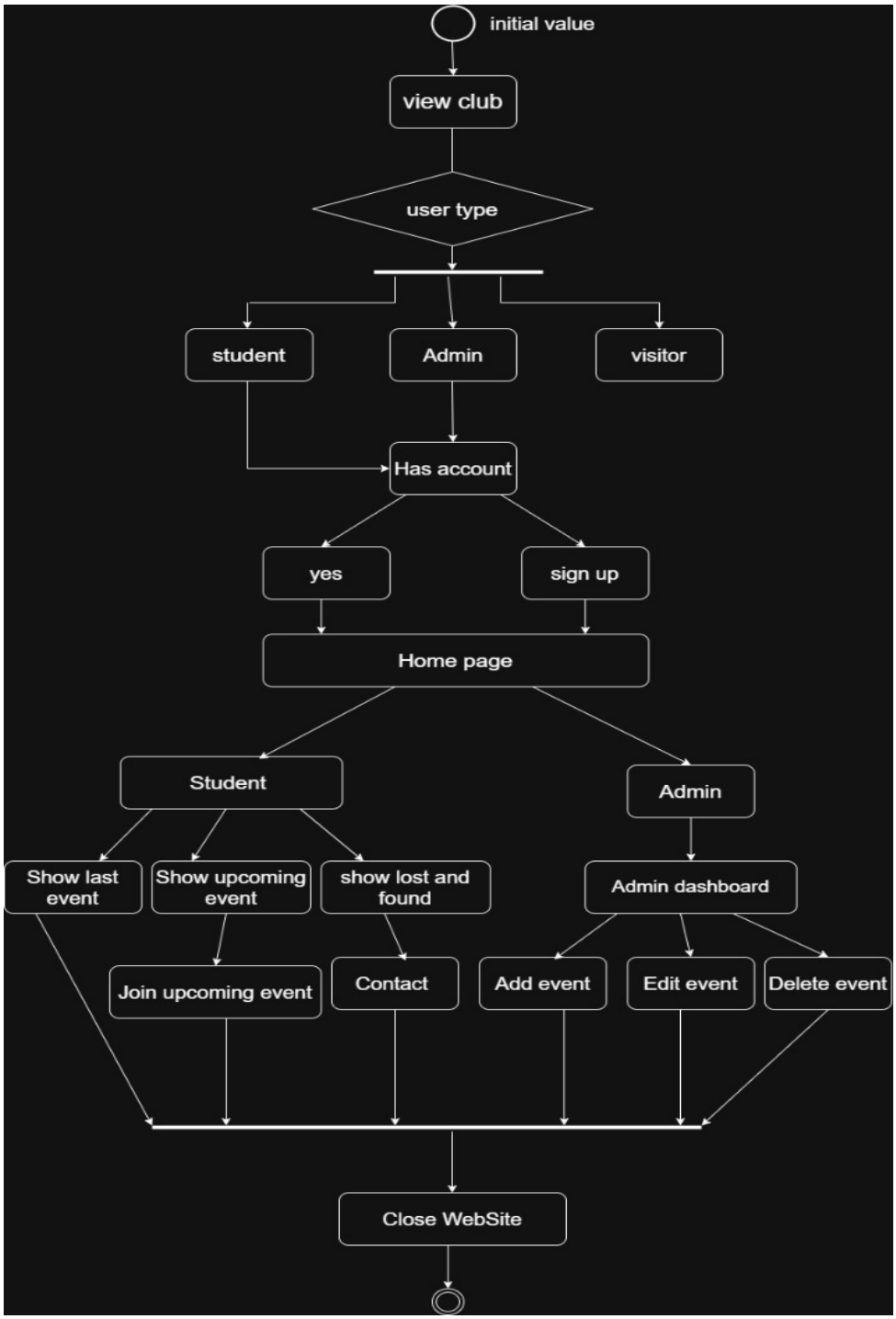
Design of University Club

Prof:Wessam Ahmed

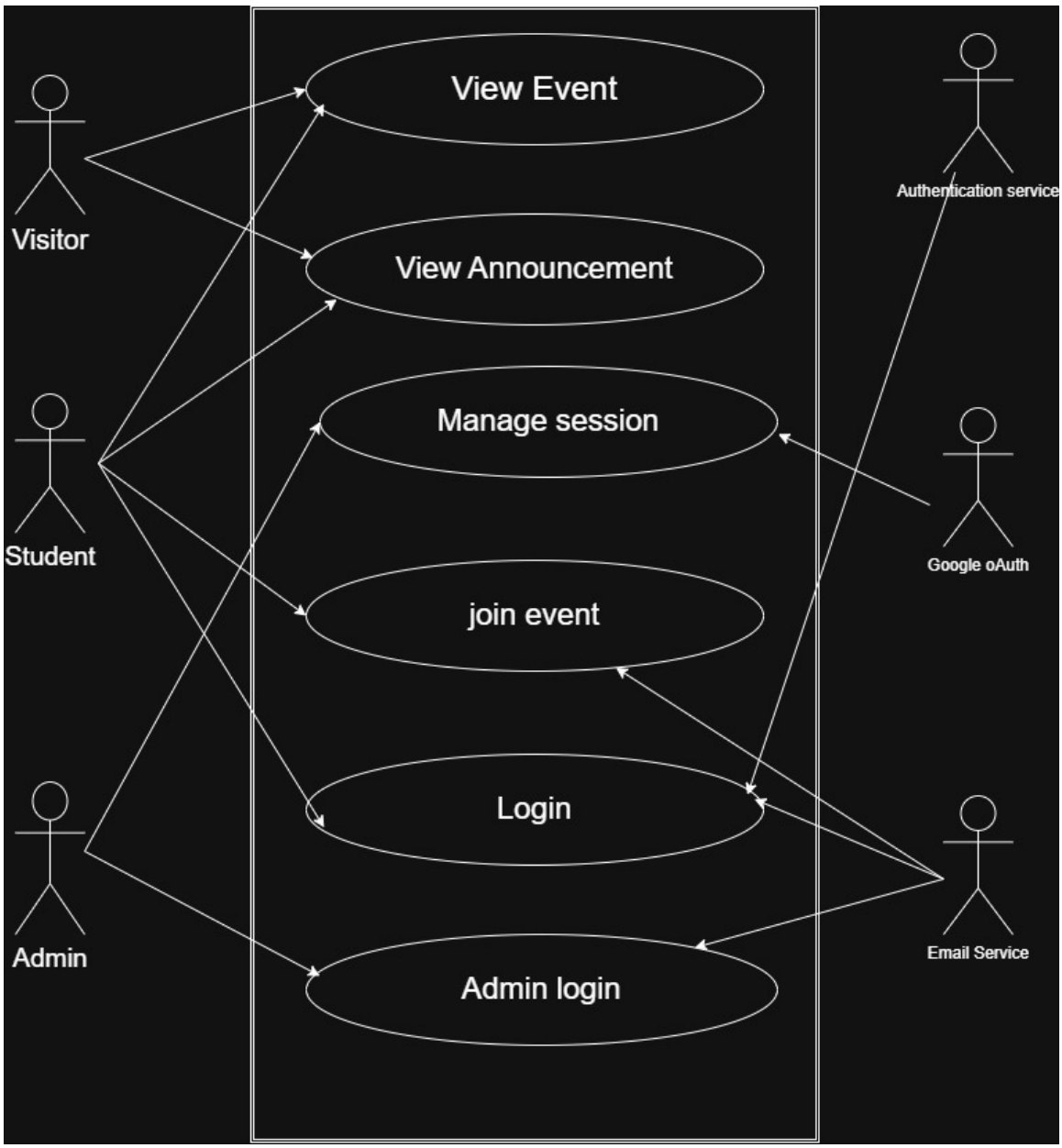
Context Model

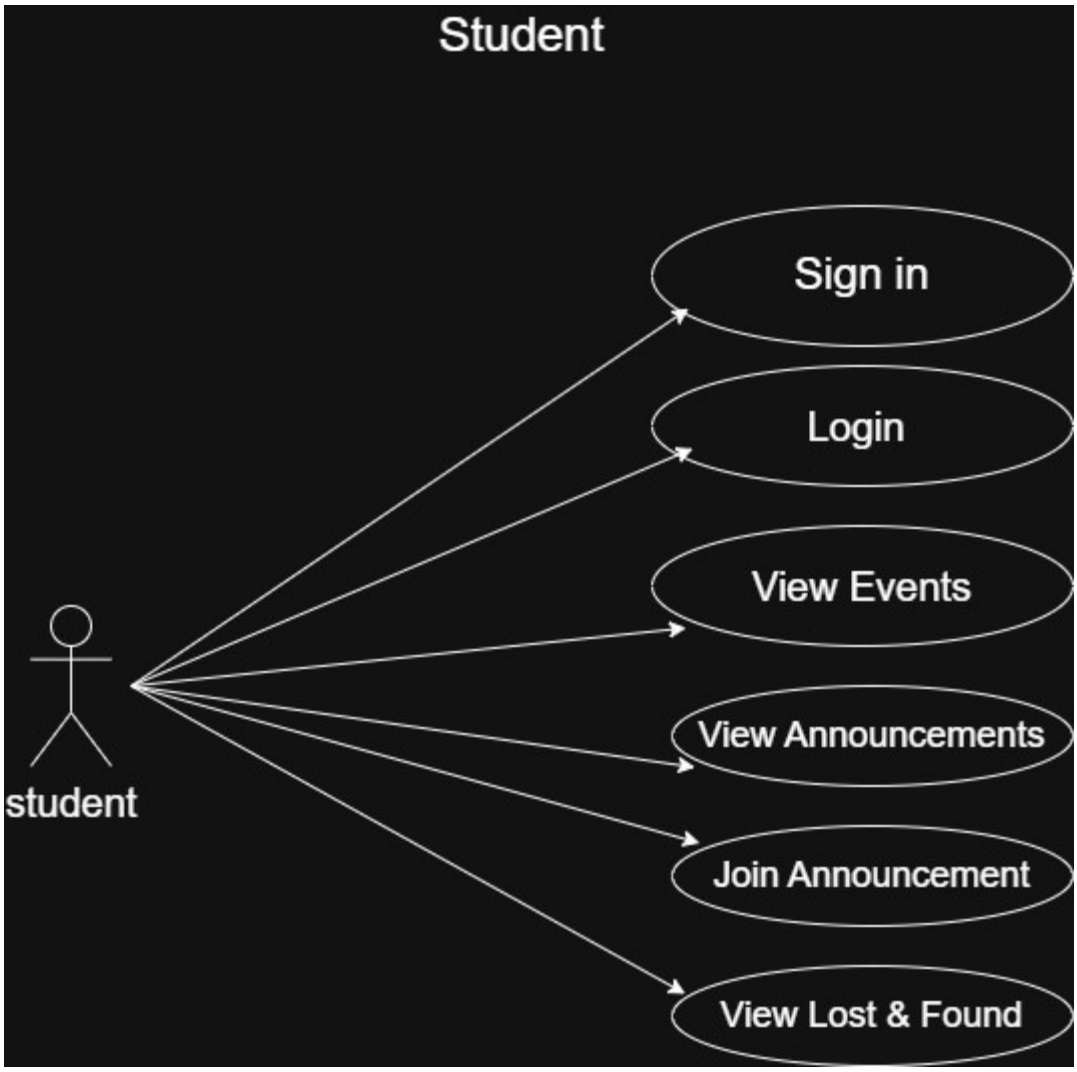


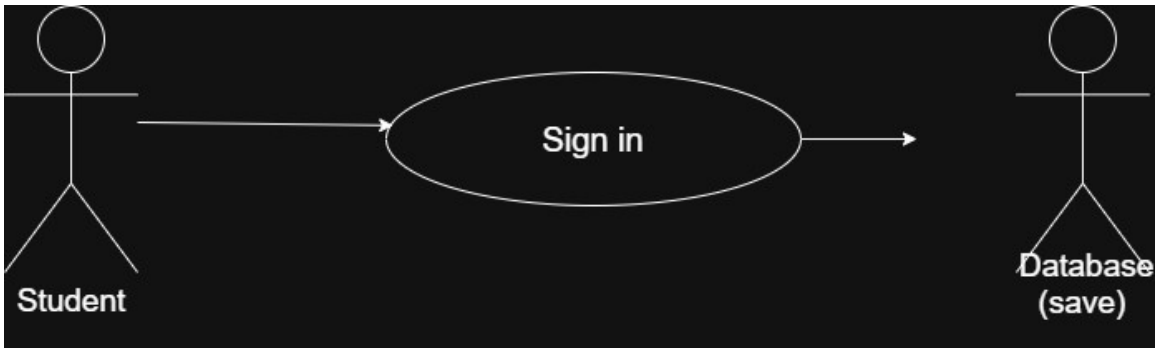
