**Our presentation:**

First, use MVC structure and then change it to Spring.

DAO🡪 basic database, control package

[Css could define a name in css file, whenever want to use the style, using html tags. (jsp🡪a way to produce html tags)]

Editing static html page, dynamics produce

[Js: computer purely on the client site, could send the request to the server.]

Suggestion from peers: Structure of web app more, how to achieve the functions

**Our main thought: Make somethings work.**

**Tutor’s suggestions:**

**Best**: separate design into 3 parts, functions flow, data process flow

**Minor**: do not flow the **layered structure** of design.

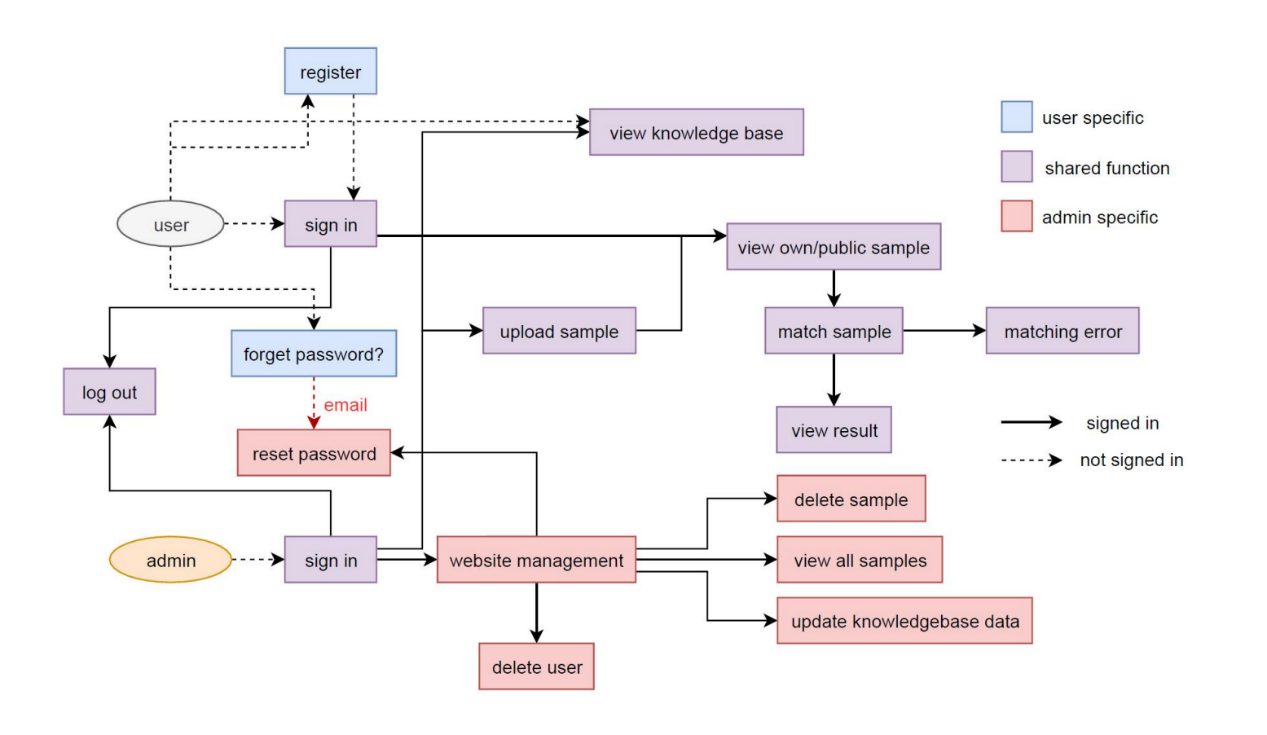
**Overview of the functions and database, flow what has been mentioned in requirement analysis**

First give an **architecture** and how to prefer the architecture (the reason why we chose this architecture over something else), how the architecture achieve the function (benefits, disadvantages), performance.

Vertical inheritance, function inherits the architecture and data inherits the functions, data flow and user activities should be put in the design interface. Database just built on the requirements analysis (maybe just one figure demonstrates the UML but we need to show one)

User function: sth connect to the architecture, what kind of result expected to deliver to them, follow user requirement diagram, (like one requirement, we use one interface to achieve one function)

Design: include the class structure, table of all the class describe the functions of class, each class for each part. Packages have what functions and what are the main classes we want to contain.

**Database UML, function flow chart (user control)** ****

