ITRW 324 – MVC PROJECT

**Van Wyk, JJ**

Student number:

25950665

[WELCOME TO LUFTGRID 1](#_Toc531646708)

[DATABASE 1](#_Toc531646710)

[2.1 Explaining the ERD 1](#_Toc531646712)

[MODELS 3](#_Toc531646713)

[3.1 Approach to creating version control 3](#_Toc531646715)

[3.2 Repository logic 3](#_Toc531646716)

[CONTROLLERS 4](#_Toc531646717)

[4.1 Approach to connecting html to MySQL 4](#_Toc531646719)

[VIEWS 5](#_Toc531646720)

[5.1 Static Inputs 5](#_Toc531646722)

[5.2 Dynamic Inputs 5](#_Toc531646723)

[GUI OVERVIEW 7](#_Toc531646724)

[6.1 HomePage.php 7](#_Toc531646726)

[6.2 SignUp.php 8](#_Toc531646727)

[6.3 FirstRepo.php 9](#_Toc531646728)

[6.4 SourcePage.php 10](#_Toc531646729)

[6.5 Sidepanel Commits 11](#_Toc531646730)

[6.6 Sidepanel Report Issues 12](#_Toc531646731)

[6.7 Sidepanel Settings 13](#_Toc531646732)

[6.8 Sidepanel Logout 14](#_Toc531646733)

List of Figures

[Figure 2‑1: ERD 1](#_Toc531646734)

[Figure 4‑1: Connecting to cloud sql (sql.php) 4](#_Toc531646735)

[Figure 5‑1: Input method for static inputs 5](#_Toc531646736)

[Figure 6‑1: Simple Login Page 7](#_Toc531646737)

WELCOME TO LUFTGRID

# Heading 1 won’t print. Don’t delete – doing so will lead to incorrect numbering.

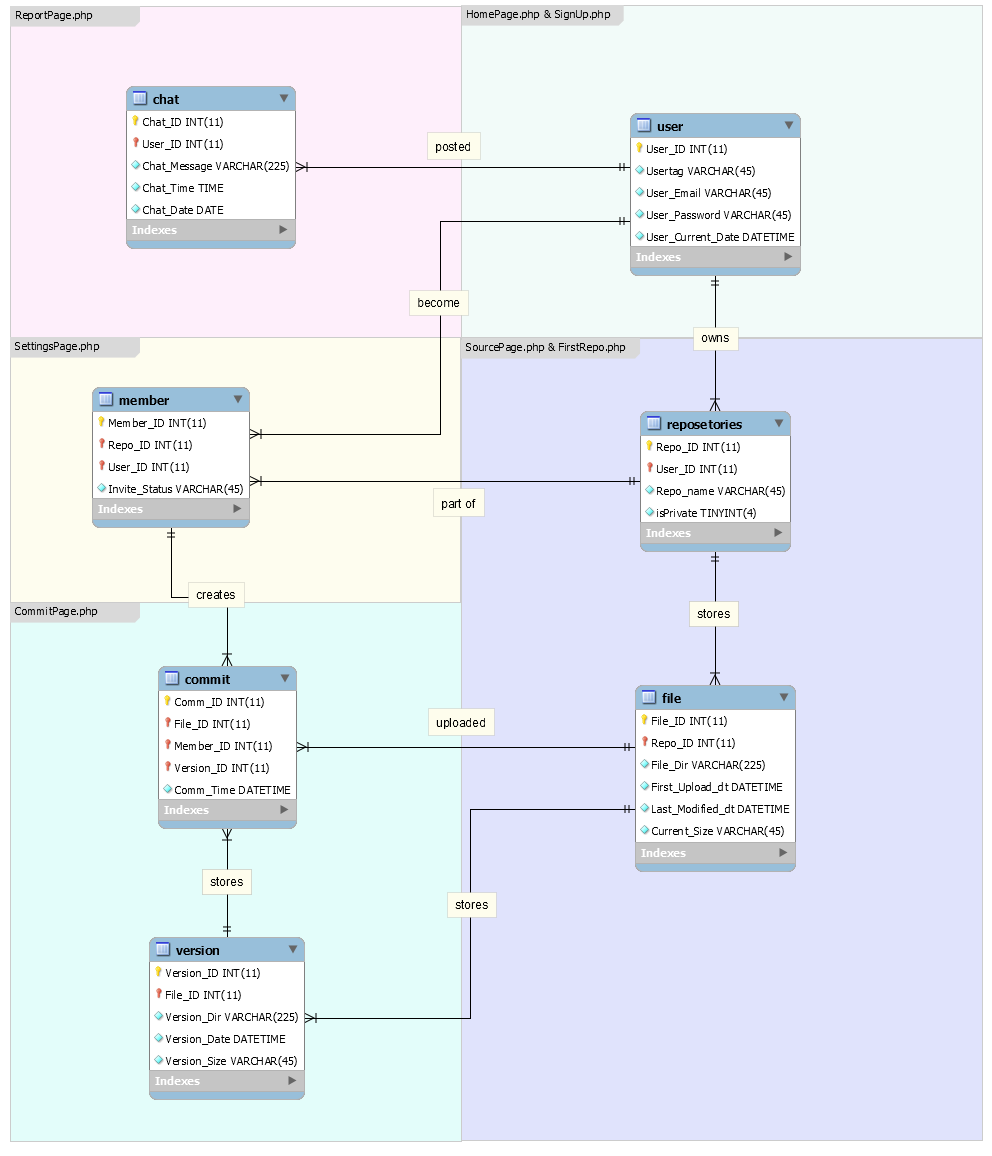
DATABASE

# Heading 1 won’t print. Don’t delete – doing so will lead to incorrect numbering.

## Explaining the ERD

* The user that creates a repository will have their User\_ID saved in the reposetories.User\_ID as well as well as becoming the first member of their repository
* Every time a commit is created a version of a file is created as well.
* Every commit made updates the “Last\_Modified\_dt” and “Current\_Size” of the related file
* A member can have 3 invite states: PENDING, ACCEPTED, REJECTED.
  + PENDING is made when a invite is send to a private repository
  + ACCEPTED is update after the owner of a repository accepts an invite (owner set to ACCEPTED automatically) or automatically set to ACCEPTED if invite is send to a public repository
  + REJECTED is updated after the owner rejects an invite, member will not be able to join the team
* Only references to the uploaded files and versions of the files are saved in the database. Files are store on the server.

