<CAU Plate>  
Web Page proposal

Web programming

# Product Overview

# <CAU Plate> is a page that conveys information about restaurants in the vicinity of Heukseok-dong and Sangdo-dong. Through this site configuration, distributed restaurant information can be collected and information can be accessed more easily. In addition to this, we will test the service according to the convenience of the users and recommend a customized restaurant to provide customized service. Also, by managing members, we will be able to raise after-hours and after-dinner delicacies of restaurants to enable mutual information exchange.

## Objective

* Informing the restaurants around CAU.
* Categorizing the restaurants around CAU.
* Recommending the suit restaurant through simple test.
* Building simple and direct User Interface.
* Concentrate restaurant users around Heukseok-dong and collect information.
* Dynamic user interaction enables realistic evaluation of restaurant tastes.
* Increase web reliability among university students.

## Software Tools

* HTML/ CSS Tool : ATOM
* Library Tool : Bootstrap
* Server Tool : Django

## Schedule

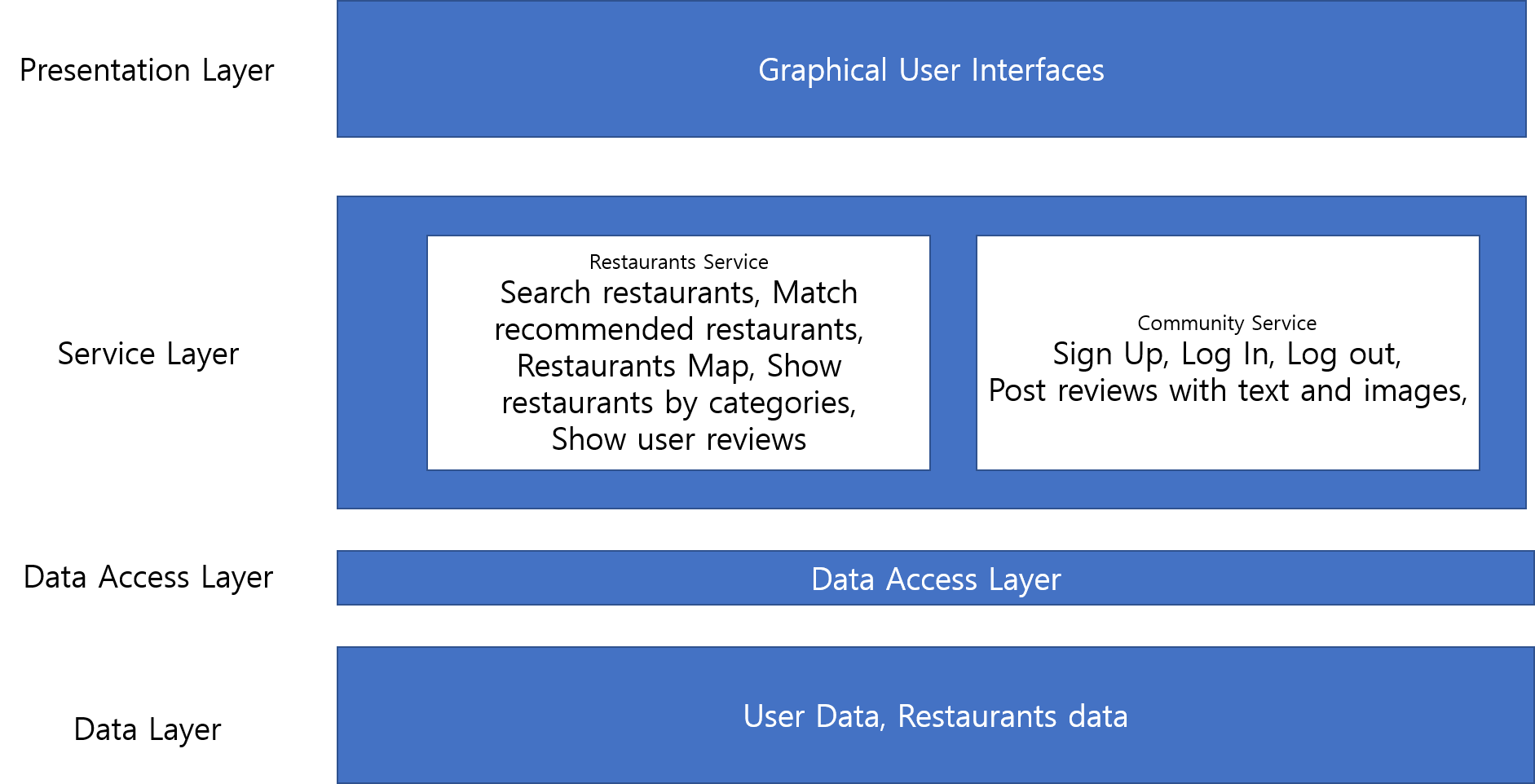
The date is the best estimate and can be changed before the contract is enforced.

