Julianna Bicz

**HDIP Project**

Research document

Table of Contents

[Application Area 2](#_Toc826688)

[Diagram pictured Web Application functionality: 3](#_Toc826689)

[A comparison with similar systems 3](#_Toc826690)

[Database 4](#_Toc826691)

[Considered implementation Technologies 5](#_Toc826692)

[Comparison between .Net Framework and .Net Core: 6](#_Toc826693)

[Overview of proposed system architecture 7](#_Toc826694)

[Links of used pages 8](#_Toc826695)

# Application Area

Beauty Therapist finder – the application will enable to find and rate beauty therapist. There are many applications, websites etc where rate or comment can be given. From restaurants, dentists, beauty salons or even mechanist. The only issue is that, mostly we can check ranking by the place, not by the people. It’s common that work of some part of the team will make good or bad rating for whole places, even for future workers. The aim of the application is to find the good specialist, and after that choose the place.

Authentication with Facebook or Gmail will be needed to create account on Application. It can help to keep eventual bulling content in minimum. Another step through making sure that profile of person is real, is to award user activity (points for adding profile photo, comments, rating).

User have 2 options of register and then log in:

1. Beauty therapist

Must add personal details like name, city, skills. Google/Facebook authentication

1. User/Seeker

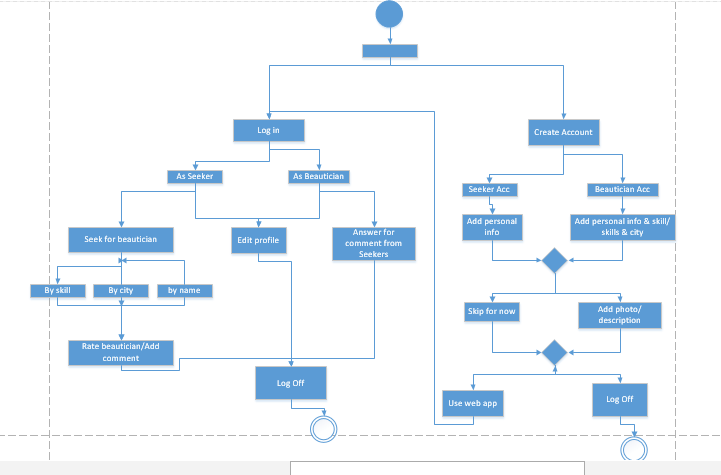
Must add person details like name. Google/Facebook authentication.

Not log in users can only view sample of application, can’t add comment or rating.

There will be also Admin user log create, not accessible for users.

One user can add 1 comment and rate for Therapist. Profanity filter will be added, but also both log in users will be able to report content to Admin.

### Diagram pictured Web Application functionality:



# A comparison with similar systems

The goal of the application is to find in the simplest way therapist with confirm by other client skills.

There are many other systems with similar functionality, which in the simplest way can point the right specialist for the user.

1. **Find your nearest doctor (findmydoctor.pk)**

Find your doctor is simple application. Start website shows search rows, using only 2 filters: City and Speciality.  
Searching output shows list of doctors (example cardiologist) with personal details (name, contact number, address) and timing.  
Biggest minus of application is inability to add any filters after search – it shows numerous pages of names and some details.

1. **Helping (helpling.ie)**

Helping is web application searching for cleaners in Ireland. After choosing preferred location application outputs cleaners accounts with rating, schedule and price. Log in required to proceed with booking, user doesn’t have to be log in to search for the content

1. **Znany lekarz (znanylekarz.pl)**

Znany lekarz is polish web application searching for a doctor. There is one type of log in – for doctors. Users can search, comment and rate without log in. Its possible to search with parameters like city, name or type of specialisation.

Database   
Database will be create using entity framework code first approach.

Why code first?

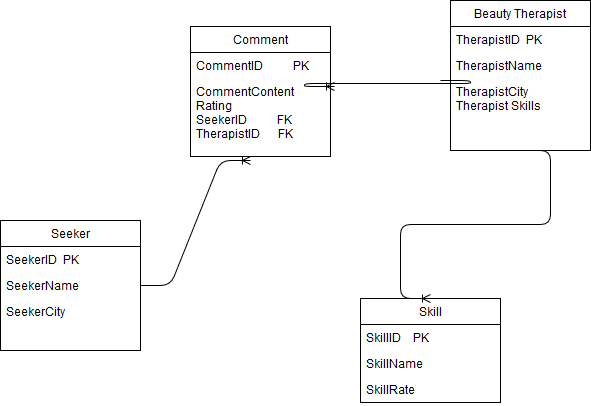
* Creating the data base will be much faster – Created code will be used for Database and model in MVC Web application. It’s easier and cleaner.
* It gives more control on data. Creating code first gives ability to specify relationships between classes, associations and contains.
* Automatic migrations are very useful, which makes data base actualization much easier. There is no need to create incremental scripts.