Figure 2‑1: ERD



MODELS

# Heading 1 won’t print. Don’t delete – doing so will lead to incorrect numbering.

## Approach to creating version control

* First and most importantly all files uploaded are not stored inside the database, only a reference the where the file is stored in the “repositories” folder, under a folder named after the repository.
* Every uploaded/committed file is stored in the “versions” folder, under a folder named after the repository as well, with a extension “.luft” followed by a unique number(ex. StudentMarks.pdf.luft67 )
* All of this means that it is very easy to version control changes to files of a repository, even files of the exact same name, and store how uploaded the file, the date of the upload and the size of the upload
  + PROS
    - Any file of any type can be stored and controlled with ease.
  + CONS
    - Since every commit saves an entire version of a file, it can that a lot of space on the server. Much large then just a normal changelog would.

## Repository logic

* Every user can have multiple repositories.
* Repositories needs a unique name.
* Repositories can be made public (anyone can join as long as they know the repositories name) or private (can only before a member by invite).
* Owner decides how becomes a member of a private repository.
* User can easily switch between there and repositories they are member of at the “SourePage.php” after login.
* After signing up a repository must be made first before continuing.

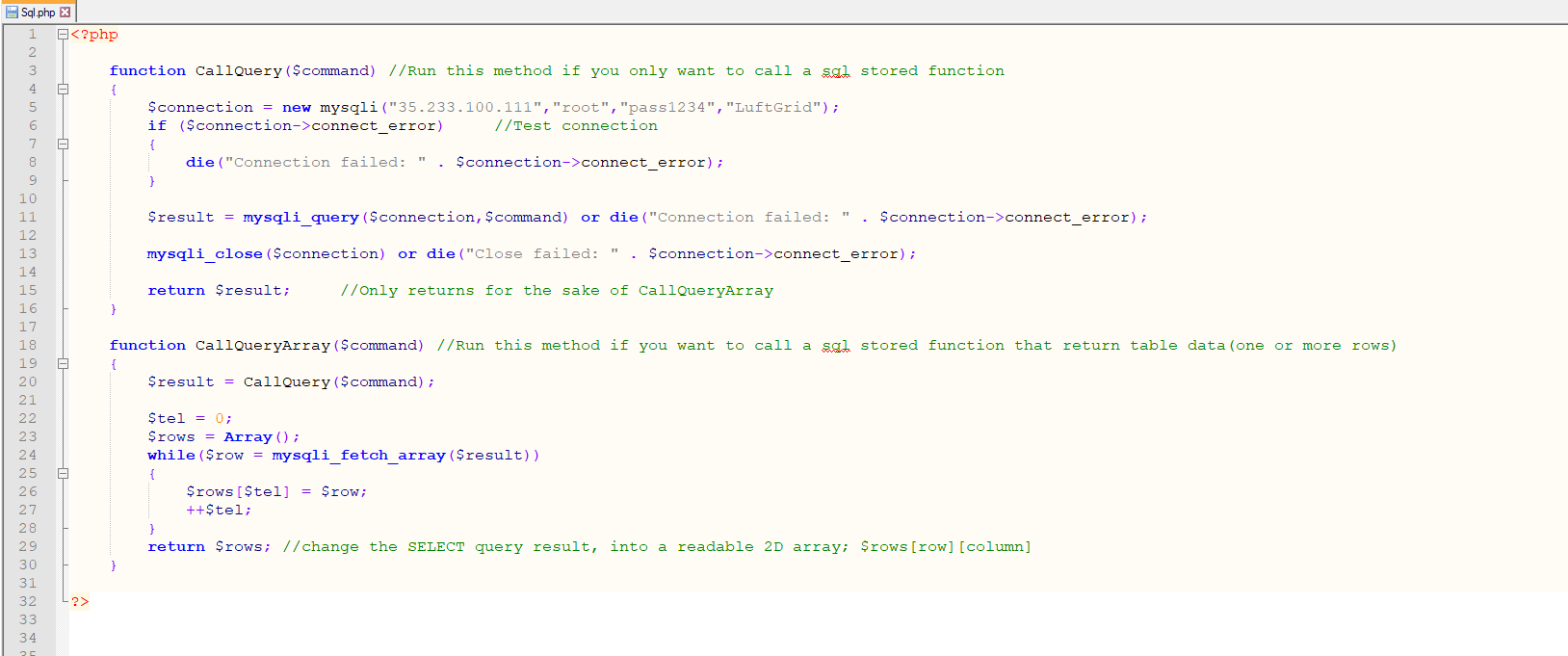
CONTROLLERS

# Heading 1 won’t print. Don’t delete – doing so will lead to incorrect numbering.

## Approach to connecting html to MySQL

* PHP was uses to code functionality to the website, as well as connect to the database
* The following code was used to call a sql query:

Figure 4‑1: Connecting to cloud sql (sql.php)



* 35.233.100.111 is the IP for the Cloud SQL database on Google Cloud (need to whitelist your computer ip the access the database first).
  + CallQuery was used for calling procedures that did not return anything from the database (ex. CallQuery($name, $email, $password); ).
  + CallQueryArray was used when returning one or more rows of info from database (ex. $result[ ][ ] = CallQueryArray($repoName, $info); ).

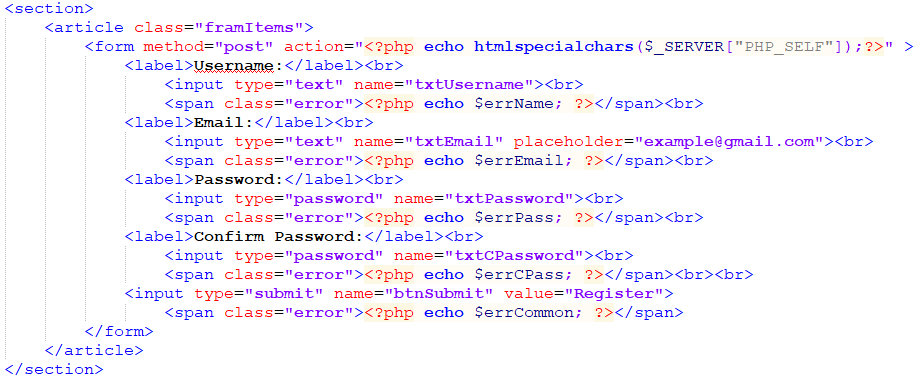
VIEWS

# Heading 1 won’t print. Don’t delete – doing so will lead to incorrect numbering.

## Static Inputs

* Most of my input on a html website used <form> tag in this format:

Figure 5‑1: Input method for static inputs

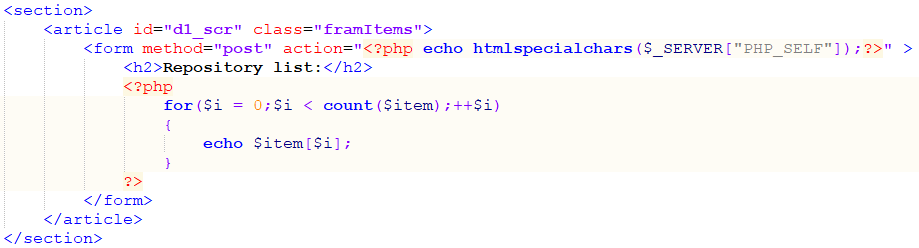


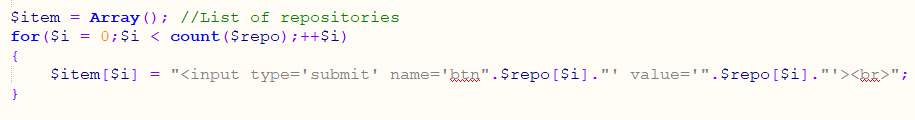
* A label followed by an input (type = text/password/submit) then an error message
* After submit button is clicked a “if ($\_SERVER["REQUEST\_METHOD"] == "POST")” runs and depending on what button was clicked ( isset($\_POST['buttonName']) ) a php function while run using all the given information

## Dynamic Inputs

* When the number of inputs where uncertain, ex. a list of buttons, the following was used

