UNIVERSIDAD INDUSTRIAL DE SANTANADER

FACULTAD DE INGENIERÍAS FISICOMECÁNICAS

**ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**PLAN DE TRABAJO DE GRADO**

**FECHA DE PRESENTACIÓN:** Bucaramanga,

**TITULO**: Desarrollo de Software Móvil para la gestión de la movilidad de la comunidad universitaria hacia y desde el campus.

**MODALIDAD**: Presencial

**AUTORES**:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Javier Eduardo Castellanos Ontiveros 2110643

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Juan Felipe Rueda Román 2110615

**DIRECTOR**:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Fernando Antonio Rojas Morales, profesor titular UIS

**ENTIDAD INTERESADA**: Universidad Industrial de Santander.

**COMITÉ DE TRABAJOS DE GRADO:**

EVALUADOR ASIGNADO:

CONCEPTO DEL EVALUADOR:

**APROBACIÓN DEL COMITÉ´:**

FECHA: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Acta No. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**TABLA DE CONTENIDO.**

**1.PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA**

El transporte público en el área metropolitana de Bucaramanga es ineficiente o de mala calidad, día a día estudiantes como profesores o miembros de la comunidad universitaria en general, usan estos medios para acudir a clases, viéndose perjudicados de una u otra forma en el desempeño normal de sus actividades académicas, además de, en el peor de los casos, se vean forzados a usar medios ilegales de transporte, poniendo en riesgo su seguridad.

Algunos miembros de la comunidad universitaria, cuentan con medios de transporte privados, los cuales no usan en su totalidad, es por esto que se plantea a través del desarrollo de un aplicativo móvil, crear una plataforma que permita la interacción entre estos dos grupos de la comunidad, logrando así suplir la necesidad de un medio de transporte que permite a los miembros de la comunidad desplazarse desde y hacia la Universidad de forma rápida, segura y efectiva.

**2. OBJETIVOS**

**2.1 OBJETIVO GENERAL**

Desarrollo de una aplicación móvil para el sistema operativo Android, de tipo carpooling que permita a la comunidad universitaria compartir lugar en sus vehículos para el traslado desde casa a la universidad y viceversa.

**2.2 OBJETIVOS ESPECÍFICOS**

• Determinar las características de una aplicación móvil tipo carpooling.

• Diseñar e implementar la aplicación en el paradigma de orientación a objetos para el sistema operativo Android.

• Diseñar e implementar una base de datos para la persistencia de objetos de rutas y usuarios.

• Realizar pruebas de funcionalidad de la aplicación.

• Documentar el código de la aplicación utilizando la herramienta javadoc.

**3. MARCO DE REFERENCIA**

* 1. **Carpooling**

El vehículo compartido también conocido por los términos en inglés carpool o carpooling es la práctica que consiste en compartir un automóvil con otras personas tanto para viajes periódicos como para trayectos puntuales. Con esta práctica se pretende reducir la congestión de tránsito en las grandes ciudades, así como facilitar los desplazamientos a personas que no dispongan de coche propio. Además, supone una notable disminución de emisiones de CO2, al reducir el número de coches en las carreteras. A su vez, favorece las relaciones sociales entre personas que realizan los mismos recorridos.

Este sistema es algo nuevo para América, y en especial para México. Sin embargo, desde hace ya algunos años se viene utilizando en Europa logrando cautivar la atención de otros países que tienen la necesidad de disminuir sus niveles de contaminación y de mejorar la economía de sus habitantes.

* 1. **Android**

Es un sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tablets y también para relojes inteligentes, televisores y automóviles. Inicialmente fue desarrollado por Android Inc., empresa que Google respaldó económicamente y más tarde, en 2005, compró. Android fue presentado en 2007 junto la fundación del Open Handset Alliance (un consorcio de compañías de hardware, software y telecomunicaciones) para avanzar en los estándares abiertos de los dispositivos móviles. El primer móvil con el sistema operativo Android fue el HTC Dream y se vendió en octubre de 2008.

La estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un framework Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual Dalvik con compilación en tiempo de ejecución hasta la versión 5.0, luego cambio al entorno Android Runtime (ART).

Las bibliotecas escritas en lenguaje C incluyen un administrador de interfaz gráfica (surface manager), un framework OpenCore, una base de datos relacional SQLite, una Interfaz de programación de API gráfica OpenGL ES 2.0 3D, un motor de renderizado WebKit, un motor gráfico SGL, SSL y una biblioteca estándar de C Bionic. El sistema operativo está compuesto por 12 millones de líneas de código, incluyendo 3 millones de líneas de XML, 2,8 millones de líneas de lenguaje C, 2,1 millones de líneas de Java y 1,75 millones de líneas de C++.

* + 1. **Android sdk 16**

Jelly Bean fue presentado el 24 de julio de 2012 en el evento "Breakfast with Sundar Pichai". A fecha de octubre de 2013 cuenta con una base de usuarios del 1,5% del total de versiones del sistema operativo.

Jelly Bean incorporó una serie de nuevas funciones entre las que destacan el soporte de Bluetooth de baja energía, de OpenGL ES 3.0 y de resolución 4K. Por primera vez en Android se incluyen los idiomas hebreo, árabe, afrikáans, amhárico, hindi, suajili y zulú. Posee un sistema de localización Wi-Fi en segundo plano, mejoras en la seguridad y en la escritura, así como modo de perfiles con acceso restringido.

Además se incluye la función de autocompletado en el Dial pad para facilitar el marcado de números, Google Now como asistente personal que proporciona información contextualizada, y la posibilidad, desde un Nexus 7 y un Nexus 10 de ver el contenido en una televisión.

En cuanto a los desarrolladores, se han incorporado nuevas opciones para creadores de apps, sistema de logging, mejoras en el análisis de compilación de aplicaciones y APIs DRM de mayor calidad. Mejoras en la seguridad y encriptación WiFi, mejoras en los codificadores de vídeo e inclusión de códec VP8, monitorización de notificaciones.

* + 1. **Android sdk 25**

Android Nougat es una versión del sistema operativo para dispositivos móviles Android. Fue dado a conocer el 18 de mayo de 2016 en el evento Google I/O.

Android Nougat incluye una característica llamada multi ventana, que permite usar 2 aplicaciones a la vez con la pantalla dividida. Fabricantes como Samsung implementó la multiventana desde el Galaxy S III. Pero ahora es compatible con cualquier dispositivo Android.

Soporte para realidad virtual

Esta versión de Android soporta de forma nativa la realidad virtual. Aunque la realidad virtual fue implementada en smartphones como el Galaxy S6 y S7. Ahora es compatible de forma nativa, es decir, con cualquier dispositivo Android.

Daydream

Es un nuevo Play Store pero dedicado a la realidad virtual. Desde aquí podremos descargar Juegos o Apps destinados especificamente para la realidad virtual. Actualmente existen pocos móviles compatibles: (Google) Pixel, Pixel XL, todas las variantes de Motorola Moto Z (excepto moto Z play) y recientemente ZTE Axon 7.

Doze On The Go

Una mejora de Doze (Introducido por primera vez en Android Marshmallow). Ahora Doze no solo se activa cuando el movil está encima de la mesa, ahora también entra en acción a la hora de apagar la pantalla

**3.2 Lenguajes de programación:**

**3.2.1 Java:**

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como WORA, o "write once, run anywhere"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es, a partir de 2012, uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente-servidor de web, con unos 10 millones de usuarios reportados.

**3.2.2 Json:**

JSON, acrónimo de JavaScript Object Notation, es un formato de texto ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript aunque hoy, debido a su amplia adopción como alternativa a XML, se considera un formato de lenguaje independiente.

Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos es que es mucho más sencillo escribir un analizador sintáctico (parser) de JSON. En JavaScript, un texto JSON se puede analizar fácilmente usando la función eval(), lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la comunidad de desarrolladores AJAX, debido a la ubicuidad de JavaScript en casi cualquier navegador web.

**3.2.3 XML:**

XML, siglas en inglés de eXtensible Markup Language, traducido como "Lenguaje de Marcado Extensible" o "Lenguaje de Marcas Extensible", es un meta-lenguaje que permite definir lenguajes de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible. Proviene del lenguaje SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML) para estructurar documentos grandes. A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones deben comunicarse entre sí o integrar información.

**3.2.4 JavaScript:**

Javascript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript podemos crear diferentes efectos e interactuar con nuestros usuarios.

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros.

Es necesario resaltar que hay dos tipos de JavaScript: por un lado está el que se ejecuta en el cliente, este es el Javascript propiamente dicho, aunque técnicamente se denomina Navigator JavaScript. Pero también existe un Javascript que se ejecuta en el servidor, es más reciente y se denomina LiveWire Javascript.

**3.2.5 Node.js**

Node.js es un entorno JavaScript del lado del servidor, basado en eventos. Node ejecuta JavaScript utilizando el motor V8, desarrollado por Google para uso de su navegador Chrome. Aprovechando el motor V8 permite a Node proporciona un entorno de ejecución del lado del servidor que compila y ejecuta JavaScript a velocidades increíbles. El aumento de velocidad es importante debido a que V8 compila Javascript en código de máquina nativo, en lugar de interpretarlo o ejecutarlo como bytecode. Además de la alta velocidad de ejecución de JavaScript, la verdadera magia detrás de Node.js es algo que se llama Bucle de Eventos (Event Loop). Para escalar grandes volúmenes de clientes, todas las operaciones intensivas I/O en Node.js se llevan a cabo de forma asíncrona. El enfoque tradicional para generar código asíncrono es engorroso y crea un espacio en memoria no trivial para un gran número de clientes. Para evitar esta ineficiencia, así como la dificultad conocida de las aplicaciones basadas en hilos, (programming threaded applications), Node.js mantiene un event loop que gestiona todas las operaciones asíncronas. Node es de código abierto, y se ejecuta en Mac OS X, Windows y Linux.

* 1. **Modelo Vista Controlador**

**3.3.1 Modelo**

Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Envía a la 'vista' aquella parte de la información que en cada momento se le solicita para que sea mostrada. Las peticiones de acceso o manipulación de la información se realizan a través del controlador.

**3.3.2 Controlador**

Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro en una base de datos). También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta el 'modelo' (por ejemplo, desplazamiento o scroll por un documento o por los diferentes registros de una base de datos)

**3.3.3 Vista**

Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.

* 1. **Plataformas.**
     1. **Firebase**

Firebase es una plataforma fundada en 2011 y comprada posteriormente por Google en 2014, que permite crear y desarrollar aplicaciones para dispositivos móviles de alta calidad y de forma muy rápida. Su objetivo es ayudar a facilitar la gestión de la app, el aumento de la base de usuarios y, por ende, su monetización.

Firebase está disponible tanto para Android, como para iOS o web. Además, contiende varias y diversas funcionalidades que todo desarrollador puede combinar y adaptar a sus propias necesidades.

**Integrated Development Environment (IDE)**

Un entorno de desarrollo integrado1 2 o entorno de desarrollo interactivo, en inglés Integrated Development Environment (IDE), es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

Normalmente, un IDE consiste de un editor de código fuente, herramientas de construcción automáticas y un depurador. La mayoría de los IDE tienen auto-completado inteligente de código (IntelliSense). Algunos IDE contienen un compilador, un intérprete, o ambos, tales como NetBeans y Eclipse; otros no, tales como SharpDevelop y Lazarus.

* + 1. **AndroidStudio**

Android Studio es el entorno de desarrollo integrado oficial para la plataforma Android. Fue anunciado el 16 de mayo de 2013 en la conferencia Google I/O, y reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. La primera versión estable fue publicada en diciembre de 2014.

Está basado en el software IntelliJ IDEA de JetBrains, y es publicado de forma gratuita a través de la Licencia Apache 2.0. Está disponible para las plataformas Microsoft Windows, Mac OS X y GNU/Linux.

* + 1. **GitHub**

Es una plataforma de desarrollo colaborativo en la cual podemos alojar proyectos utilizando un sistema de control de versiones, el código se almacena de forma publica pero se puede hacer de forma privada utilizando la versión paga de github.

