

ApliTrack

Final Report

Group 21

Prempreet Brar - 30112576

Jayden Yeo - 30113049

Fedor Prokopchuk - 30122005

CPSC 471: Data Base Management Systems

Ahmed Al Marouf

Saif Abuhajleh

December 12, 2023

Table of Contents

Abstract.....	4
Introduction.....	5
System Description.....	5
Project Design.....	5
Implementation.....	8
SQL Transaction Statements.....	12
Login & Signup.....	12
Profile.....	13
Documents.....	18
Companies & Jobs.....	22
Applications.....	31
Contacts.....	36
Interviews.....	44
Offers.....	56
Users (Admin).....	61
User Manual.....	63
Login & Signup.....	63
Profile.....	68
Documents.....	72
Companies & Jobs.....	76
Applications.....	86
Contacts.....	90
Interviews.....	98
Offers.....	103
Users (Admin).....	108
Additional Admin Actions.....	109

Logged-Out Views.....	112
Mobile and Tablet Views.....	115
Login Page.....	115
Sign-Up Page.....	116
Profile Page.....	118
Document Page.....	123
Company Page.....	125
Application Page.....	130
Contact Page.....	133
Interview Page.....	138
Offer Page.....	141
Users Page (Admin).....	146
Sample Data Instances.....	149
References.....	161
Appendices.....	161

Abstract

ApliTrack is a Job Application Tracking System (JATS) that aims to ease the burden of managing job applications for busy University students who find themselves applying to tens or hundreds of jobs, often due to limited professional experience and a highly competitive job market. With ApliTrack, applicants are able to manage the most relevant information pertaining to their job applications in one comprehensive solution, keeping track of components like the applications themselves, the jobs and companies they apply to, the documents they submit, the interviews and offers they receive, as well as any of their contacts who may be relevant to the jobs they apply to.

To create ApliTrack, the group followed the database design and implementation process from start to finish. First, an initial proposal and set of broad requirements was created to establish the purpose and general functionality of the web application. Then, an enhanced entity relation diagram (EERD) was created to convey all components of the application, more clearly illustrating the information that would be stored in the database and the resulting functionality that could be reasonably expected. Once the EERD was complete, it was mapped to a relational model (RM) that more closely resembled the underlying database structure that is used in the back-end of the application. Following this, the functional design was created, namely the Sequence and Use Case diagrams and the complete set of SQL statements used to implement all of ApliTrack's functionality. Finally, the final product was created by linking the back-end API with the front-end of the application.

Introduction

The ApliTrack JATS addresses University students' challenges pertaining to managing and monitoring large amounts of job applications simultaneously. The problem largely stems from students' tendency to apply to an extensive number of jobs to secure a single internship, often due to limited experience and high demand for such positions. This issue is particularly important to address because it has been found to lead to increased job-seeking anxiety and poorer mental health among undergraduate students.¹ As stated in the proposal, "not only is the [application tracking] process time-consuming, but it is also stressful and often affects other aspects of students' lives, such as their studies." Moreover, this issue has personally affected all group members, hence the decision to develop a database application to address this challenge.

System Description

ApliTrack provides job applicants with a single comprehensive solution for managing the most relevant information pertaining to their job applications. As with any system that requires storing personal and sensitive information, applicants are able to create accounts which ensure that private information pertaining to each applicant is accessible only by the applicant themselves. The Project Design section below provides a more thorough description of the functionality that ApliTrack users can expect.

Project Design

The project design section includes a thorough description of the complete transaction collection for all users of the system. Note that while the transactions, pages, and functionality are described below, the specific SQL statements relevant to each transaction are included further in the report, in the Implementation section.

The primary users of ApliTrack are the job applicants. Upon first visiting the application, applicants are prompted to register for the service by creating a unique username and a password. If an applicant is an existing user, they are evidently able to log in and access information that they have previously inputted into the system.

Upon logging in, applicants are able to navigate through different web pages to accomplish various tasks. The general functionality is described in this section, with more

detailed descriptions and examples of specific web pages provided in the User Manual further in the report.

First, applicants can store information about themselves that may be relevant to their job applications, such as their level of education, important personal or professional experiences, projects they have completed in the past, certifications they have earned, key skills they wish to highlight, and any competitions they have been a part of. If applicants wish to edit or remove any of the previously mentioned items, they are able to do so as well. This can all be done on the Profile page of ApliTrack. Additionally, the Profile page allows applicants to update their current passwords if necessary.

Next, the most important functionality of a JATS is the ability to effectively add and manage job applications, which ApliTrack provides in the Applications page. Applicants are able to add custom categories and any relevant URLs to the applications. Applications also reference the specific jobs to which the applications refer, and applicants can also specify which documents were submitted with the application.

On a separate Documents page, applicants are able to add various documents to the system that they may or may not submit with applications. ApliTrack allows for intuitive uploading of documents and specification of the document name and optional description. Additionally, applicants are able to sort and search through their documents seamlessly.

The system also stores information about companies that post jobs, allowing applicants to better prepare for interviews by checking crucial company details. The Companies page allows Applicants to add companies to the system, including various detailed fields that provide more information about each company. The same page also allows applicants to add jobs that are related to specific companies, including vital information like qualifications, responsibilities, salary, description, and even upload the job post file itself to the system for storage and easy access. The page provides convenient searching and sorting, as well as enabling applicants to track and/or display jobs that they may wish to apply to but have not done so yet. These features will be showcased in more detail further in the report.

Applicants also have the ability to add interviews to the system and link these interviews to specific applications and jobs that the interviews correspond to, all through the Interviews page. As with other pages, the Interviews page provides intuitive sorting and searching options – for example, using the interview Stage or the interview Date.

ApliTrack also allows applicants to track their contacts in the Contacts page. Basic contact details can be stored in the system and applicants can specify whether they personally know any of the contacts in the system. If so, applicants can store their relationships to their contacts, enabling users to better manage their networks. While other networking platforms exist, such as LinkedIn, ApliTrack enables applicants to track particularly those contacts that are relevant to their job applications, as opposed to their entire network, which can be overwhelming to effectively navigate. Applicants can further specify multiple phone numbers and emails for each contact, as well as the companies that the contacts work at, provided that the companies exist in the system. Within the same page, ApliTrack allows applicants to track any referrals from contacts, including any notes for the referral and the specific job the referral relates to.

Finally, applicants are able to track the offers through the Offers page, where applicants can add offers and link the offers to specific jobs. Moreover, applicants are able to specify vital details about offers like the response deadline, start date, compensation, and notes, and upload the job offer file. As with other pages previously discussed, the Offers page provides intuitive sorting and searching capabilities.

As with their personal details, applicants are able to edit information related to all the items mentioned above, or delete items if they are no longer relevant to the applicant. This functionality will be showcased in more detail in the Implementation and User Manual sections.

The only other users of ApliTrack are the admins of the system. Admins must initially register for an account in the same manner as regular applicants. Then, existing admins are able to change the status of regular users to admins, but this must be done manually through the back-end. There is no way to enable admin status through the frontend to prevent malicious users from attempting to do so.

Once admin status is enabled, admins are granted additional functionality to the abilities described above for applicants. As was previously outlined in the submitted Progress Report 3, additional admin functionality includes “creating, deactivating, reactivating, and deleting users, giving permissions to certain user accounts to “promote” them into admins, and resetting user passwords”, which can be done through the Users page. Additionally, since jobs, contacts, and companies are shared in the system across all users to prevent duplication, as these entities are not unique to a specific applicant, only admins are allowed to delete these entities from the database to prevent malicious users from making these deletions. Applicants can, however, add

these entities and edit information pertaining to the aforementioned entities, given that they already exist in the system. Finally, note that admins are assigned different permission levels in the system that determine the specific permissions and abilities granted to each individual admin.

It is important to note that the EERD included in Appendix A also includes an additional Developer user type. During the implementation phase of the system, it was decided that the Admin and Developer users are to be combined into one Admin user type, with developers simply having different or additional privileges and permissions compared to regular admins.

Appendix A includes the EERD of the entire system, which provides a comprehensive visual representation of all the components contained in the database. Note that several changes were made to the implementation of the system since the EERD was first constructed and presented. The EERD in Appendix A models applicants and admins as being disjoint; however, the current system implementation allows admins to perform all actions allowed by applicants, therefore modeling them as overlapping. This change was made to simplify the backend while maintaining the desired functionality. A more detailed discussion of this change is provided further below in the Implementation section. All other components of the EERD remained largely unchanged, other than small alterations to some of the weak entities' primary keys. Again, this change was done due to technical reasons and its discussion is therefore more appropriately included in the Implementation section, below.

Implementation

The first stage of the implementation process consisted of converting the EERD from Appendix A into a RM, which is provided in Appendix B. The standard algorithm, which was presented in class, was followed to convert the entity relationship diagram to a relational schema diagram:

- 1) A separate table was created for all regular entity types and their simple and composite attributes.
- 2) Weak entities were then mapped to separate tables, ensuring that each weak entity table included a foreign key referencing the primary key attribute of the strong parent entity, resulting in a composite key for each weak entity.
- 3) The next step typically involves mapping all binary 1:1 relationships. However, the EERD does not contain any 1:1 relationships, so this step in the process was skipped.

- 4) Next, all binary 1:N relationships were mapped by identifying the entity at the N-side of the relationship type and including in that relation a foreign key that represents the primary key of the other entity participating in the relationship. For example, there is a 1:N relationship between Applicant and Offer, so the Offer relation includes the ApplicantUsername foreign key that references the Username primary key of the Applicant relation. Note that this step also requires any simple attributes of the 1:N relationships to be included as attributes in the relations of the at the N-side of the relationship types; however, the EERD does not have any 1:N relationships with simple attributes, and this step was therefore skipped.
- 5) All M:N relationships were then mapped using the standard process of creating a new table to represent the relationship, where the new table includes as foreign keys all the primary keys of the participating entities in the relationship type. Note that any attributes of the M:N relationships were included as attributes in the respective tables representing these relationships, as required. For example, the Tracks M:N relationship includes the foreign keys of Applicant and Job, with the foreign keys acting as one composite primary key. Additionally, the Notes and DateToApply relationship attributes are also included in the Tracks relation in the RM.
- 6) Following the mapping of all binary relationships, any multivalued attributes were mapped using a new table that includes a composite primary key consisting of the foreign key referencing the relation that has the multivalued attribute, and the component attributes of the multivalued attribute.
- 7) The next step typically requires any N-ary relationships to be mapped. The EERD did not contain any N-ary relationships, so this step was skipped.
- 8) The last step applicable to the EERD was mapping specializations, such as the Admin and Applicant specializations of User and the Developer specialization of Admin. With multiple options available for mapping specializations, a combination of two options was used to achieve the final RM presented in Appendix B. The ‘Multiple Relations - Superclass and Subclasses’ option was used to create tables for User, Applicant, and Admin, where the attributes specific to each subclass were included in the subclass tables, along with a foreign key that acts as the primary key of each subclass and references the primary key of the User superclass, while the attributes specific to User

were included in the User table. The Developer specialization, however, was instead implemented in the RM by including a type attribute in the Admin table, together with the simple attributes of the Developer entity. This was done because both Admin and Developer entities have few attributes, and the Admin table would have been empty if the ‘Multiple Relations - Superclass and Subclasses’ option was used to map this specialization.

- 9) The final general step in mapping an EERD to a RM is mapping union types. No union types exist in the EERD, so this step was skipped.

It is important to highlight that several attribute names were updated in the RM from the EERD to add clarity to and remove ambiguity from the attribute names. These were:

- DocFileName for DOCUMENT, which was initially FileName
- AName for APPLICATION, which was initially Name
- CompanyName for COMPANY, which was initially Name
- OfferFileName for OFFER, which was initially FileName

Additionally, the HAS relationship was similarly renamed to ADMIN_HAS_PERM for more clarity.

The Sequence and Use Case diagrams are all provided in Appendix C, together with the diagram explanations that were originally submitted as part of Progress Report 3. Note that no changes were made to the diagrams or explanations since these were submitted, so please kindly refer to Appendix C to view these, if desired. The Sequence and Use Case diagrams helped provide additional insight into the functionality of our system, including user interactions with the frontend – context not available from the RM. As a result, the group found the creation of the Sequence and Use Case diagrams particularly useful.

After all diagrams were completed, the next stage of the implementation process consisted of selecting a DBMS and implementing the back-end of the web application. Initially, the group settled on using MySQL because this database was covered several times during tutorials throughout the semester. Thus, the RM schema was implemented exactly as presented in Appendix B.

Shortly after beginning development of the application as a whole, connecting to MySQL turned out more difficult than initially anticipated. Additionally, all team members were

experiencing difficulties finding a method to effectively synchronize the MySQL databases across all members' devices. As a result, the team ultimately made the decision to switch from MySQL to SQLite, which allowed all members to have the entire database in one single file that could be efficiently shared and worked on collaboratively during development.

The team's project consists of a Node.js² and Express.js³ backend, with a React.js⁴ frontend; both frameworks were chosen because they use JavaScript. This ensured team members did not have to learn multiple programming languages. To speed up development, we chose to use an ORM (Object Relational Mapper) to help with our queries. After researching online, we chose Sequelize⁵ because of its easy-to-use documentation and online support articles that helped with troubleshooting issues.

Unfortunately, while switching to Sequelize helped resolve the initial issues discussed above, it also presented some new challenges related to the RM schema. More specifically, Sequelize was unable to correctly process and implement foreign keys that reference composite primary keys. This proved to be an issue only for weak entities that were previously defined with composite keys and which were then referenced by other relations in the database. To address this, weak entities that were affected by the issue were instead given a new primary key in the form of a unique auto-assigned and incremented integer ID, enabling them to have a single primary key that can be referenced. However, to ensure that the original functionality was preserved, attributes that previously comprised the composite key were assigned unique tags, ensuring that the combination of these attributes is unique for any instance in the relation, thereby mimicking the functionality of a composite primary key. This change was applied to the following entities in the database:

- Interview
- Document
- Application
- Job

Finally, note that the final implementation of the system includes the following tables that are not being used:

- Admin
- Admin_Responsibility

- Admin_Has_Permit
- Dev_Specialization

While implementing functionality, the group realized that these tables were redundant and that combining the Admin flag and PermissionLevel in the User table simplified the backend while maintaining desired functionality. Although the tables are not used, it is worth noting that they remain in the database, with all Rest API routes defined and functional, for completeness. The Developer flag was then removed and Admin_Responsibility and Dev_Specialization tables no longer used because a developer's and admin's permissions and abilities are normally managed in the GitHub repository, and not in the application itself, as was initially intended in the database design stage.

With all issues addressed, the team was then able to implement the back-end and connect it to the front-end of the application. The Postman API platform was used to first implement all backend routes and requests that enable the application to function correctly, after which the team moved onto front-end development. Note that authorization is done on both the backend and front-end. For example, only admins can delete companies. On the front-end, the delete button is visible to *only* admins, while on the backend, we check the hasAdminFlag before processing the request.

SQL Transaction Statements

All SQL statements related to the implemented transactions are included below, separated by application pages.

Login & Signup

- Upon registering for the first time:

Executing (default): `INSERT INTO 'USER'`

`(`Username`, `Password`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel`) VALUES ($1,$2,$3,$4,$5,$6,$7);`

Executing (default): `INSERT INTO 'APPLICANT' ('Username') VALUES ($1);`

- Upon logging in:

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel`, `Password` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Profile

- Loading Profile page

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): SELECT `APPLICANT`.`Username`, `APPLICANT`.`Education`, `Experiences`.`id` AS `Experiences.id`, `Experiences`.`ApplicantUsername` AS `Experiences.ApplicantUsername`, `Experiences`.`Experience` AS `Experiences.Experience`, `Experiences`.`ExperienceDesc` AS `Experiences.ExperienceDesc`, `Projects`.`id` AS `Projects.id`, `Projects`.`ApplicantUsername` AS `Projects.ApplicantUsername`, `Projects`.`Project` AS `Projects.Project`, `Projects`.`ProjectDesc` AS `Projects.ProjectDesc`, `Certifications`.`ApplicantUsername` AS `Certifications.ApplicantUsername`, `Certifications`.`Certification` AS `Certifications.Certification`, `Skills`.`ApplicantUsername` AS `Skills.ApplicantUsername`, `Skills`.`Skill` AS `Skills.Skill`, `Competitions`.`ApplicantUsername` AS `Competitions.ApplicantUsername`, `Competitions`.`Competition` AS `Competitions.Competition`, `Documents`.`DocumentID` AS `Documents.DocumentID`, `Documents`.`ApplicantUsername` AS `Documents.ApplicantUsername`, `Documents`.`DocFileName` AS `Documents.DocFileName`, `Documents`.`DocType` AS `Documents.DocType`, `Documents`.`Description` AS `Documents.Description`, `Applications`.`ApplicationID` AS `Applications.ApplicationID`, `Applications`.`ApplicantUsername` AS `Applications.ApplicantUsername`, `Applications`.`AName` AS `Applications.AName`, `Applications`.`Notes` AS `Applications.Notes`, `Applications`.`DateSubmitted` AS `Applications.DateSubmitted`, `Applications`.`Status` AS `Applications.Status`, `Interviews`.`InterviewID` AS `Interviews.InterviewID`, `Interviews`.`ApplicantUsername` AS `Interviews.ApplicantUsername`, `Interviews`.`Stage` AS `Interviews.Stage`, `Interviews`.`Date` AS `Interviews.Date`, `Interviews`.`ApplicationID` AS `Interviews.ApplicationID`,

```

`User`.`Username` AS `User.Username`, `User`.`PasswordConfirm` AS
`User.PasswordConfirm`, `User`.`Fname` AS `User.Fname`, `User`.`Lname` AS `User.Lname`,
`User`.`IsActive` AS `User.IsActive`, `User`.`AdminFlag` AS `User.AdminFlag`,
`User`.`PermissionLevel` AS `User.PermissionLevel`, `Contacts`.`ContactID` AS
`Contacts.ContactID`, `Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`,
`Contacts`.`Fname` AS `Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->KNOWS`.`Relationship` AS `Contacts.KNOWS.Relationship`,
`Contacts->KNOWS`.`Notes` AS `Contacts.KNOWS.Notes`,
`Contacts->KNOWS`.`LastContactDate` AS `Contacts.KNOWS.LastContactDate`,
`Contacts->KNOWS`.`Username` AS `Contacts.KNOWS.Username`,
`Contacts->KNOWS`.`ContactID` AS `Contacts.KNOWS.ContactID`, `Jobs`.`PositionID` AS
`Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline` AS
`Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`, `Jobs->TRACKS`.`Notes` AS
`Jobs.TRACKS.Notes`, `Jobs->TRACKS`.`DateToApply` AS `Jobs.TRACKS.DateToApply`,
`Jobs->TRACKS`.`Username` AS `Jobs.TRACKS.Username`, `Jobs->TRACKS`.`PositionID` AS
`Jobs.TRACKS.PositionID`, `OFFERs`.`OfferID` AS `OFFERs.OfferID`,
`OFFERs`.`ApplicantUsername` AS `OFFERs.ApplicantUsername`, `OFFERs`.`OfferFileName` AS
`OFFERs.OfferFileName`, `OFFERs`.`ResponseDeadline` AS `OFFERs.ResponseDeadline`,
`OFFERs`.`Compensation` AS `OFFERs.Compensation`, `OFFERs`.`StartDate` AS
`OFFERs.StartDate`, `OFFERs`.`Notes` AS `OFFERs.Notes`, `OFFERs`.`PositionID` AS
`OFFERs.PositionID`, `Referrals`.`ReferralID` AS `Referrals.ReferralID`,
`Referrals`.`ContactID` AS `Referrals.ContactID`, `Referrals`.`Date` AS `Referrals.Date`,
`Referrals`.`Notes` AS `Referrals.Notes`, `Referrals`.`ApplicantUsername` AS
`Referrals.ApplicantUsername`, `Referrals`.`PositionID` AS `Referrals.PositionID` FROM
`APPLICANT` AS `APPLICANT` LEFT OUTER JOIN `APPLICANT_EXPERIENCE` AS
`Experiences` ON `APPLICANT`.`Username` = `Experiences`.`ApplicantUsername` LEFT
OUTER JOIN `APPLICANT_PROJECT` AS `Projects` ON `APPLICANT`.`Username` =
`Projects`.`ApplicantUsername` LEFT OUTER JOIN `APPLICANT_CERTIFICATION` AS
`Certifications` ON `APPLICANT`.`Username` = `Certifications`.`ApplicantUsername` LEFT

```

```

OUTER JOIN `APPLICANT_SKILL` AS `Skills` ON `APPLICANT`.`Username` =
`Skills`.`ApplicantUsername` LEFT OUTER JOIN `APPLICANT_COMPETITION` AS
`Competitions` ON `APPLICANT`.`Username` = `Competitions`.`ApplicantUsername` LEFT
OUTER JOIN `DOCUMENT` AS `Documents` ON `APPLICANT`.`Username` =
`Documents`.`ApplicantUsername` LEFT OUTER JOIN `APPLICATION` AS `Applications` ON
`APPLICANT`.`Username` = `Applications`.`ApplicantUsername` LEFT OUTER JOIN
`INTERVIEW` AS `Interviews` ON `APPLICANT`.`Username` =
`Interviews`.`ApplicantUsername` LEFT OUTER JOIN `USER` AS `User` ON
`APPLICANT`.`Username` = `User`.`Username` LEFT OUTER JOIN ( `KNOWS` AS
`Contacts->KNOWS` INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->KNOWS`.`ContactID` ) ON `APPLICANT`.`Username` =
`Contacts->KNOWS`.`Username` LEFT OUTER JOIN ( `TRACKS` AS `Jobs->TRACKS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` = `Jobs->TRACKS`.`PositionID` ) ON
`APPLICANT`.`Username` = `Jobs->TRACKS`.`Username` LEFT OUTER JOIN `OFFER` AS
`OFFERs` ON `APPLICANT`.`Username` = `OFFERs`.`ApplicantUsername` LEFT OUTER
JOIN `REFERRAL` AS `Referrals` ON `APPLICANT`.`Username` =
`Referrals`.`ApplicantUsername` WHERE `APPLICANT`.`Username` =
'fredprokop@gmail.com';

```

- Updating First Name, Last Name, Education

Executing (default): `SELECT `Username`, `Education` FROM `APPLICANT` AS `APPLICANT` WHERE `APPLICANT`.`Username` = 'fredprokop@gmail.com';`

Executing (default): `UPDATE `USER` SET `Password`=$1, `Lname`=$2 WHERE `Username` = $3`

Executing (default): `UPDATE `APPLICANT` SET `Education`=$1 WHERE `Username` = $2`

- Add Experience

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';`

Executing (default): `INSERT INTO `APPLICANT_EXPERIENCE`
(`id`, `ApplicantUsername`, `Experience`, `ExperienceDesc`) VALUES (NULL,$1,$2,$3);`

- Add Project

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop@gmail.com';`

Executing (default): `INSERT INTO `APPLICANT_PROJECT``

`(`id`, `ApplicantUsername`, `Project`, `ProjectDesc`) VALUES (NULL,$1,$2,$3);`

- Add Certification

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop@gmail.com';`

Executing (default): `INSERT INTO `APPLICANT_CERTIFICATION``

`(`ApplicantUsername`, `Certification`) VALUES ($1,$2);`

- Add Skill

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop@gmail.com';`

Executing (default): `INSERT INTO `APPLICANT_SKILL` (`ApplicantUsername`, `Skill`)`

`VALUES ($1,$2);`

- Add Competition

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop@gmail.com';`

Executing (default): `INSERT INTO `APPLICANT_COMPETITION``

`(`ApplicantUsername`, `Competition`) VALUES ($1,$2);`

- Edit Experience

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): SELECT `id`, `ApplicantUsername`, `Experience`, `ExperienceDesc` FROM `APPLICANT_EXPERIENCE` AS `APPLICANT_EXPERIENCE` WHERE `APPLICANT_EXPERIENCE`.`Experience` = 'ATB' AND `APPLICANT_EXPERIENCE`.`ApplicantUsername` = 'fredprokop@gmail.com' LIMIT 1;

Executing (default): UPDATE `APPLICANT_EXPERIENCE` SET `Experience`=\$1, `ExperienceDesc`=\$2 WHERE `id` = \$3

- Edit Project

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): SELECT `id`, `ApplicantUsername`, `Project`, `ProjectDesc` FROM `APPLICANT_PROJECT` AS `APPLICANT_PROJECT` WHERE `APPLICANT_PROJECT`.`Project` = 'CPSC 471 Project' AND `APPLICANT_PROJECT`.`ApplicantUsername` = 'fredprokop@gmail.com' LIMIT 1;

Executing (default): UPDATE `APPLICANT_PROJECT` SET `Project`=\$1, `ProjectDesc`=\$2 WHERE `id` = \$3

- Delete Experience

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): DELETE FROM `APPLICANT_EXPERIENCE` WHERE `Experience` = 'ATB Updated' AND `ApplicantUsername` = 'fredprokop@gmail.com'

- Delete Project

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): DELETE FROM `APPLICANT_PROJECT` WHERE `Project` = CPSC
471 Project Updated' AND `ApplicantUsername` = 'fredprokop@gmail.com'

- Delete Certification

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): DELETE FROM `APPLICANT_CERTIFICATION` WHERE
`Certification` = 'Respect in the Workplace' AND `ApplicantUsername` =
'fredprokop@gmail.com'

- Delete Skill

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): DELETE FROM `APPLICANT_SKILL` WHERE `Skill` = 'Due diligence'
AND `ApplicantUsername` = 'fredprokop@gmail.com'

