

TAREA PROGRAMADA 2

Introducción

Bombur es el más gordo y glotón de todos los Enanos de Tierra Media, por lo que siempre anda buscando los mejores restaurantes para satisfacer su apetito. Ahora, la búsqueda de los restaurantes no es un proceso sencillo, ya que usualmente se tiene que trasladar grandes distancias, a veces incluso sin contar con un pony para su transporte.

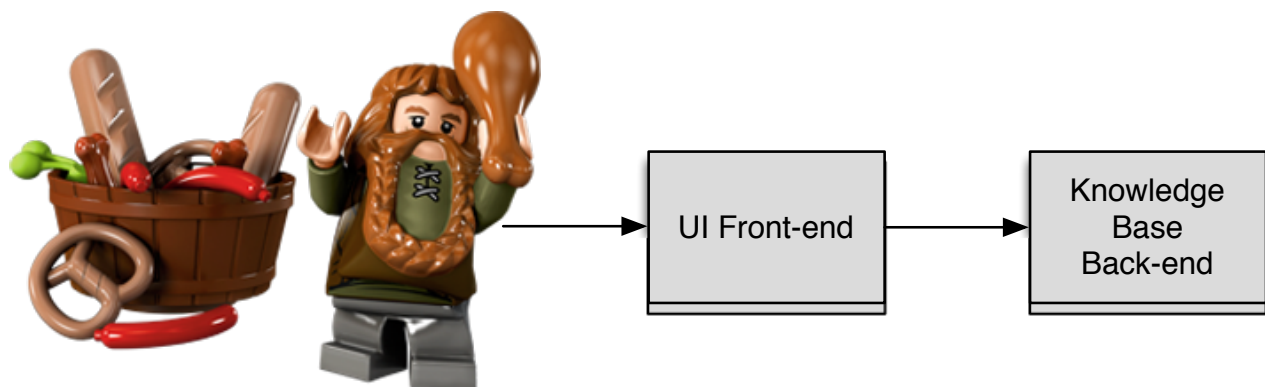
La vida de Bombur sería más sencilla si tuviera una forma de consultar los restaurantes desde la comodidad de su smartphone, por lo que se le ha encargado la tarea de desarrollar una aplicación web especialmente para ese fin.

Descripción del programa

El sistema tendrá dos modalidades principales: mantenimiento de datos y consulta. En la modalidad de mantenimiento de datos, Bombur podrá ingresar los datos de un restaurante: nombre, tipo de comida, ubicación, teléfono y horario. Adicionalmente, para cada restaurante se podrán ingresar los platillos favoritos. La información que se podrá ingresar de cada platillo es la siguiente: nombre del platillo, si es picante, salado, dulce, agri dulce o amargo, país de origen del platillo, y la lista de ingredientes.

En la modalidad de consulta, Bombur podrá hacer consultas de restaurantes con base en ciertos criterios, y el sistema deberá mostrar la información respectiva.

El sistema a su vez estará estructurado en dos componentes: Front-end y Back-end.



El Front-end será el componente encargado de manejar toda la interacción con el usuario, incluyendo entrada y salida de datos. Adicionalmente, se comunicará con el Back-end para las tareas relacionadas con mantenimiento de datos y consulta. El front-end será una aplicación web escrita en Python. Ahora, para que Bombur pueda usar esa aplicación desde el teléfono móvil, es necesario que la aplicación use diseño responsive.

El Back-end será el encargado de manejar la base de conocimientos de restaurantes, y de responder las consultas hechas por el usuario (aunque de manera indirecta, ya que el Front-end es el que realmente maneja la interacción con el usuario, y hace a su vez las consultas al Back-end).

El Back-end tendrá dos funcionalidades principales: manejar la base de conocimientos y responder las consultas.

La base de conocimientos será una serie de declaraciones en Prolog que defina todos los atributos de los restaurantes y los platillos. El Back-end deberá actualizar la base de conocimientos usando los datos que el usuario ingrese al sistema por medio del Front-end.

La estructura de la base de datos podrá ser definida por los estudiantes, no tiene que tener un formato particular. Sin embargo, en la documentación se deberán explicar las decisiones de diseño detrás de la estructura elegida..

