# Getting Ready: The LinkedIn System

Understand the LinkedIn problem and learn the questions to simplify this problem further.

**We'll cover the following**

* [Problem definition](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Problem-definition)
* [Expectations from the interviewee](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Expectations-from-the-interviewee)
  + [Discoverability](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Discoverability)
  + [Connections and following](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Connections-and-following)
  + [Groups, pages, and jobs](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Groups,-pages,-and-jobs)
  + [Alerts](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Alerts)
* [Design approach](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Design-approach)
* [Design pattern](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Design-pattern)

## Problem definition

**LinkedIn** is an employment-focused social network platform to manage the user's professional identity. This platform is primarily used to facilitate its users. It helps them find new opportunities, grow their careers, and connect with the people they know and trust professionally. Moreover, this platform allows employers to post jobs. It allows job seekers to post their resumes to find the best employment match for themselves. Therefore, LinkedIn is used for career development and professional networking.

LinkedIn allows both workers and companies to create profiles and pages, respectively. The LinkedIn users' profile page represents their education, experience, skills, achievements, and recommendations. LinkedIn also allows its users to create posts regarding a topic of interest, comment on a post, invite other users to become a connection, message other users, and create groups to form a community of similar users. LinkedIn generates a personalized feed for its users based on their connections, job, work, and education history, liked pages, groups, and the content they engage with to ensure the best experience.

LinkedIn is very similar to Facebook in terms of its layout and design. These features are more specialized because they cater to professionals. However, if we know how to use Facebook or any other similar social network, we may find LinkedIn familiar.

The home page of a LinkedIn user

## Expectations from the interviewee

LinkedIn provides multiple functionalities to its users. It is essential to narrow down the components you will include in your LinkedIn design. The section below provides an overview of some of the main expectations that the interviewer will want to hear you discuss in more detail during the interview.

### Discoverability

It is important to develop a better understanding of how LinkedIn’s discoverability feature works. The interviewer would expect you to ask the questions listed below:

* How can users search the job opportunities for themselves?
* How are companies able to find out the perfect match for their job openings?
* How are users able to search other users’ profiles?
* Can users search for other users using their company name/employment history and job type?

### Connections and following

Both connections and following are the primary features of LinkedIn. Make sure to ask the following questions from the interviewer:

* How are users able to connect with other users?
* Can a page connect with other pages as well?
* How can users follow or unfollow the pages without becoming a direct connection?

### Groups, pages, and jobs

Groups and pages on LinkedIn create a space for people looking for similar job opportunities. Make sure to define your requirements. You may ask the following questions from the interviewer:

* Can the users create both groups and pages in addition to their user profiles?
* Can both the users and companies create groups and pages?
* Can the individual user post a job?
* Can a user join any group, or are there any criteria for joining the group?

### Alerts

Notifications and alerts allow users to stay updated with the activity in their circle. Therefore, you may want to understand how alerts work in your system. You may ask the following questions:

* How will users be notified of preselected events?
* How much control do users have in choosing what notifications they can receive?

## Design approach

We’ll design LinkedIn using the bottom-up design approach. For this purpose, we will follow the steps below:

* Identify and design the smallest components first—features like a post and comment.
* Use these small components to design bigger components—a page, group, and profile.
* Repeat the steps above until we design the complete LinkedIn platform.

## Design pattern

It is always a good practice to discuss the design patterns that LinkedIn falls under, during the interview. Stating the design patterns will give the interviewer a positive impression and shows that the interviewee is well-versed in the advanced concepts of object-oriented design. We can use the Observer design pattern to design LinkedIn.

Let’s explore the requirements of the LinkedIn design in the next lesson.

# Requirements for LinkedIn

Learn about all requirements of LinkedIn.

**We'll cover the following**

* [Requirements collection](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Requirements-collection)

In this lesson, we’ll list the requirements of the LinkedIn design problem. This is a crucial step because requirements define the scope of a problem. Therefore, getting them right from the interviewer and understanding them well will make the design of the rest of the system smooth and easy.

