

JS JSP-Based_Feedback_Management_System Version control Current File Terminal Copilot Windows PowerShell Copyright (C) Microsoft Corporation. All rights reserved. Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\Deepa\IdeaProjects\JSP-Based_Feedback_Management_System> mvn -q test
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::staticFieldBase has been called by com.google.inject.internal.aop.HiddenClassDefiner (file:/C:/Program%20Files/apache-maven-3.9.11/lib/guice-5.1.0-classes.jar)
WARNING: Please consider reporting this to the maintainers of class com.google.inject.internal.aop.HiddenClassDefiner
WARNING: sun.misc.Unsafe::staticFieldBase will be removed in a future release
PS C:\Users\Deepa\IdeaProjects\JSP-Based_Feedback_Management_System> mvn -q test
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::staticFieldBase has been called by com.google.inject.internal.aop.HiddenClassDefiner (file:/C:/Program%20Files/apache-maven-3.9.11/lib/guice-5.1.0-classes.jar)
WARNING: Please consider reporting this to the maintainers of class com.google.inject.internal.aop.HiddenClassDefiner
WARNING: sun.misc.Unsafe::staticFieldBase will be removed in a future release
PS C:\Users\Deepa\IdeaProjects\JSP-Based_Feedback_Management_System> mvn test
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::staticFieldBase has been called by com.google.inject.internal.aop.HiddenClassDefiner (file:/C:/Program%20Files/apache-maven-3.9.11/lib/guice-5.1.0-classes.jar)
WARNING: Please consider reporting this to the maintainers of class com.google.inject.internal.aop.HiddenClassDefiner
WARNING: sun.misc.Unsafe::staticFieldBase will be removed in a future release
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.example:JSP-Based_Feedback_Management_System >-----
[INFO] Building JSP-Based_Feedback_Management_System Maven Webapp 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ war ]-----
[INFO]
[INFO] --- resources:3.3.1:resources (default-resources) @ JSP-Based_Feedback_Management_System ---
[INFO] Copying 0 resource from src\main\resources to target\classes
[INFO]
[INFO] --- compiler:3.8.1:compile (default-compile) @ JSP-Based_Feedback_Management_System ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- resources:3.3.1:testResources (default-testResources) @ JSP-Based_Feedback_Management_System ---
```

JSP-Based_Feedback_Management_System > src > main > webapp > WEB-INF > </> webxml 7:1 CRLF UTF-8 4 spaces 21°C Sunny ENG IN 11:12 14-11-2025

The screenshot shows an IDE interface with the following details:

- Project:** JSP-Based_Feedback_Management_System
- File:** web.xml
- Content:** The code defines a web application with a display name "Archetype Created Web Application". It specifies a welcome file list containing "index.jsp".

```
<!DOCTYPE web-app PUBLIC
"-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd" >