La lista de consultas que deberá tener la aplicación es la siguiente:

- Lista de restaurantes
- Lista de restaurantes filtrados por tipo de comida
- Búsqueda de restaurantes por nombre
- Lista de restaurantes que tienen platillos de algún país específico
- Lista de platillos de un restaurante específico
- Lista de platillos de un restaurante específico que tengan un ingrediente en particular
- Lista de regalos más solicitados por los niños, ordenados por popularidad

El modo de funcionamiento para consultas es el siguiente:

1. Bombur abre la aplicación web desde su teléfono móvil, y hace una consulta al sistema (a través del front-end, que es la parte del sistema a la que tiene acceso)
2. El front-end pasa la consulta al backend
3. El Backend hace una búsqueda en la base de conocimientos, y devuelve la información al front-end
4. El front-end muestra la información al usuario (en caso de que no hayan datos que correspondan con la búsqueda, deberán mostrar un mensaje al usuario)

El modo de funcionamiento para mantenimiento de datos es el siguiente:

1. Bombur abre la aplicación web desde su teléfono móvil, e ingresa los datos de un restaurante, junto con sus platillos
2. El front-end pasa los datos al backend
3. El Backend guarda los datos en la base de conocimientos, y le devuelve el resultado de la operación al front-end
4. El front-end muestra al usuario un mensaje que le indique que ya se terminó de realizar el almacenamiento de datos en la base de conocimientos.

Aspectos técnicos

El Back-end deberá estar escrito en el lenguaje de programación Prolog.

Con respecto al Front-end, deberá estar escrito en Python, y deberá ser una aplicación web responsive, la cual deberá tener un layout adaptado para dispositivos móviles (celulares, y opcionalmente tablets).

El proyecto completo deberá funcionar en el sistema operativo Linux.

Aspectos a investigar

Los estudiantes deberán investigar los siguientes conceptos:

- Diseño web responsive
- Desarrollo de aplicaciones web en Python (pueden usar algún framework)
- Consultas a Prolog desde Python
- Inclusión dinámica de hechos y reglas en la base de conocimientos

Documentación

La documentación es un aspecto de gran importancia en el desarrollo de programas, especialmente en tareas relacionadas con el mantenimiento de los mismos.

Para la documentación interna, deberán incluir comentarios descriptivos para cada función, con sus entradas, salidas, y restricciones.

La documentación externa deberá incluir:

- Resumen ejecutivo
- Propósito: descripción del app, requerimientos
- Funcionalidades
- Descripción de diseño de alto nivel
- Descripción detallada (decisiones de diseño, lenguajes de programación usados, tecnologías y librerías usadas)
- Problemas encontrados
- Manual de usuario
- Otros detalles

Evaluación

Documentación interna	2%
Documentación externa	8%
Inserción de datos de restaurantes	15%
Consulta de restaurantes	20%
Interconexión Prolog-Python	10%
Diseño responsive	10%
Aplicación web	15%
Revisión de tarea	20%

Aspectos administrativos

- La tarea vale un 10% de la nota del curso
- La tarea se hará en grupos de 3 personas.
- Fecha de entrega: Jueves 27 de marzo, 10 a.m. No se aceptan tareas entregadas después de esa fecha y hora.
- Los grupos deberán subir el código y la documentación de sus respectivas tareas a un repositorio en Github, de manera que el profesor pueda ver las contribuciones que las diferentes personas hacen al proyecto. La idea es que apenas empiecen a desarrollar la tarea, suban las contribuciones al repositorio, y no esperar a tener todo el código listo para subirlo.
- Deberán enviar un correo a andreifu@gmail.com, con copia a evelyn.madriz@gmail.com, en donde indiquen el url del repositorio de Github en donde se encuentra el código y la documentación de la tarea. El asunto del correo enviado tendrá el siguiente formato: **TI-3404 TP2-nombre1-nombre2-nombre3**. Las tareas que no sean entregadas por medio de Github tendrán nota cero.
- Las tareas deberán ser revisadas con el profesor o el asistente. Todos los miembros del grupo deberán participar de la revisión, ya que de lo contrario no se les asignará el puntaje correspondiente. La nota de la revisión es individual, el resto de la nota es grupal.
- El código entregado debe ser 100% original. En caso de probarse algún tipo de fraude en la elaboración de la tarea, se aplicarán todas las medidas correspondientes, según el reglamento del TEC, incluyendo una carta al expediente.