|  |  |  |
| --- | --- | --- |
| **Nombre**:  Fernando Gutiérrez Arévalo | | **Matrícula**:  2828960 |
| **Nombre del curso:**  Taller de productividad basada en herramientas tecnológicas | **Nombre del profesor**: Daniel Horacio Chávez Argott | |
| **Módulo**: Fase 1 Introspección | **Actividad**: Fase1 | |
| **Fecha**: 19/08/2020 | | |
| **Bibliografía**:  TecMilenio. (2020) Taller de productividad basada en herramientas tecnológicas. 19/08/2020, de TecMilenio Sitio web: https://cursos.tecmilenio.mx/courses/31346/pages/mi-curso?module\_item\_id=121480 | | |

**Objetivo:**

Demostrar la adquisición de la competencia del certificado, identificar fortalezas y áreas de oportunidad.

**Procedimiento:**

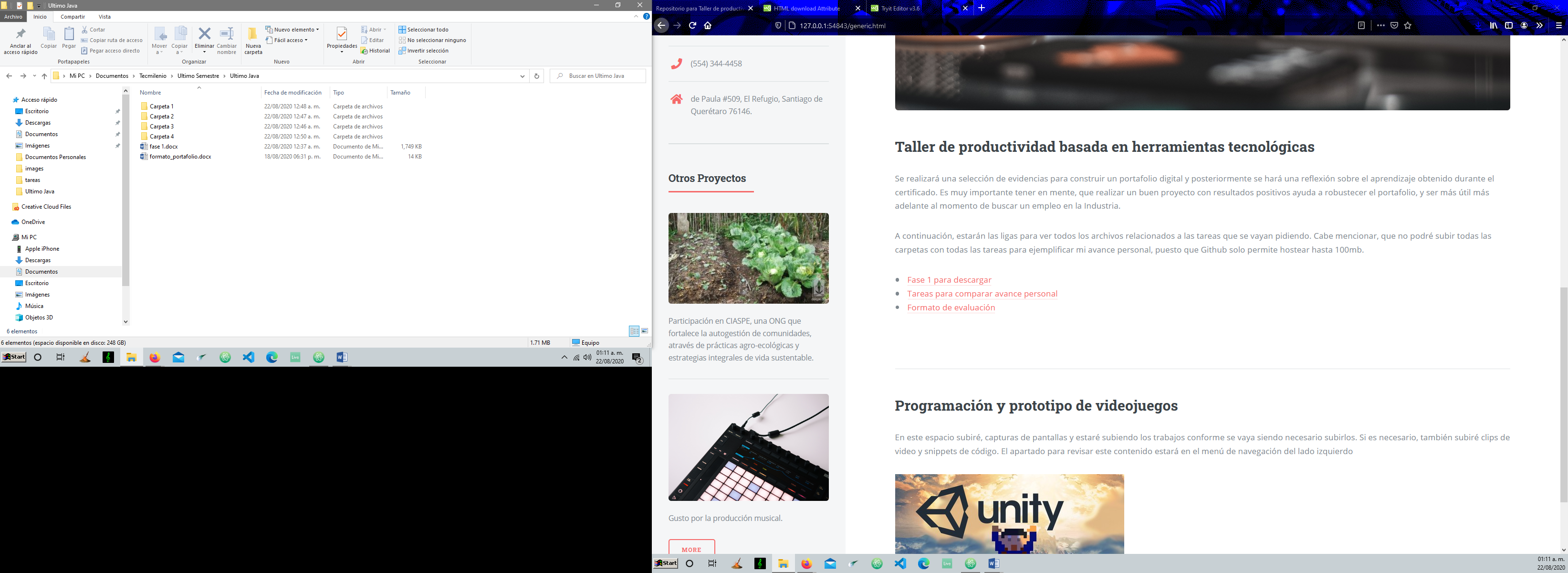
1. Revisa las Evidencias que recopilaste durante este certificado.
2. Tomando en cuenta la **competencia** del certificado, organiza tu portafolio en 4 carpetas, mismas que deberán contener lo siguiente:

**Carpeta 1:** incluye trabajos donde muestres los conocimientos que adquiriste, los que modificaste, los que ampliaste y los que aún quedan por aprender (áreas de oportunidad).  
**Carpeta 2:** incluye 2 o más trabajos que al ser comparados se pueda observar la secuencia del aprendizaje de conocimientos.  
**Carpeta 3:** incluye Evidencias donde muestres las habilidades, actitudes y valores que adquiriste, los que modificaste, los que ampliaste y los que aún quedan por aprender (áreas de oportunidad).  
**Carpeta 4:** presenta 2 o más trabajos que al ser comparados se pueda observar la secuencia del aprendizaje de habilidades, actitudes y valores.

