

Имена: Симеон Станимиров Александров

фн: 61916

Начална година: 2018

Програма: бакалавър, (СИ) **Курс:** 3

Тема: Чат бот

Дата: 2018-06-30

Предмет: wwwTech2018_10ed_SI_summer

имейл: simeon.alexandrov17@gmail.com

преподавател: доц. д-р Милен Петров

ТЕМА: Чат бот

1. Условие

Дефиниране на област от проблеми – или възможност за клъстеризация на проблемите, експертно назначаване на ,бот‘ и отговарянето на писма; да има unit-tests, показващи работата на бот-а; Варианти на идеята: а. LoremHelpBot: Може да бъде просто ,чат‘ от тип call-център, например на първа страница да дава в момента се свързахте с call центъра на предмета Уеб технологии (КН), ФМИ, СУ.

i. Изберете 1 – за български, изберете 2 – за английски, 9 – за оператор □

ii. Вие избрахте 1, изберете 1 – за оплаквания, 2 за похвали, 9 за оператор

iii. Моля изчакайте да се свържете с оператор....и генерира рандом помощ – докато чакате;

b. Вариант – задава въпрос, ако този въпрос го има в базата – връща се отговор – с възможност оператор после да оценява отговорите и да ги рейтва; Да може да има роля както ,запитващ‘, така и ,оператор‘ и ,администратор‘.

2. Въведение

World Cup Q&A е името на проекта. Идеята е да създаде информационна платформа за световното първенство по футбол в Русия, 2018. Системата позволява регистрирането на два вида потребители - запитващ (обикновен потребител) и отговарящ (оператор), като различните роли имат достъп до различна част от функционалността на системата. Това се постига с поддържането на сесия в PHP.

Функционални изисквания са:

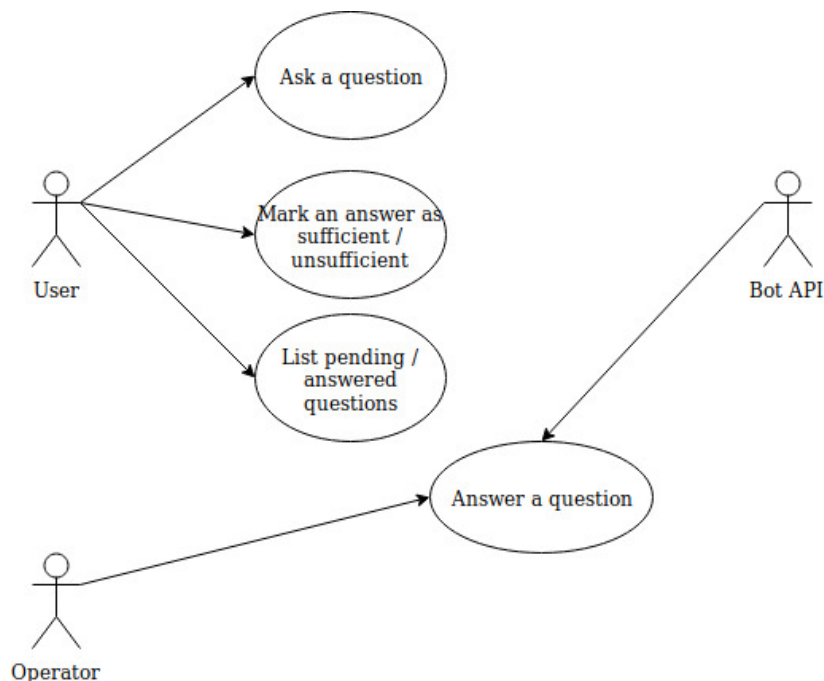
- Ползване на API за чатбот (например Dialogflow)
- Системата ще поддържа два вида потребители – запитващ и оператор.
- Потребителят ще може да оцени отговорът на бота и ако смята, че е

незадоволителен отговорът, то въпросът се маркира като НЕОТГОВОРЕН и

оператор, потребител със специална роля, може да го отговори.

