

Trabajo: Creación de una extensión para Visual Studio Code

Acá va un párrafo que describa lo que es el proyecto

Objetivo

Practicar JavaScript y de paso conocer TypeScript y aplicarlo de una forma novedosa y de una forma usada hoy en día.

Descripcion

Visual Studio Code se esta convirtiendo en un editor de codigo muy bueno gracias a su cantidad de **Extenciones**. En este caso se aplicara la creacion de una extencion usando TypeScript y algunas Librerias de NodeJS.

Pre-requisitos

Que cosas necesitas para instalar el software y como instalarlas.

- **Visual Studio Code** // Es el editor de codigo del cual anteriormente se hablo.
- **NodeJS** // Se usara para descargar paquetes de sus repositorios como lo son:
 - **Yeoman** : Es una aplicacion para generar el codigo del esqueleto inicial.
 - **generator-code** : Es uno de los generadores para Yeoman y nos servira para comenzar el desarrollo de la extencion para VS Code.

Instalación & Iniciación

Podemos Instalar todo esto desde la terminal:

```
npm install -g yo generator-code typescript
```

Una vez instalado ejecutar en terminal:

```
yo code
```

Aquí Seleccionamos New Extension (TypeScript).

```
PS C:\Users\TI> yo code

Welcome to the Visual
Studio Code Extension
generator!

? What type of extension do you want to create? (Use arrow keys)
> New Extension (TypeScript)
  New Extension (JavaScript)
  New Color Theme
  New Language Support
  New Code Snippets
  New Keymap
  New Extension Pack
  New Language Pack (Localization)
  New Web Extension (TypeScript)
  New Notebook Renderer (TypeScript)
```

Luego Agregar nombre,

nombre de extension, descripcion, etc...

```
? What type of extension do you want to create? New Extension (TypeScript)
? What's the name of your extension? Line Gapper2
? What's the identifier of your extension? gapeline2
? What's the description of your extension? Esta es una repetición de line gapper
? Initialize a git repository? No
? Bundle the source code with webpack? No
? Which package manager to use? npm

Writing in C:\Users\TI\gapeline2...
  create gapeline2\.vscode\extensions.json
  create gapeline2\.vscode\launch.json
  create gapeline2\.vscode\settings.json
  create gapeline2\.vscode\tasks.json
  create gapeline2\package.json
  create gapeline2\tsconfig.json
  create gapeline2\.vscodeignore
  create gapeline2\vsc-extension-quickstart.md
  create gapeline2\README.md
  create gapeline2\CHANGELOG.md
  create gapeline2\src\extension.ts
  create gapeline2\src\test\runTest.ts
  create gapeline2\src\test\suite\extension.test.ts
  create gapeline2\src\test\suite\index.ts
  create gapeline2\.eslintrc.json

Changes to package.json were detected.

Running npm install for you to install the required dependencies.
[ ] \ reify:estraverse: timing
```

Y

por ultimo se nos abraira nuestro VS Code con el proyecto de extención.

```
Writing in C:\Users\TI\gapeline2...
  create gapeline2\.vscode\extensions.json
  create gapeline2\.vscode\launch.json
  create gapeline2\.vscode\settings.json
  create gapeline2\.vscode\tasks.json
  create gapeline2\package.json
  create gapeline2\tsconfig.json
  create gapeline2\.vscodeignore
  create gapeline2\vsc-extension-quickstart.md
  create gapeline2\README.md
  create gapeline2\CHANGELOG.md
  create gapeline2\src\extension.ts
  create gapeline2\src\test\runTest.ts
  create gapeline2\src\test\suite\extension.test.ts
  create gapeline2\src\test\suite\index.ts
  create gapeline2\.eslintrc.json

Changes to package.json were detected.

Running npm install for you to install the required dependencies.

added 210 packages, and audited 211 packages in 36s

40 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

Your extension gapeline2 has been created!

To start editing with Visual Studio Code, use the following commands:

    code gapeline2

Open vsc-extension-quickstart.md inside the new extension for further instructions
on how to modify, test and publish your extension.

For more information, also visit http://code.visualstudio.com and follow us @code.

? Do you want to open the new folder with Visual Studio Code?
  Open with `code`
> Skip
```

Editando el codigo

Nuestro Codigo a Modificar(Codigo de ejemplo) lo encontraremos en la carpeta de **src/extencion.ts**.

por otro lado las propiedades de la extencion las modificaremos en el archivo **package.json** en la raiz del proyecto.

