**Local Web Product QR generator**

This web application is meant to be hosted locally and accessible via WiFi **only at the first line** of the chain (factories, production places etc).

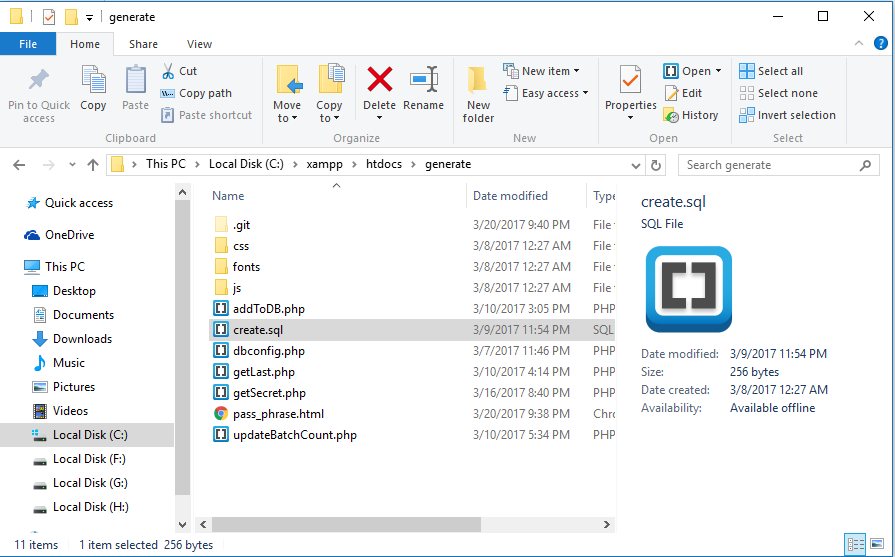
**Online Website URL:**

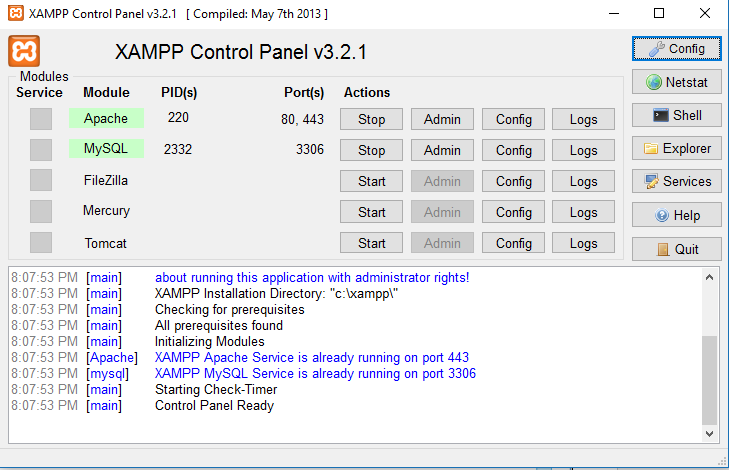
<http://pastureless-skin.000webhostapp.com/generate/pass_phrase.html>

