# ASP.NET MVC Exam (October 2015) – Snippy

You are assigned to implement a Web-based system for saving and sharing **code snippets**. The system consists of server-side logic (back-end) and client-side logic (front-end); Use C#, ASP.NET MVC, Entity Framework and Microsoft SQL Server for the back-end and HTML, CSS, Bootstrap, JavaScript and AJAX for the front-end.

## Data Model

* The system contains information about **users**, **snippets**, **languages**, **labels** and **comments**.
* Users can register in the system. After having successfully registered, users have a **username**, **email** and **password**. The username must be unique – there can be no two users with the same username. The password should never be stored as plain text. Users also have some **snippets** and **comments**.
* **Code snippets** have **title**, **description** (short information about the code, its purpose and usage, etc.), **code** (the code snippet itself), **language**, **author**, **creation time**, **some labels** (optional) and **comments**.
* **Programming languages** are identified in the system by their **name** and they have a set of snippets.
* **Labels** have **text** and **snippets**.
* **Comments** have **author**, **creation time** and **content** and are defined for a **snippet**.
* Users can view snippets, add new snippets and edit their own. They can also comment on existing snippets.

## Functionality

* **Guests** (anonymous users) can register an account with their own username, email and password.
* **Guests** can login by username and password.
* **Logged-in users** can logout.
* **Guests** can view the home page. It holds information about the 5 latest snippets (ordered from the latest to earliest), 5 latest comments (ordered from the latest to earliest) and 5 best labels (labels with the most snippets).
* **Guests** can view all snippets in the site (ordered from the latest to earliest, with paging)
* **Guests** can view snippet details (title, description, code, labels and comments).
* **Guests** can view some details about labels and languages – the titles and descriptions of their snippets.
* **Logged-in users** can add comments to existing snippets.
* **Logged-in users** can delete their own comments.
* **Logged-in users** can create new snippets and edit any snippets they have created.
* Display a **notification message** after each successful operation which changes the database.

## Design the Database

Design **entity classes** and create a **database** to hold the users, snippets, languages, labels and comments. Use Entity Framework code-first. Put your data layer in a separate Visual Studio project (class library).

The data layer should be implemented with some abstraction:

* You can use repository pattern and unit of work.
* You can use repositories and data service layer with database queries.
* You can use only data services layer on the top of the EF context.

10 score

Use **automatic migrations**. At application start-up, **fill some sample data** in the database.

Use the following data:

#### Snippets:

Bottom of Form

**Title:** Ternary Operator Madness  
**Description:** This is how you DO NOT user ternary operators in C#!  
**Code:**

bool X = Glob.UserIsAdmin ? true : false;  
**Author:** admin  
**Creation time:** 26.10.2015 17:20:33  
**Language:** C#  
**Labels:** funny

**Title:** Points Around A Circle For GameObject Placement  
**Description:** Determine points around a circle which can be used to place elements around a central point  
**Code:**

private Vector3 DrawCircle(float centerX, float centerY, float radius, float totalPoints, float currentPoint)

{

float ptRatio = currentPoint / totalPoints;

float pointX = centerX + (Mathf.Cos(ptRatio \* 2 \* Mathf.PI)) \* radius;

float pointY = centerY + (Mathf.Sin(ptRatio \* 2 \* Mathf.PI)) \* radius;

Vector3 panelCenter = new Vector3(pointX, pointY, 0.0f);

return panelCenter;

}  
**Author:** admin  
**Creation time:** 26.10.2015 20:15:30  
**Language:** C#  
**Labels:** geometry, games

**Title:** forEach. How to break?  
**Description:** Array.prototype.forEach You can't break forEach. So use "some" or "every". Array.prototype.some some is pretty much the same as forEach but it break when the callback returns true. Array.prototype.every every is almost identical to some except it's expecting false to break the loop.  
**Code:**

var ary = ["JavaScript", "Java", "CoffeeScript", "TypeScript"];

ary.some(function (value, index, \_ary) {

console.log(index + ": " + value);

return value === "CoffeeScript";

});

// output:

// 0: JavaScript

// 1: Java

// 2: CoffeeScript

ary.every(function(value, index, \_ary) {

console.log(index + ": " + value);

return value.indexOf("Script") > -1;

});

// output:

// 0: JavaScript

// 1: Java  
**Author:** newUser  
**Creation time:** 27.10.2015 10:53:20  
**Language:** JavaScript  
**Labels:** jquery, useful, web, front-end

**Title:** Numbers only in an input field  
**Description:** Method allowing only numbers (positive / negative / with commas or decimal points) in a field  
**Code:**

$("#input").keypress(function(event){

var charCode = (event.which) ? event.which : window.event.keyCode;

if (charCode <= 13) { return true; }

else {

var keyChar = String.fromCharCode(charCode);

var regex = new RegExp("[0-9,.-]");

return regex.test(keyChar);

}

});  
**Author:** someUser  
**Creation time:** 28.10.2015 09:00:56  
**Language:** JavaScript  
**Labels:** web, front-end

**Title:** Create a link directly in an SQL query  
**Description:** That's how you create links - directly in the SQL!  
**Code:**

SELECT DISTINCT

b.Id,

concat('<button type=""button"" onclick=""DeleteContact(', cast(b.Id as char), ')"">Delete...</button>') as lnkDelete

FROM tblContact b

WHERE ....  
**Author:** admin  
**Creation time:** 30.10.2015 11:20:00  
**Language:** SQL  
**Labels:** bug, funny, mysql

**Title:** Reverse a String  
**Description:** Almost not worth having a function for...  
**Code:**

def reverseString(s):

"""Reverses a string given to it."""

return s[::-1]  
**Author:** admin  
**Creation time:** 26.10.2015 09:35:13  
**Language:** Python  
**Labels:** useful

**Title:** Pure CSS Text Gradients  
**Description:** This code describes how to create text gradients using only pure CSS  
**Code:**

/\* CSS text gradients \*/

h2[data-text] {

position: relative;

}

h2[data-text]::after {

content: attr(data-text);

z-index: 10;

color: #e3e3e3;

position: absolute;

top: 0;

left: 0;

-webkit-mask-image: -webkit-gradient(linear, left top, left bottom, from(rgba(0,0,0,0)), color-stop(50%, rgba(0,0,0,1)), to(rgba(0,0,0,0)));  
**Author:** someUser  
**Creation time:** 22.10.2015 19:26:42  
**Language:** CSS  
**Labels:** web, front-end

**Title:** Check for a Boolean value in JS  
**Description:** How to check a Boolean value - the wrong but funny way :D  
**Code:**

var b = true;

if (b.toString().length < 5) {

//...

