upLift Documentation

Kayla Cinnamon, Caitlin Connerney, Ezra Dowd, Stefan Steenstrup, Crystal Yee

Table of Contents

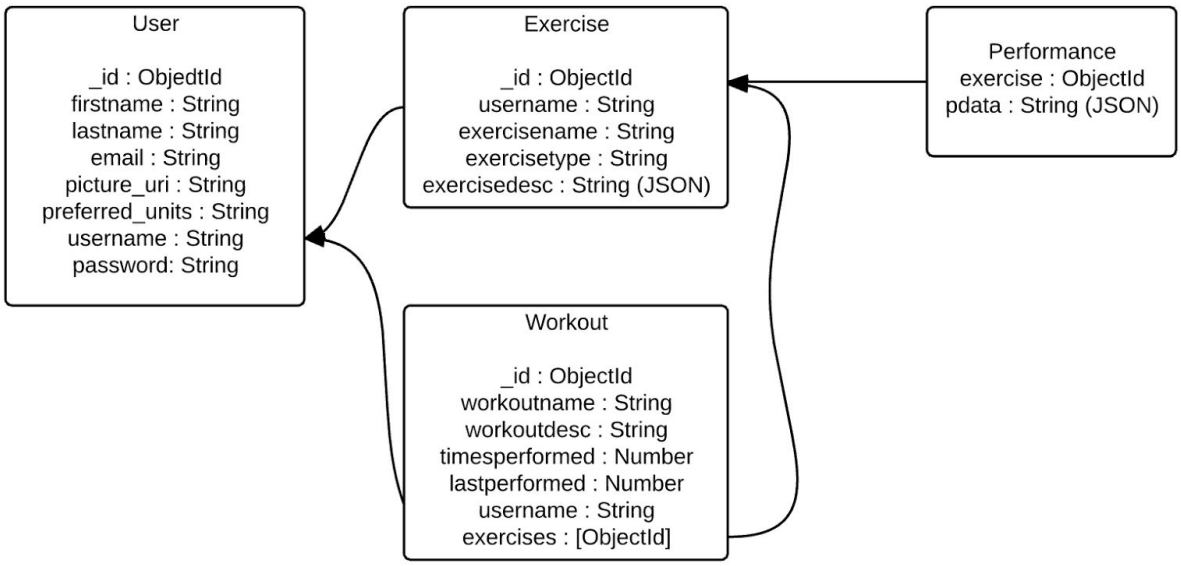
Introduction

Our term project is a workout application. The purpose behind the application is to offer customizable and trackable workouts to the advanced lifting/workout community. Users have the option to create, edit, delete, view progress on, and perform workouts. In addition, there is a dashboard for at-a-glance workout history information and a preferences page to set your unit preference, edit your information, and add a profile picture. Our application uses HTML, CSS, Javascript, AngularJS, and Express to develop integral features. Each feature is explored in further detail throughout this guide.

Installation

Installing upLift is a relatively simple process. Extract the zipped up code to a folder. Navigate into that folder. Open a terminal session. Run “npm install”. Ensure that a MongoDB session is running the background. Ensure that this session has no authentication. In your terminal session run “node server”. Open a web browser. Navigate to “http://localhost:1337/”. Enjoy the application! The live site can be found at this address: ezra.gg:1337. Also,this is the link to the Git repository: https://github.com/deApollo/TeamSeven. We also used GitHub as our bug tracker.

Data Model

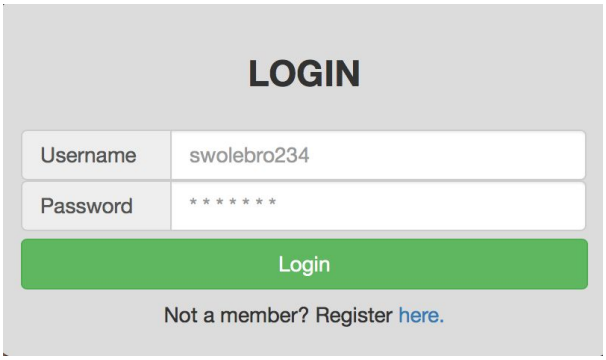


The above data model illustrates and defines the various mongoose schema objects for our application. The user object contains various pieces of information about the user including their

username, hashed and salted password, and URI to their picture. The workout object contains the workout's name, times it has been performed, date last performed, username of the user who created it, and an array of ObjectIds corresponding to the exercises it contains. The exercise object contains the exercise data encoded as JSON as well as other type information used for decoding. The performance object contains a reference to the exercise it is for and the actual performance data encoded as JSON.