- Delete Competition

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): DELETE FROM `APPLICANT_COMPETITION` WHERE `Competition` = 'TRMC2023 Case Competition' AND `ApplicantUsername` = 'fredprokop@gmail.com'

Documents

- Loading Documents page

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Document File Name! undefined

Executing (default): SELECT `DOCUMENT`.`DocumentID`,
`DOCUMENT`.`ApplicantUsername`, `DOCUMENT`.`DocFileName`,
`DOCUMENT`.`DocType`, `DOCUMENT`.`Description`, `Applicant`.`Username` AS
`Applicant.Username`, `Applicant`.`Education` AS `Applicant.Education`,
`Applications`.`ApplicationID` AS `Applications.ApplicationID`,
`Applications`.`ApplicantUsername` AS `Applications.ApplicantUsername`,
`Applications`.`AName` AS `Applications.AName`, `Applications`.`Notes` AS
`Applications.Notes`, `Applications`.`DateSubmitted` AS `Applications.DateSubmitted`,
`Applications`.`Status` AS `Applications.Status`,
`Applications->SUBMIT_WITH`.`ApplicationID` AS
`Applications.SUBMIT_WITH.ApplicationID`,
`Applications->SUBMIT_WITH`.`DocumentID` AS
`Applications.SUBMIT_WITH.DocumentID` FROM `DOCUMENT` AS `DOCUMENT` LEFT
OUTER JOIN `APPLICANT` AS `Applicant` ON `DOCUMENT`.`ApplicantUsername` =
`Applicant`.`Username` LEFT OUTER JOIN (`SUBMIT_WITH` AS
`Applications->SUBMIT_WITH` INNER JOIN `APPLICATION` AS `Applications` ON
`Applications`.`ApplicationID` = `Applications->SUBMIT_WITH`.`ApplicationID`) ON
`DOCUMENT`.`DocumentID` = `Applications->SUBMIT_WITH`.`DocumentID` ORDER BY
`DOCUMENT`.`DocFileName` ASC;

- Add document

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): INSERT INTO `DOCUMENT`
(`DocumentID`, `ApplicantUsername`, `DocFileName`, `DocType`, `Description`) VALUES
(NULL,\$1,\$2,\$3,\$4);

- Update document

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop@gmail.com';

Executing (default): SELECT `DocumentID`, `ApplicantUsername`, `DocFileName`, `DocType`, `Description` FROM `DOCUMENT` AS `DOCUMENT` WHERE `DOCUMENT`.`DocumentID` = '1';

Executing (default): UPDATE `DOCUMENT` SET `Description`=\$1 WHERE `DocumentID` = \$2

- Delete Document

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'admin@admin.com';

Executing (default): DELETE FROM `DOCUMENT` WHERE `DocumentID` = 4 AND `ApplicantUsername` = 'admin@admin.com'

- Search for Document file name

Executing (default): SELECT `DOCUMENT`.`DocumentID`,
`DOCUMENT`.`ApplicantUsername`, `DOCUMENT`.`DocFileName`,
`DOCUMENT`.`DocType`, `DOCUMENT`.`Description`, `Applicant`.`Username` AS
`Applicant.Username`, `Applicant`.`Education` AS `Applicant.Education`,
`Applications`.`ApplicationID` AS `Applications.ApplicationID`,
`Applications`.`ApplicantUsername` AS `Applications.ApplicantUsername`,
`Applications`.`AName` AS `Applications.AName`, `Applications`.`Notes` AS
`Applications.Notes`, `Applications`.`DateSubmitted` AS `Applications.DateSubmitted`,
`Applications`.`Status` AS `Applications.Status`,
`Applications->SUBMIT_WITH`.`ApplicationID` AS
`Applications.SUBMIT_WITH.ApplicationID`,
`Applications->SUBMIT_WITH`.`DocumentID` AS

```

`Applications.SUBMIT_WITH.DocumentID` FROM `DOCUMENT` AS `DOCUMENT` LEFT
OUTER JOIN `APPLICANT` AS `Applicant` ON `DOCUMENT`.`ApplicantUsername` =
`Applicant`.`Username` LEFT OUTER JOIN ( `SUBMIT_WITH` AS
`Applications->SUBMIT_WITH` INNER JOIN `APPLICATION` AS `Applications` ON
`Applications`.`ApplicationID` = `Applications->SUBMIT_WITH`.`ApplicationID`) ON
`DOCUMENT`.`DocumentID` = `Applications->SUBMIT_WITH`.`DocumentID` WHERE
`DOCUMENT`.`DocFileName` LIKE '%Cover Letter%' ORDER BY
`DOCUMENT`.`DocFileName` ASC;

```

- Apply Descending sort by Document file name

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
`fredprokop@gmail.com`;

Document File Name!

Executing (default): SELECT `DOCUMENT`.`DocumentID`,
`DOCUMENT`.`ApplicantUsername`, `DOCUMENT`.`DocFileName`,
`DOCUMENT`.`DocType`, `DOCUMENT`.`Description`, `Applicant`.`Username` AS
`Applicant.Username`, `Applicant`.`Education` AS `Applicant.Education`,
`Applications`.`ApplicationID` AS `Applications.ApplicationID`,
`Applications`.`ApplicantUsername` AS `Applications.ApplicantUsername`,
`Applications`.`AName` AS `Applications.AName`, `Applications`.`Notes` AS
`Applications.Notes`, `Applications`.`DateSubmitted` AS `Applications.DateSubmitted`,
`Applications`.`Status` AS `Applications.Status`,
`Applications->SUBMIT_WITH`.`ApplicationID` AS
`Applications.SUBMIT_WITH.ApplicationID`,
`Applications->SUBMIT_WITH`.`DocumentID` AS
`Applications.SUBMIT_WITH.DocumentID` FROM `DOCUMENT` AS `DOCUMENT` LEFT
OUTER JOIN `APPLICANT` AS `Applicant` ON `DOCUMENT`.`ApplicantUsername` =
`Applicant`.`Username` LEFT OUTER JOIN (`SUBMIT_WITH` AS
`Applications->SUBMIT_WITH` INNER JOIN `APPLICATION` AS `Applications` ON
`Applications`.`ApplicationID` = `Applications->SUBMIT_WITH`.`ApplicationID`) ON

```
'DOCUMENT'.`DocumentID` = `Applications->SUBMIT_WITH`.`DocumentID` ORDER BY  
'DOCUMENT'.`DocFileName` DESC;
```

- Apply Ascending sort by Document file name

Executing (default):
SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
'AdminFlag', `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop@gmail.com';

Document File Name!

Executing (default):
SELECT `DOCUMENT`.`DocumentID`,
'DOCUMENT'.`ApplicantUsername`, `DOCUMENT`.`DocFileName`,
'DOCUMENT'.`DocType`, `DOCUMENT`.`Description`, `Applicant`.`Username` AS
'Applicant.Username', `Applicant`.`Education` AS `Applicant.Education`,
'Applications`.`ApplicationID` AS `Applications.ApplicationID`,
'Applications`.`ApplicantUsername` AS `Applications.ApplicantUsername`,
'Applications`.`AName` AS `Applications.AName`, `Applications`.`Notes` AS
'Applications.Notes', `Applications`.`DateSubmitted` AS `Applications.DateSubmitted`,
'Applications`.`Status` AS `Applications.Status`,
'Applications->SUBMIT_WITH`.`ApplicationID` AS
'Applications.SUBMIT_WITH.ApplicationID',
'Applications->SUBMIT_WITH`.`DocumentID` AS
'Applications.SUBMIT_WITH.DocumentID' FROM `DOCUMENT` AS `DOCUMENT` LEFT
OUTER JOIN `APPLICANT` AS `Applicant` ON `DOCUMENT`.`ApplicantUsername` =
'Applicant'.`Username` LEFT OUTER JOIN (`SUBMIT_WITH` AS
'Applications->SUBMIT_WITH` INNER JOIN `APPLICATION` AS `Applications` ON
'Applications`.`ApplicationID` = `Applications->SUBMIT_WITH`.`ApplicationID`) ON
'DOCUMENT'.`DocumentID` = `Applications->SUBMIT_WITH`.`DocumentID` ORDER BY
'DOCUMENT'.`DocFileName` ASC;

Companies & Jobs

- Loading Companies page

Executing (default): SELECT `COMPANY`.`CompanyName`, `COMPANY`.`Industry`,
`COMPANY`.`HomePageURL`, `COMPANY`.`Description`, `Jobs`.`PositionID` AS
`Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline`
AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`, `Contacts`.`ContactID` AS
`Contacts.ContactID`, `Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`,
`Contacts`.`Fname` AS `Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->WORKS_AT`.`Role` AS `Contacts.WORKS_AT.Role`,
`Contacts->WORKS_AT`.`ContactID` AS `Contacts.WORKS_AT.ContactID`,
`Contacts->WORKS_AT`.`CompanyName` AS `Contacts.WORKS_AT.CompanyName` FROM
`COMPANY` AS `COMPANY` LEFT OUTER JOIN `JOB` AS `Jobs` ON
`COMPANY`.`CompanyName` = `Jobs`.`CompanyName` LEFT OUTER JOIN (`WORKS_AT`
AS `Contacts->WORKS_AT` INNER JOIN `CONTACT` AS `Contacts` ON
`Contacts`.`ContactID` = `Contacts->WORKS_AT`.`ContactID`) ON
`COMPANY`.`CompanyName` = `Contacts->WORKS_AT`.`CompanyName` ORDER BY
`COMPANY`.`CompanyName` ASC;

- Add Company

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
`fredprokop1@gmail.com`;

Executing (default): INSERT INTO `COMPANY`

(`CompanyName`, `Industry`, `HomePageURL`, `Description`) VALUES (\$1,\$2,\$3,\$4);

- Update Company

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
`admin@admin.com`;

Executing (default): SELECT `CompanyName`, `Industry`, `HomePageURL`, `Description`
FROM `COMPANY` AS `COMPANY` WHERE `COMPANY`.`CompanyName` = 'RBC';

Executing (default): UPDATE `COMPANY` SET
'Industry'=\$1,'Description'=\$2,'HomePageURL'=\$3 WHERE `CompanyName` = \$4

- Delete Company

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'admin@admin.com';

Executing (default): DELETE FROM `COMPANY` WHERE `CompanyName` = 'RBC'

- Create Job

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `JOB`

(`PositionID`,`CompanyName`,`PositionName`,`Description`,`JobPostFile`,`PositionType`,`ApplicationDeadline`,`Salary`) VALUES (NULL,\$1,\$2,\$3,\$4,\$5,\$6,\$7);

- Get Job

Executing (default): SELECT `JOB`.`PositionID`, `JOB`.`CompanyName`,
 `JOB`.`PositionName`, `JOB`.`Description`, `JOB`.`JobPostFile`, `JOB`.`PositionType`,
 `JOB`.`ApplicationDeadline`, `JOB`.`Salary`, `Qualifications`.`PositionID` AS
 `Qualifications.PositionID`, `Qualifications`.`Qualification` AS `Qualifications.Qualification`,
 `Responsibilities`.`PositionID` AS `Responsibilities.PositionID`,
 `Responsibilities`.`Responsibility` AS `Responsibilities.Responsibility`,
 `Companies`.`CompanyName` AS `Companies.CompanyName`, `Companies`.`Industry` AS
 `Companies.Industry`, `Companies`.`HomePageURL` AS `Companies.HomePageURL`,
 `Companies`.`Description` AS `Companies.Description`, `Interviews`.`InterviewID` AS
 `Interviews.InterviewID`, `Interviews`.`ApplicantUsername` AS
 `Interviews.ApplicantUsername`, `Interviews`.`Stage` AS `Interviews.Stage`, `Interviews`.`Date`
 AS `Interviews.Date`, `Interviews`.`ApplicationID` AS `Interviews.ApplicationID`,
 `Interviews->MENTIONS`.`PositionID` AS `Interviews.MENTIONS.PositionID`,

```

`Interviews->MENTIONS`.`InterviewID` AS `Interviews.MENTIONS.InterviewID`,
`Applications`.`ApplicationID` AS `Applications.ApplicationID`,
`Applications`.`ApplicantUsername` AS `Applications.ApplicantUsername`,
`Applications`.`AName` AS `Applications.AName`, `Applications`.`Notes` AS
`Applications.Notes`, `Applications`.`DateSubmitted` AS `Applications.DateSubmitted`,
`Applications`.`Status` AS `Applications.Status`,
`Applications->CORRESPONDS_TO`.`ApplicationID` AS
`Applications.CORRESPONDS_TO.ApplicationID`,
`Applications->CORRESPONDS_TO`.`PositionID` AS
`Applications.CORRESPONDS_TO.PositionID`,
`Applications->CORRESPONDS_TO`.`JobPostURL` AS
`Applications.CORRESPONDS_TO.JobPostURL`, `Applicants`.`Username` AS
`Applicants.Username`, `Applicants`.`Education` AS `Applicants.Education`,
`Applicants->TRACKS`.`Notes` AS `Applicants.TRACKS.Notes`,
`Applicants->TRACKS`.`DateToApply` AS `Applicants.TRACKS.DateToApply`,
`Applicants->TRACKS`.`Username` AS `Applicants.TRACKS.Username`,
`Applicants->TRACKS`.`PositionID` AS `Applicants.TRACKS.PositionID`,
`OFFERs`.`OfferID` AS `OFFERs.OfferID`, `OFFERs`.`ApplicantUsername` AS
`OFFERs.ApplicantUsername`, `OFFERs`.`OfferFileName` AS `OFFERs.OfferFileName`,
`OFFERs`.`ResponseDeadline` AS `OFFERs.ResponseDeadline`, `OFFERs`.`Compensation` AS
`OFFERs.Compensation`, `OFFERs`.`StartDate` AS `OFFERs.StartDate`, `OFFERs`.`Notes` AS
`OFFERs.Notes`, `OFFERs`.`PositionID` AS `OFFERs.PositionID`, `Referrals`.`ReferralID` AS
`Referrals.ReferralID`, `Referrals`.`ContactID` AS `Referrals.ContactID`, `Referrals`.`Date` AS
`Referrals.Date`, `Referrals`.`Notes` AS `Referrals.Notes`, `Referrals`.`ApplicantUsername` AS
`Referrals.ApplicantUsername`, `Referrals`.`PositionID` AS `Referrals.PositionID` FROM
`JOB` AS `JOB` LEFT OUTER JOIN `JOB_QUALIFICATION` AS `Qualifications` ON
`JOB`.`PositionID` = `Qualifications`.`PositionID` LEFT OUTER JOIN
`JOB_RESPONSIBILITY` AS `Responsibilities` ON `JOB`.`PositionID` =
`Responsibilities`.`PositionID` LEFT OUTER JOIN `COMPANY` AS `Companies` ON
`JOB`.`CompanyName` = `Companies`.`CompanyName` LEFT OUTER JOIN ( `MENTIONS` AS
`Interviews->MENTIONS` INNER JOIN `INTERVIEW` AS `Interviews` ON

```

```

`Interviews`.`InterviewID` = `Interviews->MENTIONS`.`InterviewID`) ON `JOB`.`PositionID` = `Interviews->MENTIONS`.`PositionID` LEFT OUTER JOIN (
`Applications->CORRESPONDS_TO` INNER JOIN `APPLICATION` AS `Applications` ON `Applications`.`ApplicationID` = `Applications->CORRESPONDS_TO`.`ApplicationID`)
ON `JOB`.`PositionID` = `Applications->CORRESPONDS_TO`.`PositionID` LEFT OUTER JOIN (
`TRACKS` AS `Applicants->TRACKS` INNER JOIN `APPLICANT` AS `Applicants` ON `Applicants`.`Username` = `Applicants->TRACKS`.`Username`)
ON `JOB`.`PositionID` = `Applicants->TRACKS`.`PositionID` LEFT OUTER JOIN `OFFER` AS `OFFERS` ON `JOB`.`PositionID` = `OFFERS`.`PositionID` LEFT OUTER JOIN `REFERRAL` AS `Referrals` ON `JOB`.`PositionID` = `Referrals`.`PositionID` WHERE `JOB`.`PositionID` = '2';

```

- Update Job

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `PositionID`, `CompanyName`, `PositionName`, `Description`, `JobPostFile`, `PositionType`, `ApplicationDeadline`, `Salary` FROM `JOB` AS `JOB` WHERE `JOB`.`PositionID` = '2';

Executing (default): UPDATE `JOB` SET `Salary`=\$1, `Description`=\$2, `JobPostFile`=\$3 WHERE `PositionID` = \$4

- Delete Job

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'admin@admin.com';

Executing (default): DELETE FROM `JOB` WHERE `PositionID` = 2

- Add Qualification

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): `INSERT INTO `JOB_QUALIFICATION` (`PositionID`, `Qualification`)
VALUES ($1,$2);`

- Add Responsibility

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
'AdminFlag', `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';`

Executing (default): `INSERT INTO `JOB_RESPONSIBILITY` (`PositionID`, `Responsibility`)
VALUES ($1,$2);`

- Delete Qualification

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
'AdminFlag', `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';`

Executing (default): `DELETE FROM `JOB_QUALIFICATION` WHERE `PositionID` = 3
AND `Qualification` = 'Minimum 2 years of post-secondary degree complete'`

- Delete Responsibility

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
'AdminFlag', `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';`

Executing (default): `DELETE FROM `JOB_RESPONSIBILITY` WHERE `PositionID` = 3
AND `Responsibility` = 'Build and monitor financial models'`

- Track Job

Executing (default): `SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
'AdminFlag', `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';`

Executing (default): `INSERT INTO `TRACKS` (`Username`, `PositionID`) VALUES ($1,$2);`

- Untrack Job

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `TRACKS` WHERE `PositionID` = 3 AND `Username` = 'fredprokop1@gmail.com'

- Update Tracks Relationship

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `Notes`, `DateToApply`, `Username`, `PositionID` FROM `TRACKS` AS `TRACKS` WHERE `TRACKS`.`PositionID` = 3 AND `TRACKS`.`Username` = 'fredprokop1@gmail.com';

Executing (default): UPDATE `TRACKS` SET `Notes`=\$1, `DateToApply`=\$2 WHERE `Username` = \$3 AND `PositionID` = \$4

- Search by CompanyName

Executing (default): SELECT `COMPANY`.`CompanyName`, `COMPANY`.`Industry`, `COMPANY`.`HomePageURL`, `COMPANY`.`Description`, `Jobs`.`PositionID` AS `Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS `Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS `Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline` AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`, `Contacts`.`ContactID` AS `Contacts.ContactID`, `Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS `Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`, `Contacts->WORKS_AT`.`Role` AS `Contacts.WORKS_AT.Role`, `Contacts->WORKS_AT`.`ContactID` AS `Contacts.WORKS_AT.ContactID`, `Contacts->WORKS_AT`.`CompanyName` AS `Contacts.WORKS_AT.CompanyName` FROM `COMPANY` AS `COMPANY` LEFT OUTER JOIN `JOB` AS `Jobs` ON `COMPANY`.`CompanyName` = `Jobs`.`CompanyName` LEFT OUTER JOIN (`WORKS_AT` AS `Contacts->WORKS_AT` INNER JOIN `CONTACT` AS `Contacts` ON

```

`Contacts`.`ContactID` = `Contacts->WORKS_AT`.`ContactID`) ON
`COMPANY`.`CompanyName` = `Contacts->WORKS_AT`.`CompanyName` WHERE
`COMPANY`.`CompanyName` LIKE "%IBM%" ORDER BY `COMPANY`.`CompanyName` ASC;

```

- Search by Application Deadline; note that specific jobs meeting the criteria are determined using controllers in the back-end.

Executing (default):

```

SELECT `COMPANY`.`CompanyName`, `COMPANY`.`Industry`,
`COMPANY`.`HomePageURL`, `COMPANY`.`Description`, `Jobs`.`PositionID` AS
`Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline` AS
`Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`, `Contacts`.`ContactID` AS
`Contacts.ContactID`, `Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`,
`Contacts`.`Fname` AS `Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->WORKS_AT`.`Role` AS `Contacts.WORKS_AT.Role`,
`Contacts->WORKS_AT`.`ContactID` AS `Contacts.WORKS_AT.ContactID`,
`Contacts->WORKS_AT`.`CompanyName` AS `Contacts.WORKS_AT.CompanyName` FROM
`COMPANY` AS `COMPANY` LEFT OUTER JOIN `JOB` AS `Jobs` ON
`COMPANY`.`CompanyName` = `Jobs`.`CompanyName` LEFT OUTER JOIN (`WORKS_AT` AS
`Contacts->WORKS_AT` INNER JOIN `CONTACT` AS `Contacts` ON
`Contacts`.`ContactID` = `Contacts->WORKS_AT`.`ContactID`) ON
`COMPANY`.`CompanyName` = `Contacts->WORKS_AT`.`CompanyName` ORDER BY
`COMPANY`.`CompanyName` ASC;

```

- Sort by Company Ascending

Executing (default):

```

SELECT `COMPANY`.`CompanyName`, `COMPANY`.`Industry`,
`COMPANY`.`HomePageURL`, `COMPANY`.`Description`, `Jobs`.`PositionID` AS
`Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline` AS
`Jobs.ApplicationDeadline` ORDER BY `COMPANY`.`CompanyName` ASC;

```

```

AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`, `Contacts`.`ContactID` AS
`Contacts.ContactID`, `Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`,
`Contacts`.`Fname` AS `Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->WORKS_AT`.`Role` AS `Contacts.WORKS_AT.Role`,
`Contacts->WORKS_AT`.`ContactID` AS `Contacts.WORKS_AT.ContactID`,
`Contacts->WORKS_AT`.`CompanyName` AS `Contacts.WORKS_AT.CompanyName` FROM
`COMPANY` AS `COMPANY` LEFT OUTER JOIN `JOB` AS `Jobs` ON
`COMPANY`.`CompanyName` = `Jobs`.`CompanyName` LEFT OUTER JOIN ( `WORKS_AT`
AS `Contacts->WORKS_AT` INNER JOIN `CONTACT` AS `Contacts` ON
`Contacts`.`ContactID` = `Contacts->WORKS_AT`.`ContactID`) ON
`COMPANY`.`CompanyName` = `Contacts->WORKS_AT`.`CompanyName` ORDER BY
`COMPANY`.`CompanyName` ASC;

```

- Sort by Company Descending

Executing (default):

```

SELECT `COMPANY`.`CompanyName`, `COMPANY`.`Industry`,
`COMPANY`.`HomePageURL`, `COMPANY`.`Description`, `Jobs`.`PositionID` AS
`Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline` AS
`Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`, `Contacts`.`ContactID` AS
`Contacts.ContactID`, `Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`,
`Contacts`.`Fname` AS `Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->WORKS_AT`.`Role` AS `Contacts.WORKS_AT.Role`,
`Contacts->WORKS_AT`.`ContactID` AS `Contacts.WORKS_AT.ContactID`,
`Contacts->WORKS_AT`.`CompanyName` AS `Contacts.WORKS_AT.CompanyName` FROM
`COMPANY` AS `COMPANY` LEFT OUTER JOIN `JOB` AS `Jobs` ON
`COMPANY`.`CompanyName` = `Jobs`.`CompanyName` LEFT OUTER JOIN ( `WORKS_AT`
AS `Contacts->WORKS_AT` INNER JOIN `CONTACT` AS `Contacts` ON
`Contacts`.`ContactID` = `Contacts->WORKS_AT`.`ContactID`) ON
`COMPANY`.`CompanyName` = `Contacts->WORKS_AT`.`CompanyName` ORDER BY
`COMPANY`.`CompanyName` DESC;

```

- Only Show Tracked Jobs; for each tracked job:

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `Notes`, `DateToApply`, `Username`, `PositionID` FROM `TRACKS` AS `TRACKS`;

- Show all jobs (not just Tracked jobs):

Same statement that is executed upon first opening / loading Companies page.