}  
**Author:** admin  
**Creation time:** 22.10.2015 05:30:04  
**Language:** JavaScript  
**Labels:** funny

#### Comments:

**Content:** Now that's really funny! I like it!  
**Author:** admin  
**Creation time:** 30.10.2015 11:50:38  
**Snippet:** Ternary Operator Madness

**Content:** Here, have my comment!  
**Author:** newUser  
**Creation time:** 01.11.2015 15:52:42  
**Snippet:** Ternary Operator Madness

**Content:** I didn't manage to comment first :(  
**Author:** someUser  
**Creation time:** 02.11.2015 05:30:20  
**Snippet:** Ternary Operator Madness

**Content:** That's why I love Python - everything is so simple!  
**Author:** newUser  
**Creation time:** 27.10.2015 15:28:14  
**Snippet:** Reverse a String

**Content:** I have always had problems with Geometry in school. Thanks to you I can now do this without ever having to learn this damn subject  
**Author:** someUser  
**Creation time:** 29.10.2015 15:08:42  
**Snippet:** Points Around A Circle For GameObject Placement

**Content:** Thank you. However, I think there must be a simpler way to do this. I can't figure it out now, but I'll post it when I'm ready.  
**Author:** admin  
**Creation time:** 03.11.2015 12:56:20  
**Snippet:** Numbers only in an input field

#### Languages:

C#, JavaScript, Python, CSS, SQL, PHP

#### Labels:

bug, funny, jquery, mysql, useful, web, geometry, back-end, front-end, games

#### Users:

**Username:** admin  
**Email:** admin@snippy.softuni.com  
**Password:** adminPass123  
**Role:** Admin

**Username:** someUser  
**Email:** someUser@example.com  
**Password:** someUserPass123

**Username:** newUser  
**Email:** new\_user@gmail.com  
**Password:** userPass123

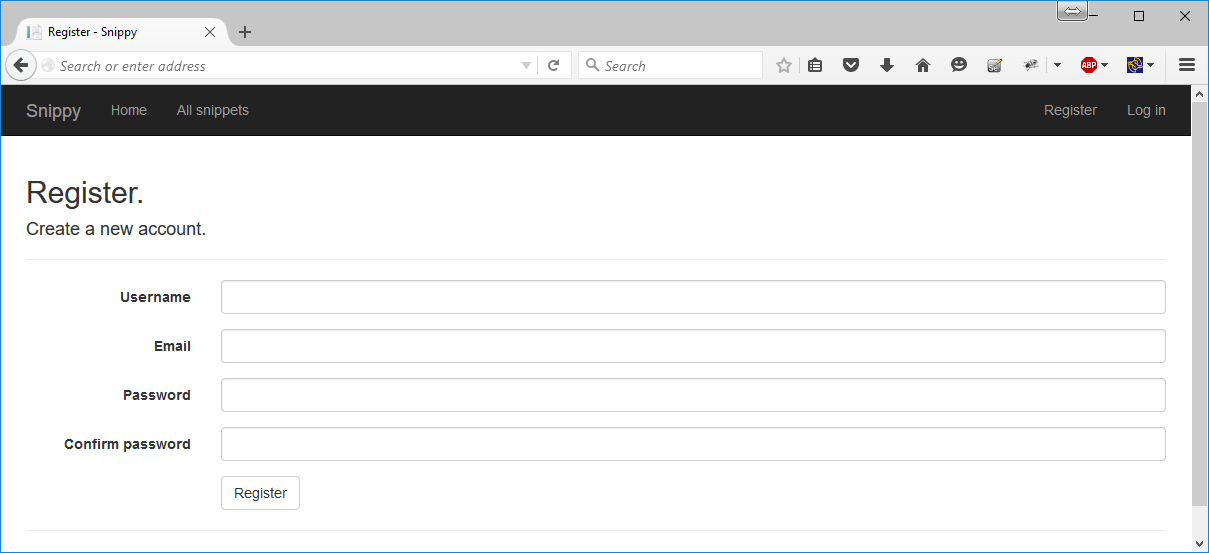
20 score

## Implement Home Page, Registration, Login and Logout

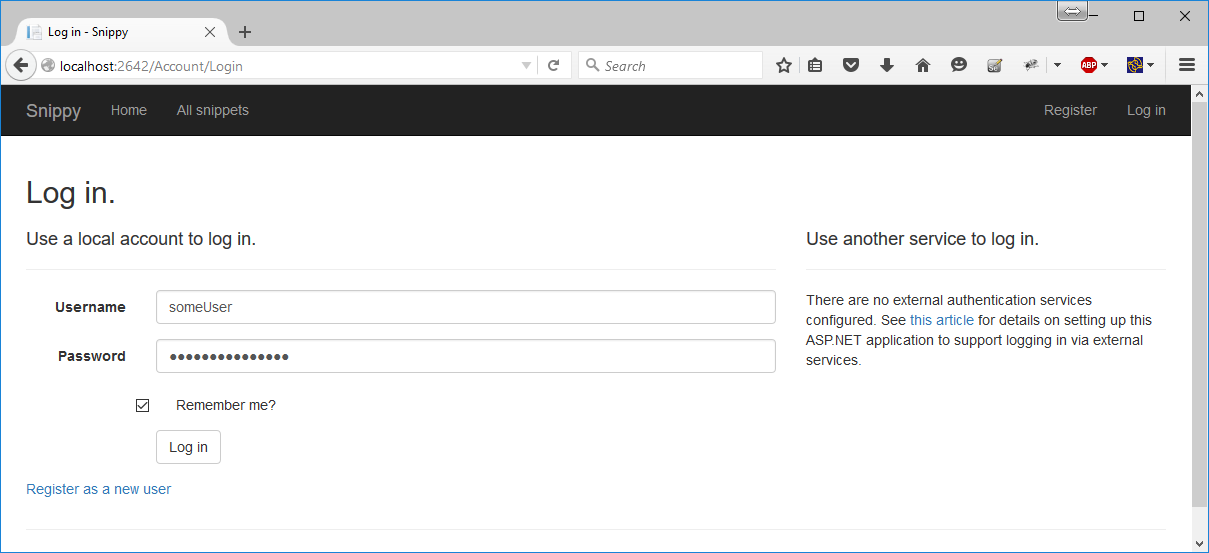
Implement user **registration**, **login** and **logout**.

* **User registration** should create a new user account and display "**User registered**" **notification message**. In case of error, display an error message. **Validate** the registration form. **All passwords must have at least one lowercase letter and one digit and should be at least 6 symbols long** (e.g. "password123").
* The **user registration form** should have **username,** **email**, **password** and **confirm password** fields (mandatory).

8 score



* **User login** should login an existing user by **username** and **password**. After login, display the site home page. In case of a problem, display an error message. **Validate** the login form.



