

# Auto System Configuration Tool

Welcome to the Auto System Configuration Tool! This Node.js project helps you automate system configuration by dynamically updating configuration files based on your specifications.

## Project Structure

```
- config
  - config.json
- core
  - index.js
  - techconfig.json
- run
  - windows
    - run_techfile.bat
  - linux
    - run_techfile.sh
```

## Configuration Setup

### 1. config/config.json

Edit this file to set the root source path where your application exists ( `sourcepath` ) and the inbox path ( `INBOX_PATH` ) for your system.

```
{
  "sourcepath": "<<sourcecoderoopath>>", // Root source where the application
exists, e.g., "C://APP1"
  "INBOX_PATH": "C:/downloads" // Path for downloaded files (parameter)
}
```

**Note:** Customize the `sourcepath` according to your system. Also you can add up as many parameters you need which can be used further below.

### 2. core/techconfig.json

Configure your system changes using JSON-based or line-based configurations.

#### JSON Configuration:

```
[
  {
    "path": "nodeservicecenter/dbconfig.json", // Path to the JSON configuration
file
    "extension": "json", // Use "json" for JSON-based configuration
    "changes": [
      {
        "input": "applicationUrl", // JSON config key to change
        "output": "${config.applicationUrl}" // New value, use ${config,<<key>>
from /config/config.json}
      }
    ]
  }
]
```

```
]
}
]
```

#### Line Configuration:

```
{
  "path": "watchdog/automatedUploads/config.env", // Path to the configuration file
  "extension": "", // Leave blank for line config change
  "changes": [
    {
      "input": "INBOX_PATH", // Keyword of the line you want to change
      "output": "INBOX_PATH=${config.INBOX_PATH}" // Use the value from config.json
    }
  ]
}
```

**Note:** Ensure proper configuration according to your application requirements.

## How to Use

1. Install dependencies:

```
npm install
```

2. Run the tool:

- For Linux systems:

```
$ cd system-configuration-automation/run/linux
$ sh run_techfile.sh
```

- For Windows systems:

Double-click on `run_techfile.bat` in the `system-configuration-automation/run/windows` folder.

## Notes

- Logs are created in the `log` folder.
- Backups of previous files are stored in the `filebackup` folder.

## Troubleshooting

If you encounter issues:

- Ensure prerequisites are installed.
- Check `config.json` formatting.
- Verify paths in `config.json`.

For any further assistance, feel free to reach out!

---

Node.js 14.17.3 license MIT Maintained? Yes PRs Welcome