UPenn Transplant Rehabilitation Assistant

A User Guide and Technical Manual

Group 28: Jerry Chang, Chloe Le, Crystal Lee

Contents

User Manual **4**

Login Page 5
Splash Page 6
Change Password Page 7

Exercise Schedule Page 8
Exercises Page 9
Nutrition Page 10
Education Page 11
Pain Page 12
Medications Page 13
Journal Page 14

Add User Page 15
Add Plan Page 16
Create Nutrition Page 17
Create Education Materials Page 18
Create Exercise Page 19
Create Medication List Page 20
Create Pain Survey Page 21

Technical Manual **22**

Login Page 23
Splash Page 24
Change Password Page 25

Exercise Schedule Page 26
Exercise Page 27
Nutrition Page 28
Education Page 29
Pain Page 30
Medications Page 31
Journal Page 32

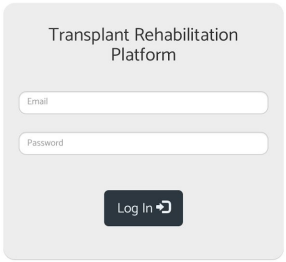
Add User Page 33
Add Plan Page 34
Create Nutrition Page 35
Create Education Materials Page 36
Create Exercise Page 37
Create Medication List Page 38
Create Pain Survey Page 39

Firebase 40

Bug Report **42**

User Manual

Login Page



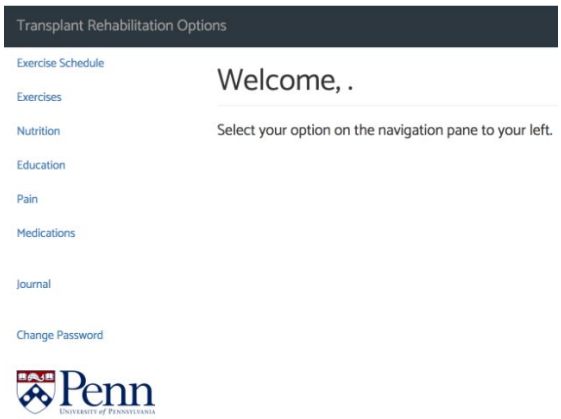
The login page is the page that the user should first see when the web app opens (assuming the user was not previously logged in).

The user should enter his/her email address and corresponding password and click the “Log In” button to enter the web app. A message telling the user to “try again” would appear if the user entered incorrect email or password. Once successfully logged in, the app will take the user to the welcome(splash) page.

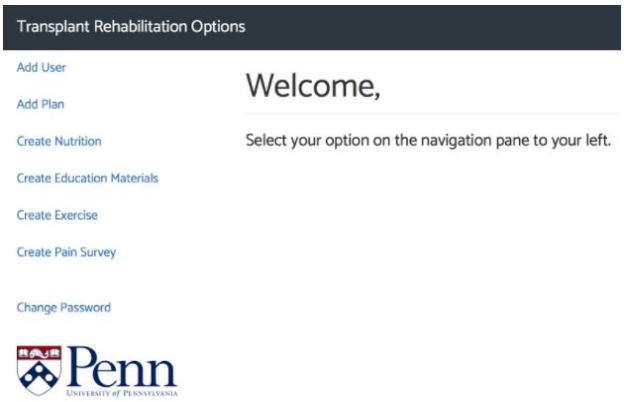
Splash Page

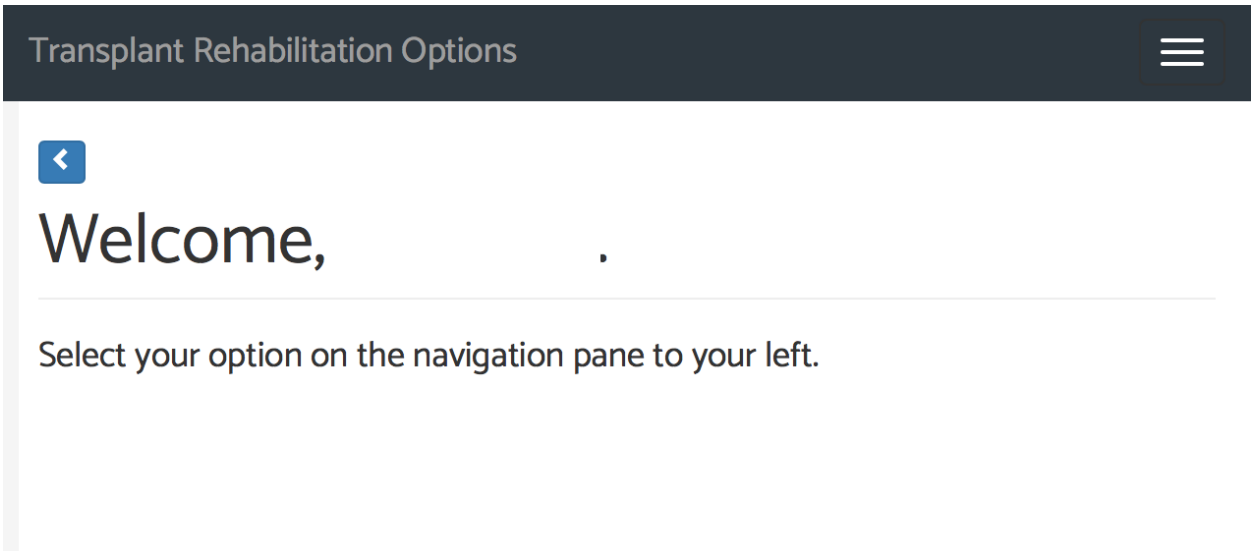
Once the user is successfully logged in, he/she should see the splash page that shows a welcome message and guides the user to select an option from the navigation panel on the left. This navigation panel is different for patients and admin accounts.

Patient account:



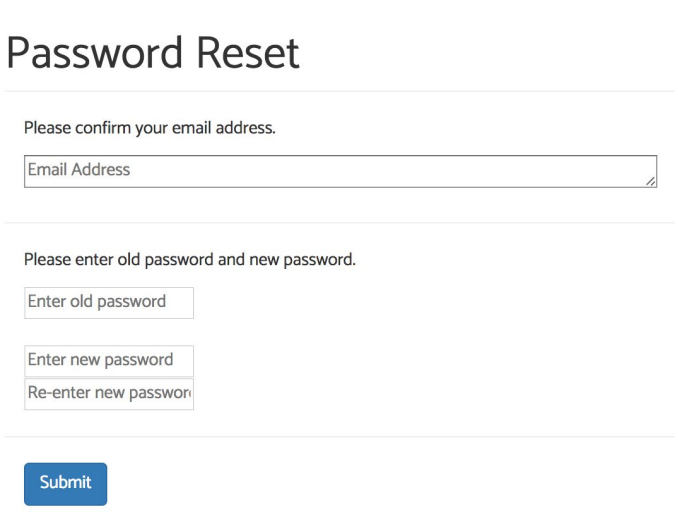
Admin account:





If the window displays an arrow on the top left corner instead of a navigation bar, the user can click the arrow to make the navigation bar appear.