Nota: podrás revisar la **competencia** del certificado haciendo clic en Inicio> ¿De dónde voy a partir? > Estructura del certificado.

1. Elabora un documento de reflexión con extensión mínima de 3 cuartillas, donde des respuesta a TODAS las siguientes preguntas:
   1. ¿Qué conocimientos, habilidades, actitudes y valores aprendiste? ¿Cómo aprendiste? ¿Dónde aplicarías dichos aprendizajes?
   2. Explica tu secuencia de aprendizaje de conocimientos. ¿Qué conocimientos adquiriste primero? ¿Cuáles después? ¿Qué relación tienen?
   3. Explica tu secuencia de aprendizaje de habilidades, actitudes y valores. ¿Cuáles adquiriste primero? ¿Cuáles después? ¿Qué relación tienen?
   4. ¿Qué factores discutiste o pensaste a lo largo del certificado? ¿Qué más necesitas saber?
   5. Compara tus mejores trabajos con aquellos no tan buenos. ¿En qué se parecen? ¿En qué son diferentes? ¿Por qué consideras que uno es mejor que otro?
   6. ¿Qué cambió en ti a lo largo del certificado? ¿A qué atribuyes dichos cambios?
   7. ¿Qué más sabes de ti mismo ahora?
2. Revisa que tu portafolio esté completo llenando el formato:  
   [Formato de revisión para autoevaluación del portafolio](https://a14121-10852750.cluster211.canvas-user-content.com/courses/14121~4611/files/14121~10852750/course%20files/cs/cs13314/anexos/Formato_portafolio.zip)

**Resultados:**



Durante el certificado, aprendí primero los conceptos de programación orientada a objetos. Entendí, como el concepto de objeto puede almacenar más conceptos que, a fin de cuentas, entendemos como información o indicadores que sirven para analizar sistemas. Empezando solo por eso, entender estos conceptos puede ser aplicados a cualquier tipo de lenguaje de programación; no solo es un concepto que aplique a una sola forma de programar. Posteriormente, entendí como es que el trabajo conjunto de sociedades, o particulares, puede beneficiar a tus propios proyectos aplicando el concepto de código compartido por medio de una “dependencia”. Entendí que todo el internet está construido por dependencias compartidas y manejadores de estas dependencias, para el funcionamiento de cada elemento visible y no visible de un sitio web. Todos estos conocimientos pueden ser aplicados a diversas situaciones superfluas, o no, en el trabajo cotidiano, aun cuando no sea de desarrollador de software.

/\*

\* CompilaciÃ³n: java AnnoBisiesto.class

\* EjecuciÃ³n: javac AnnoBisiesto 2015

\*

\* Imprime:

\* - Si el aÃ±o de entrada es bisiesto = true o false

\*

\*/

**import** java.util.Scanner; //importando librerias para la clase "leer" ... A ver si blackboard, enseña de eso. :c

**public** **class** AnnoBisiesto {

**public** **static** **void** main(String[] args){

**int** anno = Integer.parseInt(args[0]);

Scanner leer = **new** Scanner(System.in);

**boolean** esBisiesto;

// Escribe tu cÃ³digo {

**int** anno = leer.nextInt();

**if** ((anno % 4 == 0) && ((anno %100 !=0 )||(anno % 400 == 0)))

{

System.out.println("Es un año bisiesto");

esBisiesto= **true**;

}

**else**

{

System.out.println("El año no es bisiesto");

esBisiesto=**false**;