En el archivo **package.json** se editara lo siguiente

```
{
  "name": "gapline", //Aqui se encuentra el nombre de la Extencion
  "displayName": "Line Gapper", //Este es el nombre del proyecto
  "description": "Esta extencion es un ejemplo de la asignatura de clinte
servider en la tarea de crear una extencion para VSCode", //Aqui se encuentra la
descripcion del proyecto
  "version": "0.0.1",
  "engines": {
```

```

    "vscode": "^1.67.0"
  },
  "categories": [
    "Other"
  ],
  "activationEvents": [
    "onCommand:extension.gapline" // aqui llamaremos a la extencion para que
    no cargue tanto en memoria solo cuando se ocupe
  ],
  "main": "./out/extension.js",
  "contributes": {
    "commands": [{ //Aqui se llamaran a los comandos y el nombre de cada uno.
      "command": "extension.gapline", //Con este usaremos la funcion
      "title": "Gapeline" //Asi es como lo encontraremos en la consola de
      proyectos
    }]
  },
  "scripts": {
    "vscode:prepublish": "npm run compile",
    "compile": "tsc -p ./",
    "watch": "tsc -watch -p ./",
    "pretest": "npm run compile && npm run lint",
    "lint": "eslint src --ext ts",
    "test": "node ./out/test/runTest.js"
  },
  "devDependencies": {
    "@types/vscode": "^1.67.0",
    "@types/glob": "^7.2.0",
    "@types/mocha": "^9.1.1",
    "@types/node": "14.x",
    "@typescript-eslint/eslint-plugin": "^5.21.0",
    "@typescript-eslint/parser": "^5.21.0",
    "eslint": "^8.14.0",
    "glob": "^8.0.1",
    "mocha": "^9.2.2",
    "typescript": "^4.6.4",
    "@vscode/test-electron": "^2.1.3"
  }
}

```

Por otro lado esta es la Extencion Gapper Line.

```

'use strict'; // Invoca al modo estricto
import * as vscode from 'vscode'; //Importa el modulo de VSCODE
export function activate(context: vscode.ExtensionContext) { //Crea la Funcion
  para la extencion y la retorna publica para otras api
    let disposable = vscode.commands.registerCommand('extension.gapline', () => {
    //Crea el comando para ejecutar
      var editor = vscode.window.activeTextEditor; //esta guardando el editor de
      texto y lo guarda en la variable editor
      if (!editor) { return; } //Valida la funcion editor
      var selection = editor.selection; //Toma todo lo que esta presente en el
    }
  }
}