Change Password Page

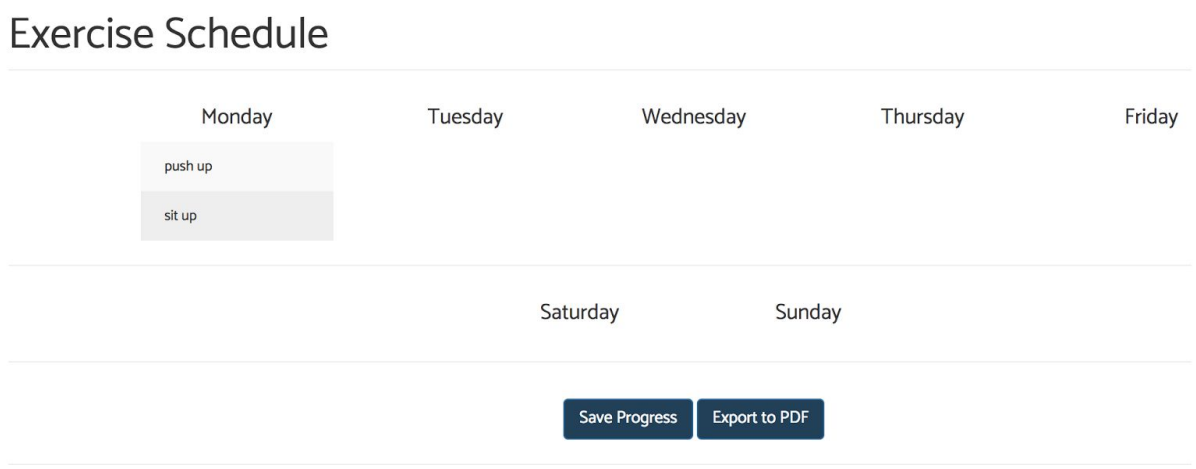


Throughout the usage of this web app, the user can freely choose to reset his/her password by clicking on the change password option in the navigation bar from the splash page.

Once inside this page, the user is asked to enter his/her login email, old password, and new password (twice). Note: the new password must be at least 6 characters long.

Click “submit” when ready. If all entries were valid, a message saying “your password has been changed successfully” will appear.

Exercise Schedule Page



The Exercise Schedule Page is a patient-side page. The patient will see the exercises assigned to him/her by the physician listed under each day. If an exercise appears underneath a day, that means the user should try to complete this exercise on that day.

Once the exercise or exercises are completed for that day, the user can check off the exercise/exercises by clicking on exercise name(s). Next, the user can click “save progress” to record the completion.

Exercise completion entries will show on the bottom of the page. If the patient wishes to print or save the entry record, he/she can choose to “export to PDF”.

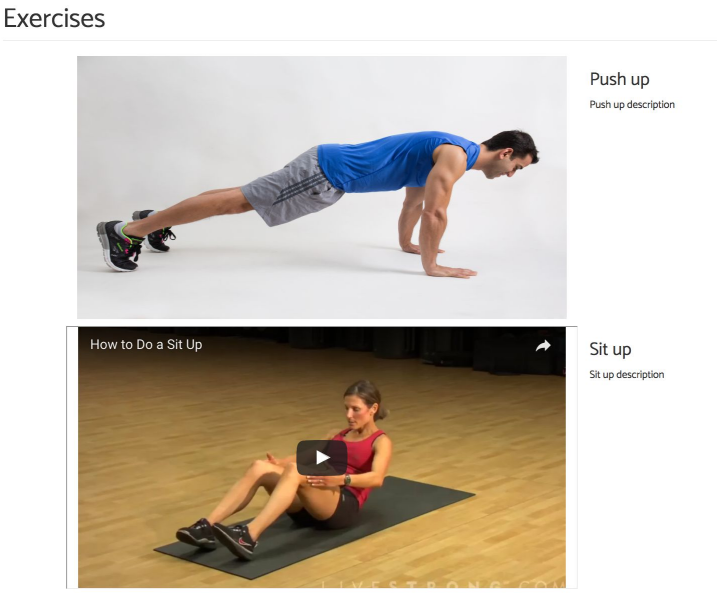
Note: simply refresh the page to uncheck the exercise(s).

Exercises Page

The Exercises Page is a patient-side page. When the patient click on “exercises” in the navigation bar, a page with a similar layout to the one on the right will appear.

On this page, the patient will see all relevant exercises corresponding to his/her transplantation plan.

For each exercise, there is a visual demonstration (either a video or an image, if available) on the left and a text-description on the right.

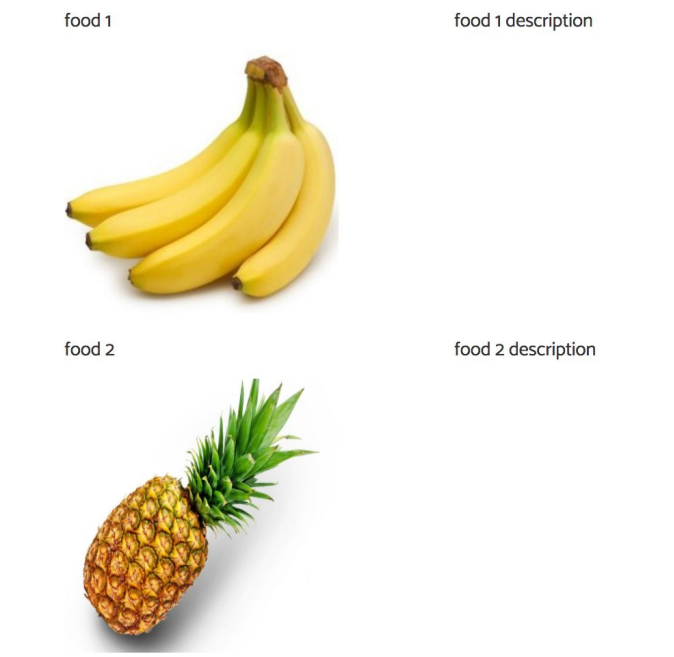


Nutrition Page

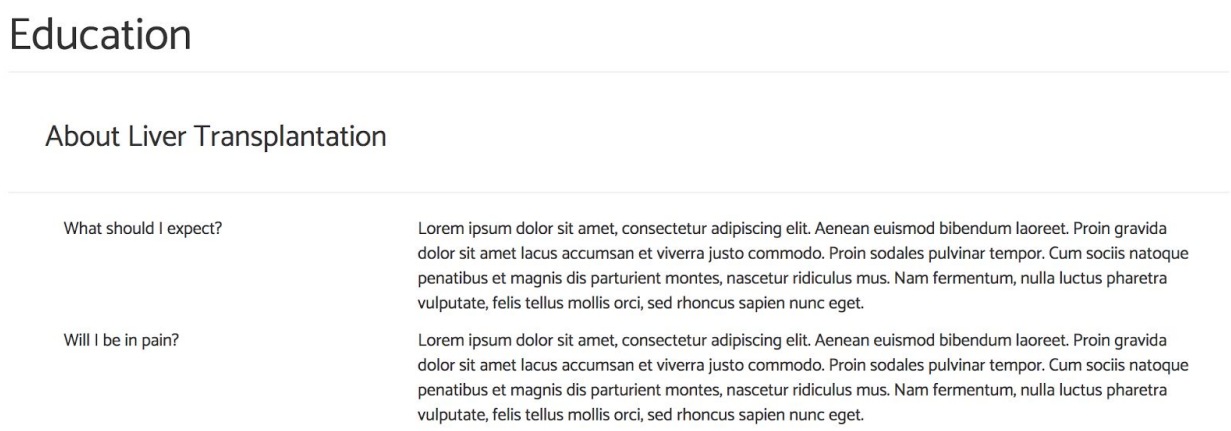
The Nutrition Page is a patient-side page. When the patient click on “nutrition” in the navigation bar, a page with a similar layout to the one on the right will appear.

On this page, the patient will see a relevant nutrition plan corresponding to his/her transplantation plan.

For each suggested food item, there is an image, if available, on the left and a text-description on the right.