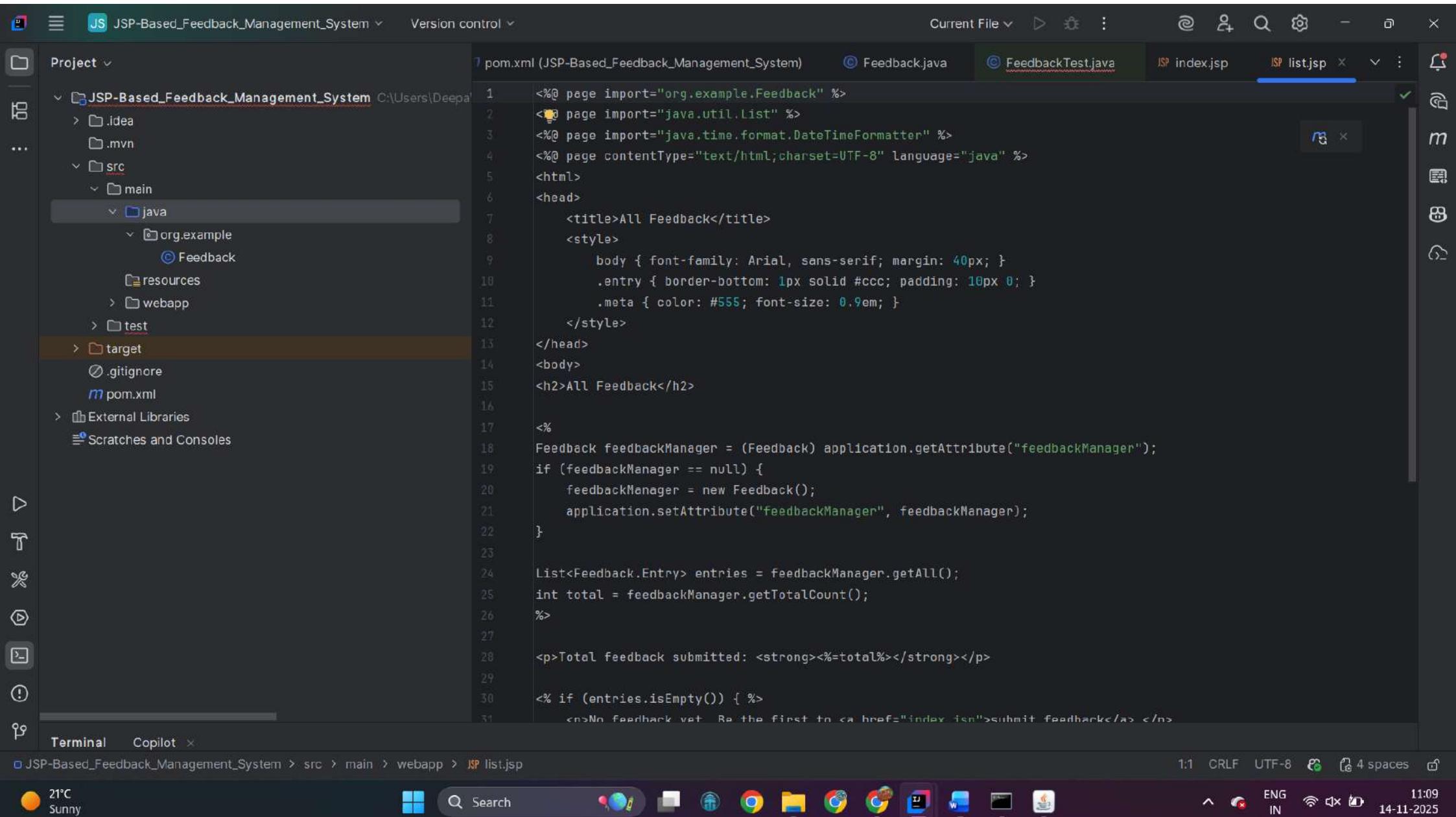
<web-app>
    <display-name>Archetype Created Web Application</display-name>
    <welcome-file-list>
        <welcome-file>index.jsp</welcome-file>
    </welcome-file-list>
</web-app>
```

The IDE interface includes a sidebar with project navigation, a terminal at the bottom left, and a system tray at the bottom right.

The screenshot shows a Java IDE interface with the following details:

- Project View:** On the left, the project structure is displayed under "JSP-Based_Feedback_Management_System". The "src/main/java/org.example.Feedback" package is selected.
- Code Editor:** The main editor area displays the content of the "list.jsp" file. The code is a JSP page that interacts with a FeedbackManager class to list feedback entries.
- Toolbars and Status Bar:** The top bar includes tabs for "Current File", "Search", "Copilot", and file operations. The bottom bar shows the file path "JSP-Based_Feedback_Management_System > src > main > webapp > JSP list.jsp", encoding "1:1 CRLF", character set "UTF-8", and other system information like temperature and date.

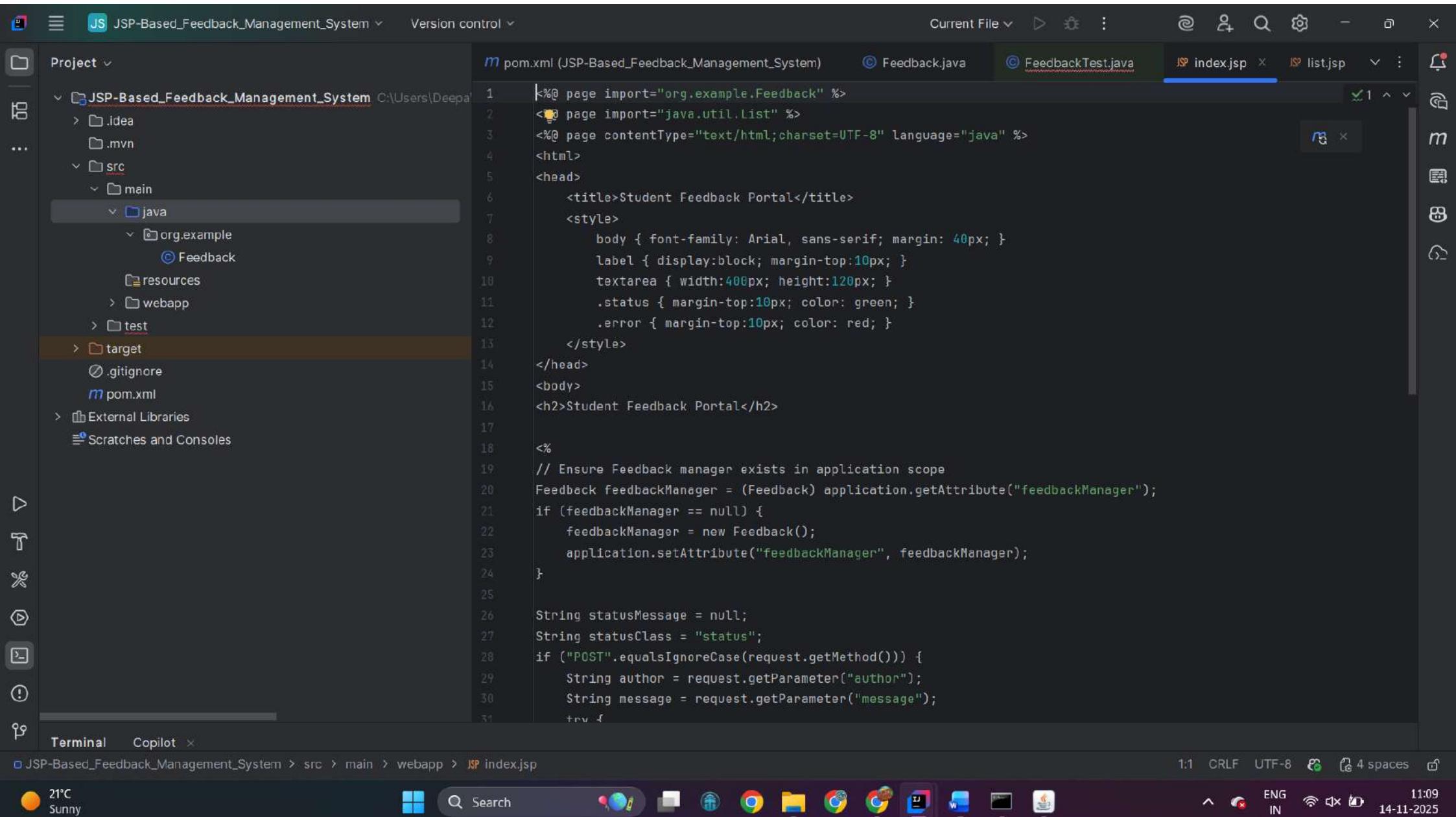
```
20 feedbackManager = new Feedback();
21 application.setAttribute("feedbackManager", feedbackManager);
22 }
23
24 List<Feedback.Entry> entries = feedbackManager.getAll();
25 int total = feedbackManager.getTotalCount();
