Milestone1 Deliverable (pts. 10.00)

Instructions

1. Checkout Milestone1 branch
2. Create a milestone1.md file in your Project folder
3. Git add/commit/push this empty file to Milestone1 (you'll need the link later)
4. Fill in the deliverable items
5. Ensure your images display correctly in the sample markdown at the bottom
    - NOTE: You may want to try to capture as much checklist evidence in your screenshots as possible, you do not need individual screenshots and are recommended to combine things when possible. Also, some screenshots may be reused if applicable.
6. Save the submission items
7. Copy/paste the markdown from the "Copy markdown to clipboard link" or via the download button
8. Paste the code into the milestone1.md file or overwrite the file
9. Git add/commit/push the md file to Milestone1
10. Double check the images load when viewing the markdown file (points will be lost for invalid/non-loading images)
11. Make a pull request from Milestone1 to dev and merge it (resolve any conflicts)
    - Make sure everything looks ok on NJIT's webserver dev
12. Make a pull request from dev to prod (resolve any conflicts)
    - Make sure everything looks ok on NJIT's webserver prod
13. Submit the direct link from github prod branch to the milestone1.md file (no other links will be accepted and will result in 0)

Desired Branch Name

Milestone 1

Deliverables

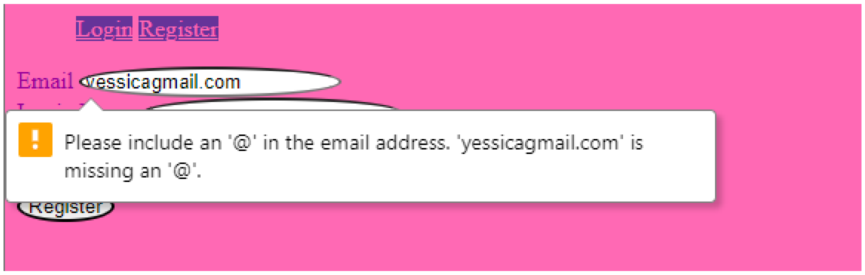
Deliverable 1: Feature: User will be able to register a new account (pts. 1)

SubTask 1: Add one or more **screenshots** of the application showing the form and validation errors per the feature requirements

The checkboxes are for your own tracking

#	Points	Details
[X] #1	1	Show invalid email validation
[X] #2	1	Show invalid password validation
[X] #3	1	Show passwords not match validation
[ ] #4	1	Show email not available validation (already registered)
[ ] #5	1	Show Login Name not available validation (logName is taken)
[X] #6	1	Show the form with valid data filled in before the form is submitted

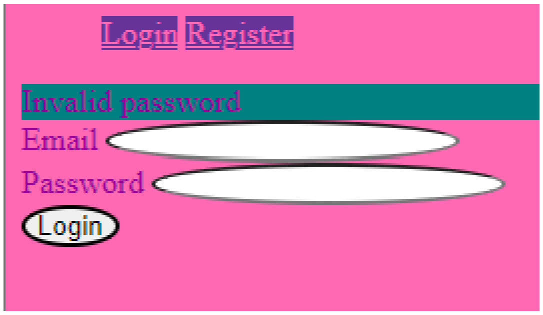
Make sure if the screenshot contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



Email yessicagmail.com

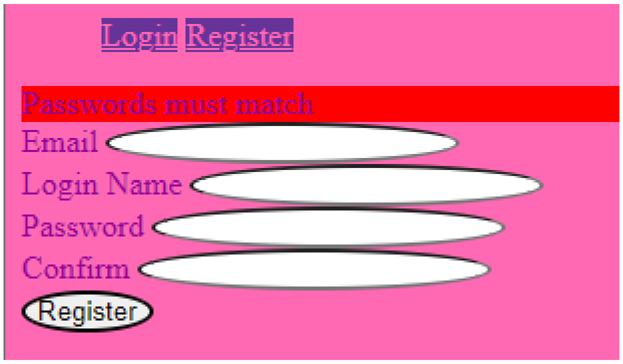
Please include an '@' in the email address. 'yessicagmail.com' is missing an '@'.