Education Page



The Education Page is a patient-side page. When the patient click on “education” in the navigation bar, a page with a similar layout to the one above will appear.

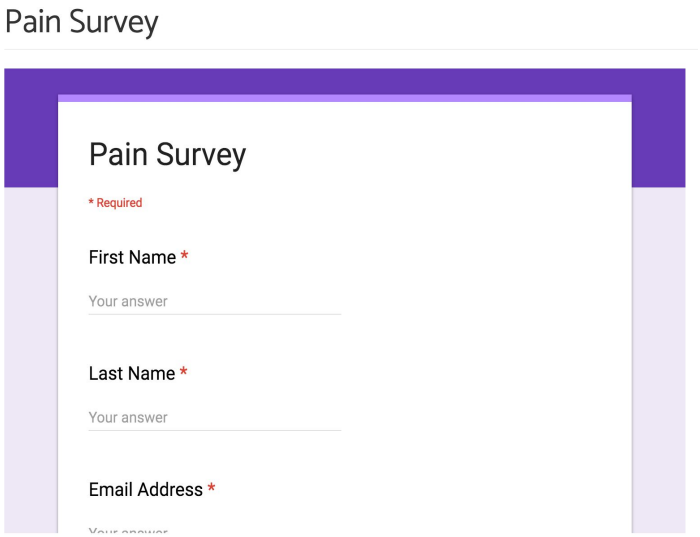
On this page, the patient can learn about relevant information corresponding to his/her transplantation.

Pain Page

The Pain Page is a patient-side page. When the patient clicks on “pain survey” in the navigation bar, he/she will see an embedded survey page show up on the right side of the window.

This survey is designed by the physician in order to allow the patient to record down the level of pain he/she felt on that day.

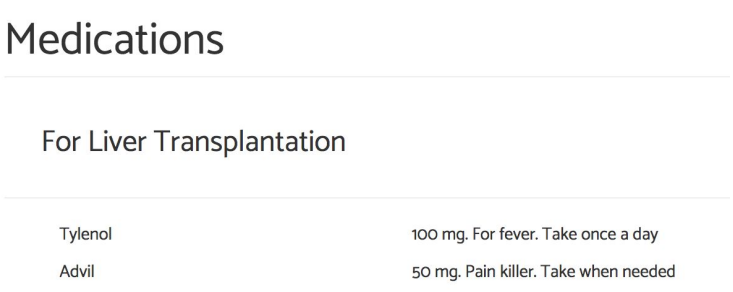
In order to take this survey, the patient will have to enter all required fields. Click “submit” when ready to turn in the survey.



Medications Page

The Medications Page is a patient-side page. When the patient clicks on the “medication” option, a page with a similar layout to the one on the right will appear.

On this page, the patient can learn about relevant medication information corresponding to his/her transplantation.



Journal Page

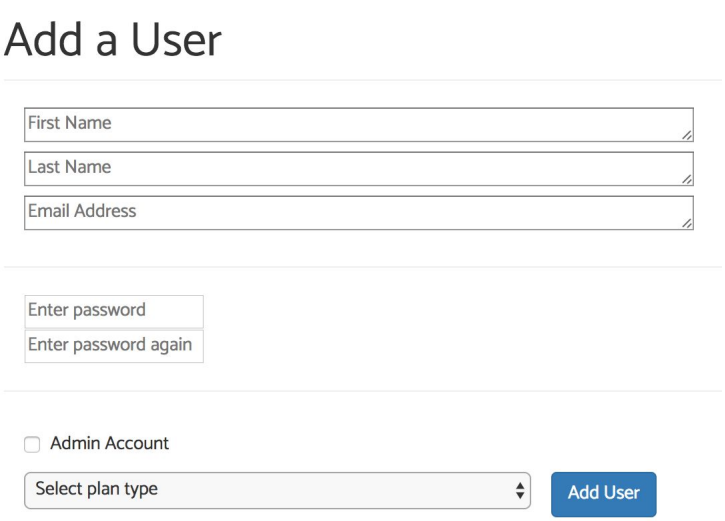
The Journal Page is a patient-side page.

In this page, a patient is able to type out an answer for the question that their physician has provided for them.

By clicking the “Save” button, the patient can save their entries to be viewed at any point in the future. The entry is saved on the bottom of the page, along with the current date.

By clicking the “Export to PDF” button, the patient can export all their previously logged entries as a PDF, which they can then print to bring to a meeting with their physician.

Add User Page



The Add User Page is an admin-side page.

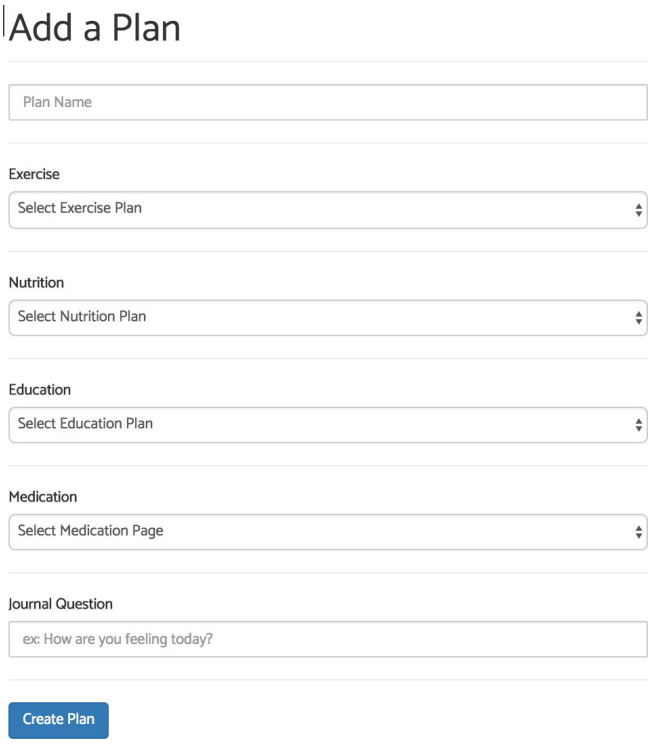
To add a new user, the physician needs to enter the new user's first and last name, email address, and initial password (which can be changed by the user later).

If the new user is a patient, do NOT check the "admin account" box; if the new user is a physician, do check the "admin account" box.

Lastly, remember to select the appropriate transplantation plan from the "select plan type" drop-down box. If adding an admin

When all fields are entered correctly, click "add user". If successful, a message will appear telling you so.

Add Plan Page



The Add Plan Page is an admin-side page.

If the physician wishes to add a new plan, he/she will need to give the new plan a name that has not been used before.

Next, the physician should select the relevant exercise, nutrition, education, medication plans from the drop-down boxes.

When choosing an exercise plan, the physician will see check boxes show up with the days he/she would like the patient to perform these exercises. Simply check the days by clicking on the boxes.

Finally, the physician should enter a question he/she wishes the patient to answer on the journal page.

When all fields are filled, click “create plan”. A message will pop up if the plan is created successfully.

Create Nutrition Page

The Create Nutrition Page is an admin-side page.

The physician should enter a new nutrition plan name that has not been used previously.

Then, he/she should enter the first food item's name, its description, and a corresponding image link.

