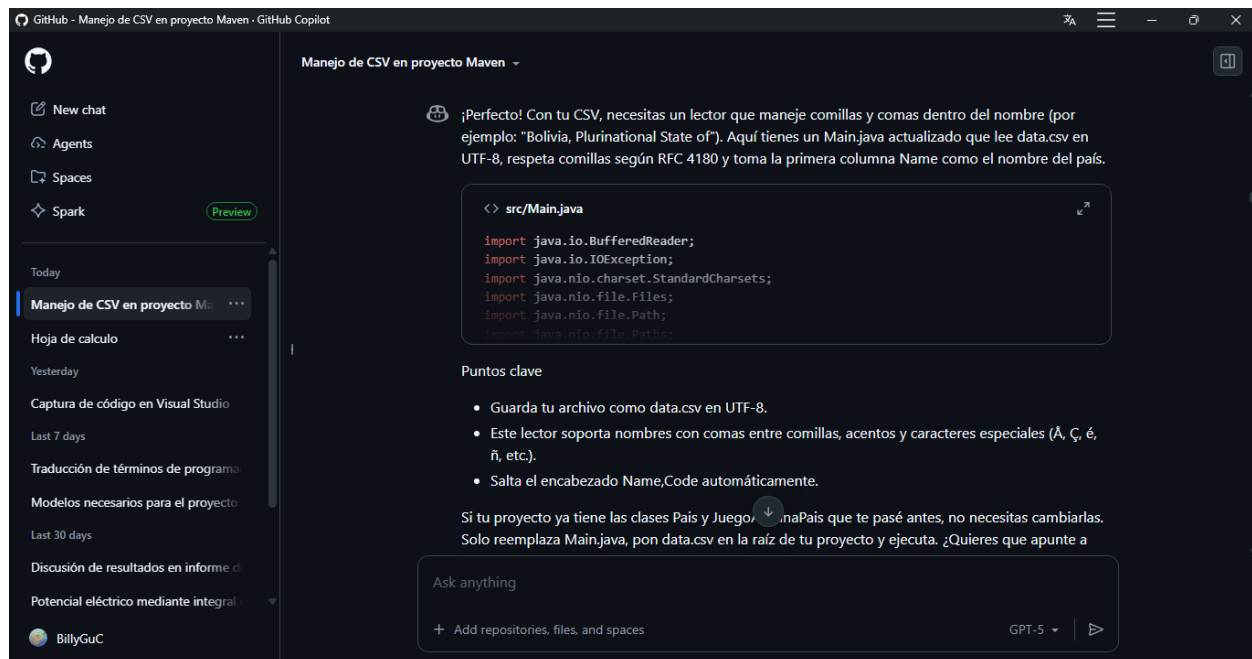
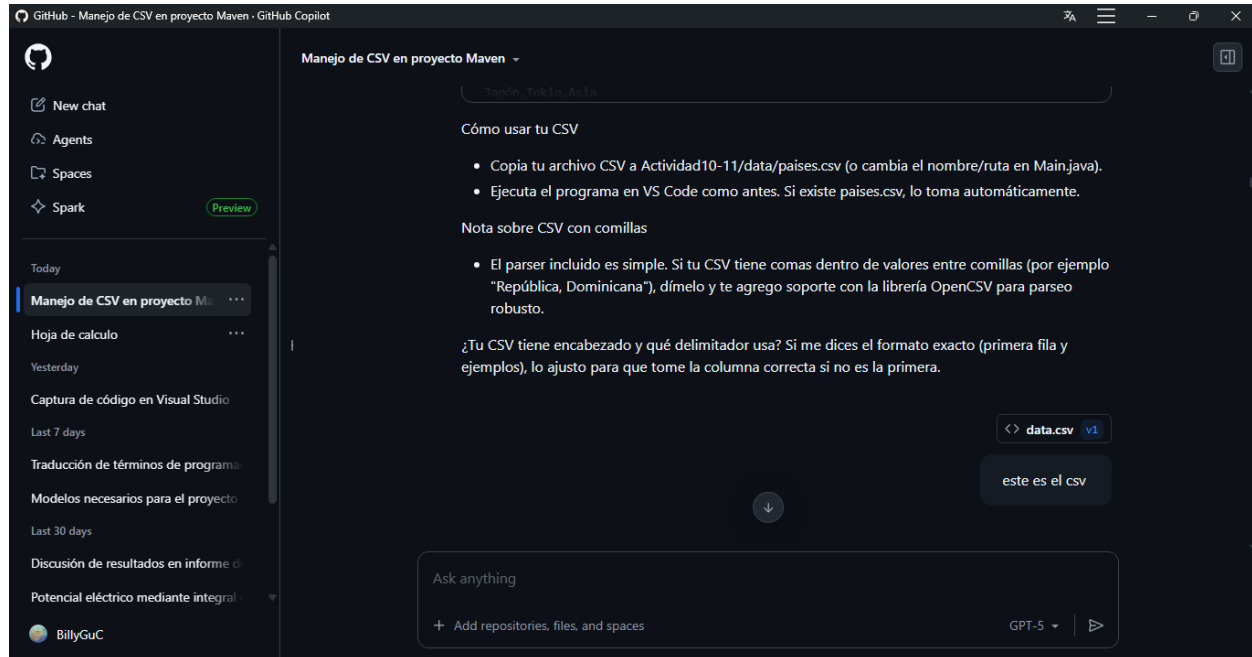
