

```

pipeline {
    agent any

    tools {
        jdk 'jdk17'
        maven 'maven3'
    }
    environment {
        DOCKER_IMAGE = "sanjay9888/boardgame:${BUILD_NUMBER}"
    }
    stages {
        stage('Checkout') {
            steps {
                git branch: 'main', url: 'https://github.com/Sanjay6372/boardgame.git'
            }
        }
        stage('Compile and Test with Maven') {
            steps {
                sh 'mvn compile && mvn test'
            }
        }
        stage('File System Scan with Trivy') {
            steps {
                sh "trivy fs --format table -o trivy-fs-report.html ."
            }
        }
        stage('Static Code Analysis with SonarQube') {
            environment {
                SONAR_URL = "http://51.20.187.155:9000"
            }
            steps {
                withCredentials([string(credentialsId: 'sonarqube', variable: 'SONAR_AUTH_TOKEN')]) {
                    sh 'mvn sonar:sonar -Dsonar.login=$SONAR_AUTH_TOKEN -Dsonar.host.url=${SONAR_URL}'
                }
            }
        }
        stage('Build artifacts with Maven') {
            steps {
                sh "mvn package"
            }
        }
        stage('Publish artifacts To Nexus') {
            steps {
                withMaven(globalMavenSettingsConfig: 'global-settings', jdk: 'jdk17', maven: 'maven3', mavenSettingsConfig: '', traceability: true) {
                    sh 'mvn clean deploy -X'
                }
            }
        }
        stage('Build & Tag Docker Image') {
            steps {
                script {
                    withDockerRegistry(credentialsId: 'docker-cred') {
                        sh "docker build -t ${DOCKER_IMAGE} ."
                    }
                }
            }
        }
    }
}

```

Activate Windows
Go to Settings to activate Windows

```
    }
  }
}
stage('Docker Image Scan') {
  steps {
    sh "trivy image --format table -o trivy-image-report.html ${DOCKER_IMAGE}"
  }
}
stage('Push Docker Image') {
  steps {
    script {
      withDockerRegistry(credentialsId: 'docker-cred') {
        sh "docker push ${DOCKER_IMAGE}"
      }
    }
  }
}
stage('Update Deployment File') {
  environment {
    GIT_REPO_NAME = "boardgame"
    GIT_USER_NAME = "Sanjay6372"
  }
  steps {
    withCredentials([string(credentialsId: 'sanjugithub', variable: 'GITHUB_TOKEN')]) {
      sh '''
        git config --global user.email "sanjay@gmail.com"
        git config --global user.name "sanjay"
      '''
    }
  }
}
```

```

        git config --global user.email "sanjay@gmail.com"
        git config --global user.name "sanjay"
        sed -i "s/18/${BUILD_NUMBER}/g" src/deployment.yml
        git add src/deployment.yml
        git commit -m "Update deployment image to version ${BUILD_NUMBER}"
        git push https://${GITHUB_TOKEN}@github.com/${GIT_USER_NAME}/${GIT_REPO_NAME} HEAD:main
    ...
}
}
}
}
}
post {
    always {
        emailxtext (
            subject: "Pipeline Status: ${BUILD_NUMBER}",
            body: '''
                <html>
                <body>
                    <p>Build Status: ${BUILD_STATUS}</p>
                    <p>Build Number: ${BUILD_NUMBER}</p>
                    <p>Check the <a href="${BUILD_URL}">console output</a>.</p>
                </body>
                </html>
            ''',
            to: 'sanjukumar6372@gmail.com',
            from: 'jenkins@example.com',

            from: 'jenkins@example.com',
            replyTo: 'jenkins@example.com',
            mimeType: 'text/html',
            attachmentsPattern: 'trivy-image-report.html'
        )
    }
}
}

```