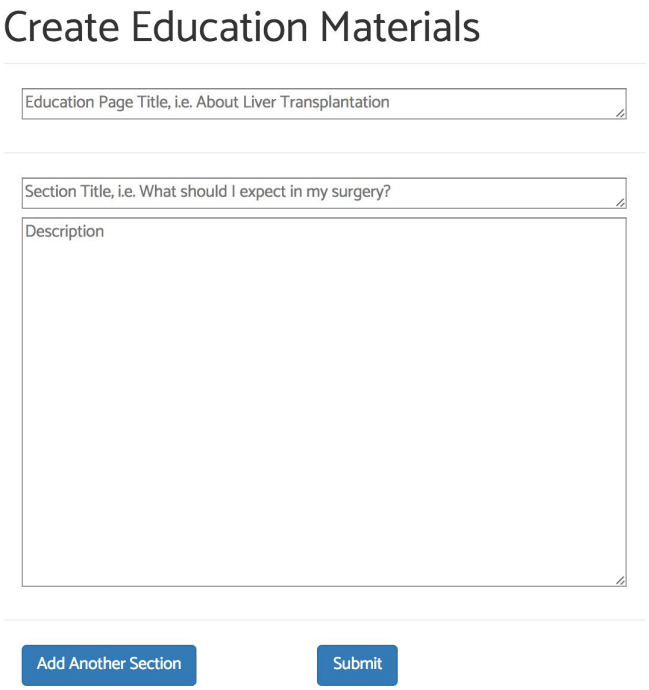
If the physician wishes to add more food item(s), he/she can click "add another food". Doing so will clear out the previous food item's entry in the text boxes, so the physician can then add another food item to this nutrition plan.

There is no limit to how many food items can be included under one nutrition plan, so the physician can keep adding food items until he/she is ready to submit this nutrition plan.

When ready, click "submit".

Create Education Materials Page

The Create Education Materials Page is an admin-side page.



The physician should enter a new education plan name that has not been used previously.

Then, he/she should enter the first section's title and its description.

If the physician wishes to add more section(s), he/she can click "add another section". Doing so will clear out the previous section's entry in the text boxes, so the physician can then add another section to this education plan.

There is no limit to how many sections can be included under one education plan, so the physician can keep adding until he/she is ready to submit this education material.

When ready, click "submit".

Create Exercise Page

The Create Exercise Page is an admin-side page.

The physician should enter a new exercise plan name that has not been used previously.

Then, he/she should enter the first exercise's name, its description, and a corresponding image/video link.

*NOTE: if both video and image links were entered, the visual demonstration that appears on patient's exercise page will be the video. Otherwise, if only one link is provided, then that visual type will be the one showing up.

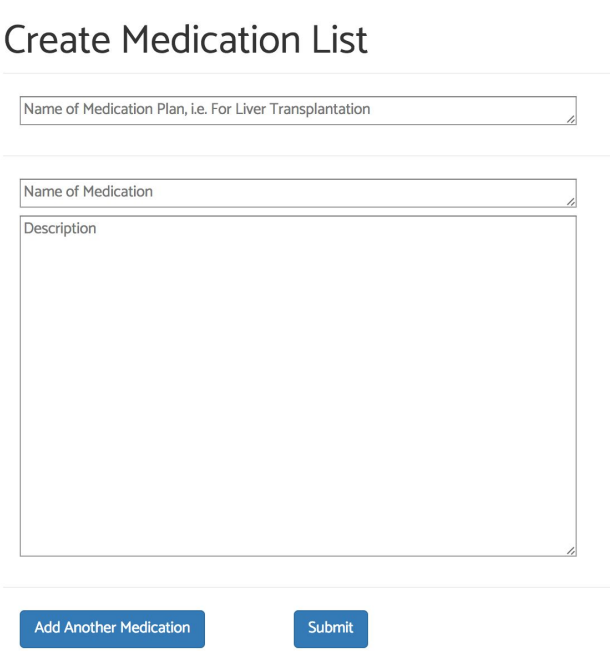
**NOTE: when entering a video link, make sure it is the EMBED LINK.

If the physician wishes to add more exercise(s), he/she can click "add another exercise". Doing so will clear out the previous exercise's entry in the text boxes, so the physician can then add another exercise to this exercise plan.

There is no limit to how many exercises can be included under one exercise plan, so the physician can keep adding exercises until he/she is ready to submit this exercise plan.

When ready, click "submit".

Create Medications List Page



The Create Medications List Page is an admin-side page.

The physician should enter a new medication plan name that has not been used previously.

Then, he/she should enter the first medication's name and its description.

If the physician wishes to add more medication item(s), he/she can click "add another medication". Doing so will clear out the previous medication's entry in the text boxes, so the physician can then add another medication to this plan.

There is no limit to how many medications can be included under one medication plan, so the physician can keep adding medications until he/she is ready to submit this medication plan.

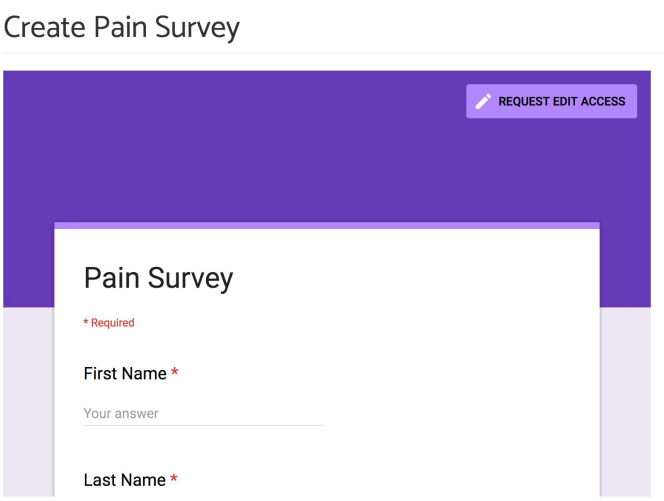
When ready, click "submit".

Create Pain Survey Page

The Create Pain Survey Page is an admin-side page.

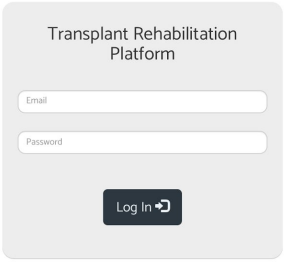
As the pain survey is hosted by Google Forms, the admin simply has to click the “request edit access” button to edit the pain survey through Google Forms.

Note: Changing this survey will change it for all patients under all plans!
Note 2: If the survey does not show up, try accessing the page in a different browser. Alternatively, you can also go to forms.google.com to access the form directly.



Technical Manual

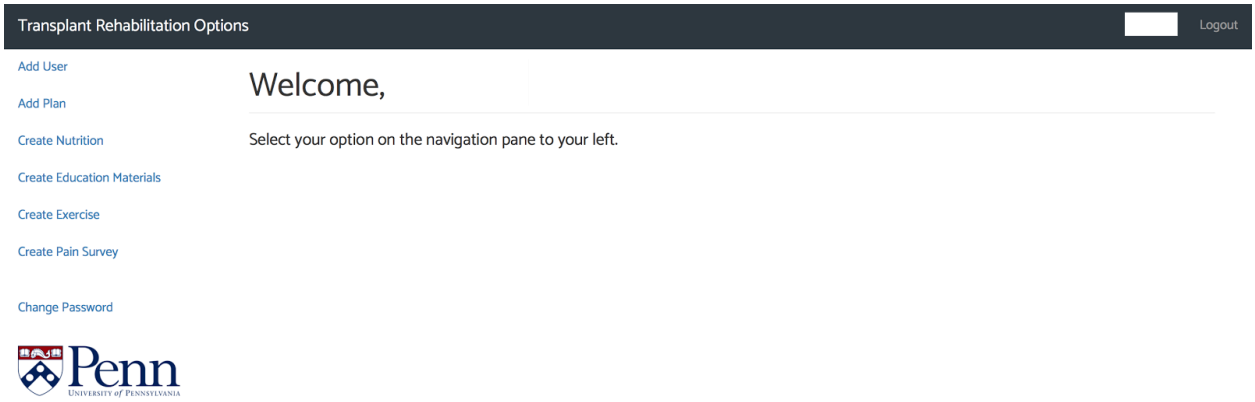
Login Page



HTML: login.html
CSS: css/login.css
JavaScript: scripts/login.js

This page needs to check whether or not the entered email and password match any user's information in the database.