Login Page

When the user loads our application, they are directed to the login page. Here they have the option to either log in or create an account if they do not have one.



Existing User

If a user has already created an account on our website, they can use the main login pictured below. All fields are required. If the password or email is incorrect, the user is notified. An HTML form is used with input fields to take the username and password. With the button click, an Angular function is called and the input is verified. Upon confirmation, the user is redirected to their dashboard.

New User

If a user does not yet exist, they can create an account using the register page. A user is redirected here upon clicking the 'register here' link on the login page. All fields are required. Once they click the Register button, and are not met with any issues, they will continue to the dashboard. The Register button, similar to the Login button, calls an AngularJS function. Though in this instance, rather than verifying the user, they are added to a database.

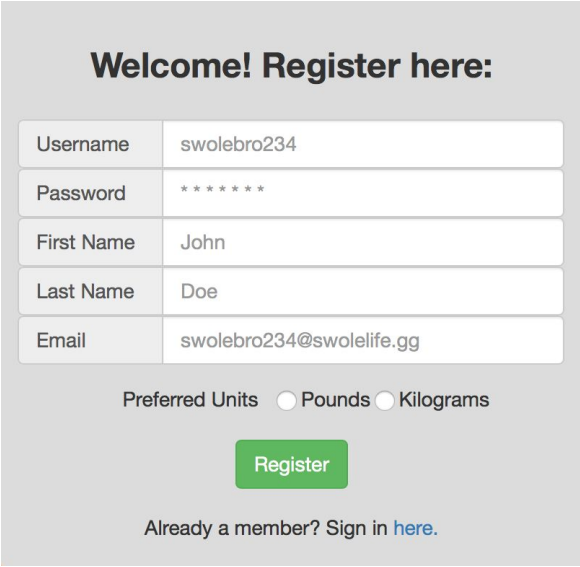
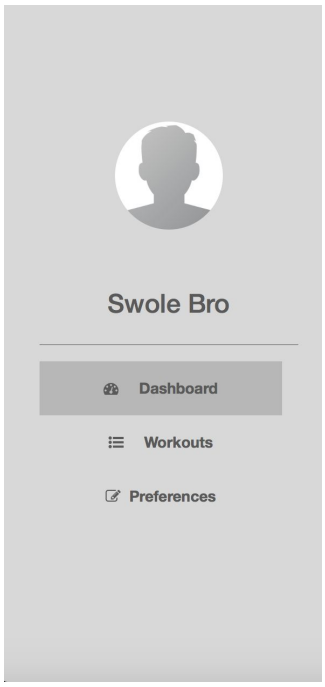


Photo Uploading

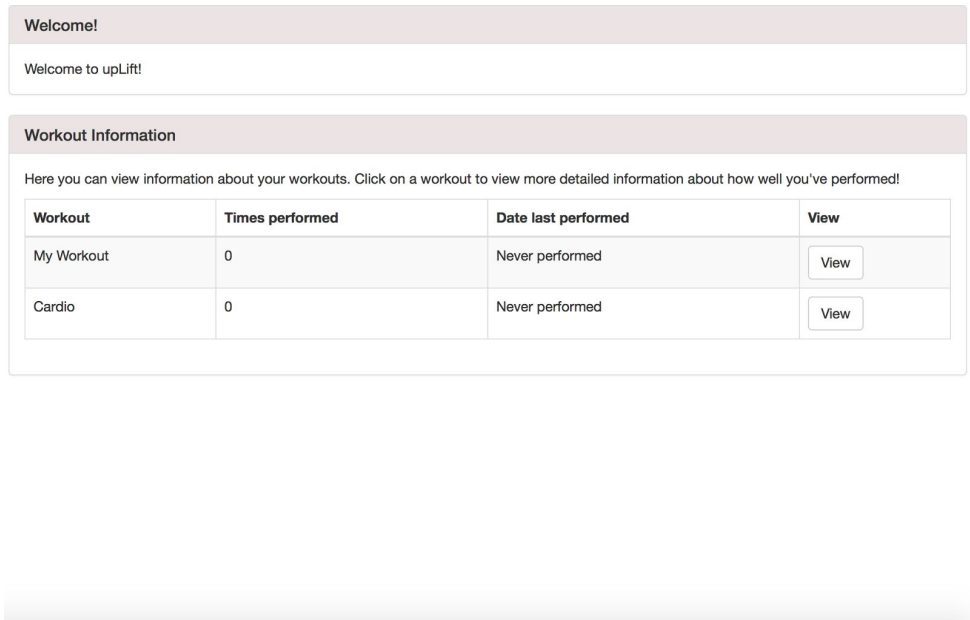
To upload a new profile photo, the user must follow a series of steps. First, the user must log into their account on the application. Once logged in, the user should navigate to the “user preferences” page using either the sidebar on the left or the navigation bar on the top, depending on whether they are on mobile or not. Once they are at the user preferences page, the user should click on the “Choose File” button next to the “Photo Upload” label. Once the user has selected a file, they should press the “Submit” button below to start the uploading process. Photo uploading is handled through a series of middleware functions. First, the user’s login session is validated. Then, if they have an old picture already uploaded, it is deleted. Then, the upload multer is used to actually receive the file and store it on disk. Lastly, a method in the data controller is called to update the user's picture URI.

