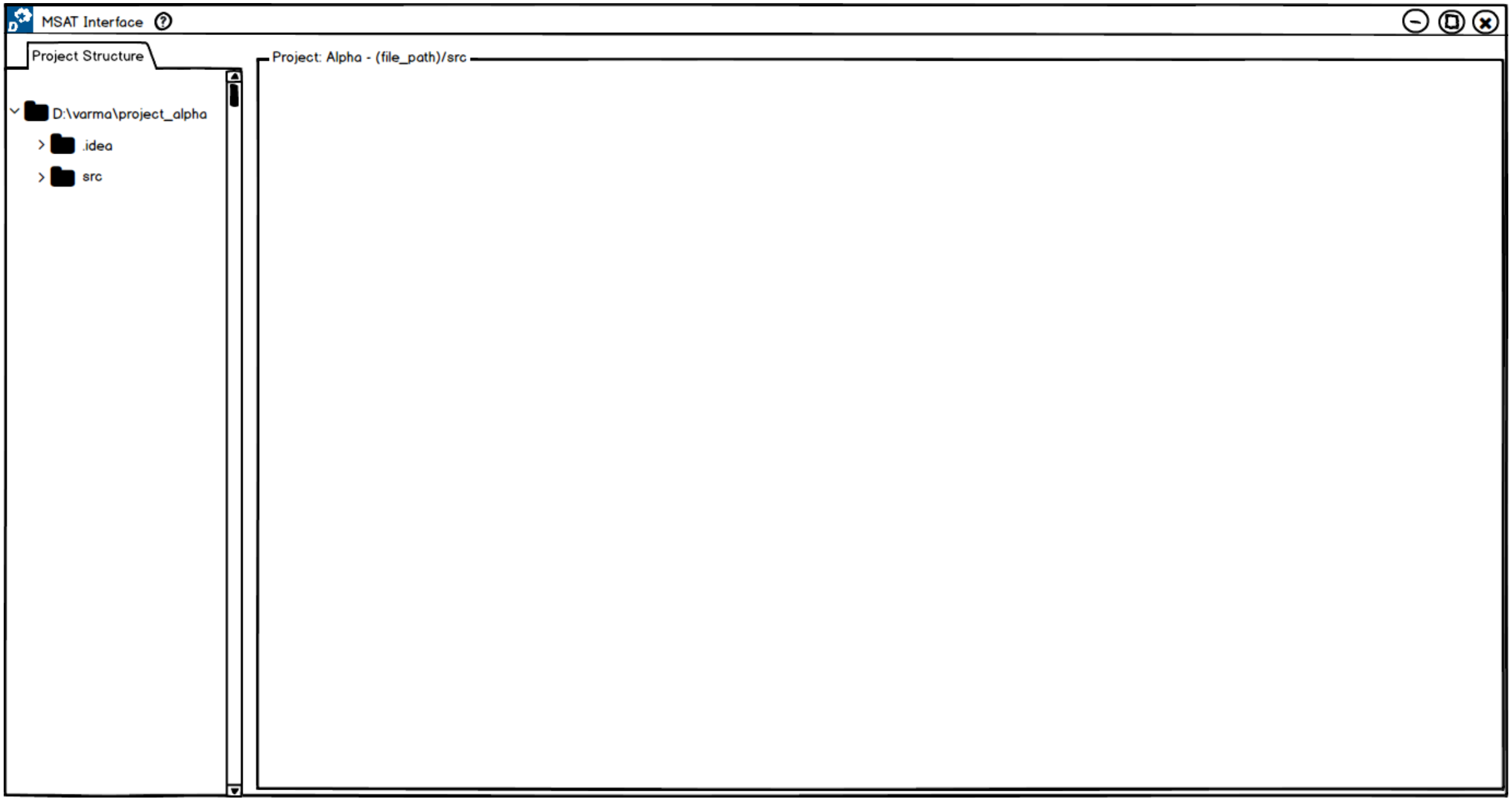
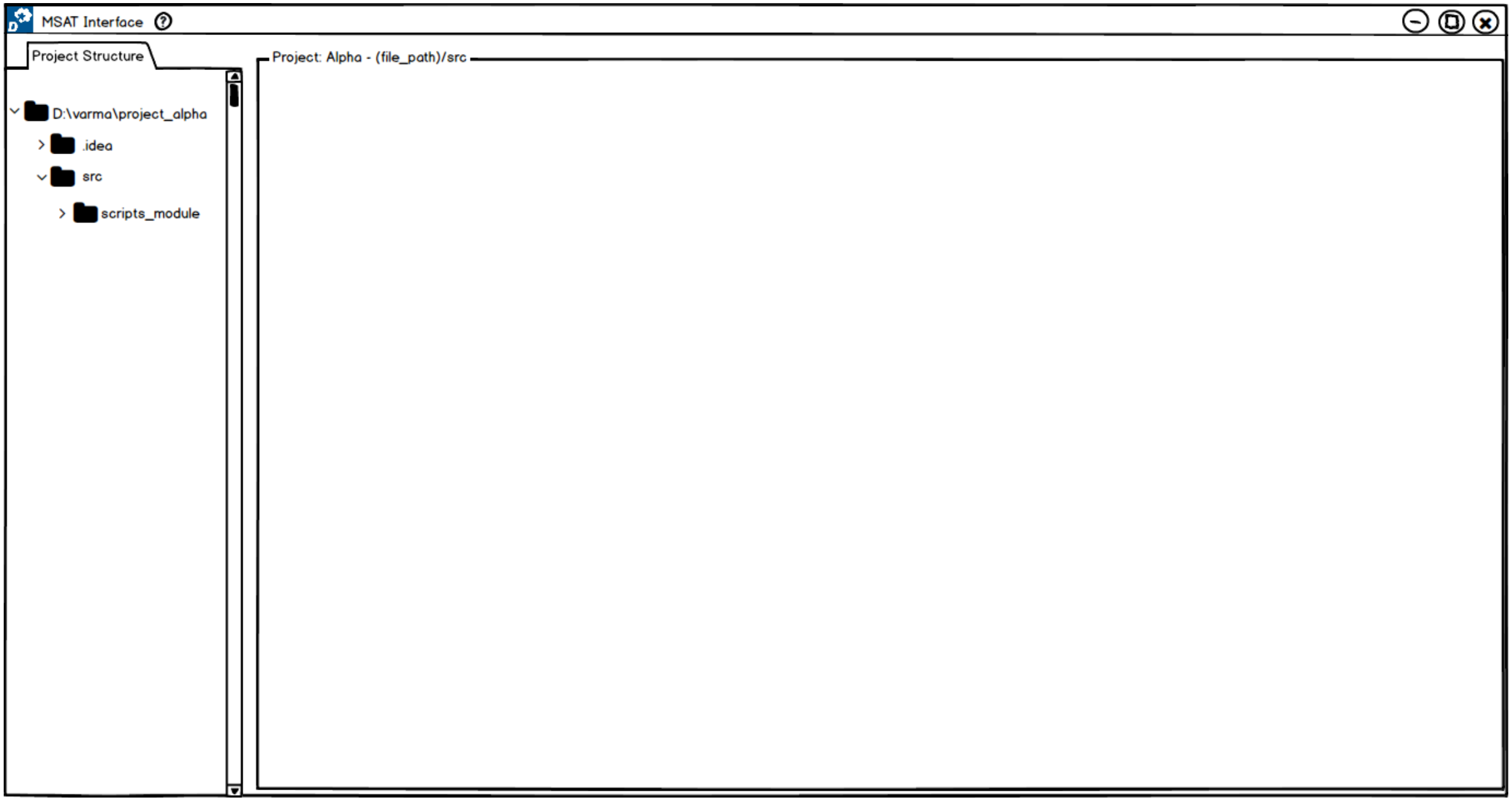
