Платформа за онлайн банкиране

Курсов проект по УЕБ програмиране

(3ти курс)

Технически Университет – Варна

Катедра СИТ

Изготвили:

Николай Панев: 20621511

Венелин Дачев: 20621502

Проверил:

Доц. В. Божикова

(подпис)

............................................

Цел на проекта:

Онлайн банкова платформа, позволяваща на потрабителите да създават банкови акаунти в една или двете банки с които работи платформата, използвайки един юнифициран онлайн акаунт.

Потребителите имат възможност да внасят и теглят валута, и да изпращат средства от една сметка в друга, независимо в коя от двете банки е сметката на наредителя или получателя.

Използване на програмата

от потребител

Начална Страница:

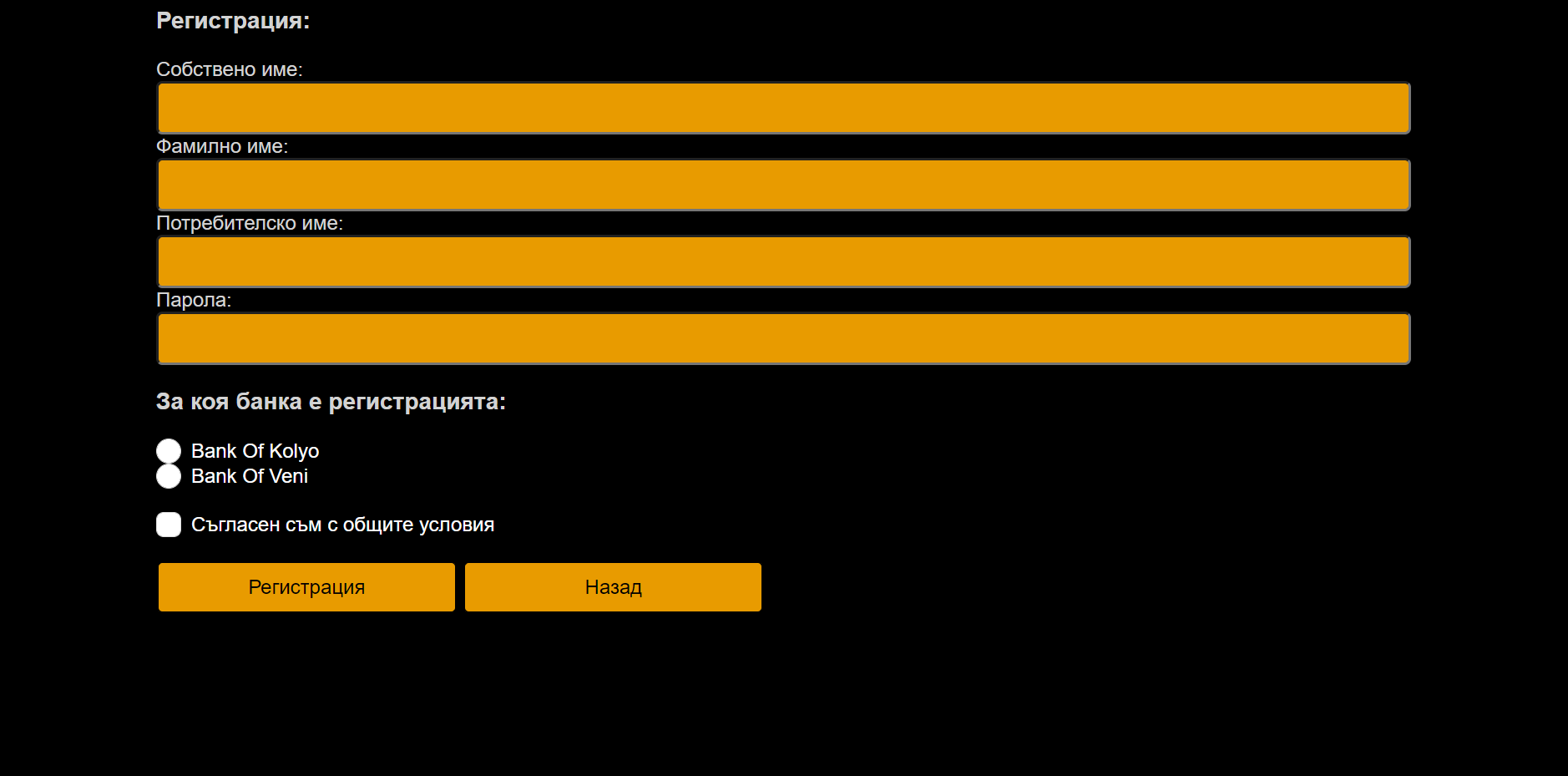
Потребителите, при влизане в платформата, попадат на началната страница, където имат избор да се впишат във вече съществуващ онлайн акаунт, или да се регистрират в платформата.