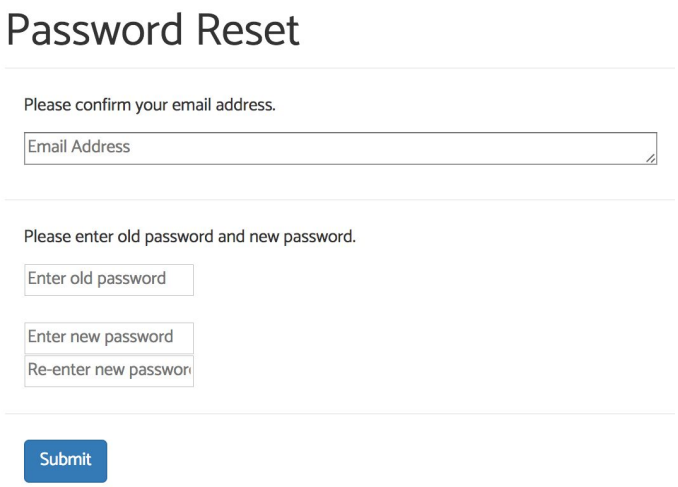
Splash Page



HTML: splash.html
CSS: css/dashboard.css
JavaScript: scripts/splash.js

This page reads from the database the current user's full name and displays it. It also reads whether the current user is an admin or not, and displays the appropriate sidebar.

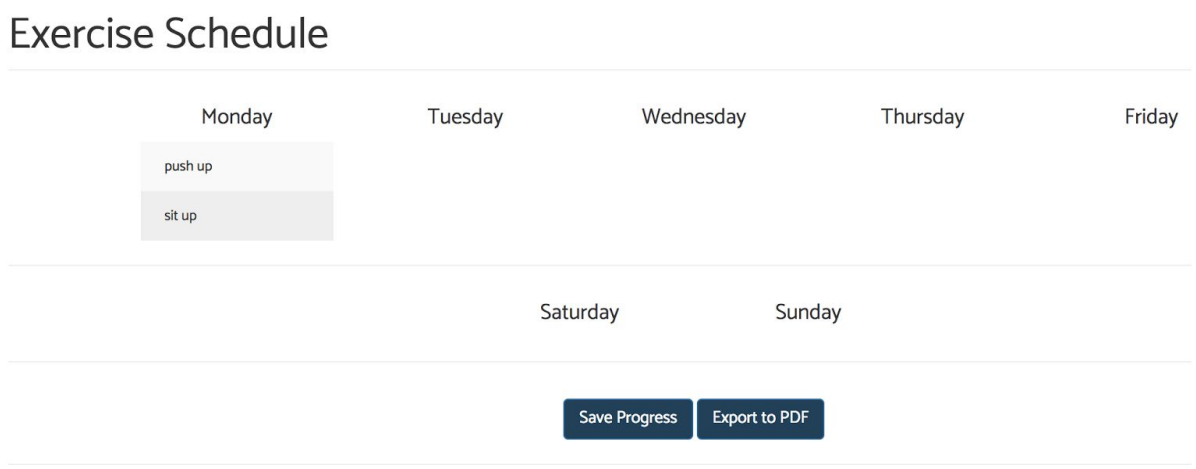
Change Password Page



HTML: change_password.html
CSS: css/dashboard.css
JavaScript: scripts/change_password.js

This page checks that the entered email and password match any user's information in the database. If so, it checks that the new password is inputted correctly, and if everything is correct, it changes the password associated with the entered email address.

Exercise Schedule Page



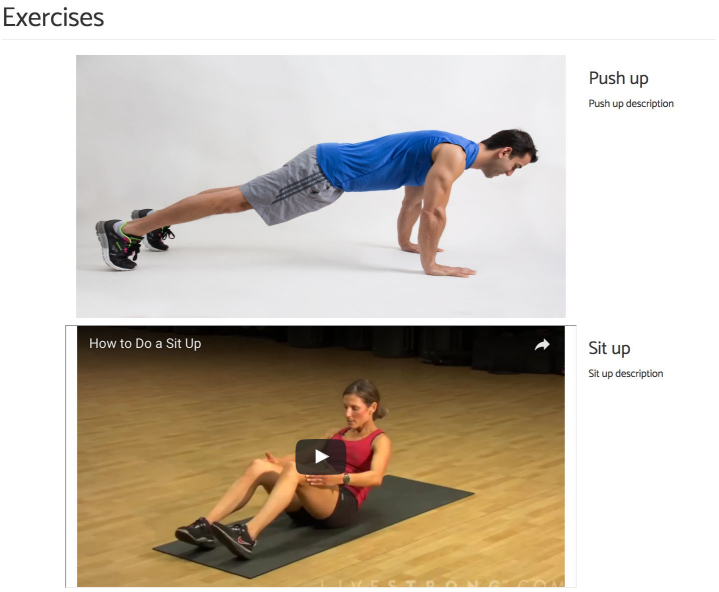
HTML: exercise_schedule.html
CSS: css/exercise_schedule.css
JavaScript: scripts/exercise_schedule.js

This page reads from the database and displays the corresponding information to the user. Each user is assigned with a unique transplantation plan. The javascript file finds the user's plan and its assigned exercise material in the database, and then formats the information to be displayed on the exercise schedule page.

When "save" is clicked, the completed exercises should get saved with a date and be written to the database: inside the user's Journal tag. Then, the javascript code calls the displayPastEntries function again to display the entries saved in real time.

When "export to PDF" is clicked, it runs the exportButton.onClick function.

Exercise Page

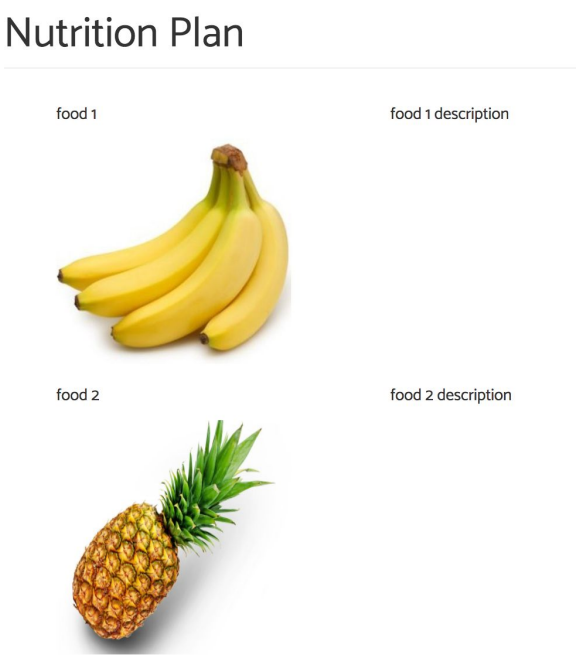


HTML: exercises.html
CSS: css/dashboard.css
JavaScript: scripts/exercise.js

This page reads from the database and displays the corresponding information to the user. Each user is assigned with a unique transplantation plan. The javascript file finds the user's plan and its assigned exercise material in the database, and then formats the information to be displayed on the exercise page.