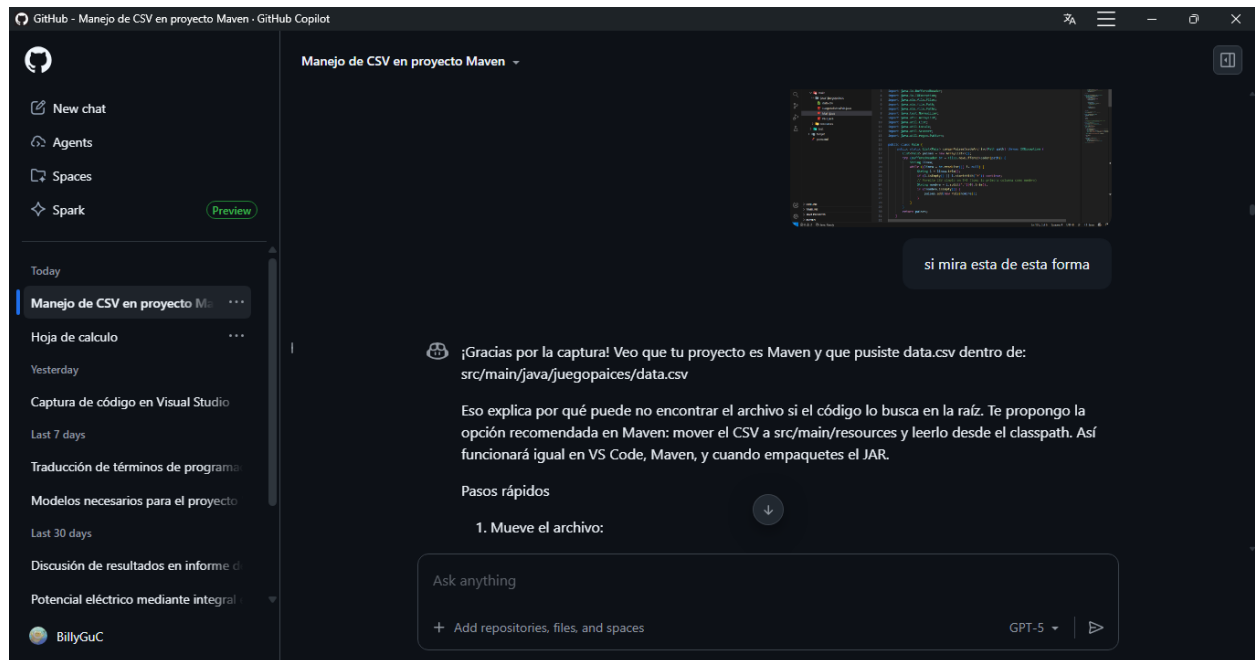
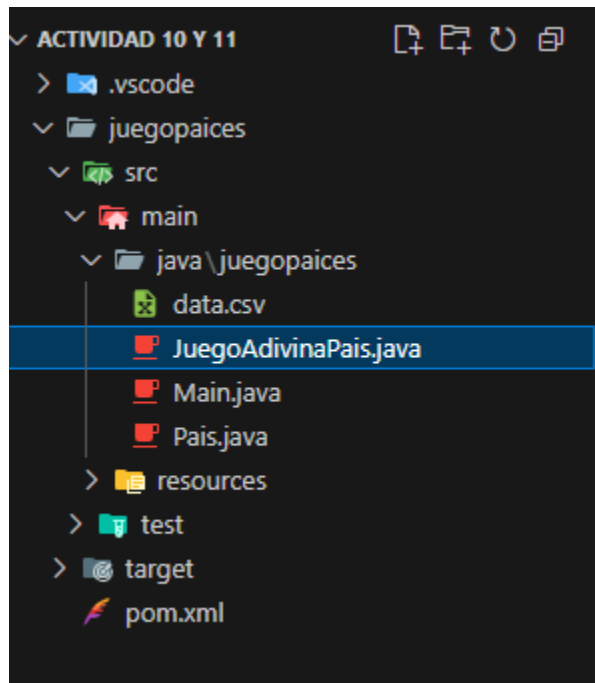


