

# YAZILIM GELİŞTİRME ORTAM VE ARAÇLARI PROJE RAPORU

[Mert Taşan]

[H5160021]

[GitHub Adresi: <https://github.com/merttsn/Proje>]

## İçindekiler

1. SPRING INITIALİZR.....	3
2. GITHUB REPOSITORY.....	4
3. GITHUB'A PROJE YÜKLEME .....	6
4. JENKINS .....	12
5. SONARQUBE.....	23
6. JAVADOC .....	27
7. TRAVIS & CODECOV .....	40
8. POSTMAN.....	44
9. JMETER.....	46
10. UML DİAGRAM .....	51

# 1.SPRING INITIALIZR

- <https://start.spring.io> adresine girerek Spring Boot kullanılacak projemizin Group, Artifact, Name, Description, Packaging, Java Version ve Dependencies bölümlerini belirliyoruz ve Generate Project'e tıklayarak projemizin indiriyoruz.

## SPRING INITIALIZR bootstrap your application now

Generate a Maven Project with Java and Spring Boot 1.5.9

### Project Metadata

Artifact coordinates

Group

com.github

Artifact

Final

Name

Final

Description

Demo project for Spring Boot

Package Name

com.github.Final

Packaging

Jar

Java Version

8

### Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Web, Security, JPA, Actuator, Devtools...

Selected Dependencies

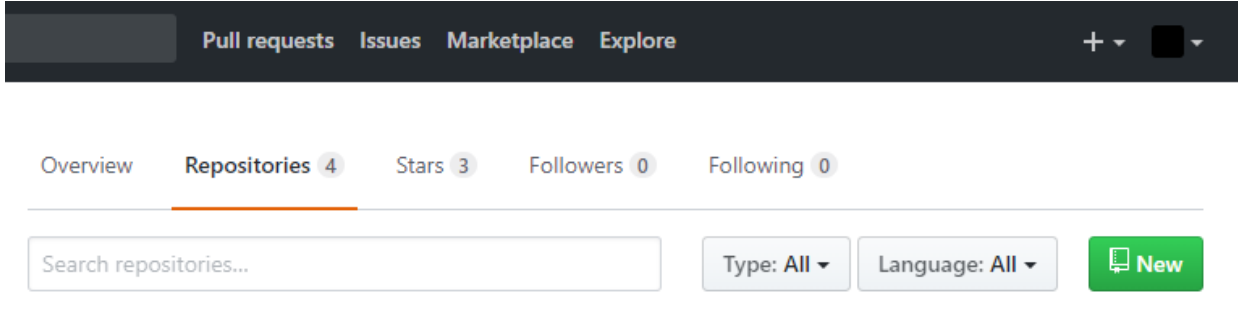
Web

Too many options? [Switch back to the simple version.](#)

Generate Project alt + ⌘

## 2. GITHUB REPOSITORY


- GitHub hesabımıza giriş yaptıktan sonra profilimizden “Repositories” sekmesine tıklıyoruz.



- Daha sonra “New” butonuna tıklıyoruz ve karşımıza gelen sayfadan “Repository Name” kısmına istediğimiz repository adını giriyor ve “Initialize this repository with a README” kısmına tik atıyoruz daha sonra “Create Repository” butonuna tıklıyoruz.

### Create a new repository

A repository contains all the files for your project, including the revision history.

Owner:  merttsn / Repository name: Final Proje ✓

Great repository name: Your new repository will be created as Final-Proje. Learn more about shiny-broccoli.

Description (optional):

☐ Public  
Anyone can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

☒ Initialize this repository with a README  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None ⓘ

Create repository

- Karşımıza gelecek sayfa böyle olacaktır.

merittsn / Final-Proje

Unwatch 1

Star 0

Fork 0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

No description, website, or topics provided.

Edit

Add topics

1 commit

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

merittsn Initial commit

Latest commit e4eb5f8 just now

README.md

Initial commit

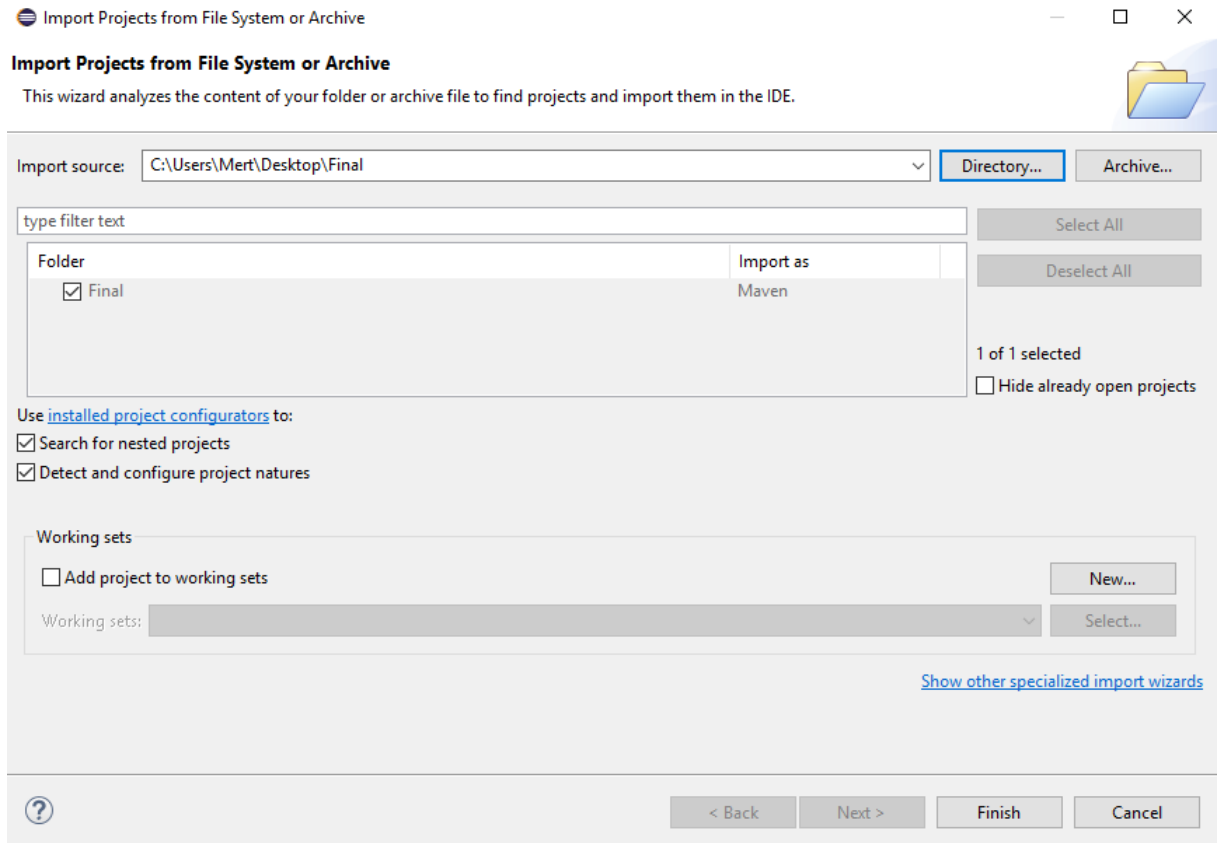
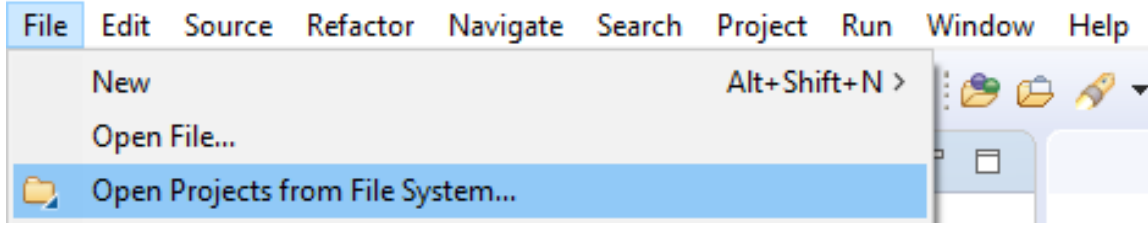
just now

README.md

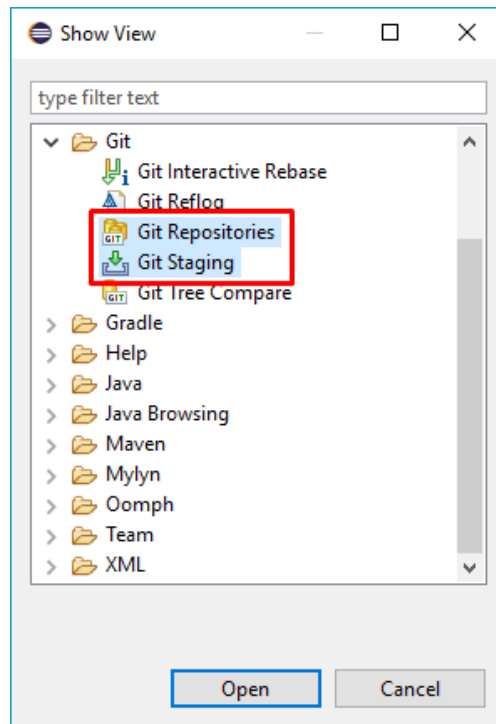
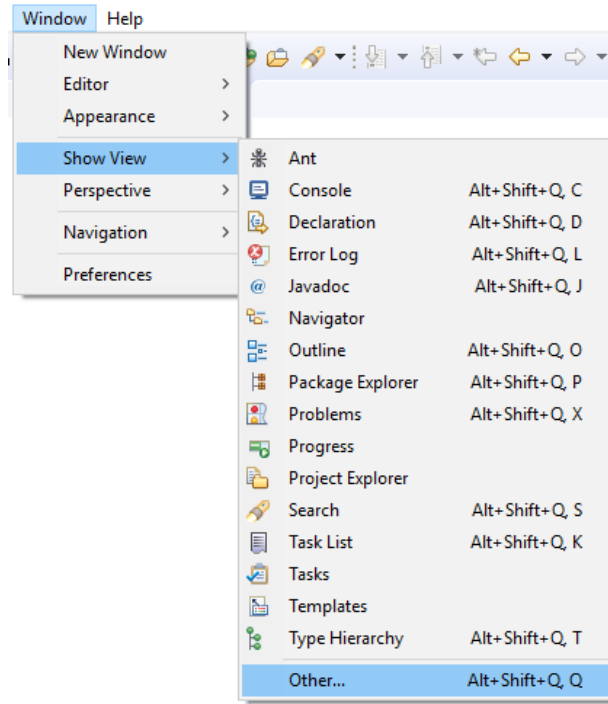
# Final-Proje

### 3. GITHUB'A PROJE YÜKLEME

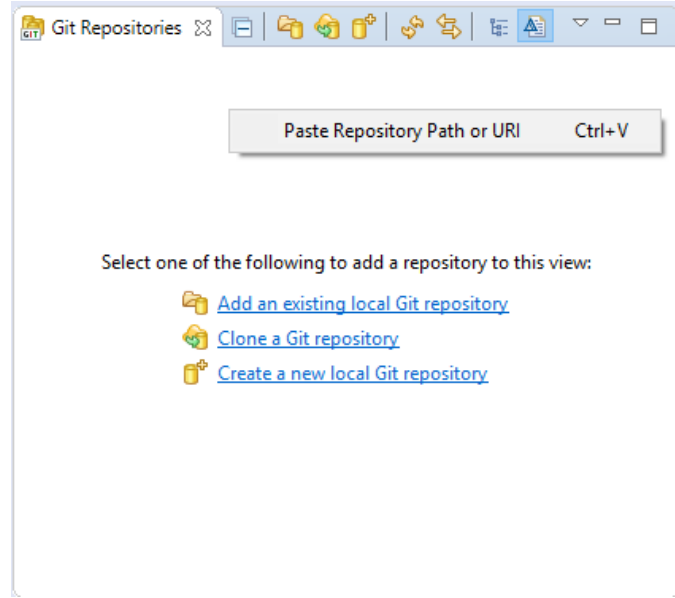
- Eclipse'i açtıktan sonra Spring.io üzerinden indirdiğimiz projemizi açıyoruz.



- GitHub Repository'mizi eklemek için gerekli olan sekmeleri Eclips'te açıyoruz.