Invalid email validation



Invalid password

Invalid password validation



Passwords must match

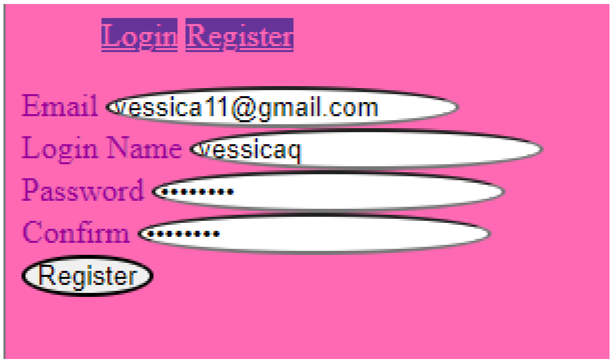
Email

Login Name

Password

Confirm

Passwords must match validation



Email yessica11@gmail.com

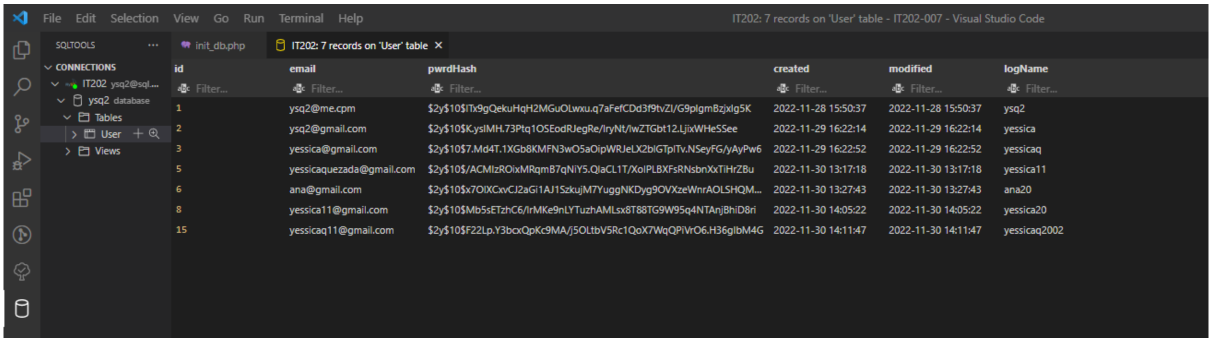
Login Name yessicaq

Password ••••••••

Confirm ••••••••

Form filled with valid data

SubTask 2: Add **screenshot** of the User table with data in it. The valid user data from Task 1 should be present in this **screenshot**. Clearly highlight which row it is.



id	email	pwrdHash	created	modified	logName
Filter...	Filter...	Filter...	Filter...	Filter...	Filter...
1	ysq2@me.cpm	$2y$10$ITx9gQekuHqH2MGuOLwxu.q7aFefCDd3f9tvZI/G9pIgmBzjxIg5K	2022-11-28 15:50:37	2022-11-28 15:50:37	ysq2
2	ysq2@gmail.com	$2y$10$K.ysIMH.73Ptq1OSEodRJegRe/IryNt/IwZTGbt12.LjixWHeSSee	2022-11-29 16:22:14	2022-11-29 16:22:14	yessica
3	yessica@gmail.com	$2y$10$7.Md4T.1XGb8KMFN3wO5aOipWRJeLX2bIGTpITv.NSeyFG/yAyPw6	2022-11-29 16:22:52	2022-11-29 16:22:52	yessicaq
5	yessicaquezada@gmail.com	$2y$10$/ACMIzROixMRqmB7qNiY5.QIaCL1T/XoIPLBXFsRNsbnXxTiHrZBu	2022-11-30 13:17:18	2022-11-30 13:17:18	yessica11
6	ana@gmail.com	$2y$10$x7OIXCxvCJ2aGi1AJ1SzkujM7YuggNKDyg9OVXzeWnrAOLSHQM...	2022-11-30 13:27:43	2022-11-30 13:27:43	ana20
8	yessica11@gmail.com	$2y$10$Mb5sETzhC6/IrMKe9nLYTuzhAMLsx8T88TG9W95q4NTAnjBhiD8ri	2022-11-30 14:05:22	2022-11-30 14:05:22	yessica20
15	yessicaq11@gmail.com	$2y$10$F22Lp.Y3bcxQpKc9MA/j5OLtbV5Rc1QoX7WqQPiVrO6.H36gIbM4G	2022-11-30 14:11:47	2022-11-30 14:11:47	yessicaq2002

SubTask 3: Add the related pull request(s) for this feature

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/19**

SubTask 4: Explain briefly how the process/code works (to register)

#	Points	Details
[X] #1	1	Explain how the form is handled and behaves
[X] #2	1	Explain the validation logic (frontend and backend)
[X] #3	1	Explain how the password is handled
[X] #4	1	Explain how the DB is utilized

#1: The form is handled based on what the user inputs. It'll behave based on the information that is being put in.
#2: With the user registering their information, their data is stored and compiled. If they were to login in, the credentials that they filled out originally has to be used in order for them to gain access to the site.
#3: In the original code, the password is hashed so the customer's information can't be accessed easily.
#4: DB is based off SQL and it's utilized to save and manipulate information.