Process Prespective



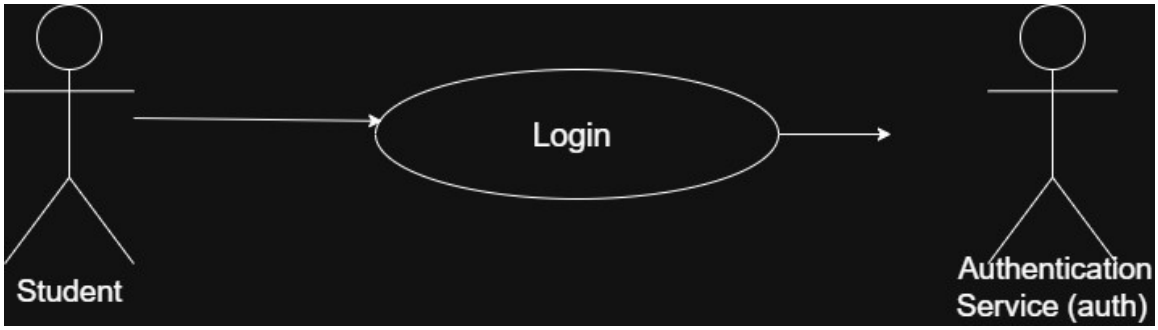
Interaction Model



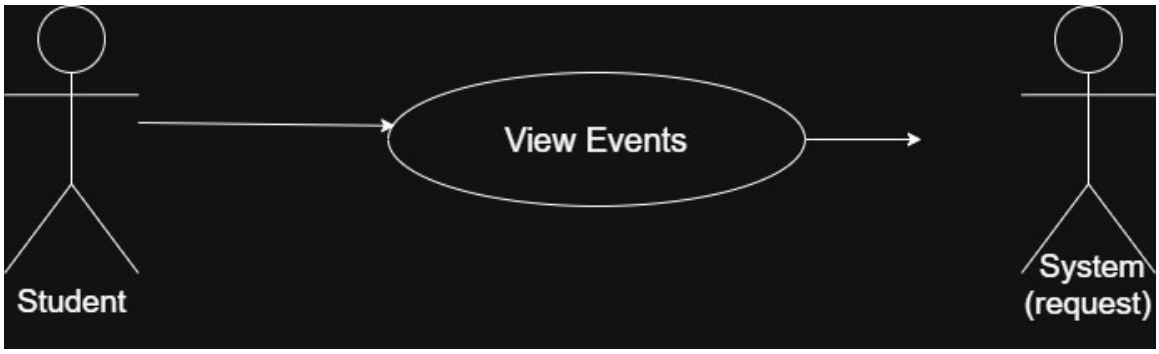

Student
Sign in
Login
View Events
View Announcements
Join Announcement
View Lost & Found
student



