## Overview:

Create a single page web application that monitors a person's consumption of a set of items in a diary form. The list of items being monitored and the list of authorized users will both be fixed in a database. Only the diary entries will be dynamically added to the database.

The application flow shall be as follows:

1. User shall connect to the application/index.html.
   1. ALL html code shall be in this file
   2. all javascript shall be in a separate file.
   3. All css shall also be in a separate file
2. User shall be prompted to authenticate.
   1. I have created user entries in the user table for each class member. All password are the same "test"
   2. Using JSON the users credentials shall be sent to the rest api to obtain a token
3. Once properly authenticated the user shall be shown
   1. A summary of their Diary
   2. The last 20 entries of their diary
   3. A well formatted set of buttons allowing them to indicate they consumed one of the tracked items.
      * The items list is obtained from making an api call
      * upon clicking a button, a JSON call shall be made to api which will update their diary with the item and the date/time.
      * The page will then update its entries via javascript
4. misc:
   1. you must always use prepare statements where appropriate
5. REST API
   1. Get Token
      * Given user and password will get a token validating the user. If no user is present or the password does not match will return status will == "FAIL"
      * passwords are hashed using the php password\_hash function
      * url: rest.php/v1/user
      * method: post
      * json\_in:
        + user
        + password
      * json\_out
        + status: "OK" or "FAIL"
        + msg:
        + token: string
      * Test:
        + curl -X 'POST' -d '{"user":"test","password":"test"}' https://ceclnx01.cec.miamioh.edu/~campbest/cse383/finalProject/restFinal.php/v1/user
   2. Get list of items
      * Return the set of items we are tracking and their key
      * rest.php/v1/items
      * method: get
      * json\_in: none
      * json\_out:
        + status
        + msg
        + items[]
          - pk
          - item
      * test:
        + <https://ceclnx01.cec.miamioh.edu/~campbest/cse383/finalProject/restFinal.php/v1/items>
        + curl https://ceclnx01.cec.miamioh.edu/~campbest/cse383/finalProject/restFinal.php/v1/items
   3. Get Items user consumed
      * rest.php/items/token
      * Call gets the tracked items for a given user
      * limit to last 30 items
      * Method: GET
      * JSON Response:
        + status: OK or AUTH\_FAIL or FAIL
        + msg: text
        + items[]
          - pk
          - item
          - timestamp
      * test
        + <https://ceclnx01.cec.miamioh.edu/~campbest/cse383/finalProject/restFinal.php/v1/items/1db4342013a7c7793edd72c249893a6a095bca71>
   4. Get Summary of Items
      * rest.php/v1/itemsSummary/token
      * method: GET
      * json\_in: none
      * json\_out
        + status
        + msg
        + items[]
          - item
          - count
      * test
        + <https://ceclnx01.cec.miamioh.edu/~campbest/cse383/finalProject/restFinal.php/v1/itemsSummary/1db4342013a7c7793edd72c249893a6a095bca71>
   5. Update Items Consumed
      * rest.php/v1/items
      * Updates item as being consumed.
      * method: post
      * JSON IN
        + token: string token
        + ItemFK: <key>
      * JSON OUT
        + status: OK or AUTH\_FAIL or FAIL
        + msg: text
      * test
        + curl -X 'POST' -d '{"token":"1db4342013a7c7793edd72c249893a6a095bca71","itemFK":2}' https://ceclnx01.cec.miamioh.edu/~campbest/cse383/finalProject/restFinal.php/v1/items

Database:

* users: user table
  + pk
  + user
  + password
  + timestamp
* diary: Item Entries
  + pk
  + userFK -> foreign Key to user - not the user but the pk of the user
  + itemFK -> foreign Key to item. Not the item but the PK of the item
  + timestamp
* diaryItems: list of items
  + pk: int
  + item: tinytext
* tokens
  + pk
  + use - actual user string
  + token - token string created randomly
  + timestamp

You must use a php based datamodel file separate from your rest code.