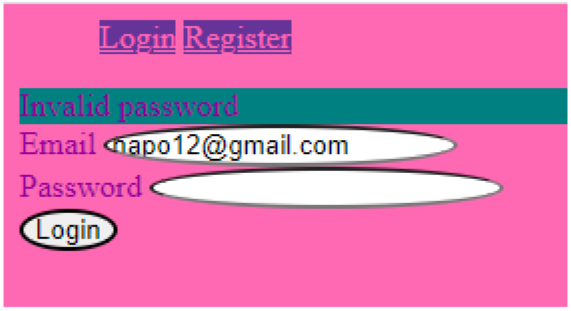
Deliverable 2: Feature: User will be able to login to their account (pts. 1)

SubTask 1: Add one or more **screenshots** of the application showing the form and validation errors per the feature requirements

The checkboxes are for your own tracking

#	Points	Details
[X] #1	1	Show password mismatch validation (doesn't match what's in the DB)
[X] #2	1	Show validation based on non-existing user

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.

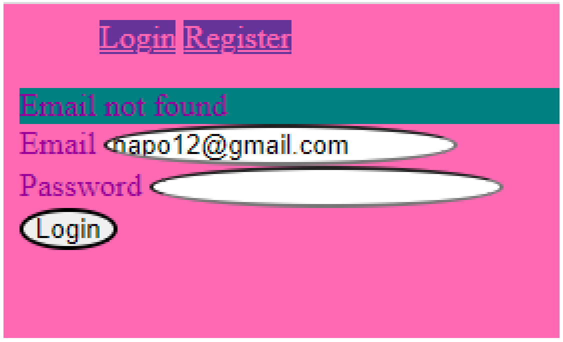


Invalid password

Email napo12@gmail.com

Password

Mismatch validation



Email not found

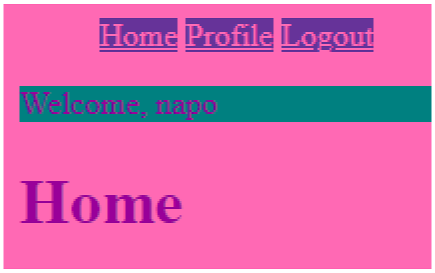
Email napo12@gmail.com

Password

Non-based user

SubTask 2: Add a **screenshot** of successful login

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



Welcome, napo

Home

Successful login

SubTask 3: Add the related pull request(s) for this feature

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/20**

SubTask 4: Explain briefly how the process/code works (to login)

The checkboxes are for your own tracking

#	**Points**	**Details**
[X] #1	1	Explain how the form is handled and behaves
[X] #2	1	Explain the validation logic (frontend and backend)
[X] #3	1	Explain how the password is handled
[X] #4	1	Explain how the DB is utilized

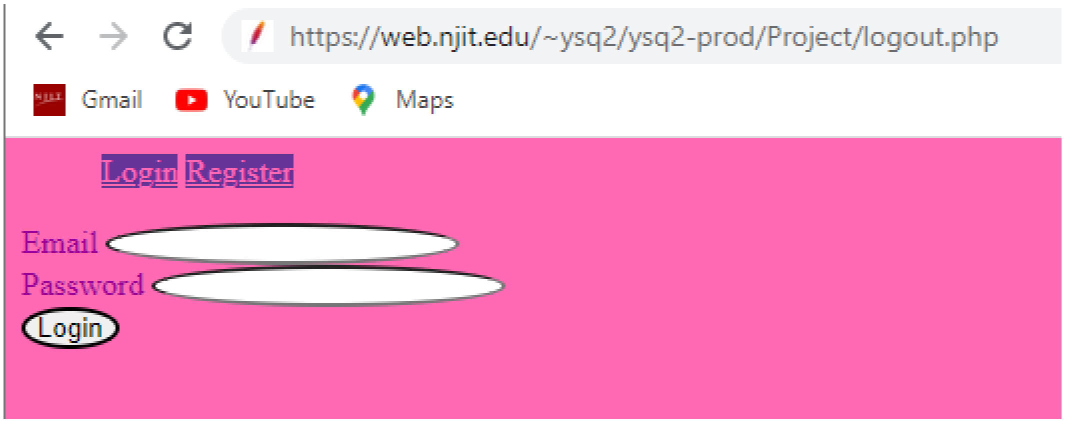
#1: With the user inputting their personal login information, if the email that was stored doesn't match with the input, it will notify the user that the email doesn't exist.
#2: If the user puts the correct information that matches with the information that is stored in the table, it will allow them to login, as well register their information.

#3: In the actual code, it shows that the password is hashed so no one can get access to the information that the user inputs.
#4: DB is utilized to save the information and store it, but also protecting it by notifying users that one is user is already taken.