4 score

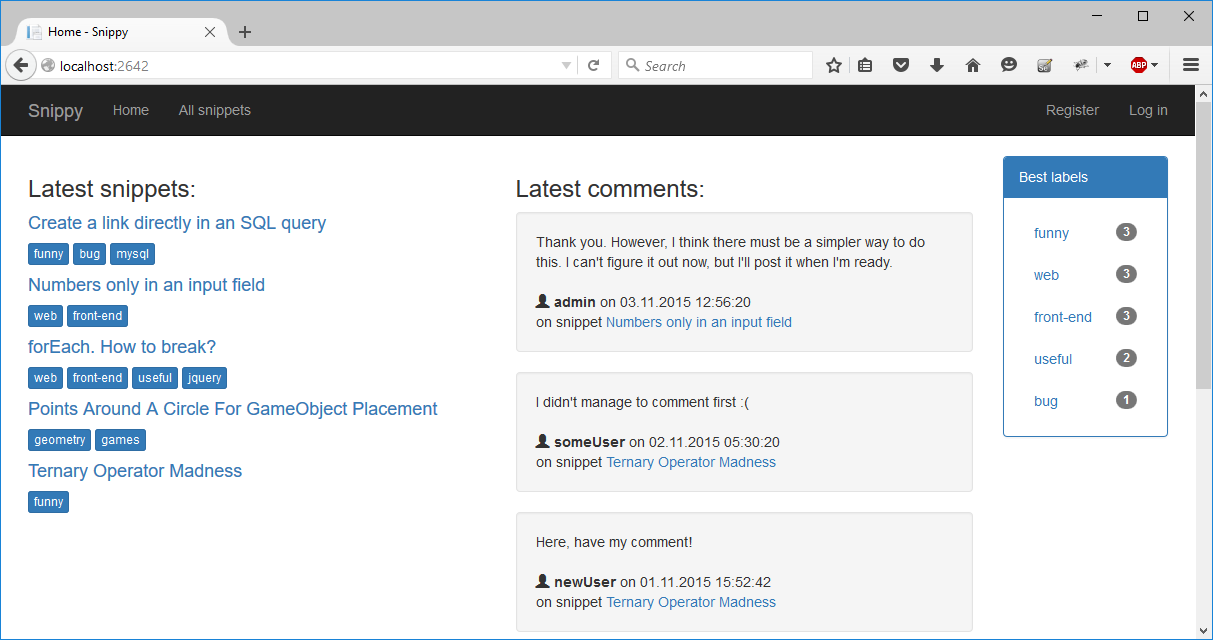
* **User logout** should invalidate user's session and redirect to the application home page.

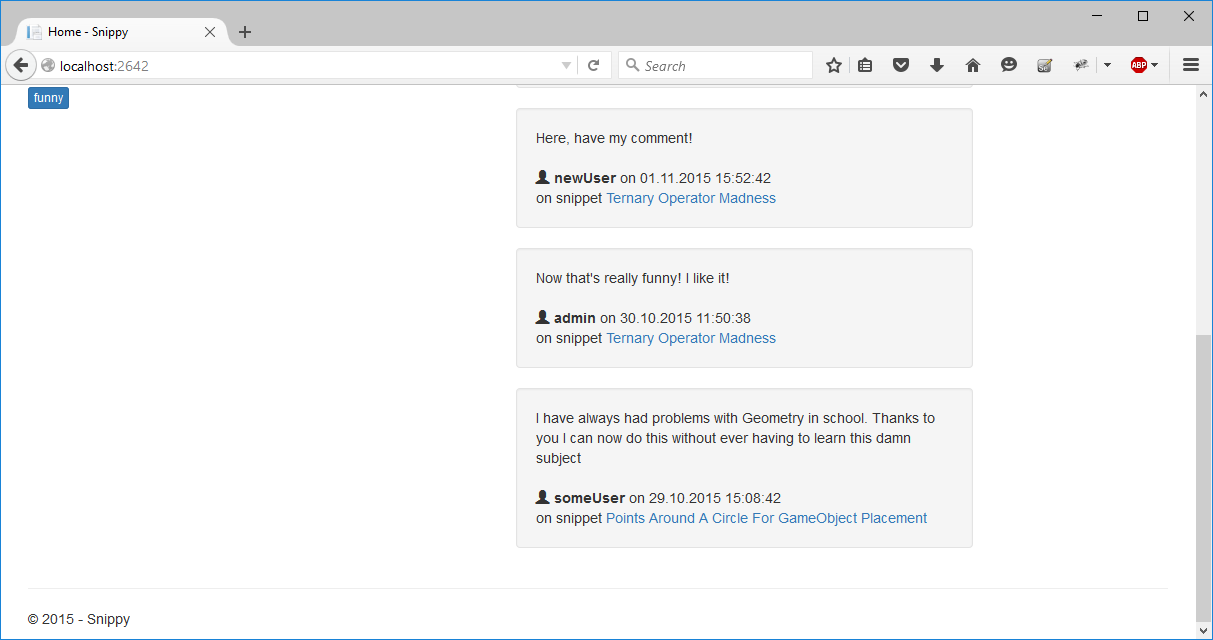
2 score

## Implement the Home Page

Anonymous users can view the **home page**.

* Display the 5 latest snippets (ordered by time of addition, from the latest to the earliest), the 5 latest comments (from the latest to earliest) and the 5 best labels (with most snippets).
* The snippet titles must be clickable. Upon clicking, the user should see the snippet details.
* The labels must be clickable. Upon clicking, the user should see the snippets for the given label.