- И двата типа потребители ще могат да разглеждат списък с въпроси и техният статус (отговорен/неотговорен)

Диаграма на основният потребителски случай:



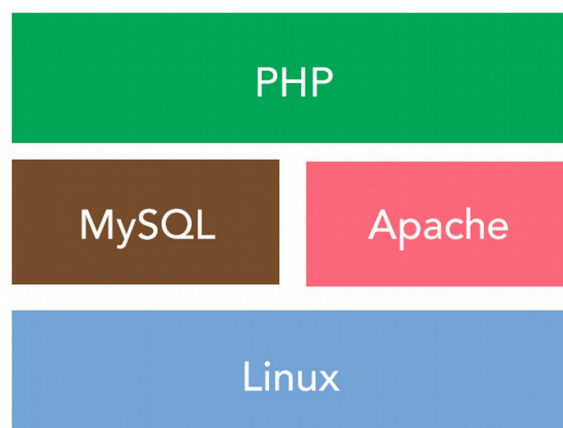
World Cup Q&A използва API на системата Dialogflow, която използва алгоритми за разпознаване на свободен език (natural language processing), за да дава отговор на зададен въпрос. Възможностите на Dialogflow са доста големи и включват доста машинно самообучение и извличане на информация от други интерфейси, но в проекта се използва ограничена функционалност на Dialogflow. На практика World Cup Q&A използва Dialogflow като база данни, в която за въпрос има няколко съответни отговори. Бонусът от ползването на външната система в условното разпознаване на свободен език.

3. Теория

...

4. Използвани технологии

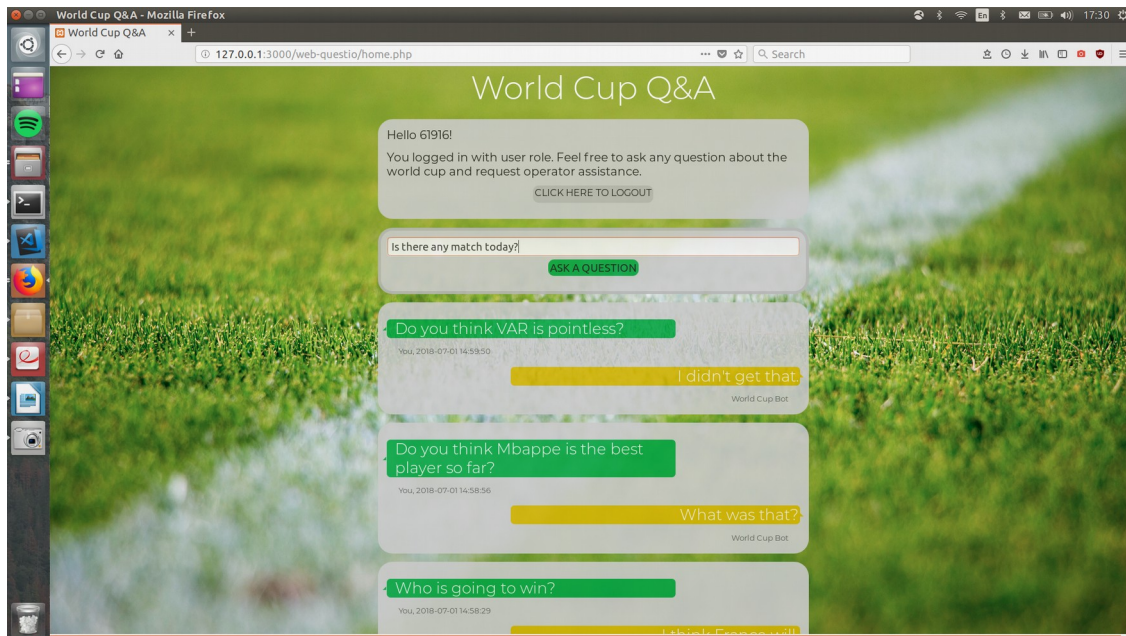
Системата е реализирана използвайки LAMP стека. Apache Web Server, който сервира страници HTML5 + JS, а бизнес логиката е имплементирана с PHP Version 5.6.33. Всичко това е разработено и публикувано в Azure Cloud чрез Linux среда. Базата данни е MySQL.