**Requirements:**

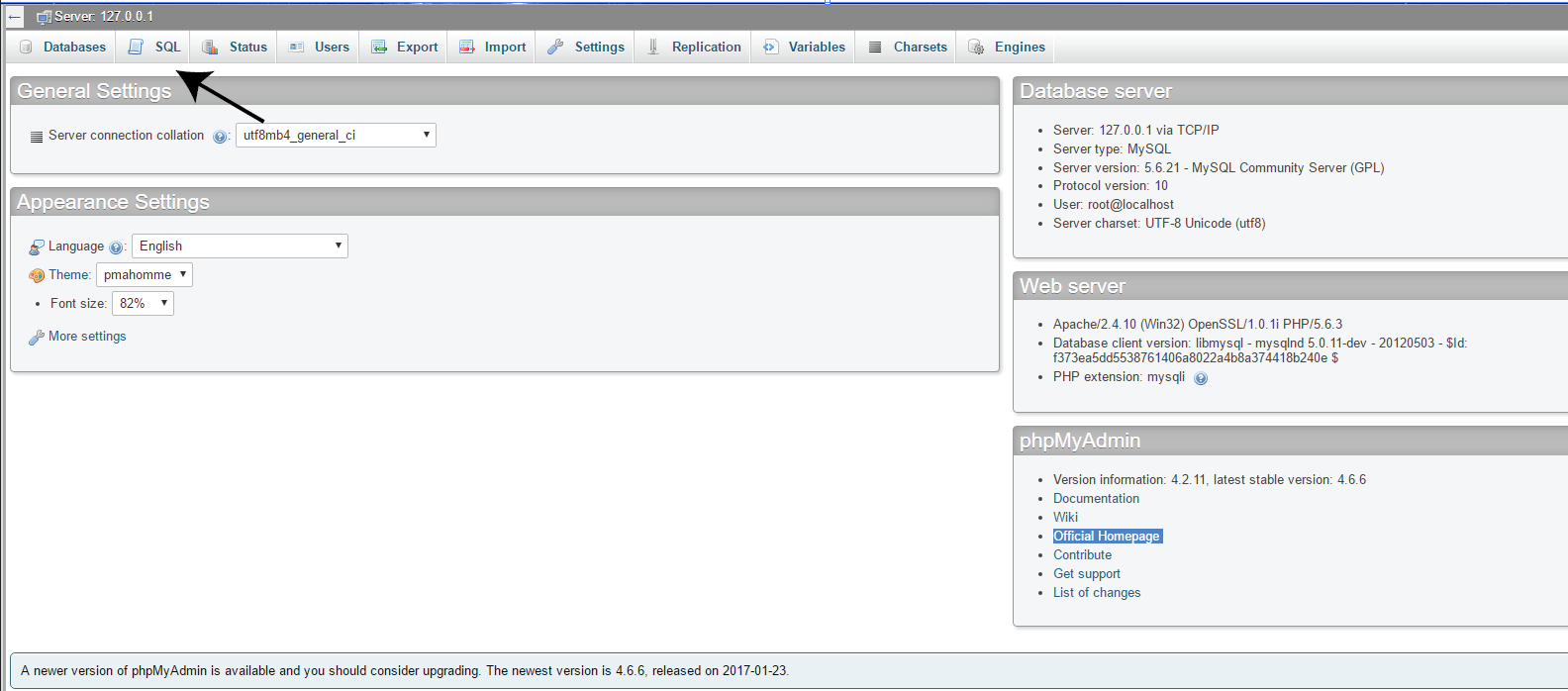
* A running XAMPP server (Apache/PhpMyAdmin).
* A web browser
* Internet connection , and is allowed to access NXT peer (<http://174.140.168.136:6876/>)
* SQL database created

**How to get the app running:**

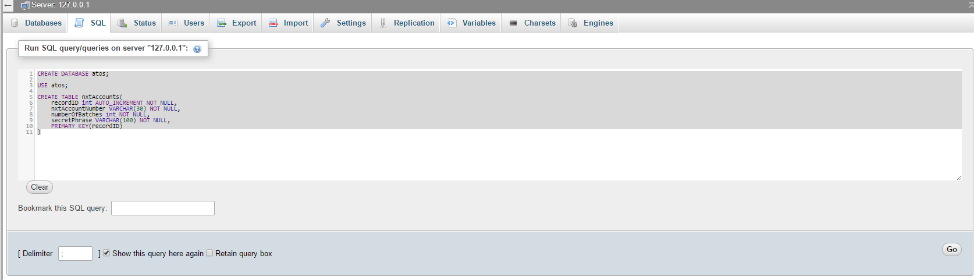
1. Have Xampp installed on a computer
2. Look for the folder with the name \_ from the attached folder.
3. Rename the folder to **generate** and move them to YourDriveLetter:\xampp\htdocs, the final path should look something like C:\xampp\htdocs\generate or your custom directory should you have one.  
     
   
4. Launch Apache and MySQL from Xampp control panel.



1. Navigate to localhost/phpmyadmin on a browser (on host machine) and click on the SQL tab



1. Copy the content from C:\xampp\htdocs\generate\create.sql and paste it inside the text field. Then press go to create the database.



1. Navigate to <http://localhost/generate/pass_phrase.html>. The web application should be loaded and everything should work well.
2. Should the mySQL database have a non-default username and password, change them in dbconfig.php.