Because of changes history it’s easy to comeback to previous state

* Easier to build integration tests

There will be 4 tables create:

1. Seeker
2. Beauty Therapist
3. Skill
4. Comment



.

# **Considered implementation Technologies**

#### **ASP.Net**

*“ASP.NET is an open-source server-side web application framework designed for web development to produce dynamic web pages.”* (ASP.NET at Curlie)

ASP.NET is fully undegradable with Microsoft Azure, which provide backend integration with server and database. It has been chosen also to deliver view of the full browser web page, including user authentication

#### **Razor pages**

*“ASP.NET Core Razor Pages is a page-focused framework for building dynamic, data-driven web sites with clean separation of concerns.” (Learnrazorpages.com)*

Razor Pages are an easier way to organized code within ASP.NET applications, keeping implementation view models and logic closer to the view implementation code. It’s more secure and easier for Unit Test.

#### **Git**

*Git is open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. (git-scm.com)*

Git is a distributed version control software that allows local repositories to be branched from the master repository, so users can have several repositories at the same time

#### **HTML 5**

HTML5 was developed to provide flexibility and allow more to be accomplished with standalone HTML.

### **Comparison between .Net Framework and .Net Core:**

Main advantages of using .Net Framework:

• There is more 3rd party libraries or NuGet packages not available in .Net Core.

The NuGet ecosystem includes client tools that provide the ability to produce and consume packages. It also has a central package repository for package authors and consumers. It is available as a Visual Studio extension.

• It supports more technologies, for example:

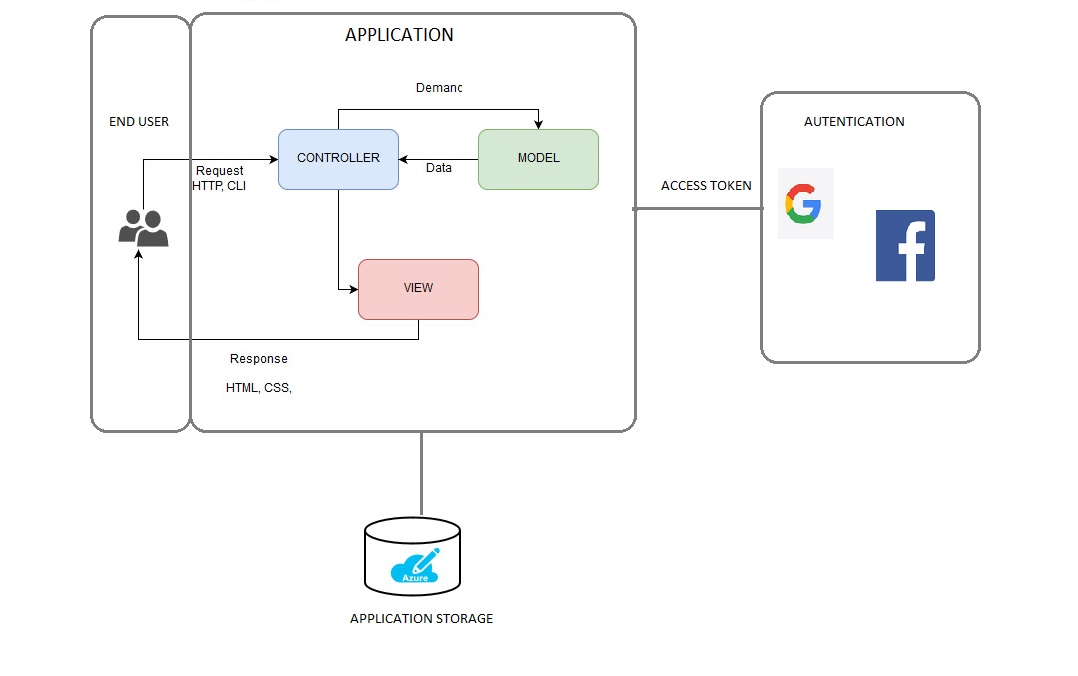
* Windows Presentation Foundation
* ASP.NET Web Forms applications
* ASP.NET Web Pages applications
* WCF services implementation
* Windows Workflow Foundation

Main advantages of using .Net Core:

• Multiple services can be run across multiple platforms (like Windows, Linus, macOS).

