



**ID1354**  
**Internet applications**  
**Assignment 2**

**Leif Lindbäck, Nima Dokoochaki**  
**leifl@kth.se, nimad@kth.se**

SCS/ICT/KTH

# Goal

- Learn **JavaScript** programming.
- Use **jQuery** to facilitate event handling and updates.
- Use a **framework** that gives good **architecture and design** for the JavaScript client, for example **Knockout**.
- You will write JavaScript that enhances the ***Tasty Recipes*** web site.

# Mandatory Tasks

The following tasks must be solved and reported at the seminar.

1. Insert a **JavaScript menu** on all pages of the ***Tasty Recipes*** web site. Requirements on the menu follow below.
2. Make it possible to **add and edit comments**. More detailed requirements follow below.
3. You must **evaluate all JavaScript code** at JSLint (**jslint.com**). You do not have to pass the evaluation without remarks, but your report must comment on the evaluation result.

# Optional Tasks

The following optional tasks gives one higher grade point each. You may choose freely among these tasks.

1. Use **jQuery** syntax and APIs in *all* your JavaScript code.
2. Use the **Knockout** framework to structure your JavaScript code. This is *only applicable to mandatory task 2*, not to the JavaScript menu.

# Task 1, Menu

- The menu shall **appear on all pages**.
- The menu must **match the style of the web site**. This applies to for example colors and fonts.
- You must **explicitly choose** font size, family and style; foreground and background color; mouse hovering and link behaviour. *None* of these properties may have the default value just because it is the default.

# Task 1, Menu (Cont'd)

- The menu shall have the labels **Home**, **Recipes**, **Contact** and **Calendar**
- **Home** takes the user to the index page, **Calendar** to the calendar page.
- **Recipes** shows a drop down menu with one item for each existing recipe.
- **Contact** shows a drop down menu with the items **About Us** and **Contact Us**. These two items correspond to **two new pages that must be created**.

# Task 2, Add and Edit Comments

- Introduce the possibility to **add and edit comments to the recipes**.
- The comments section must **match the style of the web site**. This applies to for example colors and fonts.
- You must **explicitly choose** font size, family and style; foreground and background color; mouse hovering and link behaviour. *None* of these properties may have the default value just because it is the default.

## Task 2, Add and Edit Comments (Cont'd)

- Anyone can add a comment and anyone can edit any comment. This is of course not appropriate, registration and login will be added in the next seminar.
- Comments are stored locally in the javascript code and disappear when the page is reloaded. This will also be changed in the next seminar, when comments will be stored on the server.



# Auto-Generated Code and Copying

*All HTML and CSS code must be well-formatted and you must be able to explain and motivate every single part.*

*However:*

- You are allowed to use any web development tool, you do not have to write HTML and CSS by hand.
- You are allowed to copy HTML, CSS and JavaScript from any web site. In particular, you are encouraged to get inspiration from (or use) one of the many free JavaScript menus, for example <http://www.siteground.com/tutorials/menu/javascript.htm>