Características

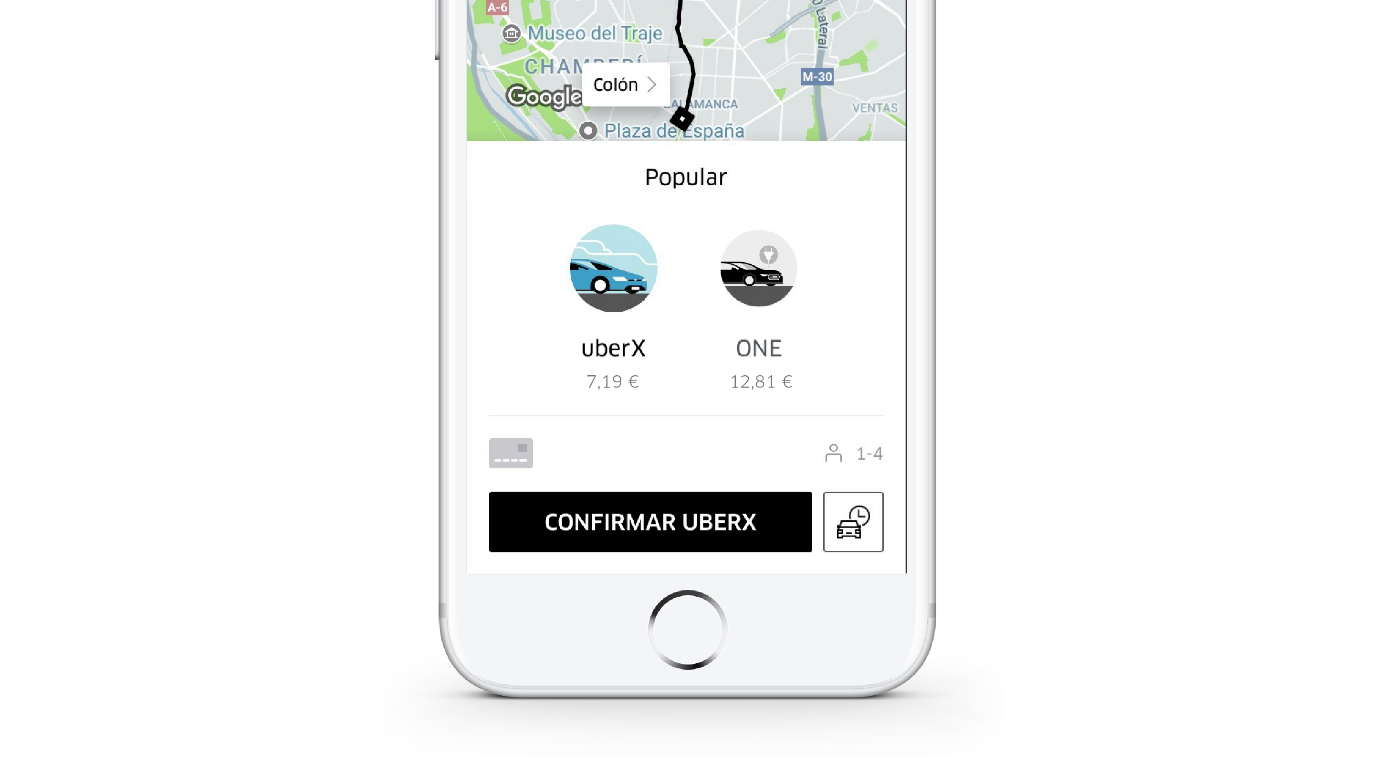
* Wiki para cada proyecto
* Página web para cada proyecto1
* Gráfico para ver cómo los desarrolladores trabajan en sus repositorios y bifurcaciones del proyecto
* Funcionalidades como si se tratase de una red social, como por ejemplo: seguidores;
* Herramienta para trabajo colaborativo entre programadores.
* Gestor de proyectos de estilo Kanban2

**4. ESTADO DEL ARTE**

La contaminación de CO por parte de los vehículos ha aumentado drásticamente con el paso de los años, para combatir este problema han ido naciendo diferentes plataformas tanto que ayudan al humano a transportarse como al medio ambiente para disminuir el CO producido por diferentes vehículos innecesariamente, algunas de estas plataformas son:

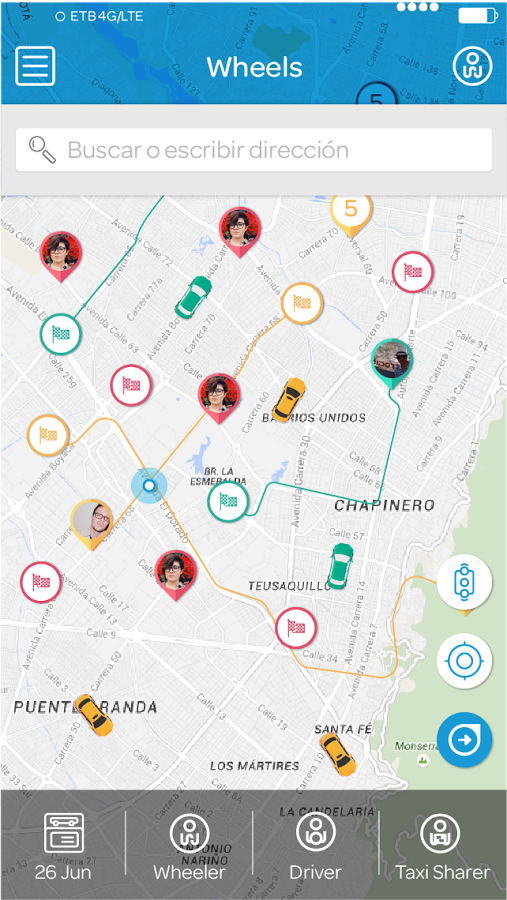
**4.1 UBER:**

Uber es líder en el mercado de transportar sus usuarios por conductores particulares algo parecido a un taxi, uber maneja distintos tipos de servicio uberX, uberPOOL, UberBLACK, UberVAN, UberANGEL, UberBICI, UberEATS. En este caso nos concentraremos en uberPOOL ya que este es el mismo servicio que se presta con uberX pero con la posibilidad que un usuario que se dirige a un destino que este dentro de la ruta que escogió el usuario principal pueda unirse a la ruta sin ningún inconveniente y llevarlo, el pago se vería reducido para el cliente principal (el que escogió el destino principal). Por medio de este método todos ganan tanto el usuario principal como los usuarios que se unieron a la ruta y el medio ambiente ya que se redujo la contaminación de CO ya que no se necesitaron 3 o 4 vehículos para llegar a sus destinos, sino solamente uno.



**4.2 WHEELS:**

Wheels es una plataforma la cual se encarga de compartir vehículos, servicios de taxi o planes que se realicen en bicicleta, en este caso nos enfocaremos en compartir el vehiculo o el servicio de taxi. Wheels consiste en definir un grupo de personas o ingresar a grupos de instituciones o empresas, el siguiente paso a realizar en la plataforma es definir la ruta que se está creando si eres conductor o si eres un pasajero esperando a que alguien te de cupo en su ruta, si eres el creador de una ruta en taxi, por ultimo definir con quienes vas a viajar enviando una notificación a los que te llamen la atención para concretar los detalles del viaje y como se encontraran.



**5. METODOLOGÍA**

**Desarrollo evolutivo:**

Se basa en la idea de desarrollar una implementación inicial, exponiéndola a los comentarios del usuario y refinándola a través de las diferentes versiones hasta que se desarrolla un sistema adecuado. Las actividades de especificación, desarrollo y validación se entrelazan en vez de separarse, con una rápida retroalimentación entre estas.

La ventaja de un proceso del software que se basa en un enfoque evolutivo es que la especificación se puede desarrollar de forma creciente. Tan pronto como los usuarios desarrollen un mejor entendimiento de su problema, éste se puede reflejar en el sistema software.

Para sistemas pequeños y de tamaño medio (500.000 líneas de código), el enfoque evolutivo de desarrollo es el mejor. Los problemas del desarrollo evolutivo se hacen particularmente agudos para sistemas grandes y complejos con un periodo de vida largo, donde diferentes equipos desarrollan distintas partes del sistema. Es difícil establecer una arquitectura del sistema estable usando este enfoque, el cual hace difícil integrar las contribuciones de los equipos. Para sistemas grandes, se recomienda un proceso mixto que incorpore las mejores características del modelo en cascada y del desarrollo evolutivo. Esto puede implicar desarrollar un prototipo desechable utilizando un enfoque evolutivo para resolver incertidumbres en la especificación del sistema. Puede entonces re implementarse utilizando un enfoque más estructurado. Las partes del sistema bien comprendidas se pueden especificar y desarrollar utilizando un proceso basado en el modelo en cascada. Las otras partes del sistema como la interfaz de usuario, que son difíciles de especificar por adelantado, se deben desarrollar siempre utilizando un enfoque de programación exploratoria.

**6. DESARROLLO DEL PROYECTO**

**6.1 REQUERIMIENTOS INICIALES**

Para determinar los requisitos iniciales que debía satisfacer nuestra aplicación, se decidió dar una mirada al sistema que están usando actualmente los estudiantes y comunidad UIS para trasladarse usando el carro compartido, el grupo llamado “UIS wheels”, donde estudiantes y comunidad UIS en general, interactúan mediante publicaciones y comentarios en estas mismas para llegar a un común acuerdo entre el conductor y los interesados de la ruta de auto compartida.

Los principales requerimientos que esto genera son los siguientes

* Visualizar una lista de las rutas activas en el momento además de permitir una búsqueda entre estas.
* Publicar una ruta con destino y hora.
* Establecer comunicación entre usuarios.

Se plantea que dentro del alcance de este proyecto, la aplicación cumpla con estos requerimientos básicos, pero a futuro, la aplicación debería responder a requerimientos tales como:

