

IOT task

Controller.php

```
<?php

class BaseController
{
    /**
     * __call magic method.
     */
    public function __call($name, $arguments)
    {
        $this->sendOutput('', array('HTTP/1.1 404 Not Found'));
    }

    /**
     * Get URI elements.
     *
     * @return array
     */
    protected function getUriSegments()
    {
        $uri = parse_url($_SERVER['REQUEST_URI'], PHP_URL_PATH);
        $uri = explode( '/', $uri );

        return $uri;
    }

    /**
     * Get querystring params.
     *
     * @return array
     */
    protected function getQueryStringParams()
    {
        parse_str($_SERVER['QUERY_STRING'], $query);
        return $query;
    }

    /**
     * Send API output.
     *

```

```

    * @param mixed $data
    * @param string $httpHeader
    */
protected function sendOutput($data, $httpHeaders=array())
{
    header_remove('Set-Cookie');

    if (is_array($httpHeaders) && count($httpHeaders)) {
        foreach ($httpHeaders as $httpHeader) {
            header($httpHeader);
        }
    }

    echo $data;
    exit;
}

```

numbercontroller

<?php

```

class NumberController extends BaseController
{
    /**
     * "/user/list" Endpoint - Get list of users
     */
    public function listAction()
    {
        $strErrorDesc = '';
        $requestMethod = $_SERVER["REQUEST_METHOD"];
        $arrQueryStringParams = $this->getQueryStringParams();

        if (strtoupper($requestMethod) == 'GET') {
            try {
                $numberModel = new NumberModel("number.sqlite");

                $intLimit = 10;
            }
        }
    }
}

```

```

        if (isset($arrQueryStringParams['limit']) &&
            $arrQueryStringParams['limit']) {

            $intLimit = $arrQueryStringParams['limit'];

        }

        $arrUsers = $numberModel->getNumber($intLimit);
        $responseData = json_encode($arrUsers);
    } catch (Error $e) {
        $strErrorDesc = $e->getMessage(). 'Something went wrong!
Please contact support.';
        $strErrorHeader = 'HTTP/1.1 500 Internal Server Error';
    }
} else {
    $strErrorDesc = 'Method not supported';
    $strErrorHeader = 'HTTP/1.1 422 Unprocessable Entity';
}

// send output
if (!$strErrorDesc) {
    $this->sendOutput(
        $responseData,
        array('Content-Type: application/json', 'HTTP/1.1 200
OK')
    );
} else {
    $this->sendOutput(json_encode(array('error' =>
$strErrorDesc)),
        array('Content-Type: application/json', $strErrorHeader)
    );
}
}

public function post()
{
    $strErrorDesc = '';
    $requestMethod = $_SERVER["REQUEST_METHOD"];
    $strErrorHeader = 'HTTP/1.1 422 Unprocessable Entity';
    if (strtoupper($requestMethod) == 'POST') {
        try {
            $myPost = array_values($_POST);
            $numberModel = new NumberModel("number.sqlite");
            if (count($myPost) == 1 && isset($_POST["number"])) {
                $number = (int)$POST["number"] ;
            } else {$this->sendOutput(json_encode(array('error' =>

```

```

$strErrorDesc)),
        array('Content-Type: application/json',
$strErrorHeader)
    );}
    if(isset($number) && is_int($number) && $number !=
null){

        $numberModel -> postNumber($number);
        $this->sendOutput(
            json_encode(["success" => "true"]),
            array('Content-Type: application/json',
'HTTP/1.1 200 OK')
        );

    } else {$this->sendOutput(json_encode(array('error' =>
$strErrorDesc)),
        array('Content-Type: application/json', $strErrorHeader)
    );}

    } catch (Error $e) {
        $strErrorDesc = $e->getMessage(). 'Something went wrong!
Please contact support.';
        $strErrorHeader = 'HTTP/1.1 500 Internal Server Error';
    }
} else {
    $strErrorDesc = 'Method not supported';
    $strErrorHeader = 'HTTP/1.1 422 Unprocessable Entity';
}

// send output
if ($strErrorDesc) {
    $this->sendOutput(json_encode(array('error' =>
$strErrorDesc)),
        array('Content-Type: application/json', $strErrorHeader)
    );

}

}
}

```

bootstrap

```
<?php

define("PROJECT_ROOT_PATH", __DIR__ . "/../");

require_once PROJECT_ROOT_PATH . "/inc/config.php";

require_once PROJECT_ROOT_PATH . "/Controller/Api/BaseController.php";
require_once PROJECT_ROOT_PATH . "/Model/NumberModel.php";
?>
```

