

Data Classes

Class	Goal
User	Any user registering to the system, including admin, candidates and HRs.
Candidate	Keeps information about candidates. Note that a candidate is a user, so it has a user in it (Composition made here).
Message	The message to be sent to a candidate when an HR schedules an interview for the former. Or when a candidate apply for a company, a message is sent to the HR of the corresponding Company
Resume	All information about a candidate, meaning her/his skills.
Interview	Details about an interview are kept here including the name of the interviewee, name of the company, the date, and the status (scheduled or attained).
Application	The name of the applicant(candidate), the number of applications and a list of applied company names. Beware that each application is dedicated to only one candidate. Although not kept in the Candidate class.
Company	The name, about, type and email of a company are here. Not to mention the requirements of the company.
HR	The human resources of a company. Here there is a user representing a company, the name of the company is also kept here.

Enum type classes

Class	Goal
UserType	Specifies the type of the users. Admin, candidate or HR.
LogState	Specifies whether the user is currently logged in or logged out.
CandidateType	Specifies if the candidate is a fresh or an experienced one.
CandidateStatus	Indicates whether a candidate has been registered, has applied to any company, has been interviewed or was hired.
InterviewStatus	Indicates whether an interview is still scheduled or attended all ready.

Forms

Forms	Goals
Form <interface>: T	Grab information from users
LoginForm	Request information from users when logging into the application
CandidateForm: Candidate	To register a candidate, takes information from user and create a Candidate object which is return.
CompanyForm	To register a Company, takes information from user and create a Company object which is return.
HRForm	To register a HR, takes information from user and create a HR object which is return.
ResumeForm	When registering a candidate, this form takes the resume of the applicants to be uploaded.

UserForm	Takes personal information about Admin , Candidate and HR for registrations
----------	--

Interfaces

Interface	Classes implementing it	Helpers	Folder
Login	<ul style="list-style-type: none"> - LoginAdmin - LoginCandidate - LoginHR 	LoginHelper	domainLayer
Controller Here's where all operations doable by all users are specified.	<ul style="list-style-type: none"> - AdminController - CandidateController - HRController. 		controllers
Repository Provides the communication Between the controllers and the database.	<ul style="list-style-type: none"> - UserRepository - CandidateRepository - HRRepository - InterviewRepository - CompanyRepository - ApplicationRepository 		repositories
Reader Providing information from the database to the Repository.	<ul style="list-style-type: none"> - UserReader - CandidateReader - HRReader - ApplicationReader - CompanyReader - InterviewReader 	<ul style="list-style-type: none"> - ReaderHelper - ToStringHelper 	dataAccessLayer
	<ul style="list-style-type: none"> - Recommendation - RecommendationHelper 	- Levenshtein	domainLayer
	<ul style="list-style-type: none"> - CheckPreconditions 		domainLayer

Other Helper Classes

Class	Goal	Folder
IdGenerator	Based on the type of user, the id is generated taking into account the last id granted. This is updated.	domainLayer

Login Info for Test Purpose

Admin:

Username: Ben
Password: mlk9bEn8*

Candidate:

Username: Abdel-Kader
Password: kaderMe

HR:

Username: Bengisu
Password: akinBeng22