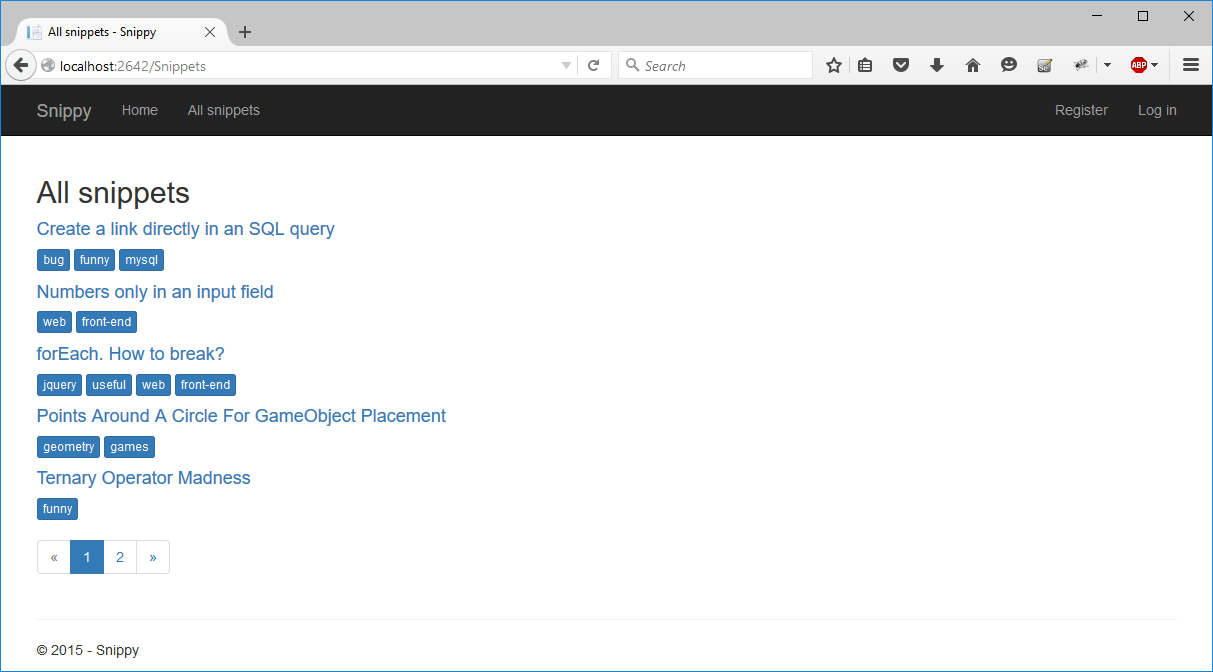




10 score

## Implement the View All Snippets Page

Anonymous users can **view all snippets**.

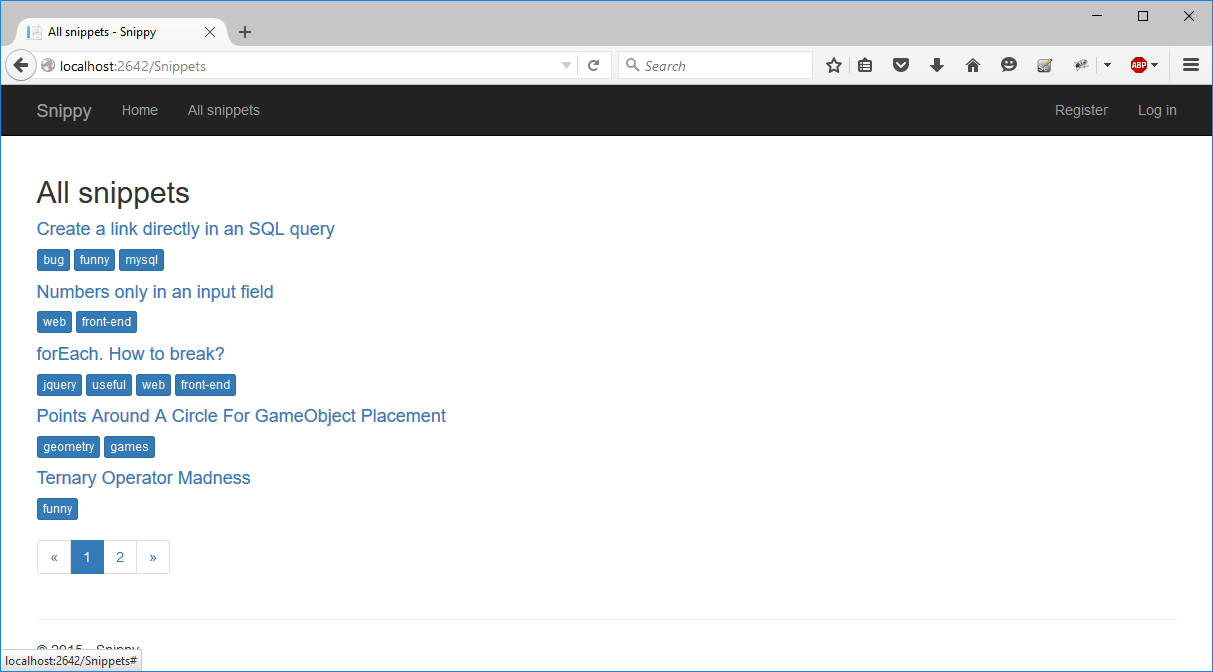


* Display all snippets, ordered by date (from the latest to the earliest).
* The snippet titles should lead to the "snippet details" page and the labels should lead to the "snippets for a label" page.

4 score

* Use pagination. Display 5 snippets per page by default. The current page should be highlighted. There should also be links to the first and last pages (« and »). When the user is on the first page, the « link should be disabled. The same applies for the last page (and »).

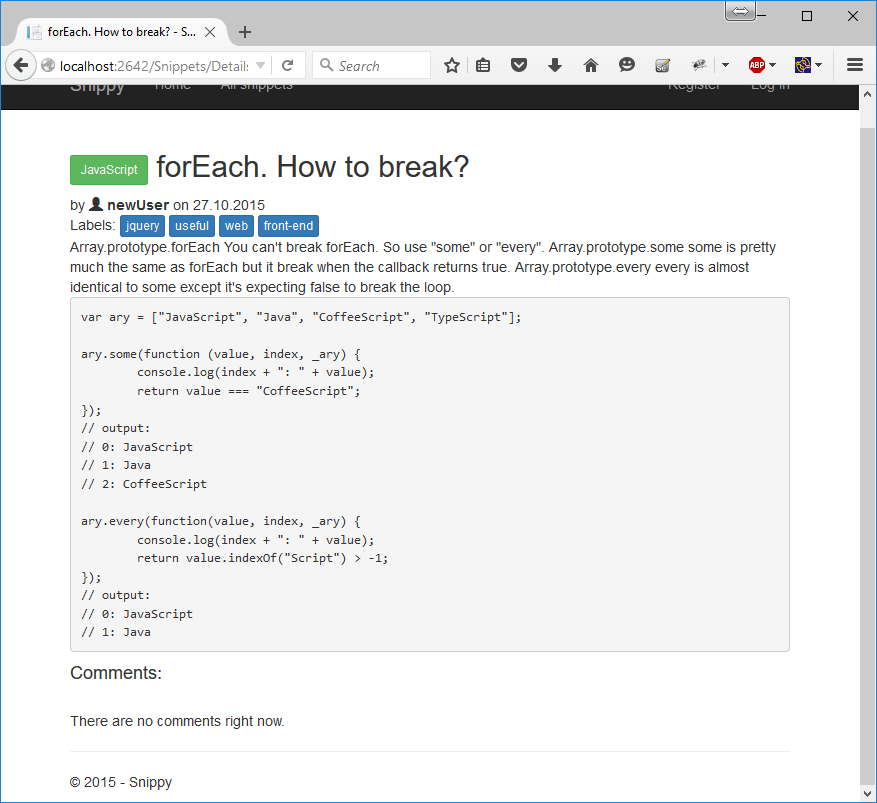
You **ARE NOT allowed** to use any JavaScript pagination plugins, HTML helpers or server-side code (such as PagedList) except Bootstrap pagination.



**BONUS:** 8 score

## Implement the Snippet Details Page

Guests can view details about all snippets.

