Amit Peled, Daniel Piro, Shahar Alon, Nofar Rozenberg

BGU / MolOpt

MolOptimizer - Final Project in Software Engineering

The Final Report & User Manual Document

Contents

[Introduction 3](#_Toc131242328)

[User Manual 4](#_Toc131242329)

[2.1 Jump Start / Getting Started With MolOpt 4](#_Toc131242330)

[2.1.1 System Requirements 4](#_Toc131242331)

[2.1.2 Installation 4](#_Toc131242332)

[2.1.3 Quick Start Guide 4](#_Toc131242333)

[2.2 Quick Help 5](#_Toc131242334)

[2.3 User Manual 6](#_Toc131242335)

[2.4 Reference 16](#_Toc131242336)

[2.4.1 Technical Specifications 16](#_Toc131242337)

[2.4.2 Acknowledgments 16](#_Toc131242338)

[2.4.3 Contact Information 16](#_Toc131242339)

[Maintenance Guide 17](#_Toc131242340)

[**Note:** Before start reading this chapter, please pay attention to all of the comment and documentation written in the code itself. 17](#_Toc131242341)

[3.1 Backend Maintenance 17](#_Toc131242342)

[3.2 Frontend Maintenance 18](#_Toc131242343)

[3.3 Usability Maintenance 18](#_Toc131242344)

**Abstract**

This document is intended for Users/Reviewers/Maintainers/Admins.

The document describes any needed data about the MolOpt web-application:

* What is MolOpt?
* How the UI looks like.
* How to use the web-application.
* How to maintain the web-application.
* How to be able to add MolOpt new features in the future.

Chapter 1

# Introduction

**Note:** If you are already familiar with MolOptimizer, feel free to skip this chapter and move directly to Chapter 2.

**What is MolOpt?**

MolOpt is a global web-application which was Developed at Akabayov Research Laboratories, at the department of Chemistry in Ben Gurion University of the Negev, Under the supervision of Dr. Barak Akabayov.

The main purpose of MolOpt and the main reason why it was developed, is to be used by researchers working in the field of small fragment-based inhibitors and is intended to be helpful  
 in optimization of ligand databases.

**How is in-charge on MolOpt?**

* Developers: Amit Peled, Daniel Piro, Shahar Alon & Nofar Rozenberg.
* Supervisors & Managers: Dr. Barak Akabayov & Samuel Viswas.

Chapter 2

# User Manual

## 2.1 Jump Start / Getting Started With MolOpt

This section provides a brief overview of how to quickly get started using MolOpt web-application.

For more detailed instructions, please refer to the rest of the manual.

### System Requirements

Before getting started, ensure that your system meets the following requirements:

* Requirement 1: Make sure you got a strong Wi-Fi connection on your device.
* Requirement 2: Make sure your device has enough storage space (that only of you will want to save the results of the algorithm runs).

### Installation

MolOpt is a global application that can be accessed from anywhere at any time.

Therefore, there is no need in installing anything, you can jump right in and getting started using MolOpt.

### Quick Start Guide

Follow these steps to start using MolOpt:

* Step 1: Go to the website URL - <http://localhost:3000/> (TBD – Change to the real URL)
* Step 2: Register to the application.

If you’re already have a user in the system, feel free to skip this step.

* Step 3: Login with you’re registered user to the application (using email & password).
* Step 4: That’s it! Now you can -
  + Run the algorithms (with the suitable input files).
  + View the runs/tasks queue and current running.
  + View all the information about MolOpt and its algorithms and processes.
  + Contact the app’s people in charge (admins and maintainers).

**For more info about what you can do in the application – See section 1.3.**

## 2.2 Quick Help

This section provides brief instructions on common tasks, questions and issues using MolOpt in a Frequently Answers & Questions (FAQ) format.

This section provides answers and brief instructions to common tasks and questions users may have when using MolOpt.

**Note:** If you have a question that is not answered here, please refer to the rest of the manual or contact MolOpt support (See Q3 below for more details).

**Q1: What are the system requirements for MolOpt?**

**Answer:** The system requirements for MolOpt are:

* Requirement 1: Make sure you got a strong Wi-Fi connection on your device.
* Requirement 2: Make sure your device has enough storage space (that only of you will want to save the results of the algorithm runs).
* No special installation is needed!

**Q2: Can I use MolOpt on multiple devices?**

**Answer:** Yes, you can use MolOpt on multiple devices if you are using a web-browser and have user registered in the system.

**Q3: How do I contact support for MolOpt?**

**Answer:** You can contact MolOpt support by writing a message in the “Report an issue” in the app.

This message is automatically sent to all the admins and maintainers of MolOpt which will be taking care of any issue as fast as possible.

**Q4: What can I do if I forgot my user credentials?**

**Answer:** In this case there are 2 options:

1. If you forgot you’re email – feel free to register a new user, but make sure that the email you’re using for the new user, is a used and known email (otherwise, you’ll not be able to benefit of what the app has to offer).
2. If you forgot you’re password – in the Login page there is an option – “Forgot my password”.

This will make you able to get your password or reset/generate a new password for your user (using the same email).

**Q5: How do I run an algorithm in the application?**

**Answer:** You can see a very detailed guide and instructions on how to run any algorithm in the system both in the Algorithm page in the app or here in section 1.3.

**Q6: Where can I view my algorithm runs?**

**Answer:** You can view all your runs in the Tasks page in the app.

**Q7: How can I know my algorithm run status?**

**Answer:** First, while you start running the algorithm, you’ll be able to see a notification notifying that the algorithm you chose has scheduled to run in the system.

In addition, you can go at any time to the Tasks page and search for your run and see there the status of your run.

## 2.3 User Manual

This section provides a “textbook” for how to use the app.