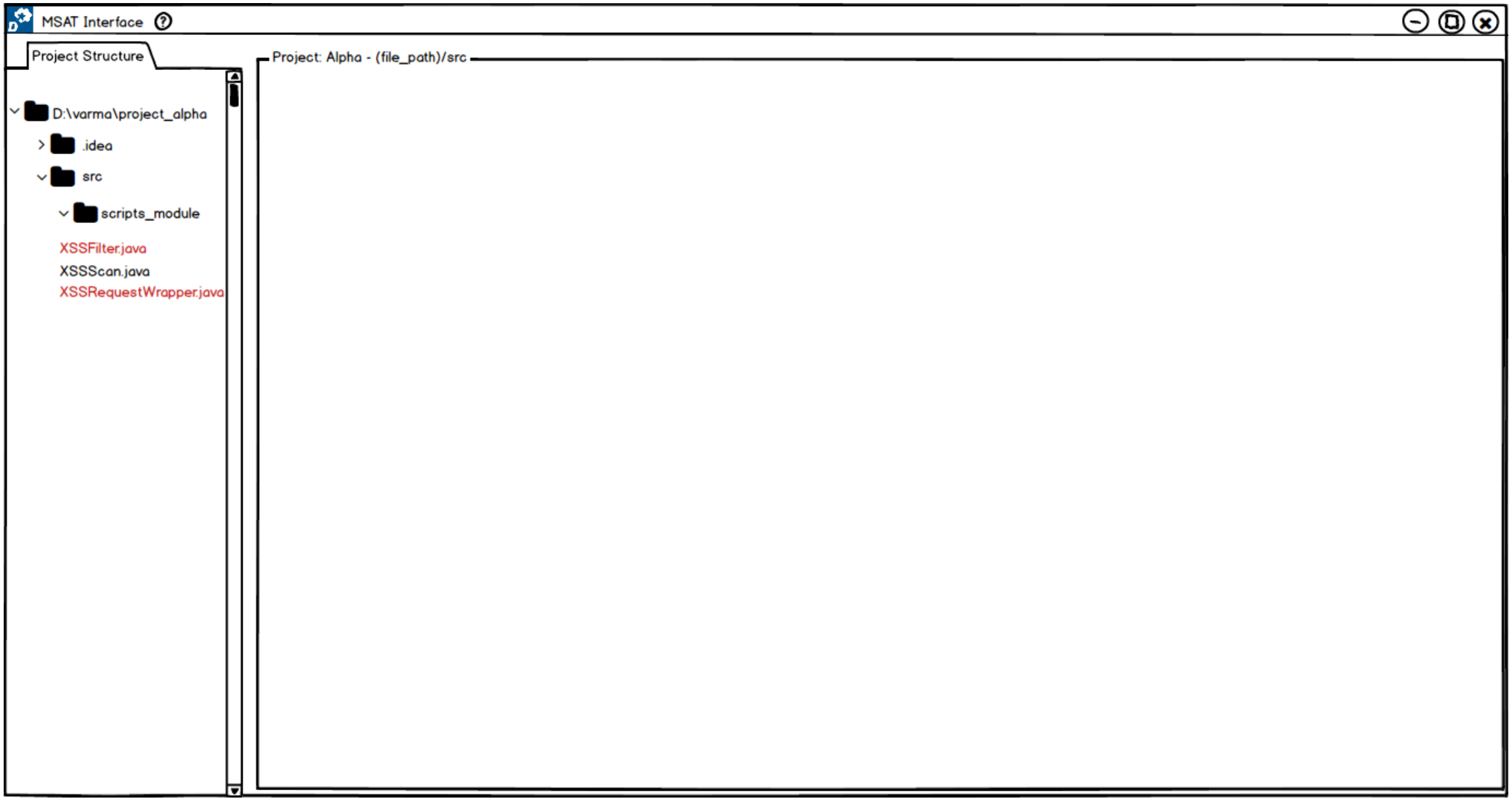