**Open source libraries used:**

* Javascypt - <https://www.fourmilab.ch/javascrypt/pass_phrase.html>
* QR generator - <https://davidshimjs.github.io/qrcodejs/>

**External API used:**

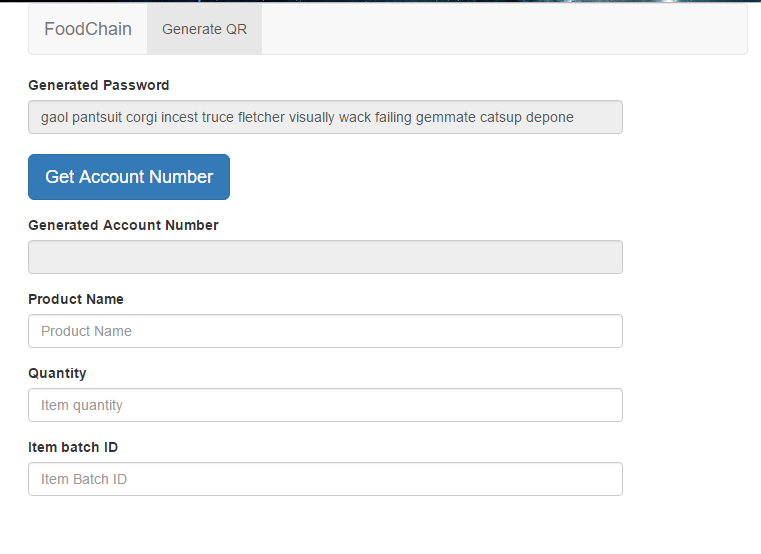
* NXT API - <https://nxtwiki.org/wiki/The_Nxt_API>

**Files to take note off:**

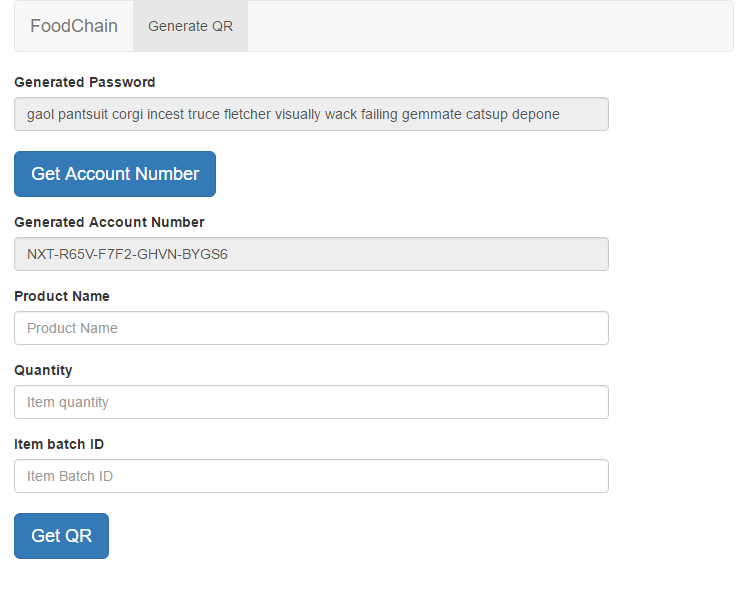
* Controller.js in the js folder – It contains most of the logics of the web application. The rest of the JavaScript files are merely dependencies for some libraries to work.

