

ITP 303 Final Project Summary

INSTRUCTIONS: Type out your answer directly below each question. If a question does not apply to your project, type in N/A. When finished, save this file as a PDF and upload it to the itpwebdev server via FileZilla. Then, add a link to this PDF in your student_page.html. Label it "Final Project Summary."

BASICS

Your name?

Hyun-Joon Yang

What is the topic of your final project?

I would like to make a website that allows users to rate the freshman dorms at USC. It would have basic information about each of the dorms as well as reviews created by users who have lived in the respective dorm to help the incoming freshmen decide where to live.

Who is the intended audience?

It would mainly be targeted for incoming freshman at USC, although its content would be generated by the residents of the dorm.

Provide a brief summary of the functionalities of your project.

From the homepage user can either browse the USC dorms using the browse page or search for a certain dorm meeting a criterion using the search page. On the search page, a user can search by name, location, rating, price, and room type. Once the user has picked a dorm to view, he/she has the option to just read about the dorm including its reviews or create a review. The information includes an about section as well as an overview which includes its overall rating (average of all reviews). Creating a review requires logging in, and if the user has not done so, the review button redirects to the log in page. There, the user can choose to sign up to become a registered member. A registered user can also update his/her own review.

FRONT-END

What is the total page count?

There are 8 main pages (index, dorm, browse, login, register, review, search, results) although there are other PHP files that do other backend functionalities (validate, redirect, etc).

List the names of any external stylesheet used in this project below.

There is only 1 stylesheet (css/style.css).

List any CSS libraries/frameworks used in this project (e.g. Bootstrap) below.

Bootstrap was used in the project.

List any JavaScript libraries/frameworks used in this project below.