Graphical user interface, text, application

Description automatically generated

Регистрация:

При регистрация, потребителите трябва да съседат собствено име, фамилно име, потребителско име и парола. Те също трябва да изберат за коя банка правят своята регистрация, използвайки радио бутоните, както и да отбележат че са съгласни с общите условия за използване на приложението.



Вход:

При вход, потребителите вече трябва да имат регистриран акаунт в платформата, в банката, където е акаунта който искат да достъпят. При вход се въвежда единствено потребителско име и парола, и от радио бутоните се избира в коя банка е акаунта в който искат да влезнат.

Graphical user interface

Description automatically generated

Потребителски интерфейс:

След регистрация/вход, потребителят попада на началната страница на потребителският интерфейс. След регистрация, балансът винаги е 0 лева, но за да симулираме внасянето на пари сме добавили полета и бутони за внасяне и теглене на пари.

Потребителите също имат възможност да изберат да направят транзакция, да видят историята на транзакциите или да излезнат от своя профил.

Graphical user interface, text

Description automatically generated

Извършване на транзакции:

За да може един потребител да извърши транзакция, той трябва да знае данните на потребителят на който би искал да изпрати средства. Получателят може да бъде от същата или от друга банка.

Нужните данни са:

* Собствено име на получател
* Фамилно име на получател
* IBAN на получател
* Сумата която потребителят иска да изпрати(Равна или по-малка от баланса на потребителя)
* Основание за транзакцията

Потребителят също може да се върне и към основната страница на потребителският интерфейс без да извърши транзакция.

A picture containing graphical user interface

Description automatically generated

История на празнакциите:

Ако потребителят избере да види историята на своите транзакции, той трябва да избере времеви диапазон за който да бъдат показани транзакциите.

Graphical user interface, application

Description automatically generated

След избор на времеви диапазон, на екрана се извеждат всички транзакции за този диапазон.

Graphical user interface

Description automatically generated

Софтуерна документация

Използвани технологии:

-MySQL

-PHP

-HTML5

-CSS

-JavaScript

-Git

-GitHub

-FTP (FileZilla)

-Oracle Data Modeler(За документация)

Структура на база от данни:

Diagram

Description automatically generated

Файл за работа с база от данни (DBFunc.PHP):

INSERT функции:

function Insert1($TABLE,$COLUMN,$VALUE){

//database credentials

include("Connect.php");

//tell the database what to insert and into which table

$sql = "INSERT INTO $TABLE($COLUMN) VALUES('$VALUE')";

//On a successfull insert

if ($conn->query($sql) === TRUE) {

echo "New record created successfully";

}

//On a failed insert

else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

Disconnect($conn);

}

////////////////////////////////////////////////////////////////////

function Insert2($TABLE,$COLUMN1,$COLUMN2,$VALUE1,$VALUE2){

//database credentials

include("Connect.php");

$sql = "INSERT INTO $TABLE($COLUMN1,$COLUMN2) VALUES('$VALUE1','$VALUE2')";

//On a successfull insert

if ($conn->query($sql) === TRUE) {

echo "New record created successfully";