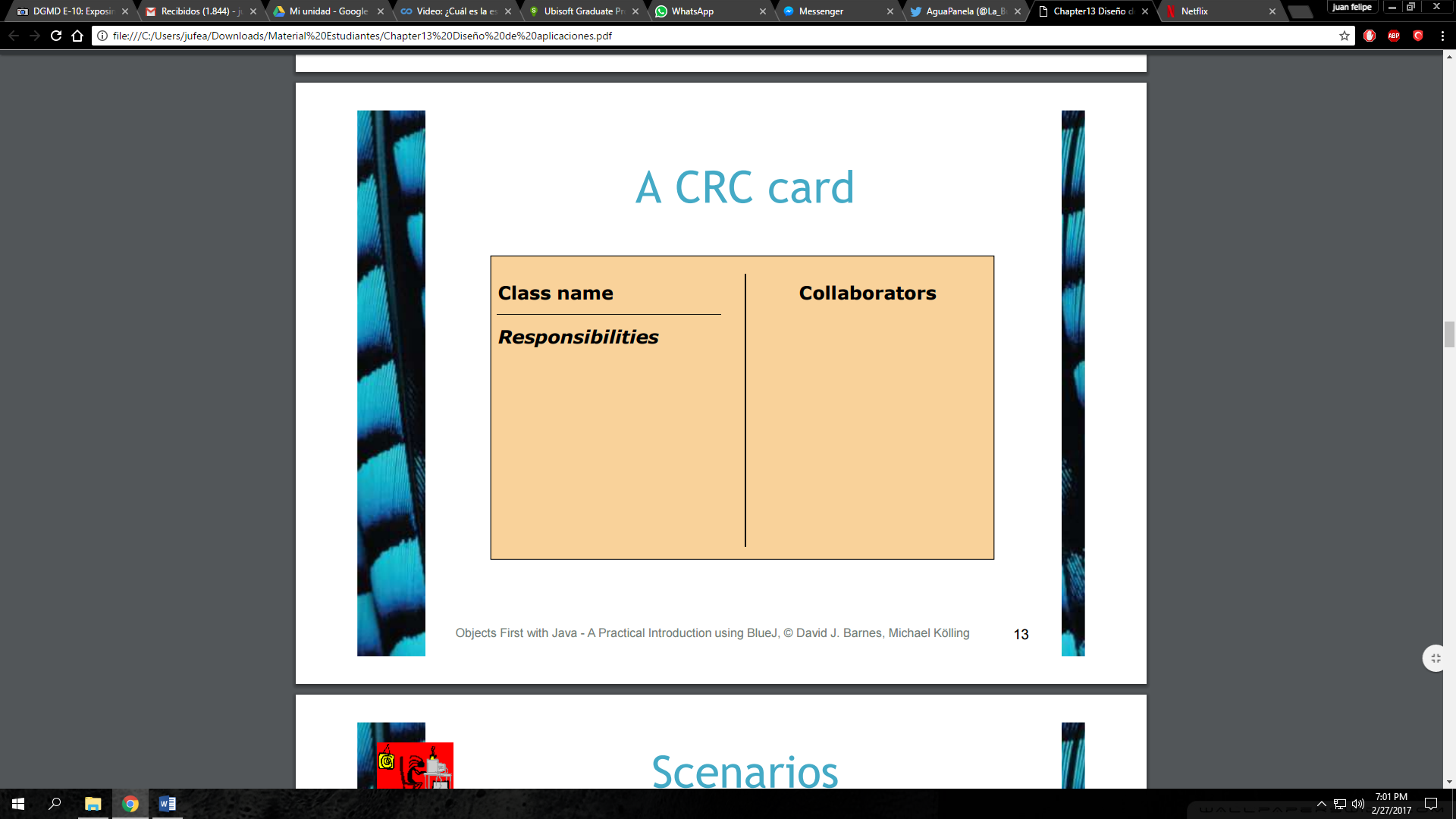
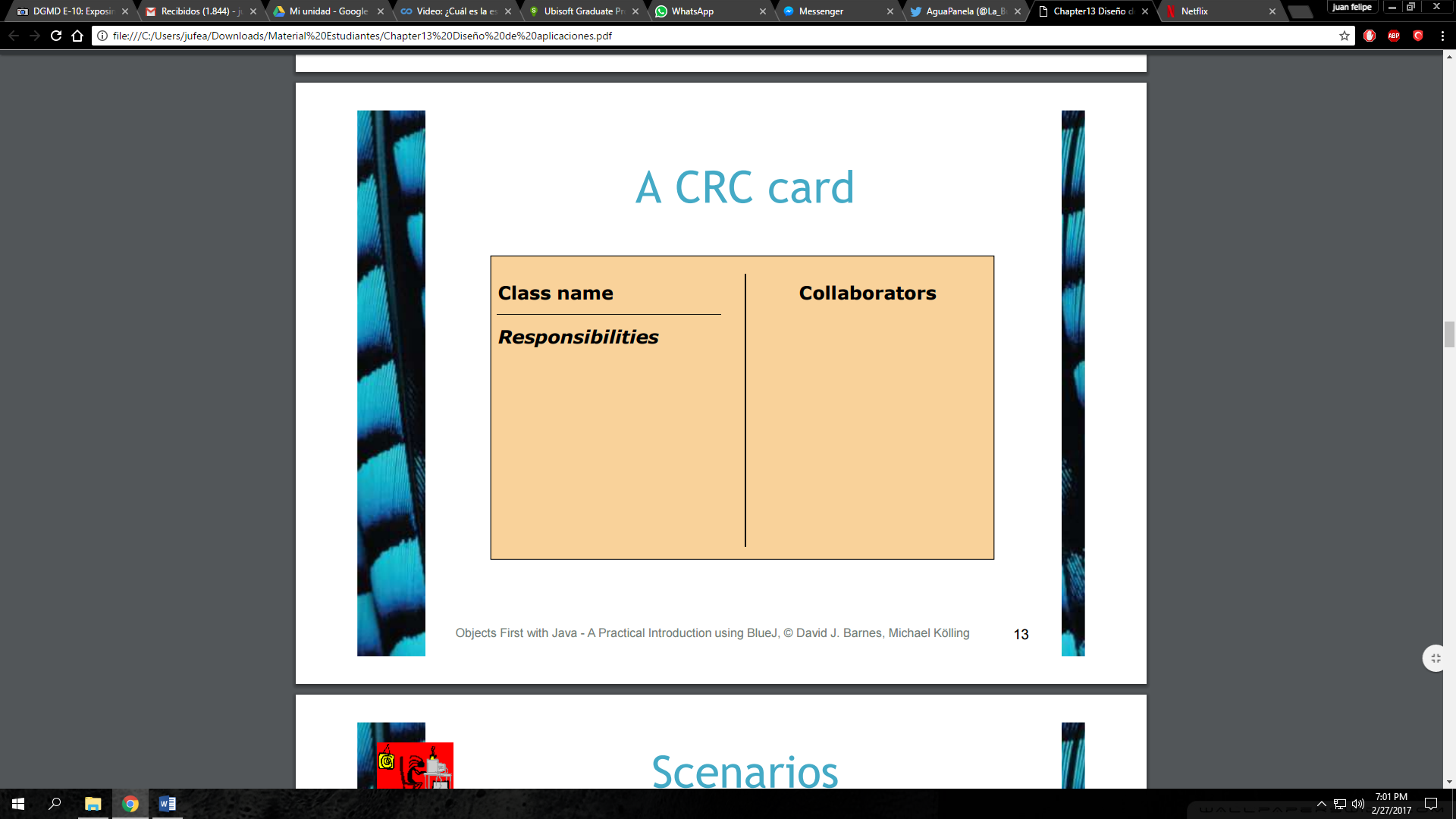
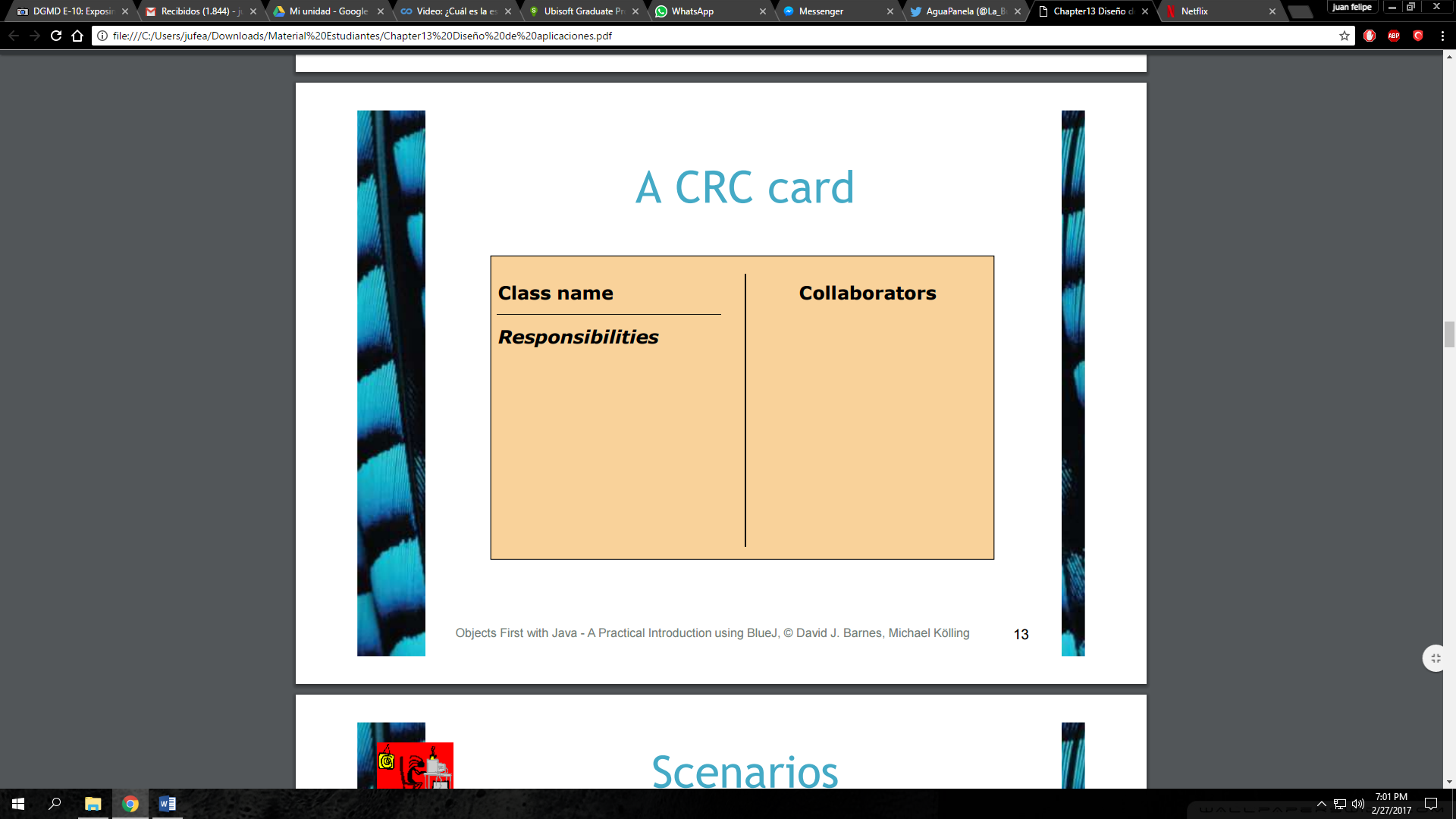
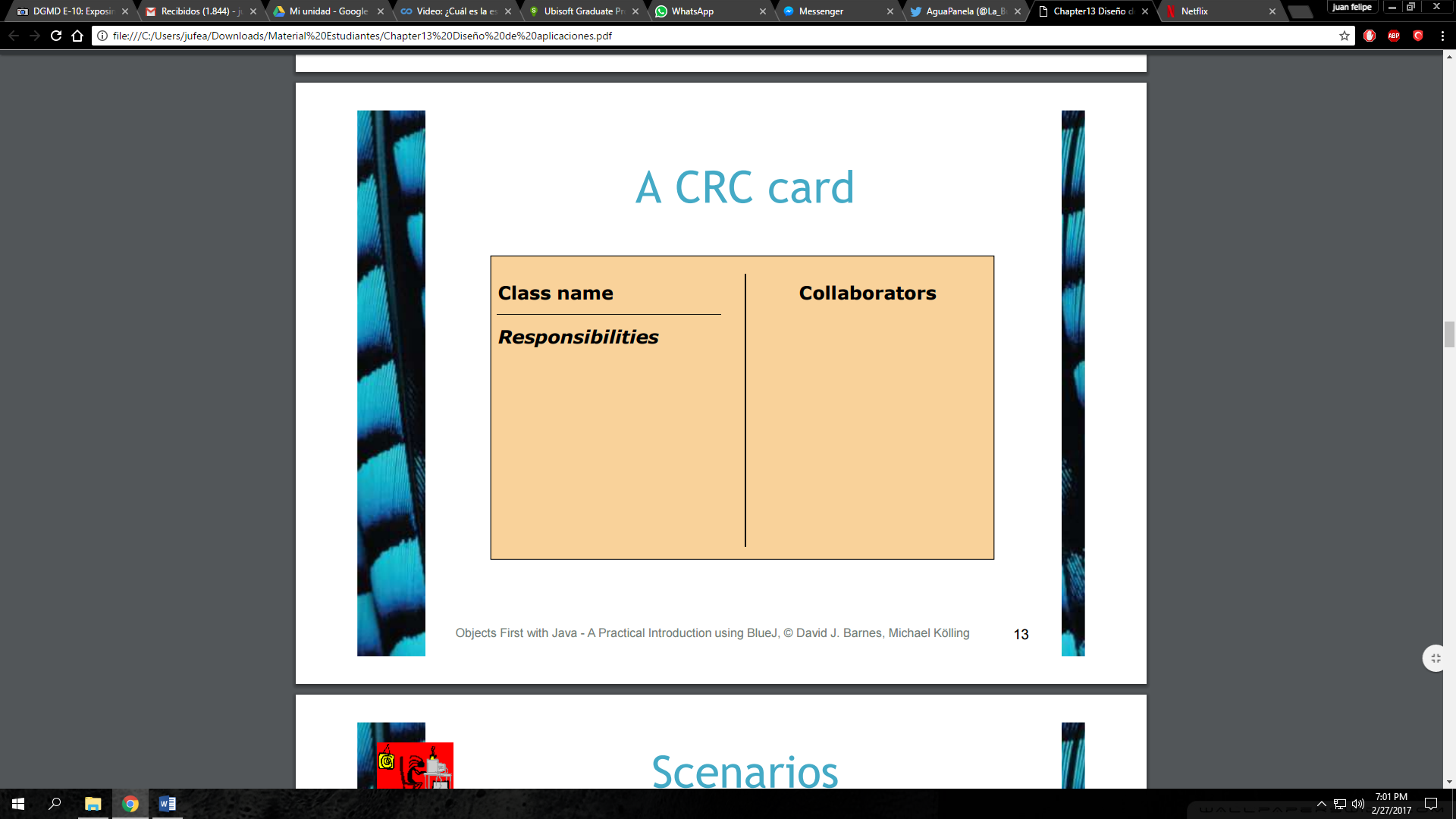
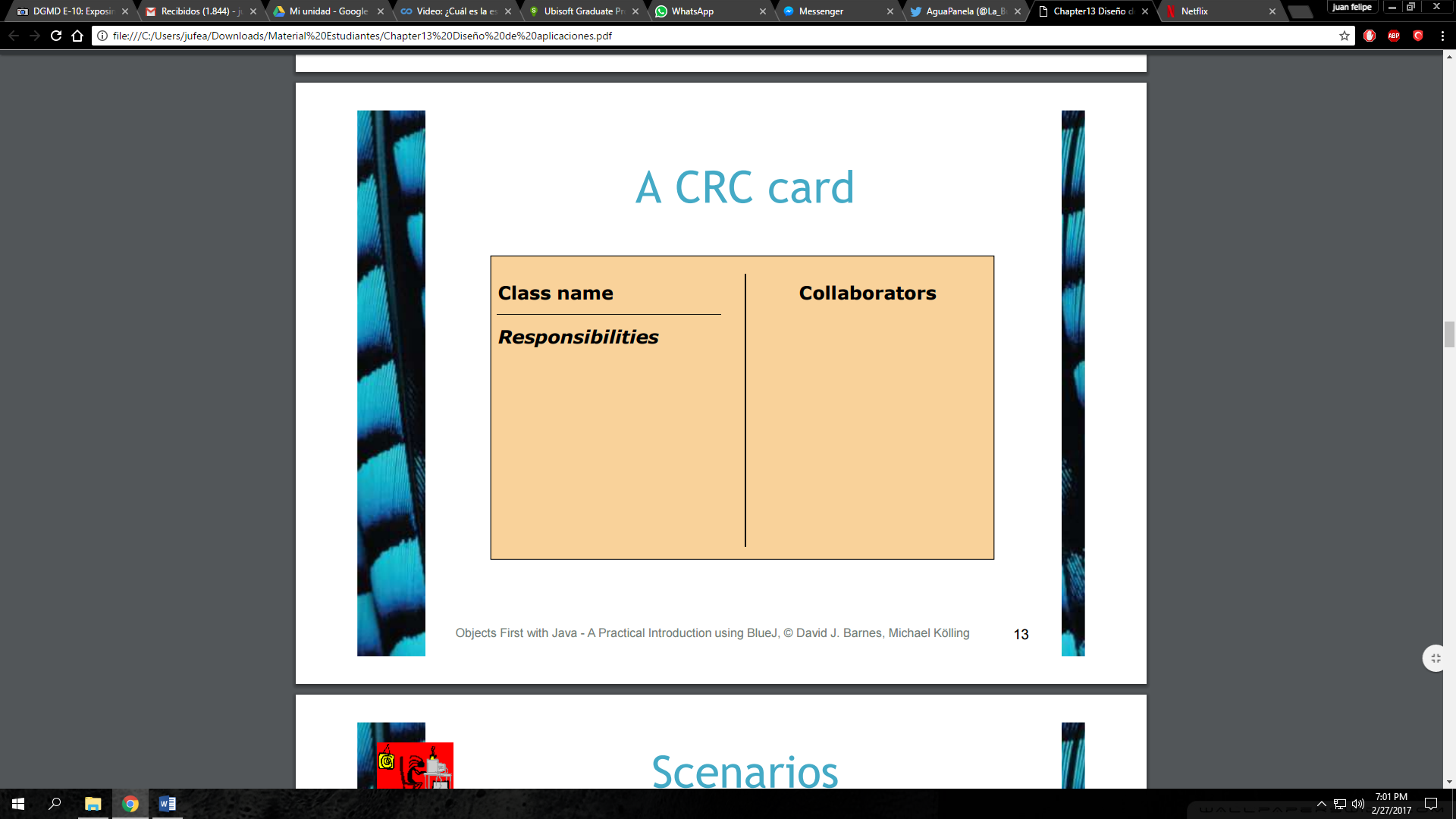
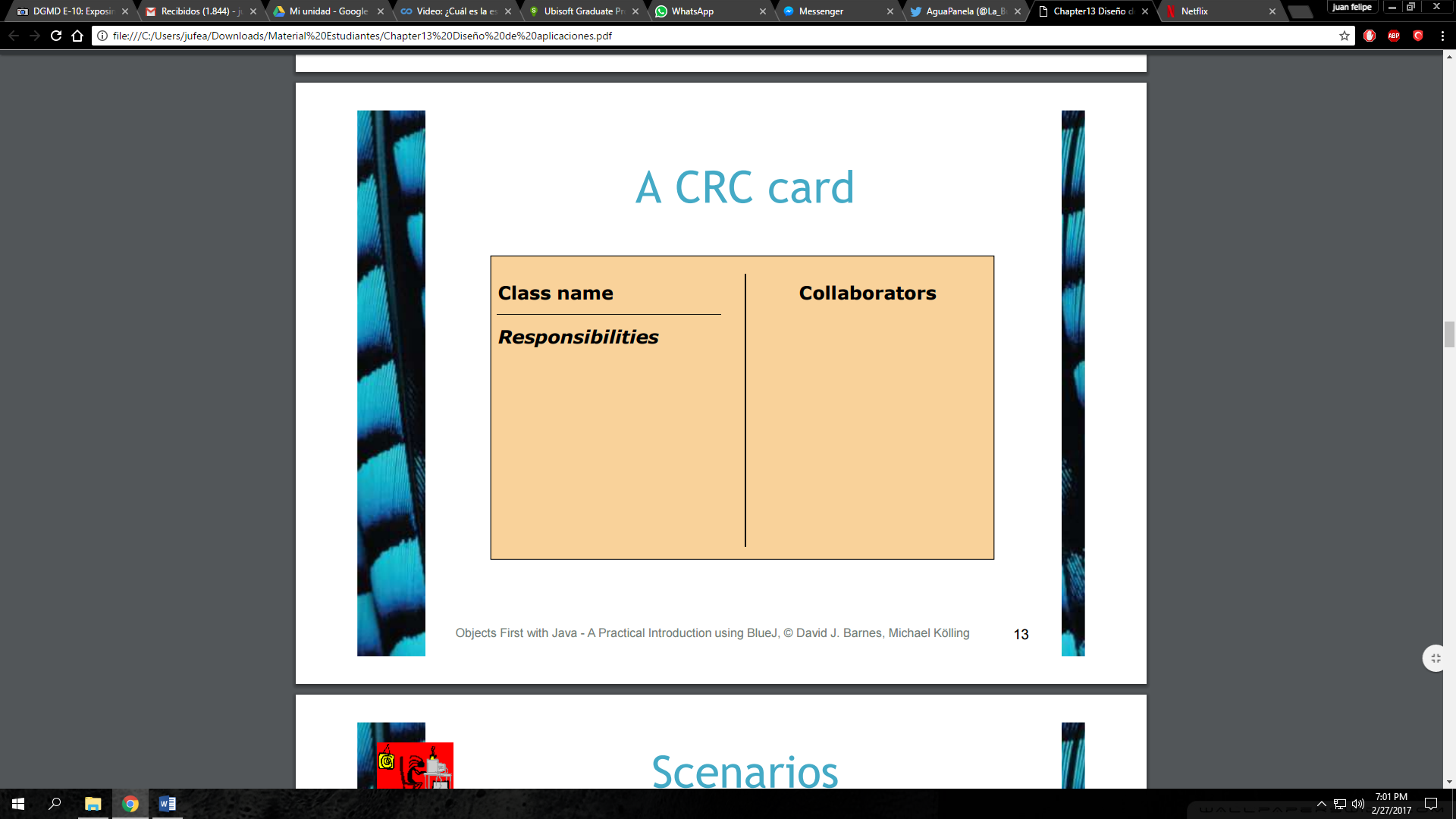