MSAT Interface
Project Structure
D:\varma\project_alpha
.idea
src
Project: Alpha - (file_path)/src


MSAT Interface
Project Structure
Project: Alpha - (file_path)/src
D:\varma\project_alpha
.idea
src
scripts_module


MSAT Interface
Project Structure
Project: Alpha - (file_path)/src
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java

MSAT Interface

Project Structure

- D:\varma\project_alpha
  - .idea
  - src
    - scripts_module
      - XSSFilter.java
      - XSSScan.java
      - XSSRequestWrapper.java

Fixed

Project: Alpha - (file_path)/XSSFilter.java

```
import java.util.regex.Pattern;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletRequestWrapper;


public class XSSFilter implements Filter {

    @Override
    public void init(FilterConfig filterConfig) throws ServletException {
    }

    @Override
    public void destroy() {
    }

    @Override
    public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
        throws IOException, ServletException {
        chain.doFilter(new XSSRequestWrapper((HttpServletRequest) request), response);
    }

}
```

MSAT Interface

Project Structure

- D:\varma\project_alpha
  - .idea
  - src
    - scripts_module
      - XSSFilter.java
      - XSSScan.java
      - XSSRequestWrapper.java

Fixed

Project: Alpha - (file_path)/XSSFilter.java

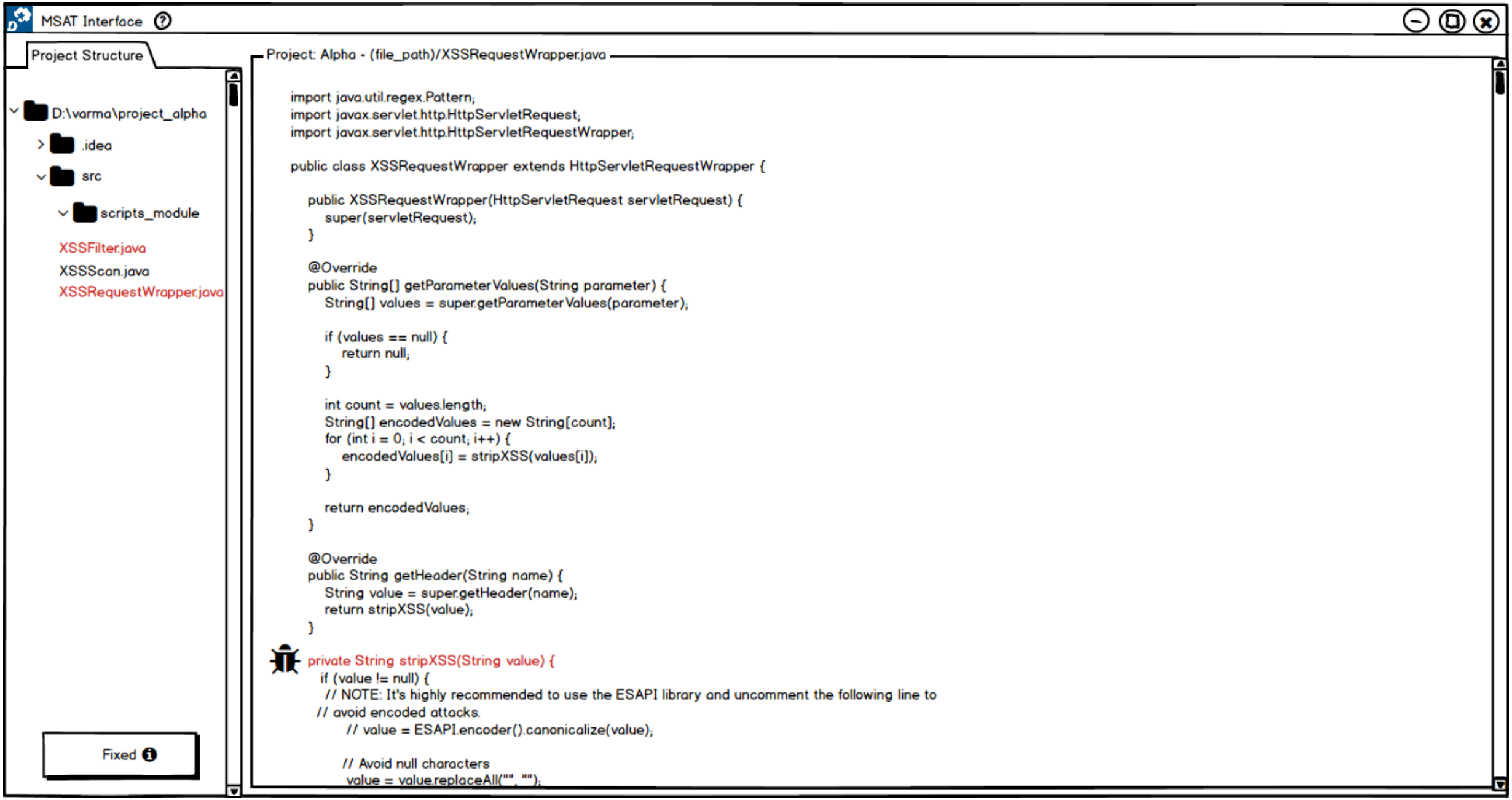
```
import java.util.regex.Pattern;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletRequestWrapper;


public class XSSFilter implements Filter {

    @Override
    public void init(FilterConfig filterConfig) throws ServletException {
    }

    @
    p
    }

    @

    }
}
```