**How To Go To The App Itself**

Go to the website URL - <http://localhost:3000/> (TBD – Change to the real URL)

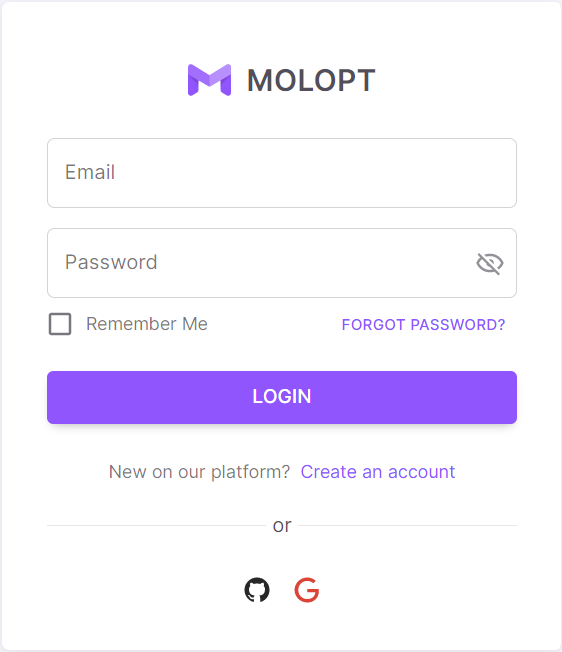
**Special Requirements And Installation**

* Requirement 1: Make sure you got a strong Wi-Fi connection on your device.
* Requirement 2: Make sure your device has enough storage space (that only of you will want to save the results of the algorithm runs).
* No special installation is needed!

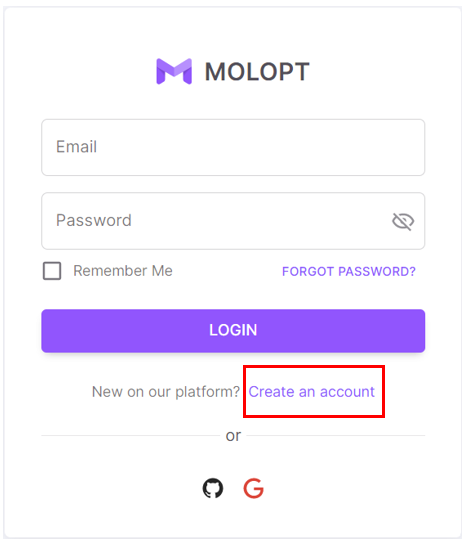
**Login**

The Login page is the first page you’ll see when you open the app (you’ll also be able to see it after registration, resetting the password using “Forgot Password” or logout).

When you open the Login page, you’ll see the following screen:



In order to login to the system, all you need to do is insert your valid email and matching password and click on the login button.



**Register**

If you want to register a new user to the system, you first need to go to the registration page.

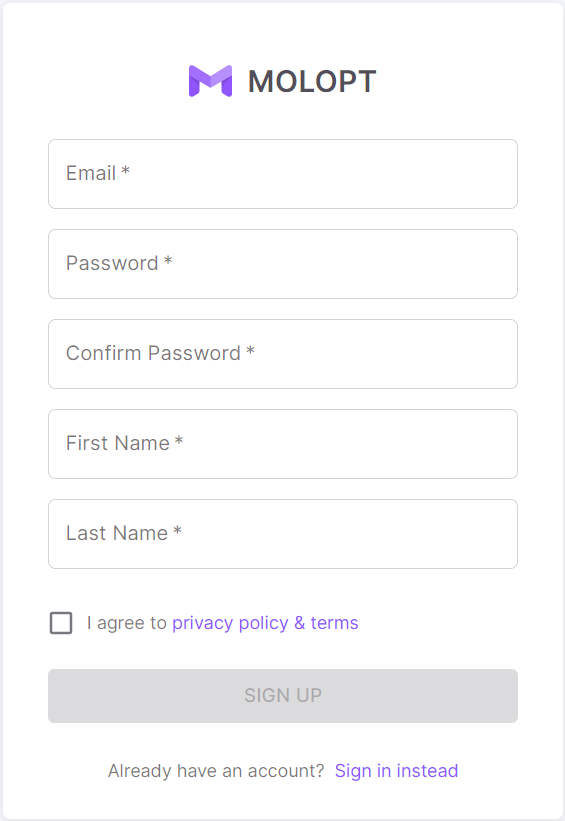
To do that you’ll need to click on the link “Create an account” located in the Login page.

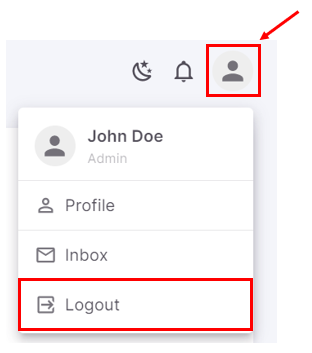
(See the image on the right)

After going to the registration page, you’ll need to insert all the valid required details as described in the register page – If some data is not valid, you’ll be notified immediately by a red comment next to the required filed text-box.

When this is done, all left to do is to agree to the website’s privacy policy and terms and click on the “Sign up” button.

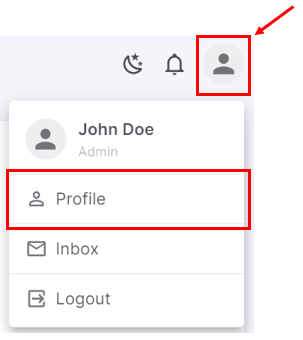
If the registration process went well, you’ll be moved back to the Login page and will be able to login to the system with the new user that you’ve just registered.  
(You can see how the Register page looks in the image down below)



**Logout**

Logging out from the application is possible only if you’re already logged in.

If you would like to logout, all you need to do is to click on the User’s picture icon on the top-right of the screen, and then click on “Logout” in the dropdown list that will be opened.