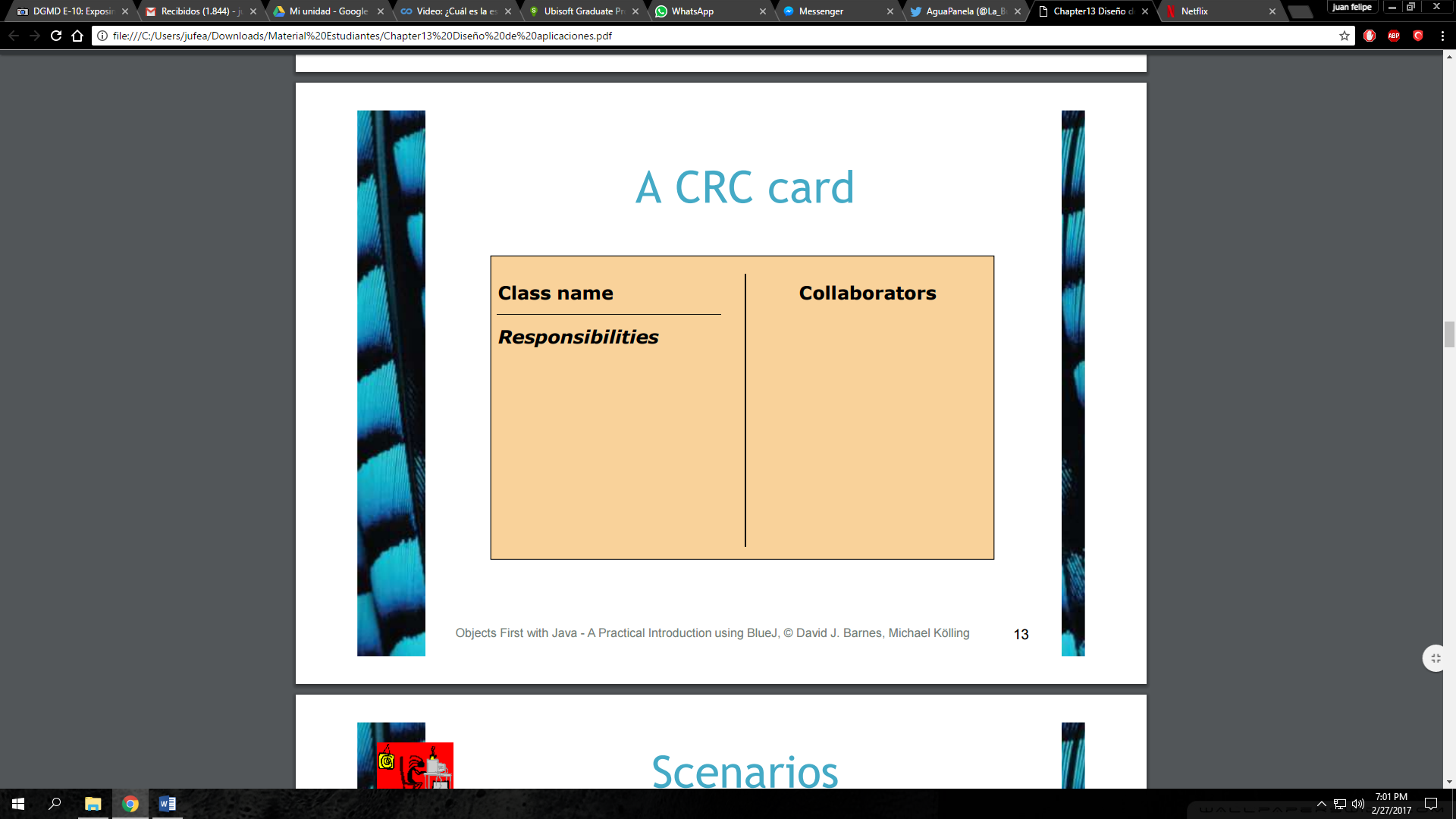
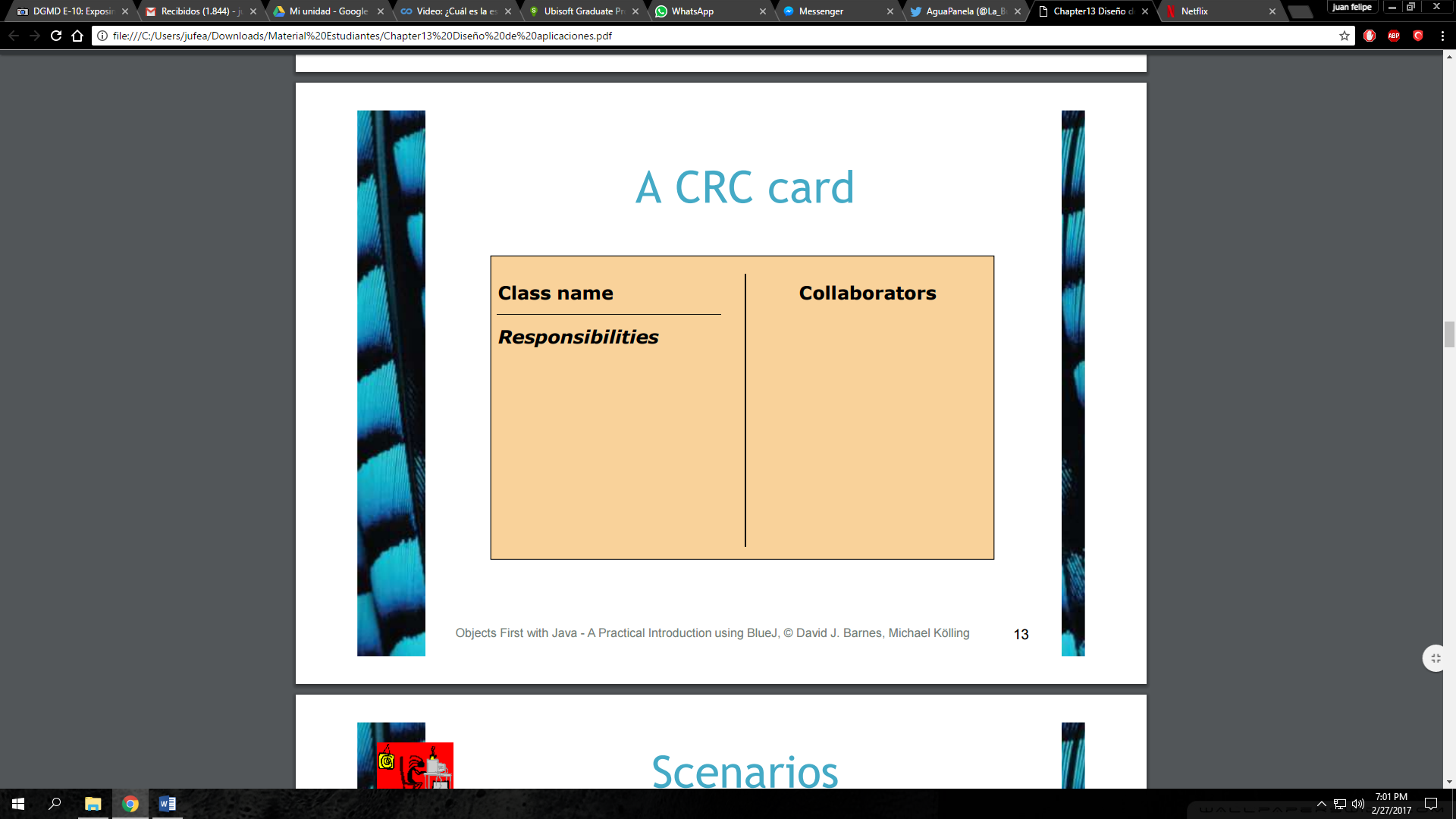
* Poder visualizar las rutas en tiempo real mediante un mapa de la ciudad.
* Implementar un sistema de chat propio de la aplicación.
* Sistemas de calificación entre usuarios.