```

GUI

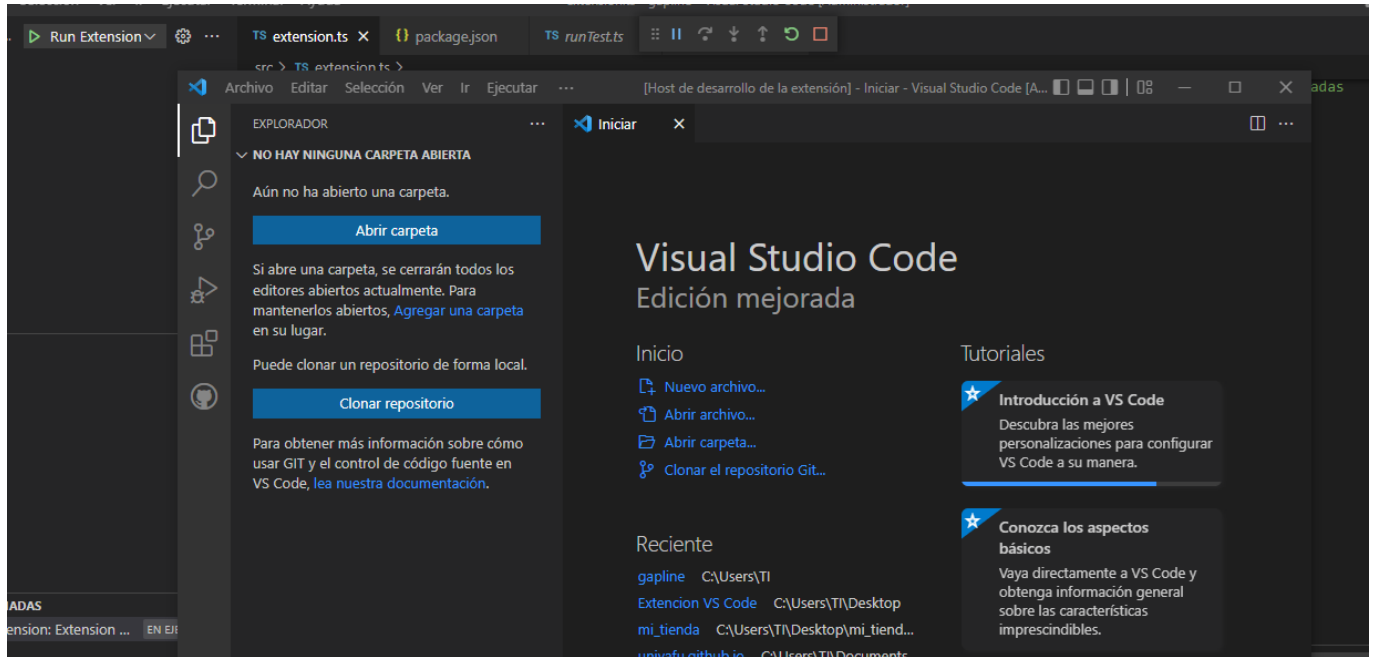
```

    var text = editor.document.getText(selection); //Toma el texto
    seleccionado y lo almacena en la variable text
    vscode.window.showInputBox({ prompt: 'Lineas?' }).then(value => { //Abre
    un recuadro donde le pregunta al usuario que introduzca
        let numberOfLines = +value; //Toma el numero de lineas
        var textInChunks: Array<string> = []; //Crea una var para almacenar el
    arreglo
        text.split('\n').forEach((currentLine: string, lineIndex) => {
    //separa el texto con espacios
            textInChunks.push(currentLine); //Manda el texto a determinada
    posicion
            if ((lineIndex+1) % numberOfLines === 0) {textInChunks.push('');}
    //Valida el push de las lineas mandadas
        });
        text = textInChunks.join('\n'); // Agrega todos los elementos a la
    cadena
        editor.edit((editBuilder) => { // Asocia el editor de texto
            var range = new vscode.Range( //crea el rango del editor de codigo
                selection.start.line, 0, //de donde va empezar el rango
                selection.end.line, //donde termina el rango
                editor.document.lineAt(selection.end.line).text.length
    //alinea el texto
            );
            editBuilder.replace(range, text); //Centro el nuevo texto con el
    nuevo rango
        });
    });
    });
    context.subscriptions.push(disposable); //deside si el contenido se puede
    añadir
}
export function deactivate() { } //sirve para realizar una limpieza

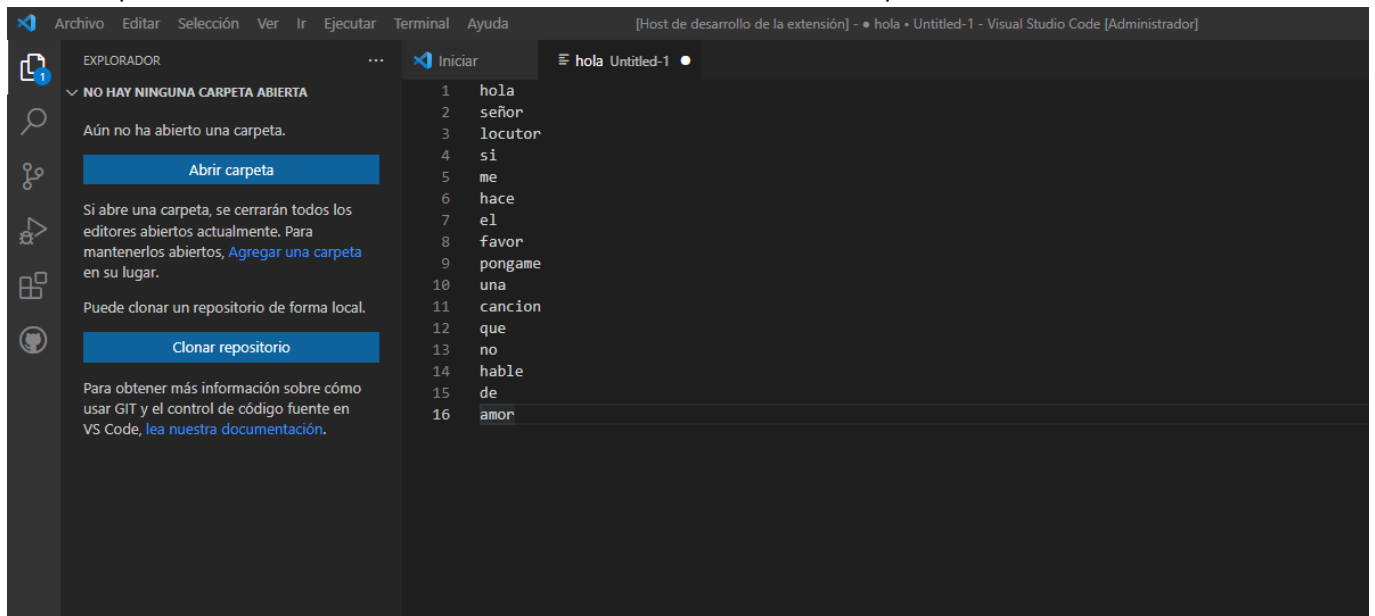
```