Deliverable 3: Feature: Users will be able to logout (pts. 1)

SubTask 1: Add a **screenshot** showing the successful logout message. Message should show something about being logged out

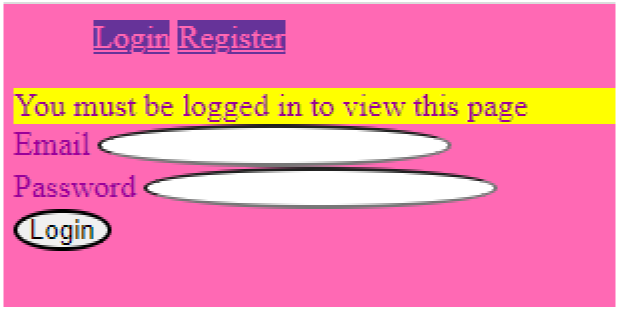
Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



When you log out, it'll take you back to the login page

SubTask 2: Add a **screenshot** showing the logged out user can't access a login-protected page. Message should show something about not being logged in

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



You must be logged in to view this page

SubTask 3: Add the related pull request(s) for this feature

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/24**

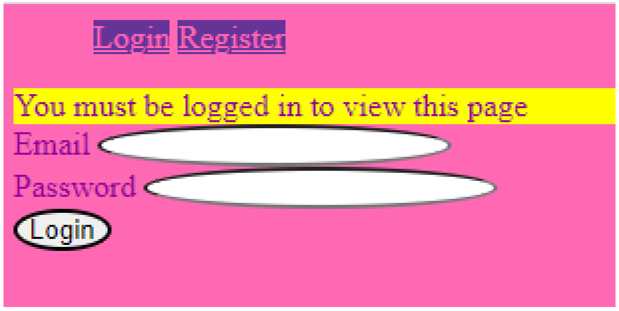
SubTask 4: Explain briefly how the process/code works. Describe the session logic and how it's used for login

Explanation: Let’s just say if the user tries to return to the main page, and tries to go back, the website will notify them saying that they have to be logged in, in order to get access to the website.

Deliverable 4: Feature: Basic Security Rules and Roles Implemented (pts. 1)

SubTask 1: Add a **screenshot** showing the logged out user can't access a login-protected page (may be the same as similar request). Message should show something about not being logged in

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



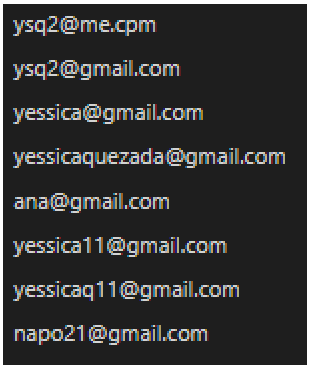
You must be logged in to view this page

SubTask 2: Add a **screenshot** showing a user without an appropriate role can't access the role-protected page. Message should show something about not having proper role or permission (i.e., different than the not logged in message)

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.

SubTask 3: Add a **screenshot** showing of the Roles table with valid data. Must have at least one valid record from the lessons (i.e., Admin)

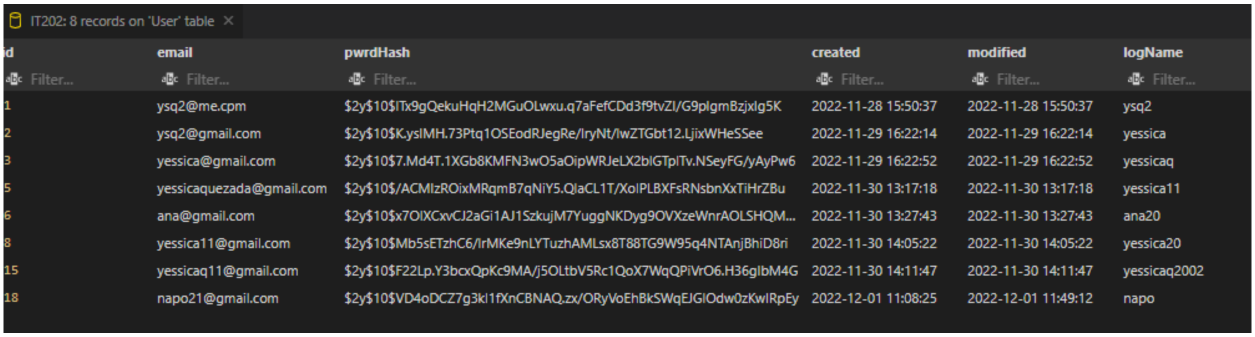
Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



ysq2@me.cpm
ysq2@gmail.com
yessica@gmail.com
yessicaquezada@gmail.com
ana@gmail.com
yessica11@gmail.com
yessicaq11@gmail.com
napo21@gmail.com

SubTask 4: Add a **screenshot** showing the UserRole table with valid data. Caption which is your administrator