Se decidió también diseñar la aplicación siguiendo los lineamientos de ”Material Desing”.

**7.3** **PROTOTIPO FUNCIONAL**

Para el desarrollo del primer prototipo se tomó la realimentación obtenida del prototipo no funcional y se plantearon tarjetas CRC, técnica creada por Kent Beck y Ward Cunningham. Usada para el diseño de software orientado a objetos consiste en dibujar una tarjeta por cada clase u objeto y dividirla en tres zonas, el nombre de la clase, las responsabilidades de esta y sus colaboradores.

Primeras tarjetas CRC



Usuario

Colaboradores

-Permite crear nuevos usuarios además del uso del perfil de Facebook para dicho fin

-Comunica usuarios entre si

-Permite reportar otros usuarios

-Permite ver perfiles de otros usuarios

-lleva conteo de reportes al usuario

BuscadorRuta

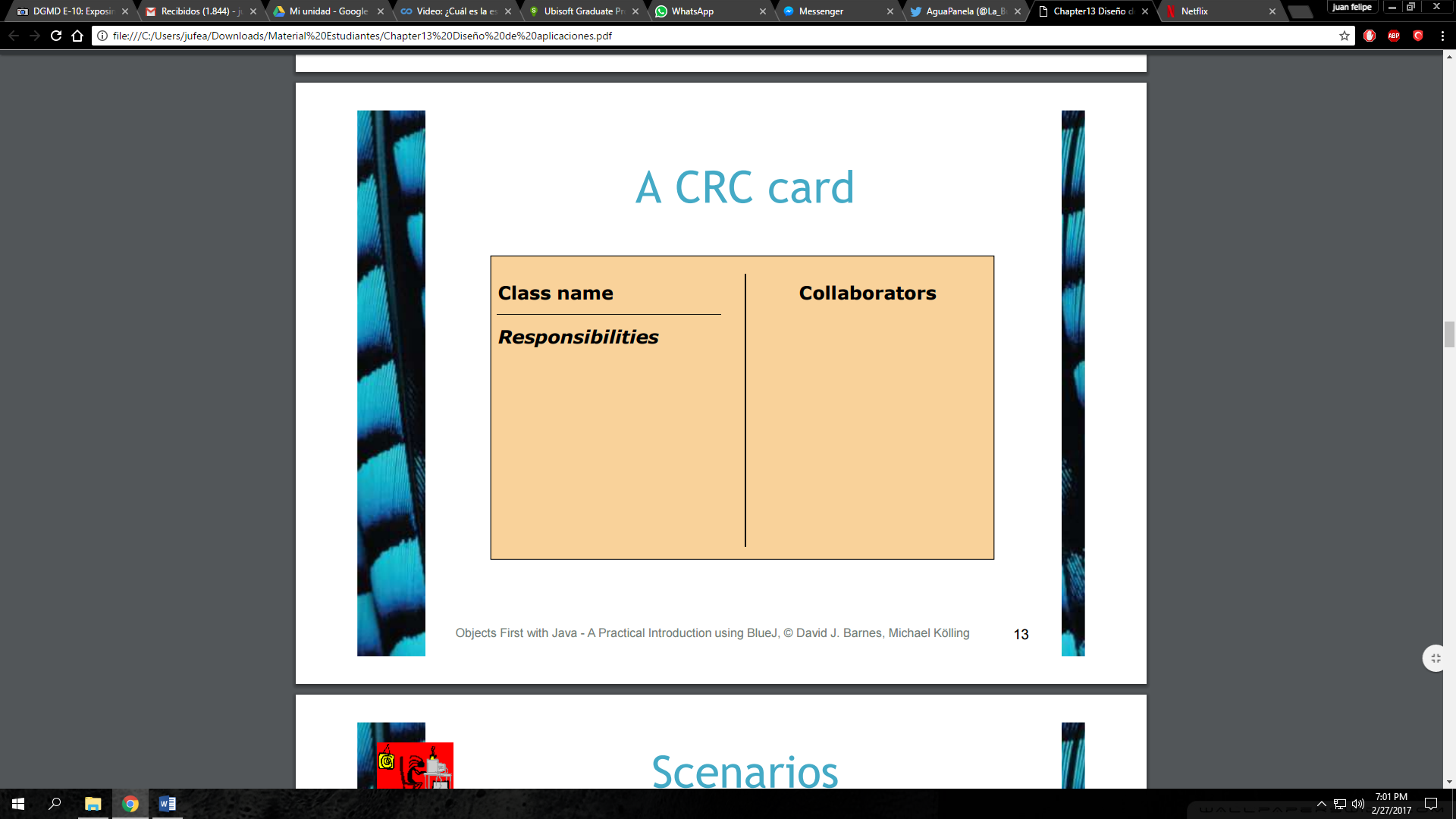
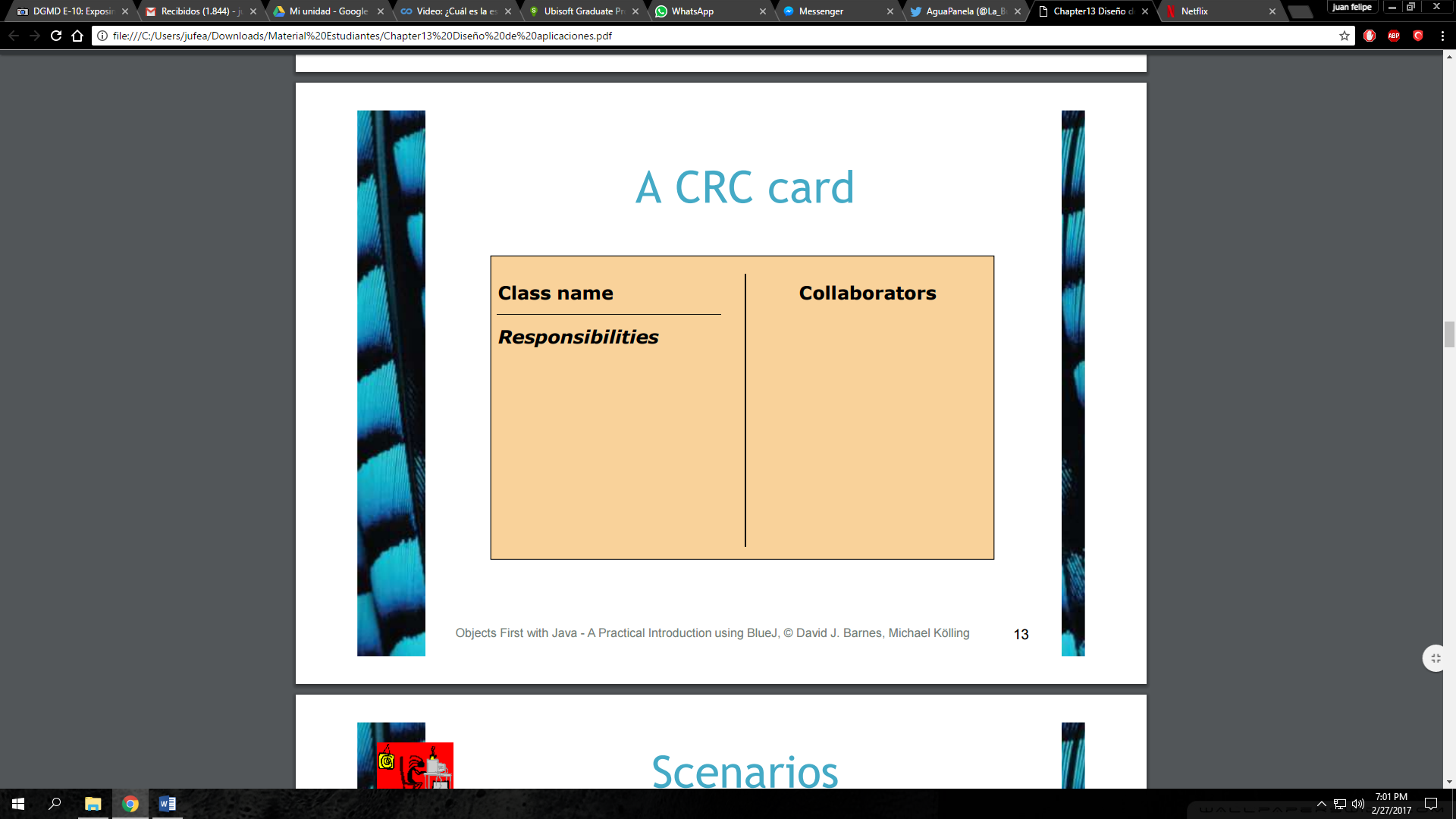
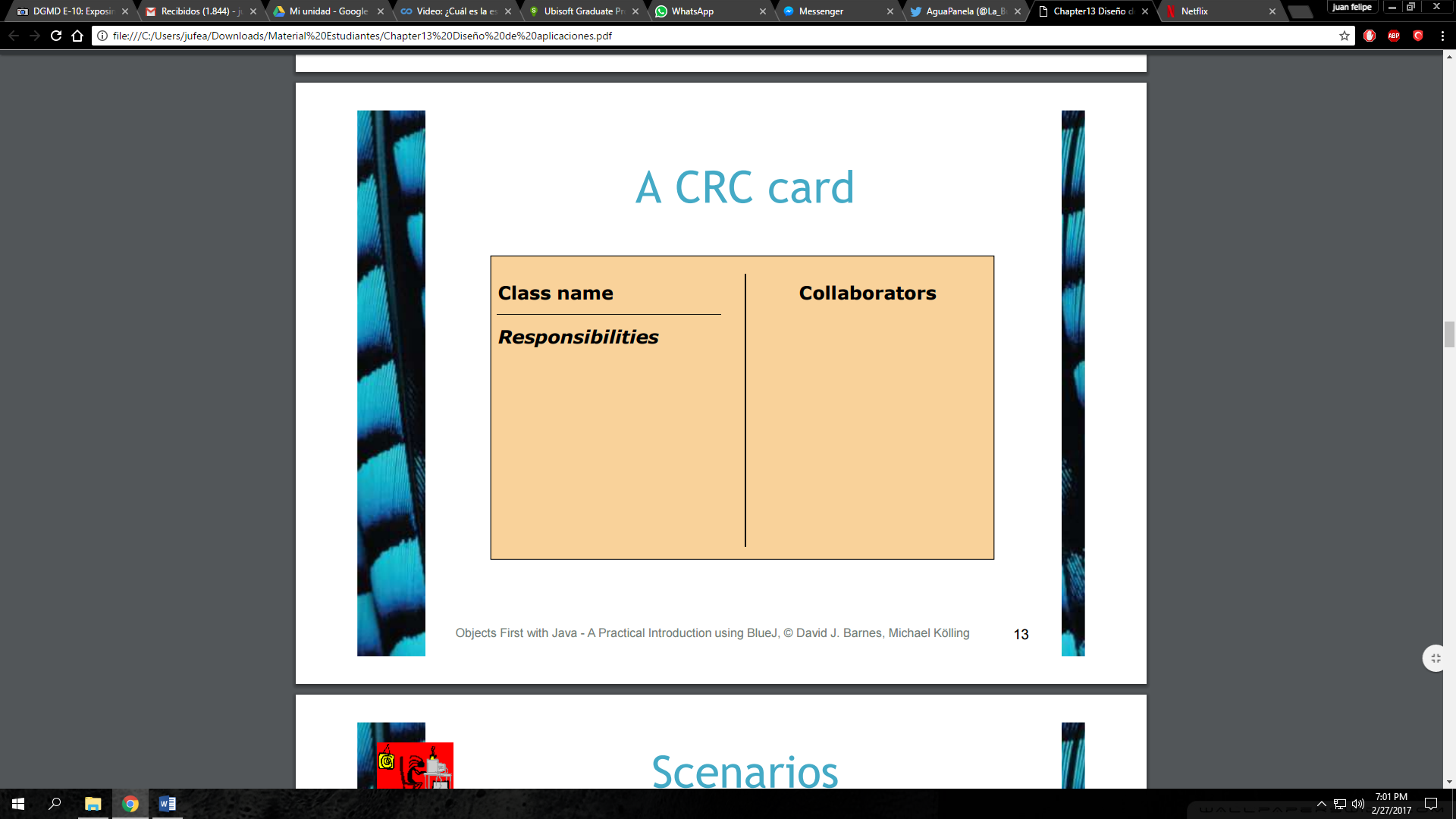
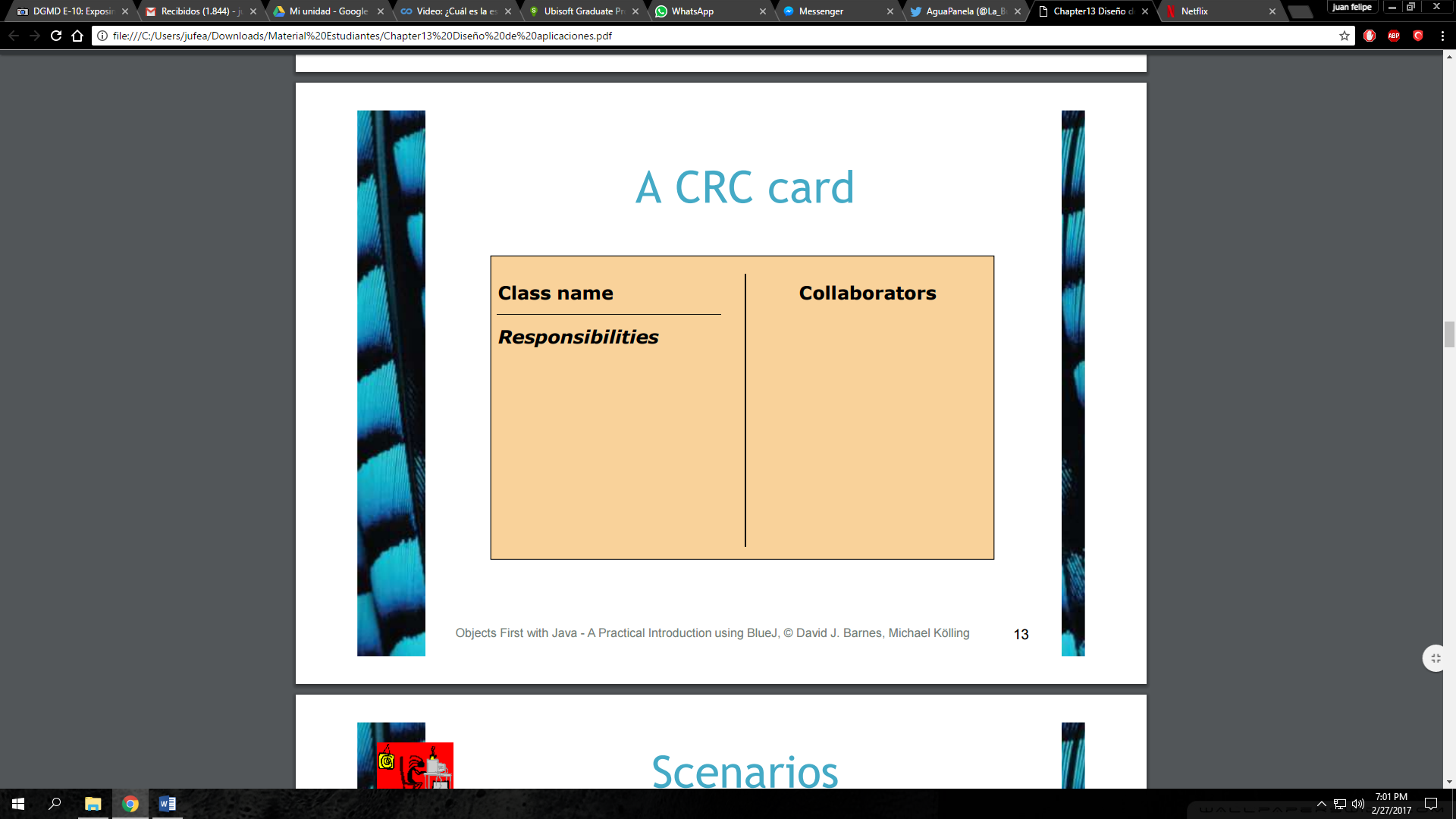
Colaboradores