We’ll use the notational convention to identify each requirement with a unique label, "Rn" where "R" is short for Requirement and "n" is a natural number.

## Requirements collection

The requirements for LinkedIn are defined below:

**R1:** Users should be able to add information to their profile including education, experiences, achievements, and skills.

**R2:** Users should be able to search for, and also view, pages, groups, and other users.

**R3:** Users should be able to send and cancel connection requests. They should also be able to respond to the connection requests of other users by either accepting or ignoring them.

**R4:** Users should be able to follow other users without adding them as their connection.

**R5:** Users should be able to view their number of connections, profile views, post impressions, and search appearances.

**R6:** Users should be able to request and give recommendations to other users.

**R7:** Users should be able to write a new post.

**R8:** Users should be able to react, share, and comment on a post. They should also be able to react or comment on an existing comment.

**R9:** A user should be able to send and receive messages from other users.

**R10:** The system should send a notification to the user to inform them about messages, connection requests, or comments on their post.

**R11:** Users should be able to create company pages. Users should be able to follow other company pages.

**R12:** Company pages should have a list of job openings that users can apply for.

**R13:** A user should be able to create and join groups.

We've identified our requirements for the problem. In the next lesson, we will define different use cases of LinkedIn.

# Use Case Diagram for LinkedIn

Learn how to define use cases and create the corresponding use case diagram for the LinkedIn problem.

**We'll cover the following**

* [System](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#System)
* [Actors](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Actors)
  + [Primary actors](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Primary-actors)
  + [Secondary actors](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Secondary-actors)
* [Use cases](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Use-cases)
  + [User](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#User)
  + [Admin](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Admin)
  + [System](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#System)
* [Relationships](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Relationships)
  + [Generalization](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Generalization)
  + [Associations](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Associations)
  + [Include](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Include)
  + [Extend](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Extend)
* [Use case diagram](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Use-case-diagram)

Let's build the use case diagram of LinkedIn and understand the relationship between its different components.

First, we’ll define the different elements of our LinkedIn system, followed by the complete use case diagram of the system.

## System

Our system is "LinkedIn."

## Actors

Now, we’ll define the main actors of LinkedIn.

### Primary actors

* **User:** This actor can create a profile including their professional information. They can create posts, apply for jobs, follow pages, and join groups. They can also interact with other users by sending them connection invitations and messages, commenting on their posts, etc.
* **Admin:** The admin is in charge of performing numerous operations, including blocking or unblocking users, enabling/disabling pages, deleting an existing group, etc.

### Secondary actors

* **System:** This is responsible for sending out notifications for new connection requests, messages, comments, posts, recommendations, etc.

## Use cases

In this section, we’ll define the use cases for LinkedIn. We have listed the use cases according to their respective interactions with a particular actor.

**Note:** You’ll see some use cases occurring multiple times because they are shared among different actors in the system.

### User

* **Add/update profile:** To add information like education, experience, skill, and achievement to update an existing profile
* **Follow/unfollow user:** To follow or unfollow other users
* **Send message:** To send a message to other users
* **Send connection invitation:** To send a connection invitation to other users
* **Accept/ignore connection invitation:** To accept or ignore a connection invitation from another user
* **Create/follow/unfollow page:** To either create a new page or perform actions including update, delete, or follow an existing page
* **Create/join/leave a group:** To either create a new group or perform actions like joining or leaving an existing group
* **Add/share post:** To add a new post or share an existing post
* **Update/delete/react post:** To update a post, delete a post, or react to the post
* **Comment on post/comment:** To comment on the post, or to add a comment on an existing comment
* **Update/delete/react comment:** To update the content of the comment, react to a particular comment, or delete a comment
* **Search users/groups/pages/jobs:** To search for other users, or any existing groups, company pages, or jobs
* **Post a job:** To post the job on the page
* **View analytics:** To see the number of connections, post impressions, profile views, and search appearances of the user
* **Request recommendation:** To send a recommendation request to other users
* **Accept recommendation:** To accept a recommendation request from the user
* **Delete recommendation:** To delete a recommendation request from the user

