Ajouter des configurations - VSCode

> Cette page va lister quelques configurations pouvant être utiles en cas d'utilisation de VSCode en tant qu'IDE (*Integrated Development Environment*)

VSCode

Il y a 3 fichiers de VSCode qui peuvent être modifiés/créés dans le dossier `.vscode` : `extensions.json`, `launch.json` et `settings.json`

extensions.json

```
{
  // For more information, visit: https://go.microsoft.com/fwlink/?linkid=827846
  "recommendations": [
    "angular.ng-template",
    "esbenp.prettier-vscode",
    "cyrilletuzi.angular-schematics",
    "formulahendry.auto-rename-tag",
    "zainchen.json",
    "davidanson.vscode-markdownlint",
    "pmneo.tsimporter",
    "orta.vscode-jest",
    "ryanluker.vscode-coverage-gutters",
    "pkief.material-icon-theme"
  ]
}
```

launch.json

```
{
  // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
  "version": "0.2.0",
```

```
  "configurations": [
    {
      "name": "ng serve",
      "type": "chrome",
      "request": "launch",
      "preLaunchTask": "npm: start",
      "url": "http://localhost:4200/"
    },
    {
      "name": "ng test",
      "type": "vscode-jest-tests",
      "request": "launch",
      "preLaunchTask": "npm: test"
    }
  ]
}
```

settings.json

```
{
  "javascript.preferences.importModuleSpecifier": "relative",
  "typescript.preferences.importModuleSpecifierEnding": "minimal",
  "typescript.preferences.importModuleSpecifier": "relative",
  "javascript.preferences.importModuleSpecifierEnding": "minimal",
  "js/ts.implicitProjectConfig.module": "ES2022",
  "editor.formatOnSave": true,
  "editor.codeActionsOnSave": {
    "source.organizeImports": "always"
  },
  "files.associations": {
    "*.prettierrc": "json"
  },
  "jest.useDashedArgs": true,
  "jest.coverageFormatter": "GutterFormatter",
  "jest.runMode": "on-demand",
  "coverage-gutters.showLineCoverage": true,
  "coverage-gutters.showGutterCoverage": false,
  "coverage-gutters.showRulerCoverage": true
}
```

Node

package.json

Il faut installer :

- `npm i @types/node`
- `npm i -D ng-mocks webpack-bundle-analyzer prettier`
- `ng add @angular-eslint/schematics`

Puis il faut ajouter, dans les scripts ;

```
"format": "prettier --write \"src/**/*.ts\"",
"ng": "ng",
"start": "ng serve",
"build": "ng build",
"build:prod": "ng build --configuration production",
"watch": "npm run build -- --watch --configuration development",
"test": "ng cache clean && ng test",
"build:stats": "npm run build -- --stats-json",
"build:stats:prod": "npm run build:prod -- --stats-json",
"analyze:bundle": "webpack-bundle-analyzer dist/project-name/stats.json",
"analyze": "npm run build:stats && npm run analyze:bundle",
"analyze:prod": "npm run build:stats:prod && npm run analyze:bundle",
"lint": "ng lint",
"lint:txt": "ng lint > lint-report.txt --silent",
"lint:html": "ng lint > lint-report.html --format html --silent",
"lint:json": "ng lint > lint-report.json --format json --silent"
```

Si jest est déjà installé, il faut remplacer la ligne

```
"test": "ng cache clean && ng test",
```

par les lignes

```
"test": "jest --verbose",
"test:watch": "npm test -- --watch",
"test:cov": "npm test -- --coverage --coverageReporters=lcov --coverageReporters=text --
coverageReporters=text-summary --coverageReporters=html",
"test:ci": "jest --ci",
"test:karma": "ng cache clean && ng test",
```

Prettier

Il faut, pour le configurer, placer un fichier `.prettierrc` à la racine du projet. Voilà un exemple de ce qu'il pourrait contenir ;

```
{
  "singleQuote": true,
  "arrowParens": "avoid",
  "proseWrap": "always",
  "htmlWhitespaceSensitivity": "ignore",
  "bracketSameLine": true,
  "printWidth": 120,
  "useTabs": false,
  "jsxSingleQuote": true,
  "trailingComma": "all",

  "singleAttributePerLine": false,
  "semi": true,
  "tabWidth": 2,
  "bracketSpacing": true,
  "embeddedLanguageFormatting": "auto",
  "endOfLine": "lf",
  "insertPragma": false,
  "quoteProps": "as-needed",
  "requirePragma": false,
  "vueIndentScriptAndStyle": false
}
```

ESLint

Il faut ajouter, dans le fichier `eslint.config.js`, la règle suivante ;

```
'@typescript-eslint/no-unused-vars': ['error', { argsIgnorePattern: '^_', varsIgnorePattern: '^_' }],
```

GitIgnore

Il faut ajouter également dans le fichier `.gitignore` le pattern suivant : `/lint-*.*`

Revision #5
Created 27 January 2025 18:43:39 by DarkMat

Updated 27 January 2025 19:23:05 by DarkMat