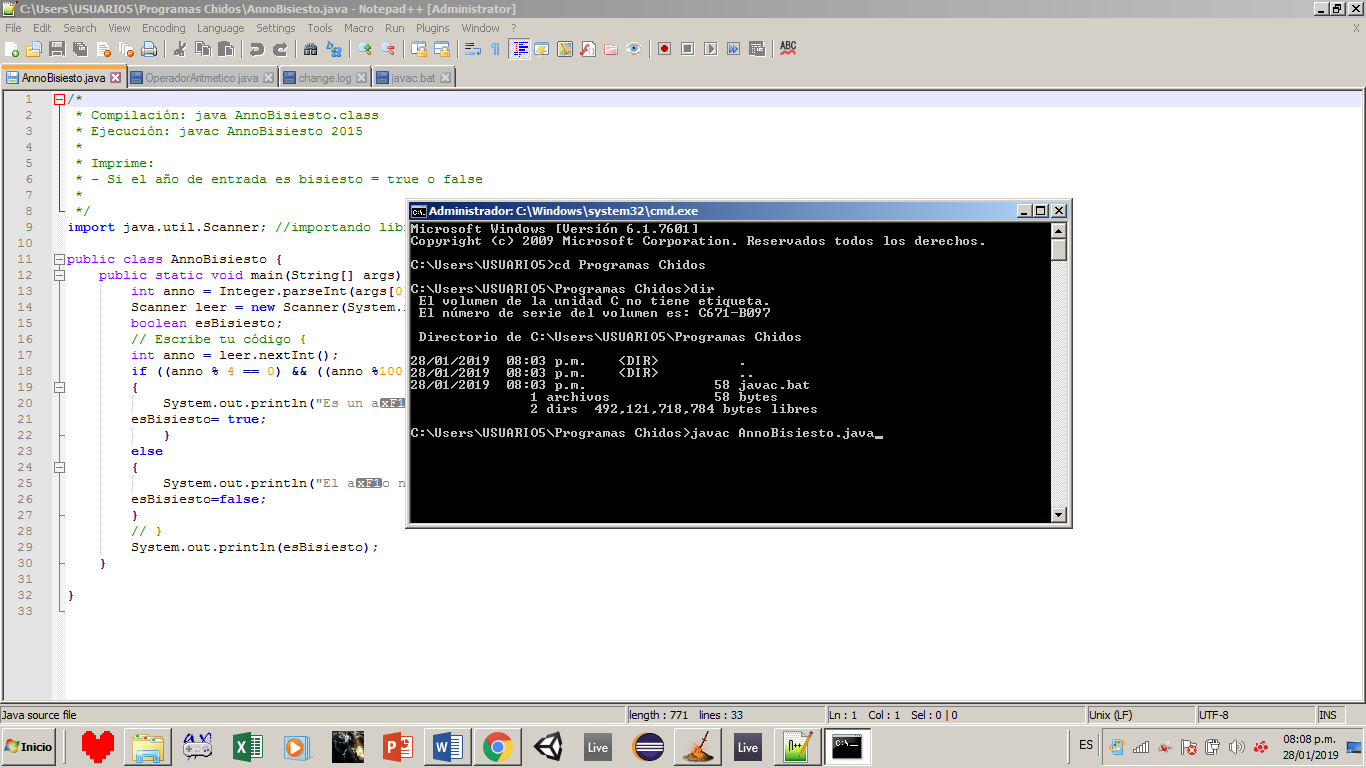
}

// }

