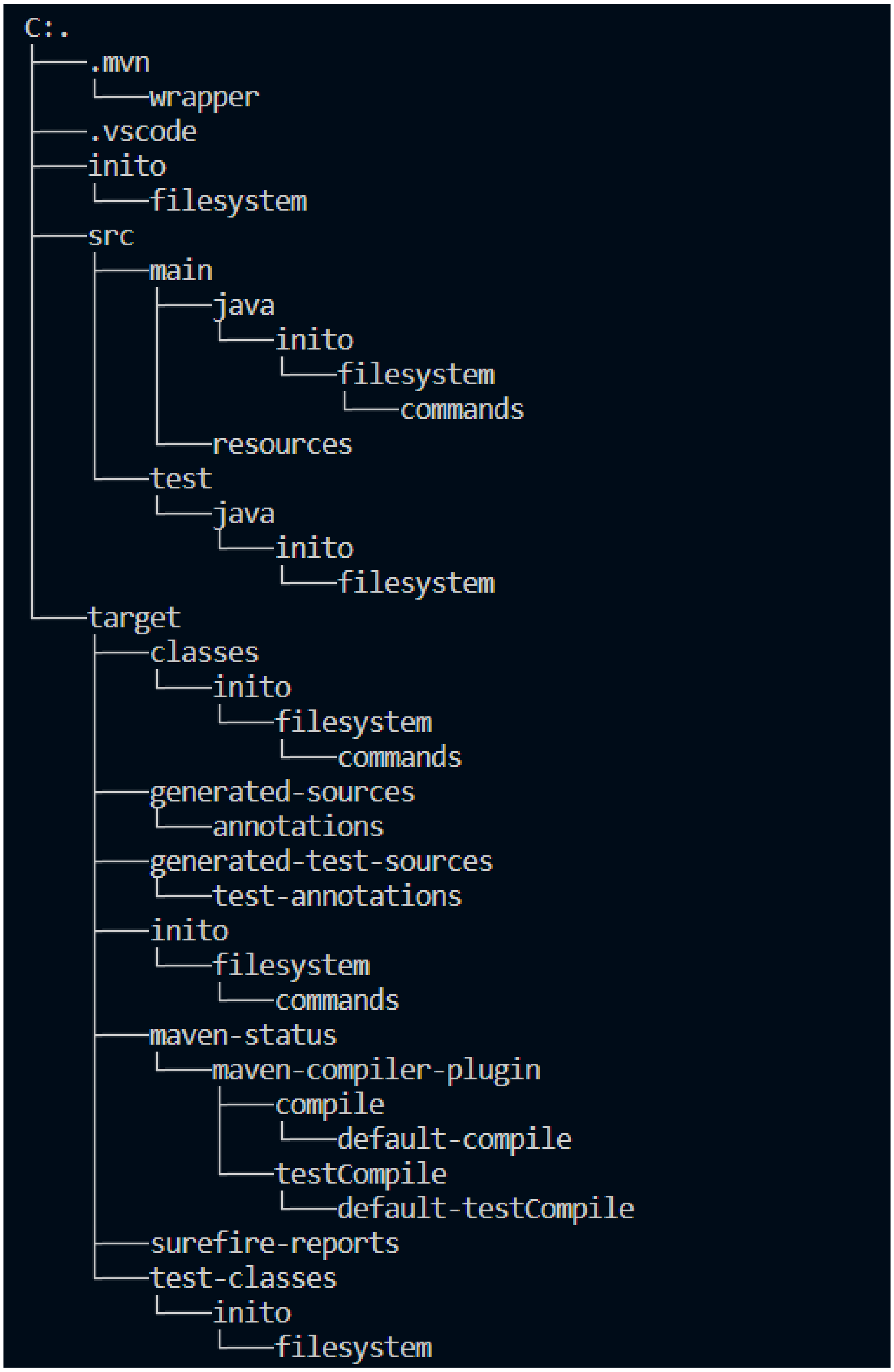
In-Memory File System (IMFS) Documentation

Overview

The In-Memory File System (IMFS) is implemented as a command-line interface (CLI) application with a focus on simplicity and efficiency. The system is built using Java and Maven for project management. The primary data structures include in-memory representations of directories and files.