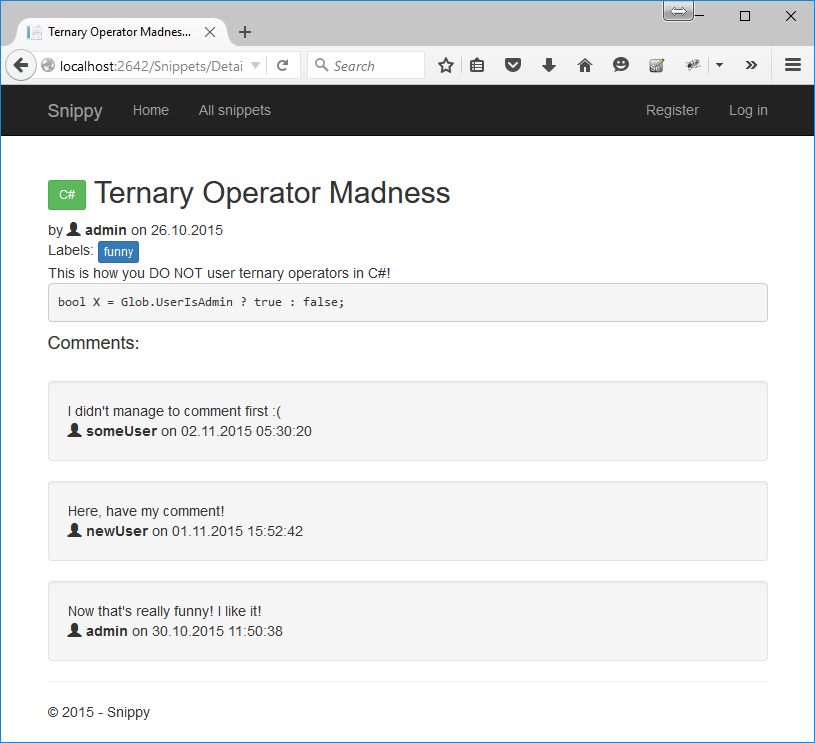


* The labels should be clickable and should display all snippets for the corresponding label. The language should also be clickable and display all snippets written in this language.

2 score

* In case there are no comments, display a message. In case there are already some comments, display them from the latest to the earliest.

2 score



## Implement Commenting to Snippets

Logged-in users can comment on any snippet.

* Implement a form to comment. The form should have comment content. Display the form only when there is a logged-in user.

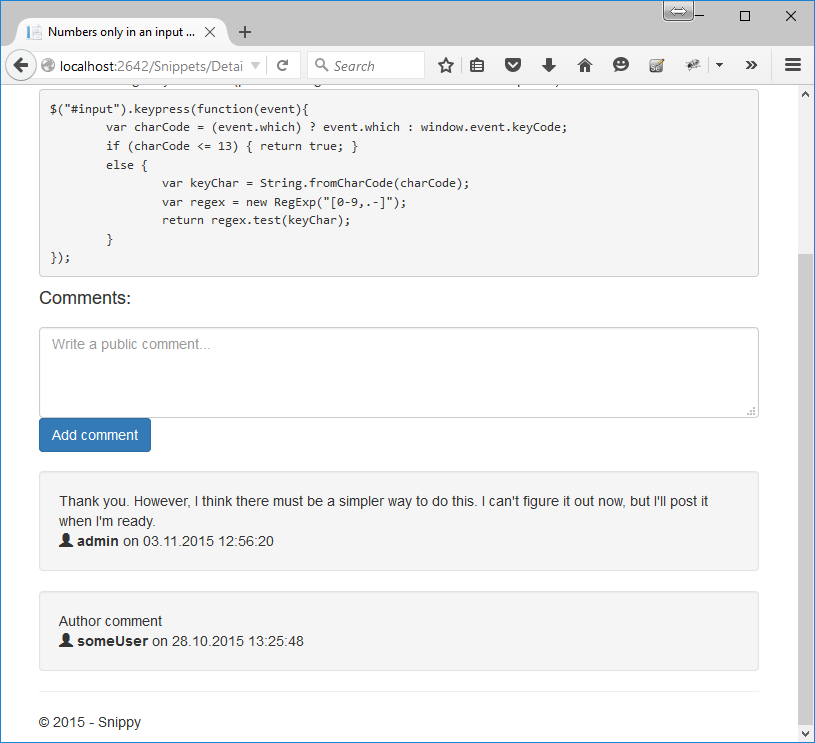
1 score

* Implement standard (non-AJAX) commenting. Display an "Add comment" button. When the user clicks the button, display a new page and allow the user to fill in their comment.

2 score

* Implement commenting using an AJAX form. When the comment has been submitted, show it at the top of all comments (since it’s the newest now, and the newest comments should be displayed at the top). Use a CSRF token to prevent request forgery attacks.
* **Note:** To get the full possible score (including the bonus score) for this problem, you will not need to have the non-AJAX form.