Ruta

usuario

-Buscar entre las diferentes rutas

-Permite filtrar por destinos u horas



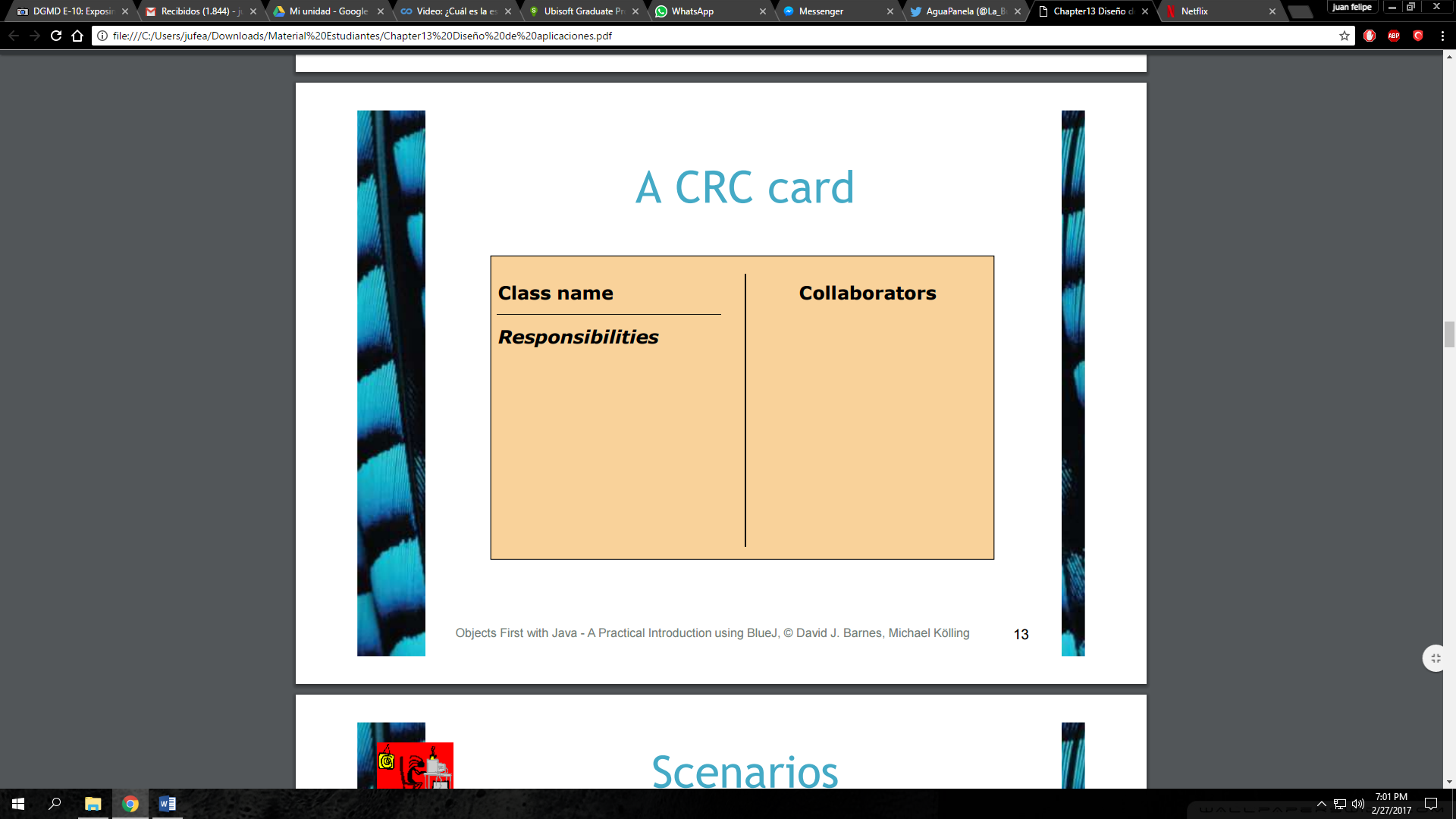
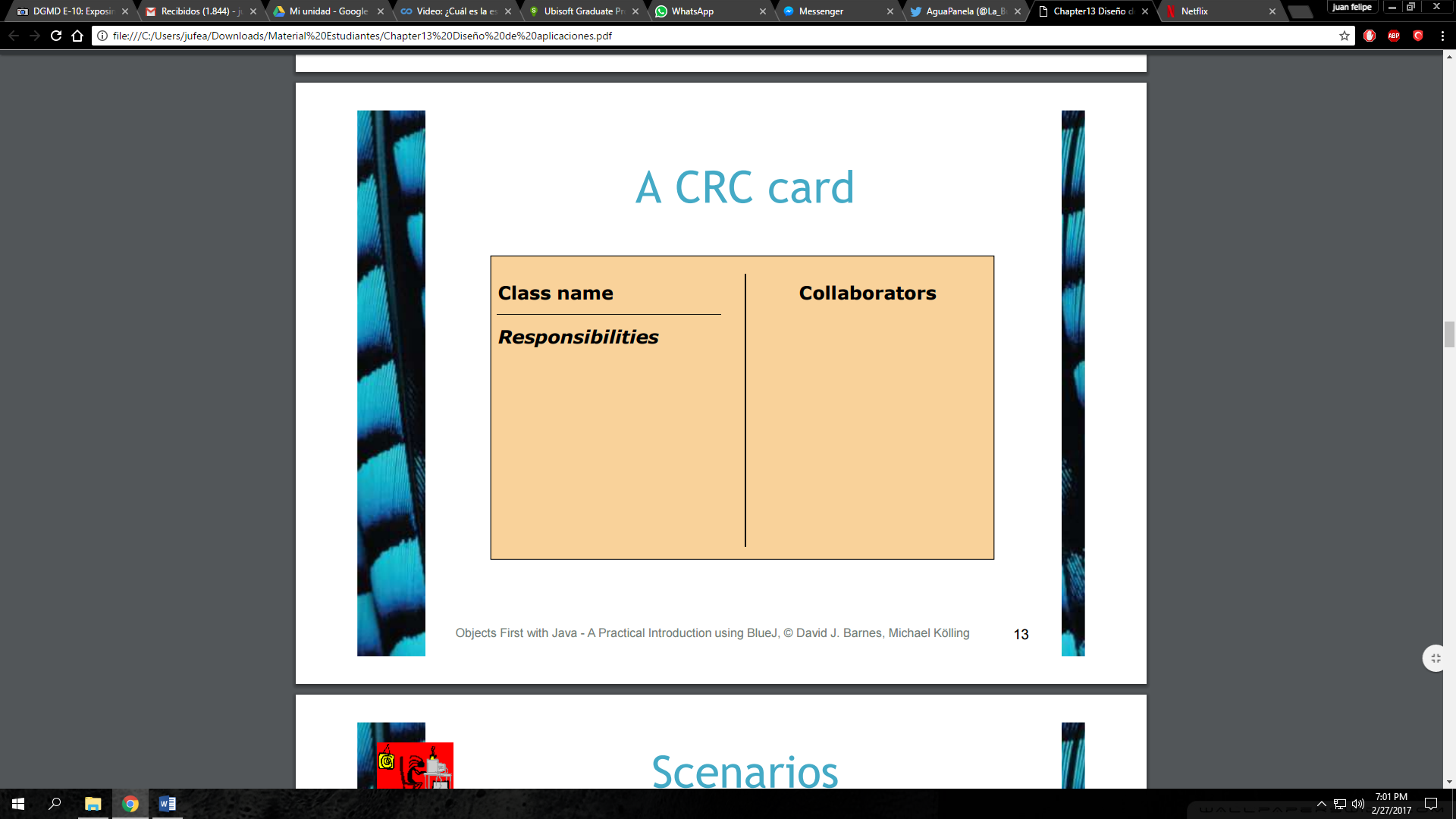
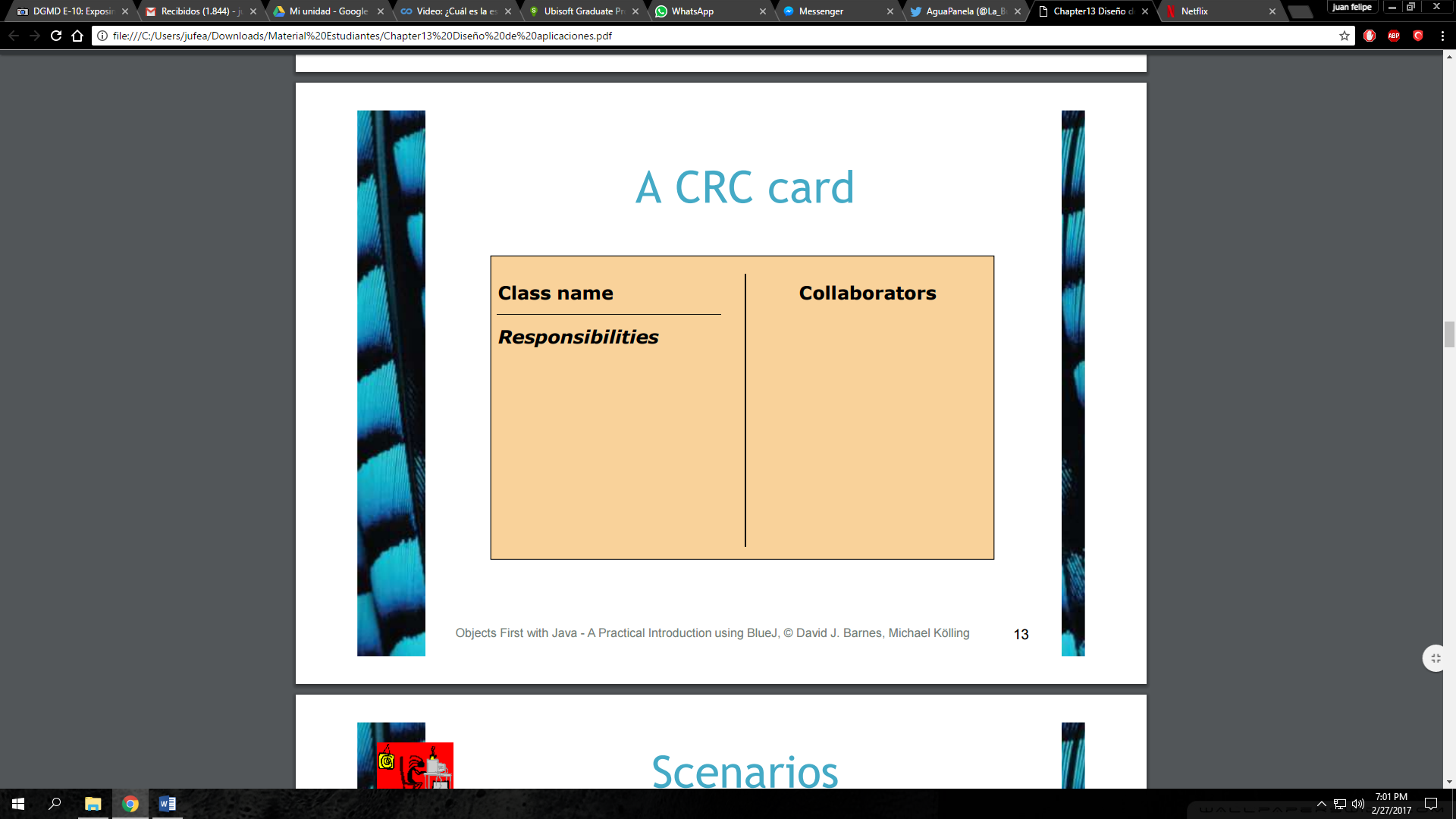
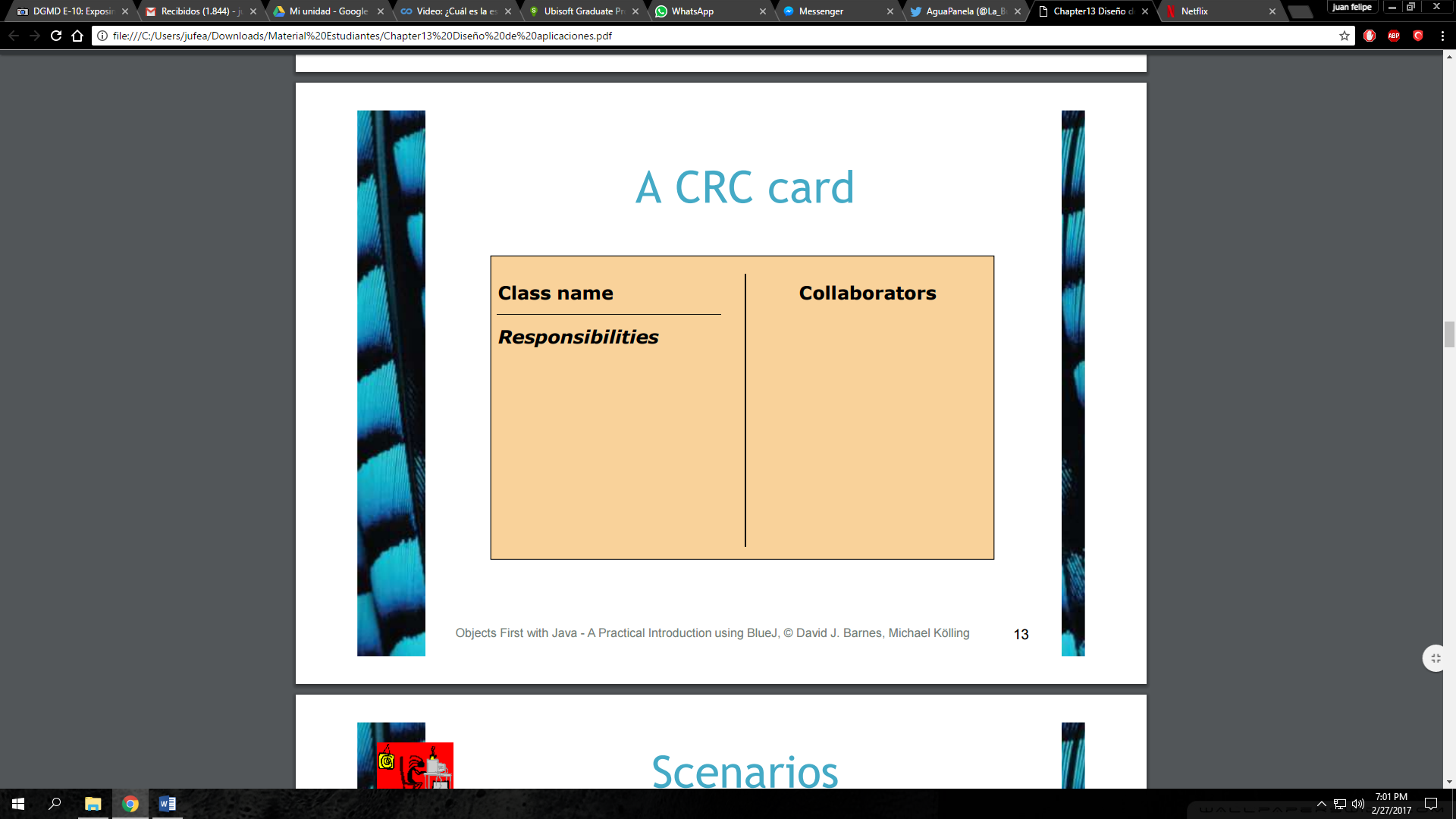
Pasajero