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



IT202: 8 records on 'User' table

id	email	pwrdHash	created	modified	logName
Filter...	Filter...	Filter...	Filter...	Filter...	Filter...
1	ysq2@me.cpm	$2y$10$ITx9gQekuHqH2MGuOLwxu.q7aFefCDd3f9tvZI/G9plgmBzjxlg5K	2022-11-28 15:50:37	2022-11-28 15:50:37	ysq2
2	ysq2@gmail.com	$2y$10$K.ysIMH.73Ptq1OSEodRJegRe/IryNt/lwZTGbt12.LjixWHeSSee	2022-11-29 16:22:14	2022-11-29 16:22:14	yessica
3	yessica@gmail.com	$2y$10$7.Md4T.1XGb8KMFN3wO5aOipWRJeLX2blGTplTv.NSeyFG/yAyPw6	2022-11-29 16:22:52	2022-11-29 16:22:52	yessicaq
5	yessicaquezada@gmail.com	$2y$10$/ACMIzROixMRqmB7qNiY5.QlaCL1T/XoIPLBXFsRNsbnXxTiHrZBu	2022-11-30 13:17:18	2022-11-30 13:17:18	yessica11
6	ana@gmail.com	$2y$10$x7OlXCxvCJ2aGi1AJ1SzkujM7YuggNKDyg9OVXzeWnrAOLSHQM...	2022-11-30 13:27:43	2022-11-30 13:27:43	ana20
8	yessica11@gmail.com	$2y$10$Mb5sETzhC6/IrMKe9nLYTuzhAMLsx8T88TG9W95q4NTAnjBhiD8ri	2022-11-30 14:05:22	2022-11-30 14:05:22	yessica20
15	yessicaq11@gmail.com	$2y$10$F22Lp.Y3bcxQpKc9MA/j5OLtbV5Rc1QoX7WqQPiVrO6.H36glbM4G	2022-11-30 14:11:47	2022-11-30 14:11:47	yessicaq2002
18	napo21@gmail.com	$2y$10$VD4oDCZ7g3kl1fXnCBNAQ.zx/ORyVoEhBkSWqEJGlOdw0zKwlRpEy	2022-12-01 11:08:25	2022-12-01 11:49:12	napo

SubTask 5: Add the related pull request(s) for this feature

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/20**

SubTask 6: Explain briefly how the process/code works for login-protected pages. Explain how the session is used and any relevant helpers

Explanation: When it comes to login protected pages, it needs to be stored in a safe place where the website will remember the login credentials. In the table provided at top, it shows the usernames, email, and hashed passwords that was used.

SubTask 7: Explain briefly how the process/code works for role-protected pages. Explain how the session is used and any relevant helpers

Explanation: When it comes to role protected pages, it'll give certain people admins. Meaning, that one admin will have access to the entire site, while others wont have that option.

Deliverable 5: Feature: Site should have basic styles/theme applied; everything should be styled (pts. 1)

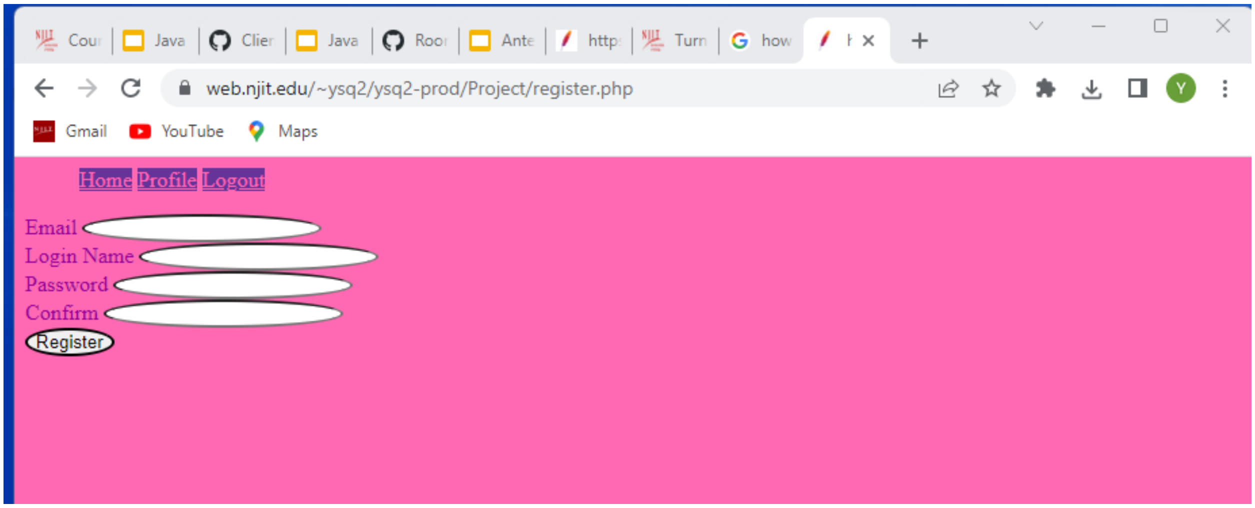
SubTask 1: Add a **screenshot** showing the successful logout message