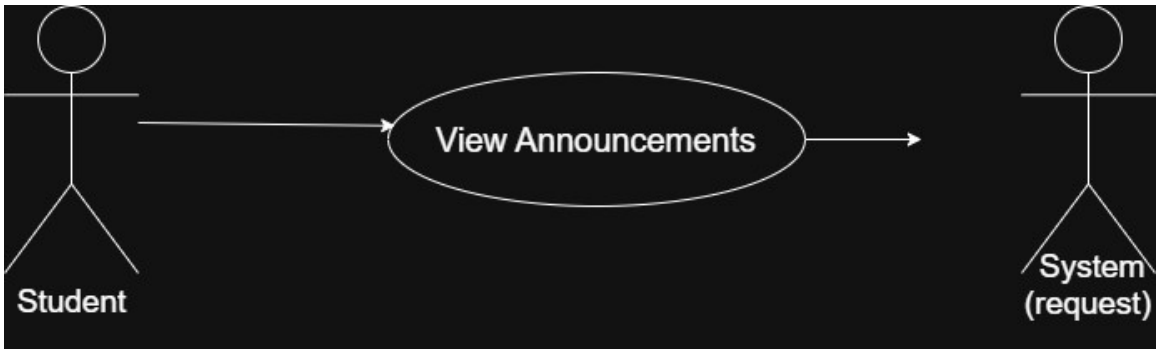
Actors	Student, Database System, Email Service
Description	A new student creates an account in the club management system by providing personal and academic information.
Data	Name, student ID, email, password, phone, faculty, academic year
Stimulus	Student submits registration form
Response	Account created, confirmation message, verification email sent
Comments	Email must be valid university email. Password requirements enforced.



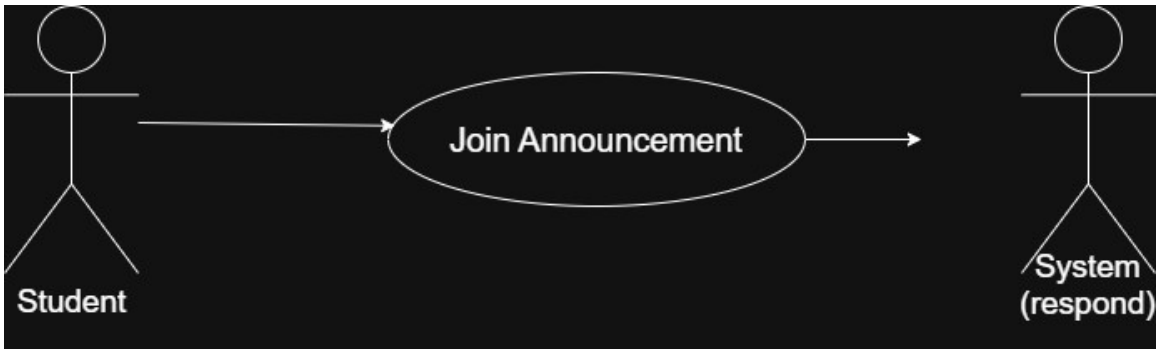
Actors	Student/Admin, Authentication System, Database
Description	User logs in using email and password stored in system database
Data	Email, password, session token
Stimulus	User submits login credentials
Response	Session established, redirect to dashboard
Comments	Failed attempts limited. Session timeout configured



Actors	Student, Event Service, Cache System
Description	Student browses list of available club events
Data	Event list with filters, pagination
Stimulus	Student navigates to events page
Response	Display events with details and filters
Comments	Events filtered by date, category. Cached for performance.

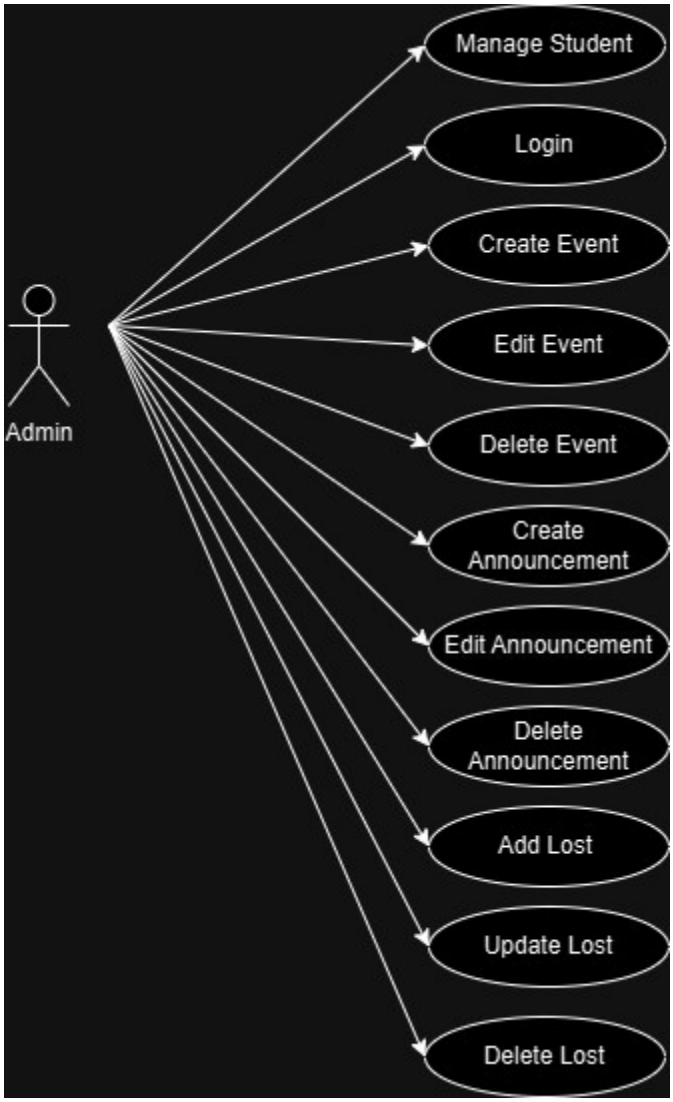


Actors	Student, Announcement Service
Description	Student views published club announcements
Data	Announcement list, filters
Stimulus	Student navigates to announcements
Response	Display announcements chronologically
Comments	Can filter by category, importance