- **Git Repositories** bölümüne sağ tıklayıp “**Paste Repository Path or URL**” diyoruz.

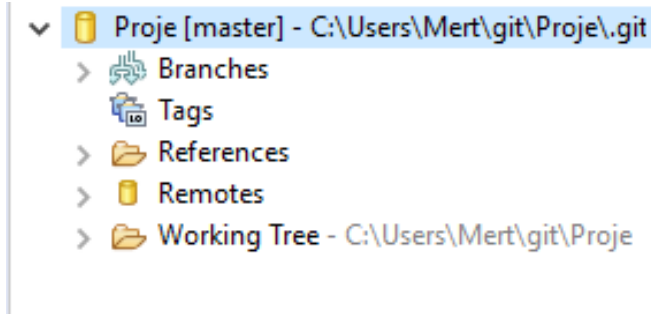


- Gerekli alanları dolduruyoruz.

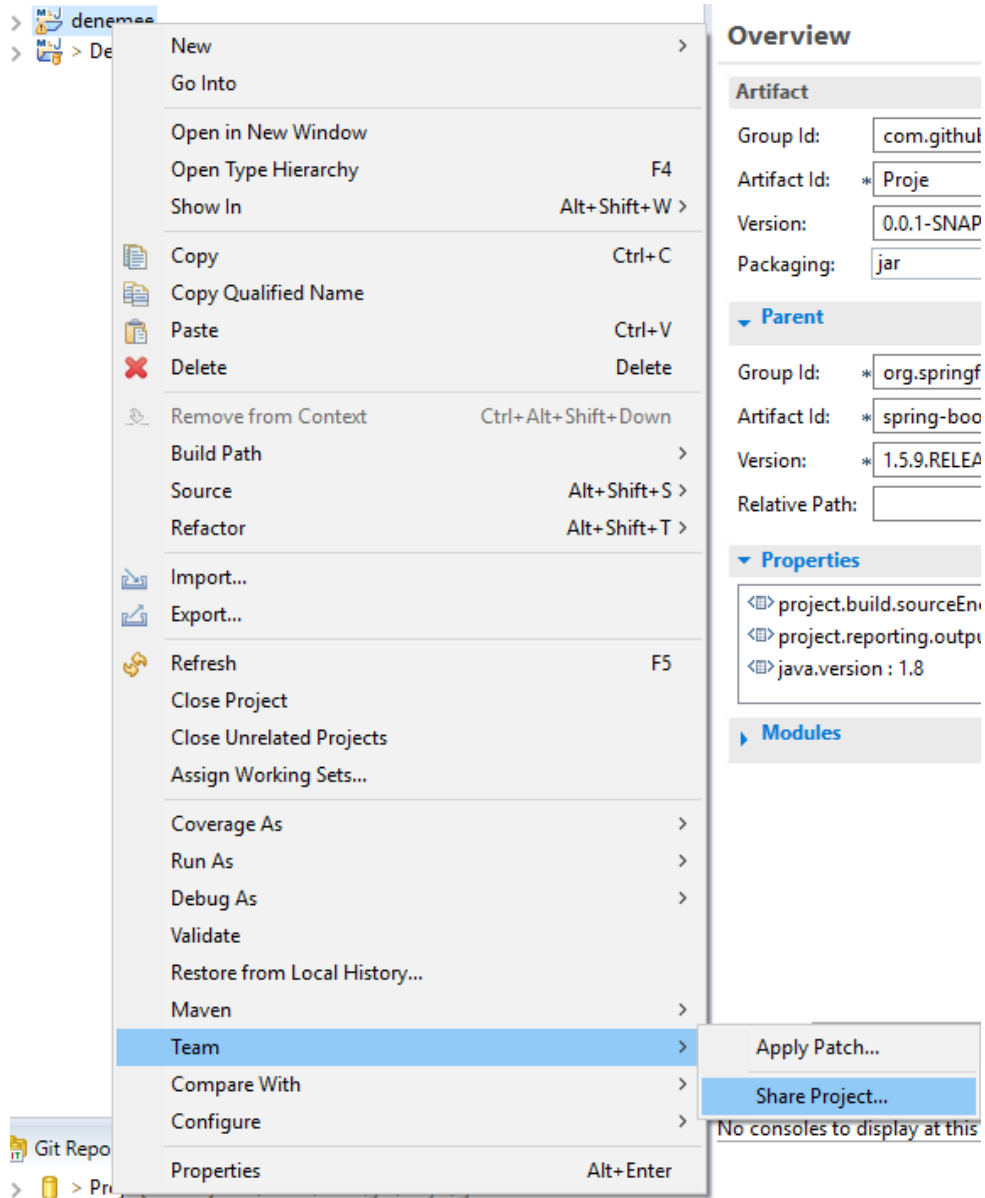
A screenshot of the 'Clone Git Repository' dialog box. The title bar says 'Clone Git Repository'. Below the title bar, there is a section titled 'Source Git Repository' with a subtitle 'Enter the location of the source repository.' and a Git logo. The dialog is divided into four sections: 'Location', 'Connection', and 'Authentication'. The 'Location' section has three text boxes: 'URL:' containing 'https://github.com/merttsn/Proje.git', 'Host:' containing 'github.com', and 'Repository path:' containing '/merttsn/Proje.git'. There is a 'Local File...' button next to the URL box. The 'Connection' section has a 'Protocol:' dropdown menu set to 'https' and an empty 'Port:' text box. The 'Authentication' section has a 'User:' text box containing 'merttsn', a 'Password:' text box filled with dots, and a checkbox labeled 'Store in Secure Store' which is unchecked. At the bottom of the dialog, there is a row of buttons: a help icon (?), '< Back', 'Next >', 'Finish', and 'Cancel'. The 'Next >' button is highlighted with a blue border.




- Repo'muz bu şekilde görünecektir.



- Projemizi GitHub Repo'muza bu adımlarla yüklüyoruz.



 Configure Git Repository

**Configure Git Repository**


Select an existing repository or create a new one

☐ Use or create repository in parent folder of project

Repository:  Create...


Working tree:




Path within repository:  Browse...


Project	Current Location	Target Location
<input checked="" type="checkbox"/>  denemee	C:/Users/Mert/eclipse-workspace/denemee	C:/Users/Mert/git/Proje/denemee



? Finish Cancel

- Yaptığımız değişiklikleri ise bu şekilde atıyoruz.

 > **Proje [master]**

Unstaged Changes (1)   


 > RESTServisTests.java - Desktop\_Proje/src/test/java/com/github/Proje

Staged Changes (0)  

 > Proje [master]

Unstaged Changes (0)

Staged Changes (1)


 RESTServisTests.java - Desktop\_Proje/src/test/java/com/github/Proje

Commit Message

test


Author: merttsn <merttnorg@gmail.com>

Committer: merttsn <merttnorg@gmail.com>

 Commit and Push...

 Commit

- Kullanıcı adımız ve şifremizi giriyoruz.

 Login

Repository

https://github.com/merttsn/Proje.git

User

merttsn

Password

.....

Store in Secure Store

☐



OK

Cancel

## 4.JENKINS

- <https://jenkins.io/download/> bu adrese girerek Jenkins'i indiriyoruz.

### Download Jenkins 2.89.3 for:

Docker
FreeBSD
Gentoo 
Mac OS X
OpenBSD 
openSUSE
Red Hat/Fedora/CentOS
Ubuntu/Debian
Windows
Generic Java package (.war)

- Cmd üzerinden indirdiğimiz yere gelip şu kodları yazıyoruz. Eğer httpPort:8080 doluysa kodumuzun sonuna - -httpPort=49001 yazıyoruz.

```
Microsoft Windows [Version 10.0.16299.125]
(c) 2017 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\Mert>cd Downloads

C:\Users\Mert\Downloads>java -jar jenkins.war
```

- Kurulum tamamlandıktan sonra şöyle bir ekran gelecektir.

```
INFO:
*****
*****
*****

Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

d403092a75e24022ab0036b8286c4490

This may also be found at: C:\Users\Mert\.jenkins\secrets\initialAdminPassword

*****
*****
*****

Oca 31, 2018 12:13:43 AM hudson.model.UpdateSite updateData
INFO: Obtained the latest update center data file for UpdateSource default
Oca 31, 2018 12:13:43 AM hudson.model.UpdateSite updateData
INFO: Obtained the latest update center data file for UpdateSource default
Oca 31, 2018 12:13:43 AM hudson.model.DownloadService$Downloadable load
INFO: Obtained the updated data file for hudson.tasks.Maven.MavenInstaller
Oca 31, 2018 12:13:43 AM hudson.WebAppMain$3 run
INFO: Jenkins is fully up and running
Oca 31, 2018 12:13:44 AM hudson.model.DownloadService$Downloadable load
INFO: Obtained the updated data file for hudson.tools.JDKInstaller
Oca 31, 2018 12:13:44 AM hudson.model.AsyncPeriodicWork$1 run
INFO: Finished Download metadata. 6.417 ms
```

- Burada görünen şifreyi kopyalıyoruz.

```
INFO:
*****
*****
*****
Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:
j403092a75e24022ab0036b8286c4490
This may also be found at: C:\Users\Mert\.jenkins\secrets\initialAdminPassword
*****
*****
*****
Oca 31, 2018 12:13:43 AM hudson.model.UpdateSite updateData
INFO: Obtained the latest update center data file for UpdateSource default
Oca 31, 2018 12:13:43 AM hudson.model.UpdateSite updateData
INFO: Obtained the latest update center data file for UpdateSource default
Oca 31, 2018 12:13:43 AM hudson.model.DownloadService$Downloadable load
INFO: Obtained the updated data file for hudson.tasks.Maven.MavenInstaller
Oca 31, 2018 12:13:43 AM hudson.WebAppMain$3 run
INFO: Jenkins is fully up and running
Oca 31, 2018 12:13:44 AM hudson.model.DownloadService$Downloadable load
INFO: Obtained the updated data file for hudson.tools.JDKInstaller
Oca 31, 2018 12:13:44 AM hudson.model.AsyncPeriodicWork$1 run
INFO: Finished Download metadata. 6.417 ms
```

- Daha sonra tarayıcımızın adres satırına localhost:49001 (httpPort=Bu kısma yazdığınız port) adresini giriyoruz. Karşımıza gelen sayfada “Administrator Password” kısmına kopyaladığımız şifreyi yapıştırıyor ve Continue butonuna tıklıyoruz.