Colaboradores

usuario

-Permite calificar conductores

-Hereda de usuario



Conductor

Colaboradores

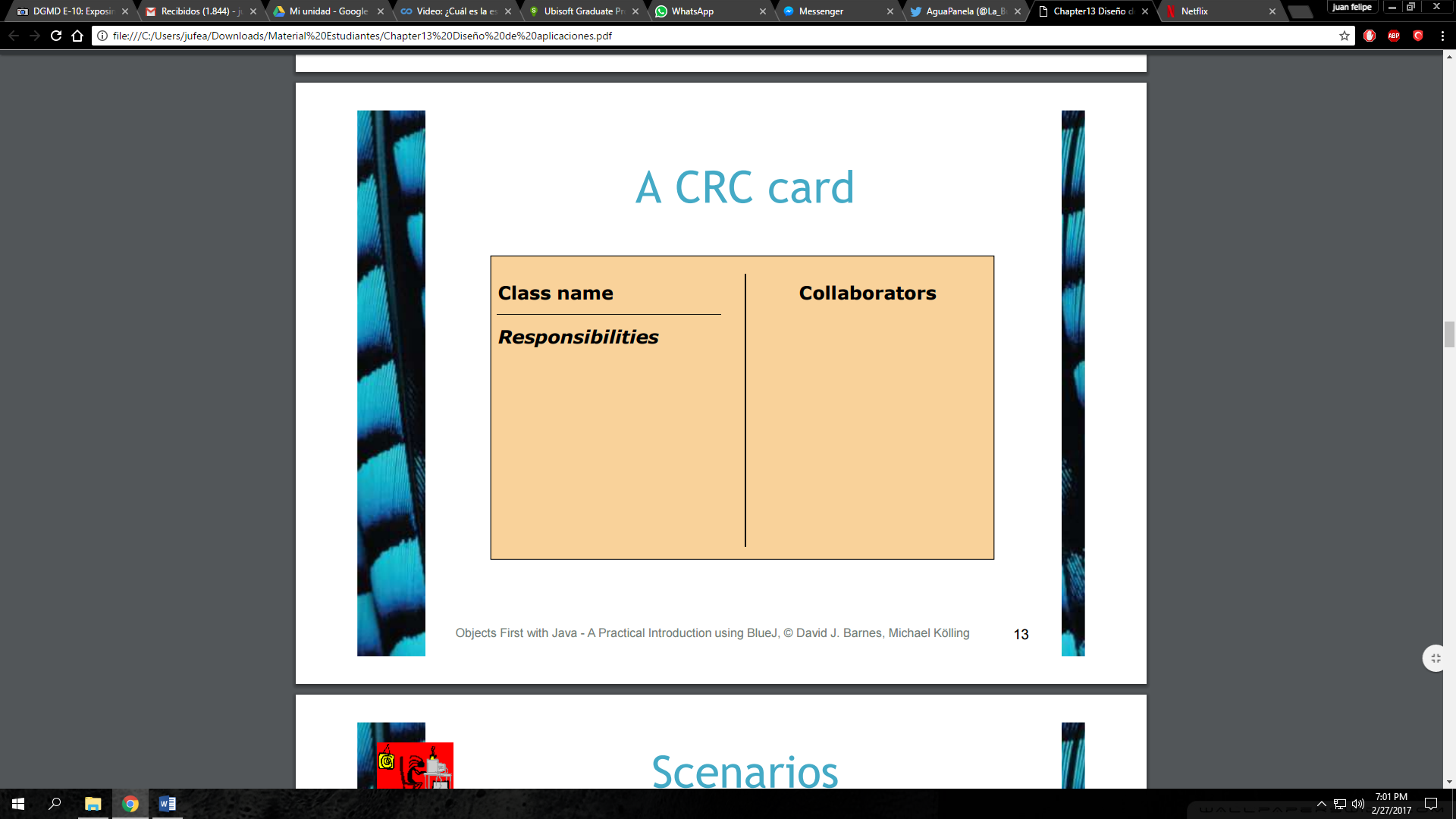
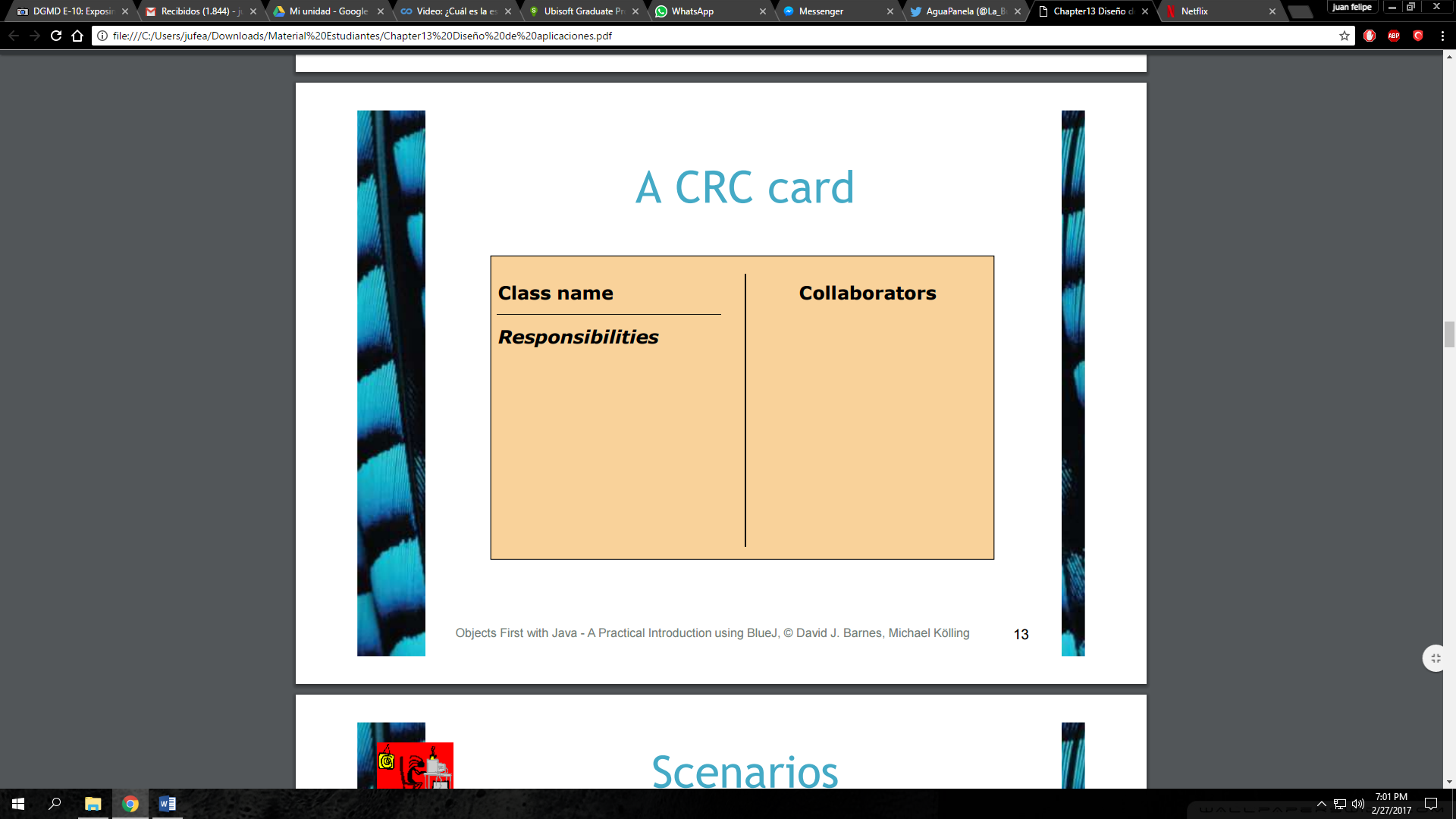
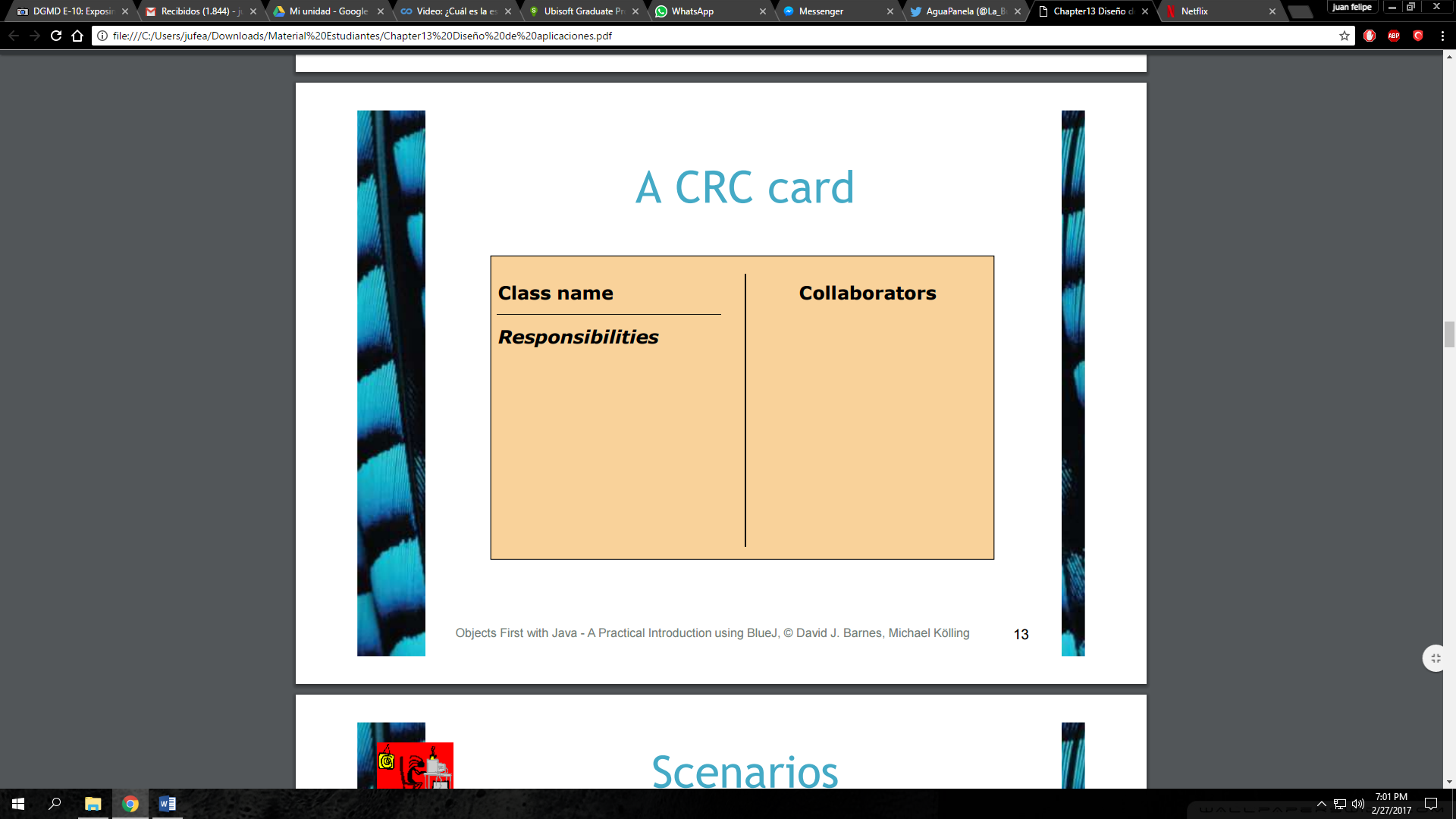
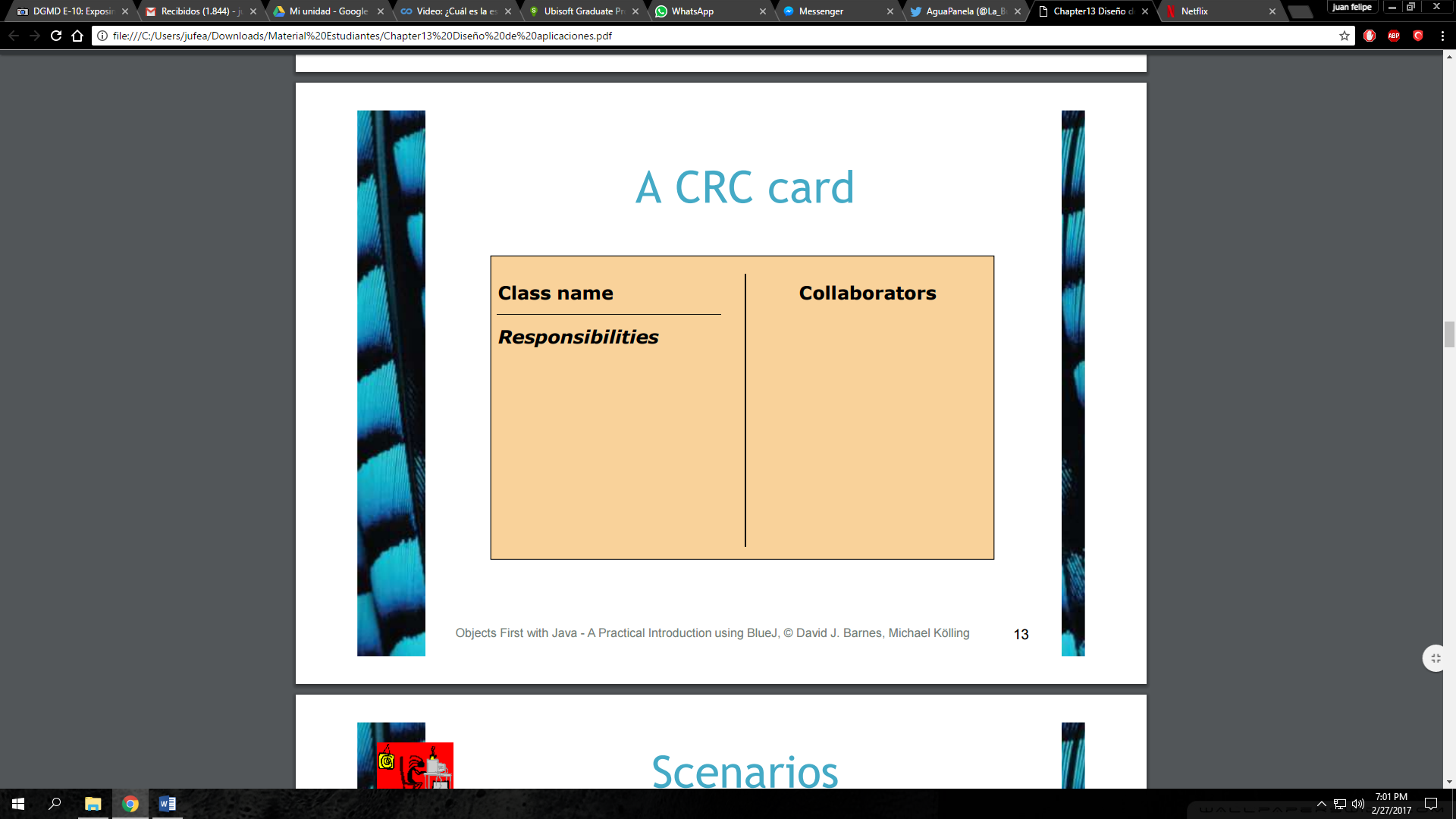
usuario

-Permite publicar rutas

-Hereda de Usuario

-Permite agregar especificaciones del automóvil

-Definir cupo del automovil



Ruta

Colaboradores

usuario

-Permite la creación del recorrido por el usuario con un destino, origen, hora y cupo

-Visualización en el mapa

-Muestra cupos disponibles en rutas afines

-Muestra la información del conductor que postea la ruta

A partir de estas tarjetas se realiza el primer diseño de clases a utilizar en este proyecto, como base para iniciar el desarrollo del primer prototipo que fue evolucionando hasta convertirse en el prototipo final.

**7.3.1** **PRIMER PROTOTIPO**

Para el desarrollo del primer prototipo se pensó en dar solución al primer requerimiento, la visualización de las rutas en tiempo real, se investigó por la web la mejor opción a la hora de establecer una base de datos que nos permitiera obtener esta funcionalidad, que fuese de bajo costo, sostenible y se adaptara a las necesidades que teníamos en ese momento.

Lo más usual es adquirir un web service

**6. CRONOGRAMA**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Actividades\semanas | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
| **Análisis** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Diseño** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Desarrollo** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Test** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Mantenimiento** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**7. PRESUPUESTO**



