

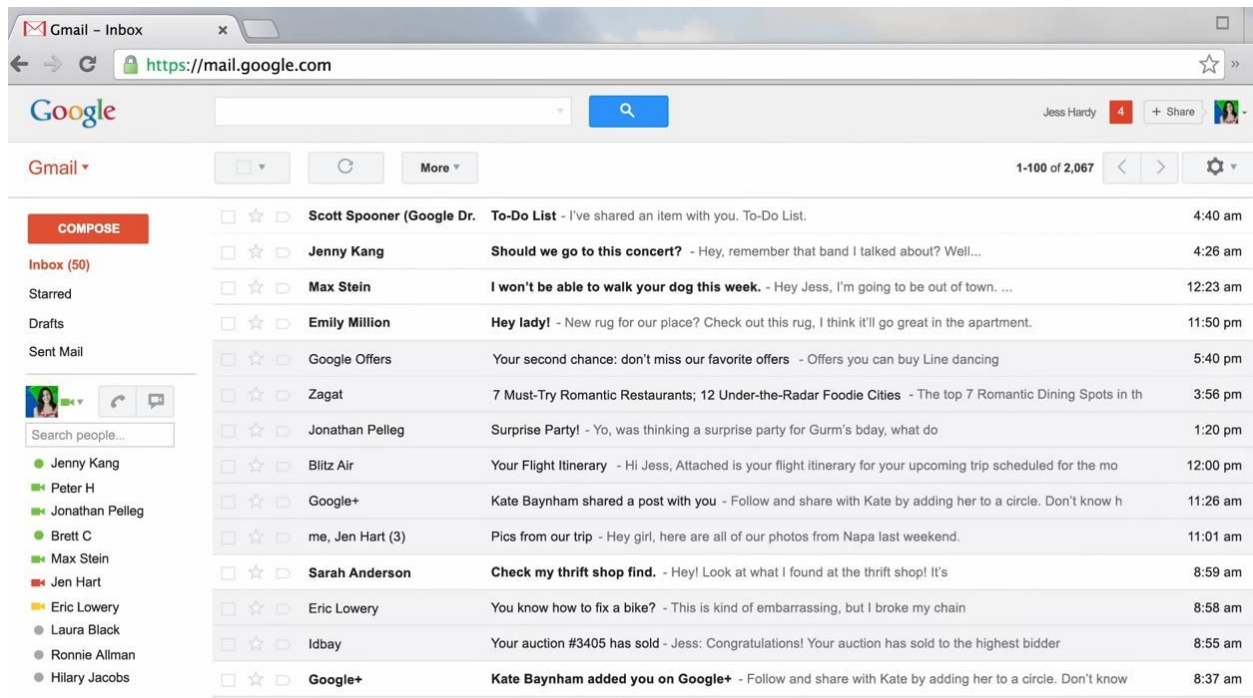
LABORATORIO II - TRABAJO FINAL 2017

Escenario:

Actualmente, toda Empresa del medio, utiliza los Emails como una herramienta de Software que es vital para su gestión diaria, para mantener a sus empleados al corrientes, principalmente para intercambiar información pública o privada y enviar archivos adjuntos.

Sabiendo que una Casilla de Correos o Emails permite a los usuarios visualizar en una sola pantalla la lista de sus emails, listados ordenadamente por la fecha / hora de ingreso de los mismos, permitiéndoles organizarlos por distintos criterios, como leídos y no leídos, archivarlos o moverlos a otras carpetas, aplicar etiquetas, realizar búsquedas por distintos criterios y palabras reservadas.

En de las más utilizadas es GMAIL:



Se le pedi diseñar un conjunto de Servicios Web que permitan reproducir las acciones que se pueden realizar dentro de la Página Principal de GMAIL (u otro sistema de gestión de emails).

Entre las acciones principales se listan:

- Listar un conjunto de emails ordenados de forma cronológica, indicando cuales fueron leídos y cuáles no.
- Leer o abrir un email.
- Crear un nuevo Emails, indicando el Remitente, el/los Destinatarios, Título y Texto.

- Eliminar uno o varios emails.
- Mover emails a otras carpetas ya que por defecto sólo están en la “Bandeja de Entrada”.
- Aplicar una etiqueta a uno o varios emails.
- Filtrar los emails dependiendo de varias criterios:
 - Destinatario
 - Etiqueta
 - Alguna palabra o frase.
 - Fecha de ingreso a la Bandeja de Entrada.

Objectivo:

El Trabajo de Fin de Año de la Materia Lab II consiste en el diseño, programación y prueba de un conjunto de Servicios REST que permitan realizar las acciones antes descritas. Deberá diseñar un Servicio REST por cada ítem detallado, utilizando lo aprendido en clases.

Entregables

- En el Sistema se tiene que demostrar el uso de:
 - Clases y relaciones entre clases.
 - Métodos y Funciones
 - Tipos de Accesibilidad
 - Colecciones
 - Interfaces
 - Palabras reservadas: static, final
 - Uso de la clase StringBuilder
 - Exceptions y manejo de Exceptions con Try..Catch
 - Algoritmos o Técnicas de Programación aprendidas en clases, como ser validaciones, algoritmos de búsqueda, etc.
- Servicios Web Restful necesarios para realizar todas las acciones mencionadas y otras en caso de considere necesario.
- Pruebas Unitarias sobre los métodos de las clases que contengan lógica o estructuras complejas, evitar hacer casos de pruebas solo de clases simples.

No son entregables:

- Diseño de página Web o cualquier otro Cliente que consuma la API Web.
- Documentación de los Servicios Web.
- Diagramas de Clases y otros tipos.
- Implementación de un módulo para persistencia en Archivos, Base de Datos, u otras tecnologías.

Herramientas a utilizar:

- JAVA, JDK
- IDE JAVA: IntelliJ
- Apache MAVEN
- Spring Boot Framework.
- JUnit Framework.
- PostMan

Modalidad de Trabajo:

- El trabajo puede ser desarrollado individualmente o por un grupo de 2 integrantes.

Modalidad de Entrega

- El alumno o el grupo debe presentar el Sistema funcionando.
- El Sistema puede ser presentado utilizando una PC del Alumno, no es necesario que el mismo sea instalado en las PCs del Laboratorio.
- El alumno utilizará las herramientas presentadas en clases para presentar el Sistema.

Modalidad de Evaluación

- El Sistema debe funcionar.
- Se deberá presentar el código utilizado para lograr que el sistema funcione.
- El Sistema será puesto a evaluación por el docente, quien realizará preguntas varias.
- El Sistema que no implemente cualquiera de los ítems previamente detallados, desaprobará.

Modalidad de Aprobación

- El Trabajo Final se considera como una Nota de Parcial.
- Un trabajo Final completo, implica aprobar el Parcial con 100 puntos.
- Un Trabajo Final incompleto, implica una desaprobar el Parcial.
- Desaprobar el Trabajo Final, NO implica desaprobar la materia, pero NO puede promocionar la misma, independientemente de la Nota del Parcial Práctico.

Fecha Final de Entrega y presentación del trabajo

- 24 de Noviembre de 2017.

Referencias

- Restful: https://en.wikipedia.org/wiki/Representational_state_transfer

- MAVEN: <https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html>
- Spring Boot: <https://spring.io/guides/gs/rest-service>
- JUnit Framework: https://www.tutorialspoint.com/junit/junit_overview.htm