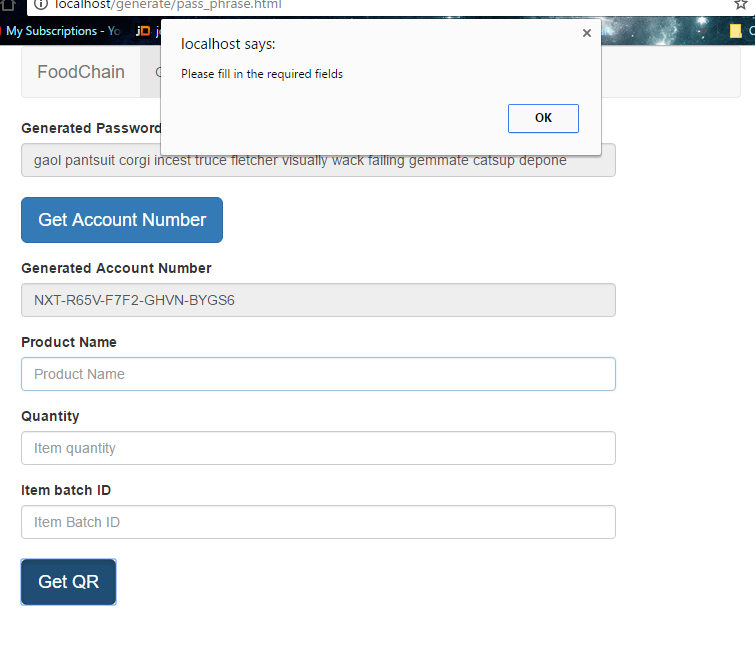
**Use Case:**

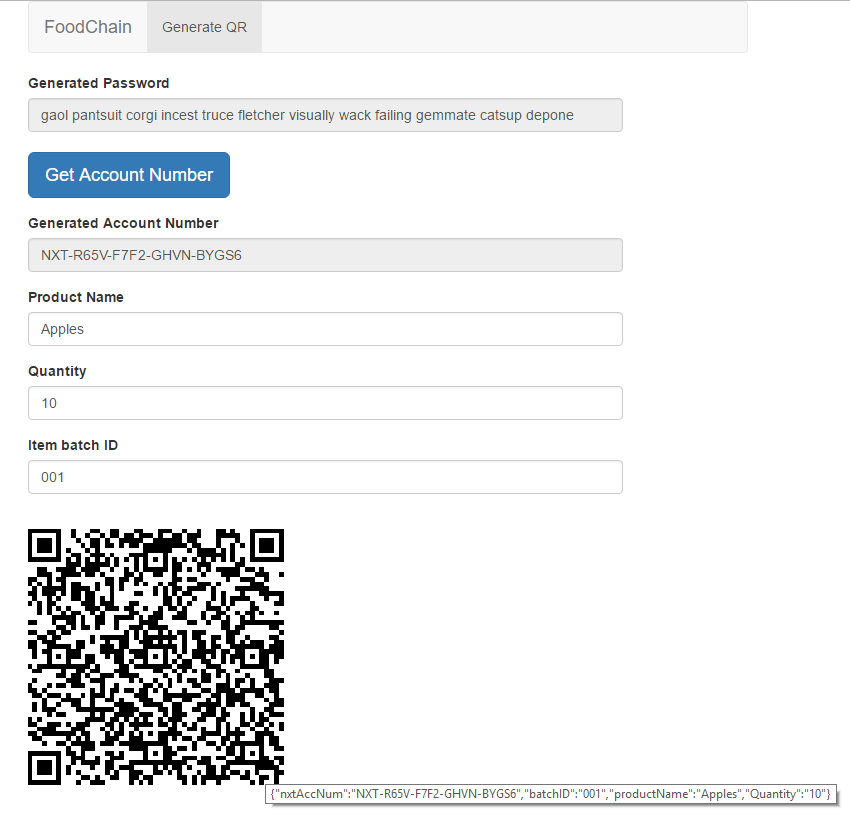
1. A worker has 60 apples, and divide them into packs to 6, ending up with 10 packs in total.
2. The worker proceeds to generating the QR for this batch of 60 apples using the ***generate web application*** (FoodChain- Product QR Generator). The QR generated will be printed and placed on each of the 10 packs to signify they are from the same batch.
   * There are two functions of product QR:  
     1. As a vessel to receive transactions made for this particular type of item. In the QR contains a NXT account number, which will receive the transactions made by the PreConsumer mobile application and also the batch ID to show which batch of product the transaction is for.  
        .
     2. As the QR contains the NXT account number and also the batch ID, when scanned by the Consumer App will query the all the transactions for that account, and retrieve the relevant ones which will later be displayed to the user.

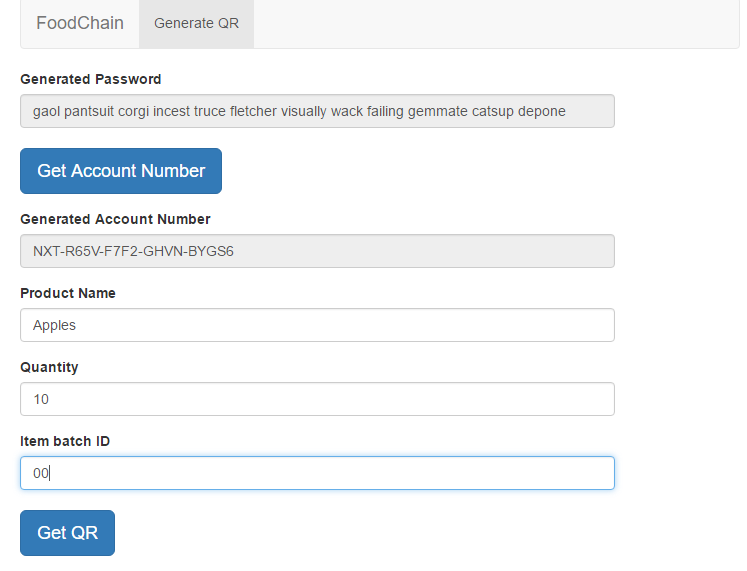
**Screenshots of usage:**If the database(atos) and table (nxtaccounts) has been created, this pass\_phrase.html should look like this when loaded on a browser (password has been generated).  
  