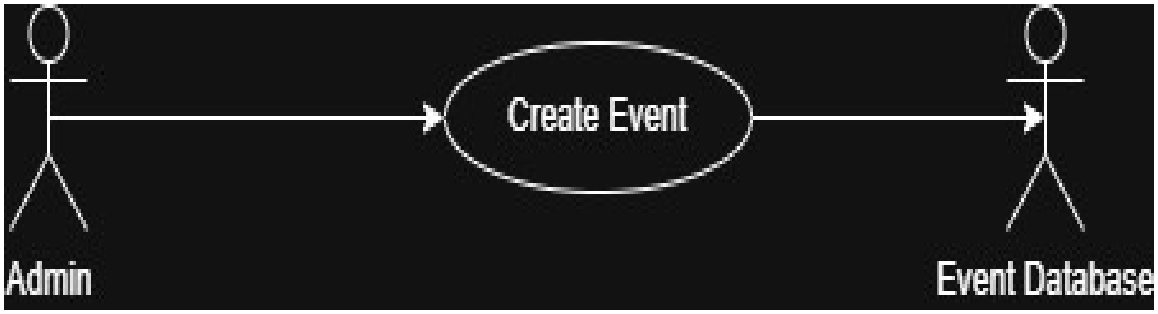


Actors	Student, Announcement System
Description	Student expresses interest in announcement (e.g., volunteer opportunity)
Data	Student ID, Announcement ID
Stimulus	Student clicks "Interested" or "Join"
Response	Interest recorded, confirmation shown
Comments	Optional feature for certain announcement types

Actors	Student, Lost & Found System
Description	Student browses lost items reported in club
Data	Lost items list with details
Stimulus	Student navigates to lost & found section
Response	Display items with status and contact info
Comments	Items filtered by date, location, status.


Admin
Manage Student
Login
Create Event
Edit Event
Delete Event
Create Announcement
Edit Announcement
Delete Announcement
Add Lost
Update Lost
Delete Lost

Actors	Admin, Authentication system
Description	Administrator logs into admin dashboard
Data	Admin credentials, session
Stimulus	Admin submits login form
Response	Admin dashboard access granted
Comments	Separate authentication from student login.



Actors	Admin, Event Database
Description	Admin creates new club event
Data	Event details, scheduling, limits
Stimulus	Admin submits event creation form
Response	Event saved, success message
Comments	Validates date/time. Optional image upload.

Actors	Admin, Event Database
Description	Admin modifies existing event details
Data	Updated event information
Stimulus	Admin submits event edit form
Response	Event updated, participants notified
Comments	Some changes may require participant notification.

Actors	Admin, Event Database
Description	Admin removes event from system
Data	Event ID, deletion reason
Stimulus	Admin confirms deletion
Response	Event removed, participants notified
Comments	May allow cancellation instead of deletion.

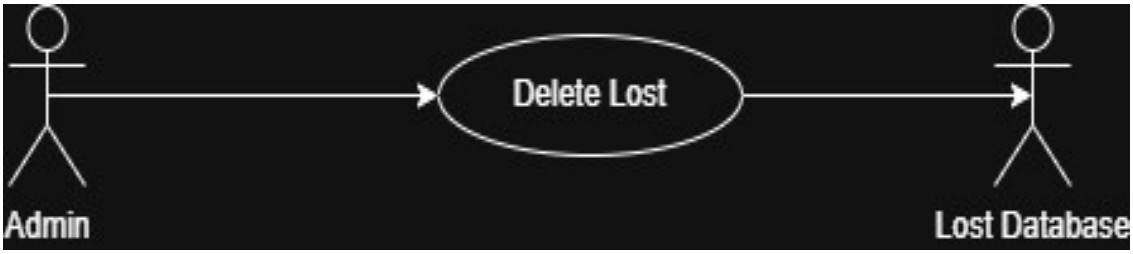
Actors	Admin, Announcement Database
Description	Admin publishes new announcement
Data	Announcement content, metadata
Stimulus	Admin submits announcement form
Response	Announcement published, visible to target audience
Comments	Can schedule publication. Target specific groups.

Actors	Admin, Announcement Database
Description	Admin modifies published announcement
Data	Updated announcement content
Stimulus	Admin submits edit form
Response	Announcement updated, edit history tracked
Comments	Major changes may require republication notice.

Actors	Admin, Announcement Database
Description	Admin removes announcement
Data	Announcement ID
Stimulus	Admin confirms deletion
Response	Announcement removed from view
Comments	May archive instead of permanent delete.

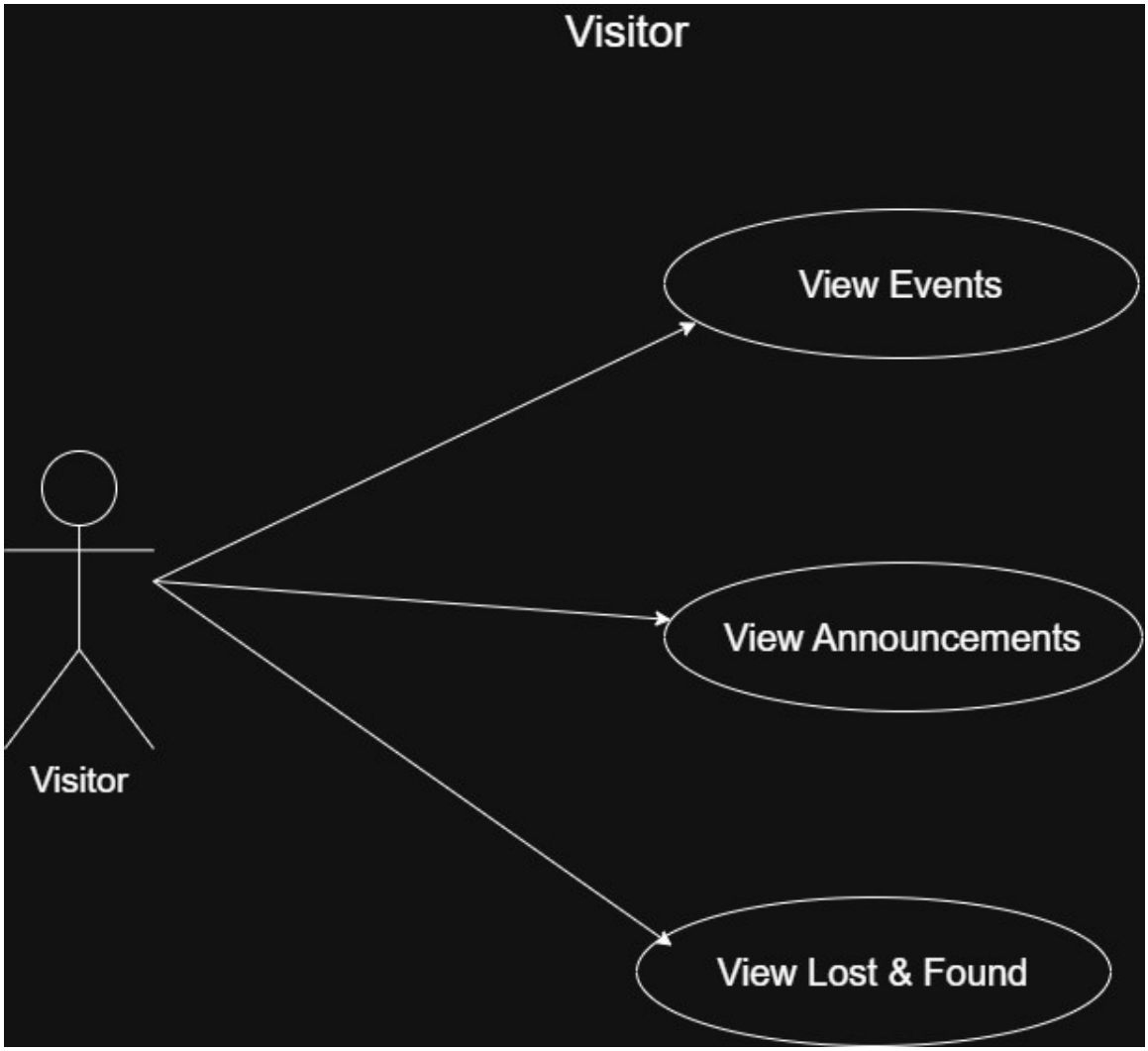
Actors	Admin, Lost Database
Description	Admin registers found/lost item
Data	Item details, location, date
Stimulus	Admin submits item report form
Response	Item added to lost & found list
Comments	Includes contact information for claimant.