|  |  |  |  |
| --- | --- | --- | --- |
| 설명 | 시작 날짜 | 끝 날짜 | 기간 |
| Start project and start planning. | 2018. 10. 01 | 2018. 10. 07 | 7 days |
| Start building databases such as restaurant data and photographs. | 2018. 10. 08. | 2018. 10. 15 | 7 days |
| Start HTML work on ‘HOME’ page and set logo. | 2018. 10. 30 | 2018. 11. 05. | 7 days |
| Completion of home screen and construction of category screen | 2018.11. 06. | 2018. 11. 13. | 7 days |
| Complete category page and add map page | 2018. 11. 13. | 2018. 11. 20 | 7 days |
| Map page completion and taste test page start. | 2018. 11. 20. | 2018. 11. 27 | 7 days |
| Final review and error review | 2018. 11. 27. | 2018. 12. 02. | 7 days |
| Feedback and completion | 2018. 12. 03. | 2018. 12. 10. | 7 days |

## Challenges and Lisks

* Error correction process in the process of getting started with programming.
* Trouble in the process of applying HTML, CSS, and JAVA to actual practice.
* The difficulty of division of work for efficient work when working as a team.
* Learning how to build servers and databases,
* It takes a long time, but it is a hard time to do the midterm and final exams.

# Software Architacture



## Restaurants Service

* Search restaurants : Users can search restaurants by part of restaurants’ name in search bar. Filtered restaurants will be shown on the filtered list.
* Match recommended restaurants : Our web service will recommend restaurants after asking some questions. Those questions will collect information of your preferred foods and costs.
* Restaurants Map : In main page of our service, users can approach to the restaurants’ location by markers map. We will use Google MAPS API.
* Show restaurants by categories: We will provide two categories. One is based on a type of foods, and other is based on location.
* Show user reviews: We will show users’ reviews in the detail view of each restaurants.

## Community Service

* Sign Up : We need users’ email and password.
* Log In/Log out: Users’ can log in/out with their email and password.
* Post reviews with text and images: After log in, users can post review of restaurants.

# Product Description

## Functional Requirement

|  |  |  |
| --- | --- | --- |
| **Functions** | **Language** | **Information** |
| Restaurant introduction | HTML | Basic information about restaurants. |
| Restaurant location search | HTML  JSP | Informing Location through visualized map. |
| Restaurant classification | HTML | Classifies restaurant by styles and location. |
| Restaurant recommendation | HTML  JSP | Recommends restaurant based on simple test. |
| Restaurant reviews | HTML | Assemble reviews on restaurants. |

## Non-Functional Requirement

|  |  |
| --- | --- |
| **Objective** | **Information** |
| Simple User Interface | * We divide the page into four major categories and make up the shallowest homepage. * Minimize writing and pursue composition centering on images and icons. * Minimize difficulties in using through intuitive page organization. |
| Minimal Errors | * Minimize errors by writing a good syntax. * Use semantic mark up to separate the work and minimize errors. |

>