Configure.php

```
<?php

class db {
    private $location ;
    public function __construct($location , $mode = SQLITE3_OPEN_CREATE)
    {
        $this->location = $location;
    }
    public function sqlite_create() {

        $connection = new SQLite3($this -> location);
        return $connection;
    }
    function sqlite_query($connection,$query)
    {
        $array['dbhandle'] = $connection;
        $array['query'] = $query;
        $result = $connection->query($query);
        return $result;
    }
    function sqlite_fetch_array(&$result,$type = SQLITE3_ASSOC)
    {
        $data= array();

        while ($res = $result->fetchArray($type))
```

```

    {
        array_push($data, $res);
    }
    return $data;
}
function seedRandInt($conn,String $tablename , String $calname , int $min = 0,
int $max = 50, $times = 1)
{
    for ($i = 0; $i < $times; $i++) :
        for ($j = 0; $j < $max; $j++):
            $value = rand($min, $max);
            try {
                $conn->exec("INSERT INTO $tablename($calname)VALUES($value);");
            } catch (Exception $e) {
                echo $e->getMessage();
                return -1;
            }
        endfor;
        return 0;
    endfor;
}
}

```

Build database

```

<?php
require_once "config.php";
$file = "number.sqlite";
$databse = new db($file);
$db = $databse -> sqlite_create();
try {
    $create_table = $db->exec("
    CREATE TABLE numbers (
        id INTEGER PRIMARY KEY,
        number INTEGER NOT NULL
    );");
} catch (Exception $e) {
    echo $e->getMessage();
}
$databse -> seedRandInt($db , "numbers" , "number");
$db ->close();

```

Numbermodel

```
<?php
class numberModel extends db
{
    public function getNumber(int $limit)
    {
        try{

            $sql = "SELECT * FROM numbers ORDER BY id ASC LIMIT $limit";
            $connection = $this -> sqlite_create();
            @$statement = $connection->prepare($sql);
            $statement->bindValue('LIMIT', $limit, SQLITE3_INTEGER);
            $result = $statement->execute();
        } catch(Exception $e){

            header("Location: http://localhost:8080/404.php");
            exit();
        }

        return $this -> sqlite_fetch_array($result);
    }
    public function postNumber(int $number)
    {
        try{

            $sql = "INSERT INTO  numbers (NUMBER) VALUES ($number)";
            $connection = $this -> sqlite_create();
            @$statement = $connection->prepare($sql);
            $statement->bindValue('VALUES', $number, SQLITE3_INTEGER);
            $statement->execute();
        } catch(Exception $e){

            header("Location: http://localhost:8080/404.php");
            exit();
        }

        return 0;
    }
}
```

Index.php

```
<?php

require __DIR__ . "/inc/bootstrap.php";

if (!file_exists("number.sqlite")) {require_once PROJECT_ROOT_PATH .
"/inc/buildDatabase.php";}
$uri = parse_url($_SERVER['REQUEST_URI'], PHP_URL_PATH);
$uri = explode( '/', $uri );

if ((isset($uri[2]) && $uri[2] != 'numbers') || !isset($uri[3])) {
    header("Location: http://localhost:8080/404.php");
    exit();
}

require PROJECT_ROOT_PATH . "/Controller/Api/NumberController.php";
$requestMethod = $_SERVER["REQUEST_METHOD"];
$objFeedController = new NumberController();
try{
    $strMethodName = $uri[3] . 'Action';
    if ($uri[3] == "post" && $requestMethod == "POST") :
        $objFeedController->post();
    endif;
    $objFeedController->{$strMethodName}();
}catch( Exception $e) {
}
?>
```

Html file

```
<!DOCTYPE
PE
html>

<html>
<head>
    <link
href="https://fonts.googleapis.com/css2?family=Nunito+Sans:wght@600;900&di
splay=swap" rel="stylesheet">
```



```

    <script src="https://kit.fontawesome.com/4b9ba14b0f.js"
crossorigin="anonymous"></script>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="mainbox">
        <div class="err">4</div>
        <i class="far fa-question-circle fa-spin"></i>
        <div class="err2">4</div>
        <div class="msg">Maybe this page moved? Got deleted? Is hiding out in
quarantine? Never existed in the first place?<p>Let's go <a
href="#">home</a> and try from there.</p></div>
    </div>
</body>
</html>

```

Css file

```

body
{
    background-color: #95c2de;
}
.mainbox {
    background-color: #95c2de;
    margin: auto;
    height: 600px;
    width: 600px;
    position: relative;
}
.err {
    color: #ffffff;
    font-family: 'Nunito Sans', sans-serif;
    font-size: 11rem;
    position: absolute;
    left: 20%;
    top: 8%;
}
.far {
    position: absolute;
    font-size: 8.5rem;
}

```

```
    left: 42%;
    top: 15%;
    color: #ffffff;
}
.err2 {
    color: #ffffff;
    font-family: 'Nunito Sans', sans-serif;
    font-size: 11rem;
    position: absolute;
    left: 68%;
    top: 8%;
}
.msg {
    text-align: center;
    font-family: 'Nunito Sans', sans-serif;
    font-size: 1.6rem;
    position: absolute;
    left: 16%;
    top: 45%;
    width: 75%;
}
a {
    text-decoration: none;
    color: white;
}
a:hover {
    text-decoration: underline;
}
```