Assignment 3 (HF, major subject)

Due: Wed 16.11.2016; 20:00h (1 Week)

Goals

After doing these exercises,

- You know your way around MySQL
- You can store data to databases with PHP scripts
- You know how to secure user credentials

Task 1: Hangman Extension — Difficulty: Medium

In tutorial 03 you started to create the hangman game, where the player needs to guess a word letter by letter. If a letter is correct, it is revealed at the correct position of a word, all other letters are displayed as underscores.

Now add these two extensions:

A) Random words
   a. Use a dictionary of words that should be guessed instead of a static word. There are many word lists on GitHub that you can use, e.g. https://github.com/atebits/Words [1].
   b. Make sure to read the words from a separate file, instead of adding them all to your .php file.
   c. Make sure only words with at least 8 letters are considered.

B) High-Score
   a. Allow the player to submit a name after s/he won the game.
   b. Create a table in a database to store high-scores. Each high-score contains the player name, guessed word, number of attempts and the date.
   c. Show a table with the high-score below the hangman game area or on a separate page. *Note: if you used the code template from the tutorial you will need to adjust some CSS to make the content visible.*

Put all your source code files and resources in the folder ‘task1’.

[1] under CC0 license, which is good if you want to include it in public repositories

Task 2: Note-Taking WebApp Difficulty: Hard

This is a quite work-intensive task, so you might want to start early during the week. We can also help you solve the task during the code labs.

Take a peek at a web page that can store user notes, e.g. Google Keep. Your task is to create a similar note-taking web-app.[2]

Consider the following requirements:

- The user can **create** notes
  - A note consists of a title, a text and a user ID. Store each note in a separate row in the database.
  - Each note also has a unique ID that is invisible to the user (you can use the PRIMARY KEY AUTO INCREMENT feature of MySQL). If it's rendered, the note-ID becomes an attribute of the HTML element and can be used to identify the note, e.g. note-id="...". This allows you to update or delete the note later.
  - Ensure that SQL injections are not possible, when the user saves a note.
- The page **retrieves** the notes and displays them as cards.
- The user can **delete** notes either one by one or multiple notes at the same time.
- Optional: Allow the user to update the notes to fulfill the CRUD paradigm.
- Optional: The user needs to be able to **sign-up** and **log-in** afterwards.
  - Create a simple registration page that takes a user name and a password. Send the data to a registration handler that hashes the password and stores the username-password-combination into the database. Use a 'users' table for that.
  - The log-in happens on the landing page. Use PHP-sessions to save the state (`$_SESSION['loggedIn]'`).

Tips:

- Use a database handler class (see breakout session in tutorial 03) to encapsulate all transactions. You can even think of an authentication handler. Make use of object-oriented programming.
- It is okay to solve the task differently than what we propose above (e.g. not using checkboxes to delete a note).
- Be creative! Maybe the notes can have different background colors or predefined labels?
- If anything in the task description is unclear, please contact us.
- Remember, you write code humans – not for machines. Make sure to comment your code as much as reasonable.

Put all your source code files and resources in the folder 'task2'.

[2] For now, it is okay to stick to PHP and MySQL (no AJAX so far), but since the lecture already introduced jQuery, you are allowed to use it to manipulate the DOM or add animations as you like.