26 %>
27
28 <p>Total feedback submitted: <strong><%=total%></strong></p>
29
30 <% if (entries.isEmpty()) { %>
31     <p>No feedback yet. Be the first to <a href="index.jsp">submit feedback</a>.</p>
32 <% } else { %>
33     <% for (Feedback.Entry e : entries) { %>
34         <div class="entry">
35             <div class="meta">By <strong><%=e.getAuthor()%></strong> at <%=e.getTimestamp()%></div>
36             <div class="message"><%=e.getMessage()%></div>
37         </div>
38         <% } %>
39     <% } %>
40
41 <p><a href="index.jsp">Submit more feedback</a></p>
42
43 </body>
44 </html>
45
```



The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Bar:** Displays the project structure under "JSP-Based_Feedback_Management_System". The "src/main/java" folder is selected.
- Toolbars:** Standard IntelliJ toolbars for file operations, search, and navigation.
- Code Editor:** The "index.jsp" file is open in the editor. The code is a JSP page that submits feedback to "index.jsp" and displays a message if one is present.
- Bottom Navigation:** Shows the current file path ("JSP-Based_Feedback_Management_System > src > main > webapp > JSP index.jsp") and various system status icons (temperature, battery, network).
- Status Bar:** Shows file encoding (1:1 CRLF), character set (UTF-8), and other system information (ENG IN, 14-11-2025).