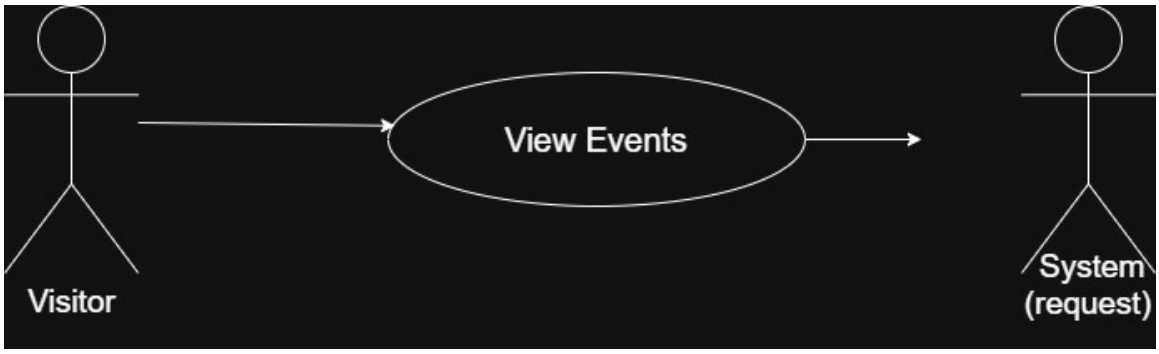
Actors	Admin, Lost Database
Description	Admin updates item status/details
Data	Updated item information
Stimulus	Admin submits update form
Response	Item record updated
Comments	Status changes (found/claimed) trigger notifications.



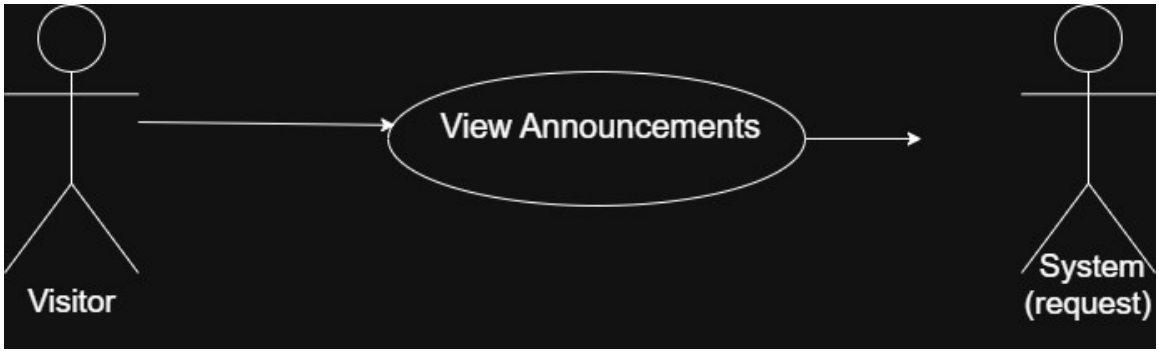
Actors	Admin, Lost Database
Description	Admin removes item record
Data	Item ID, removal reason
Stimulus	Admin confirms deletion
Response	Item removed from system
Comments	Usually after claim or extended period.

Actors	Admin, User Management System
Description	Admin views, searches, and manages student accounts
Data	Student list with filters, search results
Stimulus	Admin accesses student management page
Response	Display student accounts with management options
Comments	Can deactivate, view activity, reset passwords.