Exercise plans include links to a photo and a video to be displayed along with the descriptions. If the video link is not null, display the video inside the iFrame; if the video is null but not the image link, then display the image. Else, a blank frame will appear.

Nutrition Page

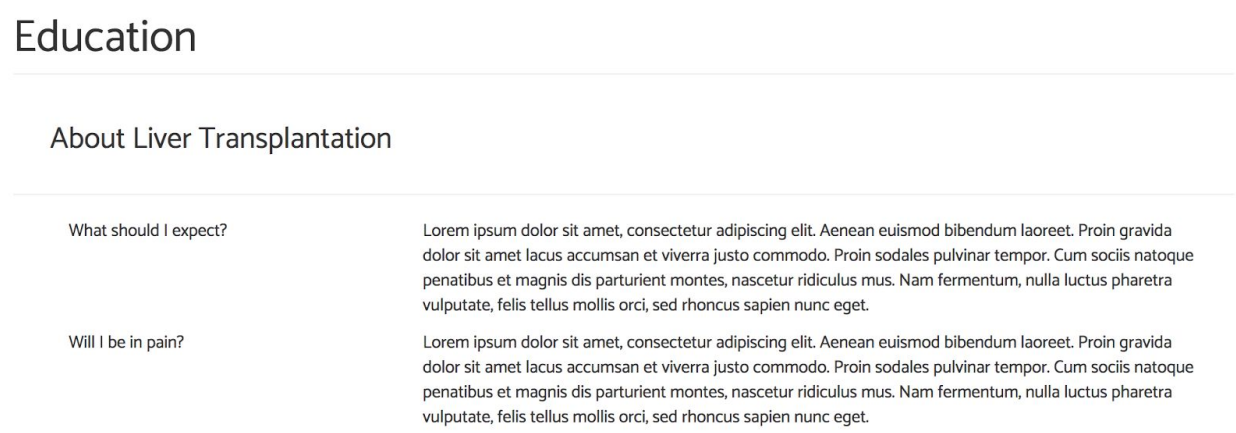


HTML: nutrition.html
CSS: css/dashboard.css
JavaScript: scripts/nutrition.js

This page reads from the database and displays the corresponding information to the user. Each user is assigned with a unique transplantation plan. The javascript file finds the user's plan and its assigned nutrition material in the database, and then formats the information to be displayed on the nutrition page.

Nutrition plans also includes a link to a photo to be displayed along with the food item. If the link is not null, display the photo alongside the descriptions.

Education Page



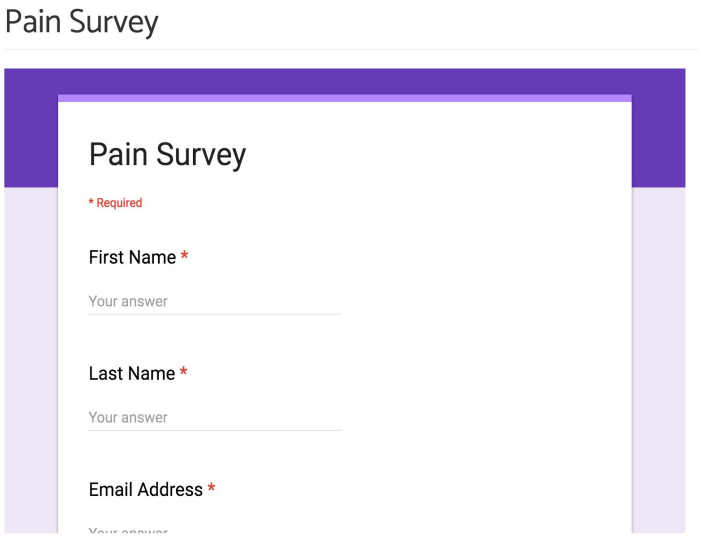
HTML: education.html
CSS: css/dashboard.css
JavaScript: scripts/education.js

This page reads from the database and displays the corresponding information to the user. Each user is assigned with a unique transplantation plan. The javascript file finds the user's plan and its assigned education material in the database, and then formats the information to be displayed on the education page.

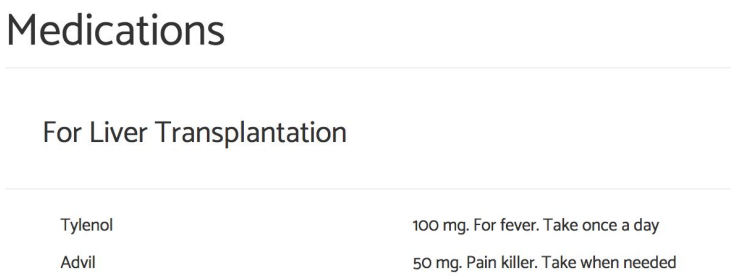
Pain Page

HTML: pain.html
CSS: css/dashboard.css

This page simply embeds a Google Form's page. The embedding is done inside the html file.



Medications Page



HTML: medication.html
CSS: css/dashboard.css
JavaScript: scripts/medication.js

This page reads from the database and displays the corresponding information to the user. Each user is assigned with a unique transplantation plan. The javascript file finds the user's plan and its assigned medication material in the database, and then formats the information to be displayed on the medications page.

Journal Page

HTML: journal.html
CSS: css/dashboard.css
JavaScript: scripts/journal.js

This page displays the question that is unique to the patient's corresponding plan. This is done within the js file's displayQuestion function.

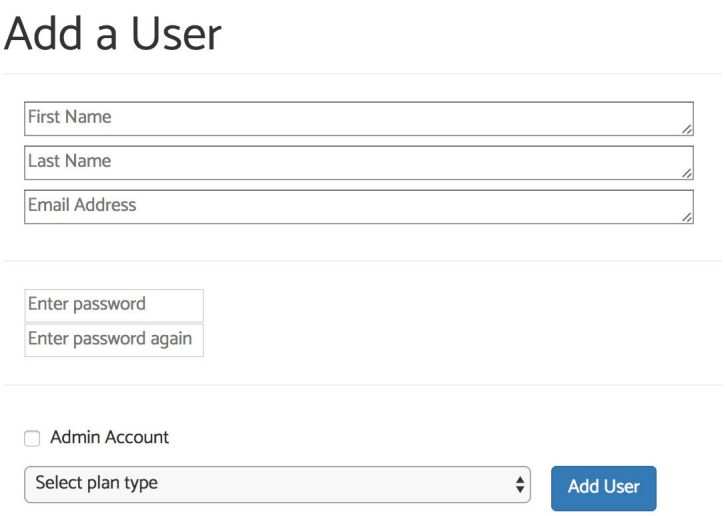
In addition, it contains a textbox for the user to answer the journal question.

Past entries are automatically loaded to be displayed at the bottom of the page when page initially opens.

When "save" is clicked, the entry that was inside the textbox should get saved with a date and be written to the database: inside the user's Journal tag. Then, the javascript code calls the displayPastEntries function again to display the entries saved in real time.

When "export to PDF" is clicked, it runs the exportButton.onClick function.