### Admin

* **Block/unblock user:** To block or unblock a user on LinkedIn
* **Enable/disable page:** To enable or disable a page
* **Delete group:** To delete an existing group

### System

* **New post notification:** To send a notification of any new posts
* **New comment notification:** To send a notification whenever another user comments on a user's post or comment
* **Connection invitation notification:** To send a notification of any connection invitation sent by any user
* **Message notification:** To send a notification of any new messages
* **Recommendation notification:** To send a notification of recommendation requested by any user
* **Add recommendation:** To add a recommendation to the user's profile

## Relationships

This section describes the relationships between and among actors and their use cases.

### Generalization

* The user can add/update their profile by either adding or updating their education, experience, skill, or achievement. This shows that the “Add/update profile” use case has a generalization relationship with “Add/update education,” “Add/update experience,” “Add/update skill,” and “Add/update achievement” use cases.

### Associations

The table below shows the association relationship between actors and their use cases.

|  |  |  |
| --- | --- | --- |
| **User** | **Admin** | **System** |
| Add/update profile | Block/unblock user | New post notification |
| Follow/unfollow user | Enable/disable page | New comment notification |
| Send message | Delete group | Connection invitation notification |
| Send connection invitation |  | Message notification |
| Accept/ignore connection invitation | Recommendation notification |
| Create/follow/unfollow page | Add recommendation |
| Create/join/leave a group |  |
| Add/share post |
| Update/delete/react post |
| Comment on post/comment |
| Update/delete/react comment |
| Search users/groups/pages/jobs |
| Post a job |
| View analytics |
| Request recommendation |
| Accept recommendation |
| Delete recommendation |

### Include

* The “Send message” use case has an include relationship with the “Message notification” use case, because whenever a user receives a message, the user is notified.
* The “Send connection invitation” use case has an include relationship with the “Connection invitation notification” use case. Whenever a user receives the connection invitation, the system notifies the user.
* When the user accepts the recommendation request, the system adds that recommendation in the user's profile. Therefore, the “Accept recommendation” use case has an include relationship with the “Add recommendation” use case.
* The “Add/share post" use case has an include relationship with the “New post notification.” Whenever a user creates a post or shares it, the system notifies the connected users.
* The “Comment on post/comment” use case has an include relationship with the “New comment notification” use case. Whenever a new comment is made by another user on a post either created by the default user or followed by the default user, the system notifies the default user.
* The “Request recommendation” use case has an include relationship with the “Recommendation notification” use case. Whenever a user gets a recommendation request, the system notifies the user.

### Extend

* When users get the recommendation request, they can either accept or delete the request. Therefore, the “Request recommendation” use case has an extend relationship with the “Accept recommendation” and "Delete recommendation" use cases.
* Whenever the users get a connection invitation request, they can either accept or ignore the request. Therefore, the “Send connection invitation” use case has an extend relationship with the “Accept/ignore connection invitation” use case.

## Use case diagram

Here’s the use case diagram of LinkedIn:

A diagram of a diagram

Description automatically generated

A screenshot of a computer screen

Description automatically generated

# Class Diagram for LinkedIn

Learn to create a class diagram for LinkedIn using the bottom-up approach.

