The God We Trust

|  |
| --- |
|  |

8th Assignment Report

<محمدمهدی قلاوند> <۹۸۲۲۲۱۲۱>

**Introduction:**

در این تمرین وظیفه ایجاد یک برنامه جاوا را داریم که برای مدیریت و دانلود داده های بازی های ویدیویی می باشد. این برنامه از اجزای جداگانه سرور و کلاینت تشکیل شده است که با استفاده از برنامه نویسی سوکت به یکدیگر متصل شده اند.

**Design and Implementation:**

این پروژه به طریقی طراحی شده است که سمت کلاینت و سرور بتوانند به روشی از پیش تعیین شده با یکدیگر ارتباط برقرار کنند برای دستیابی به این هدف می توان یک سری تعاملات Request و Response بین کلاینت و سرور پیاده سازی کرد

یک درخواست از طریق شبکه از مشتری به سرور ارسال می شود یک مشتری می تواند فهرستی از بازی های موجود اطلاعات مربوط به یک بازی خاص یا دانلود یک بازی را درخواست کند.ایجاد حساب کاربری ورود به سیستم و خروج از سیستم نیز می تواند درخواست در نظر گرفته شود

یک پاسخ از سرور به مشتری از طریق شبکه ارسال میشود ما باید انواع مختلف پاسخ مربوط به درخواست دریافتی را ایجاد می کنیم داده های مناسب را بر اساس درخواست به پاسخ پیوست می کنیم یک پاسخ ممکن است حاوی داده هایی باشد که کاربر درخواست کرده است

**Testing and Evaluation:**

* طریقه پیاده سازی کپی فایل از یک پوشه به پوشه دیگر
* public static void copyFolder(File src, File dest,String fileName)  
   throws IOException{  
    
   if(src.isDirectory()){  
    
   //if directory not exists, create it  
   if(!dest.exists()){  
   dest.mkdir();  
   System.*out*.println("Directory copied from "  
   + src + " to " + dest);  
   }  
    
   //list all the directory contents  
   List<String> files = Arrays.*stream*(src.list()).filter(c->c.equals(fileName)).collect(Collectors.*toList*());  
    
   for (String file : files) {  
   //construct the src and dest file structure  
   File srcFile = new File(src, file);  
   File destFile = new File(dest, file);  
   //recursive copy  
   *copyFolder*(srcFile,destFile,fileName);  
   }  
    
   }else{  
   //if file, then copy it  
   //Use bytes stream to support all file types  
   InputStream in = new FileInputStream(src);  
   OutputStream out = new FileOutputStream(dest);  
    
   byte[] buffer = new byte[1024];  
    
   int length;  
   //copy the file content in bytes  
   while ((length = in.read(buffer)) > 0){  
   out.write(buffer, 0, length);  
   }  
    
   in.close();  
   out.close();  
   System.*out*.println("File copied from " + src + " to " + dest);  
   }  
  }

**Conclusion:**