Side Bar

Our navigation bar is consistent for all users. It remains the same throughout every step of the application, unless you are looking at a workout progress page. In this specific instance, a new menu option appears to indicate your location. From this page, you must choose where to go, versus clicking a back button.



Dashboard



The dashboard is the user's entry point to the application once logged in. The dashboard displays an overview of the various workouts the user has entered into the application. Information displayed includes the workout name, the number of times the workout has been performed, and the last time the workout was performed. The user is also able to click on the

workout to access the history page for that workout where more detailed statistics about that workout are displayed.

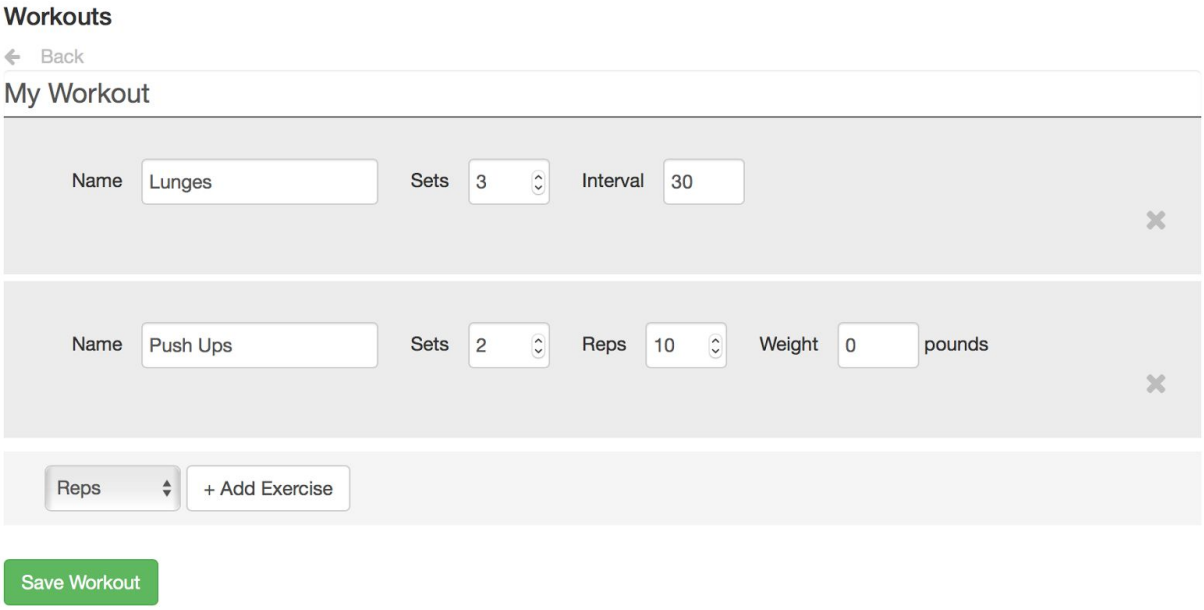
Workouts

The workouts page lists all of the workouts the user has. On this page, the user has the option to either edit, look at the progress they have made, perform, or delete existing workouts. They can also add a workout, which will take them to the Add Workout page.

Add Workout

The Add Workout button on the Workouts page brings the user to a new page where they can change the workout's name and add exercises to the workout. Exercises come in two types: Interval and Rep. Interval exercises represent timed workouts and have input fields for sets and the time interval. Rep exercises represent weighted workouts and have fields for sets, reps, and weight.