**We'll cover the following**

* [Components of LinkedIn](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Components-of-LinkedIn)
  + [Account](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Account)
  + [Education, experience, and skill](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Education,-experience,-and-skill)
  + [Recommendation, achievement, and analytics](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Recommendation,-achievement,-and-analytics)
  + [Profile](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Profile)
  + [Company page, job, and group](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Company-page,-job,-and-group)
  + [Post, comment, and message](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Post,-comment,-and-message)
  + [Connection invitation](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Connection-invitation)
  + [Person](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Person)
  + [Notification](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Notification)
  + [Search interface and search catalog](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Search-interface-and-search-catalog)
  + [Enumerations](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Enumerations)
  + [Custom data type](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Custom-data-type)
* [Relationship between the classes](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Relationship-between-the-classes)
  + [Association](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Association)
  + [Composition](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Composition)
  + [Inheritance](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Inheritance)
* [Class diagram of LinkedIn](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Class-diagram-of-LinkedIn)
* [Design pattern](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Design-pattern)

In this lesson, we’ll identify and design the classes, abstract classes, and interfaces based on the requirements we have previously gathered from the interviewer in our LinkedIn system.

## Components of LinkedIn

As mentioned earlier, we will design the LinkedIn system using a bottom-up approach.

### Account

The Account class identifies a LinkedIn user through their username and ID. Users with an account will have the option to create groups, pages, posts, comments, and like other comments and posts.

The class definition is represented below:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

### Inheritance

The following classes demonstrate an inheritance relationship:

* Both the Admin and User classes extend the Person class.

**Note:** We have already discussed the inheritance relationship between classes in the component section above one by one.

## Class diagram of LinkedIn

Here’s the complete class diagram for LinkedIn:

A diagram of a server

Description automatically generated

A screenshot of a social media pattern

Description automatically generated

# Sequence Diagram for LinkedIn

Visualize the sequence diagram to send a connection invitation on LinkedIn.

**We'll cover the following**

* [Send a connection invitation](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Send-a-connection-invitation)

Sequence diagrams are a great way to understand the interactions between different entities and objects in the system. There can be different sequence diagrams that we can create for LinkedIn. In this lesson, we’ll create sequence diagrams for the following interaction:

* **Send a connection invitation:** A user sends a connection invitation to another user.

## Send a connection invitation

The sequence diagram for sending a connection invitation should have the following actors and objects that will interact with each other:

* **Actors:** UserA and UserB
* **Objects:** SearchCatalog and ConnectionInvitation

Here are the steps for the send a connection invitation interaction:

1. User A searches for user B in the catalog.
2. If User B exists:
   1. The catalog returns user B.
   2. User A adds user B as a connection.
   3. The connection invitation is sent to user B.
   4. If the invitation is accepted:
      1. User B is added to user A's connections.
      2. User A gets a notification that the connection invitation has been accepted.
   5. Else, the invitation is ignored.
3. Else, if user B does not exist:
   1. User A receives a "user not found" error.

Based on the order above, the sequence diagram for sending a connection invitation on LinkedIn is given below:

A diagram of a web site

Description automatically generated

Next, let’s look at the activity diagrams for LinkedIn to understand the control flow of the system.

# Activity Diagram for LinkedIn

Create some activity diagrams for the LinkedIn problem.

**We'll cover the following**

* [Creating a new post](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Creating-a-new-post)
  + [States](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#States)
  + [Actions](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Actions)
* [Activity challenge: A user creates a page](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Activity-challenge:-A-user-creates-a-page)

Activity diagrams are a great way to visualize the flow of messages from one activity to the other in the system. There can be different activity diagrams that we can create for LinkedIn. In this lesson, we’ll create activity diagrams for the following two activities:

* Creating a new post
* **Activity challenge:**A user creates a page.

## Creating a new post

The states and actions that will be involved in this activity diagram are provided below.

### States

**Initial state:**The user selects the new post option.

**Final state:** The post is published.

### Actions

The user selects the new post option, selects the privacy option, adds any attachments, and publishes the post.

A screenshot of a diagram

Description automatically generated

A screenshot of a diagram

Description automatically generated

Notice that the actions in the diagram above are numbered from 1 to 10. The slots below represent the activities, and the arrows represent the flow from one activity to the other.

Can you rearrange the slots below in the correct order they should appear in the activity diagram given above?