XSS: Anti cross-site scripting filter (XSS_CONFIG)

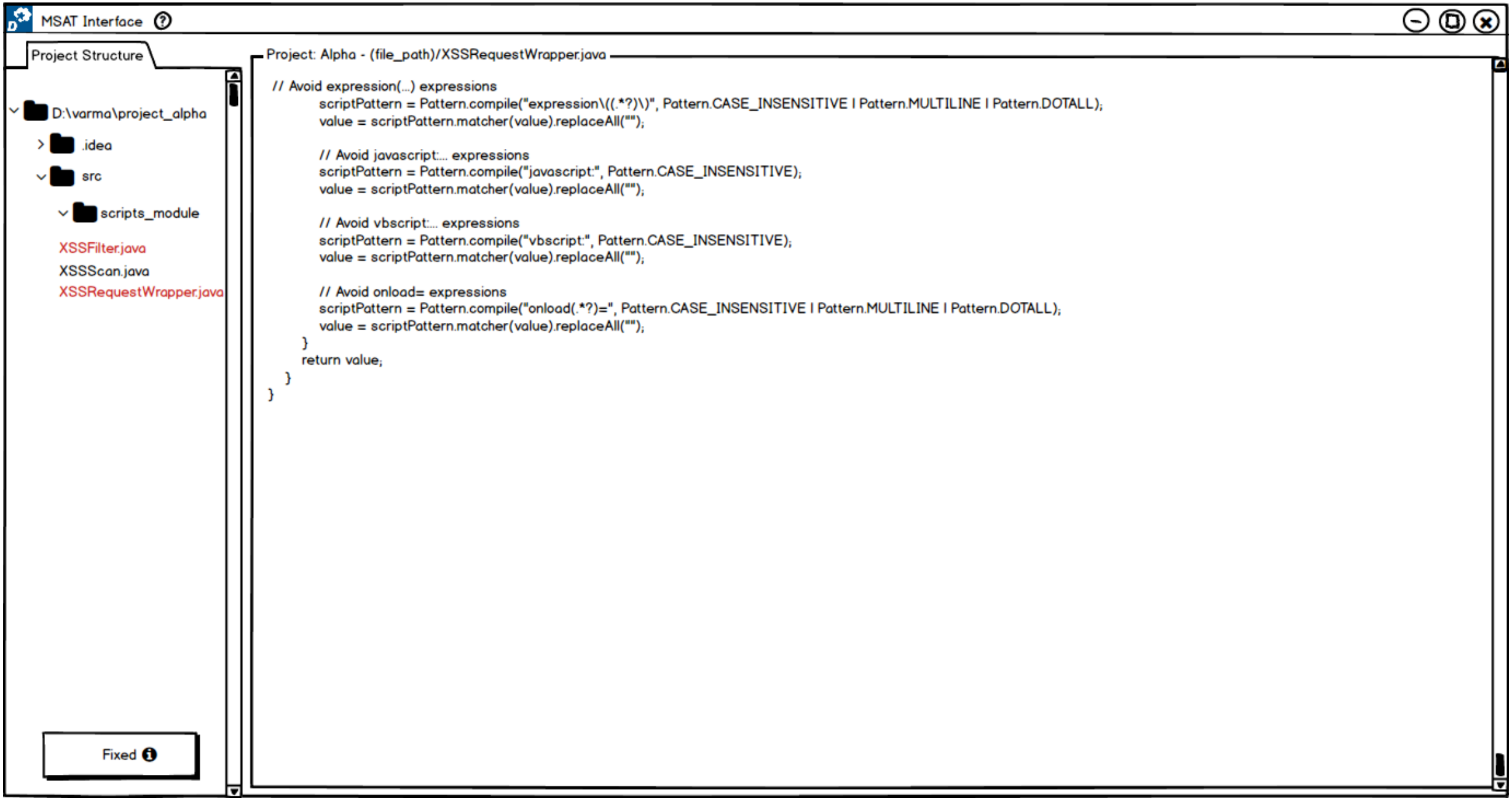
Wrap the HTTP request object in a specialized