Applications

- Loading Applications page

Executing (default): SELECT `APPLICATION`.`ApplicationID`,
`APPLICATION`.`ApplicantUsername`, `APPLICATION`.`AName`, `APPLICATION`.`Notes`,
`APPLICATION`.`DateSubmitted`, `APPLICATION`.`Status`, `Applicant`.`Username` AS
`Applicant.Username`, `Applicant`.`Education` AS `Applicant.Education`,
`RelevantURL`.`ApplicationID` AS `RelevantURL.ApplicationID`,
`RelevantURL`.`RelevantURL` AS `RelevantURL.RelevantURL`, `Category`.`ApplicationID`
AS `Category.ApplicationID`, `Category`.`Category` AS `Category.Category`,
`Jobs`.`PositionID` AS `Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`,
`Jobs`.`PositionName` AS `Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`,
`Jobs`.`JobPostFile` AS `Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`,
`Jobs`.`ApplicationDeadline` AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->CORRESPONDS_TO`.`ApplicationID` AS `Jobs.CORRESPONDS_TO.ApplicationID`,
`Jobs->CORRESPONDS_TO`.`PositionID` AS `Jobs.CORRESPONDS_TO.PositionID`,
`Jobs->CORRESPONDS_TO`.`JobPostURL` AS `Jobs.CORRESPONDS_TO.JobPostURL`,
`Documents`.`DocumentID` AS `Documents.DocumentID`, `Documents`.`ApplicantUsername`
AS `Documents.ApplicantUsername`, `Documents`.`DocFileName` AS
`Documents.DocFileName`, `Documents`.`DocType` AS `Documents.DocType`,
`Documents`.`Description` AS `Documents.Description`,

```

`Documents->SUBMIT_WITH`.`ApplicationID` AS
`Documents.SUBMIT_WITH.ApplicationID`, `Documents->SUBMIT_WITH`.`DocumentID` AS `Documents.SUBMIT_WITH.DocumentID`, `Interviews`.`InterviewID` AS
`Interviews.InterviewID`, `Interviews`.`ApplicantUsername` AS
`Interviews.ApplicantUsername`, `Interviews`.`Stage` AS `Interviews.Stage`, `Interviews`.`Date` AS `Interviews.Date`, `Interviews`.`ApplicationID` AS `Interviews.ApplicationID` FROM
`APPLICATION` AS `APPLICATION` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON `APPLICATION`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPL_RELEVANT_URL` AS `RelevantURL` ON `APPLICATION`.`ApplicationID` =
`RelevantURL`.`ApplicationID` LEFT OUTER JOIN `APPL_CATEGORY` AS `Category` ON
`APPLICATION`.`ApplicationID` = `Category`.`ApplicationID` LEFT OUTER JOIN (
`CORRESPONDS_TO` AS `Jobs->CORRESPONDS_TO` INNER JOIN `JOB` AS `Jobs` ON
`Jobs`.`PositionID` = `Jobs->CORRESPONDS_TO`.`PositionID`) ON
`APPLICATION`.`ApplicationID` = `Jobs->CORRESPONDS_TO`.`ApplicationID` LEFT
OUTER JOIN ( `SUBMIT_WITH` AS `Documents->SUBMIT_WITH` INNER JOIN
`DOCUMENT` AS `Documents` ON `Documents`.`DocumentID` =
`Documents->SUBMIT_WITH`.`DocumentID`) ON `APPLICATION`.`ApplicationID` =
`Documents->SUBMIT_WITH`.`ApplicationID` LEFT OUTER JOIN `INTERVIEW` AS
`Interviews` ON `APPLICATION`.`ApplicationID` = `Interviews`.`ApplicationID`;

```

- Add Application

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `APPLICATION`(`ApplicationID`, `ApplicantUsername`, `AName`, `Notes`, `DateSubmitted`, `Status`) VALUES (NULL,\$1,\$2,\$3,\$4,\$5);

- Update Application

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `ApplicationID`, `ApplicantUsername`, `AName`, `Notes`, `DateSubmitted`, `Status` FROM `APPLICATION` AS `APPLICATION` WHERE `APPLICATION`.`ApplicationID` = 2;

Executing (default): UPDATE `APPLICATION` SET `Notes`=\$1 WHERE `ApplicationID` = \$2

- Delete Application

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `APPLICATION` WHERE `ApplicationID` = 2 AND `ApplicantUsername` = 'fredprokop1@gmail.com'

- Add Application Category

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `APPL_CATEGORY` (`ApplicationID`, `Category`) VALUES (\$1,\$2);

- Delete Application Category

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `APPL_CATEGORY` WHERE `Category` = 'Finance' AND `ApplicationID` = 3

- Add Application Relevant URL

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `APPL_RELEVANT_URL` (`ApplicationID`, `RelevantURL`) VALUES (\$1,\$2);

- Delete Application Relevant URL

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `APPL_RELEVANT_URL` WHERE `RelevantURL` = 'www.url1.com' AND `ApplicationID` = 3

- Add Document to Application (Submit With)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `SUBMIT_WITH` (`ApplicationID`, `DocumentID`) VALUES (\$1,\$2);

- Delete Document from Application (Submit With)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `SUBMIT_WITH` WHERE `DocumentID` = 2 AND `ApplicationID` = 3

- Add Job to Application (Corresponds To)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `CORRESPONDS_TO`
(`ApplicationID`, `PositionID`, `JobPostURL`) VALUES (\$1,\$2,\$3);

- Delete Job from Application (Corresponds To)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `CORRESPONDS_TO` WHERE `PositionID` = 3 AND `ApplicationID` = 3

- Search by Name, Status, Category

Executing (default): SELECT `APPLICATION`.`ApplicationID`,
 `APPLICATION`.`ApplicantUsername`, `APPLICATION`.`AName`, `APPLICATION`.`Notes`,
 `APPLICATION`.`DateSubmitted`, `APPLICATION`.`Status`, `Applicant`.`Username` AS
 `Applicant.Username`, `Applicant`.`Education` AS `Applicant.Education`,
 `RelevantURL`.`ApplicationID` AS `RelevantURL.ApplicationID`,
 `RelevantURL`.`RelevantURL` AS `RelevantURL.RelevantURL`, `Category`.`ApplicationID`
 AS `Category.ApplicationID`, `Category`.`Category` AS `Category.Category`,
 `Jobs`.`PositionID` AS `Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`,
 `Jobs`.`PositionName` AS `Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`,
 `Jobs`.`JobPostFile` AS `Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`,
 `Jobs`.`ApplicationDeadline` AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
 `Jobs->CORRESPONDS_TO`.`ApplicationID` AS `Jobs.CORRESPONDS_TO.ApplicationID`,
 `Jobs->CORRESPONDS_TO`.`PositionID` AS `Jobs.CORRESPONDS_TO.PositionID`,
 `Jobs->CORRESPONDS_TO`.`JobPostURL` AS `Jobs.CORRESPONDS_TO.JobPostURL`,
 `Documents`.`DocumentID` AS `Documents.DocumentID`, `Documents`.`ApplicantUsername`
 AS `Documents.ApplicantUsername`, `Documents`.`DocFileName` AS
 `Documents.DocFileName`, `Documents`.`DocType` AS `Documents.DocType`,
 `Documents`.`Description` AS `Documents.Description`,
 `Documents->SUBMIT_WITH`.`ApplicationID` AS
 `Documents.SUBMIT_WITH.ApplicationID`, `Documents->SUBMIT_WITH`.`DocumentID`

```

AS `Documents.SUBMIT_WITH.DocumentID`, `Interviews`.`InterviewID` AS
`Interviews.InterviewID`, `Interviews`.`ApplicantUsername` AS
`Interviews.ApplicantUsername`, `Interviews`.`Stage` AS `Interviews.Stage`, `Interviews`.`Date`
AS `Interviews.Date`, `Interviews`.`ApplicationID` AS `Interviews.ApplicationID` FROM
`APPLICATION` AS `APPLICATION` LEFT OUTER JOIN `APPLICANT` AS `Applicant`
ON `APPLICATION`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPL_RELEVANT_URL` AS `RelevantURL` ON `APPLICATION`.`ApplicationID` =
`RelevantURL`.`ApplicationID` LEFT OUTER JOIN `APPL_CATEGORY` AS `Category` ON
`APPLICATION`.`ApplicationID` = `Category`.`ApplicationID` LEFT OUTER JOIN (
`CORRESPONDS_TO` AS `Jobs->CORRESPONDS_TO` INNER JOIN `JOB` AS `Jobs` ON
`Jobs`.`PositionID` = `Jobs->CORRESPONDS_TO`.`PositionID`) ON
`APPLICATION`.`ApplicationID` = `Jobs->CORRESPONDS_TO`.`ApplicationID` LEFT
OUTER JOIN ( `SUBMIT_WITH` AS `Documents->SUBMIT_WITH` INNER JOIN
`DOCUMENT` AS `Documents` ON `Documents`.`DocumentID` =
`Documents->SUBMIT_WITH`.`DocumentID`) ON `APPLICATION`.`ApplicationID` =
`Documents->SUBMIT_WITH`.`ApplicationID` LEFT OUTER JOIN `INTERVIEW` AS
`Interviews` ON `APPLICATION`.`ApplicationID` = `Interviews`.`ApplicationID` WHERE
`APPLICATION`.`AName` LIKE '%OneVest%' AND `APPLICATION`.`Status` LIKE
'%Submit%' AND `Category`.`Category` LIKE '%Fintech%';

```

Contacts

- Loading Contacts page

Executing (default): SELECT `CONTACT`.`ContactID`, `CONTACT`.`LinkedInURL`,
`CONTACT`.`Fname`, `CONTACT`.`Lname`, `Emails`.`ContactID` AS `Emails.ContactID`,
`Emails`.`Email` AS `Emails.Email`, `Phones`.`ContactID` AS `Phones.ContactID`,
`Phones`.`Phone` AS `Phones.Phone`, `Companies`.`CompanyName` AS
`Companies.CompanyName`, `Companies`.`Industry` AS `Companies.Industry`,
`Companies`.`HomePageURL` AS `Companies.HomePageURL`, `Companies`.`Description` AS
`Companies.Description`, `Companies->WORKS_AT`.`Role` AS
`Companies.WORKS_AT.Role`, `Companies->WORKS_AT`.`ContactID` AS
`Companies.WORKS_AT.ContactID`, `Companies->WORKS_AT`.`CompanyName` AS

```

`Companies.WORKS_AT.CompanyName`, `Interviews`.`InterviewID` AS
`Interviews.InterviewID`, `Interviews`.`ApplicantUsername` AS
`Interviews.ApplicantUsername`, `Interviews`.`Stage` AS `Interviews.Stage`, `Interviews`.`Date` AS
`Interviews.Date`, `Interviews`.`ApplicationID` AS `Interviews.ApplicationID`,
`Interviews->ATTENDS`.`ContactID` AS `Interviews.ATTENDS.ContactID`,
`Interviews->ATTENDS`.`InterviewID` AS `Interviews.ATTENDS.InterviewID`,
`Applicants`.`Username` AS `Applicants.Username`, `Applicants`.`Education` AS
`Applicants.Education`, `Applicants->KNOWS`.`Relationship` AS
`Applicants.KNOWS.Relationship`, `Applicants->KNOWS`.`Notes` AS
`Applicants.KNOWS.Notes`, `Applicants->KNOWS`.`LastContactDate` AS
`Applicants.KNOWS.LastContactDate`, `Applicants->KNOWS`.`Username` AS
`Applicants.KNOWS.Username`, `Applicants->KNOWS`.`ContactID` AS
`Applicants.KNOWS.ContactID`, `Referrals`.`ReferralID` AS `Referrals.ReferralID`,
`Referrals`.`ContactID` AS `Referrals.ContactID`, `Referrals`.`Date` AS `Referrals.Date`,
`Referrals`.`Notes` AS `Referrals.Notes`, `Referrals`.`ApplicantUsername` AS
`Referrals.ApplicantUsername`, `Referrals`.`PositionID` AS `Referrals.PositionID` FROM
`CONTACT` AS `CONTACT` LEFT OUTER JOIN `CONTACT_EMAIL` AS `Emails` ON
`CONTACT`.`ContactID` = `Emails`.`ContactID` LEFT OUTER JOIN `CONTACT_PHONE` AS `Phones` ON `CONTACT`.`ContactID` = `Phones`.`ContactID` LEFT OUTER JOIN (
`WORKS_AT` AS `Companies->WORKS_AT` INNER JOIN `COMPANY` AS `Companies` ON
`Companies`.`CompanyName` = `Companies->WORKS_AT`.`CompanyName`) ON
`CONTACT`.`ContactID` = `Companies->WORKS_AT`.`ContactID` LEFT OUTER JOIN (
`ATTENDS` AS `Interviews->ATTENDS` INNER JOIN `INTERVIEW` AS `Interviews` ON
`Interviews`.`InterviewID` = `Interviews->ATTENDS`.`InterviewID`) ON
`CONTACT`.`ContactID` = `Interviews->ATTENDS`.`ContactID` LEFT OUTER JOIN (
`KNOWS` AS `Applicants->KNOWS` INNER JOIN `APPLICANT` AS `Applicants` ON
`Applicants`.`Username` = `Applicants->KNOWS`.`Username`) ON `CONTACT`.`ContactID` =
`Applicants->KNOWS`.`ContactID` LEFT OUTER JOIN `REFERRAL` AS `Referrals` ON
`CONTACT`.`ContactID` = `Referrals`.`ContactID`;

```

- Add Contact

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `CONTACT`

(`ContactID`, `LinkedInURL`, `Fname`, `Lname`) VALUES (NULL, \$1, \$2, \$3);

- Delete Contact

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'admin@admin.com';

Executing (default): DELETE FROM `CONTACT` WHERE `ContactID` = 1

- Update Contact

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `ContactID`, `LinkedInURL`, `Fname`, `Lname` FROM `CONTACT` AS `CONTACT` WHERE `CONTACT`.`ContactID` = 2;

Executing (default): UPDATE `CONTACT` SET `Fname` = \$1, `Lname` = \$2, `LinkedInURL` = \$3 WHERE `ContactID` = \$4

- Add Phone Number

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `CONTACT_PHONE` (`ContactID`, `Phone`) VALUES (\$1, \$2);

- Delete Phone Number

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `CONTACT_PHONE` WHERE `Phone` = '403-000-0001' AND `ContactID` = 2

- Add Email

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `CONTACT_EMAIL` (`ContactID`, `Email`) VALUES (\$1,\$2);

- Delete Email

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `CONTACT_EMAIL` WHERE `Email` = 'bob_johnson@gmail.com' AND `ContactID` = 2

- Add Company to Contact (Works At)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `WORKS_AT` (`Role`, `ContactID`, `CompanyName`) VALUES (\$1,\$2,\$3);

- Delete Company from Contact (Works At)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `WORKS_AT` WHERE `CompanyName` = 'RBC' AND `ContactID` = 2

- Add Knows Relationship between Contact and User

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `KNOWS` (`Username`, `ContactID`) VALUES (\$1,\$2);

- Delete Knows Relationship between Contact and User

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `KNOWS` WHERE `ContactID` = 2 AND `Username` = 'fredprokop1@gmail.com'

- Update Knows Relationship between Contact and User

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `Relationship`, `Notes`, `LastContactDate`, `Username`, `ContactID` FROM `KNOWS` AS `KNOWS` WHERE `KNOWS`.`ContactID` = 2 AND `KNOWS`.`Username` = 'fredprokop1@gmail.com';

Executing (default): UPDATE `KNOWS` SET

`Relationship`=\$1, `Notes`=\$2, `LastContactDate`=\$3 WHERE `Username` = \$4 AND `ContactID` = \$5

- Add Referral

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): `INSERT INTO 'REFERRAL' ('ReferralID', 'ContactID', 'Date', 'Notes', 'ApplicantUsername', 'PositionID') VALUES (NULL,$1,$2,$3,$4,$5);`

- Update Referral

Executing (default): `SELECT 'Username', 'PasswordConfirm', 'Fname', 'Lname', 'IsActive', 'AdminFlag', 'PermissionLevel' FROM 'USER' AS 'USER' WHERE 'USER'. 'Username' = 'fredprokop1@gmail.com';`

Executing (default): `SELECT 'ReferralID', 'ContactID', 'Date', 'Notes', 'ApplicantUsername', 'PositionID' FROM 'REFERRAL' AS 'REFERRAL' WHERE 'REFERRAL'. 'ReferralID' = 1;`

Executing (default): `UPDATE 'REFERRAL' SET 'Notes'=$1, 'Date'=$2 WHERE 'ReferralID' = $3`

- Delete Referral

Executing (default): `SELECT 'Username', 'PasswordConfirm', 'Fname', 'Lname', 'IsActive', 'AdminFlag', 'PermissionLevel' FROM 'USER' AS 'USER' WHERE 'USER'. 'Username' = 'fredprokop1@gmail.com';`

Executing (default): `DELETE FROM 'REFERRAL' WHERE 'ReferralID' = 1`

- Activate ‘Only Show Contacts I Know’ Filter

Executing (default): `SELECT 'Username', 'PasswordConfirm', 'Fname', 'Lname', 'IsActive', 'AdminFlag', 'PermissionLevel' FROM 'USER' AS 'USER' WHERE 'USER'. 'Username' = 'fredprokop1@gmail.com';`

Executing (default): `SELECT 'Relationship', 'Notes', 'LastContactDate', 'Username', 'ContactID' FROM 'KNOWS' AS 'KNOWS';`

- Deactivate ‘Only Show Contacts I Know’ Filter

Same query as initial loading of Contacts page.

- Search by Earliest and Latest Contact Date

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `Relationship`, `Notes`, `LastContactDate`, `Username`, `ContactID` FROM `KNOWS` AS `KNOWS` WHERE `KNOWS`.`LastContactDate` BETWEEN '2023-06-01' AND '2023-11-01';

- Search by Earliest Contact Date

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `Relationship`, `Notes`, `LastContactDate`, `Username`, `ContactID` FROM `KNOWS` AS `KNOWS` WHERE `KNOWS`.`LastContactDate` >= '2023-05-01';

- Search by Latest Contact Date

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `Relationship`, `Notes`, `LastContactDate`, `Username`, `ContactID` FROM `KNOWS` AS `KNOWS` WHERE `KNOWS`.`LastContactDate` <= '2023-09-01';

- Search by First and Last Name; note that search can also be done using just First or Last Name

Executing (default): SELECT `CONTACT`.`ContactID`, `CONTACT`.`LinkedInURL`, `CONTACT`.`Fname`, `CONTACT`.`Lname`, `Emails`.`ContactID` AS `Emails.ContactID`, `Emails`.`Email` AS `Emails.Email`, `Phones`.`ContactID` AS `Phones.ContactID`, `Phones`.`Phone` AS `Phones.Phone`, `Companies`.`CompanyName` AS `Companies.CompanyName`, `Companies`.`Industry` AS `Companies.Industry`, `Companies`.`HomePageURL` AS `Companies.HomePageURL`, `Companies`.`Description` AS

```

`Companies.Description`, `Companies->WORKS_AT`.`Role` AS
`Companies.WORKS_AT.Role`, `Companies->WORKS_AT`.`ContactID` AS
`Companies.WORKS_AT.ContactID`, `Companies->WORKS_AT`.`CompanyName` AS
`Companies.WORKS_AT.CompanyName`, `Interviews`.`InterviewID` AS
`Interviews.InterviewID`, `Interviews`.`ApplicantUsername` AS
`Interviews.ApplicantUsername`, `Interviews`.`Stage` AS `Interviews.Stage`, `Interviews`.`Date` AS `Interviews.Date`, `Interviews`.`ApplicationID` AS `Interviews.ApplicationID`,
`Interviews->ATTENDS`.`ContactID` AS `Interviews.ATTENDS.ContactID`,
`Interviews->ATTENDS`.`InterviewID` AS `Interviews.ATTENDS.InterviewID`,
`Applicants`.`Username` AS `Applicants.Username`, `Applicants`.`Education` AS
`Applicants.Education`, `Applicants->KNOWS`.`Relationship` AS
`Applicants.KNOWS.Relationship`, `Applicants->KNOWS`.`Notes` AS
`Applicants.KNOWS.Notes`, `Applicants->KNOWS`.`LastContactDate` AS
`Applicants.KNOWS.LastContactDate`, `Applicants->KNOWS`.`Username` AS
`Applicants.KNOWS.Username`, `Applicants->KNOWS`.`ContactID` AS
`Applicants.KNOWS.ContactID`, `Referrals`.`ReferralID` AS `Referrals.ReferralID`,
`Referrals`.`ContactID` AS `Referrals.ContactID`, `Referrals`.`Date` AS `Referrals.Date`,
`Referrals`.`Notes` AS `Referrals.Notes`, `Referrals`.`ApplicantUsername` AS
`Referrals.ApplicantUsername`, `Referrals`.`PositionID` AS `Referrals.PositionID` FROM
`CONTACT` AS `CONTACT` LEFT OUTER JOIN `CONTACT_EMAIL` AS `Emails` ON
`CONTACT`.`ContactID` = `Emails`.`ContactID` LEFT OUTER JOIN `CONTACT_PHONE` AS `Phones` ON `CONTACT`.`ContactID` = `Phones`.`ContactID` LEFT OUTER JOIN (
`WORKS_AT` AS `Companies->WORKS_AT` INNER JOIN `COMPANY` AS `Companies` ON `Companies`.`CompanyName` = `Companies->WORKS_AT`.`CompanyName`) ON
`CONTACT`.`ContactID` = `Companies->WORKS_AT`.`ContactID` LEFT OUTER JOIN (
`ATTENDS` AS `Interviews->ATTENDS` INNER JOIN `INTERVIEW` AS `Interviews` ON
`Interviews`.`InterviewID` = `Interviews->ATTENDS`.`InterviewID`) ON
`CONTACT`.`ContactID` = `Interviews->ATTENDS`.`ContactID` LEFT OUTER JOIN (
`KNOWS` AS `Applicants->KNOWS` INNER JOIN `APPLICANT` AS `Applicants` ON
`Applicants`.`Username` = `Applicants->KNOWS`.`Username`) ON `CONTACT`.`ContactID` = `Applicants->KNOWS`.`ContactID` LEFT OUTER JOIN `REFERRAL` AS `Referrals` ON

```

```
'CONTACT`.`ContactID` = `Referrals`.`ContactID` WHERE `CONTACT`.`Fname` LIKE
'%Julia%' AND `CONTACT`.`Lname` LIKE '%J%';

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): SELECT `Relationship`, `Notes`, `LastContactDate`, `Username`,
`ContactID` FROM `KNOWS` AS `KNOWS`;
```

Interviews

- Loading Interviews page

```
SELECT `INTERVIEW`.`InterviewID`, `INTERVIEW`.`ApplicantUsername`,
`INTERVIEW`.`Stage`, `INTERVIEW`.`Date`, `INTERVIEW`.`ApplicationID`,
`Jobs`.`PositionID` AS `Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`,
`Jobs`.`PositionName` AS `Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`,
`Jobs`.`JobPostFile` AS `Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`,
`Jobs`.`ApplicationDeadline` AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN ( `MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
```

```

`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN ( `ATTENDS` AS `Contacts->ATTENDS` 
INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID` ) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
'fredprokop1@gmail.com';

```

- Add Interview

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT max(`InterviewID`) AS `max` FROM `INTERVIEW` AS `INTERVIEW` WHERE `INTERVIEW`.`ApplicantUsername` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `INTERVIEW`

(`InterviewID`, `ApplicantUsername`, `Stage`, `Date`, `ApplicationID`) VALUES
(\$1,\$2,\$3,\$4,\$5);

- Update Interview

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `InterviewID`, `ApplicantUsername`, `Stage`, `Date`, `ApplicationID` FROM `INTERVIEW` AS `INTERVIEW` WHERE `INTERVIEW`.`InterviewID` = 1;

Executing (default): UPDATE `INTERVIEW` SET `Stage`=\$1, `Date`=\$2 WHERE `InterviewID` = \$3

- Delete Interview

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `INTERVIEW` WHERE `InterviewID` = 1

- Add Contact Attending

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `ATTENDS` (`ContactID`, `InterviewID`) VALUES (\$1,\$2);

- Delete Contact Attending

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): DELETE FROM `ATTENDS` WHERE `ContactID` = 2 AND `InterviewID` = 1

- Add Job Mentioned

Executing (default): INSERT INTO `MENTIONS` (`PositionID`, `InterviewID`) VALUES (\$1,\$2);

- Delete Job Mentioned

Executing (default): DELETE FROM `MENTIONS` WHERE `PositionID` = 3 AND `InterviewID` = 1

- Search by Stage

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`,
`INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`,
`INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`,
`Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline`
AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN (`MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN (`ATTENDS` AS `Contacts->ATTENDS`
INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
`fredprokop1@gmail.com` AND `INTERVIEW`.`Stage` LIKE '%Stage 1%' ORDER BY
`INTERVIEW`.`Stage` ASC;

- Search by From and To Dates

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`,
`INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`,
`INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`,
`Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline`
AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN (`MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN (`ATTENDS` AS `Contacts->ATTENDS`

```

INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
'fredprokop1@gmail.com' AND `INTERVIEW`.`Date` BETWEEN '2023-11-20' AND
'2023-12-01' ORDER BY `INTERVIEW`.`Stage` ASC;

```

- Search by From Date only

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`,
`INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`,
`INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`,
`Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline`
AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN (`MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =

```

`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN ( `ATTENDS` AS `Contacts->ATTENDS` 
INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
'fredprokop1@gmail.com' AND `INTERVIEW`.`Date` >= '2023-11-20' ORDER BY
`INTERVIEW`.`Stage` ASC;

```

- Search by To Date only

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`,
`INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`,
`INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`,
`Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline`
AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS

```

`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN ( `MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN ( `ATTENDS` AS `Contacts->ATTENDS` INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
'fredprokop1@gmail.com' AND `INTERVIEW`.`Date` <= '2023-11-29' ORDER BY
`INTERVIEW`.`Stage` ASC;

```

- Sort by Stage, Ascending

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`, `INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`, `INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`, `Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS `Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS `Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline` AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`, `Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`, `Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`, `Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS `Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,

```

`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN ( `MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN ( `ATTENDS` AS `Contacts->ATTENDS` 
INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
'fredprokop1@gmail.com' ORDER BY `INTERVIEW`.`Stage` ASC;

```

- Sort by Stage, Descending

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`,
`INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`,
`INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`,
`Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline`
AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,

```

`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN ( `MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN ( `ATTENDS` AS `Contacts->ATTENDS` 
INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
'fredprokop1@gmail.com' ORDER BY `INTERVIEW`.`Stage` DESC;

```

- Sort by Earliest Date

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`,
`INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`,
`INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`,

```

`Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline` AS
`Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN ( `MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN ( `ATTENDS` AS `Contacts->ATTENDS` INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =
`Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =
'fredprokop1@gmail.com' ORDER BY `INTERVIEW`.`Date` ASC;

