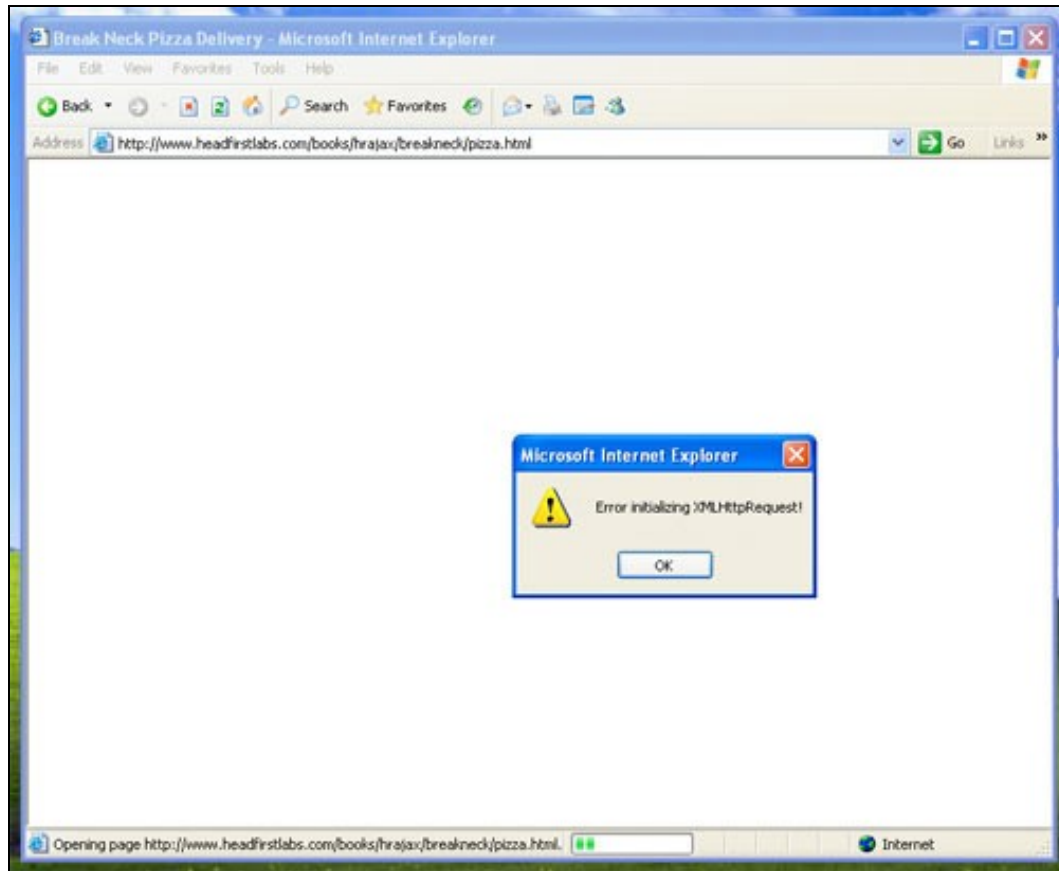


El navegador Internet Explorer y Ajax

Todo parece muy bonito? hasta que probamos este código en Internet Explorer (en la version 7.0 funciona correctamente, pero en anteriores, no).

Si lo hacemos obtendremos algo como la **figura 1**.



Vemos claramente que no está funcionando la creación del objeto en Internet Explorer.

Aproximadamente un 70% de las personas que navegan en la red, usan Internet Explorer. En otras palabras, que tendremos que corregir nuestro código para que trabaje correctamente con Internet Explorer.

Microsoft dice que soporta Ajax, pero ha creado su propia version de XMLHttpRequest. De hecho ha realizado varias versiones.

Si estamos usando una version moderna de Internet Explorer tendremos que usar un objeto llamado Msxml2.XMLHTTP; algunas versiones más antiguas usan el objeto Microsoft.XMLHTTP.

Por lo tanto tenemos que soportar estos dos tipos de objetos (sin perder el soporte que actualmente tenemos para navegadores que no son Microsoft).

En el listado 3 se ha añadido al código inicial el soporte a todos los navegadores Microsoft, para la creación del objeto XMLHttpRequest.

Listado 3. Añadimos soporte a navegadores Microsoft.

```
<script language="javascript" type="text/javascript">
var request = false;
try {
    request = new XMLHttpRequest();
} catch (trymicrosoft) {
    try {
        request = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (othermicrosoft) {
        try {
            request = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (failed) {
            request = false;
        }
    }
}
```

```
    }  
  }  
}  
  
if (!request)  
  alert("Error inicializando XMLHttpRequest!");  
</script>
```

Vamos a revisar este código de nuevo paso a paso:

1. Creamos una nueva variable llamada request y le asignamos un valor false. Usaremos esta variable para saber si el objeto ha sido creado correctamente o no.
2. Añadimos el bloque de control de excepciones try/catch:
3. Intentamos crear el objeto XMLHttpRequest.
4. Si da un error (catch (trymicrosoft)):
5. Intentamos crear un objeto compatible con las nuevas versiones de Microsoft (Msxml2.XMLHTTP).
6. Si da un error (catch (othermicrosoft)), intentamos crear un objeto compatible Microsoft usando las versiones más antiguas (Microsoft.XMLHTTP).
7. Si da un error (catch (failed)), nos aseguramos de que request tiene un valor false.
8. Comprobamos el valor de request
9. Si ha habido algún problema (y request es false), empleamos JavaScript para alertar a los usuarios de que ha habido un problema.

Si realizamos los cambios y probamos el código veremos que obtenemos el resultado de la figura 2.

Figura 2. Internet Explorer funcionando correctamente.