Figure 5‑2: Input method for dynamic inputs





* HTML code is echo’ed with the correct information

GUI OVERVIEW

# Heading 1 won’t print. Don’t delete – doing so will lead to incorrect numbering.

## HomePage.php

Figure 6‑1: Simple Login Page



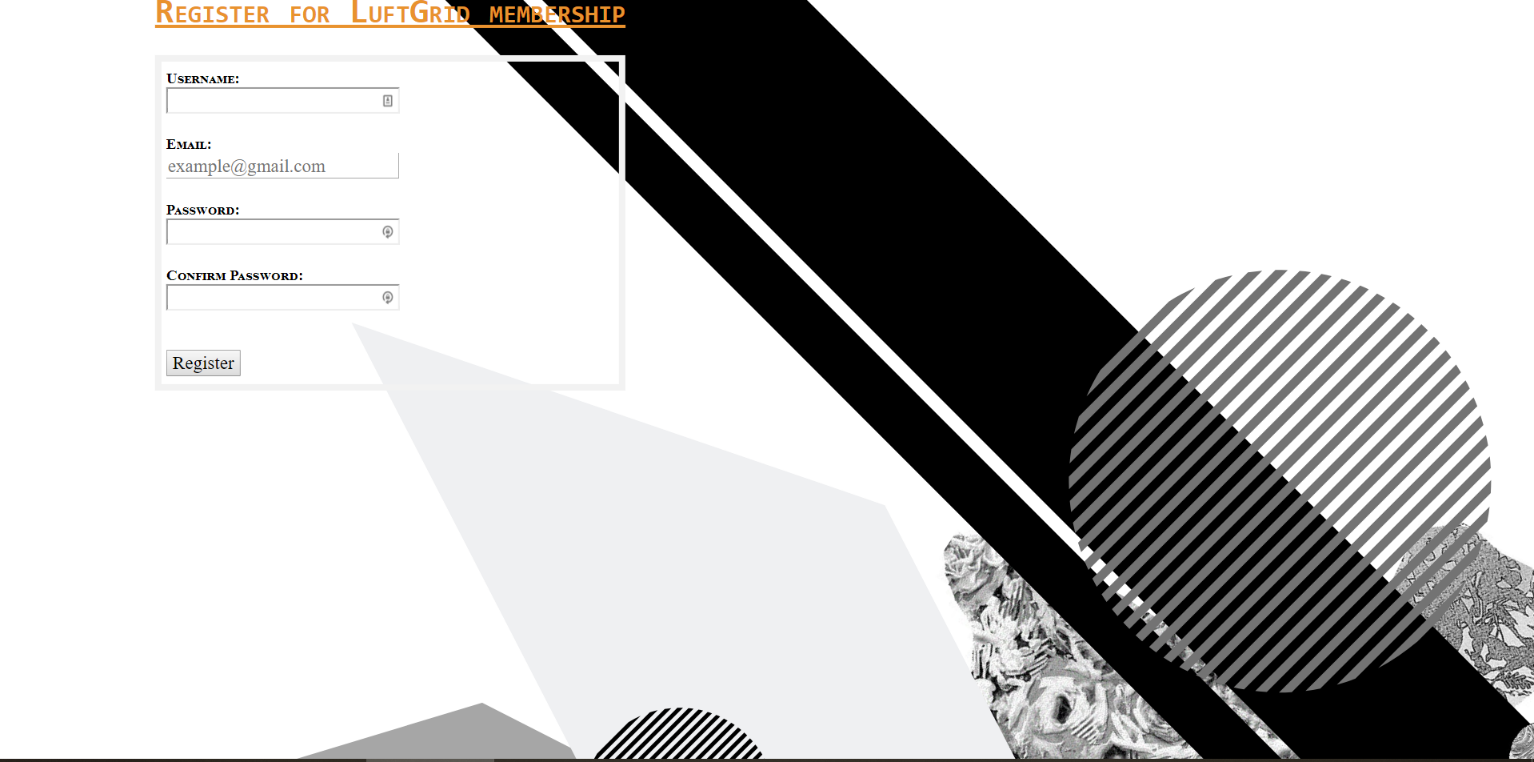
Figure 6‑2: Login with errors



* After “Signup” link is clicked

## SignUp.php

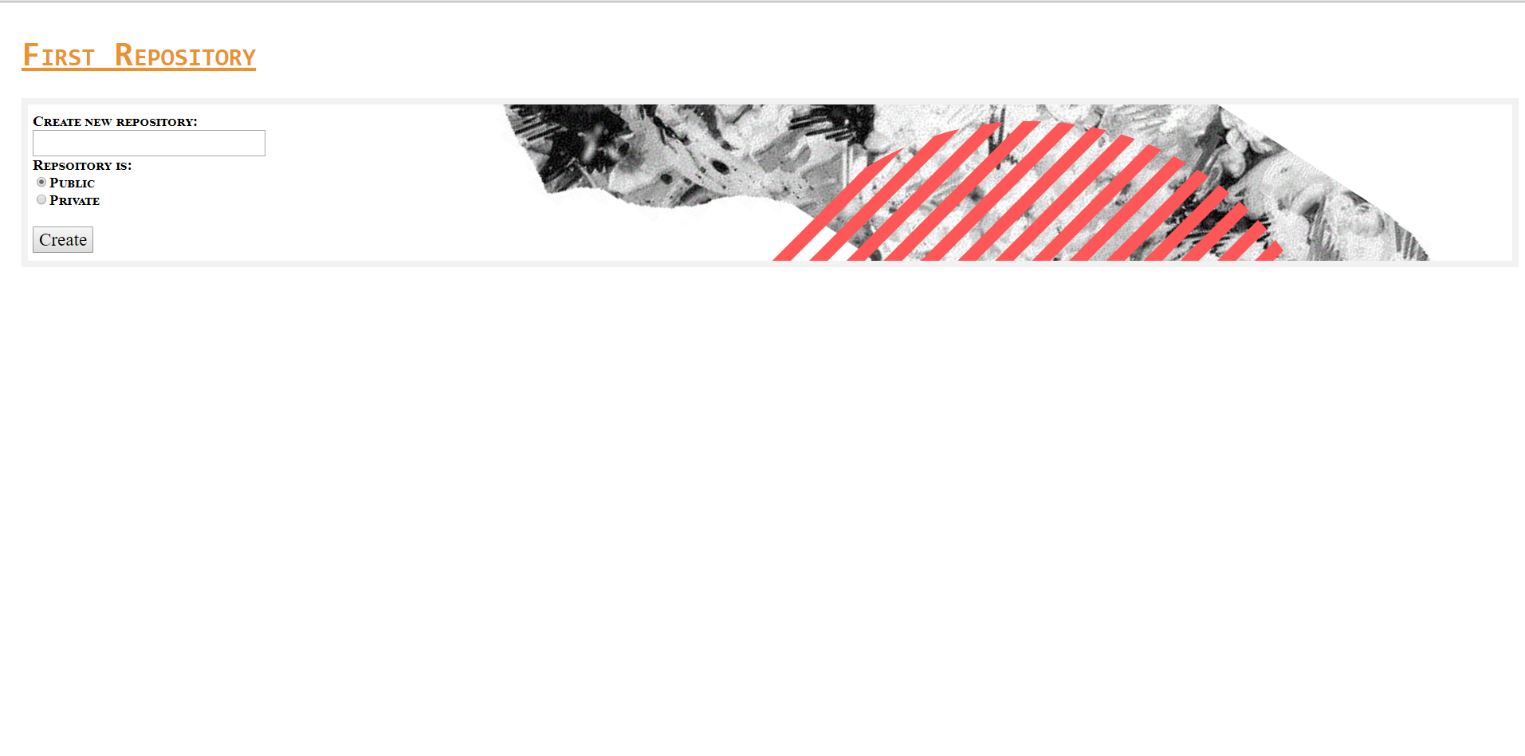
Figure 6‑3: SignUp for your own account



* After successful register

## FirstRepo.php

Figure 6‑4: Creating your first repo



* Can be made public or private
* After successful create AS WELL AS successful login

