Minns du recept sidan du gjorde i våras?
http://jakob.rolandsson.nu/Jensen/SJK16/assets/page/material.php?assignment=7
Nu är det dags för er att bygga ert egna API. Ta all funktionalitet som fanns i Oskars API och implementera den i på ditt egna valfria sätt, och om du vill så gör om din receptsida så att den använder sig av din web services istället för Oskars.

Repository:et som hanterar recepten bör implementera följande interface
public interface RecipeRepository {

    //GET-method
    /**
     * Returns a list of all recipes that have been published
     * **@return** the list of published recipes
     */
    public List<Recipe> **findAll**();

    //GET-method
    /**
     * To find a specific recipe
     * **@param** recipeKey the key of the recipe
     * **@return** the recipe
     */
    public Recipe **find**(**int** recipeKey);

    //DELETE-method
    /**
     * Delete a published recipe
     * **@param** recipeKey the key of the recipe
     */
    public void delete(**int** recipeKey);

    //POST-method
    /**
     * Updates the average rating of a cookie recipe
     * if none exist then a new one is created
     * **@param** recipeKey the key of the recipe
     * **@param** newRating a new rating sent by client, to be added with the already existing
rating
     * **@return** the recipe with the new average rating
     */
    public Recipe **update**(**int** recipeKey, **double** newRating);
    /**
     * Can be replaced by (more flexible):
     * public Recipe update(Recipe recipe);
     * Depending on your implementation

```java
    */

    //PUT-method
    /**
     * publishes a new recipe to the findAll method
     * @param recipeKey the recipe key of the recipe to be published
     */
    public void publish(int recipeKey, boolean publish);
    /**
     * Can be replaced by:
     * public void publish(Recipe recipe);
     * or
     * public void publish(Recipe recipe, boolean publish);
     * depending on your implementation
     */
}
```

Med en tillhörande Recipe POJO

Recpie (POJO)
-    amountOfVotes : int -   averageRating : double -   key : int/String -   isPublished : boolean
+   getAmountOfVotes() : int +   setAmountOfVotes(amountOfVotes : int) : void +   getAverageRating() : double +   setAverageRating(averageRating : double) : void +   getKey() : int +   setKey(key : int) : void +   isPublished() : boolean +   setPublished(isPublished : boolean) : void +   updateAverageRating(newRating : double) : void +   toString() : String

Om du hinner så börja fundera på hur du kan implementera en service där endast **du** kan lägga på ratings/publicera/ta bort just **ditt** recept. Dvs. hur implementerar du en authentication?
Läs: Best Practise REST tokens-based-authentication