Screenshots

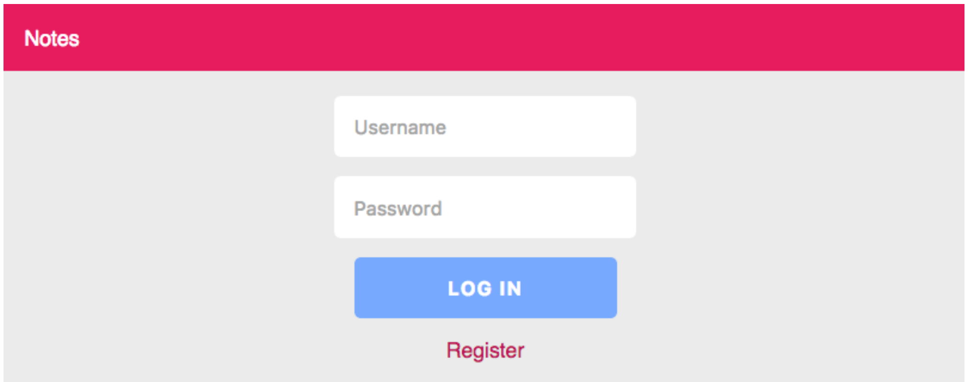
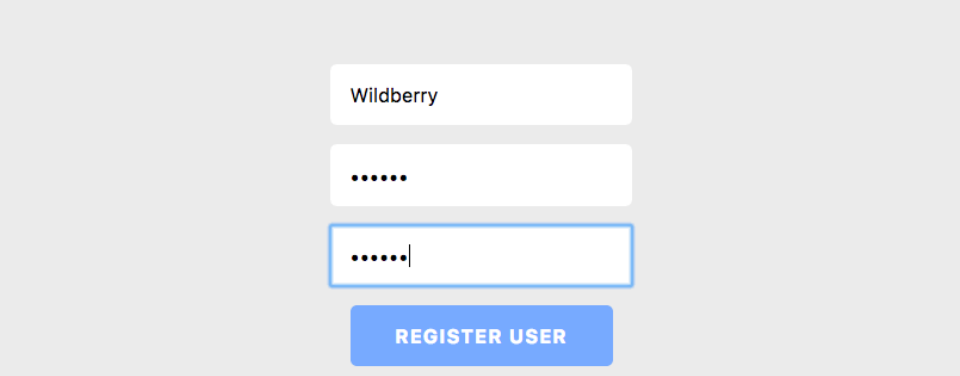


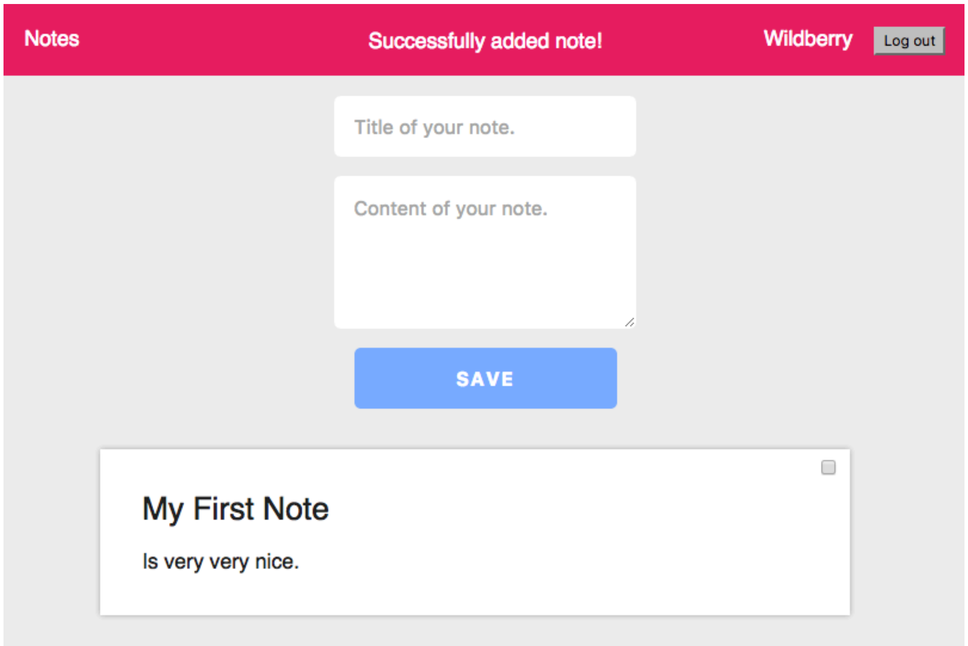
Figure 1 - Landing Page


Wildberry

••••••

••••••

Figure 2 – Registration Page



Successfully added note!

My First Note

Is very very nice.

Figure 3 - Main Page

Submission

Please turn in your solution via UniWorX. You can form groups of up to three people.

We encourage you to sign up for Slack! All you need is a CIP account and an email address that ends in "@cip.ifi.lmu.de". Ask us if you don't know how to get them.

If you have questions or comments before the submission, please contact one of the tutors. They are on Slack @tobi.seitz, @peterjuras and @thomas-weber. Remember, that they also want to enjoy their weekends ☺

It also makes sense to ask the question in our #mmn-ws1617 channel. Maybe fellow students can help or benefit from the answers, too!