**Edit Profile/User Details**

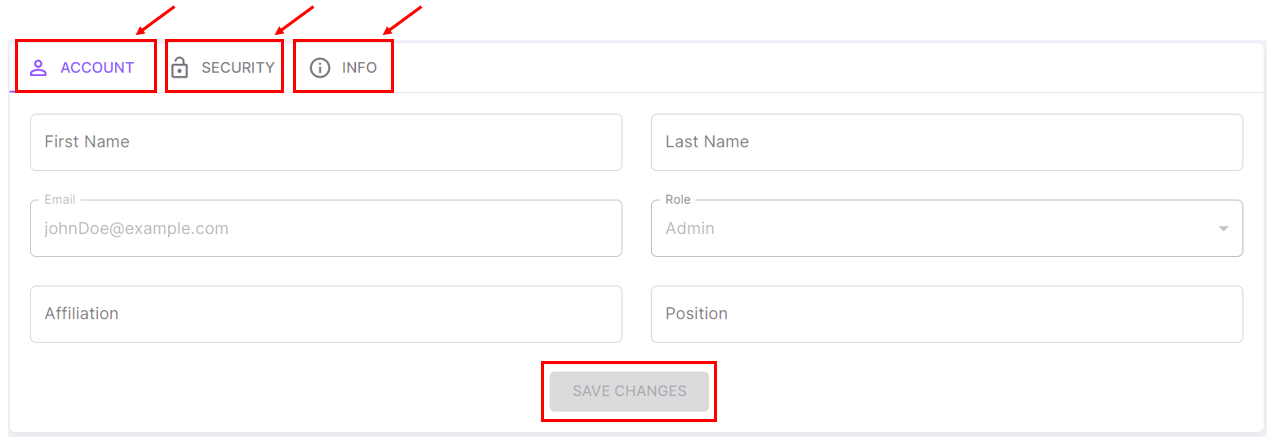
User can change his information details at any time.

If you would like to do that, you need to do is to click on the User’s picture icon on the top-right of the screen, then click on “Profile” in the dropdown list that will be opened.

After that a screen with 3 tabs will be opened (See the image down below) – each tab is relate to different type of information details:

* Account – relate to first name, last name, email, affiliation, and position.
* Security – relate to passwords.
* Info – relate to birth date (age), phone number, country, languages, and gender.

After choosing the required tab, just fill the new data (validation check here is the same as appears in the registration form) and click on “Save Changes” when you’re ready to change your details.

****

**Run An Algorithm**

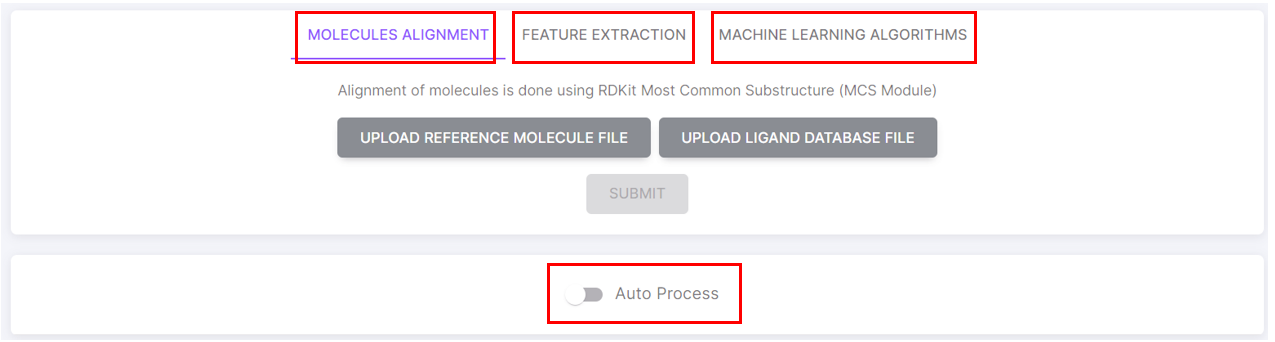
This action is the heart of the application (and probably the most complicated one in the system).

Each algorithm expects different inputs and has a different purpose.

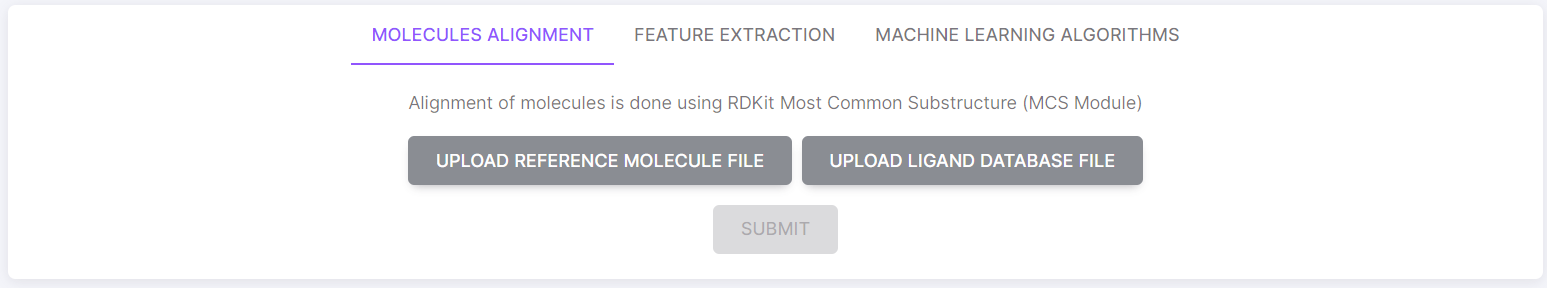
In order to understand exactly what each algorithm does and what each one of them needs to be able to run, go to the Algorithms page in the application (You can see how to go to that page in the section “Where to find all the detailed information about the algorithms/input-output files/website’s processes” in the next page of this document).

So, in order to run an algorithm, you’ll need to go to the Dashboard page, which is the homepage of the application.

Then, you’ll need to choose which algorithm you want to run (by clicking on the wanted algorithm) :



**Alignment**



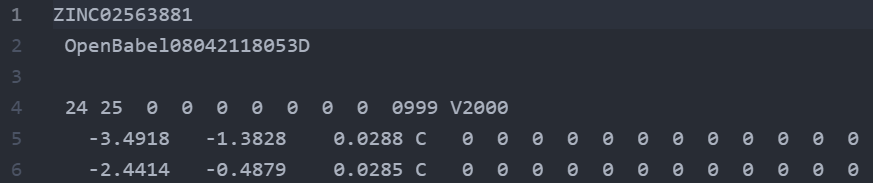
This algorithm aligns the molecule’s module that he gets according to the reference module that he gets.

