# Phishing Email Detection Documentation

## Overview

This project implements a phishing email detection system using rule-based logic and design patterns. The system detects phishing emails based on predefined rules, such as checking for suspicious links, sender addresses, and urgent language.

## Project Structure

phishing\_email\_detector/  
│  
├── detector/  
│ ├── \_\_init\_\_.py  
│ ├── base\_rule.py  
│ ├── suspicious\_links\_rule.py  
│ ├── sender\_address\_rule.py  
│ ├── urgent\_language\_rule.py  
│ └── rule\_factory.py  
│  
├── config/  
│ ├── \_\_init\_\_.py  
│ └── settings.py  
│  
├── main.py  
└── README.md

## Requirements

- Python 3.x

## Setup and Installation

1. \*\*Clone the repository:\*\*

```sh  
git clone <repository\_url>  
```  
Replace `<repository\_url>` with the actual URL of your repository.

2. \*\*Navigate to the project directory:\*\*

```sh  
cd phishing\_email\_detector  
```

3. \*\*Install the required dependencies:\*\*

If there are any additional dependencies, create a `requirements.txt` file and install them using:  
```sh  
pip install -r requirements.txt  
```

## How to Run the Project

1. \*\*Run the main script:\*\*

```sh  
python main.py  
```

## Sample Inputs and Outputs

### Sample Input

A sample email to be processed by the system could look like this:  
```python  
sample\_email = {  
 'subject': 'Urgent: Verify Your Account',  
 'sender': 'no-reply@bank.com',  
 'body': 'Please verify your account immediately by clicking the link below:\nhttp://fakebank.com/verify'  
}  
```

### Sample Output

The output will indicate whether phishing is detected based on the specified rules:  
```sh  
Detected phishing email: True  
```  
The output can be more detailed, specifying which rules were triggered:  
```sh  
Phishing detected due to:  
- Suspicious links  
- Urgent language  
```

## Detailed Steps for Running the Project

1. \*\*Open a terminal and navigate to the project directory:\*\*

```sh  
cd /path/to/phishing\_email\_detector  
```

2. \*\*Run the main script:\*\*

```sh  
python main.py  
```

3. \*\*Review the output in the terminal:\*\*

The script will process a predefined sample email and print the results indicating whether phishing is detected.

## Customization

You can customize the rules or add new rules by modifying the files in the `detector/` directory. Each rule should inherit from `base\_rule.py` and be registered in `rule\_factory.py`.