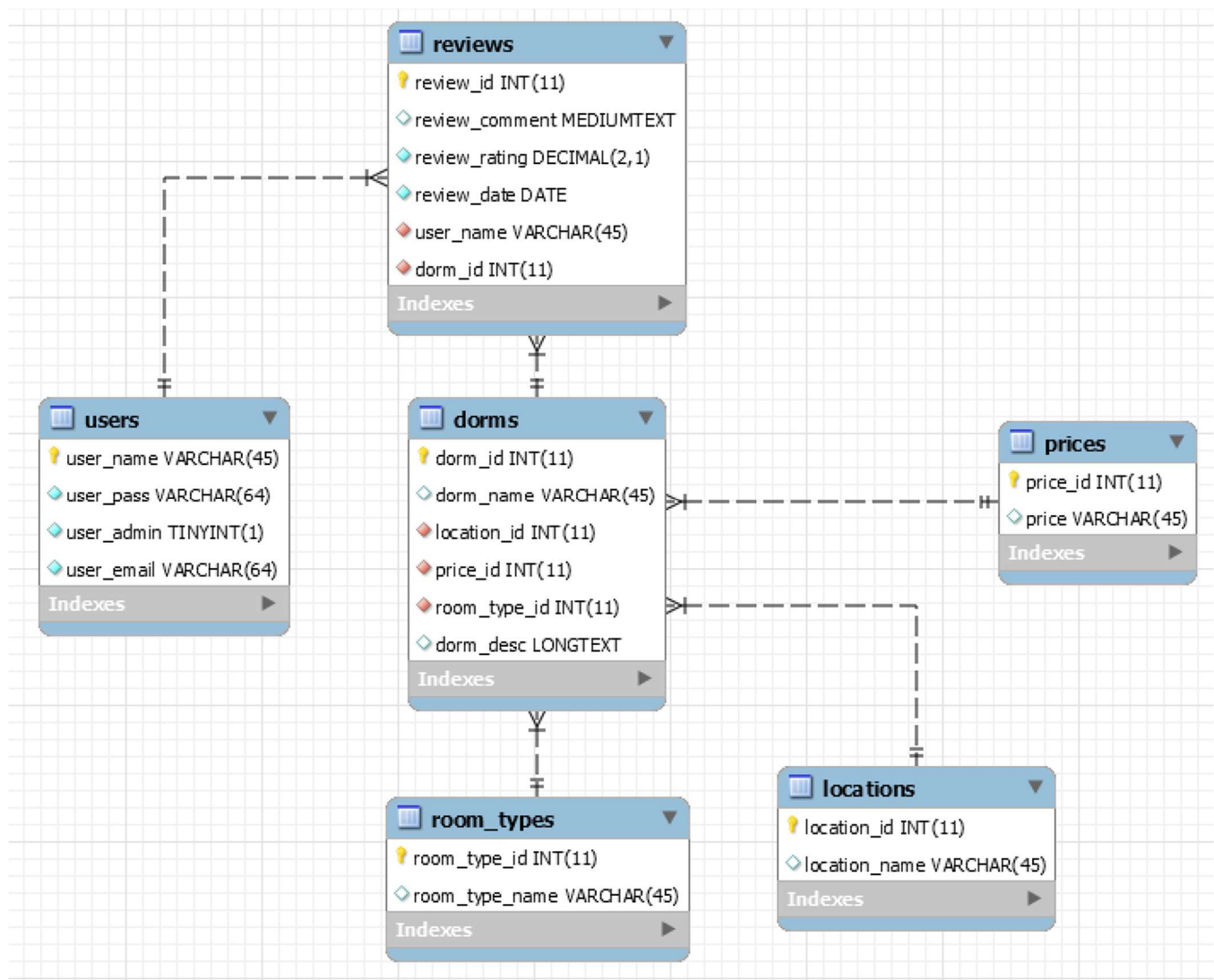
N/A

How does your project meet the Interactivity requirement?

In the "dorm.php" page, you can hover on the "About" section which will show the option to show more / less. Once clicked it will toggle the short / long version. Also, when reviewing a dorm in "review.php" page, the rating score will change depending on the slider.

BACK-END

Attach a screenshot of the final database diagram (just the diagram, do not include any records) below.



Where in your project do you insert a record to the database (the 'C' in CRUD)?

A user can register in "register.php" page which will create a new record in the "users" table of the database using "registerd.php". Also, a user can review a dorm in the "review.php" page which will create a new record in "reviews" table of the database using "review_confirmation.php"

Where in your project do you search and display record(s) from the database (the 'R' in CRUD)?

The "browse.php", "review.php", "search.php", "dorm.php", "results.php" pages all retrieve records from the database whether it is the list of dorms and their information or list of reviews.

Where in your project do you update and existing record(s) the database (the 'U' in CRUD)?

A user can update his/her own review while an admin can update any reviews in the "dorm.php" page. Clicking on "update" takes the user to "review.php" page which updates the record using "review_confirmation.php".

Where in your project do you delete existing record(s) from the database (the 'D' in CRUD)?

A user can delete his/her own review while an admin can delete any reviews in the "dorm.php" page. Clicking on "delete" uses "delete_confirmation.php" to delete the record from the database.

MISC

What two "extra" requirements did you implement and where can they be found?

Different levels of permission were implemented and it is apparent in the "dorm.php?dorm_id=1" (McCarthy Honors College) page. An unregistered user can only view reviews made by others. A registered user who has logged in can create, update, or delete his/her own review. An admin can create, update, or delete any reviews.

Pagination was implemented in the "dorm.php" page. Under the "Reviews" section of "dorm.php?dorm_id=1" (McCarthy Honors College) page, the reviews are divided into pages with each having up to 5 reviews.

If you used any APIs or JS plugins/frameworks as one of the extra requirements, list the name and a link to their documentation.

N/A

If your project requires any admin credentials (i.e. only admin users can access a certain page), list the credentials below.

Admin:

Username: admin

Password: admin

Registered User:

Username: user1

Password: thisisuser1

Provide any other information that you think the grader/instructor should know when grading your final project below.

SQL error redirects the user to an error page. Passwords are stored using hashing. SQL queries are sanitized. Invalid query strings (especially in the URL) redirects the user to an error page. The rating in the overview section of each dorm is an average of all the reviews. Clicking on review when not logged in redirects to log in page. Update / Delete button for reviews not displayed for users without permission.