Table of Contents

Project Structure



```
C:.
├───.mvn
│   └───wrapper
├───.vscode
├───inito
│   └───filesystem
├───src
│   ├───main
│   │   ├───java
│   │   │   └───inito
│   │   │       └───filesystem
│   │   │           └───commands
│   │   └───resources
│   └───test
│       └───java
│           └───inito
│               └───filesystem
└───target
    ├───classes
    │   └───inito
    │       └───filesystem
    │           └───commands
    ├───generated-sources
    │   └───annotations
    ├───generated-test-sources
    │   └───test-annotations
    ├───inito
    │   └───filesystem
    │       └───commands
    ├───maven-status
    │   └───maven-compiler-plugin
    │       ├───compile
    │       │   └───default-compile
    │       └───testCompile
    │           └───default-testCompile
    ├───surefire-reports
    └───test-classes
        └───inito
            └───filesystem
```

Features

The IMFS supports the following file system operations:

1. mkdir: Create a new directory.
2. cd: Change the current directory.
3. ls: List the contents of the current or specified directory.
4. grep: Search for a pattern in a file (bonus feature).
5. cat: Display the contents of a file.
6. touch: Create a new empty file.
7. echo: Write text to a file.
8. mv: Move a file or directory to another location.
9. cp: Copy a file or directory to another location.
10. rm: Remove a file or directory.

Setup and Installation

To set up the project, follow these steps:

Clone the Repository:

```
git clone https://github.com/your-username/inmemoryfilesystem.git

cd inmemoryfilesystem
```

Compile Java Source Files:

```
javac -cp src/main/java -d target src/main/java/inito/filesystem/*.java
src/main/java/inito/filesystem/commands/*.java
```

Run the File System:

```
java -cp target inito.filesystem.InMemoryFileSystem
```

Usage

The IMFS is a command-line application. After running the system, you can execute various commands to interact with the file system.

Example Commands:

1. mkdir new_directory: Create a new directory named "new_directory."
2. cd new_directory: Change the current directory to "new_directory."
3. ls: List the contents of the current directory.
4. cat file.txt: Display the contents of "file.txt."

Refer to the Features section for a complete list of supported commands.

Data Structures Used

Directory Structure
The core of the In-Memory File System (IMFS) is modeled as a tree structure of directories, with each directory having references to its children (subdirectories and files). The primary attributes of the Directory class include:

```
public class Directory {
    private String name;
    private Map<String, Directory> directories;
    private Map<String, File> files;
    // Other attributes and methods
}
```

File Structure
Files are represented by the File class, which holds the content of the file along with other relevant attributes:

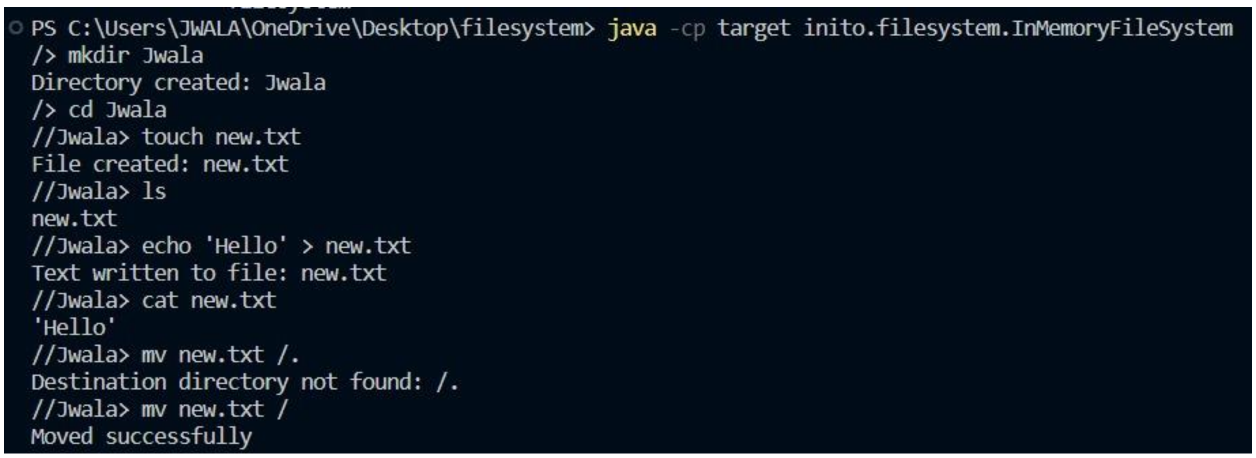
```
public class File {
    private String name;
    private String content;
    // Other attributes and methods
}
```

Design Decisions

Modularity: Implemented using the Command Design Pattern for extensibility.

In-Memory Storage: All file system data stored in memory for fast access.

Directory Structure: Hierarchical tree-like structure for effective navigation.



```
PS C:\Users\JWALA\OneDrive\Desktop\filesystem> java -cp target inito.filesystem.InMemoryFileSystem
/> mkdir Jwala
Directory created: Jwala
/> cd Jwala
//Jwala> touch new.txt
File created: new.txt
//Jwala> ls
new.txt
//Jwala> echo 'Hello' > new.txt
Text written to file: new.txt
//Jwala> cat new.txt
'Hello'
//Jwala> mv new.txt /.
Destination directory not found: /.
//Jwala> mv new.txt /
Moved successfully
```