## Getting Started

# Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

```
C:\Users\Mert\.jenkins\secrets\initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

- “Install suggested plugins” i seçiyoruz.

# Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

## Install suggested plugins

Install plugins the Jenkins community finds most useful.

## Select plugins to install

Select and install plugins most suitable for your needs.

- Bizden istenen bilgileri giriyor ve “**Save and Finish**” butonuna tıklıyoruz veya “**Continue as admin**” e tıklayarak devam ediyoruz. Sonraki sayfadan “**Start using Jenkins**” butonuna tıkladıktan sonra Jenkins ana sayfasına geçiyoruz.

# Create First Admin User

Kullanıcı Adı:

Şifre:

Şifreyi Doğrula:


Tam İsim:

E-posta adresi:

[Continue as admin](#)


[Save and Finish](#)


- Jenkins ana sayfasından “Jenkins’i Yönet” e tıklıyoruz.





# Jenkins


Jenkins ▶


 Yeni Item


 Kişiler

 Yapılandırma Geçmişi

 Jenkins'ı Yönet

 My Views

 Credentials

 New View

## Jenkinsa Hoşgeldiniz!

Lütfen başlamak için [yeni işler oluşturun](#)

### Yapılandırma Listesi

Sırada bekleyen yapılandırma yok.

### Yapılandırıcı Durumu

1	Beklemede
2	Beklemede

- “Global Tool Configuration” a giriyoruz.

## Jenkins'i Yönet



[Sistem Konfigürasyonunu Değiştir](#)

Evrensel ayarları ve pathları konfigüre et



[Configure Global Security](#)

Secure Jenkins; define who is allowed to access/use the system.



[Configure Credentials](#)

Configure the credential providers and types



[Global Tool Configuration](#)

Configure tools, their locations and automatic installers.



- “Add JDK” ya tıkladıktan sonra resimdeki gibi dolduruyoruz. JAVA\_HOME bölümü Java jdk’yı kurduğunuz yerin adresidir.

## JDK

JDK installations

Add JDK

List of JDK installations on this system

JDK

İsim

Java 8

JAVA\_HOME

C:\Program Files\Java\jdk1.8.0\_161\

☐ Install automatically

Add JDK

List of JDK installations on this system

- Git kullanmak için gerekli işlemleri yapıyoruz. “JGit” ve “JGit with Apache HTTP client” seçeneklerini seçiyoruz.

Git

Name

Default

Path to Git executable

git.exe

☐ Install automatically

Add Git

Git

JGit

JGit with Apache HTTP client

List of Gradle installations on this system

- Maven için gerekli işlemleri yapıyoruz ve Save diyoruz.

**Maven**

Maven installations **Add Maven**

List of Maven installations on this system

**Maven**

Maven installations

Maven

isim Maven 3.5.2

☒ Install automatically

**Install from Apache**

Version 3.5.2 ▼

- Aşağıdaki seçenekleri seçerek devam ediyoruz.

**Jenkins**

Jenkins ▶

**Yeni Item**

Kişiler

Yapılandırma Geçmişi

Jenkins'i Yönet

My Views

Credentials

New View

**Yapılandırma Listesi**

Sırada bekleyen yapılandırma yok.

**Yapılandırıcı Durumu**

1 Beklemede

2 Beklemede

**Jenkinsa Hoşgeldiniz!**

Lütfen başlamak için **yeni işler oluşturun**

## Enter an item name

Maven Build

» Required field



### Serbest-stil yazılım projesi yapılındır

Jenkins'in merkezi özelliği, projelerinizi yapılandırmanıza yardım etmesidir. Bu proje türünü kullanarak, herhangi bir yapılandırma sistemini herhangi bir Kaynak Kodu Yönetimi aracı ile birleştirebilirsiniz,ve hatta yazılım yapılandırmanın dışında başka tür projeler için dahi kullanabilirsiniz.



### Pipeline

Orchestrates long-running activities that can span multiple build slaves. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



### Çoklu-konfigürasyona sahip proje yapılındır

Platforma Özgü yapılandırmalara sahip veya çeşitli ortamlarda test işlemi yapan projeler gibi çok fazla sayıda konfigürasyona ihtiyaç duyan projeler için uygundur.



### Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.



### GitHub Organization

Scans a GitHub organization (or user account) for all repositories matching some defined markers.



### Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository.

OK

- GitHub Repository'mize gelip linkini aşağıdaki şekilde kopyalıyoruz. Resimlerde olduğu şekilde ayarlarımızı yapıyoruz.

merttsn / Proje

Unwatch 1 Star 0 Fork 0

<> Code

Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

No description, website, or topics provided. 

Edit

Add topics

50 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

merttsn committed 2 days ago test

Desktop\_Proje

test

.travis.yml

Update .travis.yml

README.md

Update README.md

README.md

Clone with HTTPS

Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/merttsn/Proje.git

Open in Desktop

Download ZIP

General **Kaynak Kodu Yönetimi** Yapılandırma Tetikleyiciler Yapılandırma Ortamı Yapılandırma Yapılandırma-sonrası Aksiyonlar

### Kaynak Kodu Yönetimi

☐ Hiçbirisi  
☒ Git

Repositories

Repository URL

Credentials

Branches to build

Branch Specifier (blank for 'any')

Git executable

- Daha sonra “**Yapılandırma Tetikleyiciler**” sekmesine geçiyoruz ve resimdeki seçeneği seçip kaydediyoruz.

General Kaynak Kodu Yönetimi **Yapılandırma Tetikleyiciler**

☐ Subversion

### Yapılandırma Tetikleyiciler

☐ Yapılandırmaları uzaktan tetikle (örn., scriptlerden)  
☐ Build after other projects are built  
☐ GitHub hook trigger for GITScm polling  
☐ Periyodik olarak yapılandır  
☐ SCM kontrolü

### Yapılandırma Ortamı

☐ Delete workspace before build starts  
☐ Use secret text(s) or file(s)  
☐ Abort the build if it's stuck  
☐ Add timestamps to the Console Output  
☐ With Ant

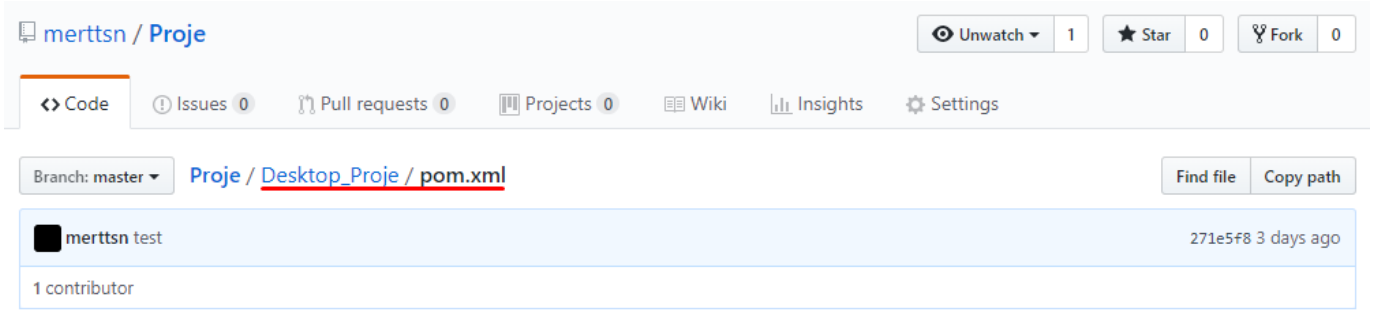
### Yapılandırma

Yapılandırma adımı ekle

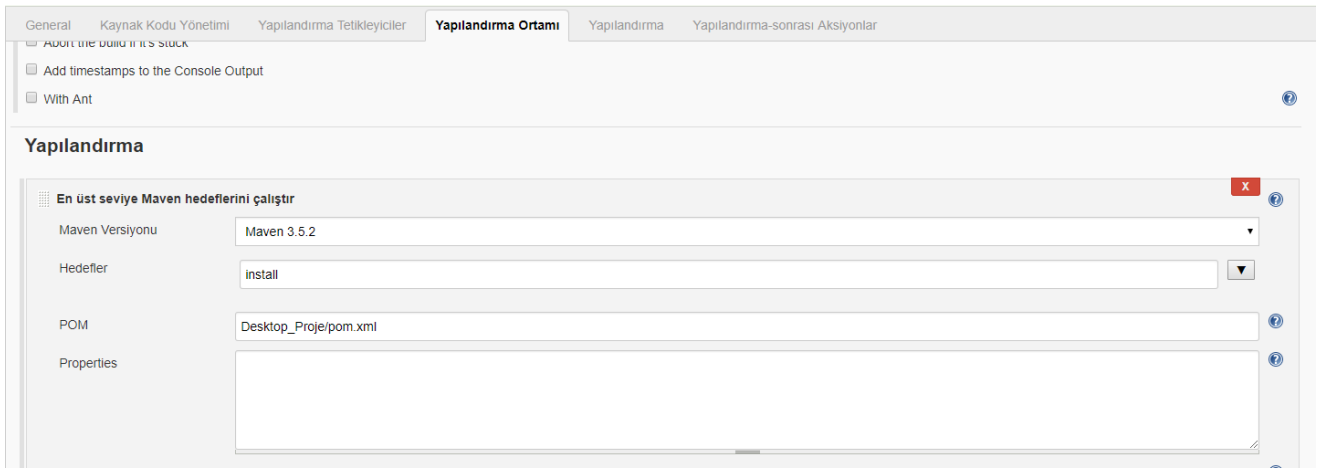
Ant çalıştır

En üst seviye Maven hedeflerini çalıştır

- GitHub'dan pom.xml dosyamızın adresini alıyoruz.



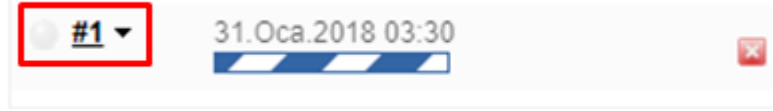
- Ayarlarımızı bu şekilde yapıp kaydediyoruz.











- Projemizi yapılandırıyoruz.



- Bu adımları izliyoruz. “BUILD SUCCES” yazısını gördüğümüz de işlemimiz tamamlanmış oluyor.



-  Projeye geri dön
-  Durum
-  Değişiklikler
-  Console Output
-  View as plain text
-  Sürüm Bilgisini Düzenle
-  Git Build Data
-  No Tags

```
[INFO] Generating "About" report          --- maven-project-info-reports-plugin:2.9:index
[INFO] Generating "Licenses" report       --- maven-project-info-reports-plugin:2.9:license
[INFO] Generating "Plugin Management" report --- maven-project-info-reports-plugin:2.9:plugin-management
[INFO] Generating "Plugins" report         --- maven-project-info-reports-plugin:2.9:plugins
[INFO] Generating "Team" report            --- maven-project-info-reports-plugin:2.9:project-team
[INFO] Generating "Source Code Management" report --- maven-project-info-reports-plugin:2.9:scm
[INFO] Generating "Summary" report         --- maven-project-info-reports-plugin:2.9:summary
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:05 min
[INFO] Finished at: 2018-01-31T03:31:58+03:00
[INFO] Final Memory: 100M/1956M
[INFO] -----
Finished: SUCCESS
```

## 5.SONARQUBE

- <https://www.sonarqube.org/downloads/> adresinden SonarQube uygulamamızı indiriyoruz.

### SonarQube 6.7.1 (LTS \*)

**Dec. 21, 2017**

Long Term Supported version, wraps together all the new feature SonarLint notifications, high availability.

\* LTS stands for Long Term Support. Blocker and Critical issues will be fixed on LATEST version.

[Documentation](#) – [Screenshots](#) – [Release notes](#) – [More details](#)

**DOWNLOAD LTS**

**MD5**

- İndirdiğimiz zip dosyasını çıkartıyoruz ve sonarqube-6.7.1\bin\windows-x86-64 yolunu izliyoruz. “**StarSonar.bat**” dosyasını çalıştırıyoruz.

sonarqube-6.7.1 > bin > windows-x86-64		
Ad		Değiştirme tarihi
lib		21.12.2017 08:48
InstallINTService.bat		21.12.2017 08:37
StartNTService.bat		21.12.2017 08:37
<u>StartSonar.bat</u>		21.12.2017 08:37
StopNTService.bat		21.12.2017 08:37
UninstallINTService.bat		21.12.2017 08:37
wrapper.exe		21.12.2017 08:37

- “SonarQube is up” ibaresini görüyorsa, SonarQube aktif demektir.