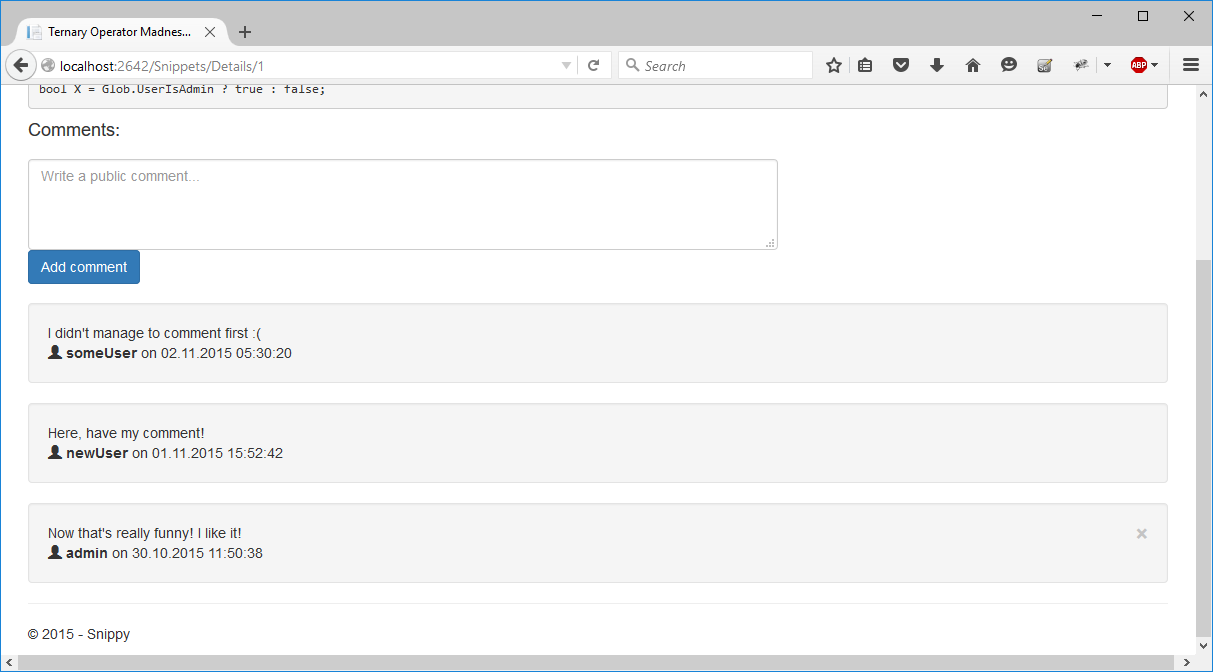
**BONUS:** 5 score

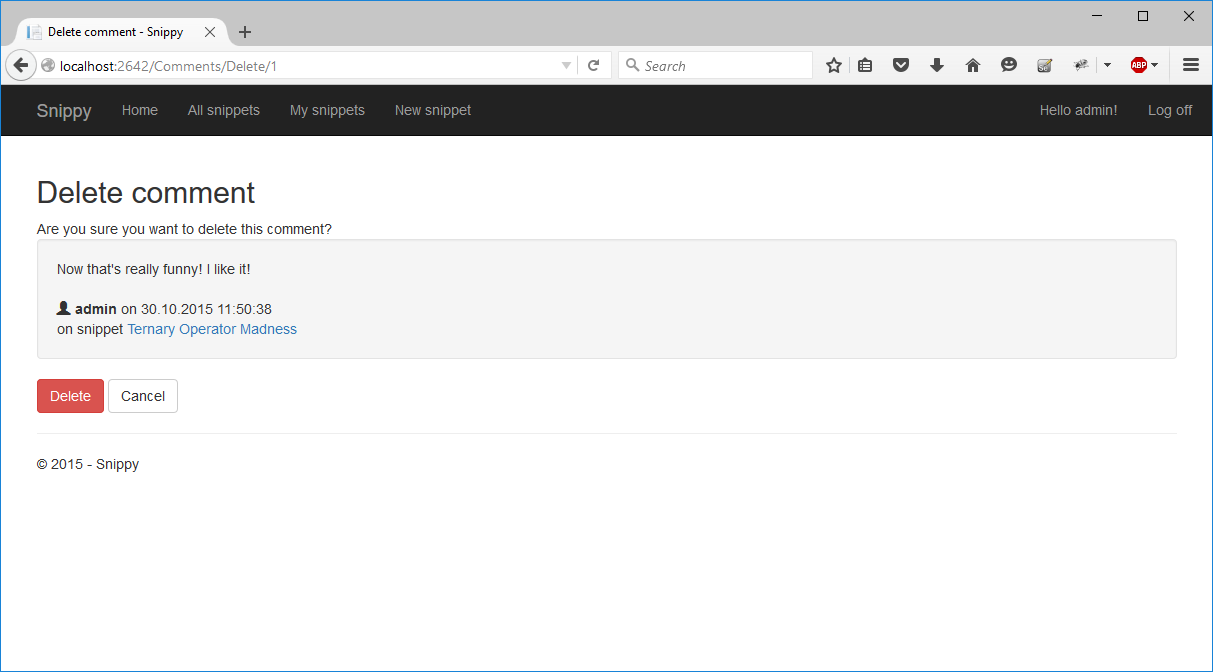


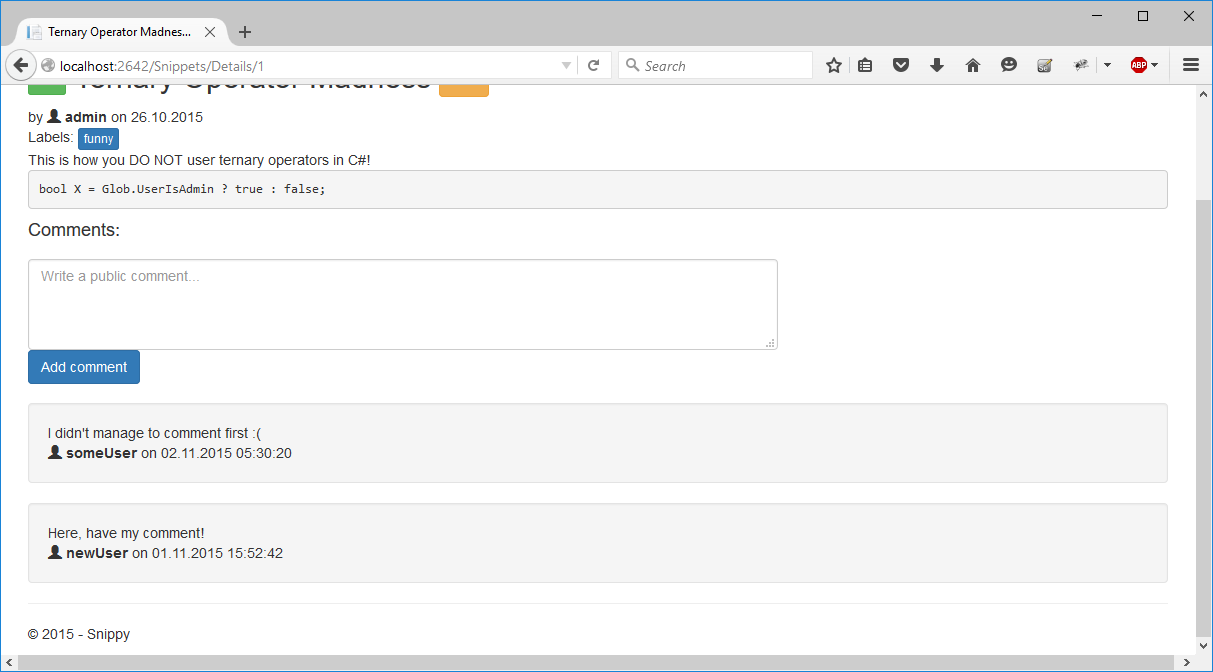
## Implement Deleting Comments

Logged in users can delete their own comments.

* Display an "X" next to each comment by the currently logged in user.
* When the user clicks the "X", display a confirmation page to ask the user whether they really want to delete the comment.
* If the user agrees, delete the comment and display the code snippet again.
* If the user disagrees, display the code snippet again without deleting the comment.