**Note:**If you are unsure, click the “Show Solution” button to check the correct answer.

Alternatively, you can click the "Show complete diagram" button below to see the complete activity diagram.

A screenshot of a diagram

Description automatically generated

We've looked at some of the activity diagrams of LinkedIn. In the next lesson, we will present the code for our designed classes in some of the most popular languages.

# Code for LinkedIn

Write object-oriented code to implement the design of the LinkedIn problem.

**We'll cover the following**

* [LinkedIn classes](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#LinkedIn-classes)
  + [Constants](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Constants)
  + [Account](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Account)
  + [Person, admin, and user](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Person,-admin,-and-user)
  + [Recommendation, achievement, and analytics](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Recommendation,-achievement,-and-analytics)
  + [Profile, experience, education, and skill](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Profile,-experience,-education,-and-skill)
  + [Company, job, and group](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Company,-job,-and-group)
  + [Post, comment, message, and connection invitation](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Post,-comment,-message,-and-connection-invitation)
  + [Search, catalog, and notification](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Search,-catalog,-and-notification)
* [Wrapping up](https://www.educative.io/courses/grokking-the-low-level-design-interview-using-ood-principles/getting-ready-the-amazon-online-shopping-system#Wrapping-up)

We've gone over the different aspects of LinkedIn and observed the attributes attached to the problem using various UML diagrams. Let’s explore the more practical side of things, where we will work on implementing the LinkedIn network using multiple languages. This is usually the last step in an object-oriented design interview process.

We have chosen the following languages to write the skeleton code of the different classes present in LinkedIn:

* Java
* C#
* Python
* C++
* JavaScript

## LinkedIn classes

In this section, we will provide the skeleton code of the classes designed in the class diagram lesson.

**Note:** For simplicity, we are not defining getter and setter functions. The reader can assume that all class attributes are private and accessed through their respective public getter methods and modified only through their public method functions.

### Constants

The following code provides the definition of the various enums and custom data types being used in the LinkedIn design:

**Note:** JavaScript does not support enumerations so we will be using the Object.freeze() method as an alternative that freezes an object and prevents further modifications.

Java

public class Address {

private int zipCode;

private String streetAddress;

private String city;

private String state;

private String country;

}

enum AccountStatus {

ACTIVE,

DEACTIVATED,

BLOCKED,

DELETED

}

enum ConnectionInviteStatus {

PENDING,

ACCEPTED,

IGNORED

}

enum JobStatus {

OPEN,

ON\_HOLD,

CLOSED

}

A screenshot of a computer

Description automatically generated

// Person will be an abstract class

public abstract class Person {

private String name;

private Address address;

private String email;

private String phone;

private Account account;

}

public class Admin extends Person {

public boolean blockUser(User user);

public boolean unblockUser(User user);

public boolean disablePage(CompanyPage page);

public boolean enablePage(CompanyPage page);

public boolean deleteGroup(Group group);

}

public class User extends Person {

private int userId;

private Date dateOfJoining;

private Profile profile;

private List<User> connections;

private List<User> followsUsers;

private List<CompanyPage> followCompanies;

private List<Group> joinedGroups;

private List<CompanyPage> createdPages;

private List<Group> createdGroups;

public boolean sendMessage(Message message);

public boolean sendInvite(ConnectionInvitation invite);

public boolean cancelInvite(ConnectionInvitation invite);

public boolean createPage(CompanyPage page);

public boolean deletePage(CompanyPage page);

public boolean createGroup(Group group);

public boolean deleteGroup(Group group);

public boolean createPost(Post post);

public boolean deletePost(Post post);

public boolean createComment(Comment comment);

public boolean deleteComment(Comment comment);

public boolean react(Post post);

public boolean requestRecommendation(User user);

public boolean acceptRecommendation(User user);

public boolean applyForJob(Job job);

