**Online Banking**

**Team Members**

|  |  |
| --- | --- |
| **Name** | **Role** |
| Gokul Kanna G S | Teamlead |
| Satish Babu Guntupalli | DB lead |
| Keerthika.P | UI lead |
| Kavin.A.S | Controller lead |
| Ejas Ahmed | DB lead |

**Technology Used:**

* Asp.Net Core MVC
* Database-Sql
* EFFramework- DatabaseFirstApproach

**Modules:**

* Admin Module
* Dashboard Module
* User Module

**Admin Module:**

**Features:**

* Login & Signup.
* Crud Operation on account details-Create/Remove/Update an account.
* Detailed list of Transactions.

**User Module:**

**Features:**

* User Login Module
* Registration for a new bank account
* Forget Password

**Dashboard Module:**

**Features:**

* Account Details
* Change Password
* Fund Transfer
* Account Statement

**DataBase OverView:**

Graphical user interface

Description automatically generated with medium confidence

**Project Task Done By Teammates**

|  |  |
| --- | --- |
| **Name** | **Task** |
| Gokul Kanna.G.S | * Implementation Of Admin module * Implementation Of User Login * Implementation Of Registration part * Implementation Of Layouts * Implementation of validation |
| Keerthika.P | * Implementation of session concept * Implementation of account details * Implementation of change password * Implementation of Dashboard Layout |
| Kavin.A.S | * Implementation of Transaction * Implementation of account statement * Implement Transfer details * Registration Module-Exception Handling |
| SatishBabu Guntupalli | * Database design * Forget Password Module * Pdf generation of account statement |
| Ejas Ahmed | * Forget Password * UI assistance for dashboard |

**Project Highlights:**

* **Session Concept Implementation**

Add Session Service in Iconfiguration service in startup.cs and Configured with Pipeline.With Help of Session concept session id will be maintained for 5 minutes.

* SetSession ID in Login module

HttpContext.Session.SetString("uid",Model.Userid);

* GetSessionID

ViewBag.uid=HttpContext.Session.GetString("uid");

* RemoveSession Id in Logout module

HttpContext.Session.Remove("uid");

* **Account Statement Download:**

In Dashboard module there is an option called account statement where we can view the transaction history and download the same.

**Code:**

@section scripts

{

<script>

$("#printPDF").click(function () {

var element = document.getElementById('parentDiv');

html2pdf().from(element).set({

margin: [30, 10, 30, 10],

pagebreak: { avoid: 'tr' },

jsPDF: {

orientation: 'landscape', unit: 'pt', format: 'letter', compressPDF: true

}

}).save()

});

}

**Js Bundles to be Downloaded:**

**--------> html2pdf.bundle.min.js**

html2pdf.js converts any webpage or element into a printable PDF entirely client-side using html2canvas and jsPDF.

**--------> site.js**

Site.js. Develop, test, and deploy the secure static or dynamic personal web site with zero configuration. Static site generator. Decentralized.

**Fund Transaction :**

#region TransactionDetails

public IActionResult transfer()

{

ViewBag.uid = HttpContext.Session.GetString("uid");

string id = ViewBag.uid;

var q = (from p in db.AccountDetails

where p.Userid == id

select p.AccNumber).FirstOrDefault();

ViewBag.FromAccNumber = q;

return View();

}

[HttpPost]

public IActionResult transfer(transfernet net)

{

TransferDetail trans = new TransferDetail();

AccountDetail account = new AccountDetail();

ViewBag.uid = HttpContext.Session.GetString("uid");

net.Userid = ViewBag.uid;

var q = (from p in db.AccountDetails

where p.Userid== net.Userid

select p.AccNumber).FirstOrDefault();

net.FromAccNumber = q;

trans.FromAccNumber = q;

ViewBag.FromAccNumber = net.FromAccNumber;

var a = from b in db.AccountDetails

where b.AccNumber == net.ToAccNumber && net.ToAccNumber != net.FromAccNumber

select b;

if (a.Any())

{

trans.ToAccNumber = net.ToAccNumber;

}

else

{

ViewBag.Message = "Invalid To Account Number";

return View();

}

var c = from b in db.AccountDetails

where b.AccNumber == net.FromAccNumber && b.Balance >= net.TransferAmount

select b;

if (c.Any())

{

trans.TransferAmount = net.TransferAmount;

}

else

{

ViewBag.Message = "Balance Insufficient in Account";

return View();