```

- Sort by Latest Date

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): SELECT `INTERVIEW`.`InterviewID`,
`INTERVIEW`.`ApplicantUsername`, `INTERVIEW`.`Stage`, `INTERVIEW`.`Date`,
`INTERVIEW`.`ApplicationID`, `Jobs`.`PositionID` AS `Jobs.PositionID`,
`Jobs`.`CompanyName` AS `Jobs.CompanyName`, `Jobs`.`PositionName` AS
`Jobs.PositionName`, `Jobs`.`Description` AS `Jobs.Description`, `Jobs`.`JobPostFile` AS
`Jobs.JobPostFile`, `Jobs`.`PositionType` AS `Jobs.PositionType`, `Jobs`.`ApplicationDeadline`
AS `Jobs.ApplicationDeadline`, `Jobs`.`Salary` AS `Jobs.Salary`,
`Jobs->MENTIONS`.`PositionID` AS `Jobs.MENTIONS.PositionID`,
`Jobs->MENTIONS`.`InterviewID` AS `Jobs.MENTIONS.InterviewID`,
`Applicant`.`Username` AS `Applicant.Username`, `Applicant`.`Education` AS
`Applicant.Education`, `Application`.`ApplicationID` AS `Application.ApplicationID`,
`Application`.`ApplicantUsername` AS `Application.ApplicantUsername`,
`Application`.`AName` AS `Application.AName`, `Application`.`Notes` AS
`Application.Notes`, `Application`.`DateSubmitted` AS `Application.DateSubmitted`,
`Application`.`Status` AS `Application.Status`, `Contacts`.`ContactID` AS `Contacts.ContactID`,
`Contacts`.`LinkedInURL` AS `Contacts.LinkedInURL`, `Contacts`.`Fname` AS
`Contacts.Fname`, `Contacts`.`Lname` AS `Contacts.Lname`,
`Contacts->ATTENDS`.`ContactID` AS `Contacts.ATTENDS.ContactID`,
`Contacts->ATTENDS`.`InterviewID` AS `Contacts.ATTENDS.InterviewID` FROM
`INTERVIEW` AS `INTERVIEW` LEFT OUTER JOIN (`MENTIONS` AS
`Jobs->MENTIONS` INNER JOIN `JOB` AS `Jobs` ON `Jobs`.`PositionID` =
`Jobs->MENTIONS`.`PositionID`) ON `INTERVIEW`.`InterviewID` =
`Jobs->MENTIONS`.`InterviewID` LEFT OUTER JOIN `APPLICANT` AS `Applicant` ON
`INTERVIEW`.`ApplicantUsername` = `Applicant`.`Username` LEFT OUTER JOIN
`APPLICATION` AS `Application` ON `INTERVIEW`.`ApplicationID` =
`Application`.`ApplicationID` LEFT OUTER JOIN (`ATTENDS` AS `Contacts->ATTENDS`
INNER JOIN `CONTACT` AS `Contacts` ON `Contacts`.`ContactID` =
`Contacts->ATTENDS`.`ContactID`) ON `INTERVIEW`.`InterviewID` =

```
'Contacts->ATTENDS`.`InterviewID` WHERE `INTERVIEW`.`ApplicantUsername` =  
'fredprokop1@gmail.com' ORDER BY `INTERVIEW`.`Date` DESC;
```

Offers

- Loading Offers page

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
 `OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
 `OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
 `Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS
 `Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
 `Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
 `Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS
 `APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
 `OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
 `Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
 `OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` ORDER BY
 `OFFER`.`ResponseDeadline` ASC;

- Create Offer

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
 `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
 'fredprokop1@gmail.com';

Executing (default): INSERT INTO `OFFER`
(`OfferID`, `ApplicantUsername`, `OfferFileName`, `ResponseDeadline`, `Compensation`, `StartDa-
te`, `Notes`, `PositionID`) VALUES (NULL,\$1,\$2,\$3,\$4,\$5,\$6,\$7);

- Update Offer

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
 `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
 'fredprokop1@gmail.com';

Executing (default): SELECT `OfferID`, `ApplicantUsername`, `OfferFileName`,
`ResponseDeadline`, `Compensation`, `StartDate`, `Notes`, `PositionID` FROM `OFFER` AS
`OFFER` WHERE `OFFER`.`OfferID` = '2';

Executing (default): UPDATE `OFFER` SET
`Compensation` = \$1, `Notes` = \$2, `PositionID` = \$3, `StartDate` = \$4 WHERE `OfferID` = \$5

- Delete Offer

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): DELETE FROM `OFFER` WHERE `OfferID` = 2 AND
`ApplicantUsername` = 'fredprokop1@gmail.com'

- Search by Lowest and Highest Compensation, Earliest and Latest Deadline, and Earliest
and Latest Start Date; note that any combination of these search fields can be used, but
the SQL query shown below demonstrates the most complex query that includes all fields

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
`OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
`OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
`Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS
`Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
`Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
`Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS
`APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
`OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
`Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
`OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` WHERE
`OFFER`.`Compensation` BETWEEN '50000' AND '60000' AND `OFFER`.`ResponseDeadline`

BETWEEN '2023-12-18' AND '2024-01-31' AND `OFFER`.`StartDate` BETWEEN
'2024-04-22' AND '2024-05-06' ORDER BY `OFFER`.`Compensation` ASC;

- Sort by Compensation Ascending

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
`OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
`OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
`Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS
`Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
`Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
`Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS
`APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
`OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
`Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
`OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` ORDER BY
`OFFER`.`Compensation` ASC;

- Sort by Compensation Descending

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
`OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
`OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
`Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS
`Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
`Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
`Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS

```

`APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
`OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
`Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
`OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` ORDER BY
`OFFER`.`Compensation` DESC;

```

- Sort by Earliest Response Deadline (Ascending)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
`fredprokop1@gmail.com`;

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
`OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
`OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
`Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS
`Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
`Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
`Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS
`APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
`OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
`Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
`OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` ORDER BY
`OFFER`.`ResponseDeadline` ASC;

- Sort by Latest Response Deadline (Descending)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
`fredprokop1@gmail.com`;

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
`OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
`OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
`Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS

```

`Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
`Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
`Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS
`APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
`OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
`Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
`OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` ORDER BY
`OFFER`.`ResponseDeadline` DESC;

```

- Sort by Earliest Start Date (Ascending)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
`fredprokop1@gmail.com`;

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
`OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
`OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
`Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS
`Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
`Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
`Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS
`APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
`OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
`Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
`OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` ORDER BY `OFFER`.`StartDate`
ASC;

- Sort by Latest Start Date (Descending)

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
`fredprokop1@gmail.com`;

Executing (default): SELECT `OFFER`.`OfferID`, `OFFER`.`ApplicantUsername`,
`OFFER`.`OfferFileName`, `OFFER`.`ResponseDeadline`, `OFFER`.`Compensation`,
`OFFER`.`StartDate`, `OFFER`.`Notes`, `OFFER`.`PositionID`, `Job`.`PositionID` AS
`Job.PositionID`, `Job`.`CompanyName` AS `Job.CompanyName`, `Job`.`PositionName` AS
`Job.PositionName`, `Job`.`Description` AS `Job.Description`, `Job`.`JobPostFile` AS
`Job.JobPostFile`, `Job`.`PositionType` AS `Job.PositionType`, `Job`.`ApplicationDeadline` AS
`Job.ApplicationDeadline`, `Job`.`Salary` AS `Job.Salary`, `APPLICANT`.`Username` AS
`APPLICANT.Username`, `APPLICANT`.`Education` AS `APPLICANT.Education` FROM
`OFFER` AS `OFFER` LEFT OUTER JOIN `JOB` AS `Job` ON `OFFER`.`PositionID` =
`Job`.`PositionID` LEFT OUTER JOIN `APPLICANT` AS `APPLICANT` ON
`OFFER`.`ApplicantUsername` = `APPLICANT`.`Username` ORDER BY `OFFER`.`StartDate`
DESC;

Users (Admin)

- Loading User page for admins

Executing (default): SELECT `USER`.`Username`, `USER`.`Password`,
`USER`.`PasswordConfirm`, `USER`.`Fname`, `USER`.`Lname`, `USER`.`IsActive`,
`USER`.`AdminFlag`, `USER`.`PermissionLevel`;

- Update names

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'admin@admin.com';

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =
'fredprokop1@gmail.com';

Executing (default): UPDATE `USER` SET `Fname`=\$1, `Lname`=\$2, WHERE `Username` =
\$3

- Activate/Deactivate account

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'admin@admin.com';

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): UPDATE `USER` SET `IsActive`=\$1 WHERE `Username` = \$2

- Update permission level

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'admin@admin.com';

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): UPDATE `USER` SET `PermissionLevel`=\$1 WHERE `Username` = \$2

- Reset password

Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`, `AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` = 'fredprokop1@gmail.com';

Executing (default): UPDATE `USER` SET `Password`=\$1 WHERE `Username` = \$2

- Search by first name

Executing (default): SELECT `USER`.`Username`, `USER`.`Password`,
 `USER`.`PasswordConfirm`, `USER`.`Fname`, `USER`.`Lname`, `USER`.`IsActive`,
 `USER`.`AdminFlag`, `USER`.`PermissionLevel`, WHERE `USER`.`Fname` LIKE "%Fred%"
 ORDER BY `USER`.`Fname` ASC;

- Search by last name

```
Executing (default): SELECT `USER`.`Username`, `USER`.`Password`,  
`USER`.`PasswordConfirm`, `USER`.`Fname`, `USER`.`Lname`, `USER`.`IsActive`,  
`USER`.`AdminFlag`, `USER`.`PermissionLevel`, WHERE `USER`.`Lname` LIKE '%Prokop%'  
ORDER BY `USER`.`Lname` ASC;
```

- Create new user

```
Executing (default): SELECT `Username`, `PasswordConfirm`, `Fname`, `Lname`, `IsActive`,  
`AdminFlag`, `PermissionLevel` FROM `USER` AS `USER` WHERE `USER`.`Username` =  
'fredprokop1@gmail.com';
```

```
Executing (default): INSERT INTO `USER`  
(`Username`, `Password`, `PasswordConfirmation`, `Fname`, `Lname`, `IsActive`, `AdminFlag`,  
`PermissionLevel`) VALUES ($1,$2,$3,$4,$5,$6,$7,$8);
```

User Manual

A user manual is provided below that showcases all features of the application, highlighting the functionality that users can expect. Note that the snapshots below include sample data that was used to populate the database and better showcase the application's features. The manual is separated by application pages for more intuitive navigation.

Note that the application handles foreign key constraint errors when users attempt to add entries that already exist. This is not highlighted in every section of the User Manual, but rather explicitly shown in the Interviews section of the User Manual. Please refer to the appropriate section to examine the way the application handles this error. Note that other fields provide a similar error when a duplicate entry is attempted by a user.

Login & Signup

Users first encounter the login page, where they are able to register for an account by pressing the button underneath 'LOG IN'.

≡ ApliTrack

LOGIN

LOG IN SIGN UP

 ApliTrack

Log in

Username

Password *

LOG IN

[Don't have an account? Sign up.](#)

Copyright © ApliTrack 2023.

Users are able to sign up by completing the fields below.

≡ ApliTrack

SIGNUP

LOG IN SIGN UP

 ApliTrack

Sign up

First Name *

Last Name *

Username

Password *

Confirm Password *

SIGN UP

[Already have an account? Sign in](#)

Copyright © ApliTrack 2023.

Example of completed signup page, below. If users attempt to create an account with a username that already exists, a quick pop-up message will appear warning them that this cannot be done.

The screenshot shows the ApliTrack sign-up interface. At the top, there's a blue header bar with the ApliTrack logo on the left, a 'SIGNUP' button in the center, and 'LOGIN' and 'SIGN UP' links on the right. Below the header is a purple circular icon with a lock symbol. The main title 'ApliTrack' is displayed in large, bold letters, with 'Sign up' underneath it. The form fields are arranged in two columns: 'First Name *' (containing 'Fedor') and 'Last Name *' (containing 'Prokopchuk'). Below these are 'Username' (containing 'fredprokop@gmail.com') and 'Password *' (containing '*****'). To the right of the password field is a 'Confirm Password *' field (also containing '*****'). At the bottom of the form is a large blue 'SIGN UP' button. Below the button, a small link reads 'Already have an account? Sign in'. At the very bottom of the page, the copyright notice 'Copyright © ApliTrack 2023.' is visible.

Users will also be warned if passwords do not match, as shown below.



ApliTrack

Sign up

First Name *	Last Name *
Fedor	Prokopchuk
Username	
fredprokopchuk999@gmail.com	
Password *	
.....	
Confirm Password *	
.....	

SIGN UP

[Already have an account? Sign in](#)

Copyright © [ApliTrack](#) 2023.

 VALIDATION ERROR. Please make sure your passwords match..

If users already have an account, they can enter their information:



ApliTrack

Log in

Username

fredprokop@gmail.com

Password *

.....

LOG IN

[Don't have an account? Sign up](#)

Copyright © ApliTrack 2023.

If the wrong password is provided, users will be made aware:

ApliTrack

LOGIN

SIGN UP

Log in

Username

fredprokop@gmail.com

Password *

LOG IN

Don't have an account? [Sign up](#)

Copyright © [ApliTrack](#) 2023.

✖ Incorrect username or password

Profile

Users first encounter a blank Profile page that can be customized to fit the specific needs of the applicant. Account information can be updated at the top, such as First and Last name, Education, and password.

The screenshot shows the ApilTrack Profile page. At the top, there is a navigation bar with the ApilTrack logo, a menu icon, and the word "PROFILE". On the right side of the top bar are the email address "fredprokop@gmail.com" and a "LOGOUT" link. Below the top bar, there is a section labeled "YOU" with a placeholder profile picture. This section contains input fields for "First Name" (Fedor), "Last Name" (Prokopchuk), and "Education", along with an "UPDATE" button. Below these fields are buttons for "Current Password", "New Password", and "Confirm New Password", followed by another "UPDATE" button. The main content area is divided into several sections: "EXPERIENCES", "PROJECTS", "CERTIFICATIONS", "SKILLS", and "COMPETITIONS". Each section has a text input field for adding new items and a blue "+" button to the right.

Applicants can update personal information or the password by pressing the corresponding Update button on the right. If the current password is incorrect or the new passwords do not match, the user will be notified with a quick pop-up message.

The screenshot shows the ApliTrack Profile page. At the top, there is a blue header bar with the ApliTrack logo, the word "PROFILE" in the center, and a user email "fredprokop@gmail.com" and "LOGOUT" on the right. Below the header is a white form area. On the left is a circular placeholder for a profile picture labeled "YOU". To the right of the placeholder are four input fields: "First Name" (Fedor), "Last Name" (Prokochuk(Updated)), "Education" (BComm, Finance Major), and three password fields ("Current Password", "New Password", "Confirm New Password"). Each password field has an "UPDATE" button to its right.

Applicants can then add Experiences, Projects, Certifications, Skills, and Competitions by populating the corresponding empty fields in the Profile page. Examples of each completed field are provided below.

The screenshot shows the "EXPERIENCES" section of the profile page. It lists three experiences: "ATB" (ATB internship description) with edit and delete icons; "Deloitte" (Deloitte internship description) with edit and delete icons; and "Tech Start" (Tech Start club description) with edit and delete icons. Below this is a form for adding a new experience, consisting of a title input field ("Enter Title of New Experience (Max 32 Characters)") and a description input field ("Enter description of new Experience") with a plus sign icon to its right.

The screenshot shows the "PROJECTS" section of the profile page. It lists one project: "CPSC 471 Project" (Database project description) with edit and delete icons. Below this is a form for adding a new project, consisting of a title input field ("Enter Title of New Project (Max 32 Characters)") and a description input field ("Enter description of new Project") with a plus sign icon to its right.

CERTIFICATIONS

Respect in the Workplace X

Financial Modelling X

Enter Name of New Certification (Max 32 Characters) +

SKILLS

Company valuation X

Due diligence X

Enter Name of New Skill (Max 32 Characters) +

COMPETITIONS

TRMC2023 Case Competition X

Enter Name of New Competition (Max 32 Characters) +

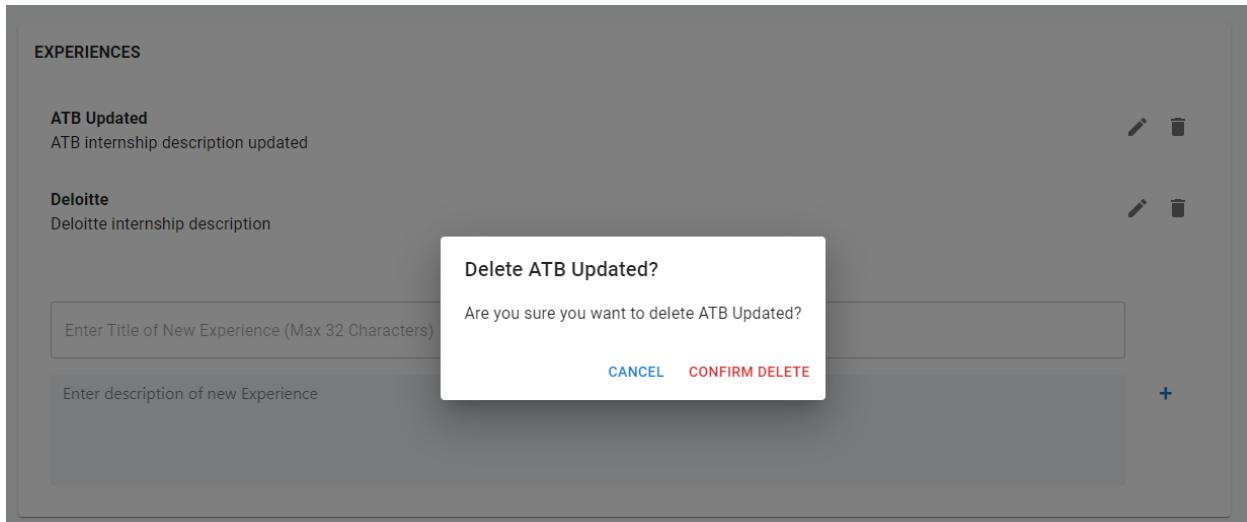
Applicants can then edit the Experiences and Projects entries as these have titles and descriptions. Entries can be edited by clicking the Pencil icon, after which a pop-up will appear, as shown below.

The screenshot shows a modal dialog titled "Edit Item" overlaid on a list of experiences. The dialog contains two input fields: "Title" with the value "ATB" and "Description" with the value "ATB internship description". At the bottom right of the dialog are "CANCEL" and "SAVE" buttons. The background is dimmed, and the experience list items are visible:

- ATB**
ATB internship description
- Deloitte**
Deloitte internship description
- Tech Start**
Tech Start club description

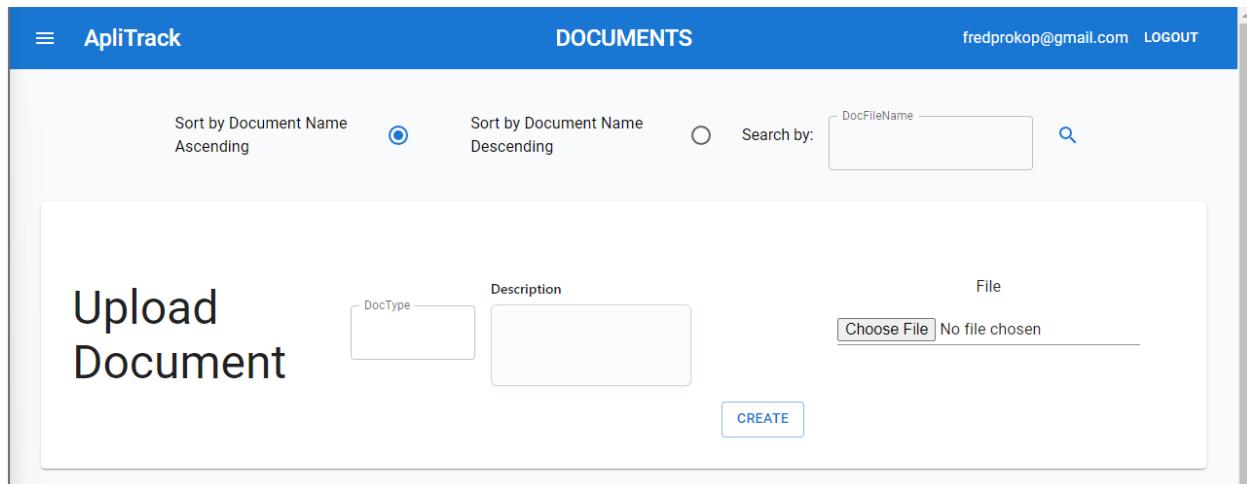
At the bottom left of the main interface, there are input fields: "Enter Title of New Experience (Max 32 Characters)" and "Enter description of new Experience". A blue "+" button is located at the bottom right.

Applicants can also delete any entries by clicking the Bin icon for Experiences and Projects, or clicking the X for Certifications, Skills, and Competitions. For Experiences and Projects, a confirmation box will pop up, while Certifications, Skills, and Competitions entries will be deleted right away, as they are simple entries.



Documents

Upon first loading the Documents page, applicants will encounter none of their Documents stored in the system. The page allows applicants to upload documents, provide a Document Type and Description, and upload the file itself.



Applicants are able to fill in the fields, as so:

☰ ApilTrack DOCUMENTS fredprokop@gmail.com LOGOUT

Sort by Document Name
Ascending

Sort by Document Name
Descending

Search by: DocFileName

Upload Document

DocType: Resume Description: Resume tailored for finance roles File: Choose File Finance Resume.pdf

CREATE

Upon adding a document, each will appear as below. Applicants are also able to update each document by updating the necessary fields and then pressing the Update button.

Document

Document File

DocType: Resume Description: Resume tailored for finance roles

Fedor Prokopchuk
Calgary, Alberta, Canada (403) 931-3997 | fedor.prokopchuk@calgary.ca

EDUCATION
Haskayne School of Business, University of Calgary
Bachelor of Commerce in Finance, combined with Minor in Computer Science
Cumulative GPA: 4.0 / 4.0; Davis's 1st Winter 2022, Winter 2021, Fall 2020
International Baccalaureate Diploma, High School
Final International Baccalaureate Points: 44 / 45; GPA: 4.5 / 4.5

Calgary, AB, Canada
Expected December 2024

Bulgaria, Serbia
Graduated May 2020

WORK EXPERIENCE
University of Calgary, Finance 311 - Introduction to Finance
Finance course covering the time value of money, the risk-return trade off, investing, and corporate financing.
Teaching Assistant
September 2022 - April 2023

- Prepared and taught weekly tutorials in preparation for quizzes and exams, showing students how to solve questions and explaining the theory and methodology behind solutions.
- Marked assignments and tests for exploring portfolio theory, the CAPM, and capital structure.

Deutsche Canada, Portfolio Manager
Calgary, AB, Canada
September 2022 - April 2023

- Identified, met, and analyzed new investment targets through research, company calls, and diligence.
- Assisted with financial evaluation through exit analyses, comparisons, and presented transactions research.
- Prepared initial memo and presentation for the Board of Directors, and the Board. Of the 10 investment memos prepared, 6 went to diligence and 3 were approved.
- Ensured the final investment memorandum was accurate and reflected the Board's decision.

Existing File: Finance Resume - [1].pdf

UPDATE

To delete a document, applicants can press the Bin icon, which will create a confirmation pop-up.

Document

The screenshot shows a document management interface. On the left, there's a sidebar with a 'DocType' dropdown set to 'Resume' and a 'Description' box containing the text 'Resume tailored for finance roles updated'. The main area displays a resume document for 'Fedor Prokopchuk' with sections for EDUCATION, WORK EXPERIENCE, and other details. A modal dialog box is overlaid on the page, asking 'Delete Finance Resume - [1].pdf?' with 'CANCEL' and 'CONFIRM DELETE' buttons. Below the modal, a message says 'Are you sure you want to delete Finance Resume - [1].pdf?'. At the bottom of the screen, there's a large 'UPDATE' button.

If multiple documents are added, they will be displayed one after another, as such.

ApliTrack

DOCUMENTS

Sort by Document Name
Ascending

Sort by Document Name
Descending

Search by:

DocFileName

Document

Description

DocType

Resume

Resume tailored for finance roles

Document File

Fedor Prokophuk

Calgary, Alberta, Canada | 101-301-097 | [View profile](#)

EDUCATION

Haskayne School of Business, University of Calgary

Bachelor of Commerce in Finance, combined with a minor in Computer Science

Cumulative GPA: 4.0 / 4.0; Dean's List Winter 2022, Winter 2021, Fall 2020

International School of Bratislava, High School

Final Grade Point Average: 4.43; GPA: 4.5 / 4.5

Calgary, AB, Canada

Bratislava, Slovakia

Graduated May 2020

WORK EXPERIENCE

University of Calgary, Finance 317 - Introduction to Finance

Calgary, AB, Canada

Finance course covering the nature of money, the real-value trade off, assessing, and corporate financing.

Teaching Assistant, September 2022 - Present

- Prepared and taught weekly tutorials in preparation for quizzes and exams, showing students how to solve questions and explaining the theory and methodology behind solutions.
- Maintained assignments that explore portfolio theory, the CAPM, and capital structure.

Order Book Capital, Portfolio Manager

Calgary, AB, Canada

Deliver capital investment capital fund, investing \$150M in Canadian technology stocks.

September 2022 - April 2023

Analyst Intern

Helped to identify and make meaningful equity investments in Series A and B companies across various industries.

- Identified, vetted, and analyzed new investment targets through research, company calls, and diligence.
- Assisted with financial evaluations through exit analyses, comparables, and precedent transactions research.
- Prepared initial term sheets, then presented them to the Investment Committee, the Executive Committee, and the Board. Of the 10 investment memos prepared, 6 went to diligence and 3 were approved.