Visitor
View Events
View Announcements
View Lost & Found
Visitor

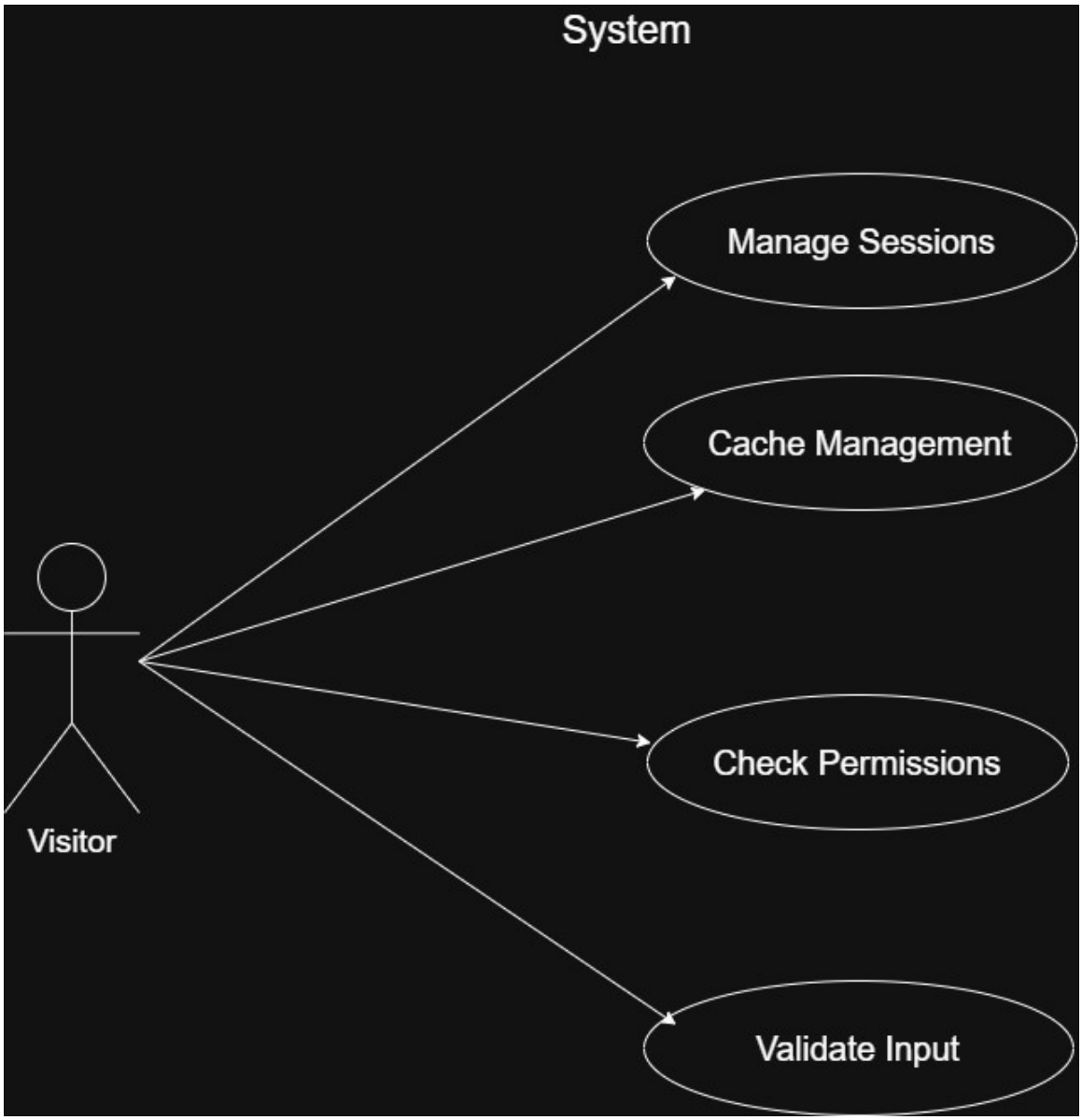


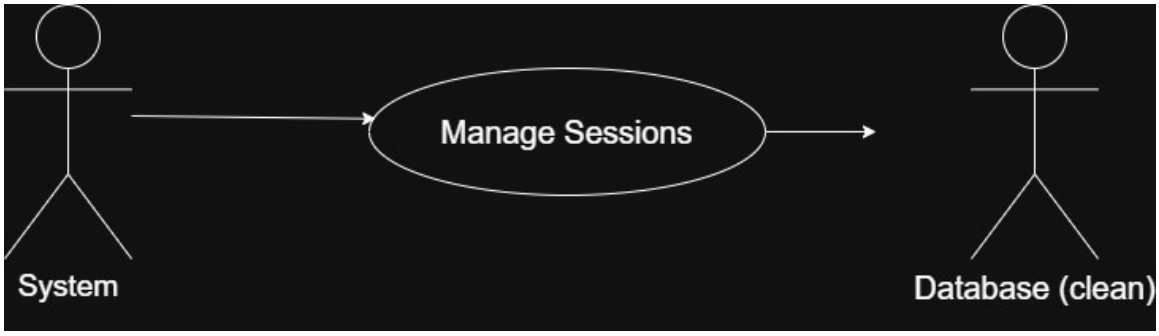
Actors	Public User, Event System
Description	Non-logged in user views public events
Data	Public event list
Stimulus	User visits public events page
Response	Display public events only
Comments	Limited information compared to logged-in view.



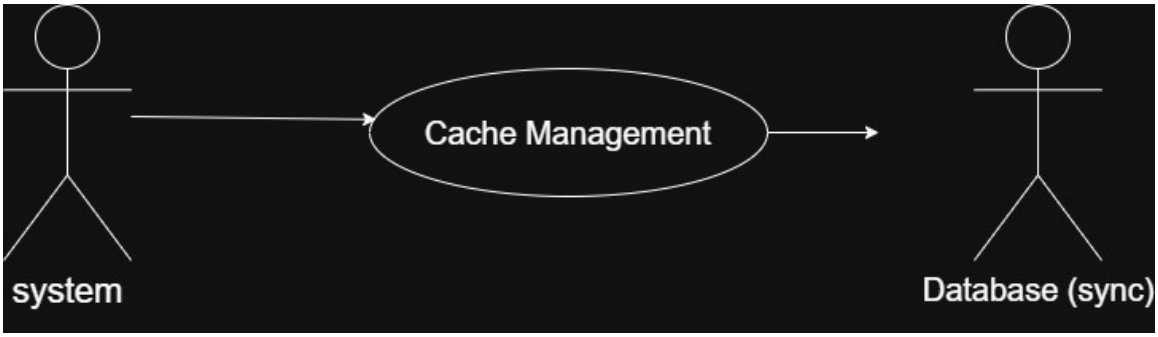
Actors	Public User, Announcement System
Description	Public user views public announcements
Data	Public announcements list
Stimulus	User visits announcements page
Response	Display public announcements
Comments	Internal announcements hidden

Actors	Public User, Lost & Found System
Description	Public user views lost items
Data	Public lost items list
Stimulus	User visits lost & found page
Response	Display items without sensitive info
Comments	Contact information may be limited.

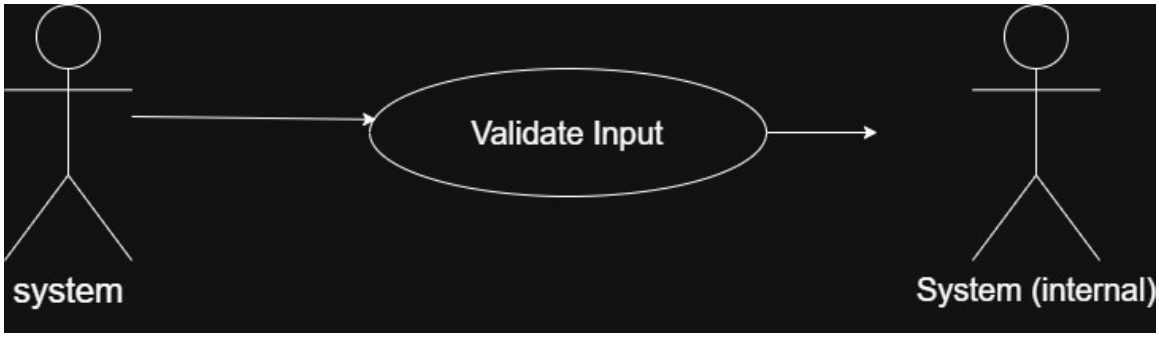

System
Manage Sessions
Cache Management
Check Permissions
Validate Input
Visitor



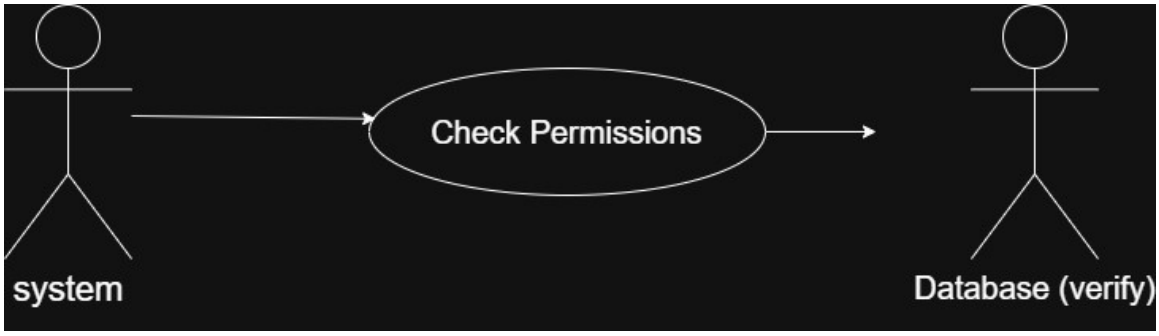
Actors	System (automated), Session Database
Description	System automatically manages user sessions
Data	Session tokens, timestamps, user IDs
Stimulus	User login/logout or timeout
Response	Session created/updated/destroyed
Comments	Automated system process. Cleans expired sessions.