**8. BIBLIOGRAFÍA**

WIKIPEDIA, JAVA (lenguaje de programación),[en línea] [citado en: 18/05/2017] disponible en: <https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)>

WIKIPEDIA, JSON, [en línea][citado en:18/05/2017] disponible en: <https://es.wikipedia.org/wiki/JSON>

WIKIPEDIA, Extensible Markup Language (XML) [en línea] [citado en: 18/05/2017] disponible en: <https://es.wikipedia.org/wiki/Extensible_Markup_Language>

DESARROLLO WEB, Modelo vista controlador, [en línea] [citado en: 18/05/2017] disponible en: <https://desarrolloweb.com/articulos/que-es-mvc.html>

FIREBASE, FIREBASE, [en línea] [citado en: 18/05/2017] disponible en: <https://firebase.google.com/?hl=es-419>

WIKIPEDIA, Android Studio, [en línea] [citado en: 18/05/2017] disponible en: <https://es.wikipedia.org/wiki/Android_Studio>

ANDROID, ANDROID, [en línea] [citado en: 18/05/2017] disponible en: <https://developer.android.com/index.html>

WIKIPEDIA, Entorno de desarrollo integrado (IDE) [en línea] [citado en: 18/05/2017] disponible en: <https://es.wikipedia.org/wiki/Entorno_de_desarrollo_integrado>

WIKIPEDIA, Android Nougar, [en línea] [citado en: 18/05/2017] disponible en: <https://es.wikipedia.org/wiki/Android_Nougat>

WIKIPEDIA, Android Jelly Bean, [en línea] [citado en: 18/05/2017] disponible en: [**https://es.wikipedia.org/wiki/Android\_Jelly\_Bean**](https://es.wikipedia.org/wiki/Android_Jelly_Bean)

WIKIPEDIA, ANDROID, [en línea] [citado en: 18/05/2017] disponible en: [**https://es.wikipedia.org/wiki/Android**](https://es.wikipedia.org/wiki/Android)

Ian, Sommerville “Desarrollo evolutivo” en: Ingenieria de software Novena edición, Editorial Pearson.