}

//On a failed insert

else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

Disconnect($conn);

}

//////////////////////////////////////////////////////////////////////

function Insert3($TABLE,$COLUMN1,$COLUMN2,$COLUMN3,$VALUE1,$VALUE2,$VALUE3){

//database credentials

include("Connect.php");

$sql = "INSERT INTO $TABLE($COLUMN1,$COLUMN2,$COLUMN3) VALUES('$VALUE1','$VALUE2','$VALUE3')";

//On a successfull insert

if ($conn->query($sql) === TRUE) {

echo "New record created successfully";

}

//On a failed insert

else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

Disconnect($conn);

}

////////////////////////////////////////////////////////////////////////

function Insert4($TABLE,$COLUMN1,$COLUMN2,$COLUMN3,$COLUMN4,$VALUE1,$VALUE2,$VALUE3,$VALUE4){

//database credentials

include("Connect.php");

$sql = "INSERT INTO $TABLE ($COLUMN1,$COLUMN2,$COLUMN3,$COLUMN4) VALUES ('$VALUE1','$VALUE2','$VALUE3','$VALUE4')";

//On a successfull insert

if ($conn->query($sql) === TRUE) {

echo "New record created successfully";

}

//On a failed insert

else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

Disconnect($conn);

}

////////////////////////////////////////////////////////////////////////

function Insert5($TABLE,$COLUMN1,$COLUMN2,$COLUMN3,$COLUMN4,$COLUMN5,$VALUE1,$VALUE2,$VALUE3,$VALUE4,$VALUE5){

//database credentials

include("Connect.php");

$sql = "INSERT INTO $TABLE ($COLUMN1,$COLUMN2,$COLUMN3,$COLUMN4, $COLUMN5) VALUES ('$VALUE1','$VALUE2','$VALUE3','$VALUE4','$VALUE5')";

//On a failed insert

if ($conn->query($sql) === FALSE) {

echo "Error: " . $sql . "<br>" . $conn->error;

}

Disconnect($conn);

}

CREDENTIALS:

//Returns true if the username doesn't exist in the database

function CheckUsernameAvailability($Username,$bankNumber){

//database credentials

include("Connect.php");

//Ask the database for the entry with that username

$bank=getBank($bankNumber);

$query = "SELECT \* FROM $bank WHERE Username='$Username';";

if(mysqli\_query($conn, $query)){

//Get what the database has answered

$result = $conn->query($query);

//If there is an entry with that username,

//disconnect and return false

if ($result->num\_rows > 0)

{

Disconnect($conn);

return 0;

}

}

//If the username is not taken yet,

//disconnect and return true

Disconnect($conn);

return 1;

}

//////////////////////////////////////////////////////////////////////////

//Checks if the user password is correct ,and returns true(1) if correct

function CheckPassword($Username,$Password){

//database credentials

include("Connect.php");

//Ask the database for the entry with these username and password

$query = "SELECT \* FROM Credentials WHERE Username='$Username' AND Password='$Password';";

if(mysqli\_query($conn, $query)){

//Get what the database has answered

$result = $conn->query($query);

//If there is and entry with these username and password,

//disconnect from the database and return true(1)

if ($result->num\_rows > 0)

{

Disconnect($conn);

return 1;

}

}

//if there isn't a user like this or the password doesn't match,

//disconnect and return false(0)

Disconnect($conn);

return 0;

}

//Checks if the user has the correct bank selected, and returns true(1) if corret

function CheckBank($Username, $bank){

//database credentials

include("Connect.php");

//Asks the database for an entry in the bank with this username

$query = "SELECT Username FROM $bank WHERE Username='$Username';";

if(mysqli\_query($conn, $query)){

//Get what the database has answered

$result = $conn->query($query);

//If there is an entry in the bank with this username,

//disconnect from the database and return true(1)

if ($result->num\_rows > 0){

Disconnect($conn);

return 1;

}

}

//if there isn't a username in this bank

//disconnect and return false(0)