## SourcePage.php

Figure 6‑5: SourcePage with examples of file already uploaded

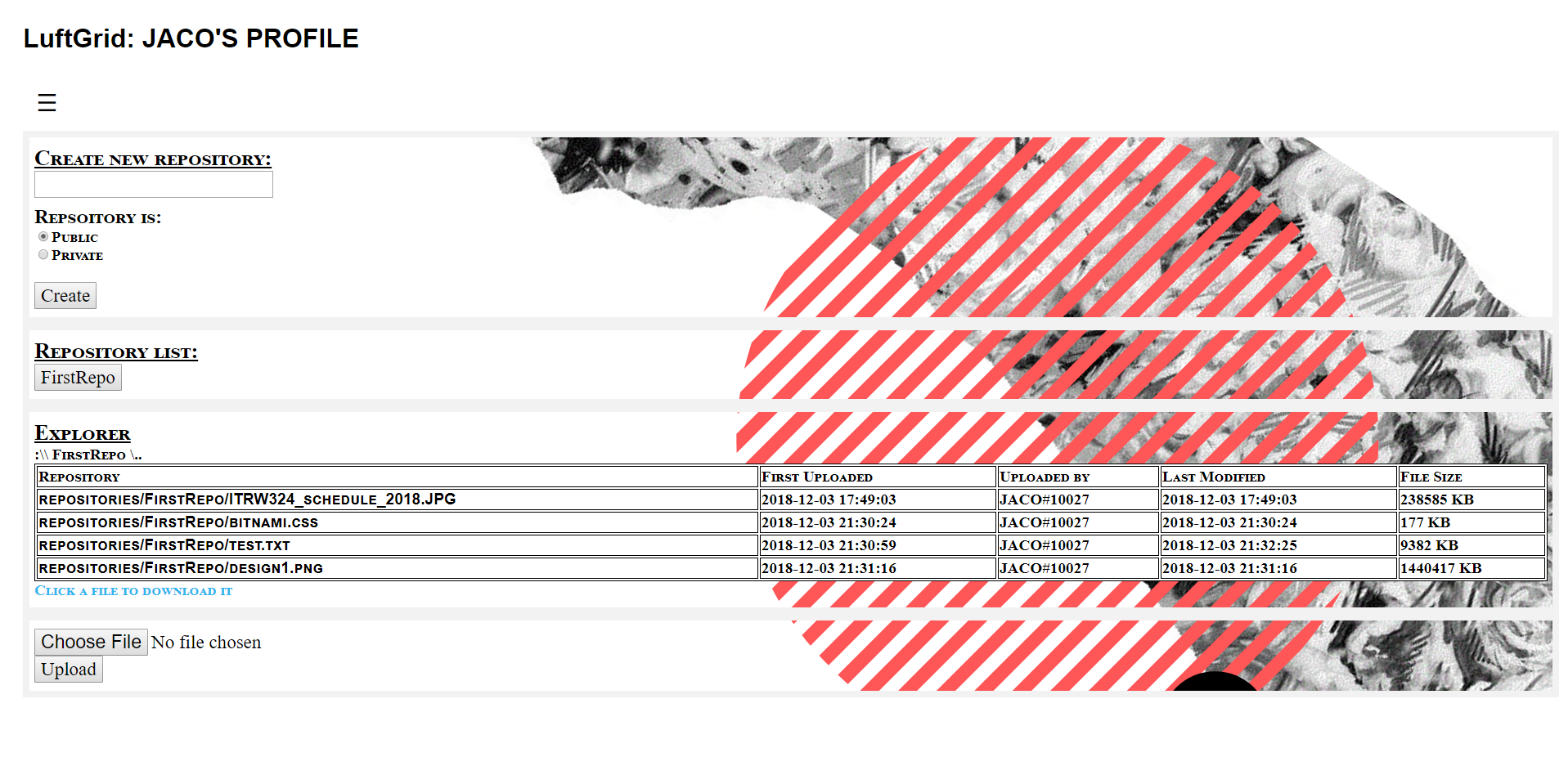
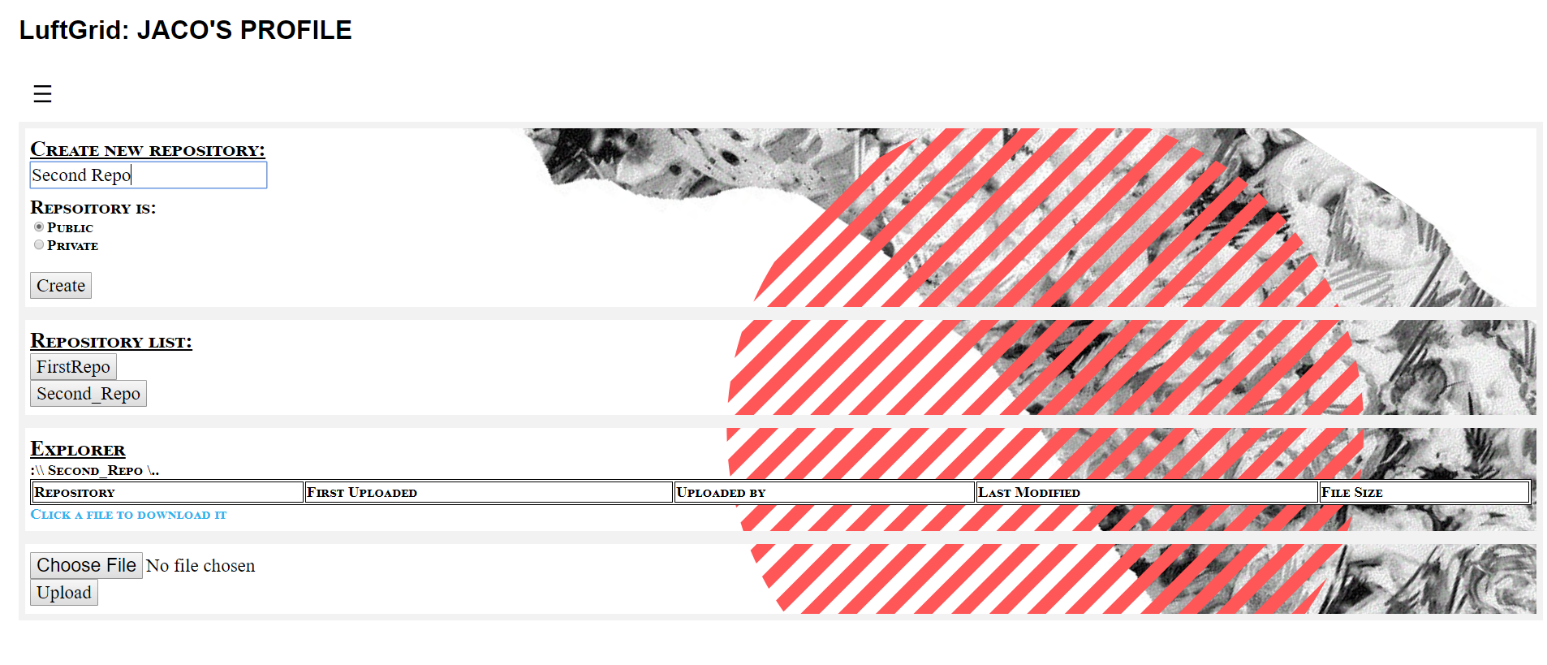


Figure 6‑6: SourcePage with new repository created



* On SourePage you can:
  + Create a new repository
  + Switch between repositories
  + See and download latest versions of files in selected repository
  + Upload a file to selected repository
* If you click the menu button (the icon with the 3 vertical lines) you should see this