}

trans.TransactionDate = net.TransactionDate;

trans.TypeOfTransaction = net.TypeOfTransaction.ToString();

trans.ModeOfTransaction = net.ModeOfTransaction;

var bal1 = (from o in db.AccountDetails

where o.AccNumber == net.FromAccNumber

select o.Balance).FirstOrDefault();

bal1 = decimal.Subtract(bal1, net.TransferAmount);

var bal2 = (from o in db.AccountDetails

where o.AccNumber == net.ToAccNumber

select o.Balance).FirstOrDefault();

bal2 = decimal.Add(bal2, net.TransferAmount);

var toacc\_userid = (from o in db.AccountDetails

where o.AccNumber == net.ToAccNumber

select o.Userid).FirstOrDefault();

if (c.Any())

{

var user = new AccountDetail() { Userid = net.Userid, Balance = bal1 };

db.AccountDetails.Attach(user);

db.Entry(user).Property(x => x.Balance).IsModified = true;

db.SaveChanges();

}

else

{

ViewBag.Message = "Balance not updated";

}

if (c.Any())

{

var user = new AccountDetail() { Userid = toacc\_userid, Balance = bal2 };

db.AccountDetails.Attach(user);

db.Entry(user).Property(x => x.Balance).IsModified = true;

db.SaveChanges();

}

else

{

ViewBag.Message = "Balance not updated";

}

trans.Balance = bal1;

trans.CurrentBalance = bal1;

trans.DepositAmount = 0;//since not null ---statically given random value

trans.WithdrawAmount = 0;//since not null ---statically given random value

trans.Userid=net.Userid;

trans.ModeOfTransaction = "online";

trans.TransactionDate = DateTime.Now;

db.TransferDetails.Add(trans);

db.SaveChanges();

return RedirectToAction("Index");

}

}

}

#endregion

**HomePage:**

Schematic

Description automatically generated

Dashboard:

Graphical user interface, application

Description automatically generated

**Dashboard Layout code:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>@ViewData["Title"] - SPAN Bank</title>

@RenderSection("head", required: false)

<link rel="stylesheet" href="~/lib/bootstrap/dist/css/bootstrap.min.css" />

<link rel="stylesheet" href="~/css/site.css" />

</head>

<body>

<header>

<nav class="navbar navbar-expand-sm navbar-toggleable-sm navbar-light bg-dark border-bottom box-shadow mb-3" style="background-color:gray">

<div class="container">

<a class="navbar-brand text-light" **asp-area**="" **asp-controller**="Home" **asp-action**="Index"><img src="/image/logo0.jpeg" class="img-center" height="40" width="40" /> SPAN BANK</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target=".navbar-collapse" aria-controls="navbarSupportedContent"

aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<ul class="navbar-nav flex-grow-1">

<li class="nav-item">

<a class="nav-link text-light" **asp-area**="" **asp-controller**="Dashboard" **asp-action**="AccountDetails">Account Details</a>

</li>

<li class="nav-item">

<a class="nav-link text-light" **asp-area**="" **asp-controller**="Transaction" **asp-action**="transfer">Fund Transfer</a>

</li>

<li class="nav-item">

<a class="nav-link text-light" **asp-area**="" **asp-controller**="Dashboard" **asp-action**="ChangePassword">Change Password</a>

</li>

<li class="nav-item">

<a class="nav-link text-light" **asp-area**="" **asp-controller**="Dashboard" **asp-action**="Statement">Account Statement</a>

</li>

</ul>

</div>

<div class="topnav-right">

<ul class="navbar-nav flex-grow-1">

<li class="nav-item"> <a class="nav-link text-light" **asp-area**="" **asp-controller**="Dashboard" **asp-action**="AccountDetails">@ViewBag.uid</a></li>

<li class="nav-item"> <a class="nav-link text-light" **asp-area**="" **asp-controller**="Home" **asp-action**="Index"> Logout </a></li>

<li class="nav-item"> <a class="nav-link text-light" **asp-area**=""> <img src="/image/Profilepic.jpeg" class="img-center" height="30" width="30" /></a></li>

</ul>

</div>

</nav>

</header>

<div class="container">

<main role="main" class="pb-3">

@RenderBody()

</main>

</div>

<footer class="border-top footer text-muted">

<div class="container">

&copy; 2021 - Span - <a **asp-area**="" **asp-controller**="Home" **asp-action**="Privacy">Privacy</a>

</div>

</footer>

<script src="~/lib/jquery/dist/jquery.min.js"></script>

<script src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>

<script **src**="~/js/site.js" **asp-append-version**="true"></script>

<script src="~/js/html2pdf.bundle.min.js"></script>

@await RenderSectionAsync("Scripts", required: false)

</body>

</html>