ProTech VC Fund 1.2 investment (\$15MM invested). Prepared initial memo and thesis presentation to the ProTech board of directors, mentioned in seedfunds review, chosen the fund.

Choose File No file chosen

Existing File: Finance Resume - [2].pdf

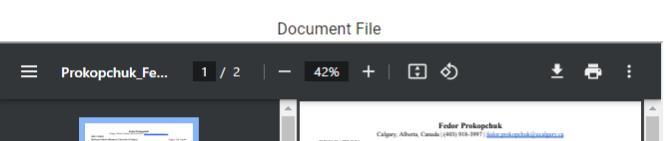
UPDATE

Document



DocType	Description
Resume	Resume tailored for fintech roles

Document File



Applicants can search for document file names. For example, an applicant may search for all file names containing the string “Cover Letter”, as shown below.

The screenshot shows the Applitrack Documents page. At the top, there are sorting options for "Sort by Document Name Ascending" and "Sort by Document Name Descending". To the right is a search bar with "Search by:" and "DocFileName" dropdown, and a magnifying glass icon. The main area displays a document titled "Cover Letter On...". The document content is as follows:

Fred Prokopchuk
Calgary, Alberta | Phone Number: (403) 918 3997 | fredprokopchuk@calgary.ca
January 30, 2022
RE: Investment Research Intern
Dear Hiring Manager,

With a quantitatively-minded mindset, an innate passion for problem-solving, and a desire to work in a fast-paced environment, I believe my skills and experience make me an ideal candidate for an Investment Research Intern. I have held highly analytical and leadership-oriented roles. I wish to apply my knowledge of investments and technology in a professional setting. I am a hard-working individual who is determined to continue to allow me to succeed as an intern. I am an international student at the University of Calgary and I located this job posting through the University Co-op Job Board.

My enthusiasm for problem-solving, analytics, and finance is evident through my demonstrated success in my previous roles as a financial analyst and investment manager. To develop my financial literacy, I analyzed financial securities and presented pitches in a wealth management student association, where I now act as a senior co-director, leading a team of students in research, analysis, and presentation of financial products and their implications on communications. I have also combined my degree in business with a degree in computer science to pursue my interest in technology, and I currently act as an executive member of an entrepreneurship club, which aims to promote and encourage development, where I help foster strategic startup creation.

Choose File | No file chosen

Existing File: Generic Cover Letter - [4...]

UPDATE

Applicants can also enhance the search by specifying the sorting order in which to display results. Upon selecting the sorting method, the Search icon on the right must be pressed to execute the search and show the results.

The screenshot shows the Applitrack Companies & Jobs page. At the top, there are sorting options for "Sort by Document Name Ascending" and "Sort by Document Name Descending". To the right is a search bar with "Search by:" and "DocFileName" dropdown, and a magnifying glass icon.

Companies & Jobs

Upon first loading the Companies page, applicants will typically encounter previously populated Company entries, as these are shared across all applicants. However, a blank Companies page is also provided for reference, below, where applicants can add new companies.

The screenshot shows the ApilTrack application interface for managing companies. At the top, there's a blue header bar with the title "ApilTrack" and the word "COMPANIES". On the right side of the header, it shows the user's email "fredprokop1@gmail.com" and a "LOGOUT" link. Below the header, there's a search section titled "Search by:" with fields for "CompanyName" (a text input), "EarliestApplicationDeadline" (a date input with a calendar icon), and "LatestApplicationDeadline" (another date input with a calendar icon). There are also two toggle buttons: "Only Show Jobs I Track" (which is turned off) and "Sort Ascending" (which is selected, indicated by a blue outline).

On the left side of the main content area, there's a button labeled "Add New Company". To its right, there are four input fields: "CompanyName" (containing "RBC"), "Industry" (containing "Banking"), "HomePageURL" (containing "https://www.rbcroyalbank.com"), and "Description" (containing "Royal Bank of Canada is"). To the far right of these fields is a blue "CREATE" button.

A new company can be added by filling in the fields, then pressing the Create button to the right.

This screenshot shows the same ApilTrack interface after a company has been added. The "Add New Company" button is no longer visible. The four input fields now contain sample data: "CompanyName" is "RBC", "Industry" is "Banking", "HomePageURL" is "https://www.rbcroyalbank.com", and "Description" is "Royal Bank of Canada is". The blue "CREATE" button is still present to the right of the fields.

The added company will appear as follows. Note that jobs can only be added to existing companies. If a company does not exist and an applicant wishes to add a job, they must add the company to the system first. Any fields for the Company can then be edited and updated with the Update button. Note that regular applicants cannot delete companies as these are shared across all users; only admins have the permission to delete companies, which is shown in the Admin section of the User Manual.

The screenshot shows the Applitrack application interface. At the top, there is a navigation bar with the title "Applitrack", the section "COMPANIES", and user information "fredprokop1@gmail.com LOGOUT".

Below the navigation bar, there is a search bar labeled "Search by:" with fields for "CompanyName", "EarliestApplicationDeadline" (MM/DD/YYYY), and "LatestApplicationDeadline" (MM/DD/YYYY). There is also a "Sort Ascending" radio button and a "Sort Descending" radio button.

The main content area displays a company record for "RBC". The company logo is shown, along with fields for "Industry" (Banking) and "HomePageURL" (<https://www.rbcroyalbank.com>). A detailed description of RBC is provided, mentioning it is a Canadian multinational financial services company and the largest bank in Canada by market capitalization. A "UPDATE" button is present.

On the left side, there is a "Jobs" section with a "Add New Job" button. The "Add New Job" form includes fields for "PositionName", "Description", "PositionType", "Salary", and a "Choose File" input field which currently shows "No file chosen". A "CREATE" button is located next to the file input field.

Once companies are populated in the database, applicants are able to add jobs that correspond to existing companies. To do so, applicants simply enter the job details in the Add New Job field, then press the Create button, as shown below for a sample Financial Analyst job for RBC. Note that applicants are also able to upload the job post file by pressing Choose File, as also shown below.

A file selection dialog box is open, showing a list of files in the "Downloads" folder. The files listed are:

Name	Date modified	Type	Size
RBC_FA.pdf.pdf	12/12/2023 16:58	Microsoft Edge P...	715 KB
Generic Cover Letter.pdf	12/12/2023 16:09	Microsoft Edge P...	58 KB
Finance Resume.pdf	12/12/2023 16:07	Microsoft Edge P...	109 KB
Fintech Resume.pdf	12/12/2023 16:07	Microsoft Edge P...	109 KB

The "Choose File" input field in the job posting form is highlighted, indicating it is active. The "CREATE" button is also visible.



Industry	Banking
HomePageURL	https://www.rbcroyalbank.com
Description	Royal Bank of Canada is a Canadian multinational financial services company and the largest bank in Canada by market capitalization. <small>(Wikipedia)</small>
UPDATE	

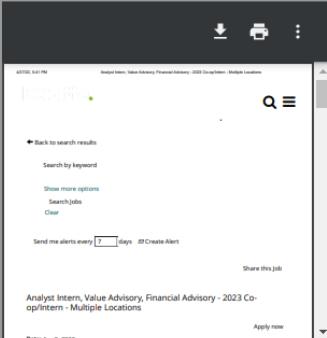
Jobs

Add New Job

PositionName	Financial Analyst
Description	Detailed job description...
ApplicationDeadline	11/30/2023
PositionType	Internship
Salary	52000
CREATE	
Choose File RBC_F...df.pdf	

Once the job has been added, it will appear as such. Note that applicants are also able to edit any of the fields below, then press Update to update the job entry.

Jobs

PositionName Financial Analyst	ApplicationDeadline 11/30/2023	Qualifications Enter New Qualifica	Responsibilities Enter New Responsibilit
PositionType Internship	Salary 52000	+ +	
Description Detailed job description...		<input type="checkbox"/> Track This Job	
Job Posting 			
<input type="button" value="Choose File"/> No file chosen Existing File: RBC_FA - [2].pdf <input type="button" value="UPDATE"/>			

With a job added, applicants have the option of adding multiple qualifications and responsibilities to the job, as shown towards the right of the screen. To add these, applicants can simply enter the new qualification and/or responsibility and press the corresponding + icon, as shown below.

Jobs

This screenshot shows a job posting form with the following fields:

- PositionName: Financial Analyst
- ApplicationDeadline: 11/30/2023
- Qualifications: Minimum 2 years of
- Responsibilities: Build and monitor financ
- PositionType: Internship
- Salary: 52000
- Description: Detailed job description...
- Track This Job checkbox (unchecked)

Once added, qualifications and responsibilities will appear as follows. While the entire text does not fit on the screen for some qualifications, users can hover over the text to read it in full.

Jobs

This screenshot shows the same job posting form with expanded qualification and responsibility sections. The qualifications and responsibilities are now listed in scrollable boxes:

- Qualifications:
 - Minimum 2 years o...
 - 3.7+ GPA
 - At least 1 finance-r...
 - Prior experience wi...
- Responsibilities:
 - Build and monitor ...
 - Continuously evaluate ongoing portfolio of assets
 - Continuously evalua...
 - Enter New Responsibility

The 'Description' and 'Track This Job' fields remain the same as in the previous screenshot.

Finally, applicants that may wish to track certain jobs have the option to do so by simply ticking the Track This Job box on the screen, which is visible in the screenshot above unticked and will provide additional fields to fill out when ticked, as shown below.

Jobs

PositionName

ApplicationDeadline

 CALENDAR

PositionType

Salary

Description

Detailed job description...

Qualifications

Minimum 2 years o... X
3.7+ GPA X
At least 1 finance-r... X
Prior experience wi... X

Responsibilities

Build and monitor ... X
Pitch financial sec... X
Continuously evalua... X

Enter New Qualifica

+

Enter New Responsibil

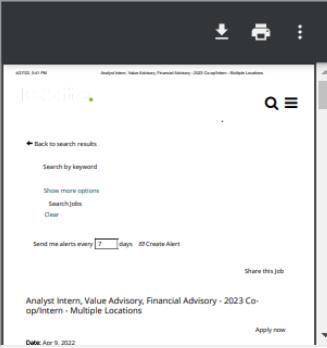
+

Track This Job

DateToApply

 CALENDAR

Notes



UPDATE

Choose File

Existing File: RBC_FA - [4].pdf

UPDATE

Once the job is being tracked, applicants can enter information specific to tracking, such as a target date to apply and any notes, and then press the Update button to save that information.

82

Track This Job

DateToApply
12/22/2023 

Notes

Could be an interesting experience; may be able to apply my technical skills well

UPDATE

Applicants are able to search the Companies page by Company Names and/or the application deadlines, as shown below. For example, searching for IBM will bring up the IBM company and its corresponding jobs.

Search by:

Only Show Jobs I Track
Sort Ascending Sort Descending

LatestApplicationDeadline

Description

American multinational technology corporation present in over 175 countries. It specializes in computer hardware, middleware, and software, and provides hosting and consulting services. (Wikipedia)

UPDATE

Jobs

CALENDAR

+

+

Detailed job description...

Track This Job

Alternatively, applicants can enter just the application deadline dates to search by jobs that match the date(s) criteria.

84

Search by:

CompanyName –

EarliestApplicationDeadline – 06/01/2023

LatestApplicationDeadline – 12/01/2023

Alternatively, applicants are able to sort companies in ascending or descending order, and only display jobs that they are personally tracking in the system. The Tracking filter will apply automatically, while sorting will only be applied once the search Icon is pressed, like it was for regular searching above. Note that if the Tracking filter is selected, all companies will still be displayed (if there is no company search applied), but only tracked jobs will display under each company. This means that if an applicant does not track any jobs from a specific company, it will appear as if the company has no jobs posted for it in the system.

Only Show Jobs I Track

Sort Ascending Sort Descending

Search by:

MM/DD/YYYY

Sort Ascending

Sort Descending

MM/DD/YYYY

MM/DD/YYYY

MM/DD/YYYY

MM/DD/YYYY

🔍

Amazon

Tech (Various)
https://www.amazon.ca/

Description

Amazon.com, Inc. is an American multinational technology company focusing on e-commerce, cloud computing, online advertising, digital streaming, and artificial intelligence.
(Wikipedia)

UPDATE

Jobs

Add New Job

MM/DD/YYYY

Choose File
No file chosen

CREATE

Applications

Upon first loading the Applications page, applicants will encounter a blank page where they are able to add their applications.

86

The screenshot shows the ApptTrack application management interface. At the top, there is a blue header bar with the logo 'ApptTrack', the title 'APPLICATIONS', and user information 'fredprokop1@gmail.com LOGOUT'. Below the header, there is a search bar with fields for 'AName', 'Status', and 'Category', followed by a magnifying glass icon. A large white box contains a form titled 'Add New Application' with fields for 'Application Name' (containing 'RBC Application'), 'Notes' (containing 'Some notes...'), 'Status' (containing '1-way interview compl'), and 'DateSubmitted' (containing '11/18/2023'). There is also a date picker icon and a 'CREATE' button.

An application can be added by simply populating the fields and pressing Create, as shown below.

This screenshot shows the 'Add New Application' form with the following values entered:

- Application Name: RBC Application
- Notes: Some notes...
- Status: 1-way interview compl
- DateSubmitted: 11/18/2023

Once added, an application appears as shown below and can be edited by editing any of the fields in the top row and pressing the Update button on the right.

This screenshot shows the application details page for 'RBC Application'. The top row contains edit fields for 'Application Name' (RBC Application), 'Notes' (Some notes...), 'Status' (1-way interview complete), and 'DateSubmitted' (11/18/2023). To the right of these fields is an 'UPDATE' button and a bin icon. Below this row, there are sections for 'Category' (Enter New Category), 'Relevant URLs' (Enter New RelevantU), 'Submit With' (Documents), 'Corresponds To' (Jobs), and additional input fields for 'Docum.' (Enter New JobPost) and 'Job' (Enter New JobPost).

Additionally, any application can be deleted by pressing the Bin icon on the top right. If an applicant wishes to delete an application, they will first have to confirm the action in the pop up window, as shown below.

The screenshot shows the 'Application' management screen. At the top, there are fields for 'Application Name' (RBC Application), 'Notes' (Some notes...updated), 'Status' (1-way interview complete), and 'DateSubmitted' (11/18/2023). A large 'UPDATE' button is visible. Below these, sections for 'Category', 'Relevant URLs', 'Submit With', and 'Corresponds To' are shown. Under 'Category', there is a field 'Enter New Category' with a '+' sign. Under 'Relevant URLs', there is a field 'Enter New RelevantU' with a '+' sign. Under 'Submit With', there is a field 'Documents' with a '+' sign. Under 'Corresponds To', there is a field 'Jobs' with a dropdown arrow and a field 'Enter New JobPost' with a '+' sign. In the center, a modal window titled 'Delete 2?' asks 'Are you sure you want to delete 2?'. It has 'CANCEL' and 'CONFIRM DELETE' buttons. In the background, there is a section for 'Add New Application' with similar fields and a 'CREATE' button.

For existing applications, applicants can add even more details by specifying categories for each application, relevant URLs, specific documents submitted for each application, and the job(s) that the application corresponds to. Categories and Relevant URLs are custom and can thus be entered into their fields as shown below.

This screenshot shows the application form with added details. The 'Category' field contains 'Finance' with a delete icon (X). The 'Relevant URLs' field contains 'www.url1.com' with a delete icon (X). Below these fields are buttons for adding new entries: '+ Enter New Category' and '+ Enter New RelevantU'.

Once added, Categories and Relevant URLs will appear as follows and can be deleted by simply pressing the X icon.

This screenshot shows the application form with added details. The 'Category' field contains 'Finance' with a delete icon (X). The 'Relevant URLs' field contains 'www.url1.com' and 'www.url2.com', each with a delete icon (X). Below these fields are buttons for adding new entries: '+ Enter New Category' and '+ Enter New RelevantU'.

To add submitted documents and jobs to the application, the documents and jobs must already exist in the database, as the adding is done using dropdown menus that only show existing options! Unfortunately, this was difficult to screenshot, but notice that the input fields for Document and Job have downward facing arrows, which represent dropdown boxes.

Submit With	Corresponds To	
<u>Documents</u>	<u>Jobs</u>	
<input type="button" value="Docum.▼"/> +	<input type="button" value="Job ▼"/> +	<input type="button" value="Enter New JobPost"/>

For each job, applicants can also add specific URLs relevant to the job itself that may not appear in the Companies page. Once Documents and Corresponding jobs have been added, they will appear as follows and can be hovered over to show the full text. These entries can also be deleted by simply pressing the X icon.

Submit With	Corresponds To	
<u>Documents</u>	<u>Jobs</u>	
<input type="button" value="2 - Finance Resum..."/> ×	<input type="button" value="3 - www.someurl.c..."/> ×	
<input type="button" value="Docum.▼"/> +	<input type="button" value="Job ▼"/> +	<input type="button" value="Enter New JobPost"/>

With applicants needing to handle tens or even hundreds of applications, they can easily search for applications by name, status, and category using the search bar at the top of the page. For example, searching for Status “Submitted” will return applications where the searched text is included in the Status field, as shown below. Searching by Name and/or Category will yield results using those categories.

Search by:

Application

Application Name	<input type="text" value="OneVest Application"/>	Notes	<input type="text" value="Some notes..."/>	Status	<input type="text" value="Submitted"/>	Date Submitted	<input type="text" value="11/18/2023"/>	
Category	<input type="text" value="Fintech"/>	Relevant URLs	<input type="text" value="www.url1.com"/> <input type="text" value="www.url2.com"/>	Submit With	<input type="text" value="Documents"/>	Corresponds To	<input type="text" value="Jobs"/> <input type="text" value="4 - www.onevesturl..."/>	
<input type="text" value="Enter New Category"/>	<input type="text" value="Enter New RelevantU"/>	<input type="text" value="Docum..."/>	<input type="text" value="Job"/>	<input type="text" value="Enter New JobPost"/>				

Contacts

Upon first loading the Contacts page, applicants will typically encounter previously populated Contact entries, as these are shared across all applicants. However, a blank Contacts page is also provided for reference, below, where applicants can add new contacts.

ApLitTrack **CONTACTS** fredprokop1@gmail.com **LOGOUT**

Only Show Contacts I Know Search by:

Add New Contact

A contact can be added by simply completing the required fields and pressing Create.

Add New Contact

First Name: Jeff

Last Name: Jefferson

LinkedInURL: https://www.linkedin.com

CREATE

Once a contact has been added, they will initially appear as shown below. The contact's first and last names, as well as their LinkedIn URL, can then be updated by changing the respective fields and pressing Update.

ApLitrack

CONTACTS

fredprokop1@gmail.com LOGOUT

Only Show Contacts I Know

Search by:

First Name: Bob

Last Name: Johnson

LinkedInURL: https://www.linkedin.com

UPDATE

I know this person!

Contact Info

Phone Number(s)

Email(s)

Enter New Phone + Enter New Email +

Works At

Companie(s)

Company + Enter New Role +

Applicants can then add multiple Phone Numbers and Emails for each contact by typing them into their respective fields and clicking the + buttons.

First Name Last Name **UPDATE**

I know this person!

LinkedInURL **UPDATE**

Contact Info

Phone Number(s) Email(s)

+ +

Once added, phone numbers and emails will appear as follows, and can be deleted by simply pressing the X icons.

First Name Last Name **UPDATE**

I know this person!

LinkedInURL **UPDATE**

Contact Info

Phone Number(s) Email(s)

403-000-0001 403-000-0002 bob_johnson@gm... bob_johnson@outl...
bob_johnson@rbc....

Enter New Phone + Enter New Email +

Next, applicants can also specify which companies each contact works at using the Works At field, which contains a dropdown showing existing companies in the database. Applicants are able to choose a company and specify the specific role that the contact holds at that company, as shown below, and then add the entry by pressing the + icon on the right.



I know this person!

First Name	Bob	Last Name	Johnson	UPDATE	
LinkedIn URL	https://www.linkedin.com			UPDATE	
Contact Info					
Phone Number(s)	403-000-0001 X 403-000-0002 X		Email(s)	bob_johnson@gm... X bob_johnson@outl... X bob_johnson@rbc... X	
Enter New Phone		+ +	Enter New Email		+ +
Works At					
Companie(s)	<input style="width: 100px; height: 25px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text" value="Company"/> RBC				Managing Director + +

Once added, the Works At section will appear as such. Note that multiple positions can be added for each contact, as shown below.

Works At

Companie(s)

RBC - Managing Di... X OneVest - Board M... X

Company	OneVest	Board Member	+ +
---------	---------	--------------	------------

With many contacts in the system, applicants can further specify whether they personally know a particular contact by selecting the ‘I know this person!’ checkbox, which will then provide additional fields for the applicant to fill in.



 I know this person!

Relationship

LastContactDate

 (Calendar)

Notes

First Name

Last Name

LinkedInURL

UPDATE

Contact Info

Phone Number(s)

403-000-0001
X
403-000-0002
X

Email(s)

bob_johnson@gm...
X
bob_johnson@outl...
X

bob_johnson@rbc...
X

Enter New Phone +

Enter New Email +

Referral(s)

Add New Referral

Date

 (Calendar)

Company

Notes

Job

Works At

Companie(s)

Company

Board Member

CREATE

Applicants can fill in the relationship details and press Update to save them, as follows.



I know this person!

Relationship
Professional

Last Contact Date
09/10/2023 

Notes
Met at financial networking event

UPDATE

Finally, applicants are also able to specify whether the contacts they know have provided any referrals, and if so, then for which existing job in the system. This can be done by filling in the fields and selecting the right Job under Referral(s), as shown below.

Referral(s)

Add New Referral

Date
09/10/2023 

Notes
Reference for Financial Analyst internship at RBC

Job
3 - RBC - [Financial Analy] 

CREATE

Once added, a referral appears as follows and can be edited by editing any of the fields and pressing the Update button. Additionally, the referral can be deleted by pressing the Bin icon, but only after confirming the deletion in a pop-up, as shown below.

Referral(s)

Delete a referral to RBC - Financial Analyst - [3]?

Are you sure you want to delete a referral to RBC - Financial Analyst - [3]?

[CANCEL](#) [CONFIRM DELETE](#)

Since there will be a large number of contacts stored in the system with multiple users, applicants can filter the Contacts page to only show contacts that they know by enabling the ‘Only Show Contacts I Know’ filter. This filter is enabled as soon as it is clicked, as shown below, and instantly provides more options for searching contacts.

Only Show Contacts I Know

Search by: First Name Last Name EarliestLastContactDate MM/DD/YYYY LatestLastContactDate MM/DD/YYYY

Bob Johnson

I know this person!

Relationship:

LastContactDate:

Contact Info

Phone Number(s):

Email(s):

Without the filter applied, applicants can typically only search for contact by First and Last Name, but can also search by Earliest and Latest Contact Date when the filter is applied. Below are examples of searching by Contact Date and by Names.

Only Show Contacts I Know

Search by: First Name Last Name EarliestLastContactDate 06/01/2023 LatestLastContactDate 11/01/2023

Bob Johnson

I know this person!

Relationship:

LastContactDate:

Notes: Met at financial networking event

Enter New Phone

Works At

Companie(s):

Referral(s):

The screenshot shows the ApliTrack Contacts page. At the top, there is a search bar with fields for First Name (Julia) and Last Name (J), a search icon, and a toggle switch for "Only Show Contacts I Know". Below the search bar is a contact card for "Julia Julien". The card includes a placeholder profile picture, a checkbox for "I know this person!", and update buttons for First Name (Julia), Last Name (Julien), and LinkedIn URL (https://www.linkedin.com). Below the card is a "Contact Info" section with fields for Phone Number(s) (403-000-0005) and Email(s) (julia_julien@amazon.com, julia_julien@outlook.com). There are also input fields for Enter New Phone and Enter New Email, each with a plus sign and a blue button labeled with a plus sign. Further down is a "Works At" section with a Company dropdown (Amazon) and an Enter New Role field.

Interviews

Upon first loading the Interviews page, applicants will encounter a blank page where they are able to add their interviews.

The screenshot shows the ApliTrack Interviews page. At the top, there are sorting options: "Sort by Stage Ascending" (selected), "Sort by Stage Descending", "Sort by Earliest Date", and "Sort by Latest Date". Below the sorting options is a search bar with fields for Stage, From-Date (MM/DD/YYYY), To-Date (MM/DD/YYYY), and a search icon. At the bottom of the page is an "Add New Interview" form with fields for Stage, Date (MM/DD/YYYY), Application, and a CREATE button.

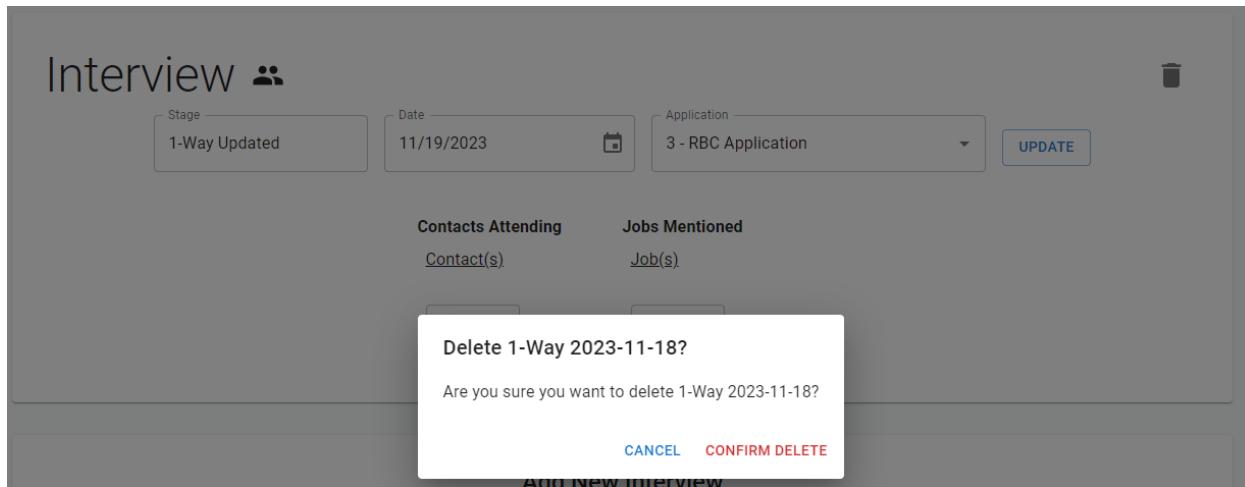
An interview can be added by filling in the fields under ‘Add New Interview’, then pressing Create. Note that while Stage and Date take any input from the user, the Application field is a dropdown menu that displays the applicant’s existing applications in the system.