Ejecutando las pruebas 

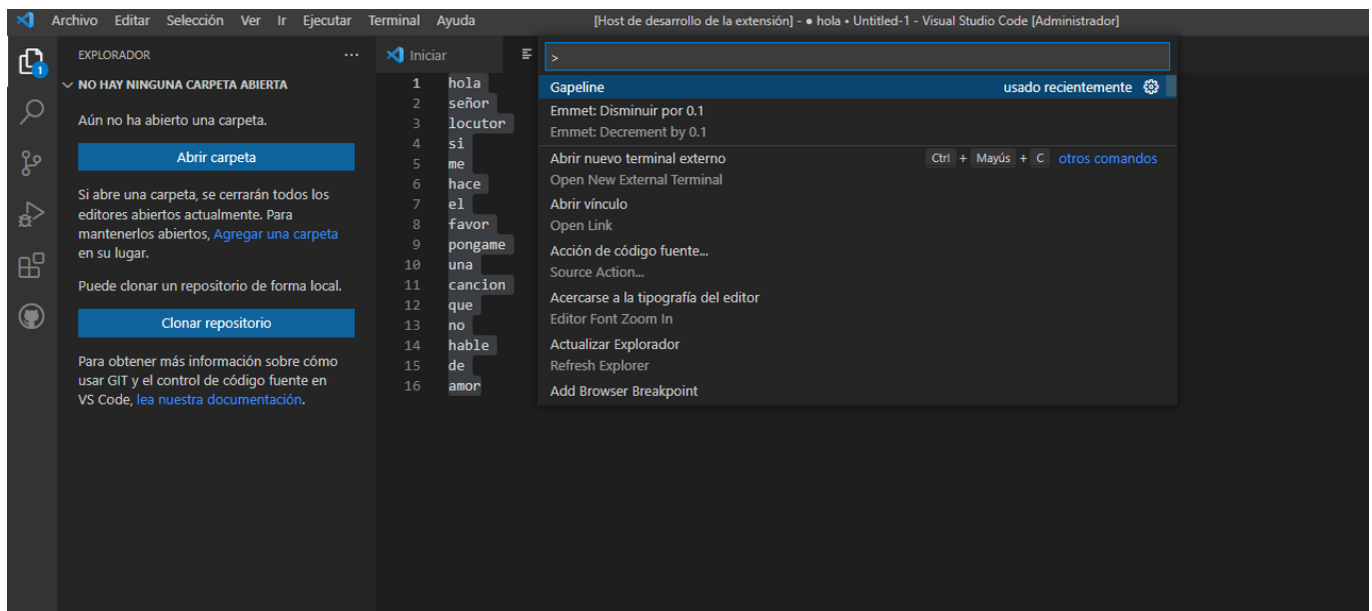
Para ejecutar la extensión es necesario precionar la Tecla **F5** el cual nos abra otra ventana de VS Code