Capturas de las preguntas hacia la GitHub Copilot





Estructura del proyecto



Métodos importantes



```
1 private static List<Pais> parsearPaisesDesdeCSV(BufferedReader br) throws IOException {
2     List<Pais> paises = new ArrayList<>();
3     String linea;
4     boolean primeraLinea = true;
5
6     while ((linea = br.readLine()) != null) {
7         if (linea.isEmpty()) continue;
8
9         // Quitar BOM si aparece al inicio
10        if (primeraLinea && linea.length() > 0 && linea.charAt(0) == '\uFEFF') {
11            linea = linea.substring(1);
12        }
13        primeraLinea = false;
14
15        // No hacemos trim global para no romper espacios dentro de comillas
16        String l = linea;
17        if (l.isEmpty() || l.startsWith("#")) continue;
18
19        List<String> cols = parseCsvLine(l, ',');
20        if (cols.isEmpty()) continue;
21
22        // Saltar encabezado "Name,Code"
23        if (esEncabezado(cols.get(0))) continue;
24
25        String nombre = cols.get(0).trim();
26        String codigo = cols.size() > 1 ? cols.get(1).trim() : "";
27
28        if (!nombre.isEmpty()) {
29            paises.add(new Pais(nombre, codigo));
30        }
31    }
32    return paises;
33 }
34
```



```
1 private static List<String> parseCsvLine(String line, char delimiter) {
2     List<String> out = new ArrayList<>();
3     StringBuilder field = new StringBuilder();
4     boolean inQuotes = false;
5
6     for (int i = 0; i < line.length(); i++) {
7         char c = line.charAt(i);
8
9         if (inQuotes) {
10             if (c == '"') {
11                 // Comilla escapada ""
12                 if (i + 1 < line.length() && line.charAt(i + 1) == '"') {
13                     field.append('"');
14                     i++; // consumir comilla escapada
15                 } else {
16                     inQuotes = false; // cierre de comillas
17                 }
18             } else {
19                 field.append(c);
20             }
21         } else {
22             if (c == '"') {
23                 inQuotes = true;
24             } else if (c == delimiter) {
25                 out.add(field.toString());
26                 field.setLength(0);
27             } else {
28                 field.append(c);
29             }
30         }
31     }
32     // último campo
33     out.add(field.toString());
34     return out;
35 }
```

```

1 private static Path findCsv(String fileName) {
2     Path wd = Paths.get("").toAbsolutePath();
3     Path[] candidates = new Path[] {
4         // Ejecutando desde el módulo "juegopaices"
5         Paths.get("src", "main", "java", "juegopaices", fileName),
6         // Ejecutando desde la raíz del workspace (una carpeta arriba del módulo)
7         Paths.get("juegopaices", "src", "main", "java", "juegopaices", fileName),
8         // Por si ejecutas desde la carpeta del paquete
9         Paths.get(fileName)
10    };
11    for (Path p : candidates) {
12        if (Files.exists(p)) return p;
13    }
14    throw new IllegalStateException(
15        "No se encontró el archivo CSV. Directorio actual: " + wd +
16        "\nProbé estas rutas relativas y no existen:\n - " + candidates[0].toString() +
17        "\n - " + candidates[1].toString() +
18        "\n - " + candidates[2].toString() +
19        "\nMueve el archivo a una de esas rutas o ajusta el código a tu ubicación real."
20    );
21 }