Add User Page



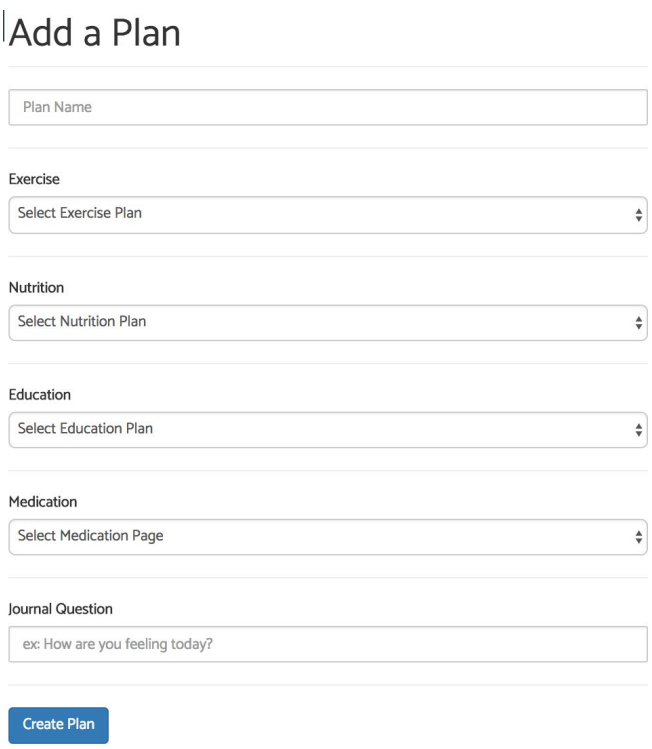
HTML: admin/add_patient.html
CSS: css/dashboard.css
JavaScript: scripts/add_patient.js

This page checks that the email address is of a valid format and that the password is over 6 characters long and is inputted correctly. It also reads from the database to find the list of plans to choose from. If the user is valid, it adds the new user into the database. If the user is specified as an admin, the user's "isAdmin" value is inputted as "Yes". Otherwise, the user's "isAdmin" value is "No".

Add Plan Page

HTML: admin/add_plan.html
CSS: css/dashboard.css
JavaScript: scripts/add_plan.js

This page reads from the database to produce the list of possible exercise plans, nutrition plans, education plans, and medication plans. If all the fields are filled out correctly, it creates a new entry in the database's "Plans" section with the inputted values.



Create Nutrition Page

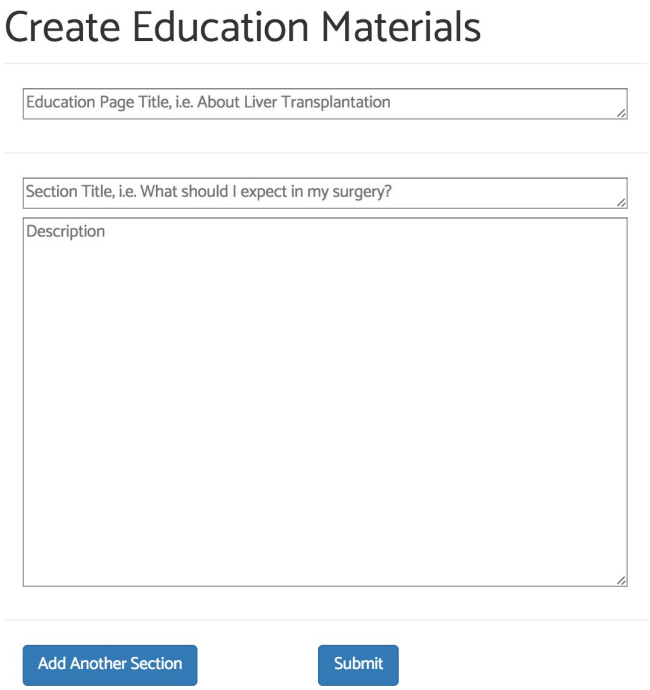
HTML: admin/create_nutrition.html
CSS: css/dashboard.css
JavaScript: scripts/create_nutrition.js

This page adds a new entry under the database's "Nutrition" section. Empty inputs are valid for all sections.

Create Education Materials Page

HTML: admin/create_education.html
CSS: css/dashboard.css
JavaScript: scripts/create_education.js

This page adds a new entry under the database's "Education" section. Empty inputs are valid for all sections.



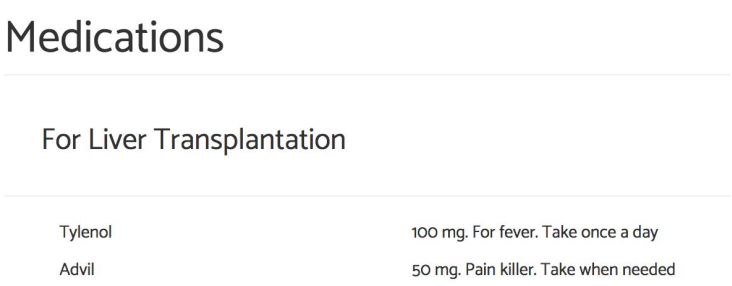
Create Exercise Page

HTML: admin/create_exercise.html
CSS: css/dashboard.css
JavaScript:
scripts/create_exercise.js

- **saveExercise():** stores data currently in the text fields (section title, description) accordingly into the Titles and Descriptions list objects.
- **savePlan():** pushes the data stored in the Exercises, Description, Images, and Video list objects to the Firebase database. The name of the page is pushed first as an entry, and the medications (title and description) are then pushed as child entries under the name of the page.

This page adds a new entry under the database's "Exercise Plans" section. Empty inputs are valid for all sections.

Create Medication List Page

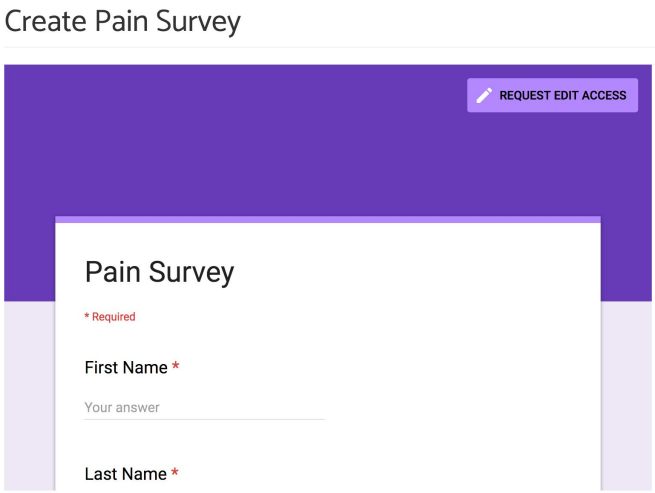


HTML: admin/create_medication.html
CSS: css/dashboard.css
JavaScript: scripts/create_medication.js

- **saveSection():** stores data currently in the text fields (section title, description) accordingly into the Titles and Descriptions list objects.
- **savePlan():** pushes the data stored in the Title and Description list objects to the Firebase database. The name of the page is pushed first as an entry, and the medications (title and description) are then pushed as child entries under the name of the page.

