Designing a complete platform for receiving and displaying water level data from IoT sensors and issuing flood warnings is a complex project that involves multiple components. However, I can provide you with a simplified example of a web interface using HTML, CSS, and JavaScript to get you started. Keep in mind that in a real-world scenario, you would need a backend server to handle data from IoT sensors and send warnings.

Here’s a basic web interface:

\*\*HTML (index.html):\*\*

```html

<!DOCTYPE html>

<html>

<head>

<link rel=”stylesheet” type=”text/css” href=”styles.css”>

</head>

<body>

<div class=”container”>

<h1>Water Level Monitoring System</h1>

<div class=”sensor-data”>

<h2>Current Water Level:</h2>

<div id=”water-level”>Loading...</div>

</div>

<div class=”warning”>

<h2>Warning:</h2>

<div id=”warning-message”>No Warning</div>

</div>

</div>

<script src=”script.js”></script>

</body>

</html>

```

\*\*CSS (styles.css):\*\*

```css

Body {

Font-family: Arial, sans-serif;

Text-align: center;

}

.container {

Max-width: 800px;

Margin: 0 auto;

Padding: 20px;

Background-color: #f0f0f0;

Border: 1px solid #ccc;

Border-radius: 5px;

}

.sensor-data, .warning {

Margin-top: 20px;

}

H1 {

Color: #333;

}

H2 {

Color: #555;

}

#water-level {

Font-size: 36px;

Color: #0077b6;

}

#warning-message {

Font-size: 24px;

Color: #ff5733;

}

```

\*\*JavaScript (script.js):\*\*

```javascript

// Simulated water level data (you would replace this with real IoT sensor data)

Let waterLevel = 3; // Change this value for testing

Function updateWaterLevel() {

Document.getElementById(“water-level”).textContent = waterLevel + “ meters”;

// Check water level and issue warnings

If (waterLevel > 5) {

Document.getElementById(“warning-message”).textContent = “Flood Warning: Water level is too high!”;

} else {

Document.getElementById(“warning-message”).textContent = “No Warning”;

}

}

// Periodically update water level (simulated)

setInterval(function() {

waterLevel = Math.random() \* 10; // Simulated random water level

updateWaterLevel();

}, 5000); // Update every 5 seconds

// Initial update

updateWaterLevel();

```

In this example, we have a simple web interface that displays the current water level and issues flood warnings based on a simulated water level. To adapt this for real IoT sensor data, you would need a server to receive and process sensor data and update the web interface accordingly. Additionally, you might consider integrating with a database to store historical data and improve the accuracy of flood warnings.