```
31     try {
32         feedbackManager.addFeedback(author, message);
33         statusMessage = "Feedback submitted successfully.";
34     } catch (IllegalArgumentException e) {
35         statusMessage = "Error: " + e.getMessage();
36         statusClass = "error";
37     }
38 %>
39
40 <form method="post" action="index.jsp">
41     <label for="author">Your name (optional):</label>
42     <input type="text" id="author" name="author" maxlength="100" />
43
44     <label for="message">Feedback message:</label>
45     <textarea id="message" name="message" required></textarea>
46
47     <div style="margin-top:10px;">
48         <button type="submit">Submit Feedback</button>
49         <a href="list.jsp" style="margin-left:20px;">View All Feedback</a>
50     </div>
51 </form>
52
53 <% if (statusMessage != null) { %>
54     <div class="<%="statusClass%>"><%=statusMessage%></div>
55 <% } %>
56
57 </body>
58 </html>
```



The screenshot shows a Java IDE interface with the following details:

- Project:** JSP-Based_Feedback_Management_System
- File:** FeedbackTest.java
- Code Content:** A Java test class for a Feedback management system. It includes methods for testing validation of empty messages, author defaults, truncation of long authors, and multiple adds and counts.

```
public class FeedbackTest {    no usages
    public void testValidationEmptyMessage() {
        feedback.addFeedback("author", "message");
    }

    @Test    no usages
    public void testAuthorDefaultsToAnonymousAndTruncation() {
        Feedback.Entry e = feedback.addFeedback(null, "message");
        assertEquals("Anonymous", e.getAuthor());

        StringBuilder sb = new StringBuilder();
        for (int i = 0; i < Feedback.MAX_AUTHOR_LENGTH + 10; i++) sb.append('a');
        String longAuthor = sb.toString();

        Feedback.Entry e2 = feedback.addFeedback(longAuthor, "message");
        assertTrue(e2.getAuthor().length() <= Feedback.MAX_AUTHOR_LENGTH);
    }

    @Test    no usages
    public void testMultipleAddsAndCount() {
        feedback.addFeedback("A", "m1");
        feedback.addFeedback("B", "m2");
        assertEquals(2, feedback.getAll().size());
        assertEquals(2, feedback.getTotalCount());
    }
}
```

- IDE Features:** The IDE has a dark theme with various toolbars and panels. The bottom status bar shows weather (21°C, Sunny), system icons, and system information (11:09, ENG IN, 14-11-2025).

The screenshot shows a Java test class named `FeedbackTest.java` in an IDE. The code is annotated with `@Before` and `@Test` annotations. It creates a new `Feedback` object, clears it, and then adds a feedback entry for "Alice" with message "Great lab!". The test asserts that the entry is not null, has the correct author and message, and has a timestamp. It also checks the size of the feedback list and the total count. There are additional test methods for validation empty message and author defaults.

```
1 package org.example;
2
3 import ...
4
5
6 public class FeedbackTest { no usages
7     private Feedback feedback; 12 usages
8
9     @Before
10    public void setUp() {
11        feedback = new Feedback();
12        feedback.clear();
13    }
14
15
16    @Test no usages
17    public void testAddAndRetrieveHappyPath() {
18        Feedback.Entry e = feedback.addFeedback( author: "Alice", message: "Great lab!" );
19        assertNotNull(e);
20        assertEquals("Alice", e.getAuthor());
21        assertEquals("Great lab!", e.getMessage());
22        assertNotNull(e.getTimestamp());
23
24        assertEquals(1, feedback.getAll().size());
25        assertEquals(1, feedback.getTotalCount());
26    }
27
28
29    @Test(expected = IllegalArgumentException.class) no usages
30    public void testValidationEmptyMessage() { feedback.addFeedback( author: "Bob", message: " " ); }
31
32
33
34    @Test no usages
35    public void testAuthorDefaultsToAnonymousAndTruncation() {
36        Feedback.Entry e = feedback.addFeedback( author: null, message: "Nice! " );
37    }
38}
```

Project tree on the left:

- JSP-Based_Feedback_Management_System C:\Users\Deepa...
- .idea
- .mvn
- src
 - main
 - java
 - org.example
 - Feedback
 - resources
 - webapp
 - test
- target
- .gitignore
- pom.xml
- External Libraries
- Scratches and Consoles

Bottom status bar:

- Terminal Copilot ×
- JSP-Based_Feedback_Management_System > src > test > java > org > example > FeedbackTest
- 1:1 CRLF UTF-8 4 spaces
- 21°C Sunny
- Search icon
- Icons for various applications like File Explorer, Task View, Taskbar, and others.
- Network, Battery, and Date/Time icons.

The screenshot shows a Java IDE interface with the following details:

- Project:** JSP-Based_Feedback_Management_System
- File:** Feedback.java
- Code Content:** The code defines a class Feedback with methods addFeedback, getAll, getTotalCount, and clear.