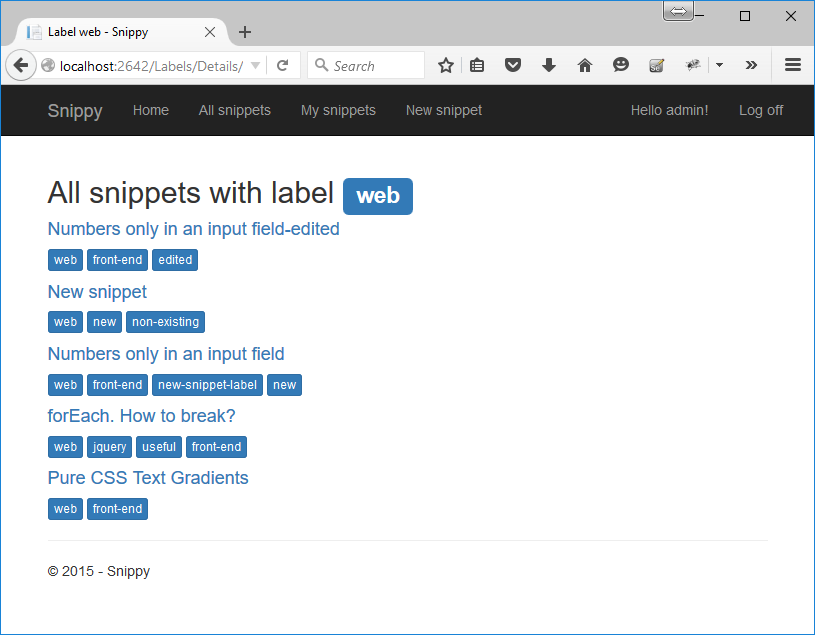
**BONUS:** 6 score

## Implement the Snippets for a Label Page

Guests can view all snippets which have a specified label.

* Display all snippets for the given label. You can omit paging if you like. The page must include the snippets’ titles and labels.

2 score

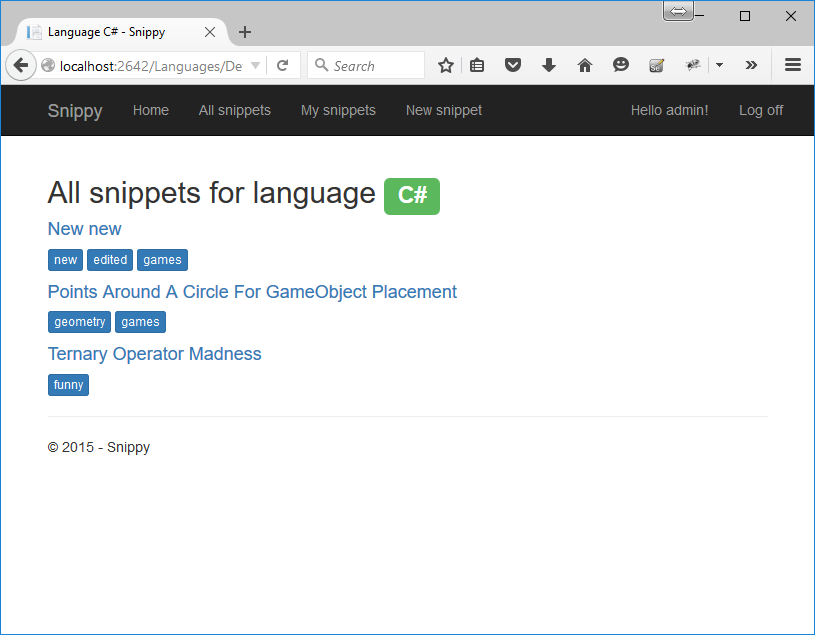


## Implement the Snippets for a Language Page

Guests can view all snippets which are written in the specified language.

* Display some details about all snippets for the given language. You can omit paging if you like. The page must include the snippets’ titles and labels.

2 score



## Implement the My Snippets Page

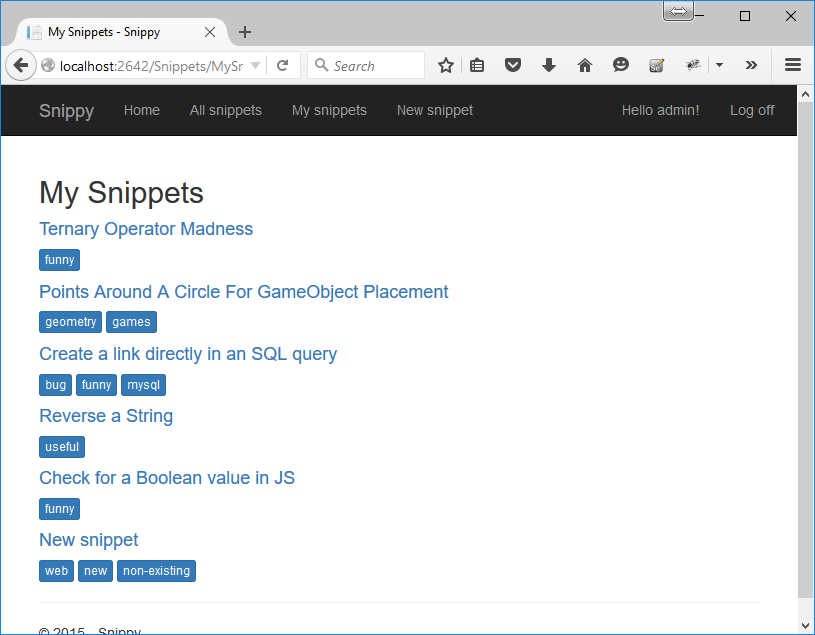
Logged-in users can view all snippets which have been created by them.

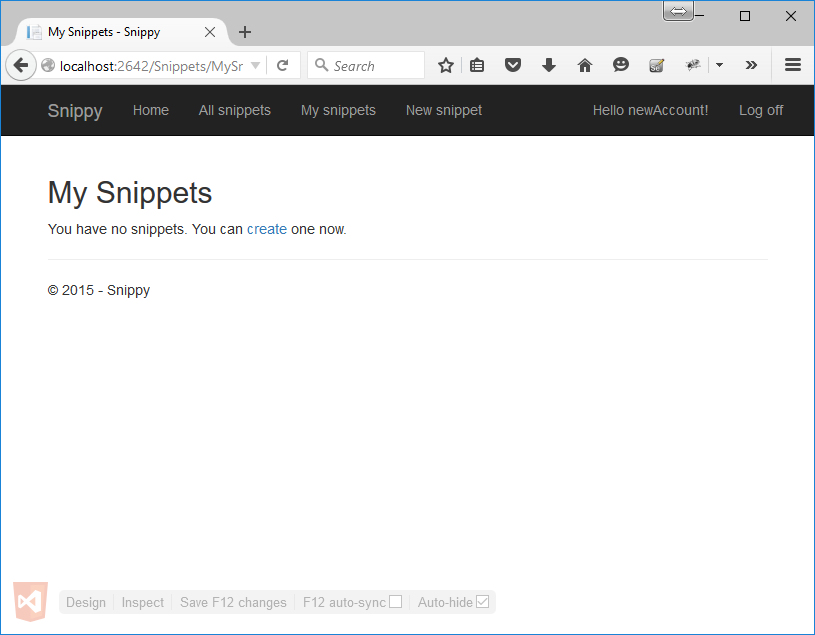
* Display some details about all snippets for the given language. You can omit paging if you like. The page must include the snippets’ titles and labels.
* In case there are no snippets created by the currently logged-in user, display a message.