The screenshot shows a form titled 'Add New Interview'. It contains three input fields: 'Stage' (set to '1-Way'), 'Date' (set to '11/18/2023'), and 'Application' (set to '3 - RBC Application'). To the right of these fields is a blue 'CREATE' button.

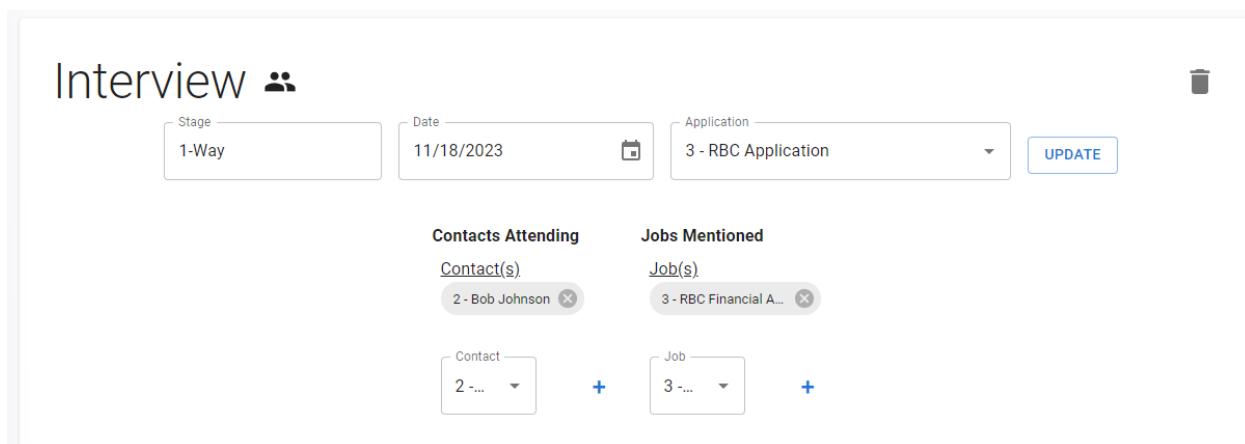
Upon creating an interview, it will be displayed as such. The interview can then be edited by updating any of the fields to the left of the Update button, and then pressing the Update button to save the changes.

The screenshot shows the 'Interview' details page. At the top, there are sorting options: 'Sort by Stage Ascending' (selected), 'Sort by Stage Descending', 'Sort by Earliest Date', and 'Sort by Latest Date'. Below these are search filters: 'Search by:' (Stage, From-Date/MM/DD/YYYY, To-Date/MM/DD/YYYY), a magnifying glass icon for search, and a bin icon for delete. The main section displays the interview details: Stage (1-Way), Date (11/18/2023), Application (3 - RBC Application), and an 'UPDATE' button. Below this, there are two sections: 'Contacts Attending' (Contact(s)) and 'Jobs Mentioned' (Job(s)). Each section has an 'Add' button (+) and a dropdown menu.

Additionally, the interview can be deleted by pressing the Bin icon, after which a confirmation window will pop up that looks as follows.



Existing interviews can then be supplemented with additional information, like any Contacts who are stored in the system who are attending the interview, as well as the specific jobs that the interview is related to. These can be selected in their corresponding dropdown menus that display entries that already exist in the application. Once added, attending contacts and mentioned jobs appear as follows. Note that each can be hovered over to display the full text.



Note that if duplicate items are attempted to be added by the applicant, they will see an error pop up informing them of this. For example, the error below shows up if the applicant attempts to add the same position to Jobs Mentioned, again.

DUPLICATE FIELD. PositionID must be unique: 3. Please use another PositionID.

With multiple interviews to manage, applicants are then able to search for interviews by the Stage field or by Date(s). An example of both search methods is provided below, first by Stage, then by a range of Dates. Note that a single From or To Date can also be used for searching.

The screenshot shows the ApliTrack Interviews page. At the top, there are four sorting options: "Sort by Stage Ascending" (selected), "Sort by Stage Descending", "Sort by Earliest Date", and "Sort by Latest Date". Below these are three search fields: "Search by: Stage" containing "Stage 1", "From-Date" set to "MM/DD/YYYY" (11/30/2023), and "To-Date" set to "MM/DD/YYYY" (11/30/2023). A search icon is located to the right of the date fields. The main content area is titled "Interview" with a user icon. It displays a row of search filters: "Stage: Stage 1", "Date: 11/30/2023", and "Application: 3 - RBC Application". An "UPDATE" button is to the right. Below this are sections for "Contacts Attending" and "Jobs Mentioned", each with dropdown menus for selecting contacts and jobs, respectively, with plus signs for adding more.

This screenshot shows the same ApliTrack Interviews page with different search parameters. The "Sort by Stage Ascending" option is still selected. The search filters are: "Search by: Stage" (empty), "From-Date" set to "11/20/2023", and "To-Date" set to "12/01/2023". The rest of the interface is identical to the first screenshot, including the interview details and attending contact/job sections.

Finally, applicants are able to Sort their interviews by Stage, ascending or descending alphabetically, and by Date, earliest or latest. Note that for the sorting option to take effect, the

search icon must be pressed, and that the default sorting method is by stage, ascending. Below are examples of sorting by stage, descending, and by latest date.

The screenshot shows the ApliTrack software interface with the title "INTERVIEWS". At the top, there are four sorting options: "Sort by Stage Ascending" (radio button), "Sort by Stage Descending" (radio button, selected), "Sort by Earliest Date" (radio button), and "Sort by Latest Date" (radio button). Below these are search fields for "Search by: Stage", "From-Date" (set to 11/30/2023), and "To-Date".

Interview 1: Stage 1, Date 11/30/2023, Application 3 - RBC Application. Contacts Attending: Contact(s) 2 - ... (with a plus sign and a delete icon). Jobs Mentioned: Job(s) 3 - RBC Application (with a plus sign and a delete icon).

Interview 2: Stage 1-Way, Date 11/18/2023, Application 3 - RBC Application. Contacts Attending: Contact(s) 2 - Bob Johnson (with a delete icon). Jobs Mentioned: Job(s) 3 - RBC Financial A... (with a delete icon).

Sort by Stage Ascending Sort by Stage Descending Sort by Earliest Date Sort by Latest Date

Search by: Stage From-Date To-Date

Interview

Stage Date Application

Contacts Attending **Jobs Mentioned**

Contact(s) Job(s)

Contact + Job +

Interview

Stage Date Application

Contacts Attending **Jobs Mentioned**

Contact(s) Job(s)

2 - Bob Johnson  3 - RBC Financial A... 

Contact + Job +

Offers

Upon first loading the Offers page, applicants will encounter a blank page where they are able to add their offers.

The screenshot shows the Applitrack Offers page. At the top, there are four radio buttons for sorting: "Sort by Compensation Ascending" (selected), "Sort by Response Deadline Descending", "Sort by Response Deadline Ascending", "Sort by Start Date Descending", "Sort by Start Date Ascending", and "Sort by Compensation Descending". Below these are search fields for "LowestCompensation" and "HighestCompensation", and date range fields for "EarliestResponseDeadline" (MM/DD/YYYY) and "LatestResponseDeadline" (MM/DD/YYYY), as well as "EarliestStartDate" (MM/DD/YYYY) and "LatestStartDate" (MM/DD/YYYY). A search icon is located at the bottom right of the search area.

Create Offer

Job Posting

ResponseDeadline MM/DD/YYYY StartDate MM/DD/YYYY Choose File No file chosen

Compensation Job

Notes

CREATE

An offer can be added by filling in the fields to the right of Create Offer. Note that the Job field is a dropdown that only allows for selection of jobs already in the system. Additionally, applicants are able to upload their job offer files to the system using the rightmost field. An example of a populated offer entry is shown below, which can be created by pressing the Create button.

The screenshot shows the Create Offer form with the following populated fields:

- ResponseDeadline: 01/05/2024
- StartDate: 05/01/2024
- Compensation: 52000
- Job: RBC - Financial Analy...
- Notes: Internship can be extended at end of term in August
- Job Posting: Choose File RBC_OFFER.pdf

Create Offer

Job Posting

ResponseDeadline 01/05/2024 StartDate 05/01/2024 Choose File RBC_OFFER.pdf

Compensation 52000 Job RBC - Financial Analy...

Notes
Internship can be extended at end of term in August

CREATE

Once created, an offer looks as follows. Note that any of the fields can be updated by changing their values and pressing the Update button at the bottom.

The screenshot shows a form titled "Offer" with a trophy icon. It includes fields for ResponseDeadline (01/05/2024), StartDate (05/01/2024), Compensation (52000), and a notes section stating "Internship can be extended at end of term in August". To the right, there's a "Job Posting" preview window showing a job listing for "Analyst Intern, Value Advisory, Financial Advisory - 2023 Co-op Intern - Multiple Locations". Below the preview is a "Choose File" button with "No file chosen" and an "UPDATE" button.

An offer can also be deleted by pressing the Bin icon on the top left, which creates a pop up window where the applicant must confirm the deletion, as shown below.

A modal dialog box titled "Delete your Offer from RBC for the Financial Analyst [3] Position?" is displayed. It contains the question "Are you sure you want to delete your Offer from RBC for the Financial Analyst [3] Position?", a "CANCEL" button, and a "CONFIRM DELETE" button in red. The background shows the same offer creation interface as the previous screenshot.

If applicants have multiple offers to decide between, they are able to easily search and compare offers by using the various search options provided, including searching by lowest and highest compensation, earliest and latest response deadline, and earliest and latest start date. For example, an application may search by lowest compensation and latest start date as follows, or even using all six fields. Note that the offer shows up in both results because it meets all criteria in both searches.

The screenshot shows the ApliTrack software interface. At the top, there is a navigation bar with the title "ApliTrack", the user email "fredprokop1@gmail.com", and a "LOGOUT" button. Below the navigation bar, the main area is titled "OFFERS".

Search filters are displayed at the top of the offers list:

- Sort by Compensation Ascending (radio button selected)
- Sort by Response Deadline Descending (radio button)
- Sort by Response Deadline Ascending (radio button)
- Sort by Start Date Descending (radio button)
- Sort by Start Date Ascending (radio button)
- Sort by Compensation Descending (radio button)

Below the filters, there is a "Search by:" section with six input fields:

- LowestCompensation: 50000
- HighestCompensation: (empty)
- EarliestResponseDeadline: MM/DD/YYYY (with calendar icon)
- LatestResponseDeadline: MM/DD/YYYY (with calendar icon)
- EarliestStartDate: MM/DD/YYYY (with calendar icon)
- LatestStartDate: 05/06/2024 (with calendar icon)

A magnifying glass icon is located next to the search fields.

The main content area displays an "Offer" record with the following details:

- ResponseDeadline:** 01/05/2024
- StartDate:** 05/01/2024
- Compensation:** 52000
- Job:** RBC - Financial Analy...
- Notes:** Internship can be extended at end of term in August for a maximum of 4 more months.

To the right of the offer details, a "Job Posting" window is open, showing a preview of a PDF document titled "Analyst Intern, Value Advisory, Financial Advisory, 2023 Co-op Intern - Multiple Locations". The PDF includes sections like "Job Description", "Requirements", and "How to Apply".

At the bottom of the offer detail view, there is a file upload section:

- "Choose File" button
- Text: "No file chosen"
- Text: "Existing File: RBC_OFFER - [6].pdf"
- "UPDATE" button

Sort by Compensation Ascending Sort by Response Deadline Descending

Sort by Response Deadline Ascending Sort by Start Date Descending

Sort by Start Date Ascending Sort by Compensation Descending

Search by:

LowestCompensation

50000

HighestCompensation

60000

EarliestResponseDeadline

12/18/2023



LatestResponseDeadline

01/31/2024



EarliestStartDate

04/22/2024



LatestStartDate

05/06/2024



Offer



ResponseDeadline	01/05/2024	
Compensation	52000	

StartDate	05/01/2024	
Job	RBC - Financial Analy...	

Notes

Internship can be extended at end of term in August for a maximum of 4 more months.

Job Posting

The screenshot shows a job posting interface. At the top, there are navigation icons for back, forward, search, and print. Below that is a search bar with placeholder text "Search by keyword". Underneath the search bar are buttons for "Show more options", "Search jobs", and "Clear". There is also a section for "Send me alerts every [] days" with a "All Create Alert" option. The main content area displays a resume titled "Analyst Intern, Value Advisory, Financial Advisory - 2023 Co-op Intern - Multiple Locations". The resume includes sections for "About", "Education", "Experience", and "Skills". At the bottom of the resume, there is a "Share this job" link and an "Apply now" button.

Choose File No file chosen

Existing File: RBC_OFFER - [6].pdf

UPDATE

Finally, applicants can also sort their offers in addition to or instead of searching. Note that for the sorting option to take effect, the search icon must be pressed underneath the search fields. For example, sorting by highest to lowest compensation will look as follows.

Sort by Compensation Ascending Sort by Response Deadline Descending
 Sort by Response Deadline Ascending Sort by Start Date Descending
 Sort by Start Date Ascending Sort by Compensation Descending

Search by:

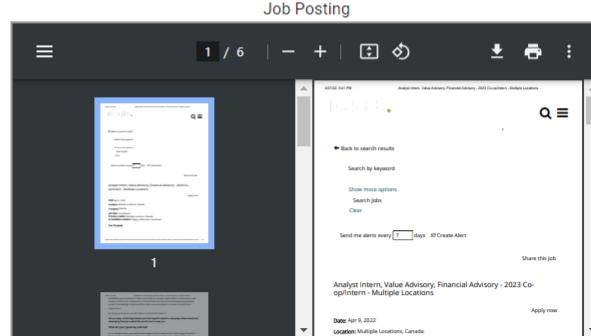
LowestCompensation	HighestCompensation
EarliestResponseDeadline <input type="date"/>	LatestResponseDeadline <input type="date"/>
EarliestStartDate <input type="date"/>	LatestStartDate <input type="date"/>

Offer 🏆

ResponseDeadline	01/05/2024 <input type="date"/>
StartDate	05/01/2024 <input type="date"/>
Compensation	52000
Job	RBC - Financial Analy...

Notes

Internship can be extended at end of term in August for a maximum of 4 more months.



No file chosen

Existing File: RBC_OFFER - [6].pdf

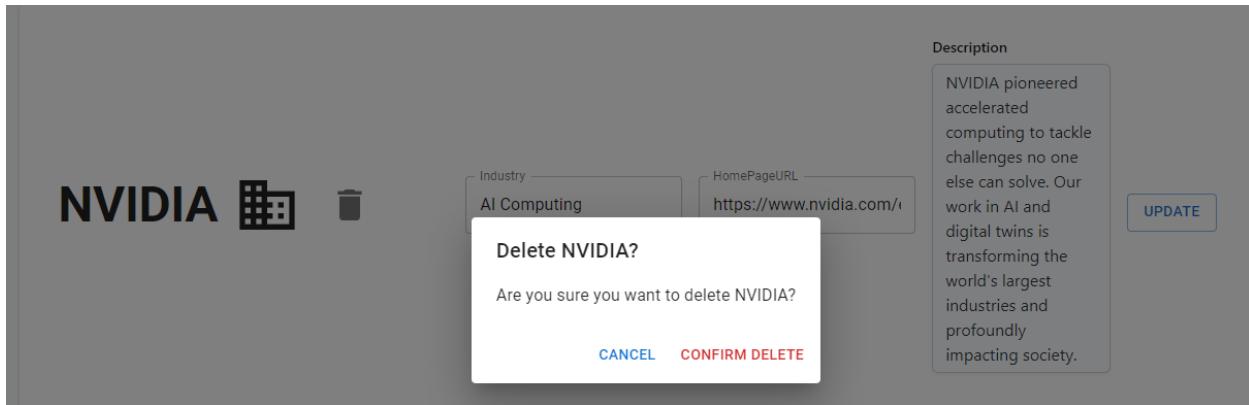
Users (Admin)

Upon loading the user page (which admins can only access) the admins will be able to see a list of users that currently exist on the system/database. From here they are able to update user's first and last names, activate and deactivate their accounts, change their permission levels and reset their passwords. They can even create or delete users from the database. Admins also have the option to search for users using their first or last names.

Additional Admin Actions

As discussed in the Project Design section, since jobs, contacts, and companies are shared in the system across all users to prevent duplication, given that these entities are not unique to a specific applicant, only admins are allowed to delete these entities from the database to prevent malicious users from making these deletions.

Admins are able to delete companies by pressing the Bin icon next to the desired company, as shown below. Note that deletion must then be confirmed through a pop-up window.



Similarly, jobs can be deleted by pressing the Bin icon under a job, as shown below. As with companies, deletion must then be confirmed through a pop-up window.

Jobs

Software Internship

PositionName	Software Internship	ApplicationDeadline	03/30/2023
PositionType	Internship	Salary	65000

Qualifications

- 2-3 years of post-s...
- Must have experie...
- Prior experience w...

Responsibilities

- Back-end impleme...
- Research and sugg...

Description

Detailed job description...

Track This Job

Enter New Qualification

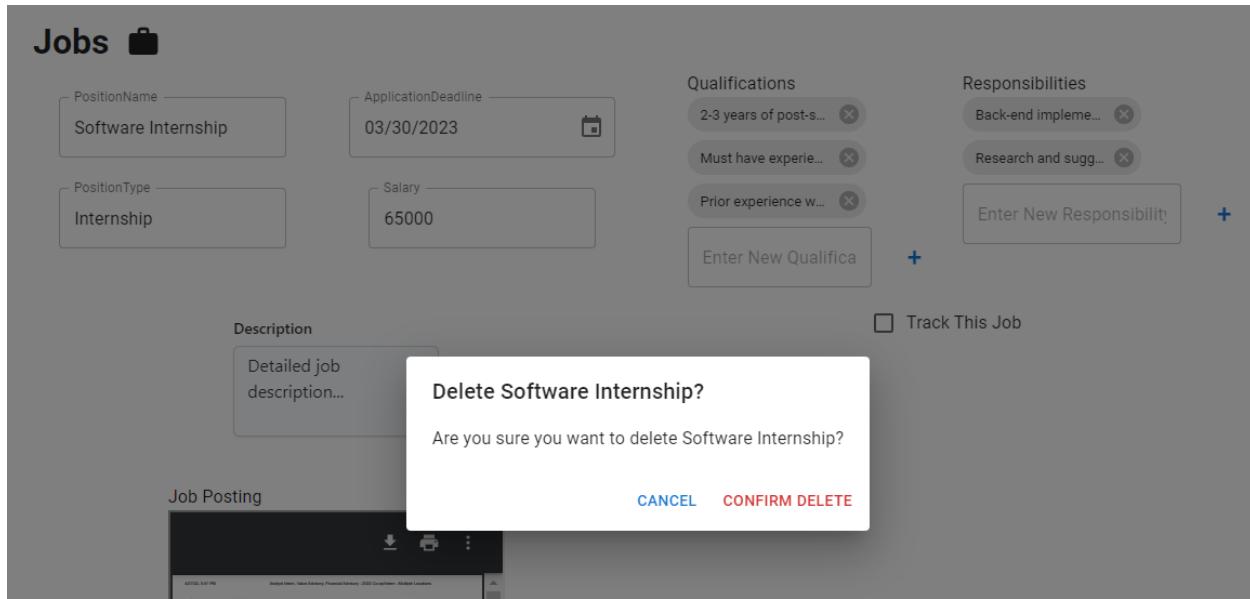
Enter New Responsibility



Choose File No file chosen

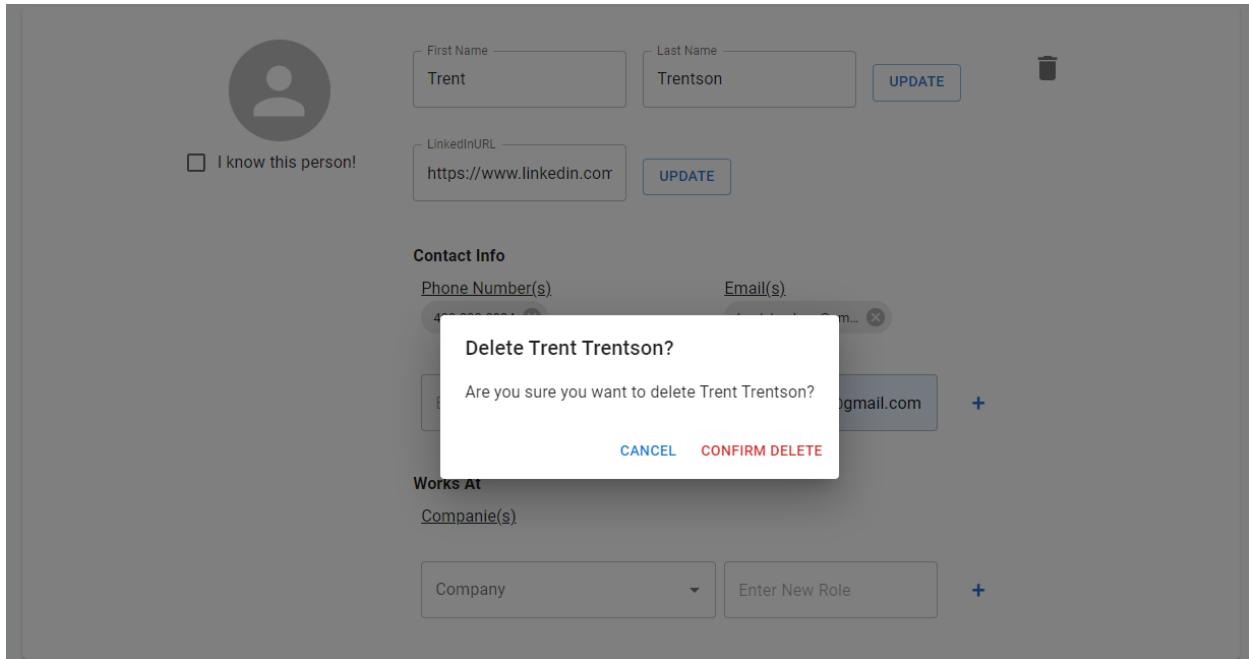
Existing File: NVIDIA_Internship - [7].pdf





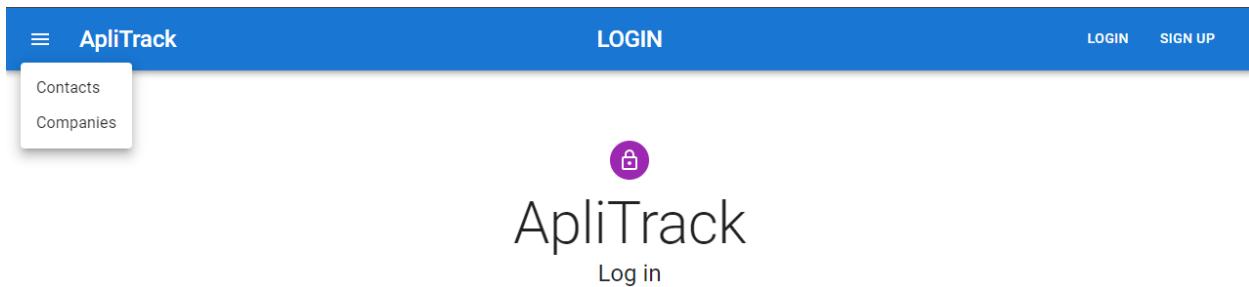
Finally, contacts can also be deleted by pressing the Bin icon to the right of a contact, as shown below. As with companies and jobs, deletion must then be confirmed through a pop-up window.

The screenshot shows a contact profile for 'Trent Trentson'. The profile includes fields for First Name ('Trent'), Last Name ('Trentson'), and an 'UPDATE' button. Below this, there is a 'LinkedInURL' field with the value 'https://www.linkedin.com/' and an 'UPDATE' button. A checkbox labeled 'I know this person!' is checked. A 'Contact Info' section contains 'Phone Number(s)' ('403-000-0004') and 'Email(s)' ('trent_trentson@gm...'). Below this, there are fields for 'Enter New Phone' and an email address 'fredprokop1@gmail.com'. A 'Works At' section shows a dropdown for 'Company' and a field for 'Enter New Role'. A bin icon is visible to the right of the contact name.



Logged-Out Views

Finally, users who are not logged in are able to view the Contacts and Companies pages, as these entries are public and not specific to any particular applicants. However, since the users are not logged in, they are unable to make any additions or updates to the data in the system, as these actions require users to be logged in. Below are the sample views for the Contacts and Companies pages, as seen by a logged-out user or a user with no existing account.



