



UNIVERSIDAD DE GRANADA

Práctica LEX
Modelos de Computación



Luis González Romero luisgonromero@correo.ugr.es

9 de diciembre de 2018

Índice

1. Problema abordado	4
2. Explicación de como se ha abordado	5
3. Ejemplo de ejecucion	14

Índice de figuras

1.	Pantalla de inicio de Instant-Gaming	4
2.	Código fuente de un videojuego	4
3.	Declaraciones fichero lex	5
4.	Reglas fichero lex	5
5.	Definiciones fichero lex	6
6.	Fragmento de código html donde actúa <i>spec_total</i>	6
7.	Fragmento de código html donde actúa <i>price</i>	7
8.	Fragmento de código html donde actúa <i>description</i>	7
9.	Fragmento de código html donde actúa <i>tags</i>	7
10.	Fragmento de código html donde actúa <i>game_name</i>	8
11.	Fragmento de código html donde actúa <i>release_date</i>	8
12.	Fragmento de código html donde actúa <i>screens</i>	8
13.	Fragmento de código html donde actúa <i>video</i>	8
14.	Definiciones fichero lex	9
15.	Método <i>main()</i>	10
16.	Método <i>to_flat_text()</i>	10
17.	Método <i>get_name()</i>	11
18.	Método <i>get_screens()</i>	12
19.	Método <i>get_price()</i>	12
20.	Método <i>get_video()</i>	13
21.	Método <i>get_tags()</i>	13
22.	Terminal tras ejecutar un ejemplo	14
23.	Carpeta creada	14
24.	Ficheros obtenidos	15
25.	Portada del videojuego	16
26.	Información obtenida	17

1. Problema abordado

Para esta práctica se ha usado la página <https://www.instant-gaming.com/es/>, obteniendo de ella códigos fuente de algunos videojuegos, obteniendo de estos información relevante: descripción, géneros, requisitos necesarios, imagen de portada...