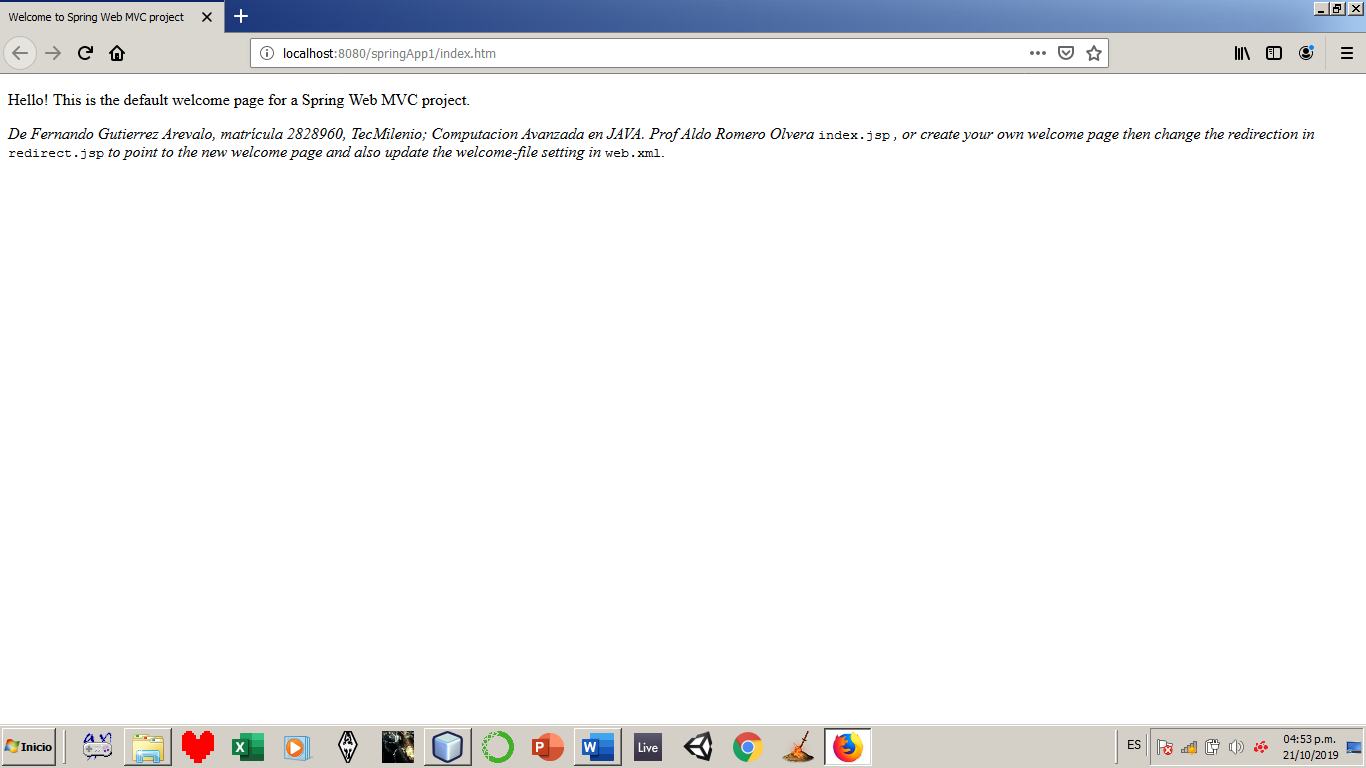
System.out.println(esBisiesto);