Actors	System, Cache Service
Description	System caches frequently accessed data
Data	Cached data, cache keys
Stimulus	Data request with cacheable content
Response	Serve cached data or update cache
Comments	Improves performance. Automatic invalidation

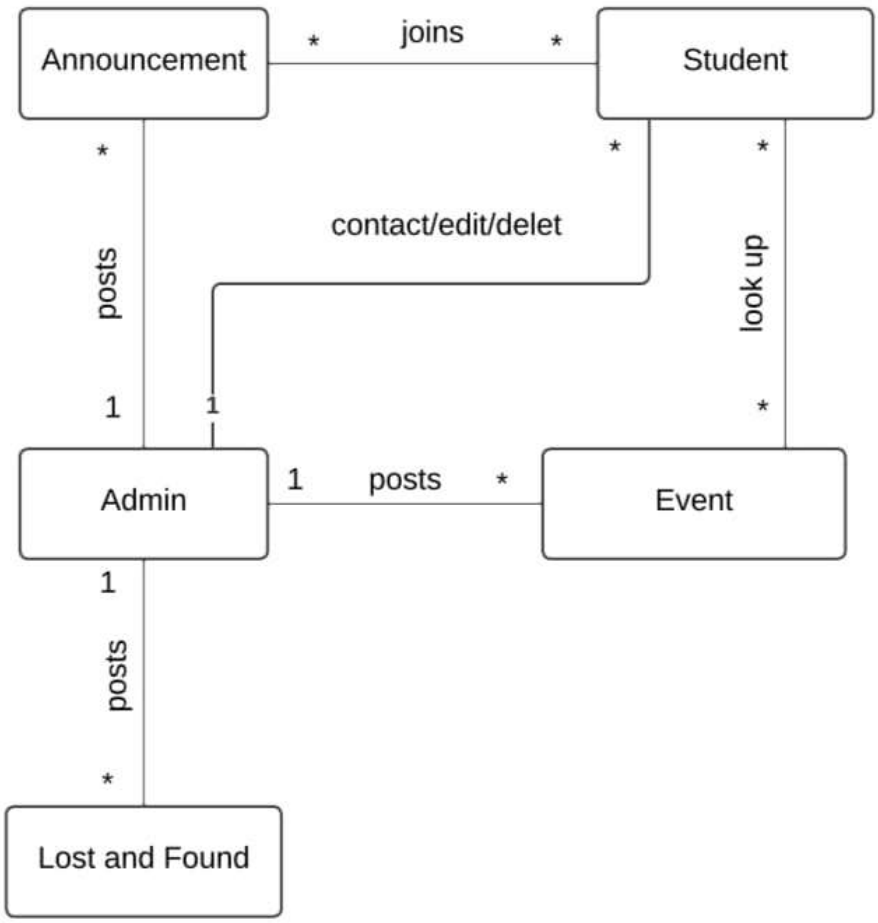


Actors	System, Validation Service
Description	System validates all user input
Data	Input data, validation rules
Stimulus	Form submission or API call
Response	Validation pass/fail with errors
Comments	Prevents invalid/malicious data



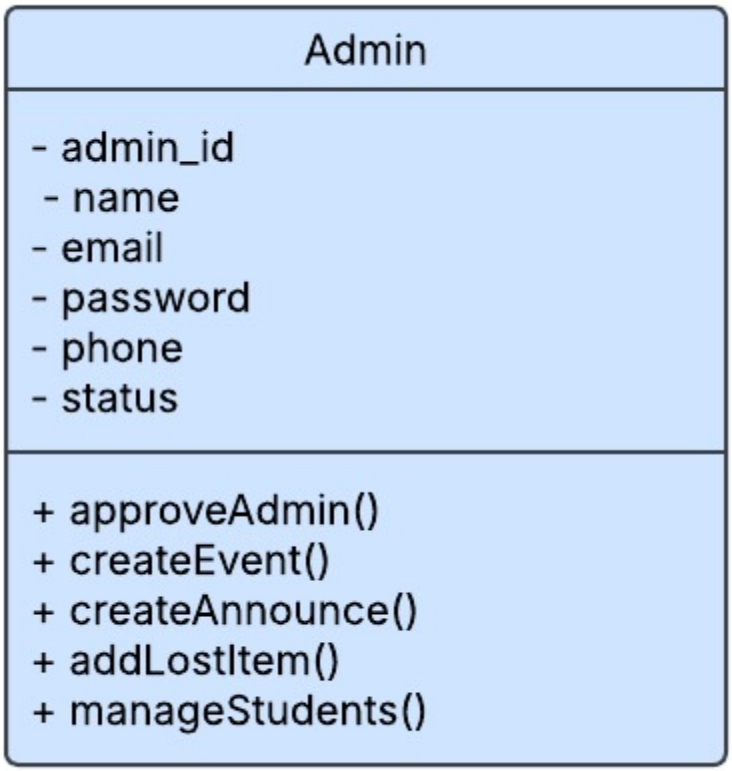
Actors	System, Permission Service
Description	System verifies user permissions for actions
Data	User role, requested action, resource
Stimulus	User attempts restricted action
Response	Allow or deny access
Comments	Role-based access control. Logs denied attempts.