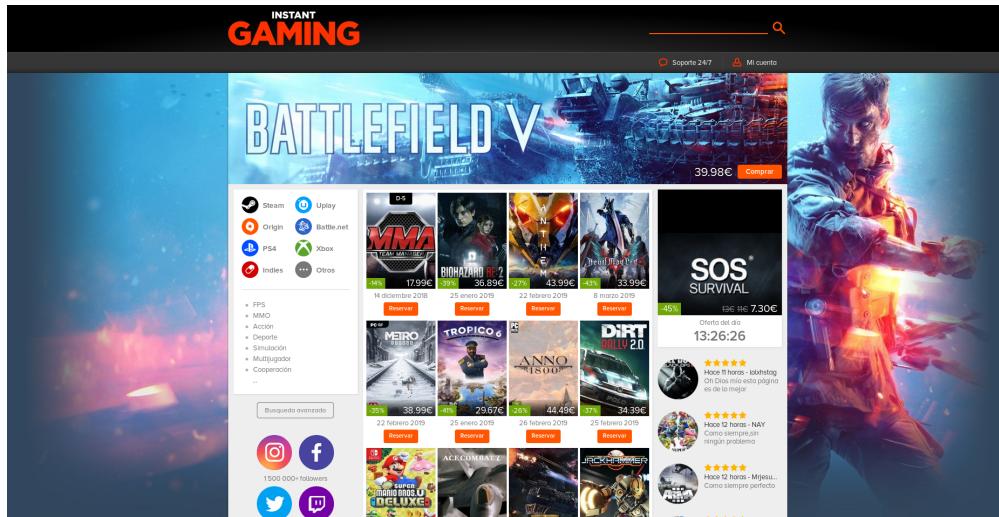


Figura 1: Pantalla de inicio de Instant-Gaming

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es" lang="es">
<head><script type="text/javascript"><window.NREUM|||NREUM=()>;nr_require=function(e,n,t){function r(t){var o=n[t]=(exports:{});e[t][0].call(o,exports,function(n){var i=e["handle"],ae=3,ue=4,f=e["ee"],get="tracer",ce="loader",s=NREUM;"undefined"==typeof window.newrelic&&(newrelics);var p=["setPageviewName","setCustomAttribute","setError",t==(),r=this,or="function"==typeof n?return i["+r"]:"tracer",c=(t.now(),e,t,r),function(){if(f.enit("?",":no-")+"fn-start",[],[c.now(),r,o],t),otry{return n.apply(this,arguments)}catch(Error(e)).i("er",[e,c.now()]))},l=2:[function(e,n,t){function r(e,n){if(!e)return 0;if(!n)return 0;if(!l)l=return 1;for(var t=l.split("."),r=0;0<t.length-1;r+=1,t[r]=t[r+1]);var i=t[t.length-1];if(i.id.aborted||!i){e&e(t,r,o);for(var a=e[0],uw=t,ru=uw.length,c=0;c<+u+1)c[i].ap(h,i,y=on,l,addEventListener,:removeEventListener,emit,t.getig,listeners:v,context:n,buffer:w,abort:t,aborted:1);return b}function i(){return new r}function a(){i().apply(null,arguments)}}],d=1:[function(e,n,t){function r(e,n,t){function i(){var ex,info=NREUM.info,nl=getElementsByTagName("script")[0];if(setTimeout(a.abort,3e4),!e&e.licenseKey&e.applicationID&n)return a.abort();c(y,function(n,t){e.Oxists&&performance.now()&Math.round(performance.now()):!(u=Math.max((new Date).getTime(),u))-x.offsetWidth var u=(new Date).getTime(),f=e["handle"],c=e(3),s=e("ee"),p=2,d>window,l=h+"*location,y=(beacon:'bam.nr-data.net',errorBeacon:'bam.nr-data.net',agent:'js-agent.newrelic.com/nr-1099.min.js',b=g6w&w6[m]&l/Cr05/test(navigator.userAgent),x=n.export:<title>Comprar Dragon Ball FighterZ Steam</title>
<meta name="robots" content="index,follow_all" />
<meta name="description" content="Comprar Dragon Ball FighterZ más barato en Instant-Gaming, la referencia para comprar tus juegos al mejor precio con entrega inmediata!" />
<meta name="keywords" content="Comprar,Dragon Ball FighterZ,Steam" />
<meta http-equiv="Content-Language" content="es" />
<meta name="google-signin-clientid" content="038762527992-a3vkt0ipe4m0501kl7695adctrilmvc7.apps.googleusercontent.com" />
<meta name="ig-uid" content="6a3bf9746f987b5f7b945e2dc3fc07" />
<meta name="format-detection" content="telephone-no" />

<meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1, maximum-scale=1" />
<link rel="shortcut icon" type="image/png" href="https://www.instant-gaming.com/favicon.ico" />
<link rel="apple-touch-icon" href="https://www.instant-gaming.com/apple-touch-icon.png" />
<link rel="icon" href="https://www.instant-gaming.com/favicon.ico" type="image/x-icon" />
<link href="https://plus.google.com/+InstantGamingPG" rel="publisher" />
<link rel="canonical" href="https://www.instant-gaming.com/es/2141/comprar-key-steam-dragon-ball-fighterz/" /> <link href="/min/ig-igv1-css.css?v=1543829223" rel="stylesheet" type="text/css" />
<script src="/min/ig-igv1-js-product.js?v=1543829223" type="text/javascript"></script>
<script>
  var dataLayer = [];
  dataLayer.push({ 'countryCode': 'ES' });
  (function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
    new Date(),event:'gtm.js'});var f=d.getElementsByTagName('script')[0],
    j=d.createElement(s),dl=l.getAttribute('dataLayer');
  j.setAttribute('dataLayer',dl);
  j.async=true;j.src=
  'https://www.googletagmanager.com/gtm.js?id='+dl;
  parentNode.insertBefore(j,f);
  })(window,document,'script','dataLayer','GTM-K27M6Z');
  dataLayer.push({ 'event': 'eeProductDetailView', 'ecommerce': { 'detail': { 'products': [{ 'name': 'Dragon Ball FighterZ', 'id': '2141', 'price': '18.77', 'category': 'Steam' }]} }});
  dataLayer.push({ 'event': 'done' });
</script>
<meta property="og:title" content="Comprar Dragon Ball FighterZ Steam" />
<meta property="og:site_name" content="Instant-Gaming.com" />
<meta property="fb:app_id" content="252076411609338" />
```

Figura 2: Código fuente de un videojuego

2. Explicación de como se ha abordado

- #### ■ Declaraciones fichero lex:

```
%{
#include <cstdio>
#include <iostream>
#include <fstream>
#include <string>
using namespace std;

string game, specs, screen, description, retail_price, discount, final_price, price,
       video, tags, release_date;
char game_directory[50];

string to_flat_text(char* text);
string get_game_name(char* name);
void get_screens(char* text);
void get_price(char* text);
void get_video(char* text);
void get_tags(char* text);
%}
```

Figura 3: Declaraciones fichero lex

- #### ■ Definiciones y reglas fichero lex:

Figura 4: Reglas fichero lex

2 EXPLICACIÓN DE COMO SE HA ABORDADO

■ Definiciones:

```
coin €
letra [a-zA-Zá-úÁ-Ú]
digito [0-9]
entero [1-9]{digito}*
real ({entero}(".")){entero}","{digito}+
identificador ({letra}|_")({letra})|_|{digito}*
special [=~@:\`\\\"\\-\\.\\(\`<>\\t\\n%_\"\";#';!{?~^°~*...以上]
pic_format [png,jpg]
percent "%"
date {entero}" "{letra}+ " {entero}
spec_total "<div class=\\\"hardspecs\\\"{letra}|{special}|{entero}+\\\"</div\\\"|\\\"</td\\\"> \\"
({letra}|{special}|{digito})+\\\"</div\\\"|\\\"</td\\\">({letra}|{special}|{digito})+*\\<div class=\\\"asterix center\\\">
game_name <h1>({letra}|{special}|{digito})</h1>
retail_price <div class=\\\"retail\\\"{letra}|{special}|{digito}|{real}+{coin}</span> </div\\\">
discount <div class=\\\"discount\\\">({letra}|{special}|{digito})+{percent}</div\\\">
final_price <div class=\\\"price\\\">{real}|{coin}</div\\\">
description <div class=\\\"description\\\"{letra}|{special}|{digito}+\\\"</li><ul\\\">\\\"<span class=\\\"number\\\">\\\"<br />({letra}|{special})+\\</div\\\">
screens <meta property=\\\"og:image\\\" content=\\\"{letra}|{special}|{digito}+\\\"{pic_format}+\\\">
price <div class=\\\"buy\\\">({letra}|{special})+{retail_price}+\\\"{discount}+\\\"{final_price}+
video <iframe id=\\\"ig-vimeo-player\\\" src=\\\"{letra}|{special}|{digito}+\\</iframe>
tags <div class=\\\"tags\\\">({letra}|{special}|{digito})+\\<div class=\\\"moretags\\\">\\\"</a> </div\\\">
release_date <div class=\\\"release\\\">({letra}|{special})+{date}
```

Figura 5: Definiciones fichero lex

Para explicar las definiciones mejor, veremos fragmentos de código de un ejemplo de html.

```
<div class="hardspecs steam">
<div class="container">
<div class="title">Configuración mínima*</div>
<div class="content"><li><strong>OS:</strong> Windows 7/8/10 (64-bit OS required)
</li><li><strong>Processor:</strong> AMD FX-4350, 4.2 GHz / Intel Core i5-3470, 3.20 GHz
</li><li><strong>Memory:</strong> 4 GB RAM
</li><li><strong>Graphics:</strong> Radeon HD 6870, 1 GB / GeForce GTX 650 Ti, 1 GB
</li><li><strong>DirectX:</strong> Version 11
</li><li><strong>Network:</strong> Broadband Internet connection
</li><li><strong>Sound Card:</strong> DirectX compatible soundcard or onboard chipset</li></div>
</div>
<div class="container">
<div class="title">Configuración recomendada*</div>
<div class="content"><li><strong>OS:</strong> Windows 7/8/10 (64-bit OS required)
</li><li><strong>Processor:</strong> AMD Ryzen 5 1400, 3.2 GHz / Intel Core i7-3770, 3.40 GHz
</li><li><strong>Memory:</strong> 8 GB RAM
</li><li><strong>Graphics:</strong> Radeon HD 7870, 2 GB / GeForce GTX 660, 2 GB
</li><li><strong>DirectX:</strong> Version 11
</li><li><strong>Network:</strong> Broadband Internet connection
</li><li><strong>Sound Card:</strong> DirectX compatible soundcard or onboard chipset</li></div>
</div>
<div class="asterix center">*Los tags y los requisitos están incluidos solo con fines informativos
```

Figura 6: Fragmento de código html donde actúa *spec_total*

```
<div class="buy">
<div class="prices ">
<div class="retail" data-retail="60.00">
Precio de venta al público: <span>60€</span> </div>
<div class="discount">-69%</div>
<div class="price">18.77€</div>
```

Figura 7: Fragmento de código html donde actúa *price*