```
jvm 1 | at org.apache.logging.log4j.core.LoggerContext.start(LoggerContext.java:261)
jvm 1 | at org.elasticsearch.common.logging.LogConfigurator.configure(LogConfigurator.java:166)
jvm 1 | at org.elasticsearch.common.logging.LogConfigurator.configure(LogConfigurator.java:122)
jvm 1 | at org.elasticsearch.bootstrap.Bootstrap.init(Bootstrap.java:307)
jvm 1 | at org.elasticsearch.bootstrap.Elasticsearch.init(Elasticsearch.java:132)
jvm 1 | at org.elasticsearch.bootstrap.Elasticsearch.execute(Elasticsearch.java:123)
jvm 1 | at org.elasticsearch.cli.EnvironmentAwareCommand.execute(EnvironmentAwareCommand.java:70)
jvm 1 | at org.elasticsearch.cli.Command.mainWithoutErrorHandling(Command.java:134)
jvm 1 | at org.elasticsearch.cli.Command.main(Command.java:90)
jvm 1 | at org.elasticsearch.bootstrap.Elasticsearch.main(Elasticsearch.java:91)
jvm 1 | at org.elasticsearch.bootstrap.Elasticsearch.main(Elasticsearch.java:84)
jvm 1 |
jvm 1 | 2018-01-31 16:38:53,425 main ERROR Null object returned for Delete in DefaultRolloverStrategy.
jvm 1 | 2018.01.31 16:38:59 INFO app[][o.s.a.SchedulerImpl] Process[es] is up
jvm 1 | 2018.01.31 16:38:59 INFO app[][o.s.a.p.ProcessLauncherImpl] Launch process[[key='web', ipcIndex=2, logFileNam
mePrefix=web]] from [C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1]: C:\Program Files\Java\jre1.8.0_161\bin\java -Djava.awt
.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1\temp -Xmx512m -Xms128m
-XX:+HeapDumpOnOutOfMemoryError -cp ./lib/common/*;./lib/server/*;C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1\lib\jdbc\h2
\h2-1.3.176.jar org.sonar.server.app.WebServer C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1\temp\sq-process793372690954617
8742properties
jvm 1 | 2018.01.31 16:39:08 INFO app[][o.s.a.SchedulerImpl] Process[web] is up
jvm 1 | 2018.01.31 16:39:08 INFO app[][o.s.a.p.ProcessLauncherImpl] Launch process[[key='ce', ipcIndex=3, logFileNam
ePrefix=ce]] from [C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1]: C:\Program Files\Java\jre1.8.0_161\bin\java -Djava.awt.h
eadless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1\temp -Xmx512m -Xms128m -X
X:+HeapDumpOnOutOfMemoryError -cp ./lib/common/*;./lib/server/*;./lib/ce/*;C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1\li
b\jdbc\h2\h2-1.3.176.jar org.sonar.ce.app.CeServer C:\Users\Mert\Desktop\ara@\sonarqube-6.7.1\temp\sq-process35741325262
69779337properties
jvm 1 | 2018.01.31 16:39:10 INFO app[][o.s.a.SchedulerImpl] Process[ce] is up
jvm 1 | 2018.01.31 16:39:10 INFO app[][o.s.a.SchedulerImpl] SonarQube is up
```

- Localhost:9000 adresini tarayıcımızda açıyoruz.

## Continuous Code Quality

Log in

Read documentation

0

Projects Analyzed

0 Bugs

0 Vulnerabilities

0 Code Smells

## Multi-Language

20+ programming languages are supported by SonarQube thanks to our in-house code analyzers, including:

Java	C/C++	C#	COBOL	ABAP	HTML	RPG	JavaScript	TypeScript	Objective C	XML
VB.NET	PL/SQL	T-SQL	Flex	Python	Groovy	PHP	Swift	Visual Basic	PL/I	

## Quality Model

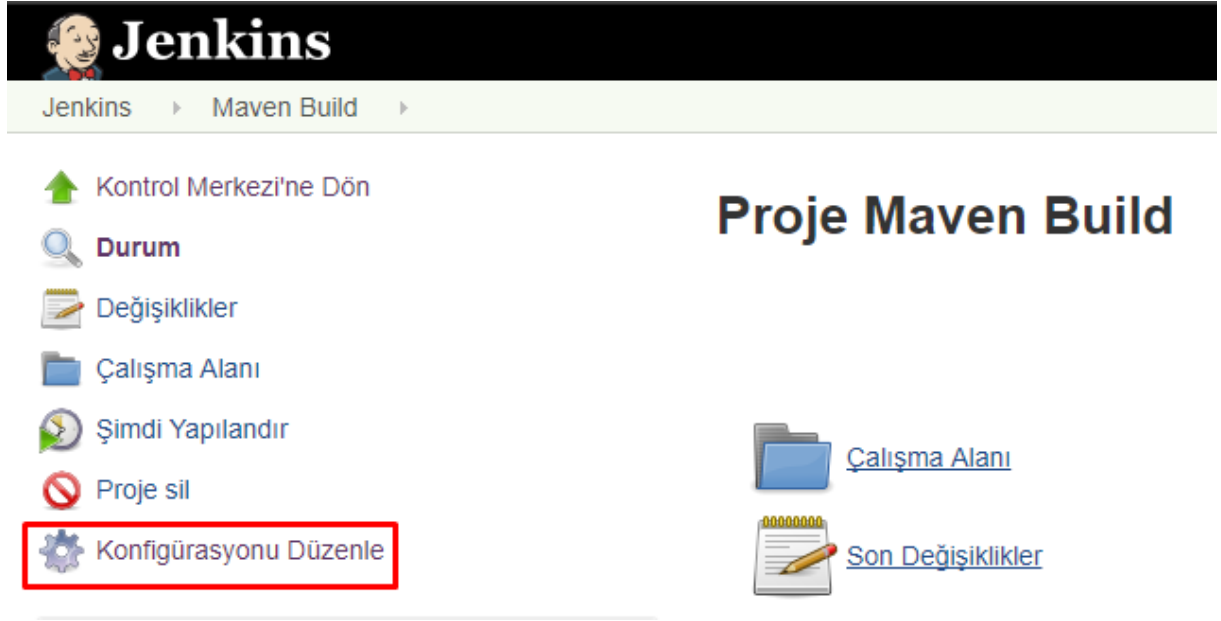
**Bugs** track code that is demonstrably wrong or highly likely to yield unexpected behavior.

**Vulnerabilities** are raised on code that is potentially vulnerable to exploitation by hackers.

**Code Smells** will confuse maintainers or give them pause. They are measured primarily in terms of the time they will take to fix.



- Daha sonra Jenkins’de oluşturduğumuz Maven Build adlı Item’ı açıyoruz ve “**Konfigurasyonu Düzenle**” kısmına giriyoruz.



The image shows the Jenkins web interface for a project named 'Maven Build'. The top navigation bar includes 'Jenkins' and 'Maven Build'. On the left sidebar, there are several links: 'Kontrol Merkezi'ne Dön', 'Durum', 'Değişiklikler', 'Çalışma Alanı', 'Şimdi Yapılandır', 'Proje sil', and 'Konfigurasyonu Düzenle' (which is highlighted with a red box). The main content area is titled 'Proje Maven Build' and contains two links: 'Çalışma Alanı' and 'Son Değişiklikler'.

- Buradan “**Yapılandırma Tetikleyiciler**” sekmesine gelerek “**Hedefler**” bölümüne “**sonar:sonar**” hedefini yazıyoruz ve “**Kaydet**” butonuna tıklıyoruz.

**Yapılandırma**

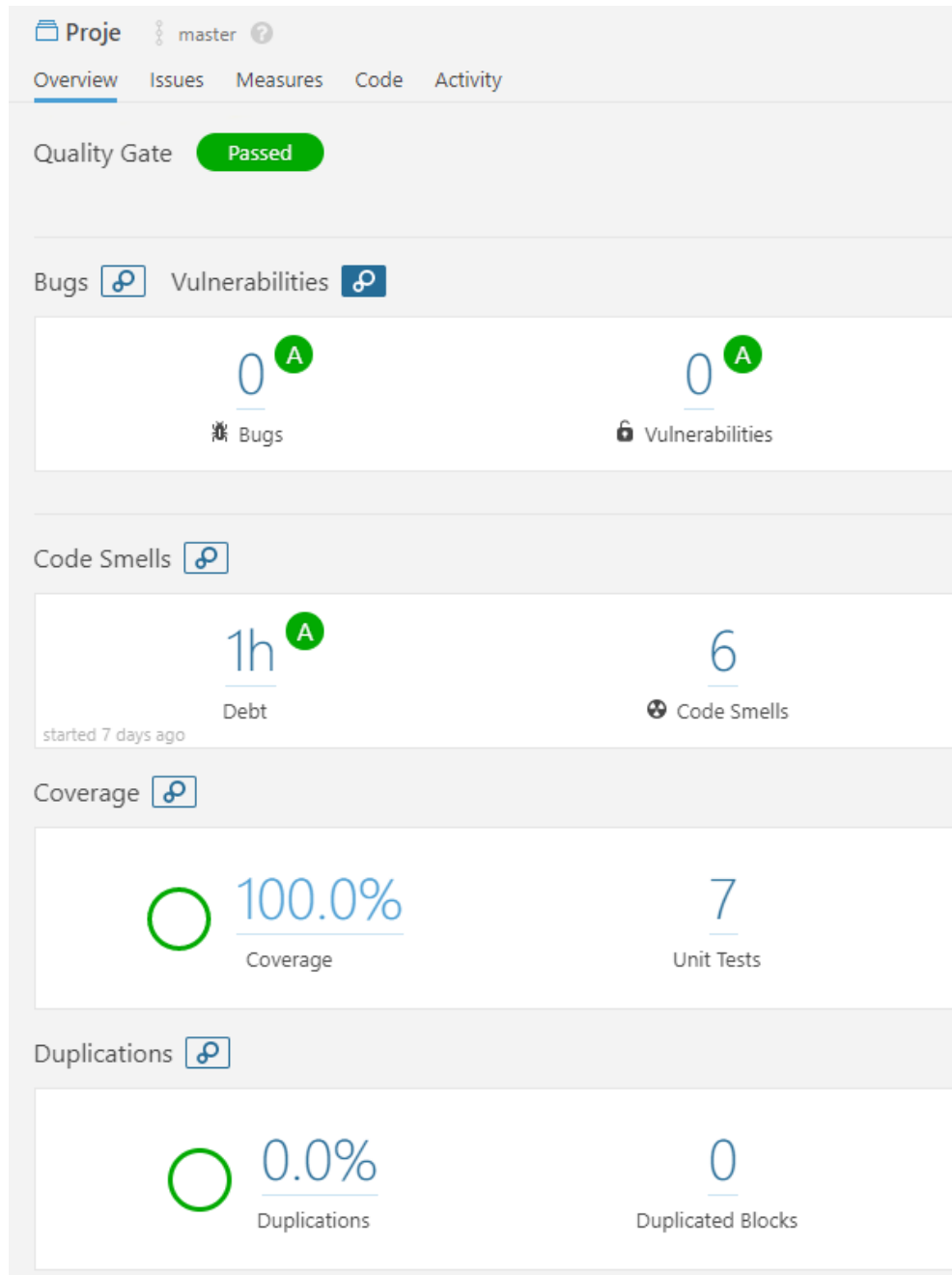
En üst seviye Maven hedeflerini çalıştır

Maven Versiyonu	Maven 3.5.2
Hedefler	install <u>sonar:sonar</u>
POM	Desktop_Proje/pom.xml

- “Şimdi Yapılandır” diyor ve “Console Output” sekmesine gelip bize vereceği SonarQube linkini alıyoruz.

```
[INFO] ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard/index/com.github:Proje
[INFO] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] More about the report processing at http://localhost:9000/api/ce/task?id=AWFMi0QLdQfLgfonWNT7
[INFO] Task total time: 3.228 s
[INFO]
```

- Tarayıcımızda açıyoruz ve SonarQube kodumuzun bilgilerini bize gösteriyor.



## 6.JAVADOC

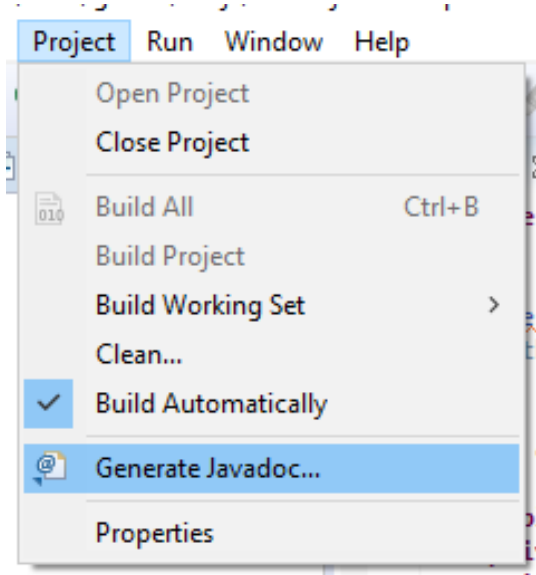
- Aşağıdaki etiketleri kullanarak Javadoc dökümantasyonumuzu Eclipse üzerinden oluşturuyoruz.

Tag	Description	Example
@author	Sınıfın yazarını tanımlar	@author tanımlama
@exception	Metot tarafından atılmış istisnayı tanımlar.	@exception istisna-ismi açıklama
@param	Metot paramentleri gösterir.	@param parametre-ismi açıklama
@return	Metodun return değerini gösterir.	@return açıklama
@since	Belirli bir değişiklik yapıldıysa release'i vurgular.	@since release
@throws	@exception ile benzer.	@throws etiketi @exception etiketi ile aynı anlama gelmektedir.
@version	Sınıfın versiyonunu belirtir.	@version info

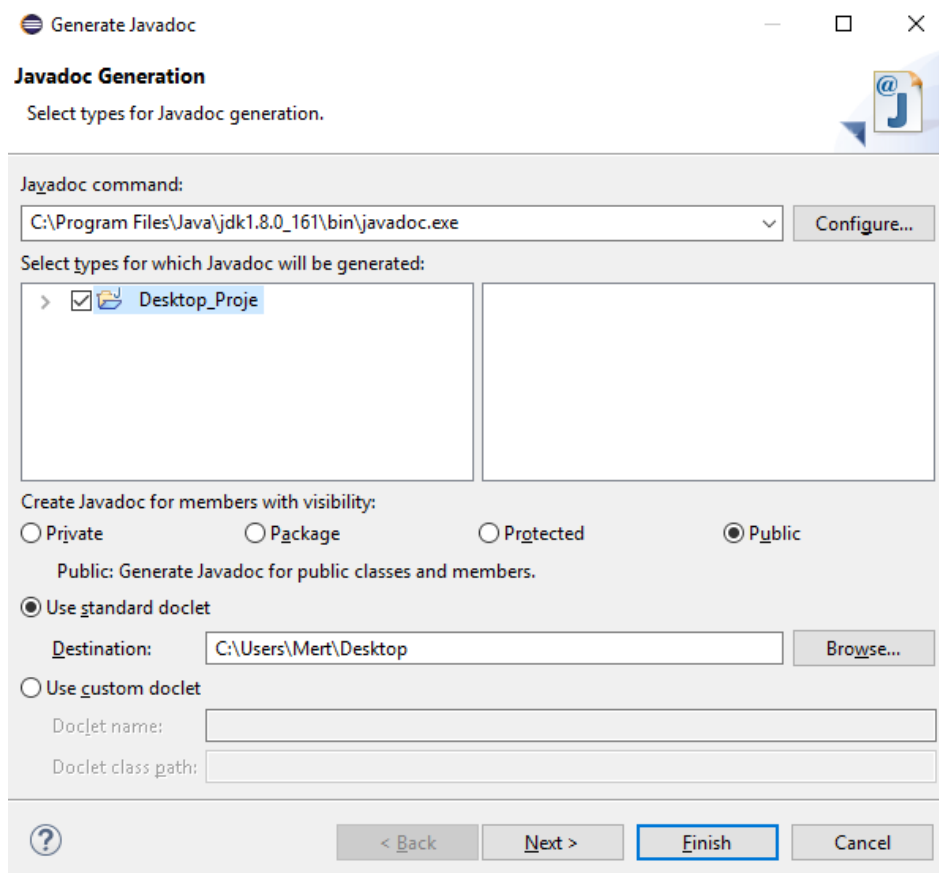
- Örneğin