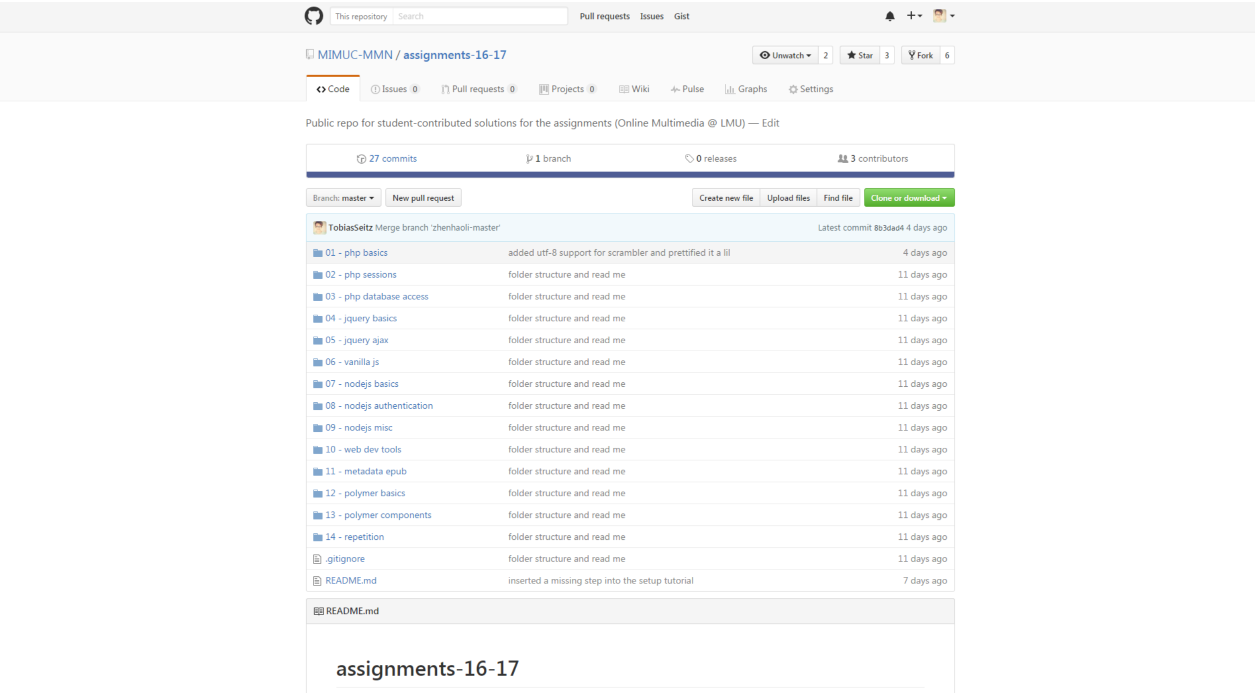
Let's collaborate on GitHub!

As we do not provide sample solutions, we encourage you to collaborate with all your peers on a sample solution for this assignment on GitHub.

We created a public repository for this purpose:
https://github.com/MIMUC-MMN/assignments-16-17

It includes a short "how to contribute" guide, to get you started.

The staff will always check what's in there and add comments or push updates.



MIMUC-MMN / assignments-16-17

Public repo for student-contributed solutions for the assignments (Online Multimedia @ LMU) — Edit

27 commits | 1 branch | 0 releases | 3 contributors

TobiasSeitz Merge branch 'zhenhaoli-master' Latest commit 8b3dad4 4 days ago

01 - php basics	added utf-8 support for scrambler and prettified it a lil	4 days ago
02 - php sessions	folder structure and read me	11 days ago
03 - php database access	folder structure and read me	11 days ago
04 - jquery basics	folder structure and read me	11 days ago
05 - jquery ajax	folder structure and read me	11 days ago
06 - vanilla js	folder structure and read me	11 days ago
07 - nodejs basics	folder structure and read me	11 days ago
08 - nodejs authentication	folder structure and read me	11 days ago
09 - nodejs misc	folder structure and read me	11 days ago
10 - web dev tools	folder structure and read me	11 days ago
11 - metadata epub	folder structure and read me	11 days ago
12 - polymer basics	folder structure and read me	11 days ago
13 - polymer components	folder structure and read me	11 days ago
14 - repetition	folder structure and read me	11 days ago
.gitignore	folder structure and read me	11 days ago
README.md	inserted a missing step into the setup tutorial	7 days ago

README.md

assignments-16-17