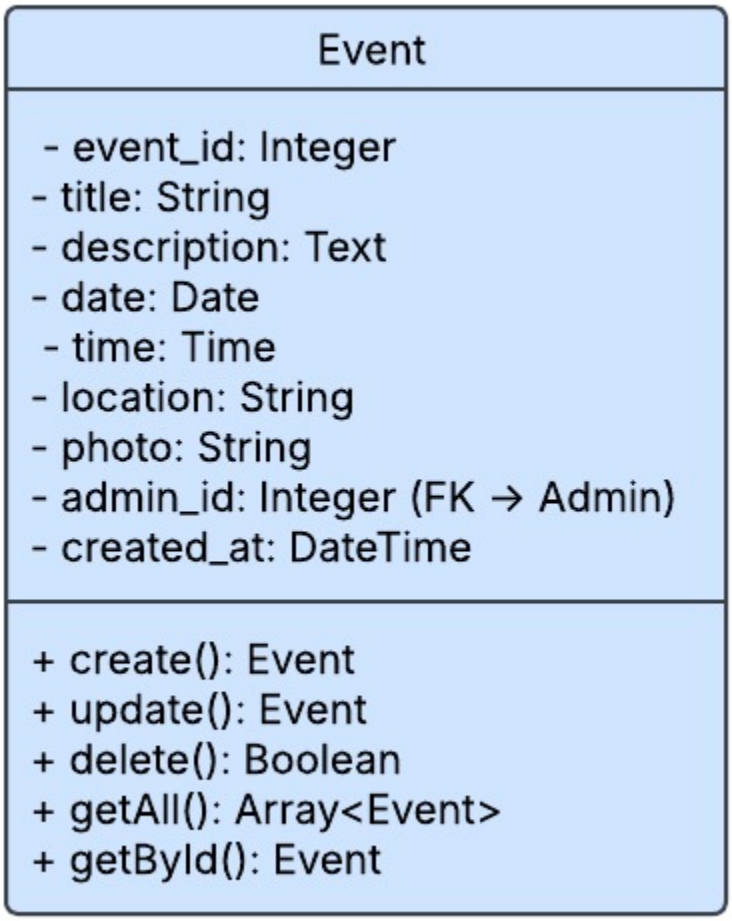
Structral Model

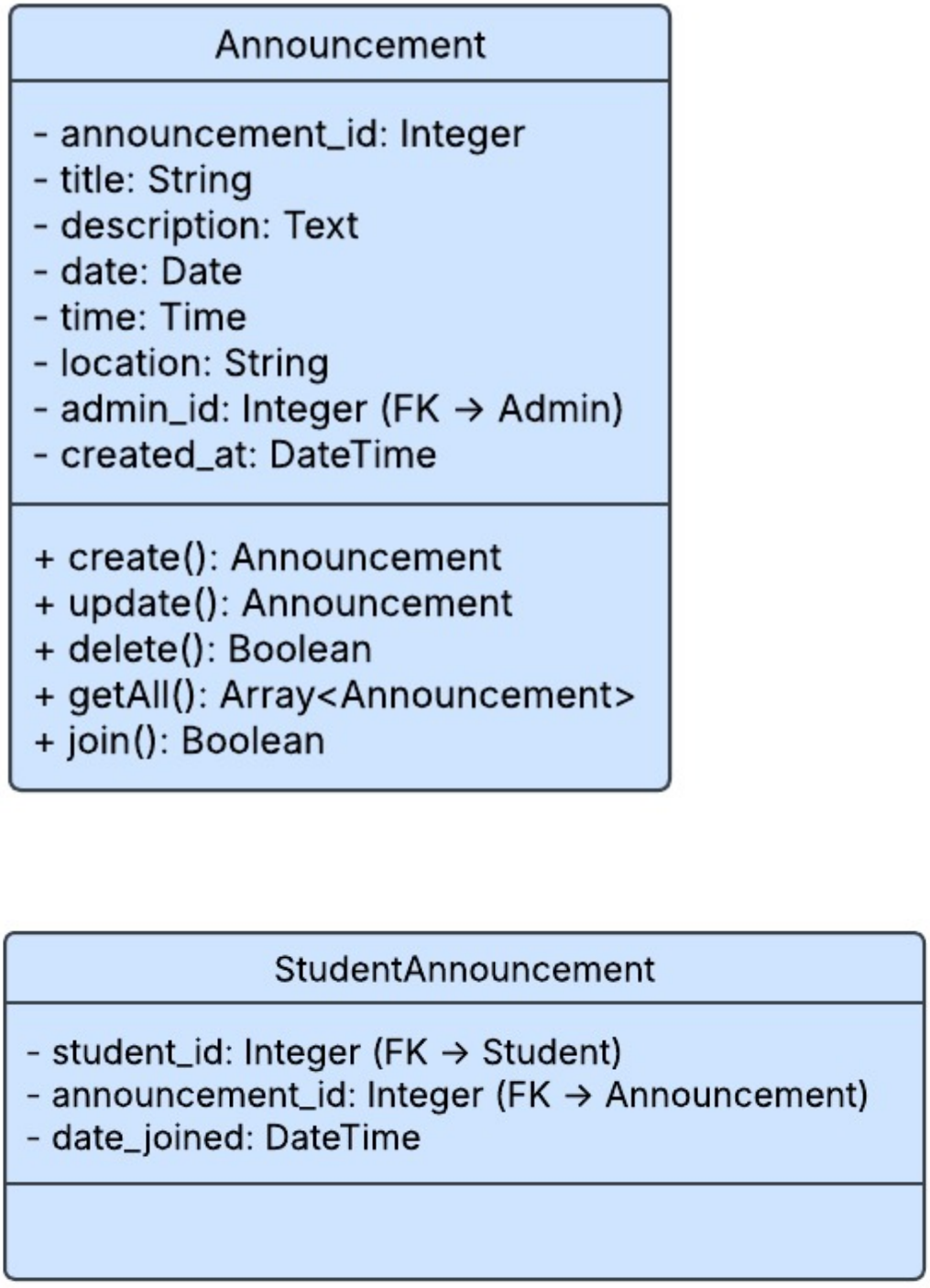


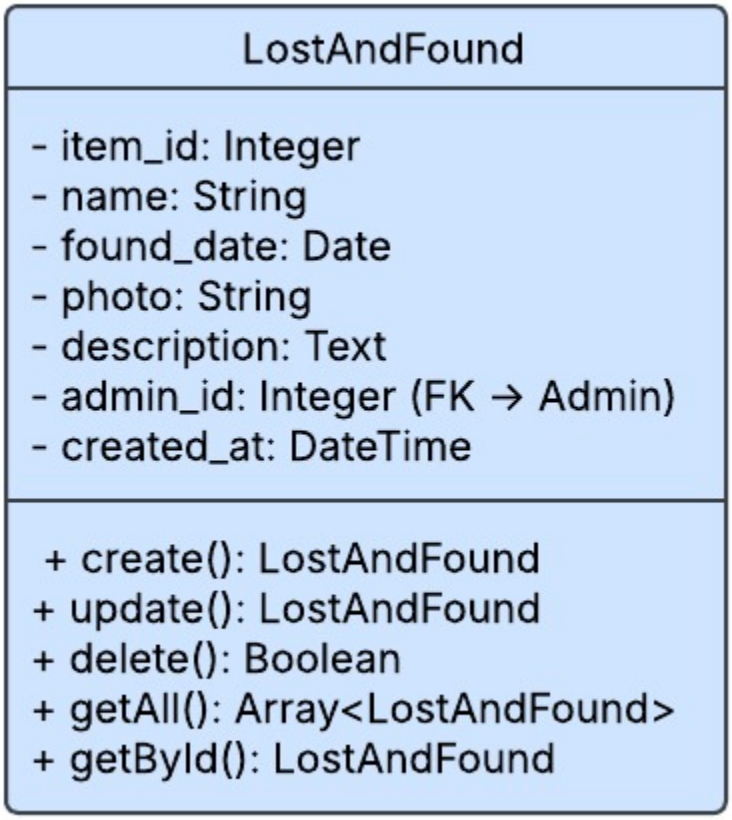
Student
-Student_id
-name
-faculty
-phone
-email
-password
-level
+login()
+joinEvent()
+updateProfile()
+getMyEvents()

Visitor
- session_id
- ip_address
- access_time
+ viewEvents()
+ viewAnnounce()
+ viewLostItems()


Admin
- admin_id
- name
- email
- password
- phone
- status
+ approveAdmin()
+ createEvent()
+ createAnnounce()
+ addLostItem()
+ manageStudents()


Event
- event_id: Integer
- title: String
- description: Text
- date: Date
- time: Time
- location: String
- photo: String
- admin_id: Integer (FK → Admin)
- created_at: DateTime
+ create(): Event
+ update(): Event
+ delete(): Boolean
+ getAll(): Array<Event>
+ getById(): Event


Announcement
- announcement_id: Integer
- title: String
- description: Text
- date: Date
- time: Time
- location: String
- admin_id: Integer (FK → Admin)
- created_at: DateTime
+ create(): Announcement
+ update(): Announcement
+ delete(): Boolean
+ getAll(): Array<Announcement>
+ join(): Boolean
StudentAnnouncement
- student_id: Integer (FK → Student)
- announcement_id: Integer (FK → Announcement)
- date_joined: DateTime


LostAndFound
- item_id: Integer
- name: String
- found_date: Date
- photo: String
- description: Text
- admin_id: Integer (FK → Admin)
- created_at: DateTime
+ create(): LostAndFound
+ update(): LostAndFound
+ delete(): Boolean
+ getAll(): Array<LostAndFound>
+ getById(): LostAndFound