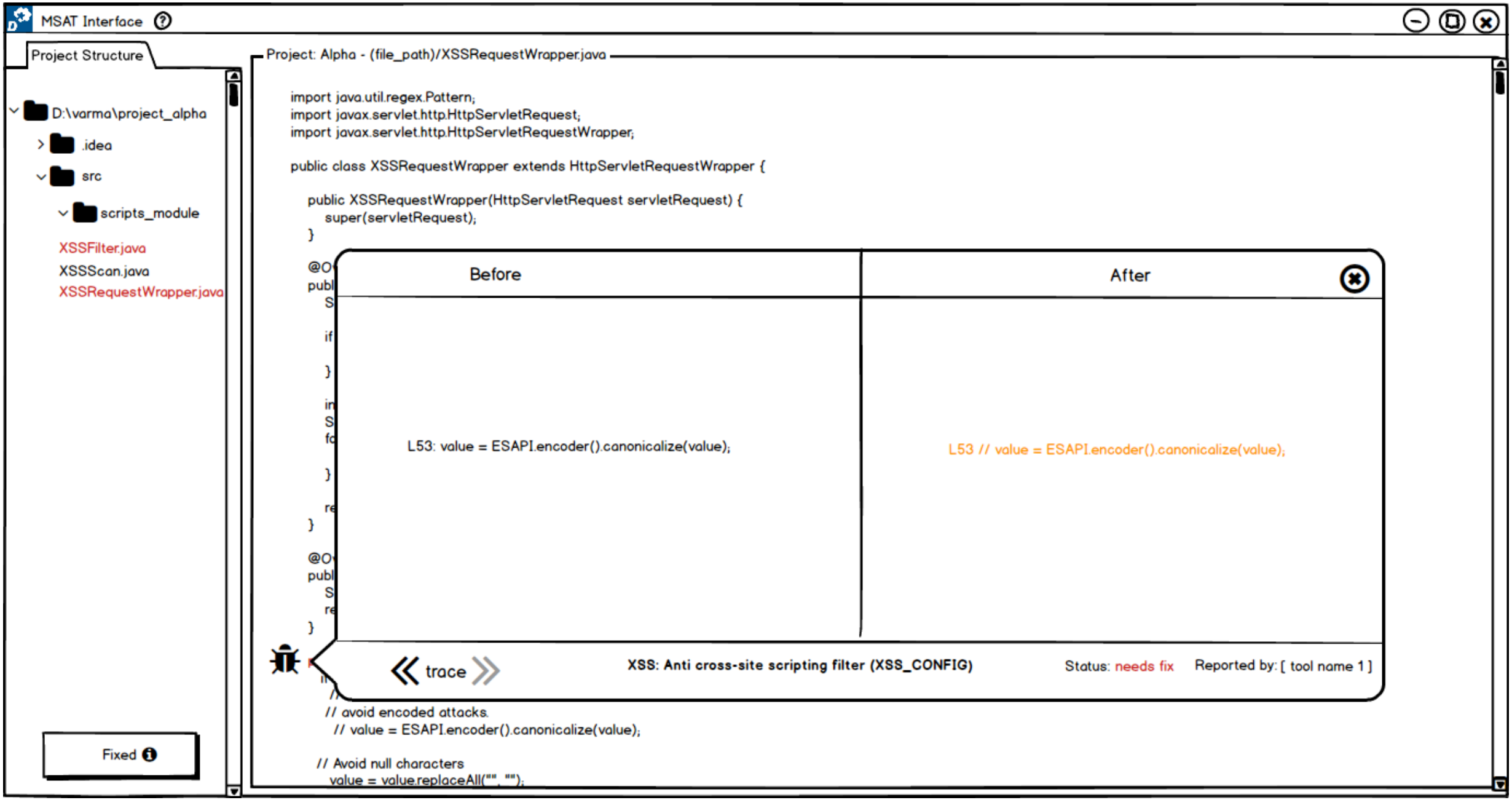
HttpServletRequestWrapper that will perform filtering.

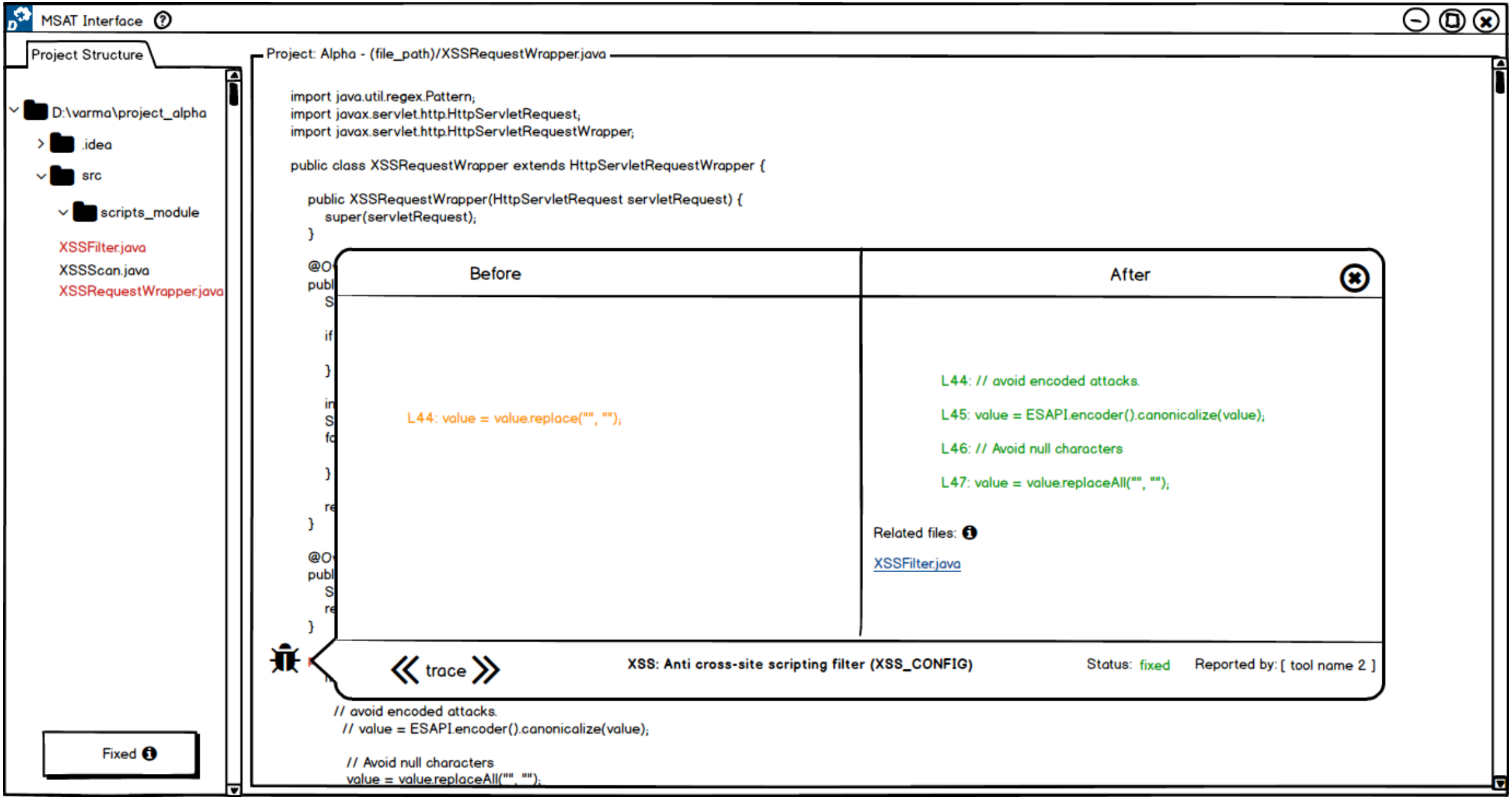
Reported by:

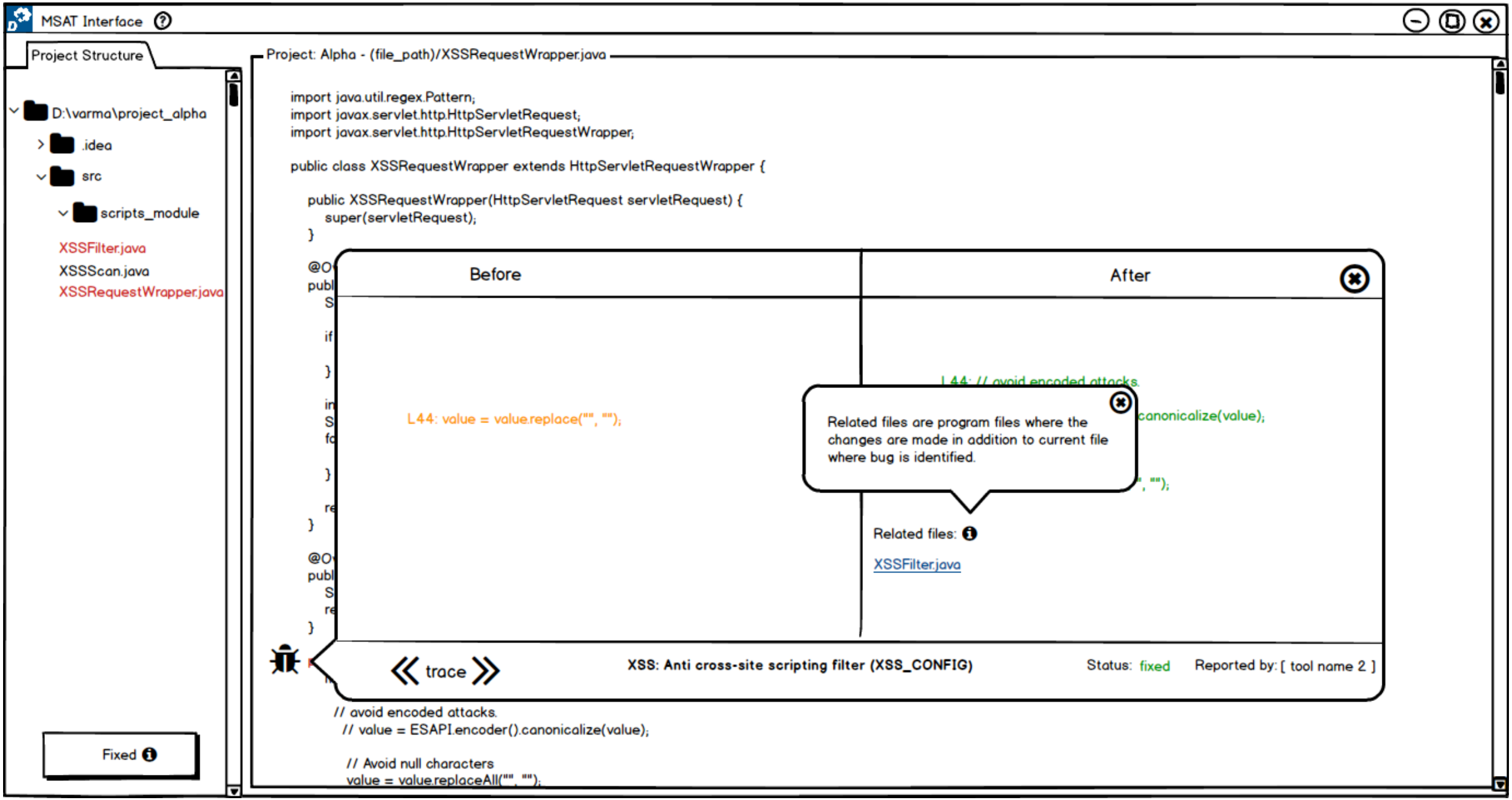

MSAT Interface
Project Structure
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java
Fixed
Project: Alpha - (file_path)/XSSRequestWrapper.java
import java.util.regex.Pattern;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletRequestWrapper;
public class XSSRequestWrapper extends HttpServletRequestWrapper {
public XSSRequestWrapper(HttpServletRequest servletRequest) {
super(servletRequest);
}
@Override
public String[] getParameterValues(String parameter) {
String[] values = super.getParameterValues(parameter);
if (values == null) {
return null;
}
int count = values.length;
String[] encodedValues = new String[count];
for (int i = 0; i < count; i++) {
encodedValues[i] = stripXSS(values[i]);
}
return encodedValues;
}
@Override
public String getHeader(String name) {
String value = super.getHeader(name);
return stripXSS(value);
}
private String stripXSS(String value) {
if (value != null) {
// NOTE: It's highly recommended to use the ESAPI library and uncomment the following line to
// avoid encoded attacks.
// value = ESAPI.encoder().canonicalize(value);
// Avoid null characters
value = value.replaceAll("", "");