Both reference and new molecule’s modules are required in “.sdf” files.

“.sdf” file format should be as follows:

* First 2 rows will contain the title and name of the file.
* The third line will be blank to separate the title from the content of the file.
* The fourth line will contain the headers – separated by tab (\t).
* From the fifth line and until the end of the file, each line will contain the values for each header mentioned in line 4 – separated by tab (\t) as well.
* For ending the file/table in the file, just write these two lines:
  + M END
  + $$$$

For Example:



**Note:** for files that contains multiple tables as explained above, all you need to do is just add another blank line right after the “$$$$” and repeat the steps above.

The output of this run will be a “.mol2” file – the aligned molecule file.

**Feature Extraction**

Graphical user interface, text, application

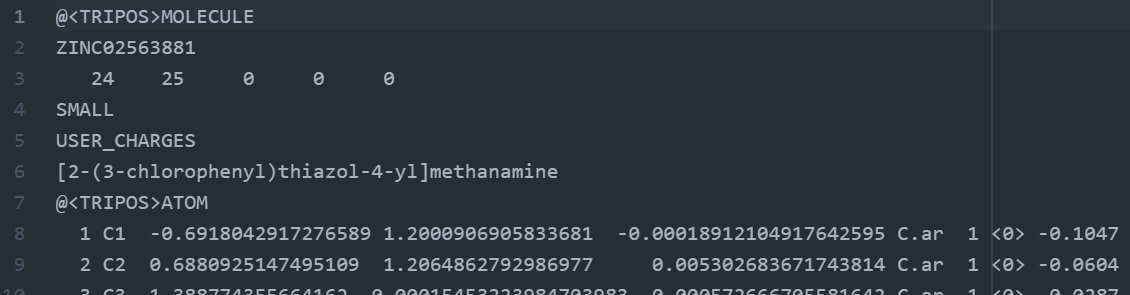
Description automatically generated

This algorithm gets the molecule in a “.mol2” file (the aligned molecule file) and gives details about his characteristic’s values according to the type of feature extraction that has been chosen.

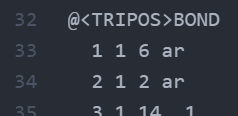
(These values are critical for evaluating the docking level of the molecule).

“.mol2” file format should be as follows:

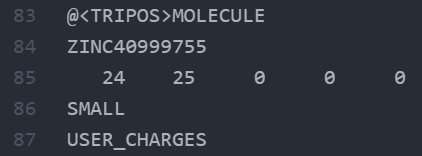
* First section of the file should look like:



* Second section of the file should look like:



* Final part of the molecule description should look like:



**Note:** for files that contains multiple molecules as explained above, all you need to do is just add another blank line right after the “USER\_CHANGES” and repeat the steps above.

This algorithm also require that the user will pick at least one of the feature extraction options:   
RDKit / Mordred.

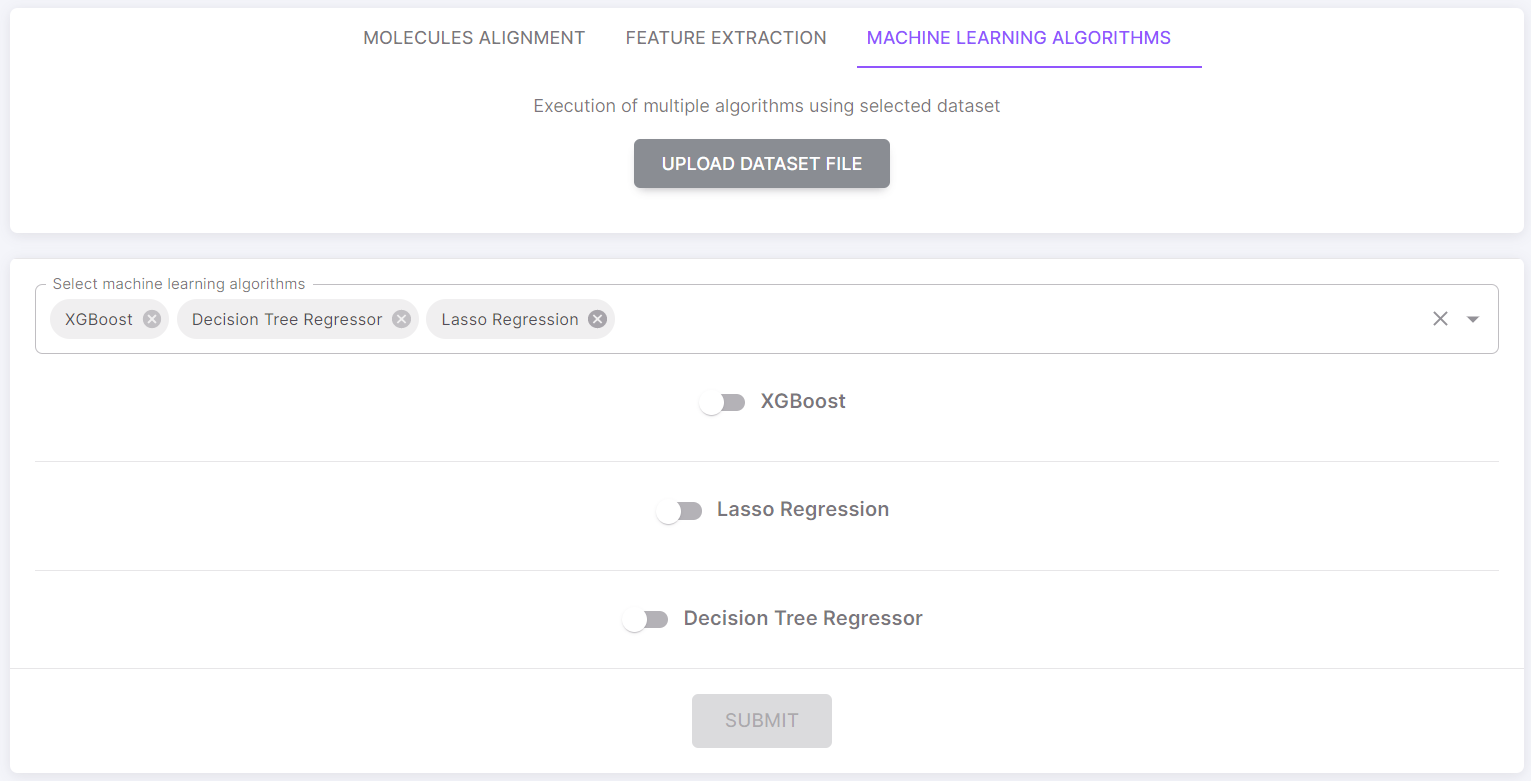
(Otherwise, there is no way to know which characteristics values we need to get out of the algorithm results).