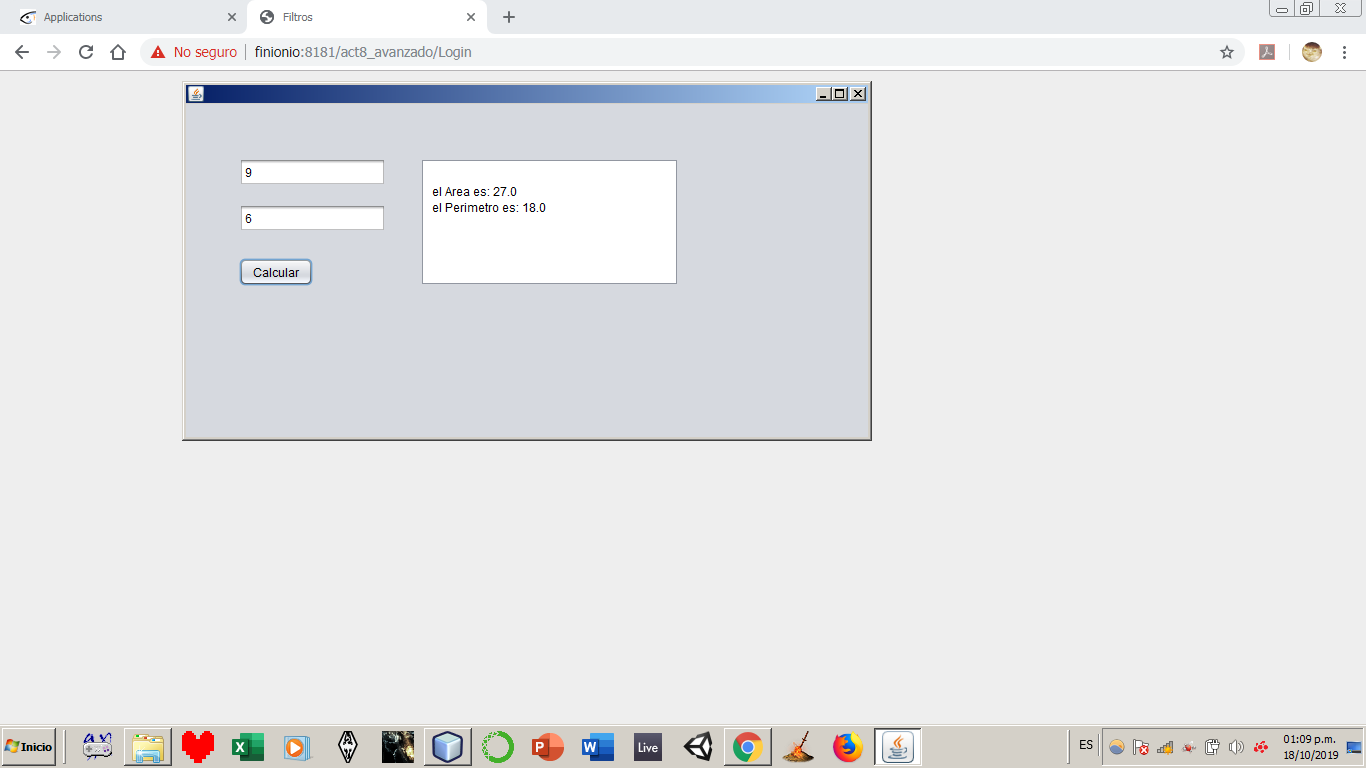
}

}



Snippet de mis primeros códigos

Una mejor implementación en el servidor local, del framework Spring Web

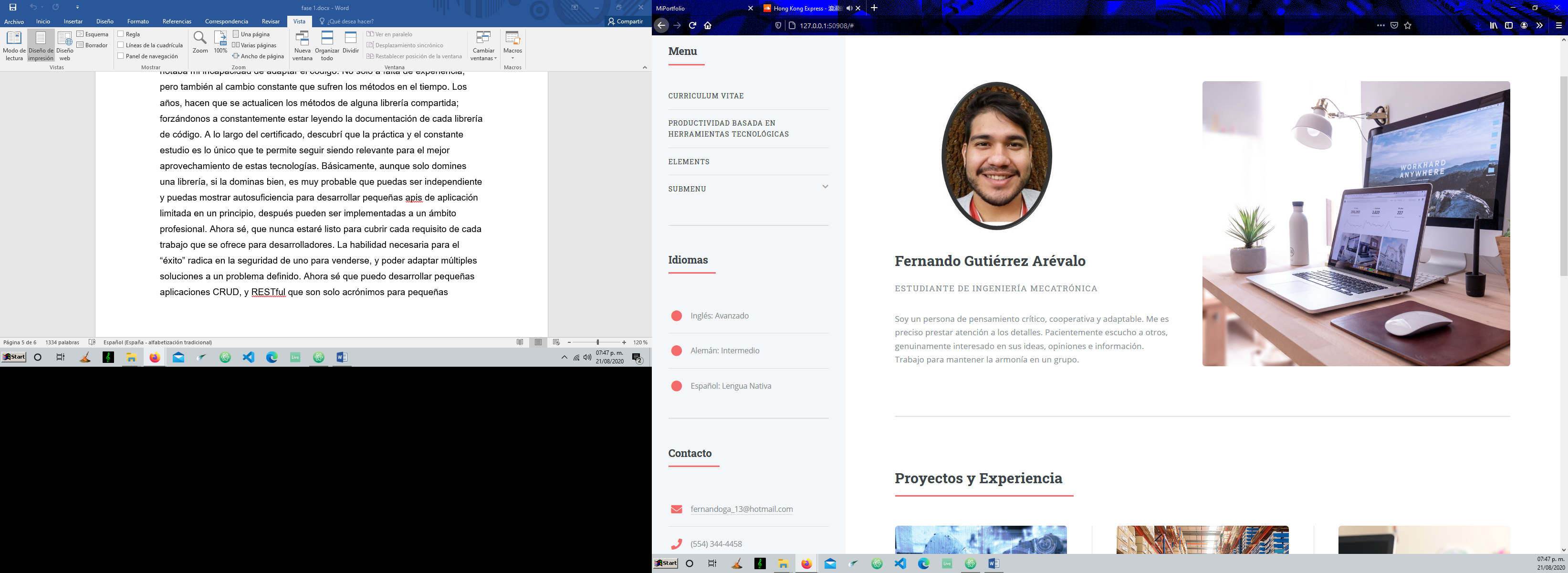


Este trabajo, es un pequeño programa para sacar áreas hosteado en un servidor de glassfish.