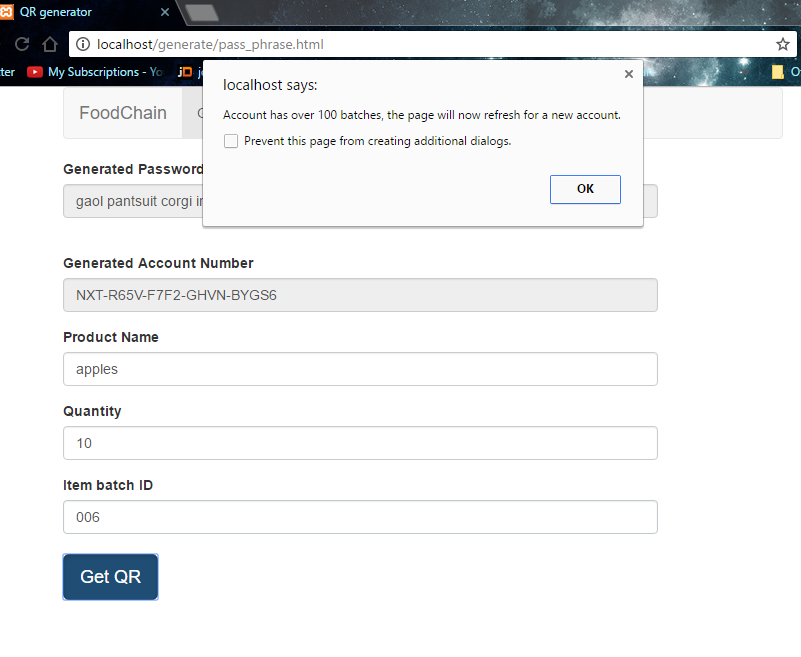
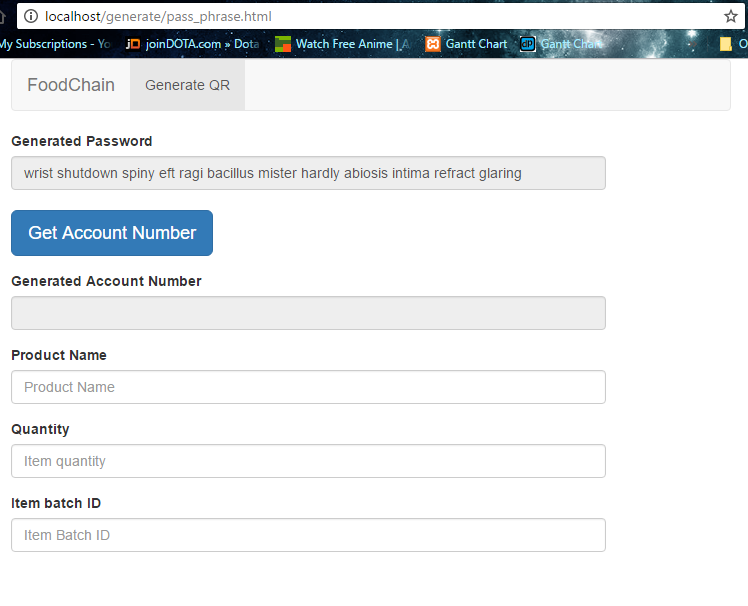
After pressing the get account number button, the generated password will be used to create a NXT account via API call, and the generated account number should appear, along with the get QR button.



If the get QR button is pressed without the all the 3 fields being filled in, an error prompt will be shown.  
  


If all the fields are filled, pressing the get QR will generate a QR with the entered information.  
  
  
  
If a QR has been generated, and any of the fields (product name, quantity or batch ID) changes, the generated QR will be hidden and the get QR button will be shown



If the account has 99 batches for instance, and you proceed to generate 2 more without refreshing the page (one account is meant to have 100), the attempt to create the 101th QR will be greeted with a message prompt, which then refreshes the page.  
  
  
  
  
After the page refreshes, the generated account number will be empty as the new account has not been created.   
  


Every time when creating a new account no matter if it’s the first account in the database or the previous account has reached 100 batches, a random password will be generated on each refresh. This password will be used to query the blockchain via an API call. It will then return a NXT Account number. Following it will be another query to the blockchain to check if there are any transactions for the account number returned. If it does not have any transactions (which is a new account), it will be appear into the generated account number and the user is able to proceed.

If the generated password returns an account number which has transactions on it (unlikely but possible), it shows that it belongs to someone, the page will alert the user about the happening and refresh the page for a new password.