```
/**  
 * Haber başlık ve içeriklerinin bulunduğu sınıf.  
 * @author Mert Taşan  
 */
```

- Etiketleri yazdıktan sonra **Project** -> **Generate Javadoc** kısmına geliyoruz.



- Configure kısmından Java/jdk içinde bulunan javadoc.exe'nin yerini gösteriyoruz.



 Generate Javadoc

**Javadoc Generation**  
Configure Javadoc arguments for standard doclet.

☐ Document title:

Basic Options

☒ Generate use page

☒ Generate hierarchy tree

☒ Generate navigator bar

☒ Generate index

☒ Separate index per letter

Document these tags

☒ @author

☒ @version

☒ @deprecated

☒ deprecated list

Select referenced archives and projects to which links should be generated:

☐ access-bridge-64.jar - not configured

☐ accessors-smart-1.1.jar - not configured

☐ android-json-0.0.20131108.vaadin1.jar - not configured

☐ asm-5.0.3.jar - not configured

☐ assertj-core-2.6.0.jar - not configured

☐ charsets.jar - <https://docs.oracle.com/javase/8/docs/api/>


☐ classes - not configured

Select All


Clear All

Browse...

☐ Style sheet:



- Türkçe karakterleri görebilmemiz için Extra Javadoc Options kısmına belirtilen UTF-8 kodlarını ekliyoruz.

 Generate Javadoc

**Javadoc Generation**  
Configure Javadoc arguments.

☐ Overview:

VM options (prefixed with '-J', e.g. -J-Xmx180m for larger heap space):


Extra Javadoc options (path names with white spaces must be enclosed in quotes):

JRE source compatibility:

☐ Save the settings of this Javadoc export as an Ant script:

Ant Script:

☒ Open generated index file in browser



- Generate işleminin tamamlanmasını bekliyoruz.

```
Constructing Javadoc information...
Standard Doclet version 1.8.0_161
Building tree for all the packages and classes...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\com\github\Proje\RETServisTests.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\com\github\Proje\package-frame.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\com\github\Proje\package-summary.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\com\github\Proje\package-tree.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\constant-values.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\com\github\Proje\class-use\RETServisTests.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\com\github\Proje\package-use.html...
Building index for all the packages and classes...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\overview-tree.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\index-files\index-1.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\index-files\index-2.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\index-files\index-3.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\deprecated-list.html...
Building index for all classes...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\allclasses-frame.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\allclasses-noframe.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\index.html...
Generating C:\Users\Mert\git\Proje\Desktop_Proje\help-doc.html...
```

- Tüm Javadoc dosyalarımız burada mevcut olacaktır.

```
> src/main/java
> src/main/resources
> src/test/java
> JRE System Library [JavaSE-1.8]
> Maven Dependencies
▼ > doc
  > com
  > index-files
    allclasses-frame.html
    allclasses-noframe.html
    constant-values.html
    deprecated-list.html
    help-doc.html
    index.html
    overview-tree.html
    package-list
    script.js
    stylesheet.css
```

- Index.html dosyasımızı tarayıcımız ile açıyoruz.

All Classes	PACKAGE	CLASS	USE	TREE	DEPRECATED	INDEX	HELP
Haber ProjeApplication RETServis RETServisTests	PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES			
<b>Package com.github.Proje</b>							
<b>Class Summary</b>							
Class	Description						
Haber	Haber başlık ve içeriklerinin bulunduğu sınıf.						
ProjeApplication	ProjeApplication Sınıfı						
RETServis	Rest Servisi Sınıfı						
RETServisTests	Projedeki tüm sınıfları test eder.						
PACKAGE	CLASS	USE	TREE	DEPRECATED	INDEX	HELP	
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES				

- Sırayla tüm sınıfları inceleyelim:

## Haber Sınıfı

### Class Haber

java.lang.Object  
com.github.Proje.Haber

```
public class Haber  
extends java.lang.Object
```

Haber başlık ve içeriklerinin bulunduğu sınıf.

Author:

Mert Taşan

#### Field Summary

##### Fields

Modifier and Type	Field and Description
static int	COUNT

#### Constructor Summary

##### Constructors

Constructor and Description
Haber(java.lang.String baslik, java.lang.String icerik)

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type		Method and Description
java.lang.String		getBaslik() Haber Başlığını döndürür.
java.lang.String		getIcerik() Haber İçeriğini döndürür.
int		getId() id'yi döndürür.

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

COUNT
public static int COUNT

Constructor Detail

Haber
<pre>public Haber(java.lang.String baslik,               java.lang.String icerik)</pre> <p>Parameters:</p> <p>baslik - RESTServis de yazılan haber başlığını çağırır.</p> <p>icerik - RESTServis de yazılan haber içeriğini çağırır.</p>



## ***Method Detail***

### **getBaslik**

```
public java.lang.String getBaslik()
```

Haber Başlığını döndürür.

Returns:

Haber Başlığı

### **getIcerik**

```
public java.lang.String getIcerik()
```

Haber İçeriğini döndürür.

Returns:

Haber İçeriği

### **getId**

```
public int getId()
```

id'yi döndürür.

Returns:

Haber id'si

# ProjeApplication Sınıfı

## Class ProjeApplication

java.lang.Object  
com.github.Proje.ProjeApplication

```
@SpringBootApplication
public class ProjeApplication
extends java.lang.Object
```

ProjeApplication Sınıfı

Author:  
Mert Taşan

### Constructor Summary

#### Constructors

##### Constructor and Description

ProjeApplication()

### Method Summary

#### All Methods

#### Static Methods

#### Concrete Methods

Modifier and Type	Method and Description
static void	main(java.lang.String[] args)

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructor Detail

#### ProjeApplication

```
public ProjeApplication()
```

### Method Detail

#### main

```
public static void main(java.lang.String[] args)
```

# RETServis Sınıfı

## Class RETServis

java.lang.Object  
com.github.Proje.RETServis

```
@RestController
public class RETServis
extends java.lang.Object
```

Rest Servisi Sınıfı

Author:  
Mert Taşan

### Constructor Summary

Constructors
Constructor and Description
<code>RETServis()</code>

### Method Summary

All Methods	Static Methods	Concrete Methods
Modifier and Type		Method and Description
<code>static java.util.List&lt;Haber&gt;</code>		<code>haberListele()</code> Tüm haberleri listeler.
<code>static Haber</code>		<code>haberOlustur(java.lang.String baslik, java.lang.String icerik)</code> Haberleri oluşturmak için kullanılır.
<code>static java.lang.String</code>		<code>haberSil(int index)</code> Belirli bir haberi silmek için kullanılır.
<code>static java.lang.String</code>		<code>haberTemizle()</code> Girilen tüm haberleri silmek için kullanılır.

### Methods inherited from class java.lang.Object

`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructor Detail

### RETServis

```
public RETServis()
```

## Method Detail

### haberOlustur

```
@RequestMapping(value="/haber/olustur")
public static Haber haberOlustur(java.lang.String baslik,
                                java.lang.String icerik)
```

Haberleri oluşturmak için kullanılır.

#### Parameters:

baslik - Gireceğimiz haberin başlığını belirler.

icerik - Gireceğimiz haberin içeriğini belirler.

#### Returns:

Girdiler sonucu oluşacak haberi döndürür.

### haberListele

```
@RequestMapping(value="/haber/listele")
public static java.util.List<Haber> haberListele()
```

Tüm haberleri listeler.

#### Returns:

Girilmiş tüm haberleri döndürür.

### haberTemizle

```
@RequestMapping(value="/haber/temizle")
public static java.lang.String haberTemizle()
```

Girilen tüm haberleri silmek için kullanılır.

#### Returns:

"Tüm haberler temizlendi" mesajını döndürür.

### haberSil

```
@RequestMapping(value="/haber/sil")
public static java.lang.String haberSil(int index)
```

Belirli bir haberi silmek için kullanılır.

#### Parameters:

index - Girilen index değerini belirtir.

#### Returns:

Girilen değerdeki haberin silindiğini gösteren "Belirlenen haber kaldırıldı" mesajını döndürür.

# RETServisTests Sınıfı

## Class RETServisTests

java.lang.Object  
com.github.Proje.RETServisTests

```
@SpringBootTest
@JsonTest
public class RETServisTests
extends java.lang.Object
```

Projedeki tüm sınıfları test eder.

Author:  
Mert Taşan

### Constructor Summary

Constructors
Constructor and Description
<a href="#">RETServisTests ()</a>

### Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type		Method and Description
void		<a href="#">testClass ()</a>
void		<a href="#">testHaber ()</a> Haber sınıfını test eder.
void		<a href="#">testHaberListele ()</a> Haberleri listeleyen sınıfı test eder.
void		<a href="#">testHaberOlustur ()</a> Haber oluşturan sınıfı test eder.
void		<a href="#">testHaberSil ()</a> Belirli haberleri silen sınıfı test eder.
void		<a href="#">testHaberTemizle ()</a> Haberlerin tamamını temizleyen sınıfı test eder.
void		<a href="#">testProjeApplication ()</a> ProjeApplication sınıfını test eder.

## Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructor Detail

### RETServisTests

```
public RETServisTests()
```

## Method Detail

### testClass

```
public void testClass()
```

### testHaber

```
public void testHaber()
           throws java.lang.Exception
```

Haber simfini test eder.

Throws:

`java.lang.Exception` - Hata verir.

### testProjeApplication

```
public void testProjeApplication()
```

ProjeApplication simfini test eder.

### testHaberOlustur

```
public void testHaberOlustur()
           throws java.lang.Exception
```

Haber oluşturan simfi test eder.

Throws:

`java.lang.Exception` - hata verir.

#### testHaberListele

```
public void testHaberListele()  
    throws java.lang.Exception
```

Haberleri listeleyen sınıfı test eder.

Throws:

java.lang.Exception - Hata verir.

#### testHaberTemizle

```
public void testHaberTemizle()  
    throws java.lang.Exception
```

Haberlerin tamamını temizleyen sınıfı test eder.

Throws:

java.lang.Exception - Hata verir.

#### testHaberSil