The output of this run will be an Excel file – the characteristics values for the docking stage.

**Machine Learning Algorithm**

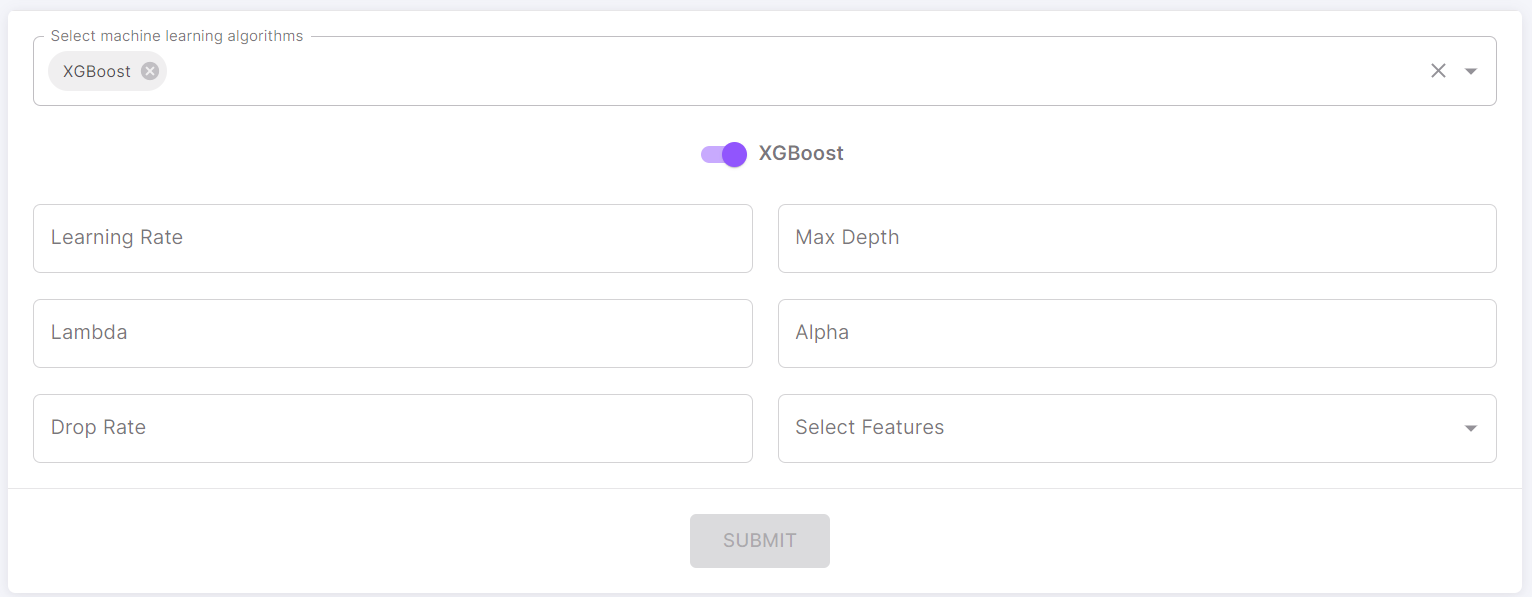
This type of algorithms has 2 running-modes in the system:

* Auto = runs the algorithms with defaults input values.
* Manual = runs with the user’s input values.



This algorithm gets the molecule in an Excel file with the molecule characteristics values and run a ML algorithm with a values defined in advance.

Here is an example for how it looks to run one of the in the system:



The Excel file format should be as follows:

* First line should contain the headers (each cell will have the name of different character of the molecule).
* The rest of the lines will contains the values for each character.