Figure 6‑7: Sidepanel popup

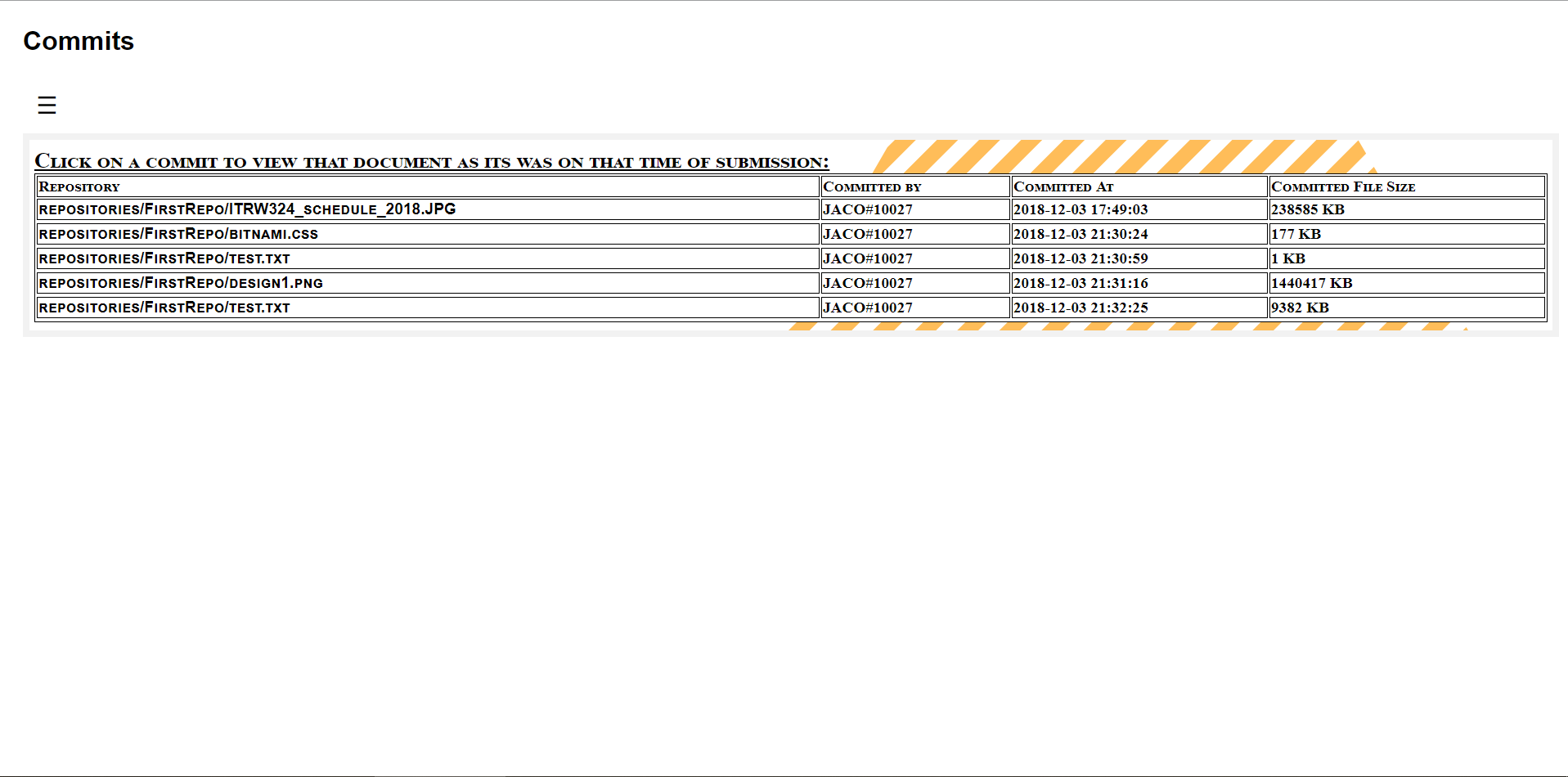


* “Source” is what we are currently on
* Switch between the 4 menu items, the selected repository will be carried over between them