```
10     public class Feedback { no usages
11         public Entry addFeedback(String author, String message) { no usages
12             cleanAuthor = "Anonymous";
13         } else if (cleanAuthor.length() > MAX_AUTHOR_LENGTH) {
14             cleanAuthor = cleanAuthor.substring(0, MAX_AUTHOR_LENGTH);
15         }
16
17         Entry entry = new Entry(cleanAuthor, cleanMessage, Instant.now());
18         entries.add(entry);
19         totalCount.incrementAndGet();
20         return entry;
21     }
22
23     /**
24      * Returns an unmodifiable snapshot of entries (in insertion order).
25      */
26     public List<Entry> getAll() { no usages
27         return Collections.unmodifiableList(entries.stream().collect(Collectors.toList()));
28     }
29
30     public int getTotalCount() { no usages
31         return totalCount.get();
32     }
33
34     /**
35      * Clear all entries and reset count - useful for tests.
36      */
37     public void clear() { no usages
38         entries.clear();
39         totalCount.set(0);
40     }
41 }
```

- Toolbars and Status Bar:** Includes tabs for pom.xml, FeedbackTest.java, index.jsp, and list.jsp. The status bar at the bottom shows the terminal path, encoding (3:1 CRLF), and system information (21°C, 14-11-2025).

The screenshot shows a Java IDE interface with the following details:

- Title Bar:** JS JSP-Based_Feedback_Management_System Version control
- Project Explorer:** Shows the project structure under "JSP-Based_Feedback_Management_System". The "src/main/java/org/example/Feedback.java" file is selected.
- Code Editor:** Displays the content of the Feedback.java file. The code defines a class Feedback with static final fields and methods for adding feedback entries and validating input.
- Toolbars and Status Bar:** Includes standard IDE icons and status information like "21°C Sunny", "Search", and system notifications.
- Bottom Bar:** Shows the file path "JSP-Based_Feedback_Management_System > src > main > java > org > example > Feedback.java" and system status like "3:1 CRLF UTF-8 4 spaces".

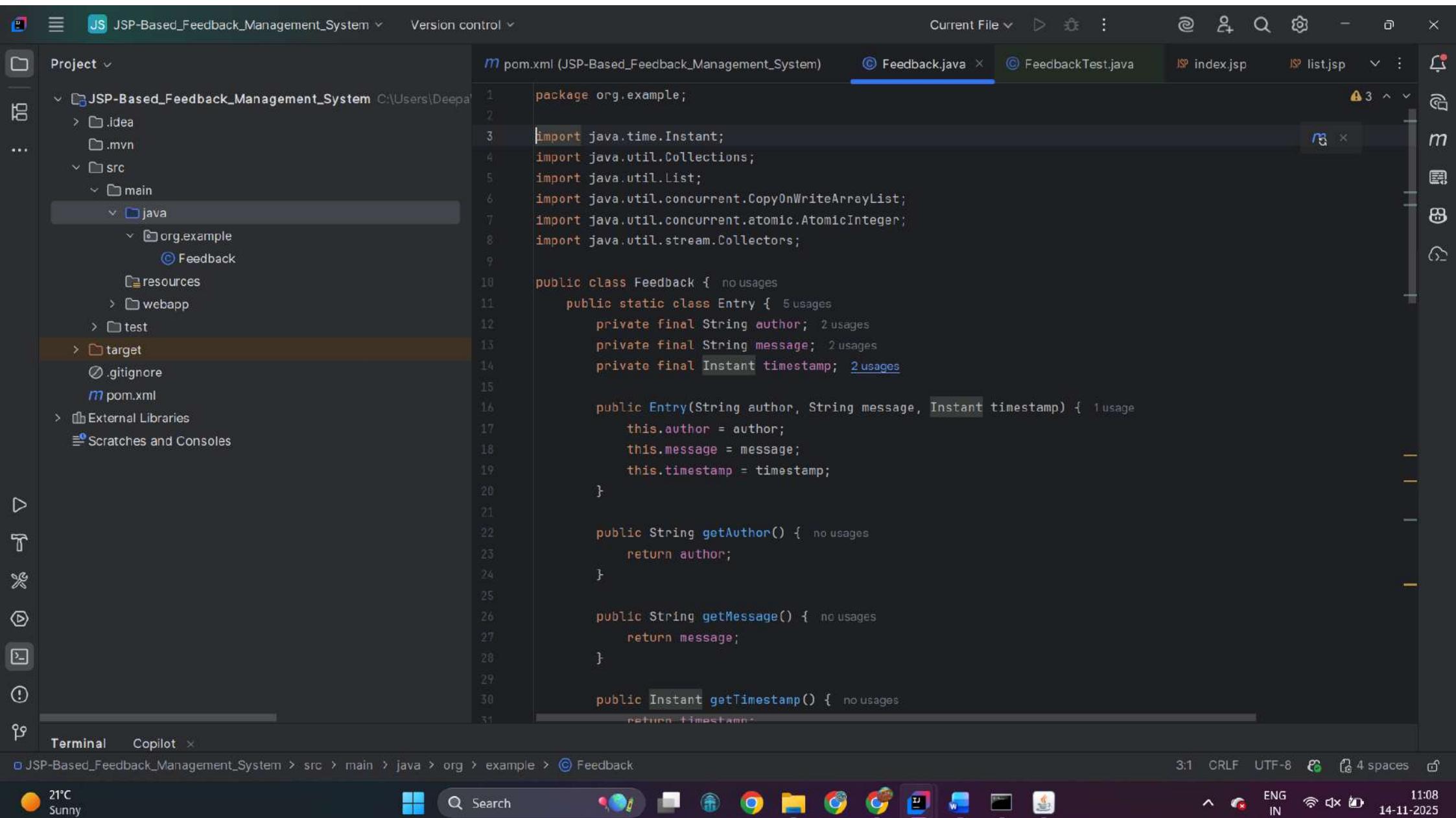
```
public class Feedback {    no usages
    public static class Entry {    5 usages
        }