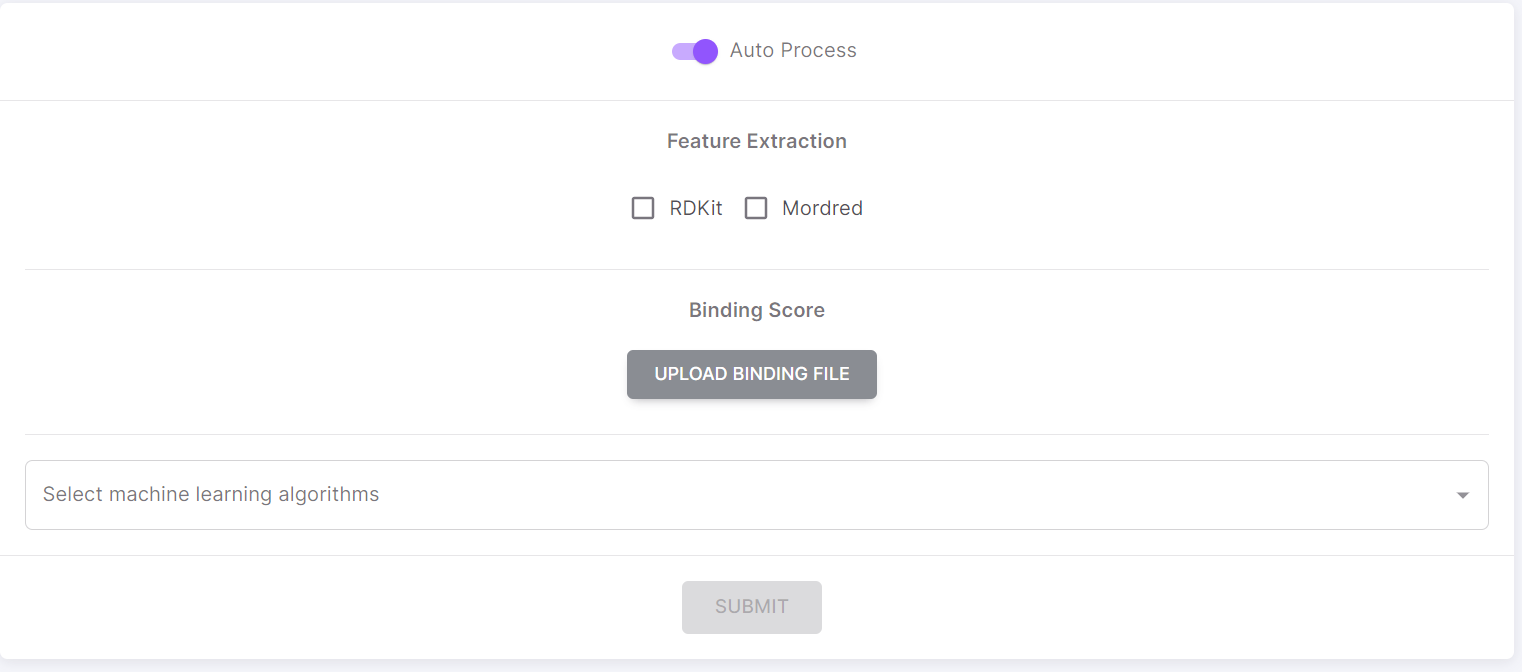
**Important Note: You’ll have to make sure that there is no blank or null cell in the Excel file!**

This algorithm also require that the user will pick at least one of the feature extraction options:   
RDKit / Mordred.

(Otherwise, there is no way to know which characteristics values we need to get out of the algorithm results).

The output of this run will be an Excel file – the docking level and fit level to another molecule (Each excel output file, related to a different ML algorithm selected).

**Auto Process**



This type of run is a mix of running all of the algorithms one after the other:

This run only need an .sdf binding score file, required feature extraction run types and required machine learning run types (with their required input values).

The output of this run will be an Excel file – the docking level and fit level to another molecule (Same as in the ML algorithm runs).

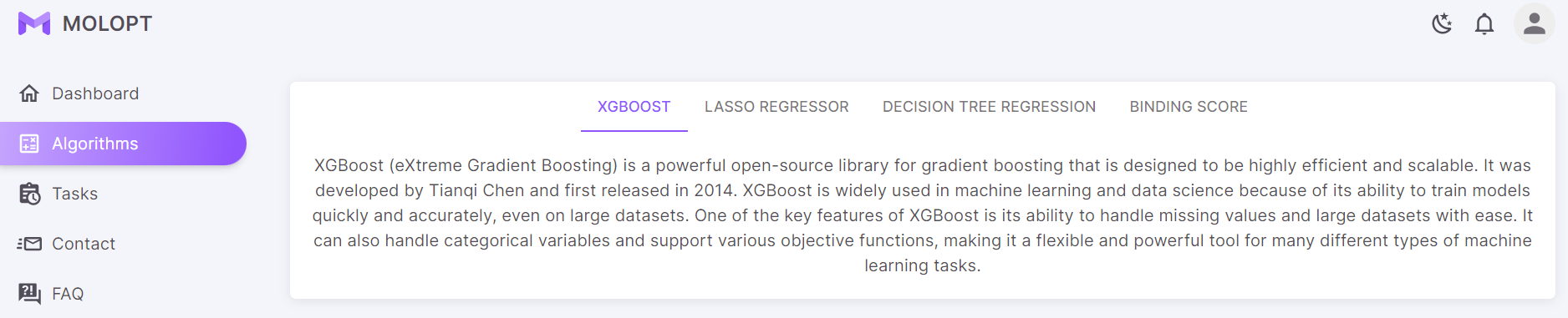
Graphical user interface, application

Description automatically generated**Where to find all the detailed information about the algorithms/input-output files/website’s processes**

If you want to see all the information related to the algorithms in the system, you’ll need to go to the Algorithms page.

You can find the link to that page in the menu on the top-left side of the screen (You can see this in the image on the right).

When clicking on “Algorithms”, The page down below will be opened and you’ll be able to read anything you need to know about any algorithm in the system.

(The image contain only a tiny part of this page)

Graphical user interface, application

Description automatically generated**View Tasks Queue / View Tasks Status**

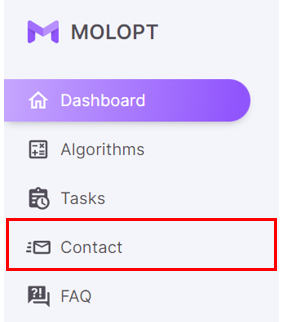
If you want to see all the tasks that you ran in the past, currently running or will be run, you’ll need to go to the Tasks page.

You can find the link to that page in the menu on the top-left side of the screen (You can see this in the image on the right).

After getting in the Tasks page, you’ll be able to see all the algorithms related to you and their details and status.