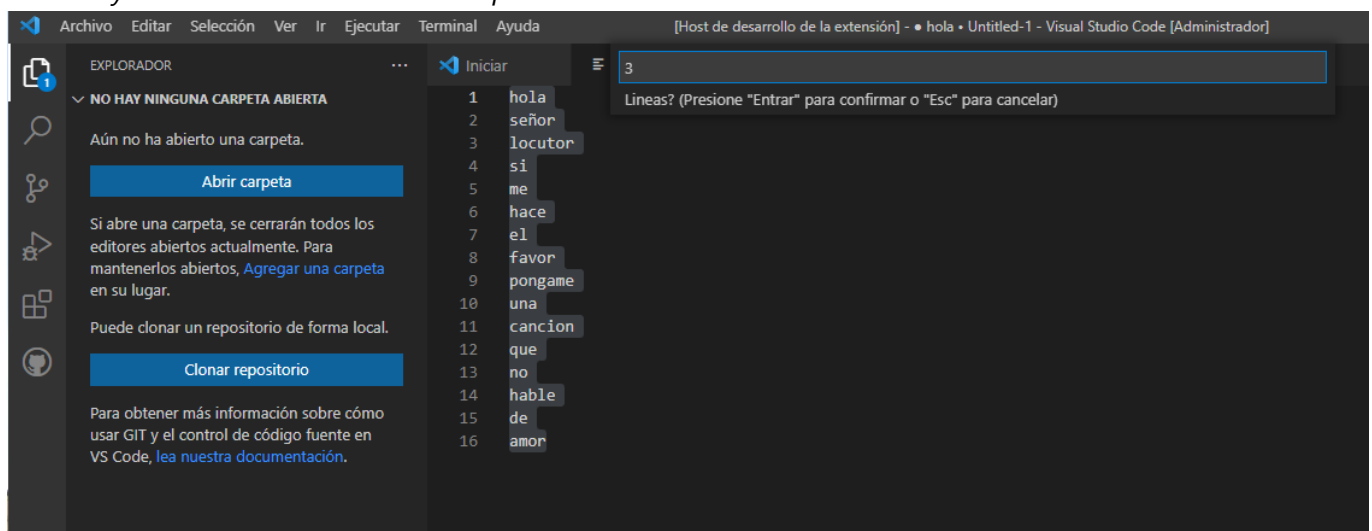
Una vez que se nos abrio la ventana nueva crear un documento de texto plano.



Despues precionar las teclas **Ctrl + Mayus + P** el cual nos abra un TextBox en el cual se pone el nombre de la extensión.