The checkboxes are for your own tracking

#	Points	Details
[X] #1	1	Navigation should be styled
[X] #2	1	Forms should be styled
[ ] #3	1	UI should be clean and not have my "hideous" example CSS

#	Points	Details
[X] #4	1	Data output should be in a clean manner (i.e., table, row/cols, list groups, etc). Basically not exactly like dumped plaintext

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



SubTask 2: Add the related pull request(s) for this feature

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/29**

SubTask 3: Briefly explain your CSS at a high level. Highlight the basic styling you chose for the general elements like nav, forms, etc.

Explanation: In my CSS file, its implemented to specific messages only. Not all messages will have the same color. If the message connects with the input, it'll appear.

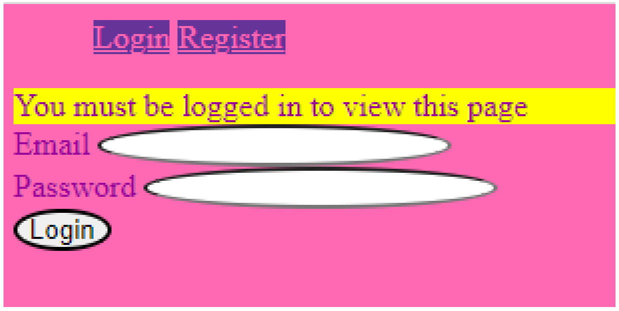
Deliverable 6: Feature: Any output messages/errors should be "user friendly" (pts. 1)

SubTask 1: Add **screenshots** showing of some examples of errors/messages

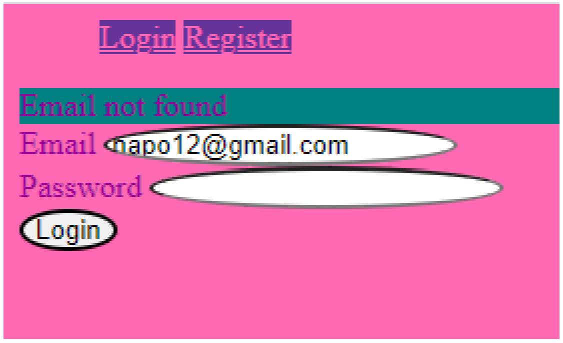
The checkboxes are for your own tracking

#	Points	Details
[X] #1	1	This can include previous screenshots of protected pages and/or validation messages.
[X] #2	1	Show at least 3 different "error" messages

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.

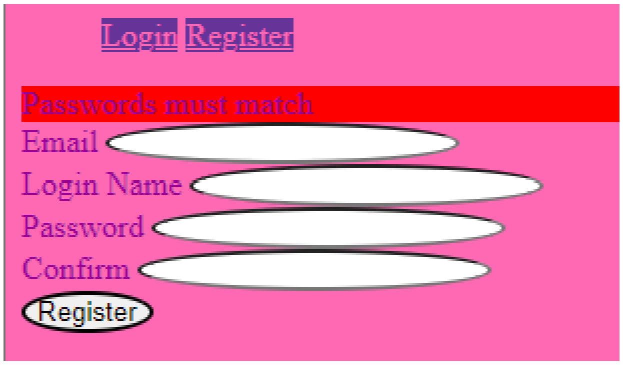


You must be logged in to view this page



Email not found

Email napo12@gmail.com



Passwords must match

SubTask 2: Add a related pull request(s)

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/24**

SubTask 3: Explain briefly how you made messages user friendly. Describe what we've been doing to handle techical messages and converting them to being friendly to our users