Create Pain Survey Page

HTML: admin/create_survey.html
CSS: css/dashboard.css
JavaScript: scripts/create_survey.js



This page embeds the Google Form page according to the admin's google account, which can be accessed from:

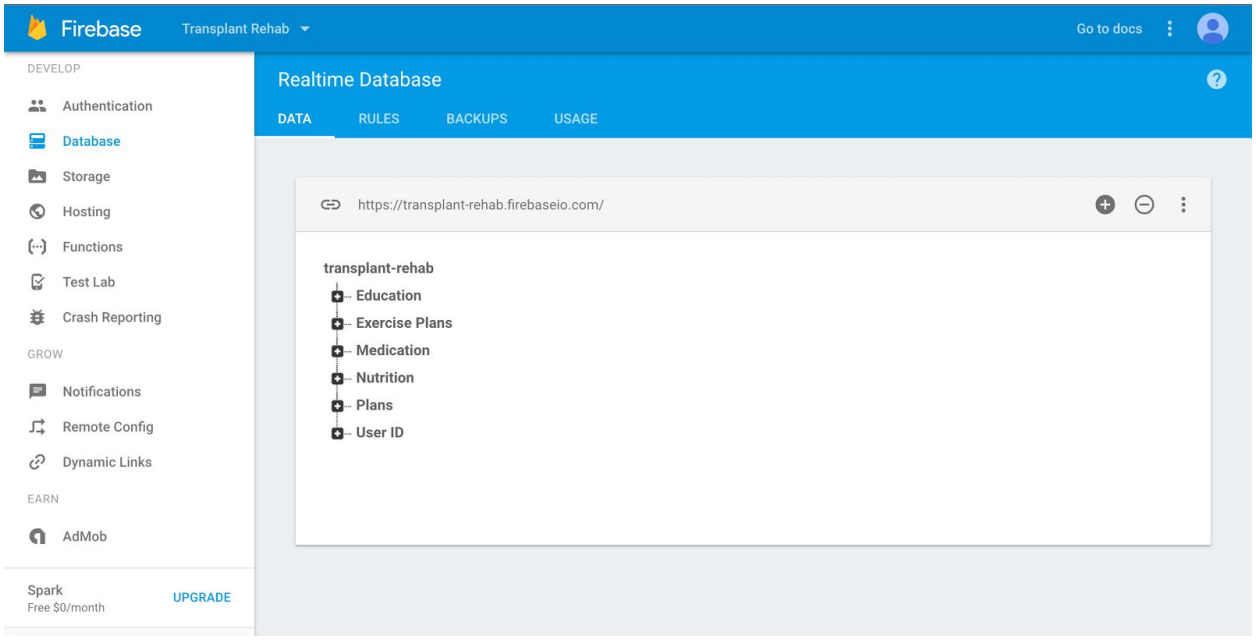
Email: upenntransplantrehab@gmail.com
Password: transplant1234

After signing in, the physician can either access responses or modify the questions. Both actions can be done within the web app or by opening up a new Google Form window.

The embedding of the page is done inside the html file:

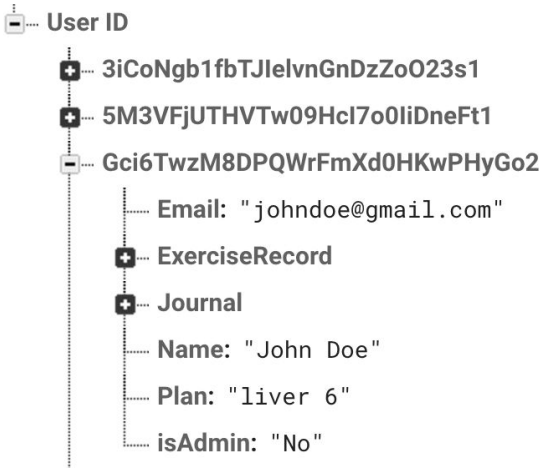
```
96          <div class = "survey">
97            <iframe src="https://docs.google.com/forms/d/1oToxg7aZKhvxePN4l579iktCF4GPp3K-
              AbKsxiLTFWI/edit?embedded=true" width="760" height="500" frameborder="0" marginheight="0"
              marginwidth="0">Loading...</iframe>
98          </div>
99        </div>
```

Firebase



The database behind the Transplant Rehabilitation Assistant can be accessed through firebase.google.com. The login details are as follows:

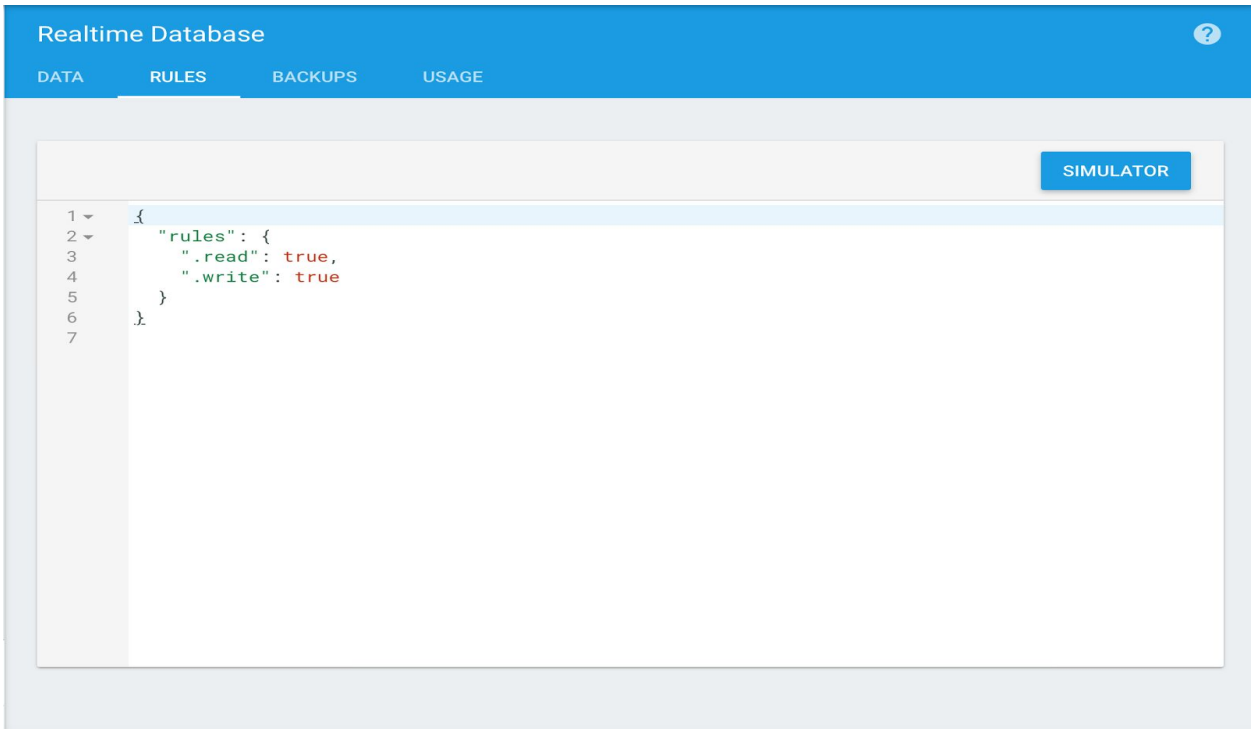
Email: upenntransplantrehab@gmail.com
Password: transplant1234



Through this database, an admin can delete and modify existing plans and users. Firebase uses unique keys to store its information as seen in the above image, so unfortunately, an admin must search through these keys to find the entries they want.

To make this process easier, admins are encouraged to regularly delete unneeded users and plans.

NOTE: If the app ever experiences read/write to database issues, please go to the “rules” section in Firebase, and make sure it follows the code below.



NOTE: every html file that wishes to interact with the database needs to initialize firebase:

```
<!--FIREBASE INITIALIZATION -->

<script src="https://www.gstatic.com/firebasejs/3.7.3/firebase.js"></script>
<script>
  // Initialize Firebase
  var config = {
    apiKey: "AIzaSyCK2L9denM40KSqqNExFrRnZhGpijcvgDc",
    authDomain: "transplant-rehab.firebaseapp.com",
    databaseURL: "https://transplant-rehab.firebaseio.com",
    storageBucket: "transplant-rehab.appspot.com",
    messagingSenderId: "592884792214"
  };
  firebase.initializeApp(config);
</script>
```

Bug Report

Here are a list of bugs we know are present in the app in its current version:

- When you deploy the app to firebase, a bug where you have trouble writing to the database may occur
    - A solution to fix that bug is to follow the note about changing “rules” in Firebase on page 41 of this manual.