Disconnect($conn);

return 0;

}

FETCH:

//"Username", $Username, $bank, "IBAN");

function fetchValue($checkCol, $checkValue, $table, $value){

//database credentials

include("Connect.php");

//Select the Value from the appropriate bank

$query = "SELECT $value FROM $table WHERE $checkCol='$checkValue';";

//Get what the database has answered

$result = $conn->query($query);

$row = $result->fetch\_assoc();

$fetchedValue = $row[$value];

Disconnect($conn);

return $fetchedValue;

}

function bankName($iban){

if (strpos($iban, "BOKB") !== false) return "BankOfKolyo";

if (strpos($iban, "BOVB") !== false) return "BankOfVeni";

return "wrong";

}

TRANSACTION функции:

function sendFunds($senderIBAN, $recepientIBAN, $amount, $reason, $currency){

///Check if user submitted his own IBAN

if ($senderIBAN == $recepientIBAN) return "<script>alert('Грешка: Не можеш да си пращаш пари');location='sendFundsInterface.php?iban=$senderIBAN';</script>";

//Check if amount is 0 or negative

if ($amount <= 0) return "<script>alert('Грешка: Трябва да пратиш положителна сума');location='sendFundsInterface.php?iban=$senderIBAN';</script>";

$recepientBank = bankName($recepientIBAN);

//Check if the IBAN exists in the written names

if ($recepientBank == "wrong") return "<script>alert('Грешка: Не съществува такъв IBAN');location='sendFundsInterface.php?iban=$senderIBAN';</script>";

include("Connect.php");

$query = "SELECT \* FROM $recepientBank WHERE IBAN='$recepientIBAN';";

if(mysqli\_query($conn, $query)){

//Get what the database has answered

$result = $conn->query($query);

//If there isn't an entry with this IBAN,

//disconnect from the database and return

if ($result->num\_rows <= 0)

{

Disconnect($conn);

return "<script>alert('Грешка: Не съществува такъв IBAN');location='sendFundsInterface.php?iban=$senderIBAN';</script>";

}

//Check if Sender has enough money

$senderBank = bankName($senderIBAN);

$senderBal = fetchValue("IBAN", $senderIBAN, $senderBank, "Balance");

if($senderBal < $amount) return "<script>alert('Грешка: Недостатъчни средства');location='sendFundsInterface.php?iban=$senderIBAN';</script>";

//Update sender balance

$senderBal -= $amount;

$sql = "UPDATE $senderBank SET Balance=$senderBal WHERE IBAN='$senderIBAN';";

$conn->query($sql);

//Update recepient balance

$recepientBal = fetchValue("IBAN", $recepientIBAN, $recepientBank, "Balance");

$recepientBal += $amount;

$sql = "UPDATE $recepientBank SET Balance=$recepientBal WHERE IBAN='$recepientIBAN';";

$conn->query($sql);

Disconnect($conn);

Insert5("Transactions", "TransactionDate", "SenderIBAN", "RecipientIBAN", "Amount", "Note", date('y-m-d-H:i:s'), $senderIBAN, $recepientIBAN, $amount, $reason);

return "<script>alert('Успешно изпрати $amount $currency');location='sendFundsInterface.php?iban=$senderIBAN';</script>";

}

}

function transactionHistory($IBAN, $StartDate, $EndDate){

include("Connect.php");

$StartDate = date("Ymd", strtotime($StartDate));

$EndDate = date("Ymd", strtotime($EndDate . ' +1 day'));

//Sent funds query

$query = "SELECT TransactionDate, SenderIBAN, RecipientIBAN, Amount, Note FROM Transactions WHERE (SenderIBAN='$IBAN' OR RecipientIBAN='$IBAN') AND (TransactionDate >= '$StartDate' AND TransactionDate <= '$EndDate') ORDER BY TransactionDate DESC;";

if(mysqli\_query($conn, $query)){

//Get what the database answered

$result = $conn->query($query);

return $result;

}

}