Edit Workout



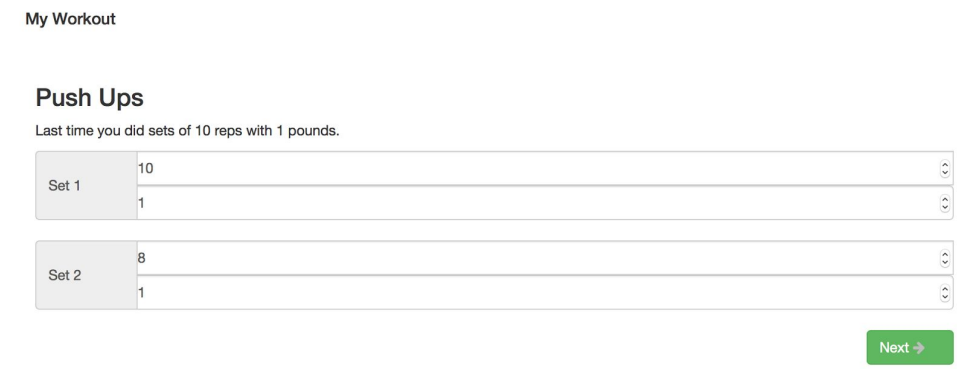
The edit icon on each workout from the Workouts page brings the user to a new page identical to workout creation, but with data filled in from the database. The user can add exercises with the “+Add Exercise” button, remove them by pressing the “x” button, switch the order of exercises by dragging them, or change the workout name. Changes are saved after pressing the “Save Workout” button.

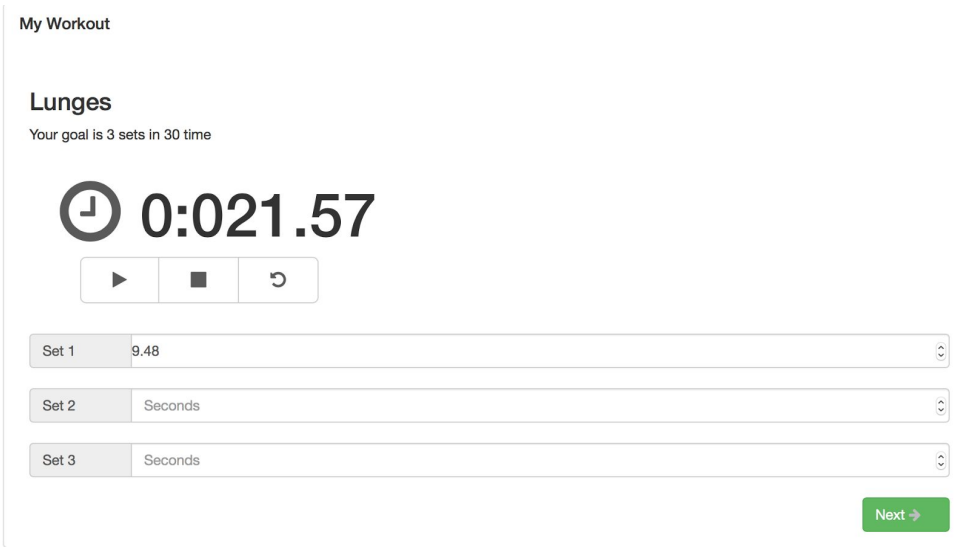
Remove Workout

If for some reason a user wants to remove a workout, they can click on the red trash can icon next to that workout and it will be removed from their dashboard and list of workouts. All data for that workout will also be removed from the database.