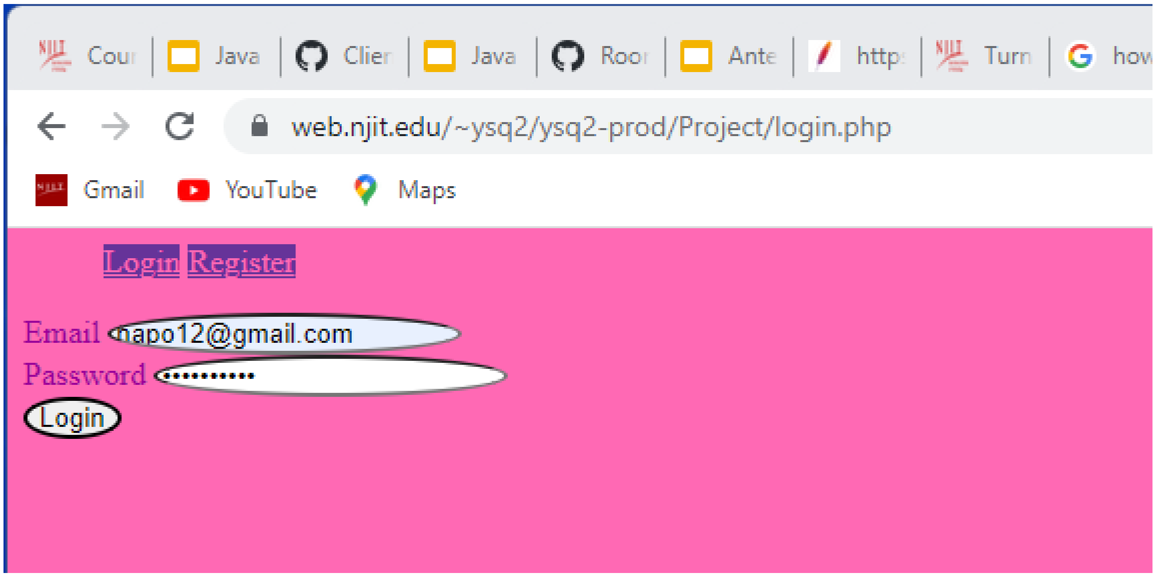
Explanation: I used a CSS code in order to create user friendly. Depending on what message pops up, its highlighted with specific colors to match the message.

Deliverable 7: Feature: Users will be able to see their profile (pts. 1)

<>

SubTask 1: Add a **screenshot** showing the successful logout message. Email and Login Name should prefill properly.

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



Email napo12@gmail.com

Password •••••••••••

SubTask 2: Add the related pull request(s) for this feature

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/26**

SubTask 3: Explain briefly how the process/code works (view only). Only talk about how the data is retrieved about the user and displayed in the form

Explanation: In this logout page, it was programmed to automatically go back to the login php.

Deliverable 8: Feature: User will be able to edit their profile (pts. 1)

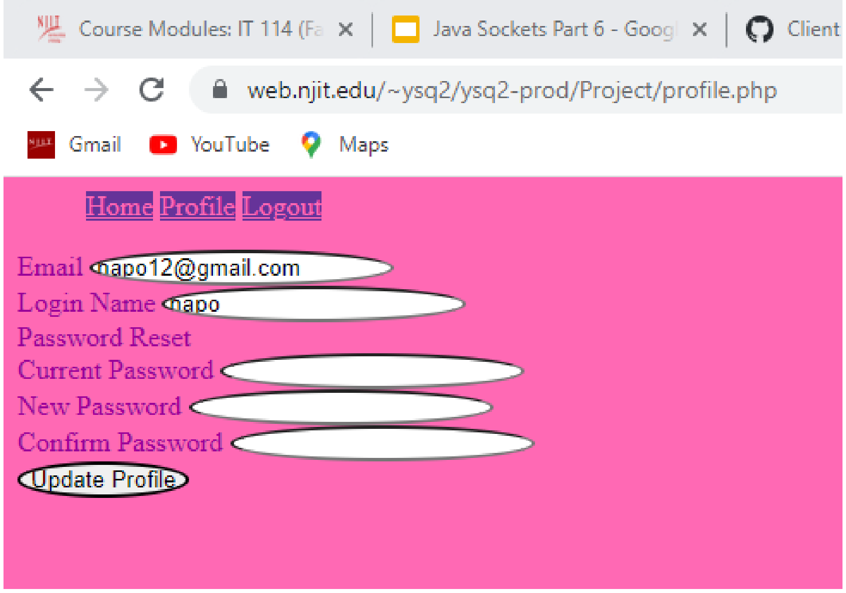
SubTask 1: Add **screenshots** of the User Profile page validation messages and success messages

The checkboxes are for your own tracking

#	Points	Details
[ ] #1	1	Show Login Name validation message
[X] #2	1	Show email validation message
[X] #3	1	Show password validation message

#	Points	Details
[X] #4	1	Show password mismatch message
[ ] #5	1	Show email/Login Name already in use message

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.



Course Modules: IT 114 (Fa
Java Sockets Part 6 - Googl
Client
web.njit.edu/~ysq2/ysq2-prod/Project/profile.php
Gmail
YouTube
Maps
Home Profile Logout