Search by: 



First Name
Last Name
LinkedInURL

Contact InfoPhone Number(s)

[403-000-0001](#) [403-000-0002](#)

Email(s)

bob_johnson@gm... bob_johnson@outl...
bob_johnson@rbc....

Works AtCompanie(s)

[OneVest - Board M...](#) [RBC - Managing Di...](#)

Search by:

Sort Ascending Sort Descending MM/DD/YYYY MM/DD/YYYY 

Amazon

 Tech (Various) <https://www.amazon.ca/>**Description**

Amazon.com, Inc. is an American multinational technology company focusing on e-commerce, cloud computing, online advertising, digital streaming, and artificial intelligence. (Wikipedia)

Jobs

 Cloud Engineer 09/30/2023 **Qualifications**

- Must be able to de...
- Prior experience wi...
- Software engineer...

Responsibilities

- Assist in monitor...
- Implement new clo...

 Entry Level 105000**Description**

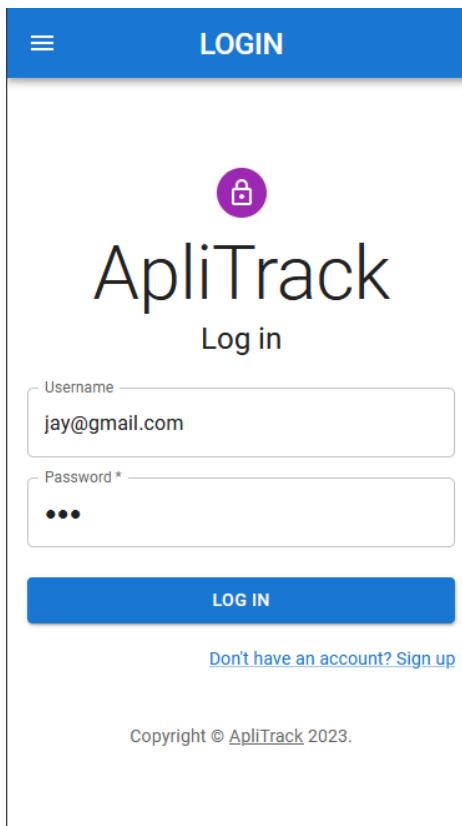
Detailed job description...

Mobile and Tablet Views

Since our group spends considerable time on mobile or tablet devices, we had an incentive to make our application compatible with both mobile and tablet screen sizes. In addition, at the start of the semester, it was also mentioned that a bonus mark would be awarded for mobile-responsive applications (providing yet another incentive). See screenshots below of both mobile and tablet views.

Login Page

Mobile View



Tablet View



ApliTrack

Log in

Username —

Password * —

LOG IN

[Don't have an account? Sign up](#)

Copyright © ApliTrack 2023.

Sign-Up Page

Mobile View



SIGNUP



ApliTrack

Sign up

- First Name * -

jay

- Last Name * -

yeo

- Username -

jay@gmail.com

- Password * -

•••

- Confirm Password * -

•••

SIGN UP

[Already have an account? Sign in](#)

Copyright © ApliTrack 2023.

Tablet View



ApliTrack

Sign up

First Name * —

Last Name * —

Username —

Password * —

Confirm Password * —

SIGN UP

[Already have an account? Sign in](#)

Copyright © ApliTrack 2023.

Profile Page

Mobile View



PROFILE

[LOGOUT](#)**YOU**

First Name

jay

Last Name

yeo

Education

High school

UPDATE

Current Password

New Password

Confirm New Password

UPDATE

EXPERIENCES

Volunteer work

volunteered at a charity event



Enter Title of New Experience (

Enter description of new Experience



PROJECTS

CPSC 471

Enter Title of New Project (Ma)

Enter description of new Project



CERTIFICATIONS

High school diploma X

Enter Name of New Certification +

SKILLS

C++ X Javascript X

Enter Name of New Skill (Max) +

COMPETITIONS

Hackathon X

Enter Name of New Competition +

Tablet View



YOU

First Name

Last Name

Education

UPDATE

Current Password

New Password

Confirm New Password

UPDATE**EXPERIENCES****Volunteer work**

volunteered at a charity event



Enter Title of New Experience (Max 32 Characters)

Enter description of new Experience

+**PROJECTS**

PROJECTS

CPSC 471



Enter Title of New Project (Max 32 Characters)

Enter description of new Project



CERTIFICATIONS

High school diploma

Enter Name of New Certification (Max 32 Characters)



SKILLS

C++ Javascript

CERTIFICATIONS

High school diploma X

Enter Name of New Certification (Max 32 Characters) +

SKILLS

C++ X Javascript X

Enter Name of New Skill (Max 32 Characters) +

COMPETITIONS

Hackathon X

Enter Name of New Competition (Max 32 Characters) +

Document Page

Mobile View

DOCUMENTS LOGOUT

Sort by Document Name
Descending

Sort by Document Name
Ascending

Search by: DocFileName

Search by:

Upload Document

DocType

Description
G 1

File Jayden Yeo ...me v5.docx

DOCUMENTS

Descending

Search by: DocFileName

Upload Document

DocType

Description
G 1

File Jayden Yeo ...me v5.docx

CREATE

Tablet View

Sort by Document Name Ascending Sort by Document Name Descending

Search by:



Upload Document

DocType

Resume

Description

Resume for my 1st internship



File

 Jayden Yeo Resume v5.docx

Company Page

Mobile View



COMPANIES

[LOGOUT](#)

Only Show Jobs I Track

Sort Ascending

Sort Descending

Search by:

CompanyName —

EarliestApplicationDeadline —
MM/DD/YYYY

LatestApplicationDeadline —
MM/DD/YYYY



Amazon

Industry —
Shipping

HomePageURL —
www.amazon.com

www.amazon.com

Description

Canada's largest online
retailer

[UPDATE](#)

Jobs Add New Job

PositionName —
Software developer

PositionType —
Internship

Salary —
\$2000

ApplicationDeadline —
12/15/2023

Description

Will mainly be unit testing

[Browse...](#) Jayden Yeo...e v5.docx

Jayden Yeo...e v5.docx

Add New Company

CompanyName

Industry

HomePageURL

Description

Tablet View

Search by:

CompanyName _____

Only Show Jobs I Track

Sort Ascending Sort Descending

EarliestApplicationDeadline _____
MM/DD/YYYY

LatestApplicationDeadline _____
MM/DD/YYYY



Amazon

Description

Canada's largest online retailer

UPDATE

Jobs

PositionName
Software developer

ApplicationDeadline
12/15/2023

PositionType _____

Salary _____

Qualifications

Enter New Qualification



Responsibilities

Jobs

PositionName
Software developer

ApplicationDeadline
12/15/2023

Qualifications

Enter New Qualification



PositionType
Internship

Salary
\$2000

Responsibilities

Enter New Responsibility



Description

Will mainly be unit testing

Track This Job

Job Posting

No file selected.

Existing File: Javden Yeo Resume v5 - [1].docx

No file selected.

Existing File: Jayden Yeo Resume v5 - [1].docx

Add New Job

PositionName	Description	ApplicationDeadline
<input type="text"/>	<input type="text"/>	MM/DD/YYYY <input type="text"/>

No file selected.

Add New Company

CompanyName	Industry	HomePageURL	Description
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Application Page

Mobile View



APPLICATIONS

[LOGOUT](#)

Search by:



Application



Application Name

Amazon application

Notes

Some notes...

Status

1-way interview completed

DateSubmitted

DateSubmitted

12/18/2023

UPDATE

Category

Software

Enter New Category



Relevant URLs

www.amazon.com

Enter New Relevant URL



Submit With

Documents

1 - Jayden Yeo Res...

Document

1 - ...



Corresponds To

Jobs

Corresponds To

Jobs

Job
 software developer

Job
 software developer

www.someURL... X

www.someURL... +

Job
 software developer

www.someURL... +

Add New Application

Application Name

 Amazon application

Notes

 Some notes...

Status

 1-way interview completed

DateSubmitted

 12/18/2023

CREATE

Tablet View

Search by: 

Application



Application Name Amazon application	Notes Some notes...	Status 1-way interview completed	DateSubmitted 12/18/2023	
Category Software 	Relevant URLs www.amazon.com 	Submit With Documents 1 - Jayden Yeo Res... 	Corresponds To Jobs 1 - www.someURL... 	
<input type="text" value="Enter New Catego"/> +	<input type="text" value="Enter New Relevan"/> +	<input type="text" value="Docum..."/> +	<input type="text" value="Job"/> +	<input type="text" value="Enter New Job"/> +

Add New Application

Notes

Application Name	Status	DateSubmitted MM/DD/YYYY	
------------------	--------	-----------------------------	--

Contact Page

Mobile View



CONTACTS

[LOGOUT](#)

Only Show Contacts I Know

Search by:

First Name

Last Name



I know this person!

Relationship

Co-worker

LastContactDate

12/01/2023

Notes

Met at a job search workshop

Notes

Met at a job search workshop

[UPDATE](#)

Referral(s)



Date

12/10/2023

Notes

Reference for Software
Developer at Amazon

Job

Amazon - Software dev ... ▾

[UPDATE](#)

Add New Referral

Date

12/10/2023

Add New Referral

Date

Notes

Job

CREATE

First Name

Last Name

UPDATE

LinkedInURL **UPDATE**

Contact Info

Phone Number(s) X

Enter New Phone +

Email(s) X

Enter New Email +

Works At

Companie(s) X

Company ▼

Software Developer +

Contact Info

Add New Contact

Company(s)

Amazon - Software... 

Company
A ... 

Software Develop^e 

Add New Contact



First Name

Last Name

LinkedInURL

CREATE

Tablet View

Only Show Contacts I Know

Search by:

 I know this person!

Relationship

Co-worker

LastContactDate

12/01/2023

Notes

Met at a job search workshop

Dave
Davidson
<https://www.linkedin.com>**Contact Info**Phone Number(s)

403-111-2222

Email(s)

dave@gmail.com

Enter New Phone

+

Enter New Email

+

Works AtCompanie(s)

Amazon - Software...

Referral(s)



Company

Amazon

Software Developer

+

Referral(s)



Company

Amazon

Software Developer

+

Date

12/10/2023

Notes

Reference for Software Developer at Amazon

Job

Amazon - Software dev ...

Add New Referral

Date

12/10/2023

Notes

Job

Amazon - Software dev ... ▾

UPDATE

Add New Referral

Date
12/10/2023

Notes

Job ▾

CREATE

Add New Contact

 First Name _____

Last Name _____

LinkedInURL _____

CREATE

Interview Page

Mobile View



INTERVIEWS

[LOGOUT](#)

Sort by Stage Ascending

Sort by Stage Descending

Sort by Earliest Date

Sort by Latest Date

Search by:

Stage

From-Date

MM/DD/YYYY

To-Date

MM/DD/YYYY



Interview

Stage

1-Way

Date

12/19/2023

Date

12/19/2023

Application▼

UPDATE

Contacts Attending

Contact(s)

Contact▼



Jobs Mentioned

Job(s)

Job ▼



Add New Interview

Stage

Contact(s)

Contact

+

Jobs Mentioned

Job(s)

Job ▾

+

Add New Interview

Stage

Date

12/19/2023

Application

1 - ... ▾

CREATE

Tablet View

ApITrack **INTERVIEWS** jay@gmail.com LOGOUT

Sort by Stage Ascending

Sort by Stage Descending

Sort by Earliest Date

Sort by Latest Date

Search by: Stage From-Date MM/DD/YYYY To-Date MM/DD/YYYY

Interview

1-Way 12/19/2023

Contacts Attending **Jobs Mentioned**

Contact(s) Job(s)

+ +

Sort by Latest Date

Search by: Stage From-Date MM/DD/YYYY To-Date MM/DD/YYYY

Interview

1-Way 12/19/2023

Contacts Attending **Jobs Mentioned**

Contact(s) Job(s)

+ +

Add New Interview

Stage Date 12/19/2023 Application 1 - Amazon application

Offer Page

Mobile View



OFFERS

LOGOUT

Sort by Compensation
Ascending

Sort by Response Deadline
Ascending

Sort by Start Date Ascending

Sort by Response Deadline
Descending

Sort by Start Date Descending

Sort by Compensation
Descending

Search by:

LowestCompensation

HighestCompensation

EarliestResponseDeadline
MM/DD/YYYY

LatestResponseDeadline
MM/DD/YYYY

EarliestStartDate
MM/DD/YYYY

EarliestStartDate
MM/DD/YYYY

LatestStartDate
MM/DD/YYYY



Offer

ResponseDeadline
12/22/2023

StartDate
12/25/2023

Compensation
\$2000

Job
Amazon - Software d ... ▾

Notes

Internship can be extended

Job Posting

[1] of 2

Job Posting



Browse... No file selected.

Existing File: Unofficial_Transcript - [1...]

UPDATE

Create Offer

ResponseDeadline _____
MM/DD/YYYY

Create Offer

ResponseDeadline _____
MM/DD/YYYY

StartDate _____
MM/DD/YYYY

Compensation _____

Job **Job** ▾

Notes

C

Job Posting

Browse... No file selected.

CREATE

Table View

Sort by Compensation Ascending Sort by Response Deadline Descending

Sort by Response Deadline Ascending Sort by Start Date Descending

Sort by Start Date Ascending Sort by Compensation Descending

Search by:

LowestCompensation	HighestCompensation
--------------------	---------------------

EarliestResponseDeadline MM/DD/YYYY	LatestResponseDeadline MM/DD/YYYY
--	--------------------------------------

EarliestStartDate MM/DD/YYYY	LatestStartDate MM/DD/YYYY
---------------------------------	-------------------------------



Offer



ResponseDeadline

12/22/2023

StartDate

12/25/2023

Compensation

\$2000

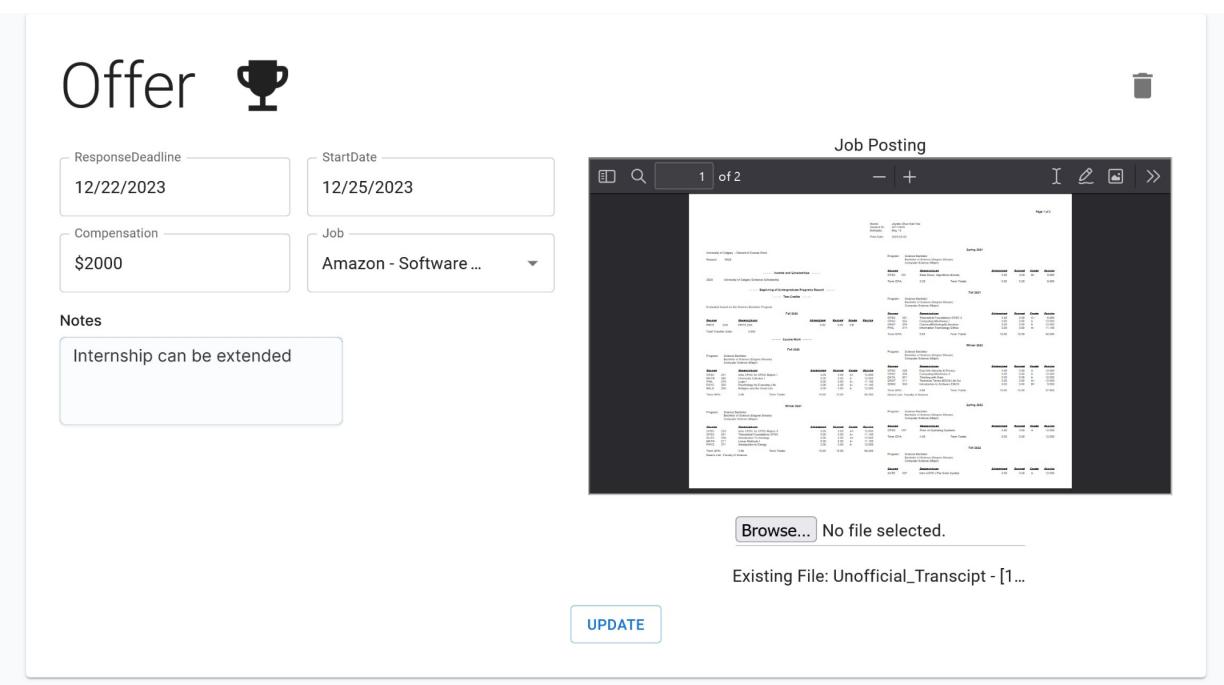
Job

Amazon - Software ...

Notes

Job Posting

Offer



<input type="text" value="MM/DD/YYYY"/>	<input type="text" value="MM/DD/YYYY"/>	Job Posting
ResponseDeadline	StartDate	Browse... No file selected.
<hr/>		
<input type="text" value="Compensation"/>	<input type="text"/>	
<hr/>		

Internship can be extended





No file selected.

Existing File: Unofficial_Transcript - [1...]

Create Offer

ResponseDeadline

StartDate

Compensation

Job

Notes

Users Page (Admin)

Mobile View

The wireframe illustrates a user management application layout on a tablet. The top navigation bar includes a menu icon, the word "USERS", and a "LOGOUT" link. On the left side, there's a search section labeled "Search by:" with fields for "First Name" and "Last Name", and a magnifying glass icon. Below the search is a user profile card for "Jill Bobson" with edit fields for "First Name" (Jill), "Last Name" (Bobson), and "Account Status" (Activated). There are "UPDATE" and "DELETE" buttons. The right side shows a detailed view for "Bobson" with fields for "Last Name" (Bobson), "Account Status" (Activated), and "Permission Level" (1). It includes "UPDATE" and "RESET PASSWORD" buttons. A separate "Add New User" card on the right has fields for "First Name" and "Last Name" and a "CREATE" button.

USERS LOGOUT

Search by:

First Name

Last Name

Jill Bobson

First Name Jill

Last Name Bobson

Account Status Activated ACTIVATE DEACTIVATE

Permission Level 1

Add New User

First Name

Last Name

Tablet View

Search by:

First NameLast Name

First Name

Jill

Last Name

Bobson

UPDATE

Account Status

Activated

ACTIVATE

DEACTIVATE

Permission Level

1

UPDATE

RESET PASSWORD

Add New User



First Name

Last Name

CREATE

Sample Data Instances

Please find below all sample data, shown for each table, used to populate the database.

APPLICANT

Username	Education
fredprokopchuk9991@gmail.com	Test education
fredprokop@gmail.com	BComm, Finance Major
fredprokop1@gmail.com	BComm, Finance Major, CPSC Minor
test@gmail.com	NULL
prempreeet@gmail.com	BComm & BSc CPSC Dual Degree
prempreet1@gmail.com	BComm & BSc CPSC Dual Degree
prem@gmail.com	test edu
prem2@gmail.com	educationupdated
prempreetb@gmail.com	BComm & BSc CPSC Dual Degree
jayden@gmail.com	Bsc, AI&ML Specialization

APPLICANT_CERTIFICATION

ApplicantUsername	Certification
fredprokop@gmail.com	Financial Modelling
fredprokop@gmail.com	Respect in the Workplace
fredprokop1@gmail.com	Respect in the Workplace
fredprokop1@gmail.com	Financial Modelling
prempreetb@gmail.com	Full Stack Advanced Certificate
jayden@gmail.com	Quantum Computing Introduction
jayden@gmail.com	Intermediate Quantum Theory

APPLICANT_COMPETITION

ApplicantUsername	Competition
fredprokop@gmail.com	TRMC2023 Case Competition
fredprokop1@gmail.com	TRMC2023 Case Competition
prempreetb@gmail.com	Hackathon 2022 April - 1st Place
jayden@gmail.com	Full-Stack Hackathon 2023 - 1st Place

APPLICANT_EXPERIENCE

id	ApplicantUsername	Experience	ExperienceDesc
3	fredprokop@gmail.com	Tech Start	Tech Start club description
4	fredprokop@gmail.com	Deloitte	Deloitte internship description
5	fredprokop@gmail.com	ATB	ATB internship description
6	fredprokop1@gmail.com	ATB	ATB internship description
7	fredprokop1@gmail.com	Deloitte	Deloitte internship description
8	fredprokop1@gmail.com	Tech Start	Tech Start club description
10	prempreetb@gmail.com	Software Internship	Software Internship description
11	prempreetb@gmail.com	Amazon Internship	Amazon Internship description
12	jayden@gmail.com	Amazon Internship	Detailed internship description
13	jayden@gmail.com	Facebook Internship	Detailed internship description
14	jayden@gmail.com	Google Internship	Detailed internship description
15	jayden@gmail.com	City of Calgary internship	Less detailed internship description

APPLICANT_PROJECT

id	ApplicantUsername	Project	ProjectDesc
1	fredprokop@gmail.com	CPSC 471 Project Updated	Database project description updated
2	fredprokop1@gmail.com	CPSC 471 Project	Database project description
3	prempreetb@gmail.com	Personal Project - AI	AI project description
4	prempreetb@gmail.com	Personal Project - ML	ML project description
5	jayden@gmail.com	Robotics Project	Project description...didn't enjoy much
6	jayden@gmail.com	Quantum Simulation	Loved it, 10/10 would do again
7	jayden@gmail.com	Front-end Project	Did not love it, would prefer not to do again

APPLICANT_SKILL

ApplicantUsername	Skill
fredprokop@gmail.com	Company valuation
fredprokop@gmail.com	Due diligence
fredprokop1@gmail.com	Company valuation
fredprokop1@gmail.com	Due diligence
prempreetb@gmail.com	Back-end development
prempreetb@gmail.com	Cloud services
prempreetb@gmail.com	Front-end development
jayden@gmail.com	Full-stack development
jayden@gmail.com	DevOps
jayden@gmail.com	Physics

APPLICATION

ApplicationID	ApplicantUsername	AName	Notes	DateSubmitted	Status
3	fredprokop1@gmail.com	RBC Application	Some notes...	2023-11-18	1-way interview complete
4	fredprokop1@gmail.com	OneVest Application	Some notes...	2023-11-15	Submitted
5	prempreetb@gmail.com	Amazon Application	Some notes...	2023-09-18	Final interview complete
6	prempreetb@gmail.com	Microsoft Application	Some notes...	2023-06-18	Rejected
7	jayden@gmail.com	NVIDIA Application	Some notes...	2023-03-25	Accepted
8	jayden@gmail.com	IBM Application	Some notes...	2023-11-15	Submitted

APPL_CATEGORY

ApplicationID	Category
3	Finance
4	Fintech
5	On-Site
6	Remote
7	Accepted
8	Submitted
7	Submitted

APPL_RELEVANT_URL

ApplicationID	RelevantURL
3	www.url1.com
3	www.url2.com
4	www.url1.com
4	www.url2.com
5	www.url2amz.com
5	www.url1amz.com
6	www.url1ms.com
6	www.url2ms.com
7	www.nvdurl1.com
7	www.nvdurl2.com
8	www.ibmurl1.com
8	www.ibmurl2.com

ATTENDS

ContactID	InterviewID
2	2
5	4
4	5

COMPANY

Table was recreated due to spacing issues.