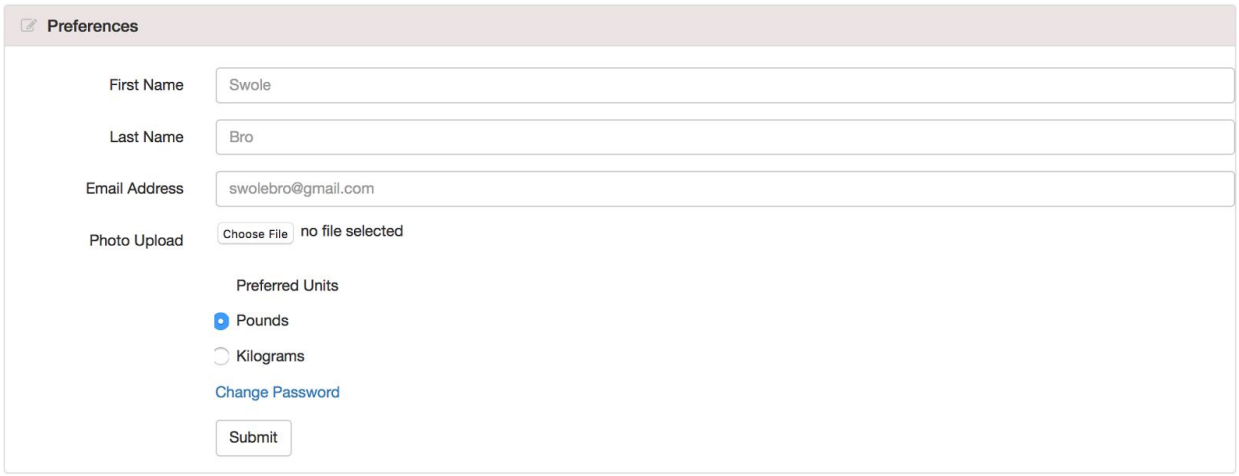
Perform Workout

When the user chooses to start performing a workout, they are brought to the first exercise in their workout. If an exercise is a rep-based exercise, then the user will input the number of reps they completed and the amount of weight they were using for each set. If an exercise is interval-based, then there is a stopwatch at the top of the page and when the user stops the timer it auto populates the respective field with the amount of time doing the set. If the user does not want to use the stopwatch, they can also enter the time interval in by hand. At the top of each page, right below the name of the exercise being performed, is the goal for that day for the exercise. Once the user has finished their exercise, they can click on “Next” to go to the next exercise in their workout.





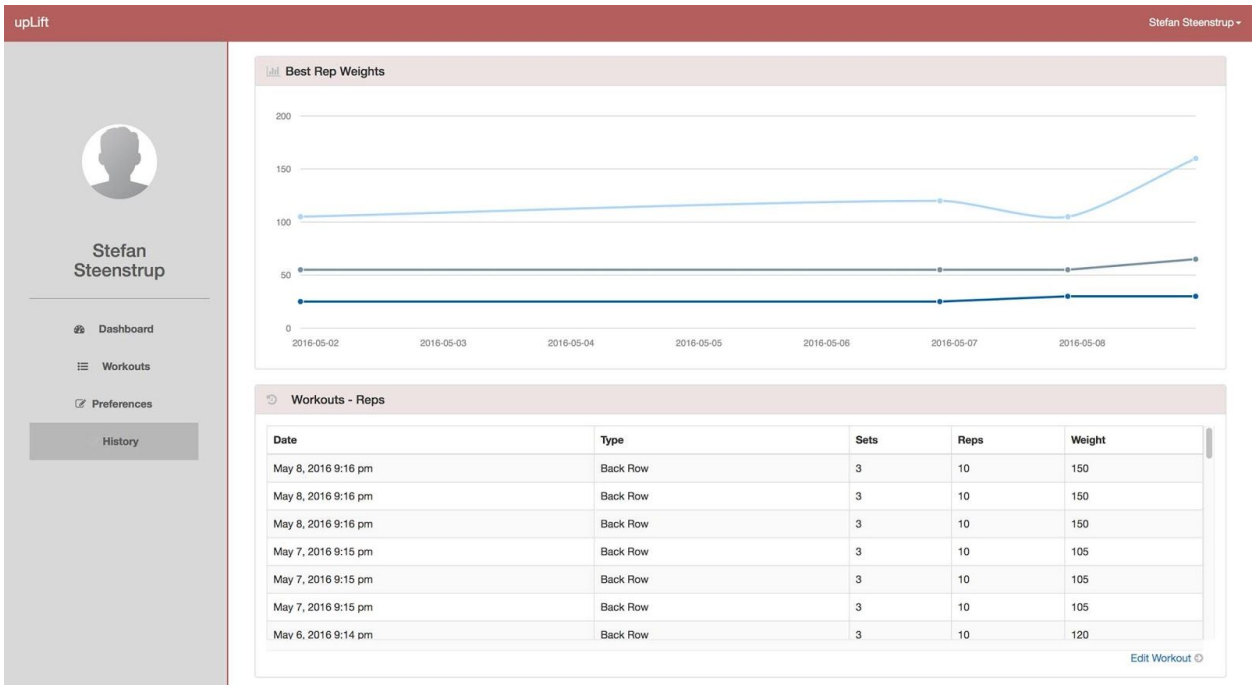
Preferences



On the Preferences page, the input boxes are auto-populated with the user's current information. From here, the user has the ability to edit their first name, last name, email address, password, and preferred units. They can also upload and change their profile picture if they wish

to do so. Once they click submit, these changes are sent to the database and their information is updated.

History



The History page shows the progress the user has made over time on a specific workout. The graphs show lines for each different exercise. The points on the graphs are either their best weight or time (depending if the exercise was for reps or intervals) for that specific date of performance. The charts list every performance for each exercise, including the date, type, and number of sets for that exercise. The chart would either show time spent or weight lifted, depending on the exercise type (rep or interval).