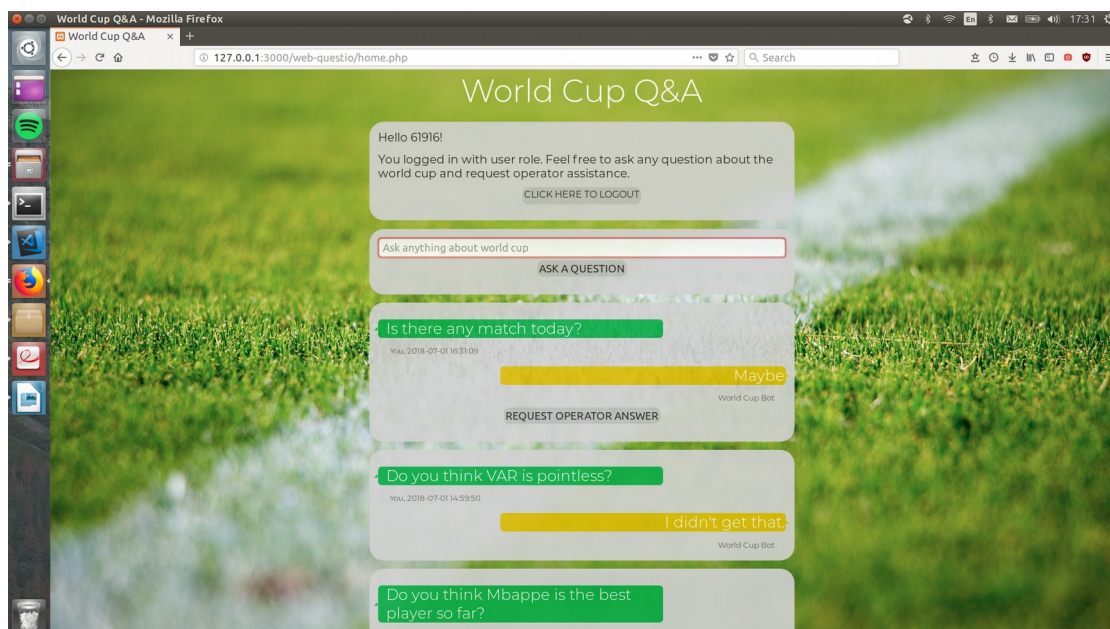
5. Инсталация и настройки

Проектът е разработен с XAMPP в Linux среда и предполагам така ще бъде тестван. В папката има текстов файл *questio.sql*, който дефинира базата данни и необходимите таблици. Има и конфигурационен файл *config.php*, който служи за настройване на връзката със базата. Този документ е също в директорията на проекта и служи за пояснение.