3 score

* Display a link in the top menu if there is a logged-in user. Do not display the link if the user is a guest.

1 score





## Implement the New Snippet Page

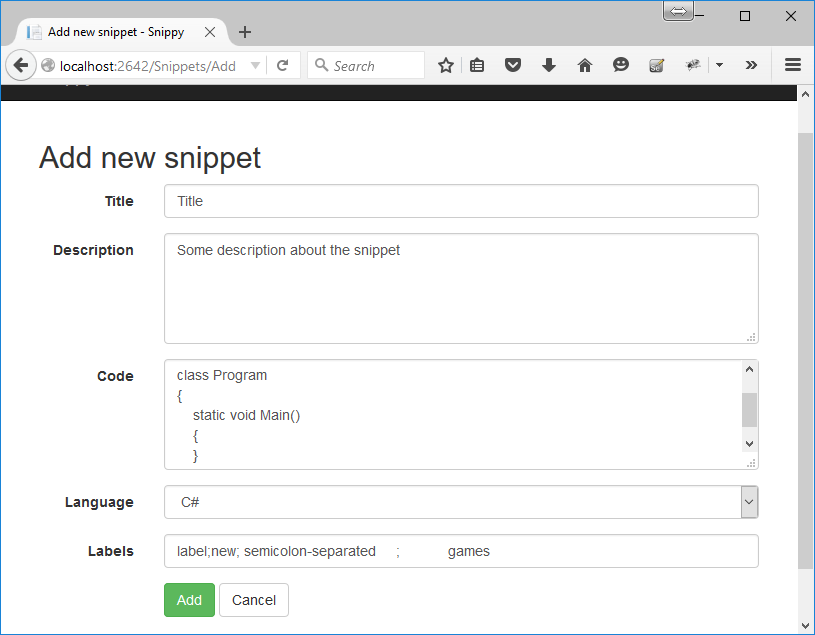
Logged-in users can create new snippets.

* A new snippet has a title, description, code, language (one of the languages which already exist in the database) and labels.

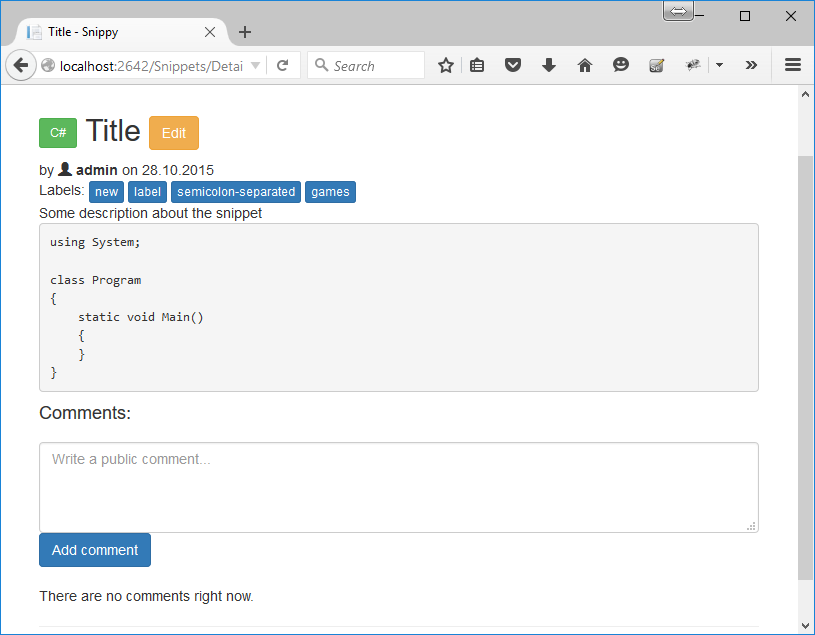
5 score

* The labels are separated by a single semicolon (and optionally some spaces around it). If a label exists, reuse it, and if it doesn’t – create it.
* The language should be a dropdown containing all languages registered in the system.

7 score



* When the new snippet is added, the application should display the newly created snippet.

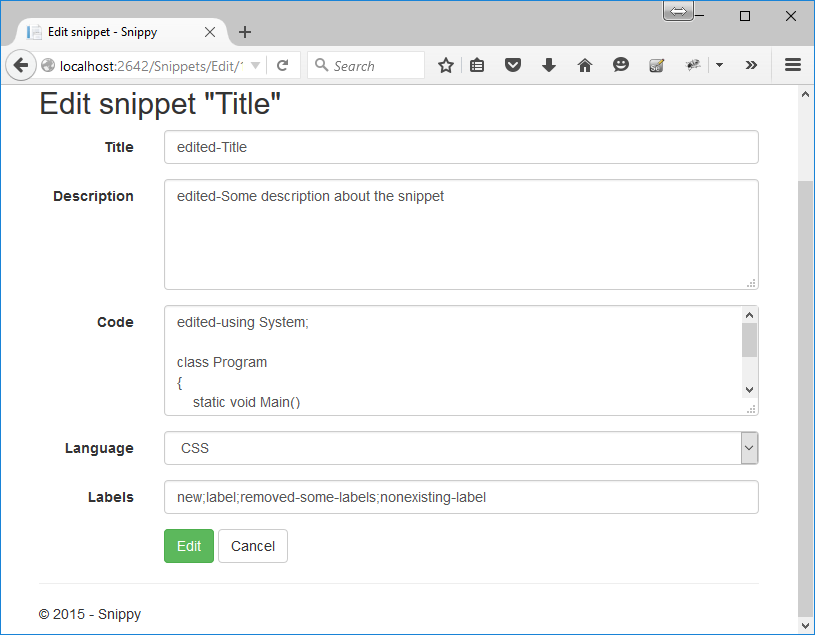


2 score

## Implement the Edit Snippet Page

Logged-in users can edit their own snippets.

* On the snippet details page display an Edit button next to the title if the snippet has been created by the currently logged in user.
* When the user pressed the Edit button, open a page to edit the snippet’s title, description, code, language and labels.



10 score

* Upon successful editing, the application should display the newly created snippet.

1 score

## Search by Title and Labels

Implement searching for snippets by title and labels.

* Display a search form in the top menu. When the user enters a search phrase, look for snippets which contain the search phrase in their titles or labels.
* Display all found snippets along with their labels (similar to the home page).

**BONUS:** 5 score

## Project Infrastructure Bonus

Bonus points code quality / good application structure / additional effort.

* Bonus points for implementing **separate data layer**.
* Bonus points for using **dependency inversion**.
* Bonus points for using **AutoMapper**.
* Bonus points for using **services** for different jobs.

**BONUS:** 5 score