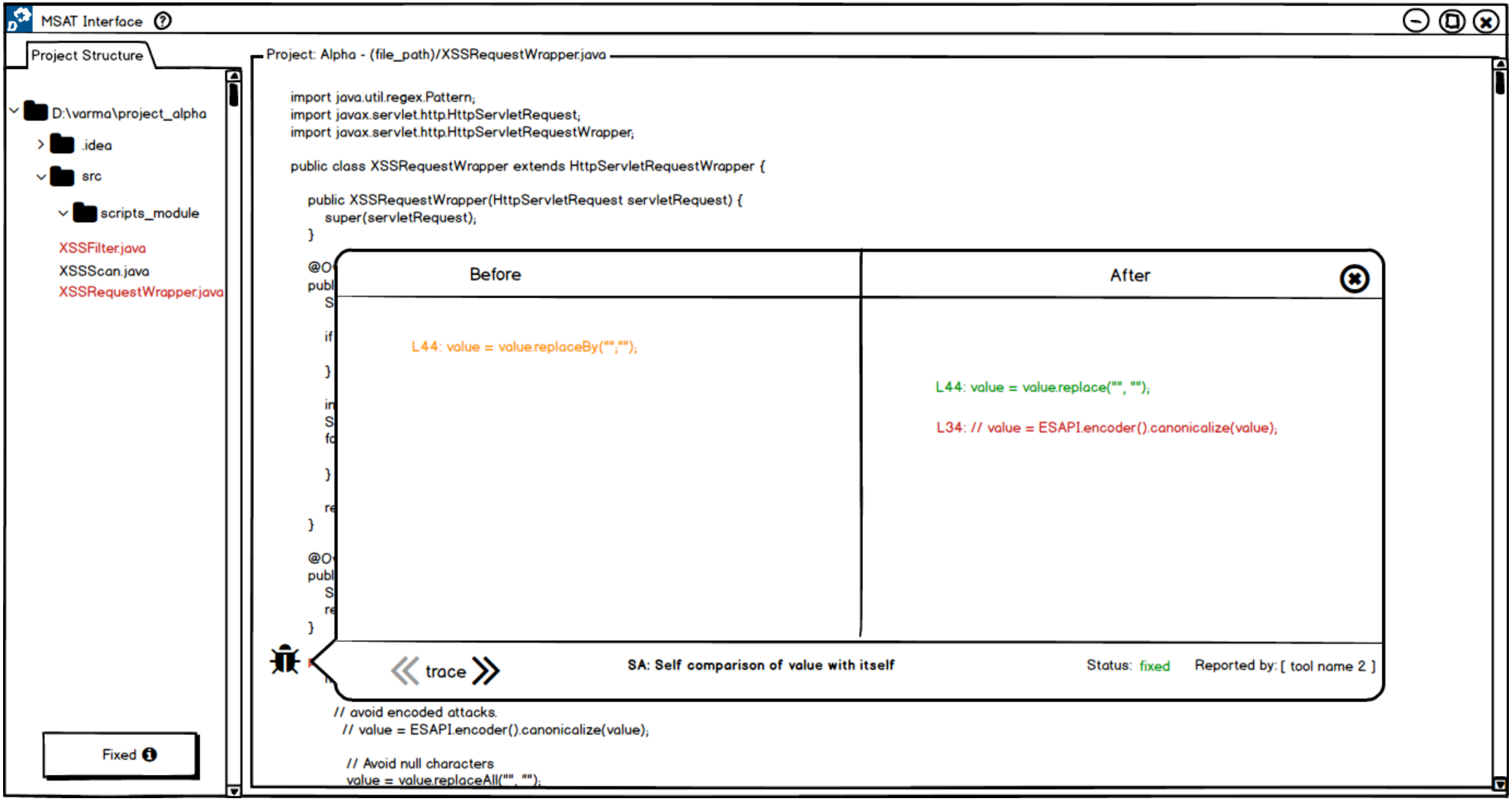

MSAT Interface
Project Structure
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java
Fixed
Project: Alpha - (file_path)/XSSRequestWrapper.java
return encodedValues;
}
@Override
public String getParameter(String parameter) {
String value = super.getParameter(parameter);
return stripXSS(value);
}
@Override
public String getHeader(String name) {
String value = super.getHeader(name);
return stripXSS(value);
}
private String stripXSS(String value) {
if (value != null) {
// NOTE: It's highly recommended to use the ESAPI library and uncomment the following line to
// avoid encoded attacks.
// value = ESAPI.encoder().canonicalize(value);
// Avoid null characters
value = value.replaceAll("", "");
// Avoid anything between script tags
Pattern scriptPattern = Pattern.compile("<script>(.*?)</script>", Pattern.CASE_INSENSITIVE);
value = scriptPattern.matcher(value).replaceAll("");
// Avoid anything in a src='...' type of expression
scriptPattern = Pattern.compile("src[r]=[]\\'(.*?)\\'", Pattern.CASE_INSENSITIVE | Pattern.MULTILINE | Pattern.DOTALL);
value = scriptPattern.matcher(value).replaceAll("");
scriptPattern = Pattern.compile("src[]*=[\]*\\"(.*?)\\"", Pattern.CASE_INSENSITIVE | Pattern.MULTILINE | Pattern.DOTALL);
value = scriptPattern.matcher(value).replaceAll("");
// Remove any lonesome </script> tag
scriptPattern = Pattern.compile("</script>", Pattern.CASE_INSENSITIVE);
value = scriptPattern.matcher(value).replaceAll("");


MSAT Interface
Project Structure
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java
Fixed
Project: Alpha - (file_path)/XSSRequestWrapper.java
// Avoid expression(...) expressions
scriptPattern = Pattern.compile("expression\((.*?)\)", Pattern.CASE_INSENSITIVE | Pattern.MULTILINE | Pattern.DOTALL);
value = scriptPattern.matcher(value).replaceAll("");
// Avoid javascript:... expressions
scriptPattern = Pattern.compile("javascript:", Pattern.CASE_INSENSITIVE);
value = scriptPattern.matcher(value).replaceAll("");
// Avoid vbscript:... expressions
scriptPattern = Pattern.compile("vbscript:", Pattern.CASE_INSENSITIVE);
value = scriptPattern.matcher(value).replaceAll("");
// Avoid onload= expressions
scriptPattern = Pattern.compile("onload(.*?)=", Pattern.CASE_INSENSITIVE | Pattern.MULTILINE | Pattern.DOTALL);
value = scriptPattern.matcher(value).replaceAll("");
}
return value;
}
}


MSAT Interface
Project Structure
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java
Fixed
Project: Alpha - (file_path)/XSSRequestWrapper.java
import java.util.regex.Pattern;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletRequestWrapper;
public class XSSRequestWrapper extends HttpServletRequestWrapper {
public XSSRequestWrapper(HttpServletRequest servletRequest) {
super(servletRequest);
}
// avoid encoded attacks.
// value = ESAPI.encoder().canonicalize(value);