## Sidepanel Commits

* Clicking on the “Commits” link

Figure 6‑8: Displays all commits by every member and time

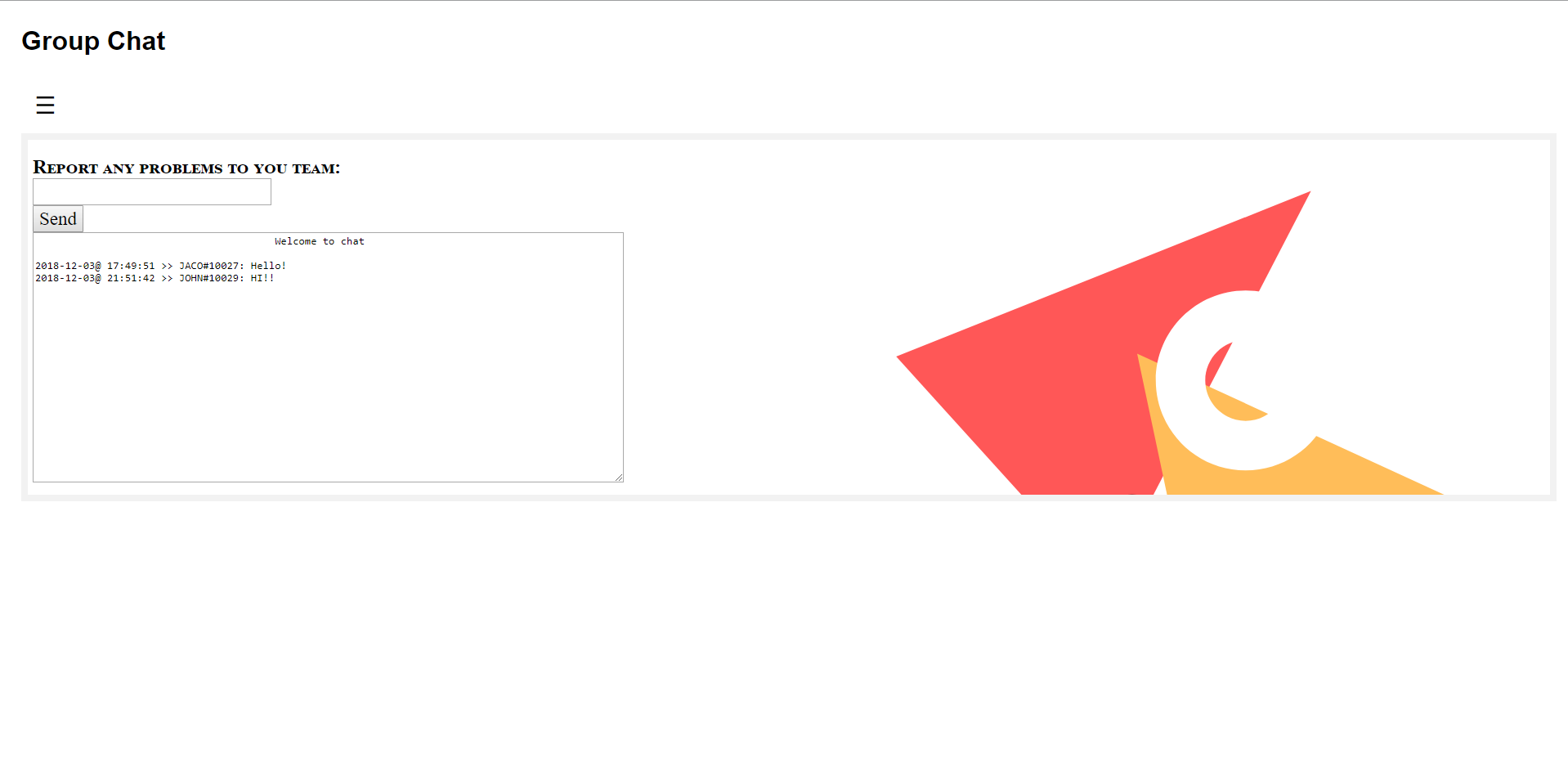


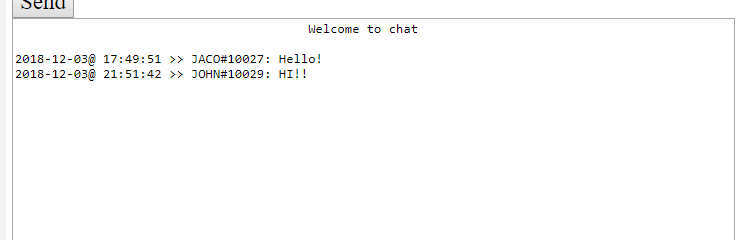
* Test.txt has 2 versions of different times and sizes (thus the file was modified)

## Sidepanel Report Issues

* Live chat between group member of the same repository

Figure 6‑9: Chat between Jaco and John





## Sidepanel Settings

* In setting you can:
  + Join a group by typing the repositories name under “Join a project”
  + Accept or reject invite (if any)
  + Change your email or password, each can be done separately
  + Delete current account