CompanyName	Industry	HomePageURL	Description
RBC	Banking	https://www.rbcroyalbank.com/personal.html	Royal Bank of Canada is a Canadian multinational financial services company and the largest bank in Canada by market capitalization. (Wikipedia)
OneVest	Fintech	https://www.onevest.com/	With legacy experiences and traditional platforms, the wealth industry just hasn't kept up. OneVest believes in empowering challengers and incumbents alike to make investing better for everyone. (OneVest)
Amazon	Tech (Various)	https://www.amazon.ca/	Amazon.com, Inc. is an American multinational technology company focusing on e-commerce, cloud computing, online advertising, digital streaming, and artificial intelligence. (Wikipedia)

Microsoft	Tech (Various)	https://www.microsoft.com/en-ca	Microsoft Corporation is an American multinational technology corporation with a variety of software products. (Wikipedia)
NVIDIA	AI Computing	https://www.nvidia.com/en-us/	NVIDIA pioneered accelerated computing to tackle challenges no one else can solve. Our work in AI and digital twins is transforming the world's largest industries and profoundly impacting society.
IBM	Tech Hardware	https://www.ibm.com/ca-en	American multinational technology corporation present in over 175 countries. It specializes in computer hardware, middleware, and software, and provides hosting and consulting services. (Wikipedia)

CONTACT

ContactID	LinkedInURL	Fname	Lname
2	https://www.linkedin.com/in/bob-johnson/	Bob	Johnson
3	https://www.linkedin.com/in/jeff-jefferson/	Jeff	Jefferson
4	https://www.linkedin.com/in/trent-trentson/	Trent	Trentson
5	https://www.linkedin.com/in/julia-julien/	Julia	Julien
6	https://www.linkedin.com/in/amanda-trest/	Amanda	Trest

CONTACT_EMAIL

ContactID	Email
2	bob_johnson@gmail.com
2	bob_johnson@outlook.com
2	bob_johnson@rbc.ca
3	jeff_jefferson@gmail.com
4	trent_trentson@gmail.com
5	julia_julien@outlook.com
5	julia_julien@amazon.com
6	amanda_trest@gmail.com
6	amanda_trest@ibm.ca

CONTACT_PHONE

ContactID	Phone
2	403-000-0001
2	403-000-0002
3	403-000-0003
4	403-000-0004
5	403-000-0005
6	403-000-0006
6	403-000-0007

CORRESPONDS_TO

ApplicationID	PositionID	JobPostURL
3	3	www.someurl.com
4	4	www.onevesturl.com
5	5	www.joburl.com
6	8	www.msjoburl.com
7	6	www.nvidjoburl.com
8	7	www.quantjoburl.com

DOCUMENT

DocumentID	ApplicantUsername	DocFileName	DocType	Description
2	fredprokop@gmail.com	Finance Resume - [2].pdf	Resume	Resume tailored for finance roles
3	fredprokop@gmail.com	Fintech Resume - [3].pdf	Resume	Resume tailored for fintech roles
4	fredprokop@gmail.com	Generic Cover Letter - [4].pdf	Cover Letter	Cover Letter to edit for roles
5	prempreetb@gmail.com	Amazon Resume - [5].pdf	Resume	Resume submitted to Amazon
6	prempreetb@gmail.com	Microsoft Resume - [6].pdf	Resume	Resume submitted to Microsoft
7	prempreetb@gmail.com	Prem Generic Cover Letter - [7].pdf	Cover Letter	Cover Letter to edit for roles
8	jayden@gmail.com	Jayden Generic Cover Letter - [8].pdf	Cover Letter	Cover Letter to edit for roles
9	jayden@gmail.com	NVIDIA Resume - [9].pdf	Resume	Resume submitted to NVIDIA
10	jayden@gmail.com	IBM Resume - [10].pdf	Resume	Resume submitted to IBM

INTERVIEW

InterviewID	ApplicantUsername	Stage	Date	ApplicationID
1	fredprokop1@gmail.com	1-Way	2023-11-18	3
2	fredprokop1@gmail.com	Stage 1	2023-11-30	3
3	prempreetb@gmail.com	Stage 1	2023-09-30	5
4	prempreetb@gmail.com	Stage 2	2023-10-15	5
5	prempreetb@gmail.com	Stage 3	2023-11-05	5
6	jayden@gmail.com	Stage 1	2023-04-10	7
7	jayden@gmail.com	Stage 2	2023-05-02	7

JOB

PositionID	CompanyName	PositionName	Description	JobPostFile	PositionType	ApplicationDeadline	Salary
3	RBC	Financial Analyst	Detailed job description...	RBC_FA - [4].pdf	Internship	2023-11-30	52000
4	OneVest	Equity Research Analyst	Detailed job description...	OneVest_EquityRA - [5].pdf	Entry Level	2023-12-10	67000
5	Amazon	Cloud Engineer	Detailed job description...	Amazon_Cloud - [6].pdf	Entry Level	2023-09-30	105000
6	NVIDIA	Software Internship	Detailed job description...	NVIDIA_Internship - [7].pdf	Internship	2023-03-30	65000
7	IBM	Quantum Intern	Detailed job description...	IBM_Quantum - [8].pdf	Internship	2023-11-30	75000
8	Microsoft	Software Engineer	Detailed job description...	MS_SoftEng - [9].pdf	Entry Level	2023-06-25	93000

JOB_QUALIFICATION

PositionID	Qualification
3	Minimum 2 years of post-secondary degree complete
3	3.7+ GPA
3	At least 1 finance-related course complete
3	Prior experience with financial modelling preferred
4	3.6+ GPA
4	Prior experience with equity research
4	Prior experience with pitching investments
4	Ability to work independently
4	Ability to meet tight deadlines
5	Software engineering or computer science post-secondary degree
5	Must be able to demonstrate strong interest in cloud computing
5	Prior experience with cloud services preferred
6	2-3 years of post-secondary degree complete
6	Must have experience working with low-level languages
6	Prior experience working on AI-related projects preferred
7	At least 1 course related to quantum theory must be complete
7	Must be able to demonstrate strong interest in quantum computing
7	Must be returning to studies after internship
8	3.8+ GPA
8	Software engineering or computer science post-secondary degree
8	Must be able to demonstrate prior software projects

JOB_RESPONSIBILITY

PositionID	Responsibility
3	Build and monitor financial models
3	Pitch financial securities to clients
3	Continuously evaluate ongoing portfolio of assets
4	Perform fundamental and technical analysis of public securities
4	Pitch public securities to management
4	Mentor and guide equity research interns
5	Assist in monitoring and expanding cloud services
5	Implement new cloud services initiatives
6	Back-end implementation of new software initiatives
6	Research and suggest new software initiatives
7	Assist in development of IBM's next quantum computer iteration
7	Test functionality of IBM's next quantum computer iteration
7	Test capabilities of IBM's next quantum computer iteration
8	Back-end implementation of new software initiatives
8	Mentor and guide software development interns

KNOWS

Relationship	Notes	LastContactDate	Username	ContactID
Professional	Met at financial networking event	2023-09-10	fredprokop1@gmail.com	2
Classmate	Took several classes together in University; great student	2023-05-25	fredprokop1@gmail.com	3
Classmate	Worked together in several projects in University	2023-08-16	prempreetb@gmail.com	3
Colleague	Met and worked together during past internship at Amazon	2023-04-20	prempreetb@gmail.com	4
Professional	Met at job fair in March	2023-03-26	prempreetb@gmail.com	5
Close Friend	Met through personal hobbies	2023-10-09	jayden@gmail.com	6

MENTIONS

PositionID	InterviewID
3	1
3	2
5	3
5	4
5	5
6	6
6	7
3	6

OFFER

Table was recreated due to spacing issues.

OfferID	Applicant Username	OfferFileName	Response Deadline	Compensation	Start Date	Notes	Position ID
3	fredprokop1@gmail.com	RBC_OFFER-[6].pdf	2024-01-05	52000	2024-05-01	Internship can be extended at end of term in August for a maximum of 4 more months.	3
4	jayden@mail.com	NVIDIA_OFFER-[7].pdf	2023-05-10	65000	2023-05-29	Internship can be extended at end of term in August	6

REFERRAL

ReferralID	ContactID	Date	Notes	ApplicantUsername	PositionID
2	2	2023-09-10	Reference for Financial Analyst internship at RBC	fredprokop1@gmail.com	3
3	6	2023-10-01	Reference for Quantum Intern position at IBM	jayden@gmail.com	7

SUBMIT_WITH

ApplicationID	DocumentID
3	2
4	3
5	5
6	6
8	10
7	9

TRACKS

Notes	DateToApply	Username	PositionID
Interesting but lacks tech aspect; low salary	2023-11-25	prempreetb@gmail.com	4
Could be an interesting experience; may be able to apply my technical skills well	2023-11-23	jayden@gmail.com	3

USER

Table was recreated due to spacing issues.

Username	Password	Password Confirm	Fname	Lname	IsActive	Admin Flag	Permission Level
fredproko p1@gmail .com	91945d00a4f7c04d a3fbe645ca22da37 e933e6f6ff5f9e847 b90f22c71f25106	NULL	Fedor	Prokop chuk	1	0	0
prempreet b@gmail.c	\$2b\$12\$2brPuve9v vTC4EigS7V3NO/	NULL	Prempr eet	Brar	1	0	0

om	0JWOvB0BgeIPA. XpgsJ8VAxh843M vW						
jayden@g mail.com	\$2b\$12\$Sg/RXapF wbUwGKjeYkuioe EFNSoCebcKxIyq YTgf4yomuChHj Pve	NULL	Jayden	Yeo	1	0	0

WORKS_AT

Role	ContactID	CompanyName
Managing Director	2	RBC
Board Member	2	OneVest
Senior Cloud Engineer	5	Amazon
Quantum Specialist	6	IBM

References

¹ Kim, J., Oh, J., & Rajaguru, V. (2022). Job-Seeking Anxiety and Job Preparation Behavior of Undergraduate Students. *Healthcare (Basel, Switzerland)*, 10(2), 288.

<https://doi.org/10.3390/healthcare10020288>

² Node.js. (n.d.). <https://nodejs.org/en>

³ Express.js. (n.d.). <https://expressjs.com/>

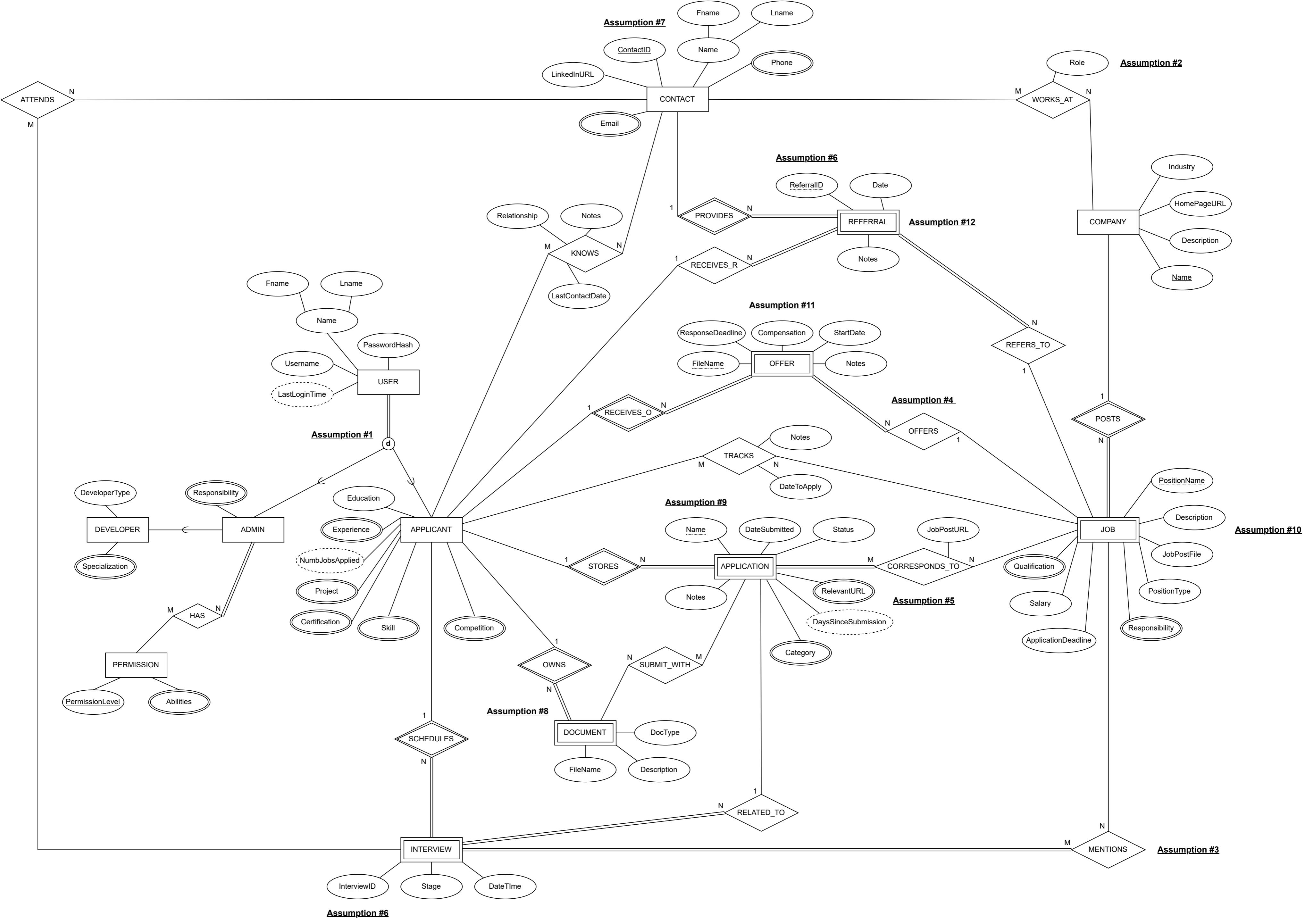
⁴ React.js. (n.d.). <https://react.dev/>

⁵ Sequelize. (n.d.). <https://sequelize.org/>

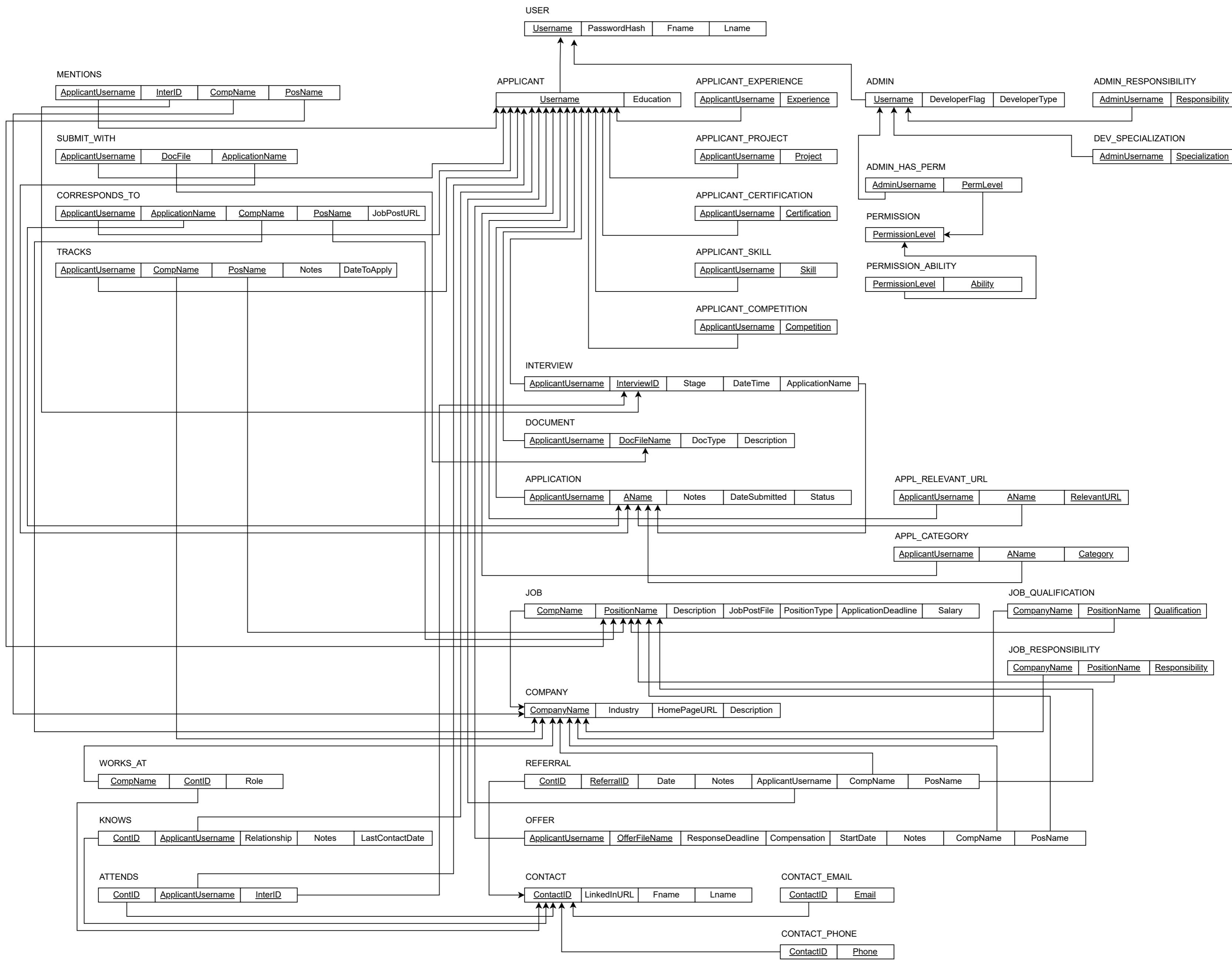
Appendices

Appendices begin on the next page.

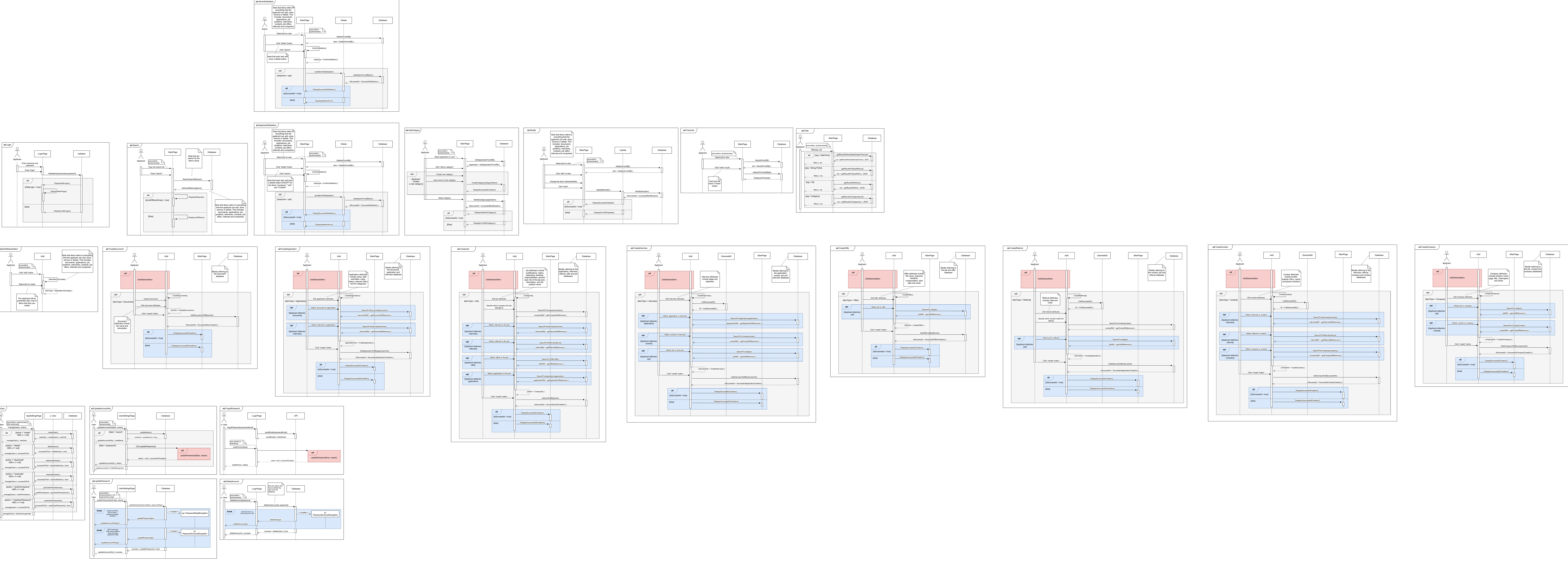
Appendix A

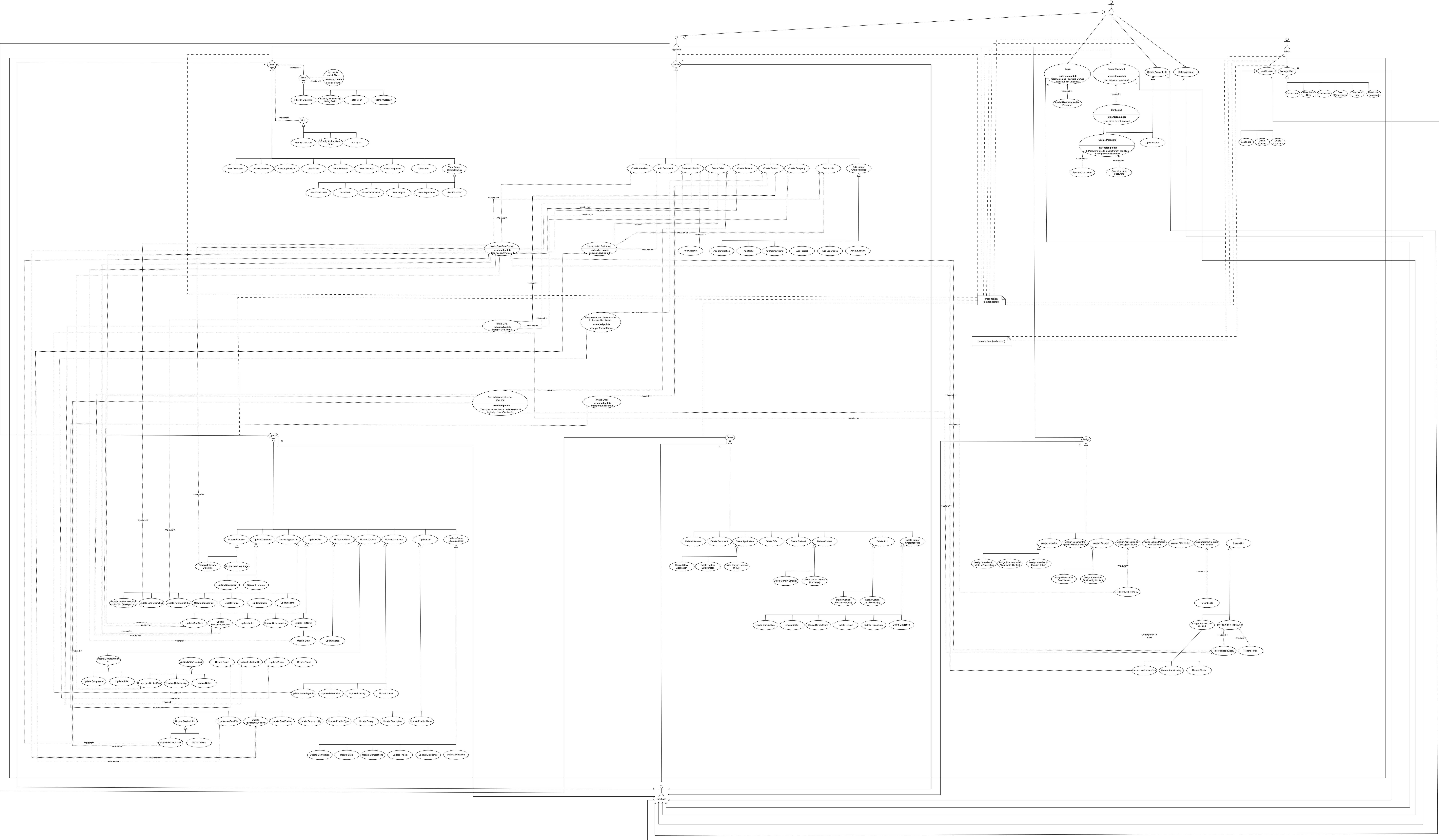


Appendix B



Appendix C





Use Case Diagram

Below is a small explanation of some of the modelling decisions in the use case diagram. Note that we only modelled the main functions/features for brevity (ie. some miscellaneous functionality may not be modelled).

1. There are 4 main actions an applicant can do: View, Create, Update, Delete. In addition, they can also assign some of their created entities to other parts of the system (such as assigning an offer to a job posting).
 - a. When viewing components of the system, users can filter, sort, or view. Note that not all filter/sorting functionality will be available for each part of the system. For example, a user cannot “Filter by Category” for non-categorized items (such as referrals).
 - b. Users do not have free reign to update certain parts of the system. For example, on the “Update Interview” use case, the user can only update the Interview Stage or Interview DateTime; they cannot update the ID. We used our discretion regarding what values should be updatable in the diagram.
 - c. When deleting, certain use cases provide the user the option to delete specific parts of a component. For example, a user can “Delete Whole Contact” or “Delete Certain Email(s).” We used our discretion regarding what values should be deletable (ie. if a contact no longer uses an email, the applicant may wish to delete their inactive email but still keep them in the system). This was typically for multi-valued attributes.
2. There are no multiplicities listed from the applicant to the use cases (which means that the multiplicity is implicitly 1). This is because while an applicant, for example, can create multiple interviews, they have to create each interview one-by-one (ie. they cannot create multiple interviews simultaneously). In other words, the multiplicity is for use case participation, NOT the Enhanced-Entity-Relationship diagram. An applicant can only participate in one instance of each use case at a given moment in time.
 - a. There **are** multiplicity listed from the database to each use case; this is because the database is dealing with multiple view, create, update, delete, and assign operations simultaneously.

3. The admin has additional functionality not shown in our EERD or our Relational Model. This is because we felt the admin would otherwise not be too differentiable from a regular user.
 - a. Additional functionality includes creating, deactivating, reactivating, and deleting users, giving permissions to certain user accounts (to “promote” them into admins), and resetting user passwords.
 - b. Note that jobs, contacts are shared across all users to prevent duplication in the database. Consequently, only the admin is allowed to delete jobs, contacts, and companies (not users); this is because we do not want malicious users deleting jobs, contacts, or companies from the database. Users can however, update the job’s, contact’s, or company’s attributes; this includes deleting job qualifications and responsibilities, and contact email(s) and phone number(s).
4. The user has additional functionality not shown in our EERD or our Relational Model (such as logging in, forgetting password, updating account info, and deleting account).
5. There is an arrow going from the “Email sent” use case to the “Update Password” use case; this is because after a user sends their “Forgot Password” link, the system will then allow them to update their password using that link (bypassing the preconditions of authentication).
6. In our EERD diagram, we modelled applicants and admins as being disjoint (separate); however, we have now decided to allow admins to perform all actions allowed by applicants (which is why you see admin inheriting from applicant in the Use Case diagram).

Sequence Diagram

Below is a small explanation of some of the modelling decisions in the sequence diagram. Note that we only modelled the main functions/features for brevity (ie. some miscellaneous functionality or rare exceptions may not be modelled).

1. There are differences across some diagrams where boolean status codes are obtained from the database (in the actual system this will be from the API). For example, in the “ManageUser” sequence diagram, boolean status codes are named SuccessOrFail. In the

“UpdateAccountInfo” sequence diagram, boolean status codes are named Success; this is because in the “UpdateAccountInfo” diagram, the case of failure is handled in an exception. In other words, where there are exceptions, the regular sequence is only labelled with success.

2. Note that you may see slight differences in syntax across sequence diagrams; this is because multiple group members worked on the diagrams (each using their own style). However, when looking at a sequence diagram in isolation, it is still correct (ie. even though group members used different forms of syntax, both forms still follow UML standards).