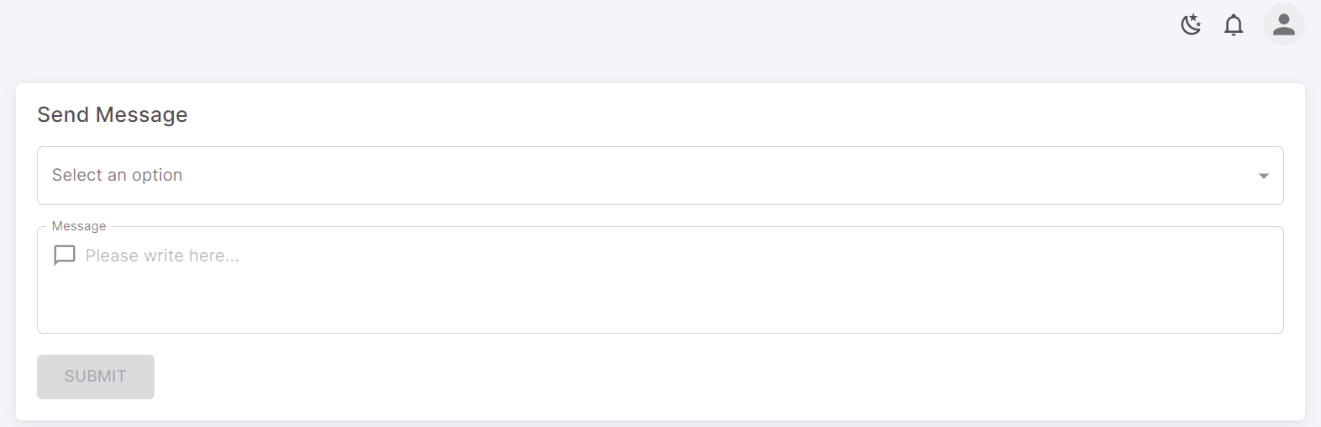
Graphical user interface, application

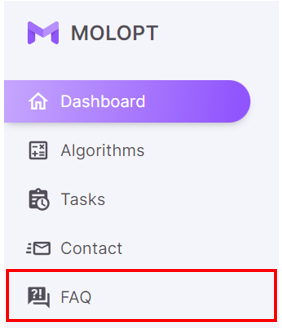
Description automatically generated(See the image below to see how this page looks like)

**Contact For Support / Contact The People In Charge**

If you want to contact the people in charge or the maintainers of the website (for advice, report an issue, random questions, etc.), you’ll need to go to the Contact page.

You can find the link to that page in the menu on the top-left side of the screen (You can see this in the image on the right).

When clicking on “Contact”, The page down below will be opened. In this page, you can select the reason you write this message (advice, report an issue, random questions, etc.), and write the content of your message. After that, by clicking on Submit, this message will be sent to the relevant people (Admin and/or Maintainers)

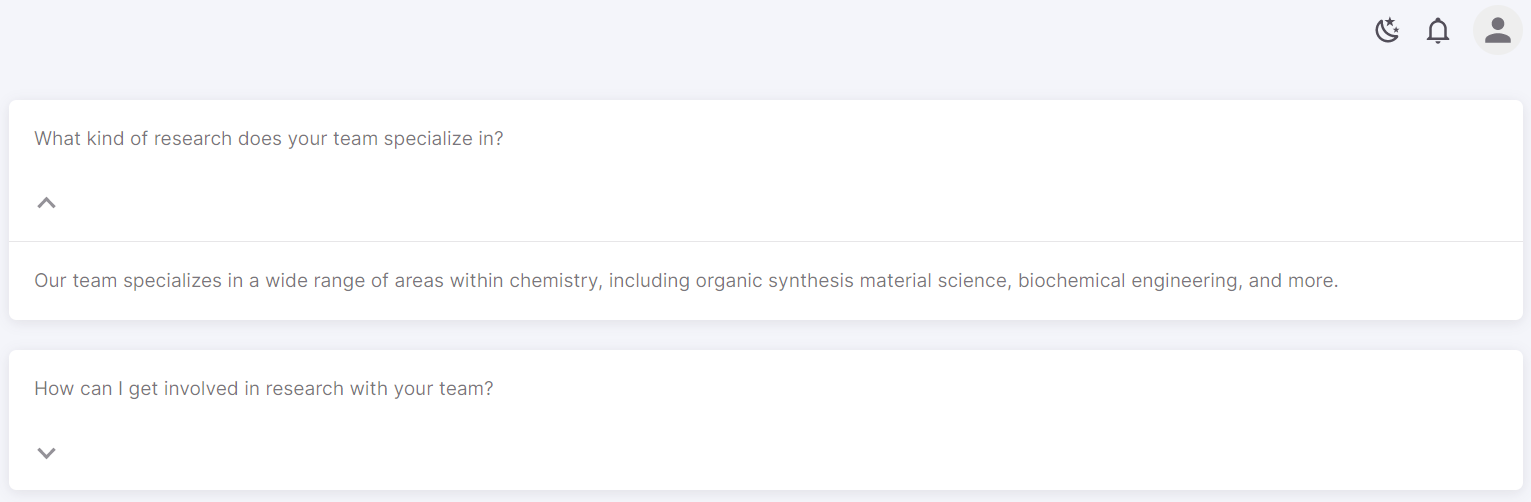
**FAQ**

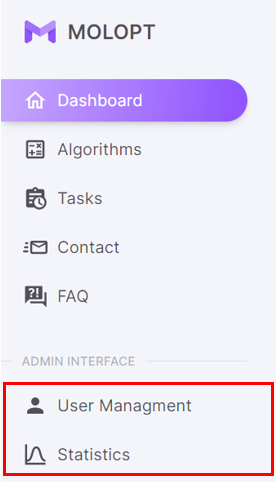
If you want to see all FAQ, you’ll need to go to the FAQ page.

You can find the link to that page in the menu on the top-left side of the screen (You can see this in the image on the right).

When clicking on “FAQ”, The page down below will be opened and you’ll be able to read all the Q&A that have been asked the most.