Figure 6‑10: Settings page

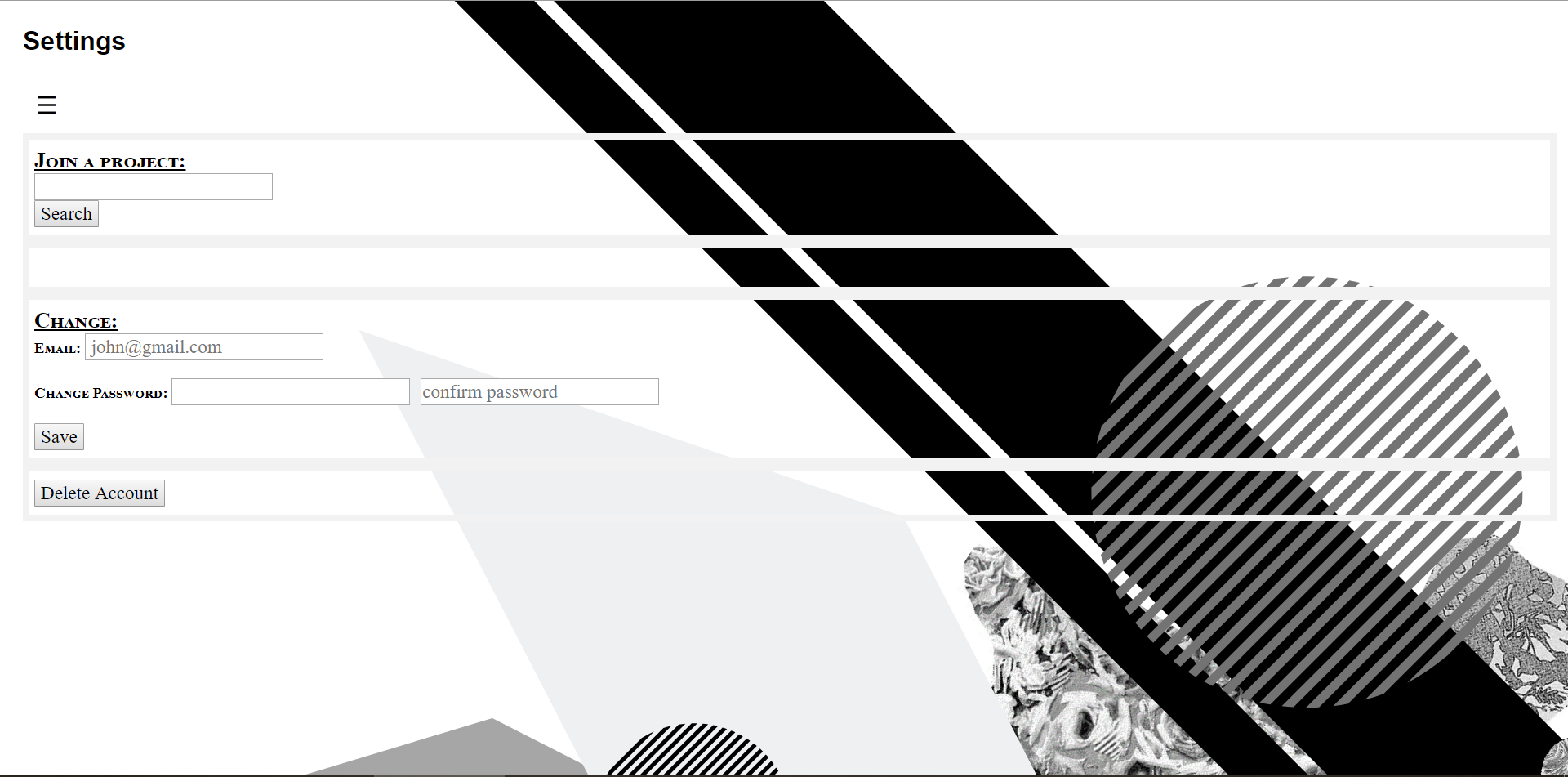


Figure 6‑11: Sent an invite

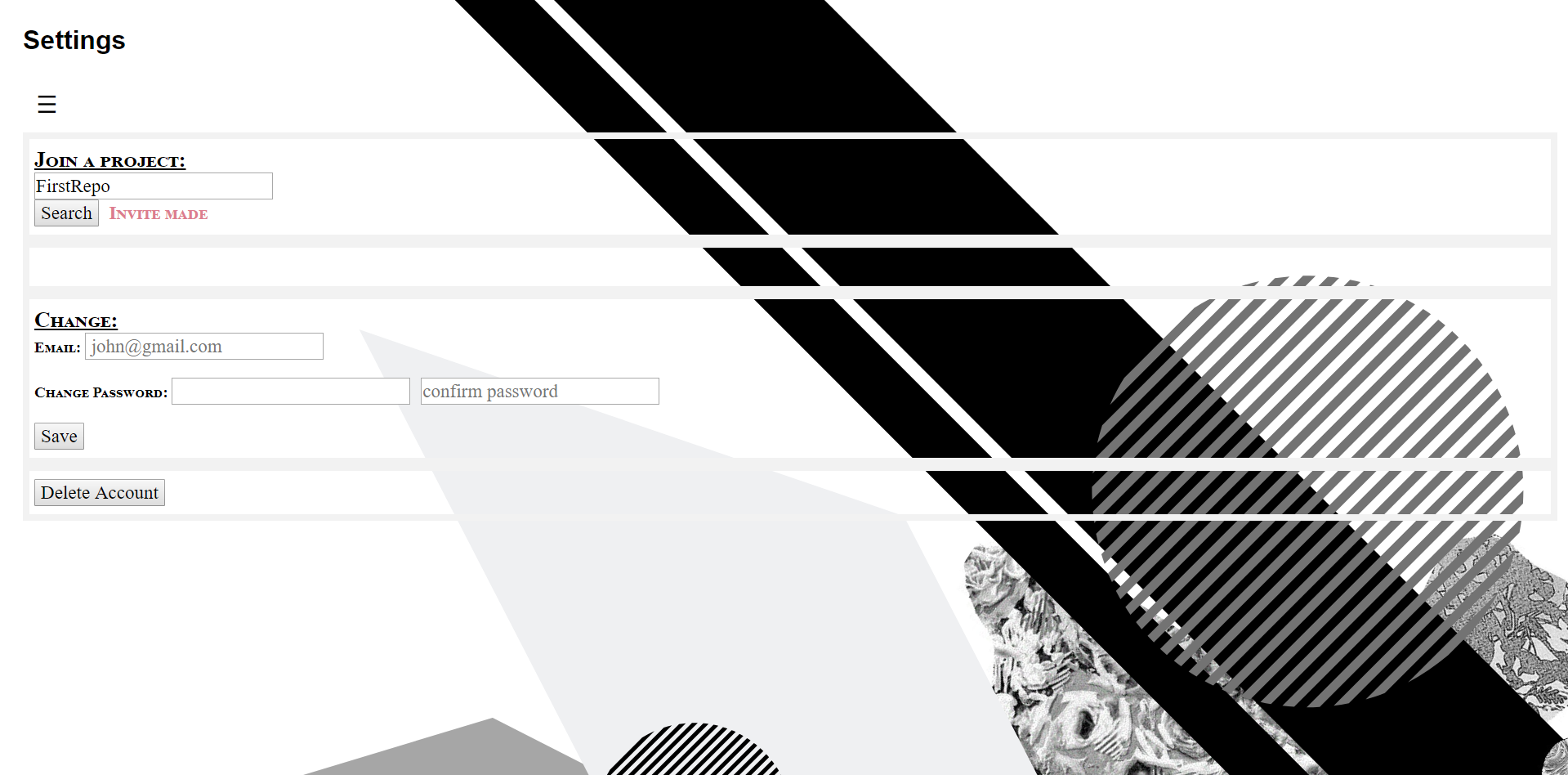
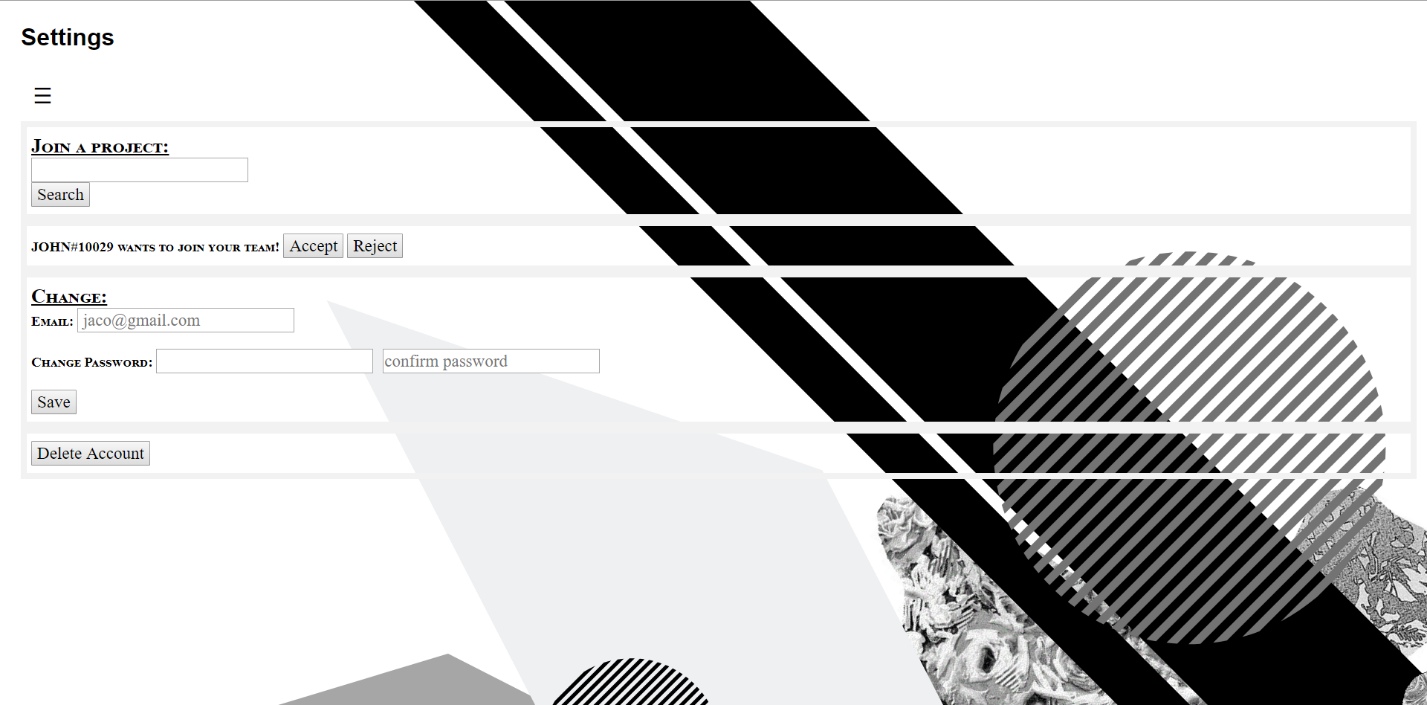


Figure 6‑12: Invite request



## Sidepanel Logout

* Clicking the logout link will take you back to the HomePage