// Avoid null characters
value = value.replaceAll("", "");
Before
After
L53: value = ESAPI.encoder().canonicalize(value);
L53 // value = ESAPI.encoder().canonicalize(value);
trace
XSS: Anti cross-site scripting filter (XSS_CONFIG)
Status: needs fix
Reported by: [tool name 1]


MSAT Interface
Project Structure
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java
Fixed
Project: Alpha - (file_path)/XSSRequestWrapper.java
import java.util.regex.Pattern;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletRequestWrapper;
public class XSSRequestWrapper extends HttpServletRequestWrapper {
public XSSRequestWrapper(HttpServletRequest servletRequest) {
super(servletRequest);
}
Before
After
L44: value = value.replace("", "");
L44: // avoid encoded attacks.
L45: value = ESAPI.encoder().canonicalize(value);
L46: // Avoid null characters
L47: value = value.replaceAll("", "");
Related files:
XSSFilter.java
trace
XSS: Anti cross-site scripting filter (XSS_CONFIG)
Status: fixed
Reported by: [tool name 2]
// avoid encoded attacks.
// value = ESAPI.encoder().canonicalize(value);
// Avoid null characters
value = value.replaceAll("", "");


MSAT Interface
Project Structure
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java
Fixed
Project: Alpha - (file_path)/XSSRequestWrapper.java
import java.util.regex.Pattern;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletRequestWrapper;
public class XSSRequestWrapper extends HttpServletRequestWrapper {
public XSSRequestWrapper(HttpServletRequest servletRequest) {
super(servletRequest);
}
Before
After
L44: value = value.replace("", "");
L44: // avoid encoded attacks.
canonicalize(value);
", "");
Related files are program files where the changes are made in addition to current file where bug is identified.
Related files:
XSSFilter.java
trace
XSS: Anti cross-site scripting filter (XSS_CONFIG)
Status: fixed
Reported by: [tool name 2]
// avoid encoded attacks.
// value = ESAPI.encoder().canonicalize(value);
// Avoid null characters
value = value.replaceAll("", "");


MSAT Interface
Project Structure
Project: Alpha - (file_path)/XSSRequestWrapper.java
D:\varma\project_alpha
.idea
src
scripts_module
XSSFilter.java
XSSScan.java
XSSRequestWrapper.java
Fixed
import java.util.regex.Pattern;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletRequestWrapper;
public class XSSRequestWrapper extends HttpServletRequestWrapper {
public XSSRequestWrapper(HttpServletRequest servletRequest) {
super(servletRequest);
}
Before
After
L44: value = value.replaceBy("","");
L44: value = value.replace("", "");
L34: // value = ESAPI.encoder().canonicalize(value);
trace
SA: Self comparison of value with itself
Status: fixed
Reported by: [tool name 2]
// avoid encoded attacks.
// value = ESAPI.encoder().canonicalize(value);
// Avoid null characters
value = value.replaceAll("", "");