Faculty of Computers, Informatics and Microelectronics

Technical University of Moldova

SM

Laboratory work #5

Web technologies.

Authors:
Daniel Calancea

Supervisor:
Doina Bisericanu

Laboratory work #4

1 Purpose of the laboratory

Familiarization with web technologies, frameworks and languages that are used to make web applications.

2 Laboratory Work Requirements

Create a web application.

3 Laboratory work implementation

3.1 Web application

In computing, a web application or web app is a client–server computer program which the client (including the user interface and client-side logic) runs in a web browser.[1] Common web applications include webmail, online retail sales, online auctions, wikis, instant messaging services and many other functions.

The general distinction between a dynamic web page of any kind and a "web application" is unclear. Web sites most likely to be referred to as "web applications" are those which have similar functionality to a desktop software application, or to a mobile app. HTML5 introduced explicit language support for making applications that are loaded as web pages, but can store data locally and continue to function while offline.

Single-page applications are more application-like because they reject the more typical web paradigm of moving between distinct pages with different URLs. Single-page frameworks like Sencha Touch and AngularJS might be used to speed development of such a web app for a mobile platform.

Mobile web applications Further information: Multiple phone web-based application framework There are several ways of targeting mobile devices when making a web application:

Responsive web design can be used to make a web application - whether a conventional web site or a single-page application viewable on small screens and work well with touchscreens. Progressive Web Apps are a hybrid of regular web pages (or websites) and a mobile application. Native apps or "mobile apps" run directly on a mobile device, just as a conventional software application runs directly on a desktop computer, without a web browser (and potentially without the need for Internet connectivity); these are typically written in Java (for Android devices) or Objective-C or Swift (for iOS devices). Recently, frameworks like React Native, Flutter and Xamarin allow the development of native apps for all platforms using languages other than each standard native language. Hybrid apps embed a mobile web site inside a native app, possibly using a hybrid framework like Apache Cordova and Ionic or Appcelerator Titanium. This allows development using web technologies (and possibly directly copying code from an existing mobile web site) while also retaining certain advantages of native apps (e.g. direct access to device hardware, offline operation, app store visibility).

3.2 Implementation

The web application was implemented in ASP.NET C# using MVC pattern. The application has the following functionality:

- Create a local database on server where user data will be stored.
- Endpoints that are implemented using MVC.
- Embedded video show.

Figure 3.1 – Main page.

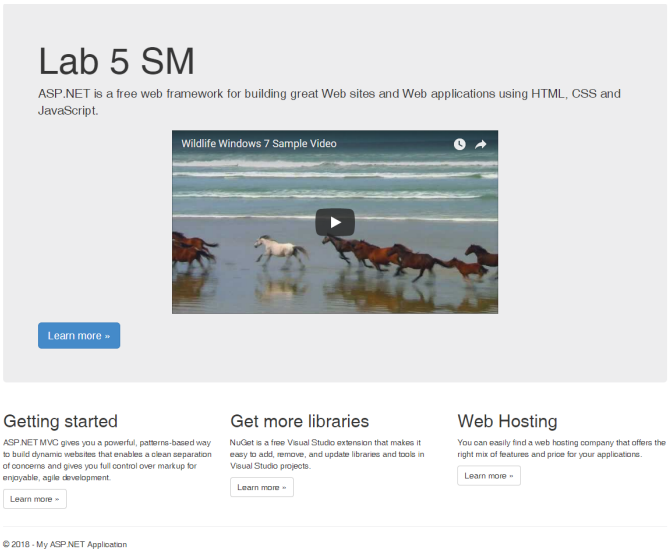


Figure 3.2 – Register page.

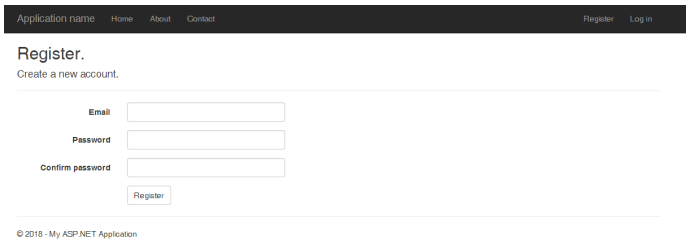
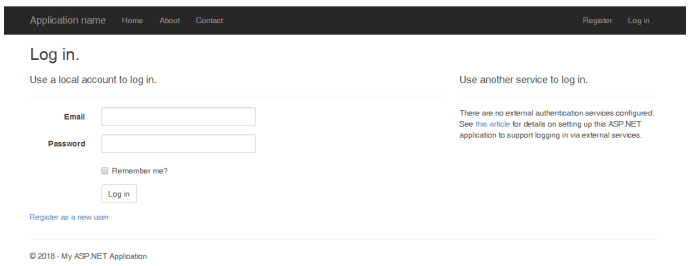


Figure 3.3 – Log in page.



Conclusions

In this laboratory work we got familiarized with web technologies and saw how convenient web applications can be, since they are available on every platform without any drawbacks.