(The image contain only a tiny part of this page)

****

**Admin Capabilities In The System**

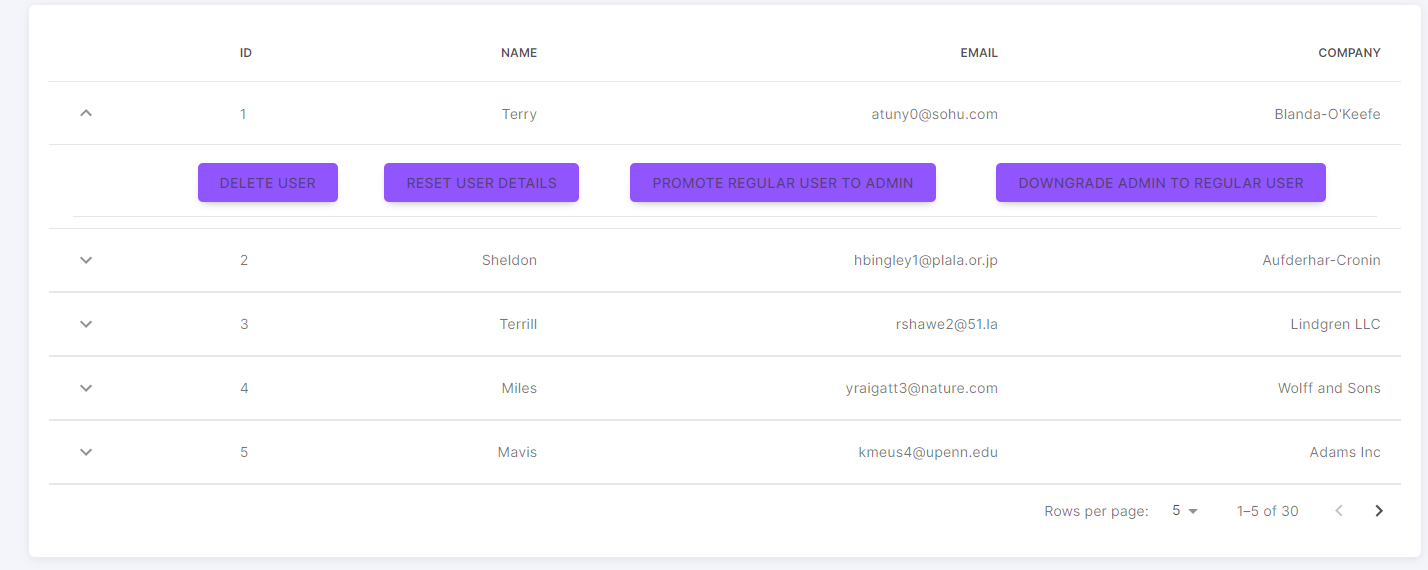
For Admins in the system the menu on the top-left side of the screen has 2 more options: **User Management & Statistics**

**User Management**

This page contains anything the admin needs in order to manage any user in the system:

* Delete User.
* Reset User Details.
* Promote User To Admin.
* Downgrade Admin To Regular User.

When clicking on “User Management”, the admin will be able to see a list of all the users in the system, and each of them will contain the buttons of all the possible actions the admin can do related to them.

****

**Statistics**

This page contains all the details admin need to know about the system/application itself:

* How many users visited/registered/logged in.
* Graphs of user’s runs.
* System status.

All of that information will be presented to the admin when clicking on “Statistics”.

## 2.4 Reference

### 2.4.1 Technical Specifications

* MolOpt is a web-application that has 2 main sections – Backend and Frontend.
  + The backend side is written in Python using Django.
  + The backend also has a database management – using SQLite for saving and storing all the data.
  + The frontend side is written in React JS using also Material UI (MUI).
* MolOpt is running on the BGU servers and using its security and storage abilities.

### 2.4.2 Acknowledgments

Special credits are for 2 main supporters of this project:

1. Ben Gurion University Of The Negev (BGU).
2. BGU’s chemistry department and Akabayov Research Laboratories.

### 2.4.3 Contact Information

* Akabayov Research Laboratories – [Email]
* [Admin Emails…]
* [Maintainers Emails…]

(TBD – Write the real emails of the people In charge)

Chapter 3

# Maintenance Guide

## **Note:** Before start reading this chapter, please pay attention to all of the comment and documentation written in the code itself.

## 3.1 Backend Maintenance

All of the content of the backend is located in the project in a folder named “Server”.

The backend is written in Django and divided to many folders, each one has different purpose:

* Algos folder

This folder contains all the content of the algorithms.

If you would like to change the algorithm itself – there you’ll be able to find what you’re looking for.

* API folder

This folder contains all the api calls the backend has to offer.

If you wish to change the way the frontend side calls an api or add a new api call – there you’ll be able to find what you’re looking for.

* Runs folder

This folder contains all the content of any actions/logics related to algorithm runs in the system.

If you would like to add or edit feature in the system related to runs in the system – there you’ll be able to find what you’re looking for.

* Server folder

This folder contains all the Django settings.

In our opinion (Developers) this folder is better to leave untouched.

* Users folder

This folder contains all the content of any actions/logics related to users in the system (including admins).

If you would like to add or edit feature in the system related to users in the system – there you’ll be able to find what you’re looking for.

* Venv folder

This folder contains all the settings of the Python virtual environment.

In our opinion (Developers) this folder is better to leave untouched.

## 3.2 Frontend Maintenance

All of the content of the frontend is located in the project in a folder named “Client”.

The backend is written in Django and divided to many folders, each one has different purpose:

* .next, node modules, public, configs, context, hooks, lib, utils folders

All of these folders contains all the settings of React JS application.

In our opinion (Developers) this folder is better to leave untouched.

* Components folder

This folder contains all the “little” components that appears in the website.

If you would like to add or edit a component (one that will be used more than once) in the system – there you can put your new component there and import it from any page you want.

* Pages folder

This folder contains all the pages that are in the website.

If you would like to add or edit a page in the system – there you can put your new page.

* Styles folder

This folder contains only one “.css” file.

This file includes all the style setting related to components in the system.

We recommend not to delete any of these style’s setting but feel free to add as many new styles as you want there.

## 3.3 Usability Maintenance

**For maintenance purposes and the checking new issues that might come up related to users / user’s permissions / specific users in the system, we created 2 “Fake” users that will be available to use for maintenance / testing / debugging purposes only!**

**Fake/Reference Users:**

* **Admin User**
  + Username/Email – [Admin@gmail.com](mailto:AdminDemo@gmail.com)
  + Password – Admin123!
* **Registered User**
  + Username/Email – [User@gmail.com](mailto:User@gmail.com)
  + Password – User123!

You can also view all the user in the system (including these user) on:

<http://localhost:8000/api/users/>

(TBD – Change to the real URL)