Email napo12@gmail.com

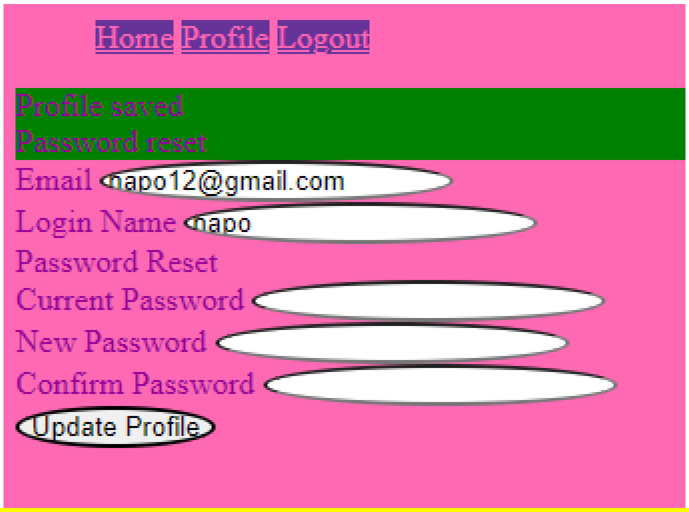
Login Name napo

Password Reset

Current Password

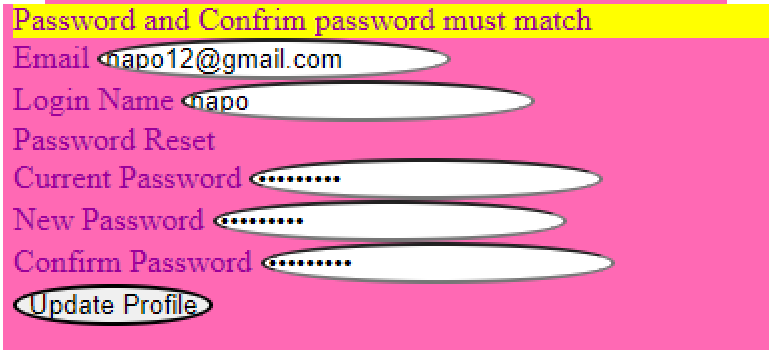
New Password

Confirm Password



Profile saved
Password reset

Email napo12@gmail.com
Login Name napo
Password Reset
Current Password
New Password
Confirm Password



Password and Confrim password must match

Email napo12@gmail.com
Login Name napo
Password Reset
Current Password •••••••••
New Password •••••••••
Confirm Password •••••••••

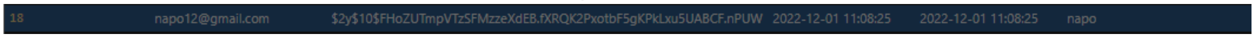
SubTask 2: Add before and after **screenshots** showing the User table when a user edits their profile

The checkboxes are for your own tracking

#	Points	Details
[X] #1	1	There should be two screenshots
[X] #2	1	It should be highlighted or note which record changes between both screenshots along with what changed

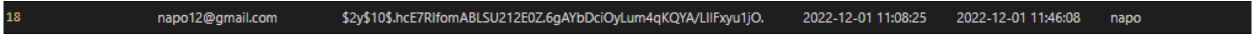
Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.

Before password was changed



```
18    napo12@gmail.com    $2y$10$FHoZUTmpVTzSFMzzeXdEB.fXRQK2PxotbF5gKPkLxu5UABCF.nPUW  2022-12-01 11:08:25    2022-12-01 11:08:25    napo
```

After password was changed



```
18    napo12@gmail.com    $2y$10$.hcE7RIfomABLSU212E0Z.6gAYbDciOyLum4qKQYA/LIIFxyu1jO.    2022-12-01 11:08:25    2022-12-01 11:46:08    napo
```

SubTask 3: Add the related pull request(s) for this feature

Required url pattern: https?://github.com/(?:[^/.]+/)+(?:pull/.+)
URL(s): **https://github.com/yessicaquezada/IT202-007/pull/26**

SubTask 4: Explain briefly how the process/code works (edit only). Talk about how the edit logic is handled for updating email, logName, and password (don't forget to briefly mention the validation)

Explanation: The edit features works but updating the users credentials. On the website itself, it lets users update their information. Once their credentials is updated, on the table’s end it’ll show what was updated. In this case, the password was updated, and you can see that the has and date time are different.

Deliverable 9: Issues and Project Board (pts. 2)

SubTask 1: Add **screenshots** showing which issues are done/closed (project board) Incomplete Issues should not be closed

The checkboxes are for your own tracking

#	Points	Details
[X] #1	1	At least one issue item per Milestone feature
[X] #2	1	Show from the Project Board perspective (may need multiple screenshots)

Make sure if the **screenshot** contains code that you have a relevant comment with your ucid, date, and explanation of what you're attempting, if not max grade for this item is 75%.

SubTask 2: Include a direct link to your Project Board. URL should end in /project/# (likely 1, but the number isn't important)

URL:
https://github.com/users/yessicaquezada/projects/2/views/1?layout=board

SubTask 3: Prod Application Link to Login Page. Link must be from the NJIT webserver prod folder to your application's login

Required url pattern: https?://web.njit.edu/.+-prod.*
URL: **https://web.njit.edu/~ysq2/ysq2-prod/Project/login.php**