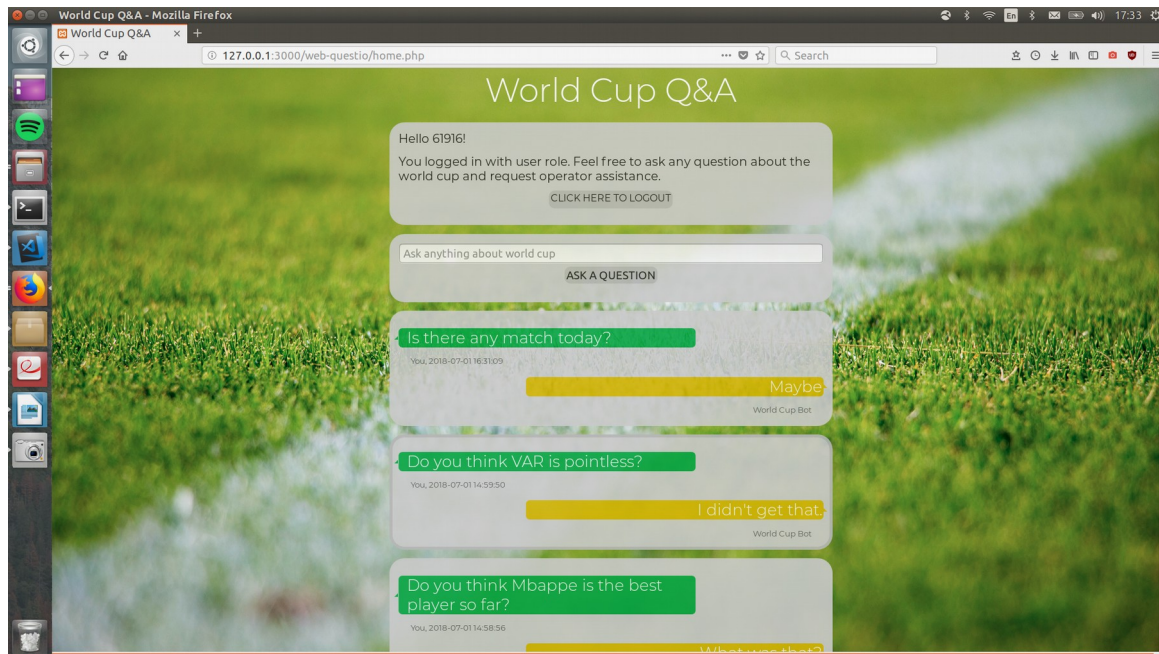
6. Кратко ръководство на потребителя



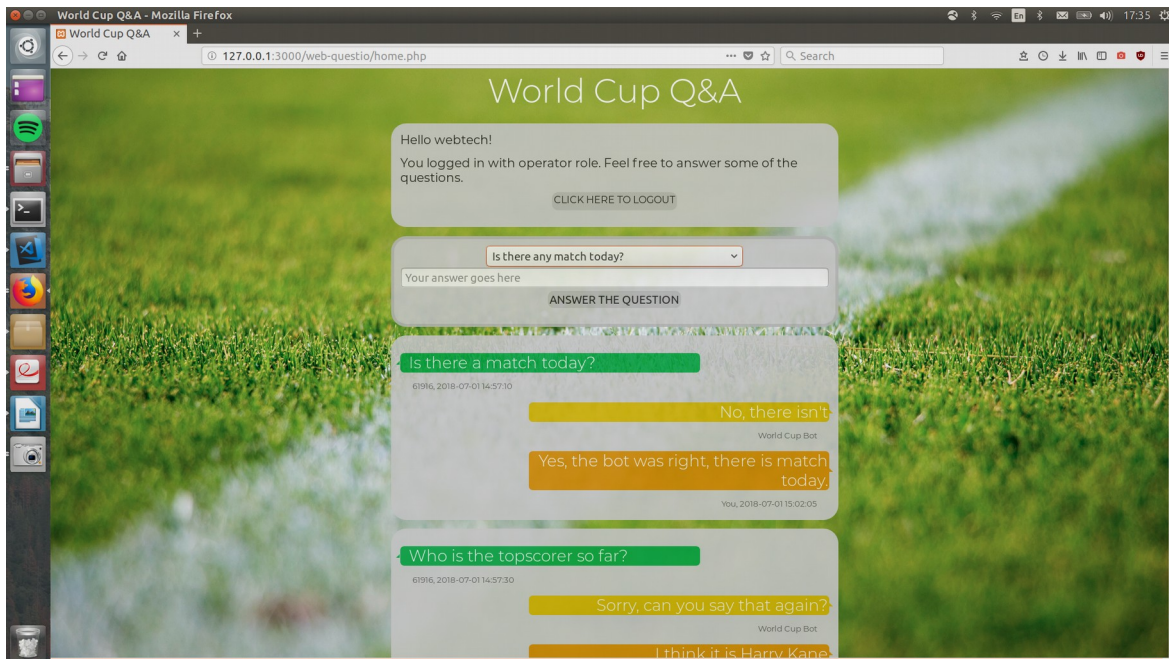
Това е основният екран на приложението, което запитващият вижда след като предостави име и парола. Той може да зададе въпрос.