```
<div class="description" itemprop="description">DRAGON BALL FighterZ nace de lo que hace a DRAGON BALL tan famosa: luchas espe
<br />
Con Arc System Works, DRAGON BALL FighterZ maximiza los gráficos de anime de alta calidad y ofrece un juego de lucha fácil de
<br />
Gráficos de anime de alta calidad<br />
Con el poder del motor Unreal y el talentoso equipo de Arc System Works, DRAGON BALL FighterZ es toda una hazaña visual.<br />
<br />
En equipo/soporte 3vs3<br />
Forma tu equipo soñado y domina rápidas combinaciones entre luchadores.<br />
<br />
Emocionante juego en línea<br />
Partidas igualadas, salas interactivas, locas partidas de grupos de 6 jugadores... ¡Hay para todos los gustos!<br />
<br />
Modo historia exclusivo<br />
Descubre una aventura nunca vista con la Androide N°21, un nuevo personaje de creación supervisada por Akira Toriyama.<br />
<br />
Luchas espectaculares<br />
¡Experimenta combos aéreos, escenarios destructibles y escenas famosas del anime de DRAGON BALL reproducidas en una resolución
<span class="number">Ahora hay 774 personas jugando a este juego en Steam</span>
```

Figura 8: Fragmento de código html donde actúa *description*

```
<div class="tags">
<a class="tag" href="https://www.instant-gaming.com/es/juegos/acción/">Acción</a> <a class="tag" href="https://www.instant-gaming.com/es/juegos/cooperación/">Cooperación</a> <a href="https://www.instant-gaming.com/es/busquedas/?q=2D_Fighter" class="searchtag">2D Fighter</a>
<a href="https://www.instant-gaming.com/es/busquedas/?q=Competitive" class="searchtag">Competitive</a>
<a href="https://www.instant-gaming.com/es/busquedas/?q=Local_Multiplayer" class="searchtag">Local Multiplayer</a>
<a href="https://www.instant-gaming.com/es/busquedas/?q=2.5D" class="searchtag">2.5D</a>
<a href="https://www.instant-gaming.com/es/busquedas/?q=Controller" class="searchtag">Controller</a>
<a href="https://www.instant-gaming.com/es/busquedas/?q=Singleplayer" class="searchtag">Singleplayer</a>
<a href="https://www.instant-gaming.com/es/busquedas/?q=Story_Rich" class="searchtag">Story Rich</a>
<a href="https://www.instant-gaming.com/es/busquedas/?q=Difficult" class="searchtag">Difficult</a>
<div class="moretags">...</div>
```

Figura 9: Fragmento de código html donde actúa *tags*

```
<h1>Dragon Ball FighterZ</h1>
```

Figura 10: Fragmento de código html donde actúa *game_name*

```
<div class="release">Fecha de lanzamiento  
<span>26 enero 2018</span>
```

Figura 11: Fragmento de código html donde actúa *release_date*

```
<meta property="og:image" content="https://s3.gaming-cdn.com/images/products/2141/orig/dragon-ball-fighterz-cover.jpg" />
```

Figura 12: Fragmento de código html donde actúa *screens*

```
<iframe id="ig-vimeo-player" src="https://player.vimeo.com/video/221403113" width="776" height="437" frameborder="0" webkitallowfullscreen="" mozallowfullscreen="" allowfullscreen="true">
```

Figura 13: Fragmento de código html donde actúa *video*

- Reglas:

```
%%

{game_name} {game = get_game_name(yytext);}
{spec_total} {specs = to_flat_text(yytext);}
{screens} {get_screens(yytext);}
{description} {description = to_flat_text(yytext);}
{price} {get_price(yytext);}
{video} {get_video(yytext);}
{tags} {get_tags(yytext);}
{release_date} {release_date = to_flat_text(yytext);}

. {}
\n {}

%%
```

Figura 14: Definiciones fichero lex

■ Código fuente:

```

int main(int argc, char *argv[]) {
    if (argc ==2) { // Se da fichero como entrada
        yyin= fopen(argv[1], "rt"); // Se abre fichero para lectura en modo texto
        if (!yyin) { // Error
            cout<< "No se pudo abrir el fichero "<< argv[1]<< endl;
            return 0;
        }
    } else { // No se da fichero como entrada: Se coge la entrada desde consola (entrada estandar)

        yyin= stdin;
    }

    yylex(); // Llamada al reconocedor
    ofstream ofs;
    char gname[50], buff[50], gfile[50];
    char* comando1 ="mv ";

    strcpy(gfile,game_directory);
    strcat(gfile,".txt");

    ofs.open (gfile, ofstream::out | ofstream::app);
    ofs << game << "\ttrailer:" + video << "\n\nGéneros: " << tags << "\n\n" << release_date << endl << specs << endl << description << endl << price << endl;
    ofs.close();

    strcpy(buff,comando1);
    strcat(buff,gfile);
    strcat(buff," ");
    strcat(buff,game_directory);

    int get = system(buff);

    return 0;
}

```

Figura 15: Método main()

```

string to_flat_text(char* text) {
    string tmp(text), flat_text;
    bool label = false;

    for(string::iterator it=tmp.begin(); it!=tmp.end(); ++it) {
        if((*it) == '<') label = true;
        else if ((*it) == '>') label = false;
        else if(!label) flat_text.push_back(*it);
    }
    return flat_text;
}

```

Figura 16: Método to_flat_text()

```
string get_game_name(char* name) {
    string tmp(name), gname, gname_command;
    bool label = false;
    char buff[500], buff2[500];
    char* command ="mkdir ";
    char* comando1 ="cd ";
    char* comando2 =" && wget ";

    for(string::iterator it=tmp.begin(); it!=tmp.end(); ++it) {
        if((*it) == '<') label = true;
        else if ((*it) == '>') label = false;
        else if(!label) {
            if((*it) != '(' && (*it) != ')') gname.push_back(*it);
            if(*it == '_') gname_command.push_back('_');
            else if((*it) != '(' && (*it) != ')') gname_command.push_back(*it);
        }
    }

    strcpy(buff,command);
    strcat(buff,gname_command.c_str());
    strcpy(game_directory,gname_command.c_str());
    int get=system(buff);

    strcpy(buff2,comando1);
    strcat(buff2,game_directory);
    strcat(buff2,comando2);
    strcat(buff2,screen.c_str());
    int get2=system(buff2);

    return gname;
}
```

Figura 17: Método get_name()

```
void get_screens(char* text) {
    bool start = false;
    char buff[500];
    char * comando1 ="cd ";
    char * comando2 =" && wget ";
    string tmp(text);

    for(string::iterator it=tmp.begin(); it!=tmp.end(); ++it) {
        if((*it) == 'h') {
            start = true;
            screen.push_back(*it);
        }else if (start && (*it) == '') start = false;
            else if(start) screen.push_back(*it);
    }

}
```

Figura 18: Método get_screens()

```
void get_price(char* text) {
    string tmp(text);
    bool label = false;
    int case_= 0;

    for(string::iterator it=tmp.begin(); it!=tmp.end(); ++it) {
        if((*it) == '<') label = true;
        else if ((*it) == '>') label = false;
        else if(!label)
            if(!label) { if((*it) == '\n') price += " "; else price.push_back(*it);}
    }
}
```

Figura 19: Método get_price()

```

void get_video(char* text) {
    string tmp(text);
    bool url = false, end = false;

    for(string::iterator it=tmp.begin(); it!=tmp.end() && !end; ++it) {
        if((*it) == 'h') url = true;

        if(url && ((*it) == '\"')) {url = false;end=true;}

        if(url) video.push_back(*it);
    }
}

```

Figura 20: Método get_video()

```

void get_tags(char* text) {
    string tmp(text);
    bool label = false, insert=false;
    int case_=0;

    for(string::iterator it=tmp.begin(); it!=tmp.end(); ++it) {
        if((*it) == '<') label = true;
        else if ((*it) == '>') label = false;
        else if(!label) {
            switch (case_) {
                case 0: if((*it) == '\n') case_ = 1;
                          else tags.push_back(*it);
                          break;
                case 1: if((*it) == '\n' || (*it) == ' ') tags+=" ";
                          else tags.push_back(*it);
                          break;
            }
        }
        tags.pop_back();tags.pop_back();
    }
}

```

Figura 21: Método get_tags()

3. Ejemplo de ejecucion

- Ejecutamos en la terminal un ejemplo de la carpeta *html_input*:

```
luts@luts-pc:~/Documentos/3/1/mc/practicas/lex/entrega$ ./program html_input/dbfz.html
--2018-12-09 11:49:55-- https://s3.gaming-cdn.com/images/products/2141/orig/dragon-ball-fighterz-cover.jpg
Resolviendo s3.gaming-cdn.com (s3.gaming-cdn.com)... 104.25.187.25, 104.25.180.25
Conectando con s3.gaming-cdn.com (s3.gaming-cdn.com)[104.25.187.25]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 507695 (554K) [image/jpeg]
Guardando como: "dragon-ball-fighterz-cover.jpg"

dragon-ball-fighterz-cover.jpg          100%[=====] 554,39K   738KB/s  en 0,8s

2018-12-09 11:49:56 (738 KB/s) - "dragon-ball-fighterz-cover.jpg" guardado [567699/567699]

luts@luts-pc:~/Documentos/3/1/mc/practicas/lex/entrega$
```

Figura 22: Terminal tras ejecutar un ejemplo

- Estado de la carpeta tras ejecutar:

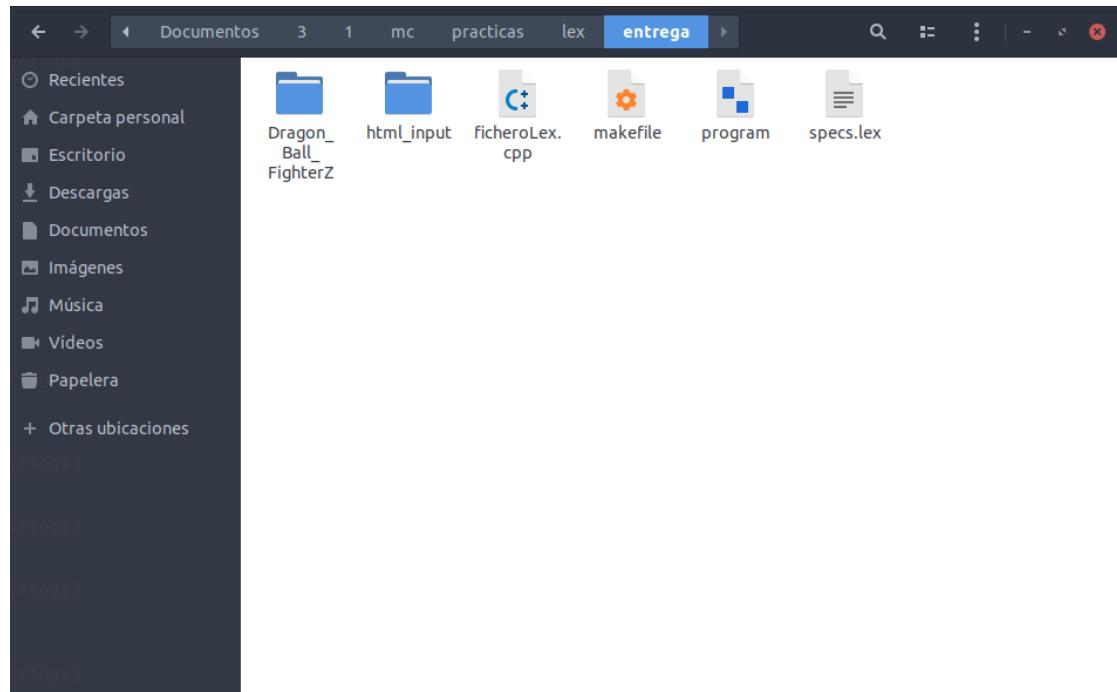


Figura 23: Carpeta creada

- Contenido de la carpeta:

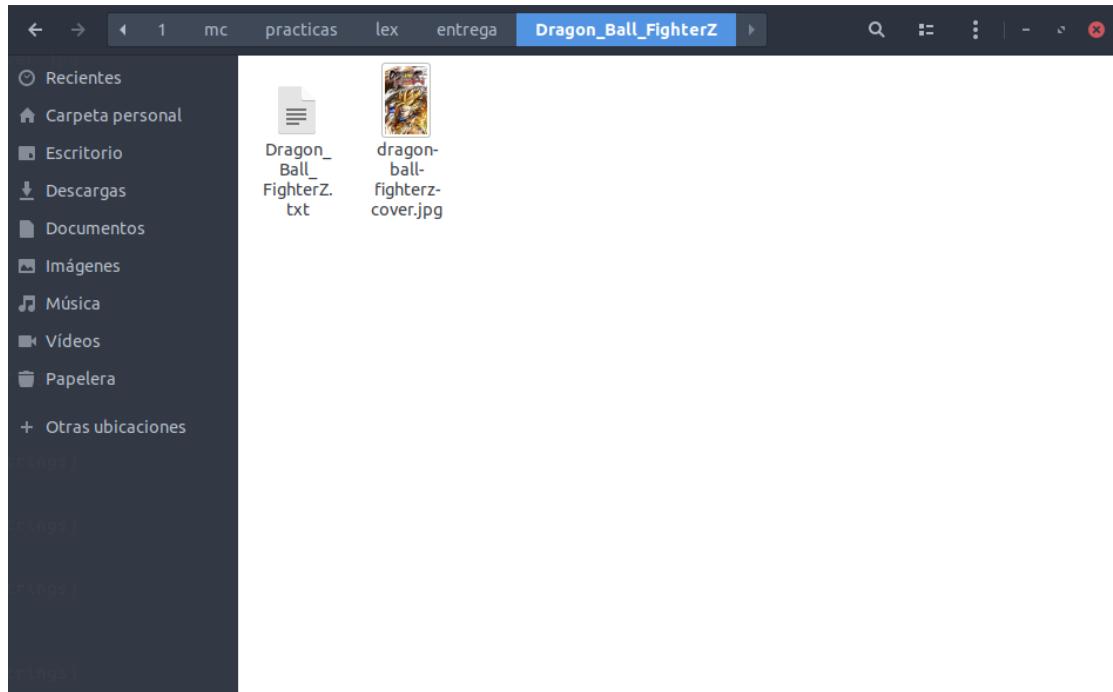


Figura 24: Ficheros obtenidos



Figura 25: Portada del videojuego

3 EJEMPLO DE EJECUCION

The screenshot shows a Microsoft Word document with the following content:

Dragon Ball FighterZ trailer:<https://player.vimeo.com/video/221403113>

Géneros: Acción, Arcade, Beat'em, all, Lucha, Multijugador, Cooperación, Anime, 2D, Fighter, Competitive, Local, Multiplayer, 2.5D, Controller, Singleplayer, Story, Rich, Difficult

Fecha de lanzamiento
26 enero 2018

Configuración mínima*

OS: Windows 7/8/10 (64-bit OS required)
Processor: AMD FX-4350, 4.2 GHz / Intel Core i5-3470, 3.20 GHz
Memory: 4 GB RAM