}

A screenshot of a computer

Description automatically generated

### Profile, experience, education, and skill

The Experience, Education, and Skill classes will provide a user's personal information and make up the Profile class. The definition of these classes is given below:

public class Experience {

private String title;

private String company;

private Address location;

private Date startDate;

private Date endDate;

private String description;

}

public class Education {

private String school;

private String degree;

private Date startDate;

private Date endDate;

private String description;

}

public class Skill {

private string name;

}

public class Profile {

private String headline;

private String about;

private String gender;

private List<byte> profilePicture;

private List<byte> coverPhoto;

private List<Experience> experiences;

private List<Education> educations;

private List<Skill> skills;

private List<Achievement> achievements;

private List<Recommendation> recommendations;

private Analytics analytics;

public boolean addExperience(Experience experience);

public boolean addEducation(Education education);

public boolean addSkill(Skill skill);

public boolean addAchievement(Achievement achievement);

}

### Company, job, and group

LinkedIn users can create groups and company pages. The company page contains information about the company. The company pages will host various job postings. The Job, CompanyPage, and Group classes are shown below:

public class Job {

private int jobId;

private String jobTitle;

private Date dateOfPosting;

private String description;

private CompanyPage company;

private String employmentType;

private Address location;

private JobStatus status;

}

public class CompanyPage {

private int pageId;

private String name;

private String description;

private String type;

private int companySize;

private User createdBy;

private List<Job> jobs;

public boolean createJobPosting();

public boolean deleteJobPosting(Job job);

}

public class Group {

private int groupId;

private String name;

private String description;

private int totalMembers;

private List<User> members;

public boolean updateDescription();

}

### Post, comment, message, and connection invitation

LinkedIn users can create posts and comments. They can also send messages and connection invitations to other users. The definition of Post, Comment, Message, and ConnectionInvitation classes is given below:

public class Post {

private int postId;

private User postOwner;

private String text;

private List<byte> media;

private int totalReacts;

private int totalShares;

private List<Comment> comments;

public boolean updateText();

}

public class Comment {

private int commentId;

private User commentOwner;

private String text;

private int totalReacts;

private List<Comment> comments;

public boolean updateText();

}

public class Message {

private int messageId;

private User sender;

private List<User> recipients;

private String text;

private List<byte> media;

public boolean addRecipients(List<User> recipients);

}

public class ConnectionInvitation {

private User sender;

private User recipients;

private Date dateCreated;

private ConnectionInviteStatus status;

public boolean acceptConnection();

public boolean rejectConnection();

}

### Search, catalog, and notification

The SearchCatalog class contains information on users, company pages, groups, and jobs. It also implements the Search interface class to enable the search functionality based on the given criteria (user, company page, group, and job keywords).

The Notification class is responsible for sending notifications to users about any new messages, comments, posts, or connection invitations via the built-in notification option.

The definition of these classes is given below:

public interface Search {

// Interface method (does not have a body)

public List<User> searchUser(String name);

public List<CompanyPage> searchCompany(String name);

public List<Group> searchGroup(String name);

public List<Job> searchJob(String title);

}

public class SearchCatalog implements Search {

private HashMap<String, List<User>> users;

private HashMap<String, List<CompanyPage>> companies;

private HashMap<String, List<Group>> groups;

private HashMap<String, List<Job>> jobs;

public void addNewUser(User user);

public void addNewCompany(CompanyPage company);

public void addNewGroup(Group group);

public void addNewJob(Job job);

public List<User> searchUser(String name) {

// functionality

}

public List<CompanyPage> searchCompany(String name) {

// functionality

}

public List<Job> searchJobs(String title) {

// functionality

}

public List<Group> searchGroups(String name) {

// functionality

}

}

public class Notification {

private int notificationId;

private Date createdOn;

private String content;

public boolean sendNotification(Account account);

}

**Wrapping up**

We've explored the complete design of LinkedIn in this chapter. We've looked at how LinkedIn can be visualized using various UML diagrams and designed using object-oriented principles and design patterns.