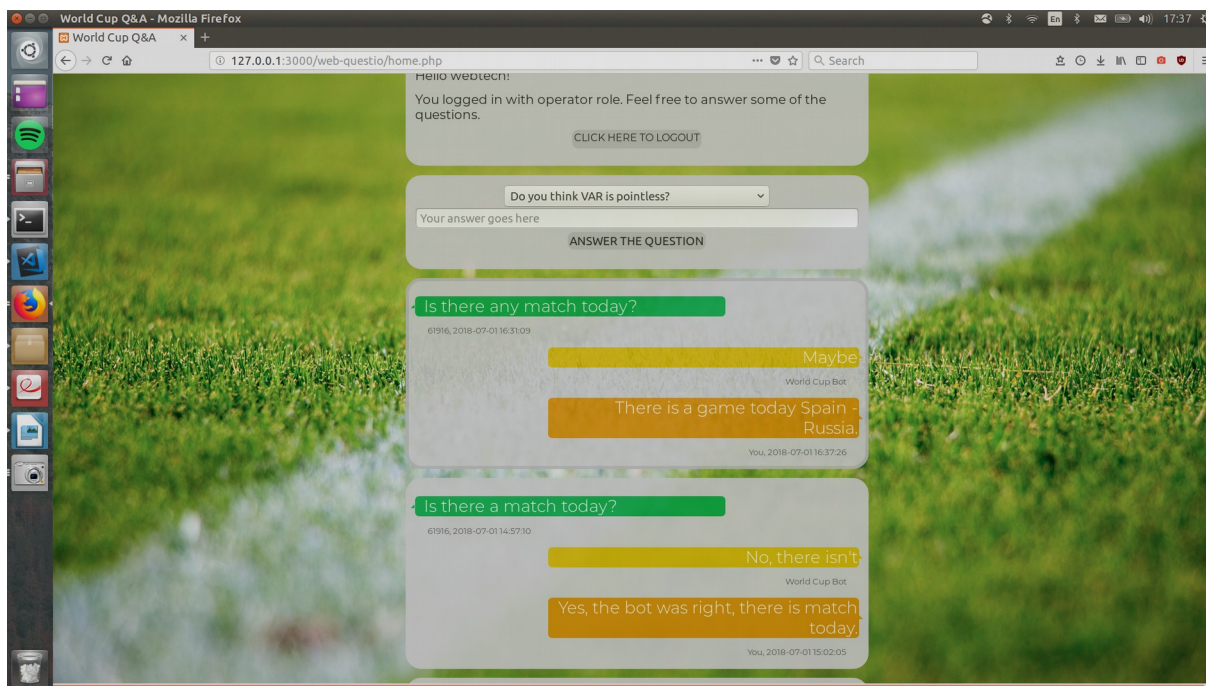
Ботът връща отговор и ако потребителят не е доволен от отговора, той може да подаде заявка въпросът да бъде разгледан от оператор.



Заявката може да се подаде само веднъж за всеки въпрос.



Това е екран на друг потребител – този път с ролята на оператор. Той може да избира от списък с въпроси, за които е подадена заявка.



При отговор и двамата потребители виждат в историята на използването на системата резултата от запитването и отговора.

7. Примерни данни

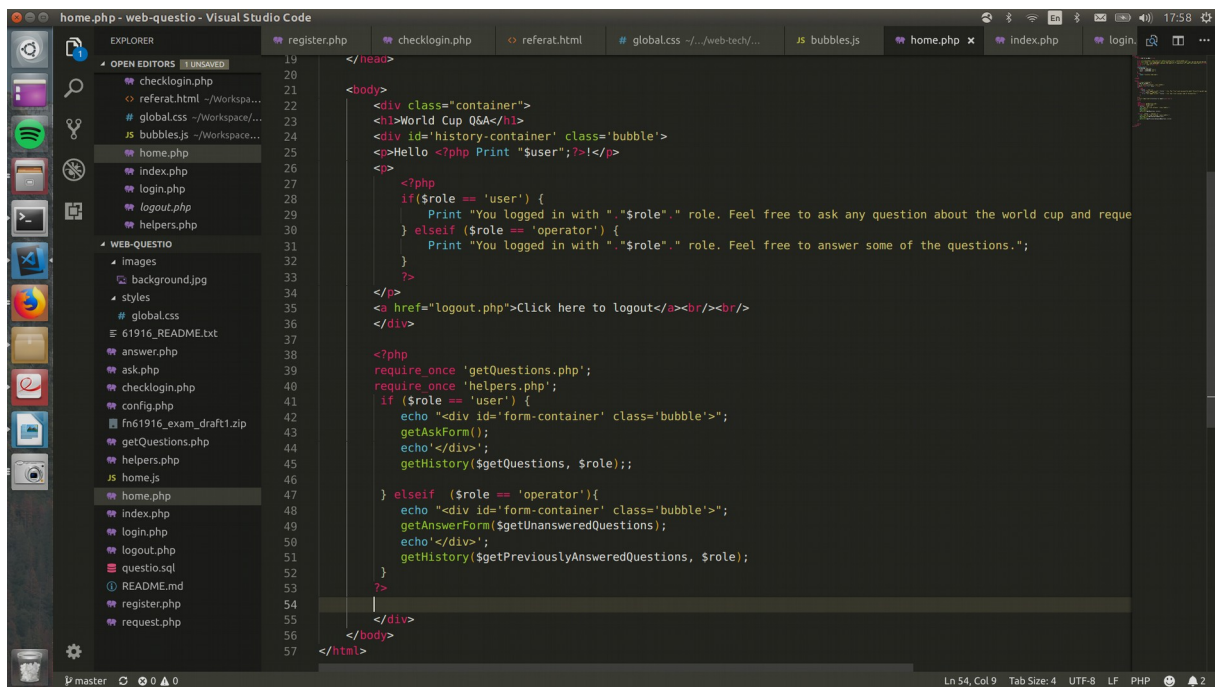
Проектът е публикуван в Azure Cloud на следния адрес: <https://really-cool-php-app.azurewebsites.net>, където са създадени двама потребители:

- username: 61916 password: 61916 role: user
- username: webtech password: webtech role: operator

На този адрес може да се използва системата в реална обстановка - единият потребител може да задава въпроси, а другият да отговаря. Добавени са и няколко въпроса за пояснение.

8. Описание на програмния код

Кодът на основната страница – вижда се разделението на това какво потребителят ще вижда в зависимост от ролята му.

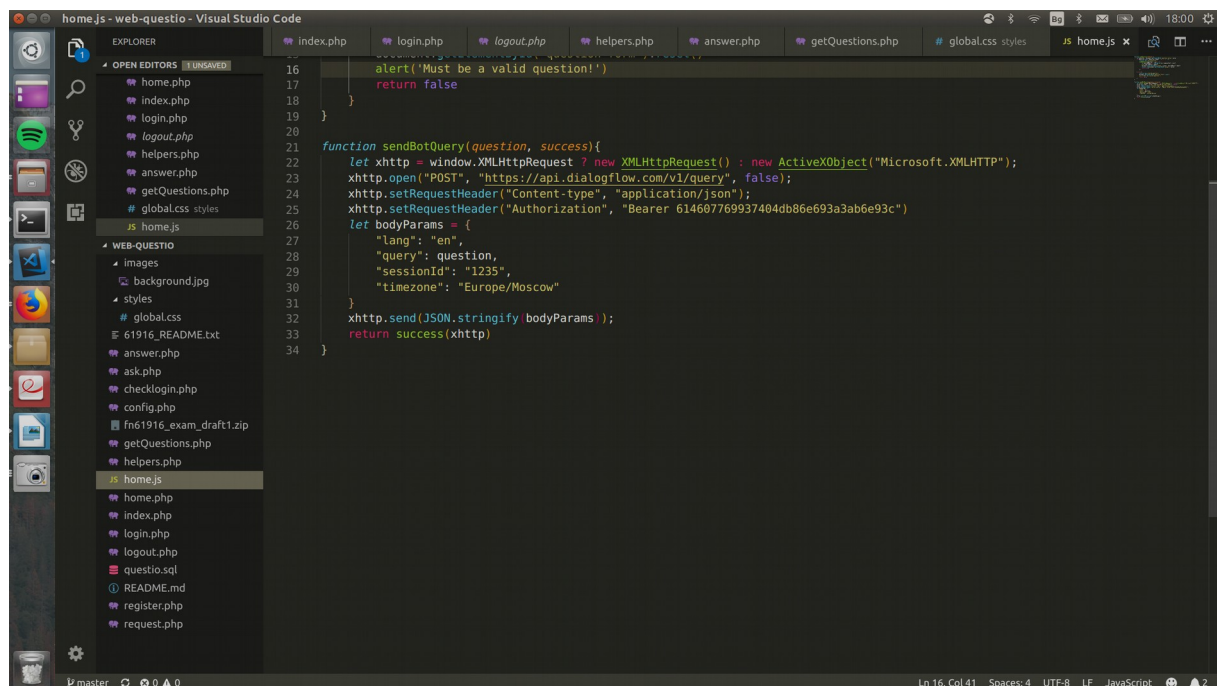


```
<?php
require_once 'helpers.php';
require_once 'getQuestions.php';

if ($role == 'user') {
    echo "<div id='form-container' class='bubble'>";
    getAskForm();
    echo "</div>";
    getHistory($getQuestions, $role);
} elseif ($role == 'operator') {
    echo "<div id='form-container' class='bubble'>";
    getAnswerForm($getUnansweredQuestions);
    echo "</div>";
    getHistory($getPreviouslyAnsweredQuestions, $role);
}

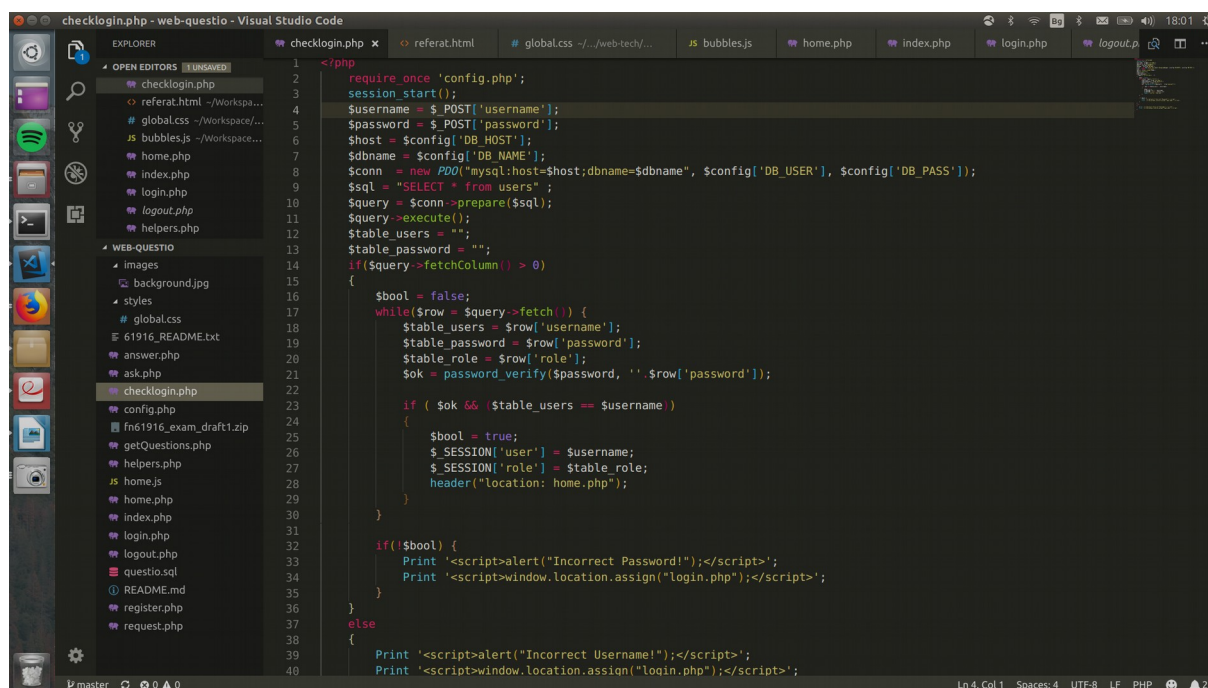
?>
</div>
</body>
</html>
```