Graphics: Radeon HD 6870, 1 GB / GeForce GTX 650 Ti, 1 GB
DirectX: Version 11
Network: Broadband Internet connection
Sound Card: DirectX compatible soundcard or onboard chipset

Configuración recomendada*

OS: Windows 7/8/10 (64-bit OS required)
Processor: AMD Ryzen 5 1400, 3.2 GHz / Intel Core i7-3770, 3.40 GHz
Memory: 8 GB RAM
Graphics: Radeon HD 7870, 2 GB / GeForce GTX 660, 2 GB
DirectX: Version 11
Network: Broadband Internet connection
Sound Card: DirectX compatible soundcard or onboard chipset

DRAGON BALL FighterZ nace de lo que hace a DRAGON BALL tan famosa: luchas espectaculares e interminables con luchadores superpoderosos.

Con Arc System Works, DRAGON BALL FighterZ maximiza los gráficos de anime de alta calidad y ofrece un juego de lucha fácil de aprender pero difícil de dominar.

Gráficos de anime de alta calidad.
Con el poder del motor Unreal y el talentoso equipo de Arc System Works, DRAGON BALL FighterZ es toda una hazaña visual.

En equipo/sopporte 3vs3
Forma tu equipo soñado y domina rápidas combinaciones entre luchadores.

Emocionante juego en linea
Partidas igualadas, salas interactivas, locas partidas de grupos de 6 jugadores... ¡Hay para todos los gustos!

Modo historia exclusivo
Descubre una aventura nunca vista con la Androide N°21, un nuevo personaje de creación supervisada por Akira Toriyama.

Luchas espectaculares
¡Experimenta combos aéreos, escenarios destructibles y escenas famosas del anime de DRAGON BALL reproducidas en una resolución de 60FPS y 1080p!

Precio de venta al público: 60€ -69% 18.77€

Figura 26: Información obtenida