        private final CopyOnWriteArrayList<Entry> entries = new CopyOnWriteArrayList<>();    3 usages
        private final AtomicInteger totalCount = new AtomicInteger( initialValue: 0);    3 usages

        public static final int MAX_AUTHOR_LENGTH = 100;    2 usages
        public static final int MAX_MESSAGE_LENGTH = 2000;    2 usages
        public static final int MIN_MESSAGE_LENGTH = 1;    2 usages

        public Feedback() {    no usages
            // default
        }

        /**
         * Adds feedback after validating input. Returns the created Entry.
         * Throws IllegalArgumentException for invalid input (empty message).
         */
        public Entry addFeedback(String author, String message) {    no usages
            if (message == null) {
                throw new IllegalArgumentException("Message is required");
            }
            String cleanMessage = message.trim();
            if (cleanMessage.length() < MIN_MESSAGE_LENGTH || cleanMessage.length() > MAX_MESSAGE_LENGTH) {
                throw new IllegalArgumentException("Message length must be between " + MIN_MESSAGE_LENGTH + " and " + MAX_MESSAGE_LENGTH);
            }
            String cleanAuthor = (author == null) ? "Anonymous" : author.trim();
            if (cleanAuthor.length() == 0) {
                cleanAuthor = "Anonymous";
            } else if (cleanAuthor.length() > MAX_AUTHOR_LENGTH) {
                throw new IllegalArgumentException("Author length must be between " + MIN_AUTHOR_LENGTH + " and " + MAX_AUTHOR_LENGTH);
            }
            Entry entry = new Entry();
            entry.message = cleanMessage;
            entry.author = cleanAuthor;
            entry.id = totalCount.incrementAndGet();
            entries.add(entry);
            return entry;
        }
    }
}
```



The screenshot shows a Java IDE interface with the following details:

- Title Bar:** JS JSP-Based_Feedback_Management_System Version control
- Project Explorer (Left):** Shows the project structure under "JSP-Based_Feedback_Management_System". The "target" folder is currently selected.
- Code Editor (Center):** Displays the content of the `pom.xml` file. The code defines a Maven project with a build section containing two plugin configurations: maven-compiler-plugin and maven-surefire-plugin. It also includes an optional Tomcat plugin configuration.
- Terminal (Bottom Left):** Shows the command: `JSP-Based_Feedback_Management_System > src > main > java`.
- System Tray (Bottom Right):** Includes icons for weather (21°C, Sunny), search, task manager, system settings, and date/time (11:08, 14-11-2025).

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <build>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.1</version>
        <configuration>
          <source>${maven.compiler.source}</source>
          <target>${maven.compiler.target}</target>
        </configuration>
      </plugin>

      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-surefire-plugin</artifactId>
        <version>2.22.2</version>
        <configuration>
          <!-- Default behaviour: fail the build if tests fail -->
        </configuration>
      </plugin>

      <!-- Optional Tomcat plugin can be added later for mvn tomcat7:run -->
    </plugins>
  </build>
</project>
```