Така става заявката към Dialogflow.



```
function sendBotQuery(question, success){
    let xhttp = window.XMLHttpRequest ? new XMLHttpRequest() : new ActiveXObject("Microsoft.XMLHTTP");
    xhttp.open("POST", "https://api.dialogflow.com/v1/query", false);
    xhttp.setRequestHeader("Content-type", "application/json");
    xhttp.setRequestHeader("Authorization", "Bearer 614607769937404db86e693a3ab6e93c");
    let bodyParams = {
        "lang": "en",
        "query": question,
        "sessionId": "1235",
        "timezone": "Europe/Moscow"
    };
    xhttp.send(JSON.stringify(bodyParams));
    return success(xhttp);
}
```

Проверка при логин



9. Приноси на студента, ограничения и възможности за бъдещо разширение

Бих искал да добавя страница със статистика – най-любопитни потребители, оператори с най-много отговори и най-често срещани въпроси.

10. Какво научих

Научих доста за PHP, как да публикувам уеб приложения в Azure, как да интегрирам външна система като Dialogflow. Като цяло се радвам, че успях да вдигна система с frontend и backend за няколко дни с документация към нея и публикувана в Azure.

11. Използвани източници

Bot API: <https://dialogflow.com/>

Заден фон: https://unsplash.com/photos/HgwY_YQ1m0w

CSS на съобщенията: <https://leaverou.github.io/bubbly/>

LAMP стек: <http://www.startlandnews.com/wp-content/uploads/2016/01/Screenshot-2016-01-28-at-11.09.34-AM.png>

Предал (подпис):

/фн, имена, спец., група/

Приел (подпис):

/доц. Милен Петров/