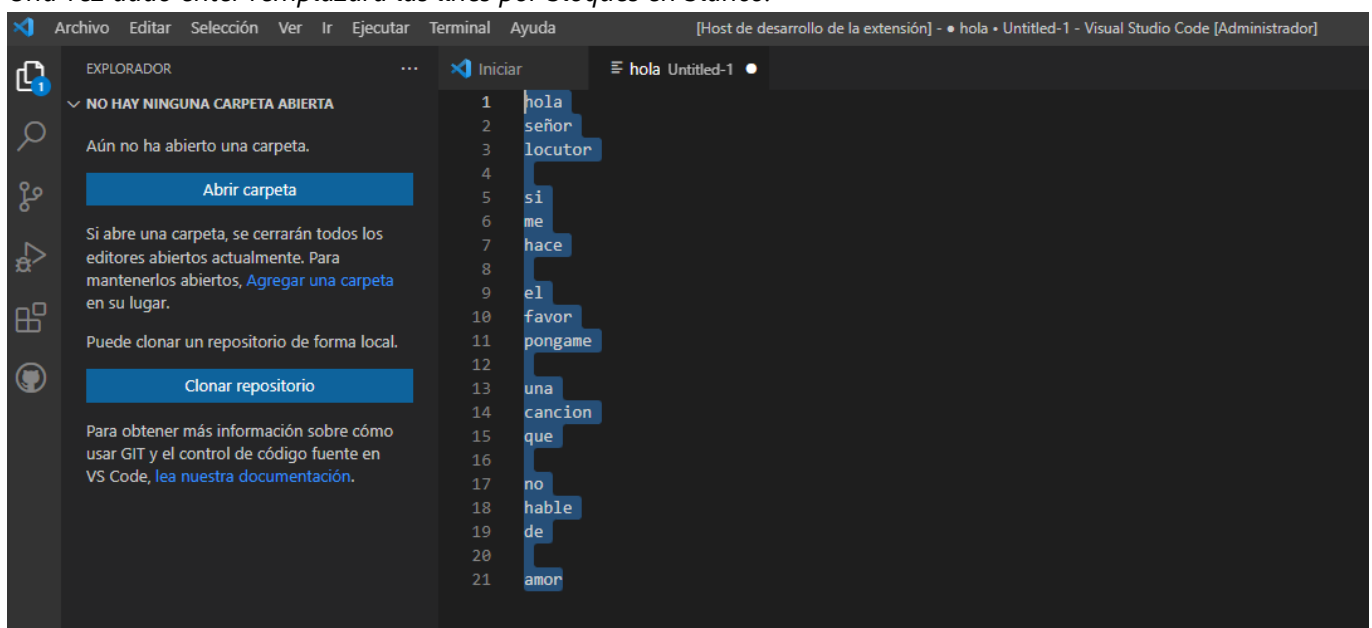


Una vez ya seleccionada la extensión te pedirá el número de líneas.



Y las pruebas de estilo de codificación 

Una vez dado enter remplazará las líneas por bloques en blanco.



Despliegue

*En caso de que el programa no funcione verificar el archivo **package.json** y verificar nombres y comandos esten de forma correcta y que correspondan con la funcion descrita en el TS*

Construido con

Menciona las herramientas que utilizaste para crear tu proyecto

- [Visual Studio Code](#) - El IDE Usado
- [Documentacion de Extenciones VS Code](#) - Encuentras la documentacion de extenciones
- [TypeScript](#) - Lenguaje basado en JavaScript de Microsoft
- [NodeJS](#) - Motor de JavaScript
- [Yeoman](#) - Aplicacion para generar el esqueleto de un proyecto web
- [Generator-Code](#) - Es uno de los generadores para Yeoman y sirve para comenzar el desarrollo de una extencion de VS Code

Licencia

Este practica fue proporcionada por la Universidad de la Rioja para la materia de Cliente Servidor.

 con  por [Luisgr95](#) 