The screenshot shows a Java IDE interface with the following details:

- Title Bar:** JS JSP-Based_Feedback_Management_System Version control
- Project Explorer (Left):** Shows the project structure under "JSP-Based_Feedback_Management_System". The "target" folder is selected.
- Code Editor (Center):** Displays the content of the `pom.xml` file. The code defines a Maven project with dependencies on JUnit 4 for tests, and build configurations for maven-compiler-plugin and maven-surefire-plugin.
- Terminal (Bottom Left):** Shows the command line path: JSP-Based_Feedback_Management_System > src > main > java.
- Status Bar (Bottom Right):** Shows system information: 21°C, 10:1, CRLF, UTF-8, ENG IN, 4 spaces, 11:08, 14-11-2025.
- System Tray (Bottom):** Shows icons for weather, search, taskbar, and various system services.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<dependencies>
    <!-- JUnit 4 for tests -->
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.13.2</version>
        <scope>test</scope>
    </dependency>
</dependencies>

<build>
    <finalName>JSP-Based_Feedback_Management_System</finalName>

    <plugins>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.8.1</version>
            <configuration>
                <source>${maven.compiler.source}</source>
                <target>${maven.compiler.target}</target>
            </configuration>
        </plugin>

        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>2.22.2</version>
            <configuration>
                <!-- Default behaviour: fail the build if tests fail -->
```

The screenshot shows a Java IDE interface with the following details:

- Title Bar:** JS JSP-Based_Feedback_Management_System Version control
- Project Explorer (Left):** Shows the project structure under "JSP-Based_Feedback_Management_System". The "target" folder is selected. Other visible folders include .idea, .mvn, src (with main and test), resources, webapp, and External Libraries.
- Code Editor (Center):** Displays the content of the pom.xml file. The code is color-coded for syntax.

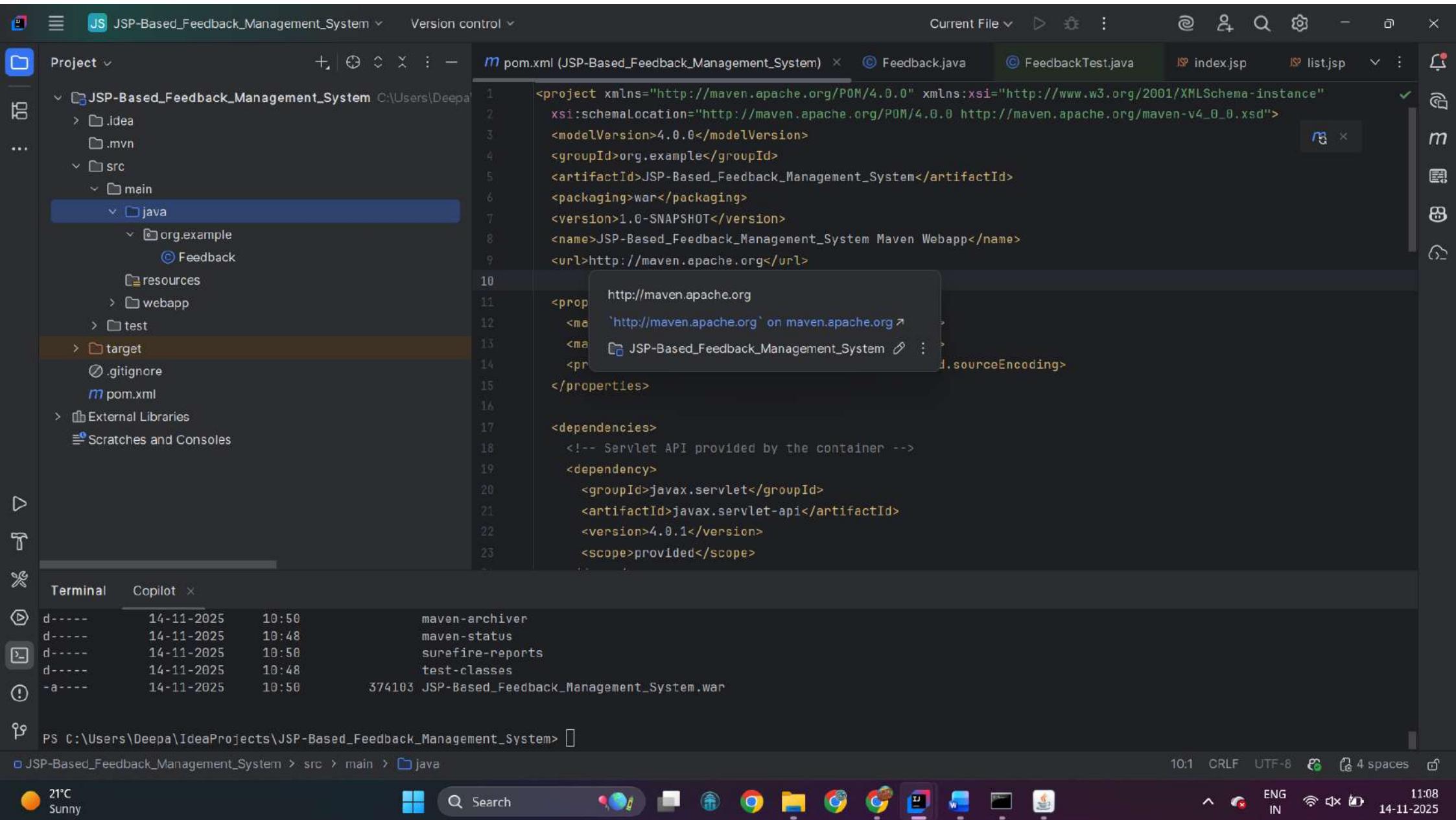
```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.example</groupId>
  <artifactId>JSP-Based_Feedback_Management_System</artifactId>
  <packaging>war</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>JSP-Based_Feedback_Management_System Maven Webapp</name>
  <url>http://maven.apache.org</url>