```
public void testHaberSil()  
    throws java.lang.Exception
```

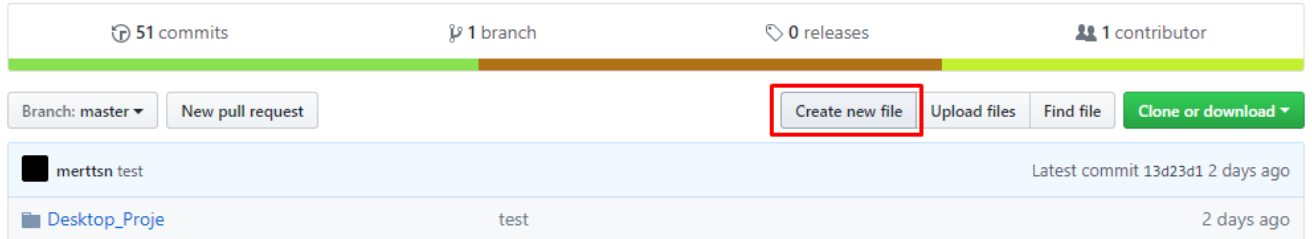
Belirli haberleri silen sınıfı test eder.

Throws:

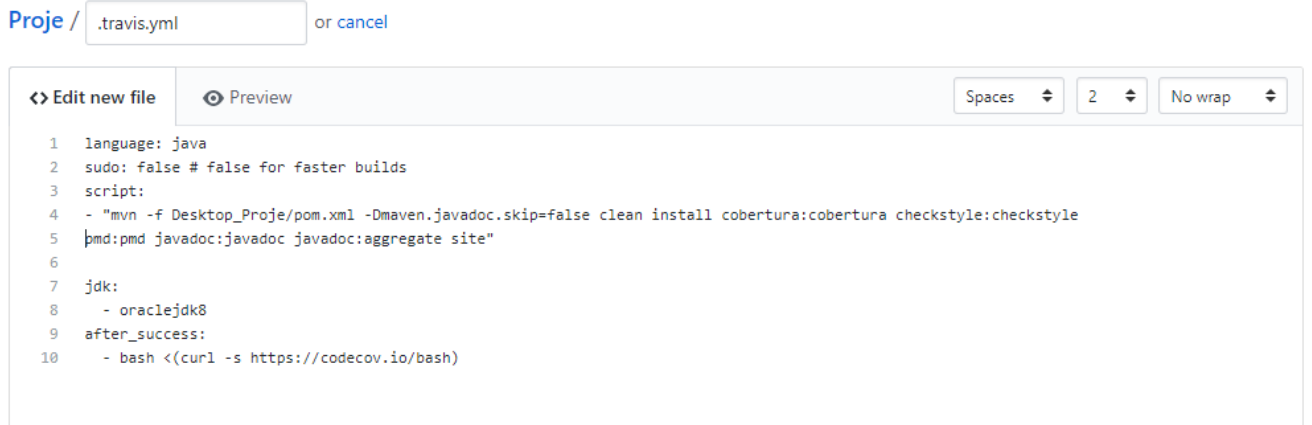
java.lang.Exception - Hata verir.

## 7. TRAVIS & CODECOV

- Öncelikle repository'mize giriyoruz ve "Create new file" a tıklıyoruz.



- Gerekli alanları dolduruyor ve "Commit new file" butonuna tıklıyoruz.



- <https://travis-ci.org/> sitesine giriyoruz ve GitHub hesabımızla giriş yapıyoruz.



## Test and Deploy with Confidence

Easily sync your GitHub projects with Travis CI and you'll be testing your code in minutes!



Sign Up



- Giriş yaptıktan sonra profilimize girip projemizin bulunduğu repository'i aktif ediyoruz.

Travis CI

About Us Blog Status Help

merttsn

Profile Sign Out Sync account

merttsn  
4 repositories  
Token: <img alt="token icon" data-bbox="195 215 205 225"/>

Beta Features

See what's new!

Organizations

You are not currently a member of any organization.

Is an organization missing?  
Review and add your authorized organizations.

We're only showing your public repositories. You can find your private projects on [travis-ci.com](https://travis-ci.com).

1 Flick the repository switch on

2 Add .travis.yml file to your repository

3 Trigger your first build with a git push

Filter repositories

merttsn/deneme

merttsn/MyCourses

merttsn/Proje

merttsn/YAZILIM\_ARACLARI\_REPO

- Anasayfaya geçiş yaptıktan sonra projemizin test edilmeye başlandığını görüyoruz.

master test

#37 queued

Cancel build

Commit 13d23d1

Compare a0acec9..13d23d1

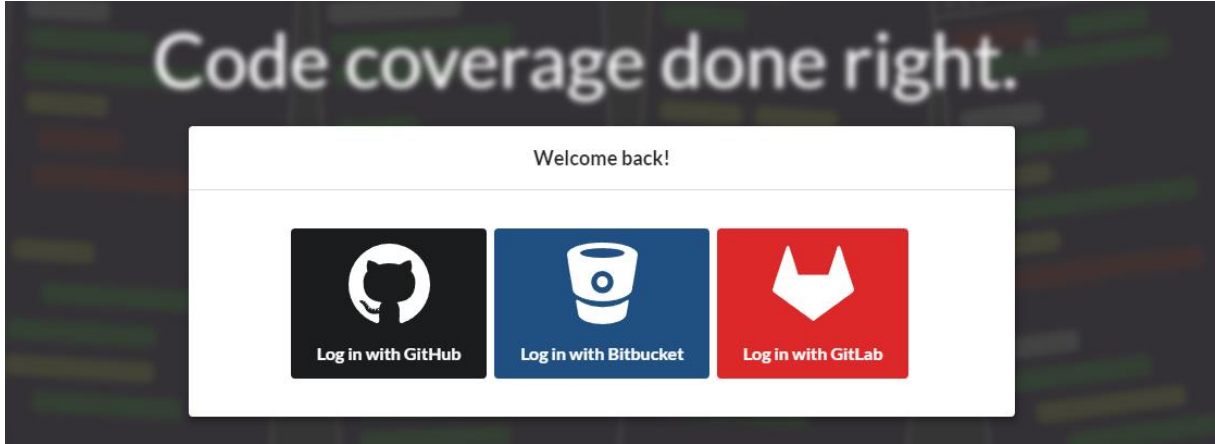
Branch master

merttsn authored and committed

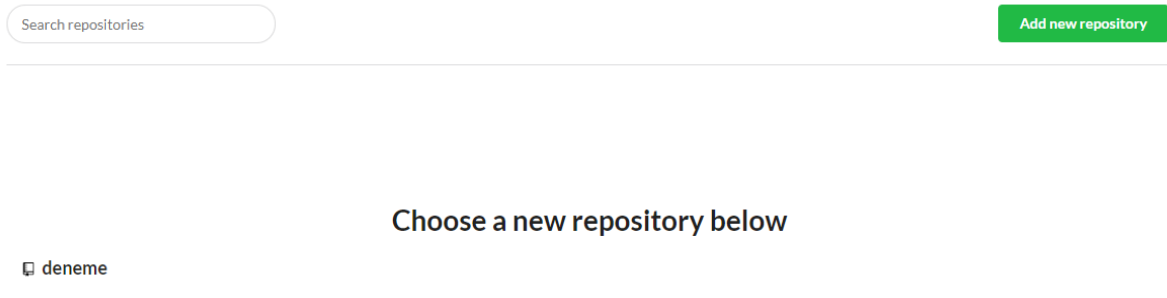
- Eğer sorunsuz bir şekilde ayarlarımızı yaptıysak “Build Success” yazısını görüyoruz.