```

```

1 public void jugar(Scanner scanner) {
2     Pais elegido = elegirPaisAleatorio();
3     String primera = primeraLetraVisible(elegido.getNombre());
4     System.out.println("El país que elegí empieza con " + primera);
5     System.out.print("¿Cuál crees que es? -> ");
6     String intento = scanner.nextLine();
7
8     if (igualesSinAcentosNiMayusculas(intento, elegido.getNombre())) {
9         System.out.println("¡Correcto! Era: " + elegido.getNombre() + " (" + elegido.getCodigo() + ")");
10    } else {
11        System.out.println("No es correcto. El país era: " + elegido.getNombre() + " (" + elegido.getCodigo() + ")");
12    }
13 }

```

```

1  private String primeraLetraVisible(String nombre) {
2      if (nombre == null || nombre.isBlank()) return "?";
3      for (int i = 0; i < nombre.length(); i++) {
4          char c = nombre.charAt(i);
5          if (Character.isLetter(c)) {
6              return String.valueOf(c).toUpperCase(Locale.ROOT);
7          }
8      }
9      return String.valueOf(nombre.charAt(0)).toUpperCase(Locale.ROOT);
10 }

```

Captura de Juego

The screenshot shows the VS Code interface with the following details:

- Explorer Panel:** Shows the project structure. The 'juegopaices' folder is selected, containing 'data.csv', 'Main.java', 'Pais.java', 'resources', 'test', and 'target'.
- Source Explorer:** Shows the 'Main.java' file, which contains the code for the game.
- Output Panel:** Shows the execution of 'juegoAdivinaPais.java'. The output includes:


```

PS C:\Users\Billy\Documents\IPC1_Actividades_202200182\Actividad 10 y 11> & 'C:\Users\Billy\.vscode\extensions\redhat.java-1.46.0-win32-x64\jre\21.0.8-win32-x86_64\bin\java.exe' '-Xc:sShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Billy\Documents\IPC1_Actividades_202200182\Actividad 10 y 11\juegopaices\target\Classes' 'juegopaices.Main'
Se cargaron 249 países desde C:\Users\Billy\Documents\IPC1_Actividades_202200182\Actividad 10 y 11\juegopaices\src\main\java\juegopaices\data.csv
¡Bienvenido/a al juego de adivinar países!
El país que elegí empieza con C
¿Cuál crees que es? -> colombia
No es correcto. El país era: Curaçao (CW)
¿Quieres jugar otra vez? (s/n): n
Gracias por jugar.
PS C:\Users\Billy\Documents\IPC1_Actividades_202200182\Actividad 10 y 11>

```
- Terminal Panel:** Shows the command to run the program and the resulting output.