• Microservices (form of Service Oriented Architecture) are being used. Each of the services can run unique process, be deployed independently and be created in different programming applications.

# Overview of proposed system architecture

The applications will be implemented using an MVC architecture with the ASP.NET framework.

*“MVC is a design pattern used to decouple user-interface (view), data (model), and application logic (controller). This pattern helps to achieve separation of concerns.” (Microsoft.com)*

*“Microsoft Azure is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications and services through Microsoft-managed data centers.” (Microsoft.com)*

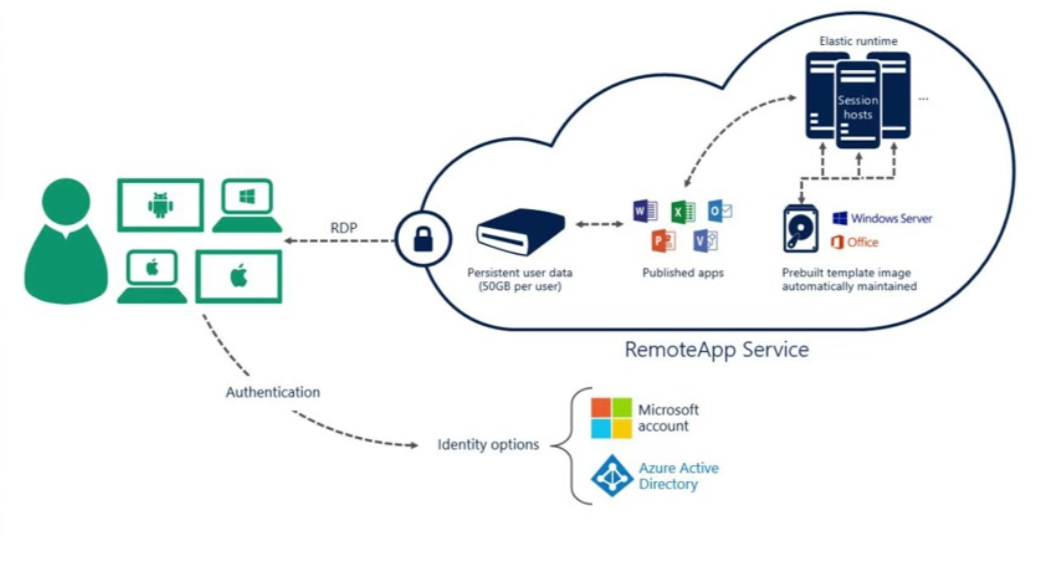
SQL Database will be deployed on Microsoft Azure.

To Create SQL Database form some values need to be provide: (Database name, **Subscription Resource group, Select source)**

Created empty SQL data base on Azure it will be connect to MVC application (containing local database – create from code). Connection is implemented by pasting connection string from Microsoft Azure to Web.config file in web application.

The Azure SQL Database Deployment task is the primary mechanism to deploy a database to Azure. This task, as with other built-in Azure tasks, requires an Azure service connection as an input. The Azure service connection stores the credentials to connect from Azure Pipelines or TFS to Azure.

Azure allows for quick and simple deploy and the update. It’s very useful to connect azure account with GitHub account and set it to automatic update



## Links of used pages

<https://stackoverflow.com/questions/46777404/why-is-razor-pages-the-recommended-approach-to-create-a-web-ui-in-asp-net-core-2>

<https://stackify.com/asp-net-razor-pages-vs-mvc/>

<http://james.newtonking.com/archive/2009/07/03/simple-net-profanity-filter>

<https://stackify.com/net-core-vs-net-framework/>

<https://nofluffjobs.com/blog/net-developer-zarobki-w-najwiekszych-polskich-miastach/>

<https://www.spidersweb.pl/2016/06/microsoft-dotnet-core.html>

<https://www.reddit.com/r/dotnet/comments/5r3iza/why_i_decided_to_abandon_dotnet_core_and_return/>

<https://www.tutorialspoint.com/uml/uml_activity_diagram.htm>

<http://youcc-technologies.net>