```
[INFO] Generating "About" report      --- maven-project-info-reports-plugin:2.9:index
[INFO] Generating "Licenses" report   --- maven-project-info-reports-plugin:2.9:license
[INFO] Generating "Plugin Management" report --- maven-project-info-reports-plugin:2.9:plugin-management
[INFO] Generating "Plugins" report      --- maven-project-info-reports-plugin:2.9:plugins
[INFO] Generating "Team" report         --- maven-project-info-reports-plugin:2.9:project-team
[INFO] Generating "Source Code Management" report --- maven-project-info-reports-plugin:2.9:scm
[INFO] Generating "Summary" report      --- maven-project-info-reports-plugin:2.9:summary
[INFO] -----
[INFO] BUILD SUCCESS
```

- <https://codecov.io/> sitesine girdikten sonra “Log In” bölümünden GitHub ile giriş yapıyoruz.

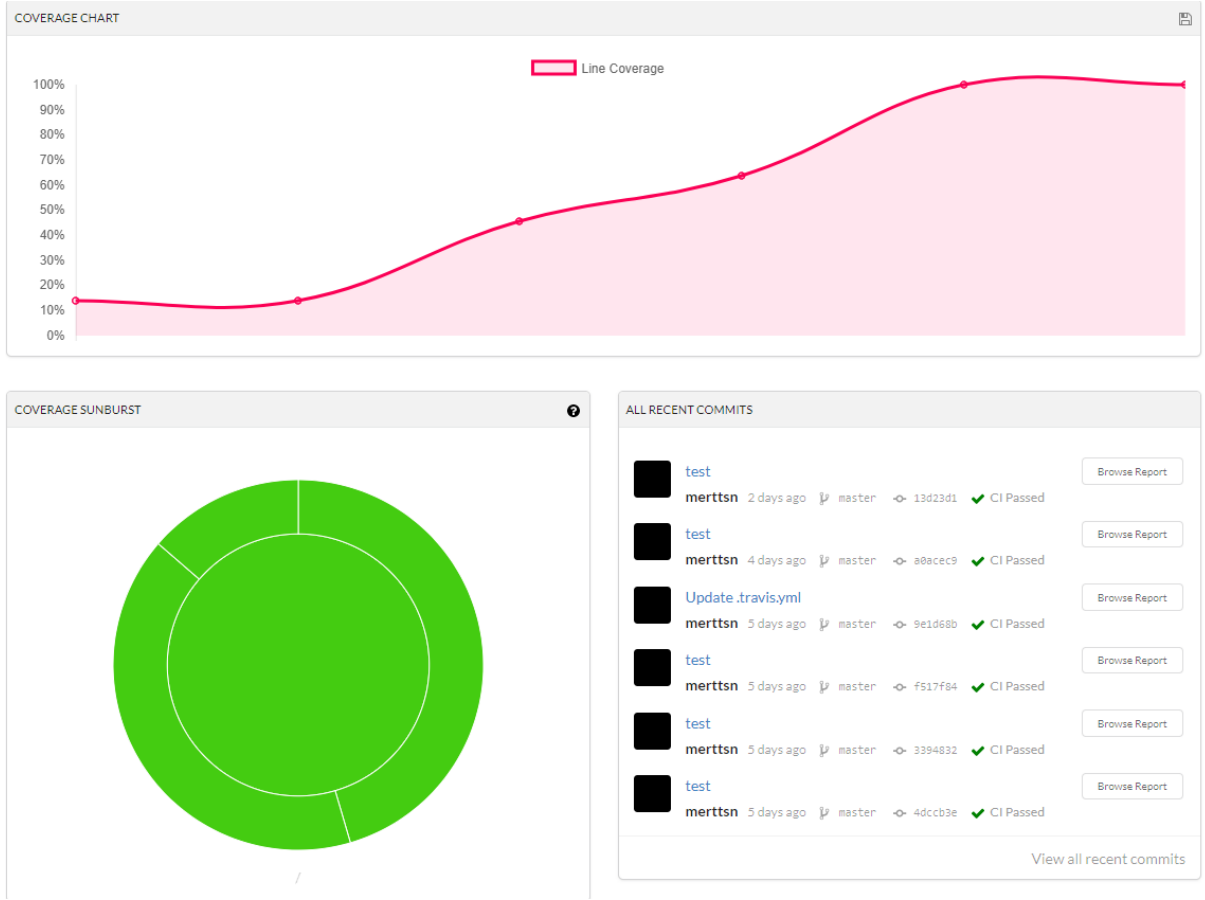


- Giriş yaptıktan sonra profilimizden “Add new repository” tıklıyoruz ve repository’mizi seçiyoruz.



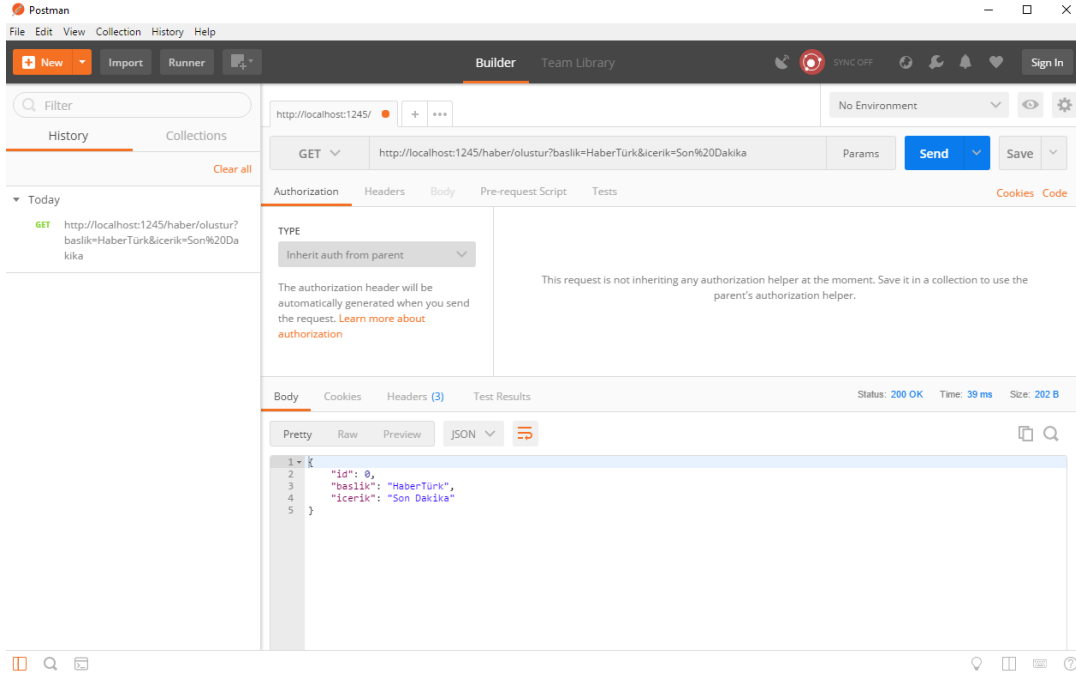
- **Travis ve CodeCov’ı** tetiklemek için projemizde değişiklik yapıp GitHub’a yüklememiz yeterli olacaktır. (bkz. [#GITHUB’A PROJE YÜKLEME](#))

- GitHub'a yeniden yükleme yaptığımızda CodeCov tetiklenecek ve %'lik şeklinde projenizin test durumunu gösterecektir.

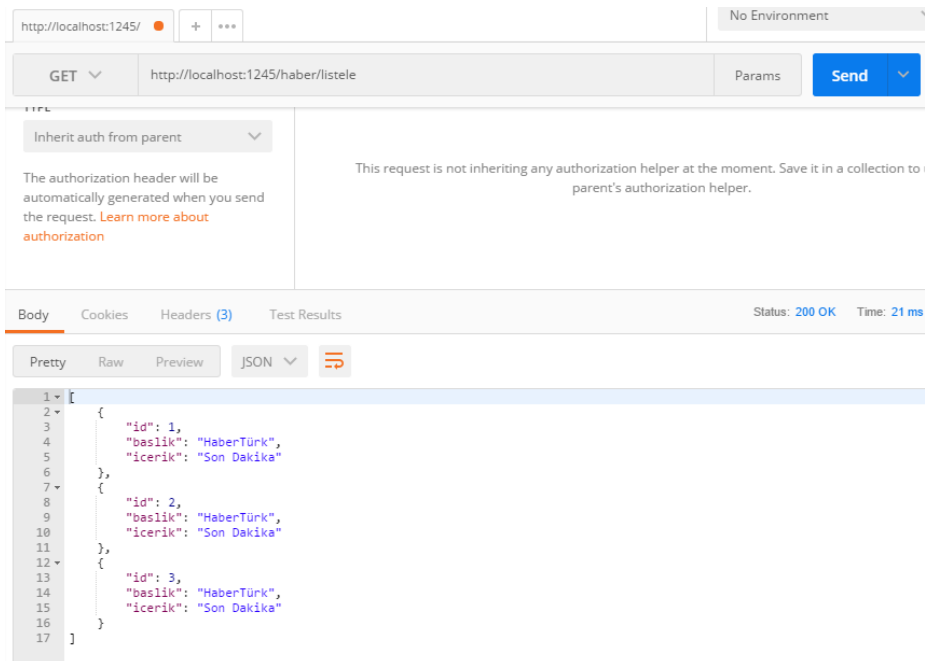


## 8. POSTMAN

- <https://www.getpostman.com/> sitesinden uygulamayı indiriyoruz ve kuruyoruz. Uygulama ekranına **/haber/olustur** url'mizi kopyalıyor ve **"Send"** butonuna basıyoruz. Bu ekleme işlemini birkaç kere tekrarlıyoruz.



- Sırasıyla **/haber/listele** , **/haber/sil** ve **/haber/temizle** servisimizi deniyoruz.



http://localhost:1245/

+

...

No Environment

GET

http://localhost:1245/haber/sil?index=2

Params

Send

Save

Inherit auth from parent

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

This request is not inheriting any authorization helper at the moment. Save it in a collection to use the parent's authorization helper.

Body

Cookies

Headers (3)

Test Results

Status: 200 OK

Time: 20 ms

Size: 146 B

Pretty

Raw

Preview

Text

1

Belirlenen haber kaldırıldı

http://localhost:1245/

+

...

No Environment

GET

http://localhost:1245/haber/temizle

Params

Send

Save

Inherit auth from parent

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

This request is not inheriting any authorization helper at the moment. Save it in a collection to use the parent's authorization helper.

Body

Cookies

Headers (3)

Test Results

Status: 200 OK

Time: 21 ms

Size: 140 B

Pretty

Raw

Preview

Text

1

Tüm haberler temizlendi

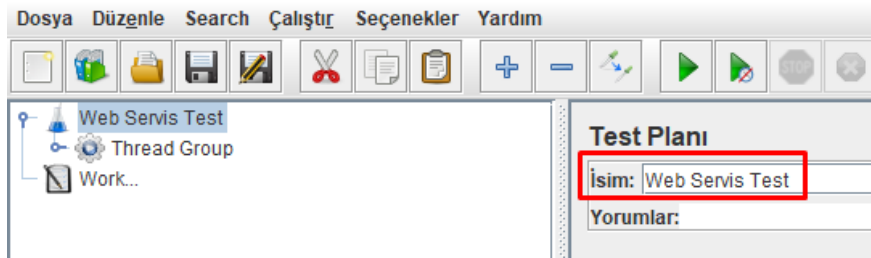
## 9.JMETER

- [http://jmeter.apache.org/download\\_jmeter.cgi](http://jmeter.apache.org/download_jmeter.cgi) adresine girdikten sonra resimde görünen dosyayı indiriyoruz.

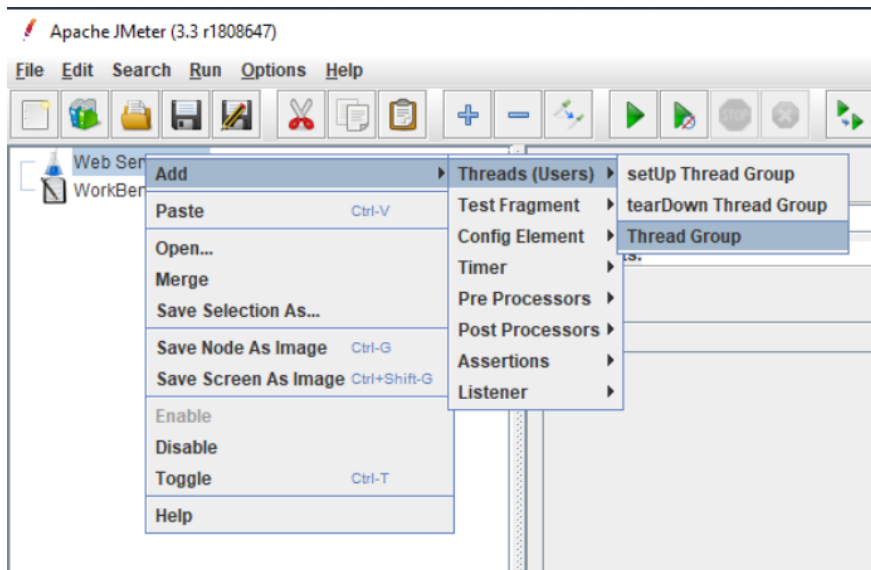
### Binaries

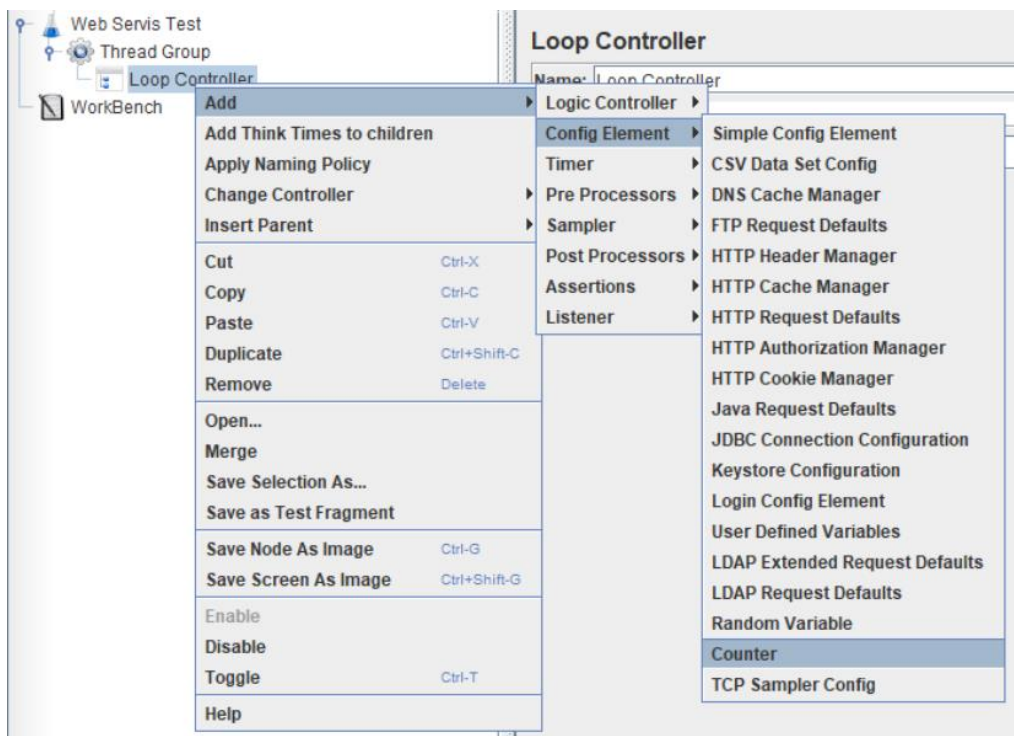
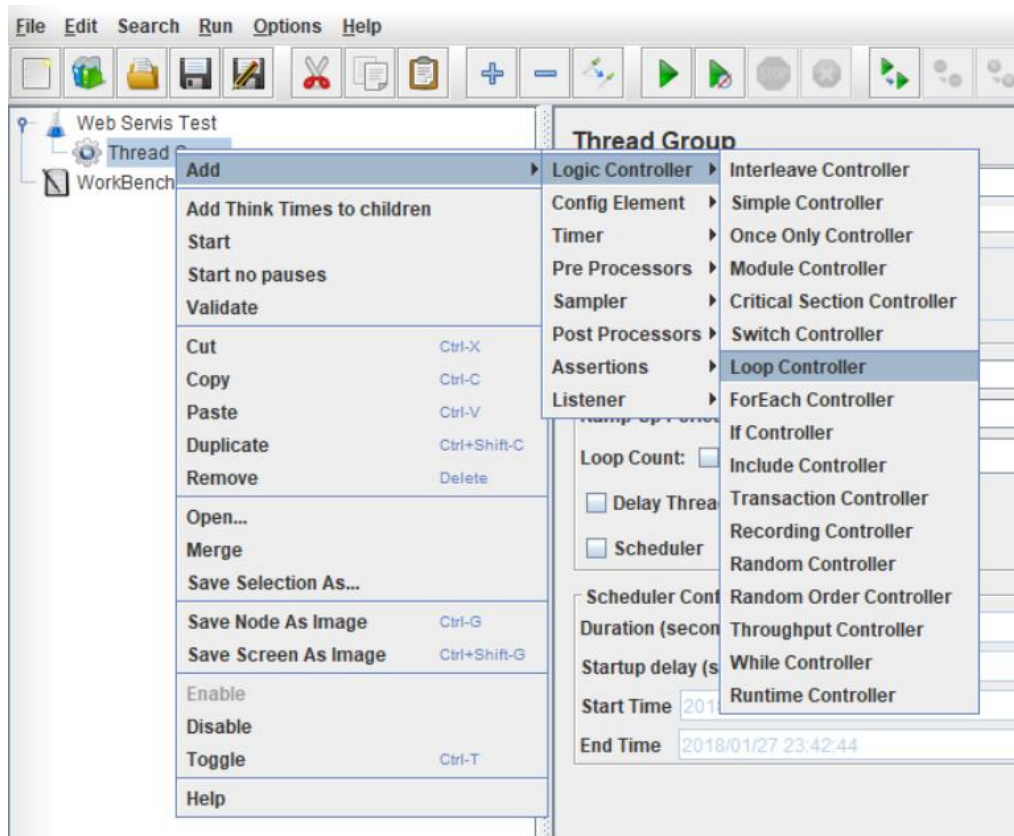