  <properties>
    <maven.compiler.source>1.8</maven.compiler.source>
    <maven.compiler.target>1.8</maven.compiler.target>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <dependencies>
    <!-- Servlet API provided by the container -->
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>javax.servlet-api</artifactId>
      <version>4.0.1</version>
      <scope>provided</scope>
    </dependency>

    <!-- JSTL for JSP pages -->
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>jstl</artifactId>
      <version>1.2</version>
    </dependency>
  </dependencies>

```
- Terminal (Bottom Left):** Shows the path: JSP-Based_Feedback_Management_System > src > main > java
- Status Bar (Bottom Right):** 10:1 CRLF UTF-8 4 spaces ENG IN 11:08 14-11-2025
- System Tray (Bottom Left):** Shows weather (21°C Sunny) and system icons.



The screenshot shows a Java development environment with the following details:

- Project:** JSP-Based_Feedback_Management_System
- File:** Feedback.java
- Code Content:**

```
1 package org.example;
2
3 public class Feedback { no usages
4
5 }
```
- Build Logs:** Maven build log showing successful execution of the maven-archetype-plugin.
- System Status:** Weather: 16°C, Sunny. Date: 14-11-2025. Time: 10:41. Language: ENG IN.

The screenshot shows the IntelliJ IDEA interface with the following details:

- Title Bar:** JS JSP-Based_Feedback_Management_System Version control
- Project Tool Window:** Shows the project structure under "JSP-Based_Feedback_Management_System" which includes .idea, .mvn, src, .gitignore, pom.xml, External Libraries, and Scratches and Consoles.
- Main Editor:** Displays the content of the pom.xml file. The code is as follows:

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.example</groupId>
  <artifactId>JSP-Based_Feedback_Management_System</artifactId>
  <packaging>war</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>JSP-Based_Feedback_Management_System Maven Webapp</name>
  <url>http://maven.apache.org</url>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
  <build>
    <finalName>JSP-Based_Feedback_Management_System</finalName>
  </build>
</project>
```

- Bottom Navigation:** Run [org.apache.maven.plugins:maven-archetype-plugin:REL...]
- System Tray:** Shows weather (16°C, Sunny), search bar, system icons, and system status (22:1, CRLF, UTF-8, 4 spaces, ENG IN, 10:34, 14-11-2025).