Como mencioné, primero tuve que aprender conceptos que han forjado las sintaxis de diferentes lenguajes de programación. Variables, constantes, objetos, clases, superclases, herencias, subclases, seriaciones, tipos de objetos, formas de instanciar, métodos y enfoqué, o alcance de la variable (scope). Estos conceptos, serían los principales que se necesitan para poder dominar cualquier tipo de lenguaje de programación. Esta curva de aprendizaje podría ser tediosa y muy pronunciada, pero es esencial para forjar mentalidad de programador. Todos estos conocimientos se traducen en el empleo de código ajeno, con su respectivo “disclaimer”, el poder ocupar el trabajo ajeno a tu código y saber cómo integrarlo y adaptarlo es el fruto del entendimiento de dichos conceptos simples. La distribución de todo este conocimiento conceptual, se debe a nuestra interconectividad en redes sociales enfocadas, o no, a la programación. Github, siendo la red social más relevante para compartir proyectos y proveer de servicios de host a tus aplicaciones, es la referencia para comunicar y reducir esfuerzos en un mundo que constantemente parece adaptarse a las necesidades y condiciones que nos rodean en nuestro diario. Mis conocimientos posteriores, son más fáciles de digerir, ya que se trata del uso de marcos de trabajo (frameworks) que contienen ruteos, clases establecidas, y métodos que sintetizan la lógica de la transmisión de datos en formatos binarios/hexadecimales. Ahora sé cómo ocupar Express de Node JS, y usar frameworks como angular o reactjs.

Al principio, pensaba que compartir código no era útil, porque entendía como cada lenguaje de programación tiene su particularidad, y encontraba fútil e ineficiente utilizar lógica hecha para otro fin… Hoy, entiendo que el trabajo del programador consiste en esa problemática que me plantee, en poder adaptar el código ajeno a tu propia aplicación para reducir esfuerzos e incrementar la capacidad de “troubleshooting”. Obvio, tengo mucho por aprender, mis áreas de oportunidad son el desarrollo de bases NoSQL para servicios en servidores que limitan la capacidad en memoria de las aplicaciones que uno desarrolla. Así, como también podría reforzar mis conocimientos de lenguajes de programación de alto nivel (Python).

Mis mejores se trabajos se parecen a los no tan buenos, en el ámbito que se utilizó github para compartir un repositorio con fracciones de código faltantes. Entonces, nosotros teníamos que adaptar un código bien estructurado y referenciado a una lógica que apenas y podíamos dominar. Los mejores trabajos resultaron en una mejor adaptación de este código compartido, donde los peores trabajos, además de que apenas entendía que estaba realizando, notaba mi incapacidad de adaptar el código. No solo a falta de experiencia, pero también al cambio constante que sufren los métodos en el tiempo. Los años, hacen que se actualicen los métodos de alguna librería compartida; forzándonos a constantemente estar leyendo la documentación de cada librería de código. A lo largo del certificado, descubrí que la práctica y el constante estudio es lo único que te permite seguir siendo relevante para el mejor aprovechamiento de estas tecnologías. Básicamente, aunque solo domines una librería, si la dominas bien, es muy probable que puedas ser independiente y puedas mostrar autosuficiencia para desarrollar pequeñas apis de aplicación limitada en un principio, después pueden ser implementadas a un ámbito profesional. Ahora sé, que nunca estaré listo para cubrir cada requisito de cada trabajo que se ofrece para desarrolladores. La habilidad necesaria para el “éxito” radica en la seguridad de uno para venderse, y poder adaptar múltiples soluciones a un problema definido. Ahora sé que puedo desarrollar pequeñas aplicaciones CRUD, y RESTful que son solo acrónimos para pequeñas atribuciones en un ambiente profesional. Es empezar con lo más básico, aunque ya esté hecho, para poder entender una aplicación más compleja.



Ahora puedo hacer mis sitios responsivos, en html + css + javascript. Este es mi CV, el cual está en un repositorio de Github y esta hosteado por el servicio. Sin embargo, temo que no pueda subir grandes cantidades de mb, por tanto, estaré subiendo solo los proyectos más relevantes para este curso y capturas de pantalla. Optaré por subir mis trabajos, de evidencia en mi sitio en github:  
  
**Nota**: Github solo puede almacenar 25mb por cuenta, entonces dejo una liga que caduca en una semana de unos pocos trabajos de mi curso con JAVA:

**https://we.tl/t-eCFfj6IriQ**

**Conclusión:**

En esta fase, me permití recordar la importancia de lo que el análisis de las librerías bajo distintos lenguajes de programación puede permitirte alcanzar. Muchas veces nuestra percepción laboral, se ve solamente eclipsada por el contexto de nuestra educación, y experiencia previa en que nos capaciten para desarrollar mejores programas, cuando la realidad es que nadie te capacita, solo la prueba y el error. Si se realiza la suficiente indagación en la documentación de diferentes tecnologías, podemos encontrar soluciones sencillas, no evidentes, para nuestros primeros trabajos.