[apache-jmeter-3.3.tgz](#) [md5](#) [sha512](#) [pgp](#)  
[apache-jmeter-3.3.zip](#) [md5](#) [sha512](#) [pgp](#)

- Dosyayı .zip'ten çıkardıktan sonra "apache-jmeter-3.3\bin" yolunu izleyerek "jmeter.bat" dosyasını çalıştırıyoruz.
- "Test Plan" yazan kısmı "Web Servis Test" olarak değiştiriyoruz.



- Sonrasında şu adımları izliyoruz.





Web Servis Test

Thread Group

Loop Controller

Counter

WorkBench

### Counter

Name: Counter

Comments:

Starting value 2

Increment 2

Maximum value

Number format

Reference Name c1

☐ Track counter independently for each user

☐ Reset counter on each Thread Group Iteration

Web Servis Test

Thread Group

Loop Controller

Counter

WorkBench

### Loop Controller

Name: Loop Controller

Starting value 1

Add

Add Think Times to children

Apply Naming Policy

Change Controller

Insert Parent

Cut Ctrl-X

Copy Ctrl-C

Paste Ctrl-V

Duplicate Ctrl+Shift-C

Remove Delete

Open...

Merge

Save Selection As...

Save as Test Fragment

Save Node As Image Ctrl-G

Save Screen As Image Ctrl+Shift-G

Enable

Disable

Toggle Ctrl-T

Help

Logic Controller

Config Element

Timer

Pre Processors

Sampler

Post Processors

Assertions

Listener

AJP/1.3 Sampler

Debug Sampler

BeanShell Sampler

Mail Reader Sampler

FTP Request

HTTP Request

Java Request

JDBC Request

JMS Subscriber

JMS Point-to-Point

JMS Publisher

JSR223 Sampler

JUnit Request

LDAP Extended Request

LDAP Request

Access Log Sampler

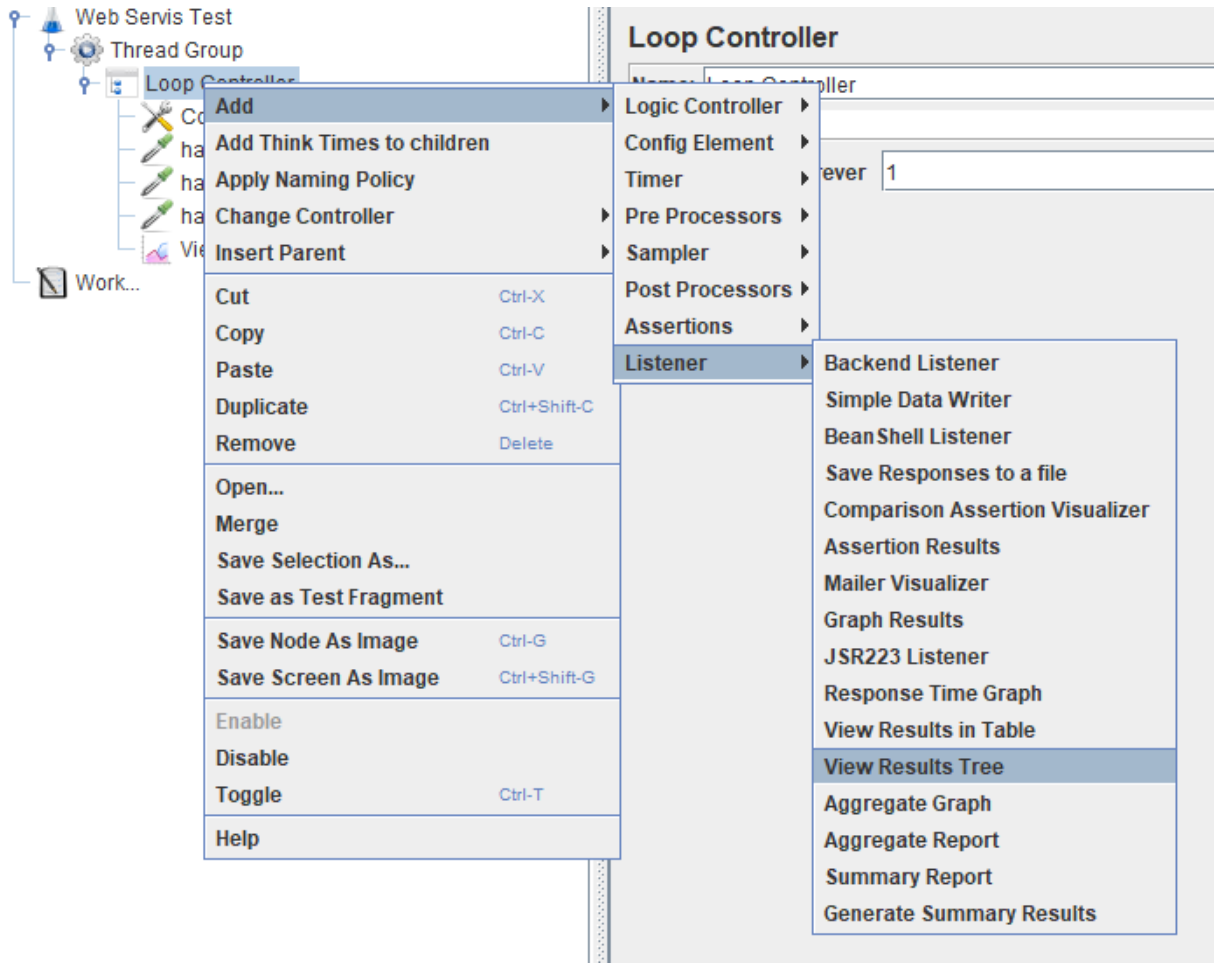
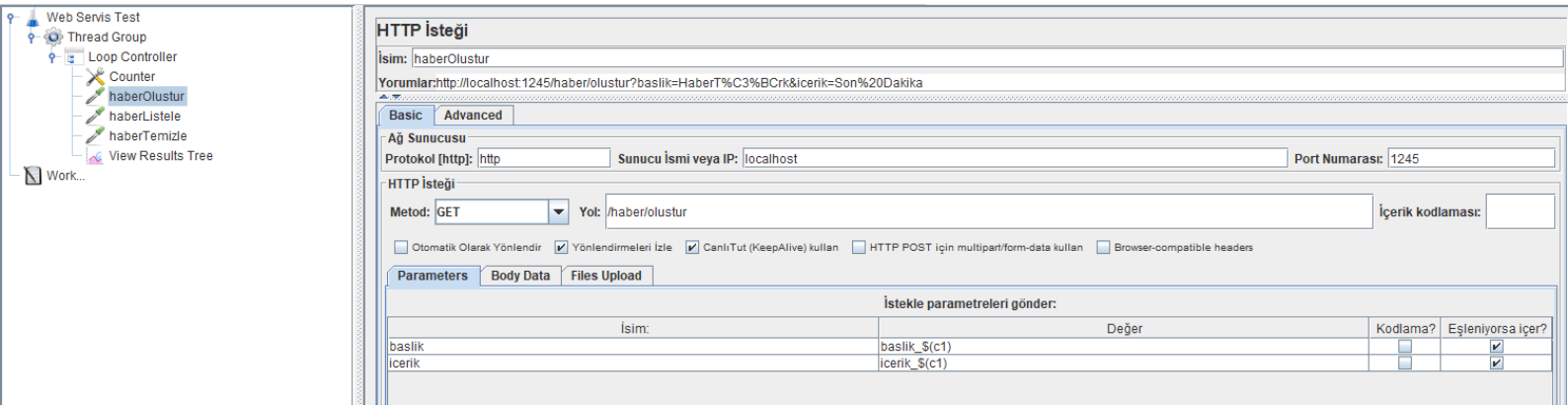
OS Process Sampler

SMTP Sampler

TCP Sampler

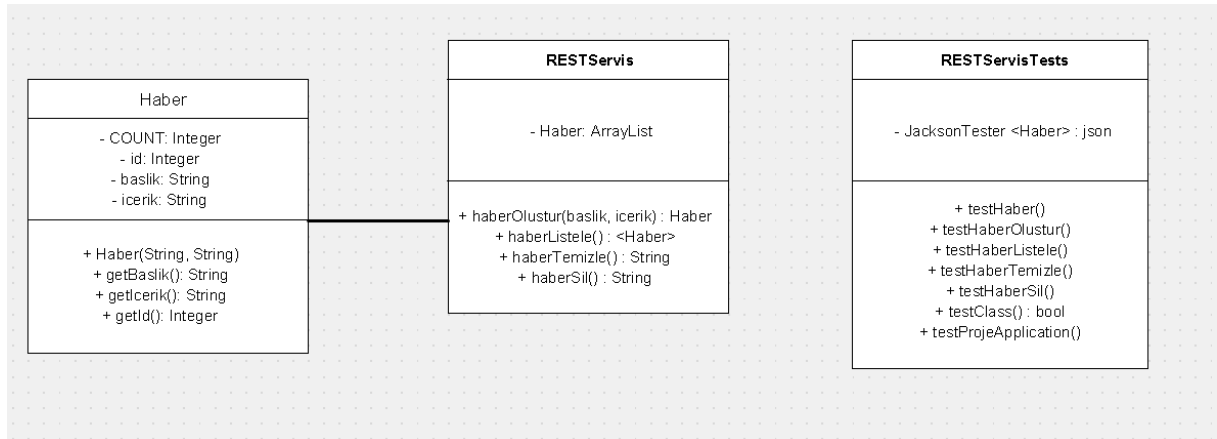
Test Action







## 10. UML DİAGRAM



- **Haber Class**

Özellikleri tanımladığımız bölümde “COUNT” ve “Id” değişkenlerimizi integer olarak tanımlıyoruz. “baslik” ve “icerik” değişkenlerini ise String olarak tanımlıyoruz. Dördü de private olduğu için başlarına “-” koyuyoruz.

İşlevleri tanımladığımız bölümde “Haber” “getBaslik” “getIcerik” değişkenlerimizi String olarak tanımlıyoruz. “getId” değişkenimizi ise Integer olarak tanımlıyoruz. Public oldukları için başlarına “+” ekliyoruz.

- **RESTServis Class**

Özellikleri tanımladığımız bölümde sadece ArrayList olan “Haber” değişkenimiz bulunuyor. Private olduğu için başına “-” koyuyoruz.

İşlevleri tanımladığımız bölümde “haberOlustur” “Haber” üzerinde işlem yaptığı için metodunu ona göre ayarlıyoruz. “haberListele” List<Haber> üzerinden işlem yaptığı için metodunu ayarlıyoruz. “haberTemizle” ve “haberSil” ise String tipinde kalıyor. Public oldukları için hepsi “+” alıyor.

- **RESTServisTests Class**

Özellikleri tanımladığımız bölümde testlerimizi yapmamızı sağlayan JacksonTester bulunuyor. Tipini json yapıyor, private olduğu için başına “-” ekliyoruz.

İşlevleri tanımladığımız bölümde “testClass” bool geri kalan void olduğu için boş bırakıyor ve başlarına public oldukları için “+” koyarak bitiriyoruz.