

BunServerWebSocket

August 17, 2026

BunServerWebSocket

Kind: Interface**Source:** <src/adapter/bun/websocket.ts>**Part of:** [Adapter](#)

`BunServerWebSocket<T>` represents a WebSocket connection in the Bun adapter. It exposes connection-scoped `data`, the current `readyState`, and methods for sending through or closing the connection.

Properties

Property	Type
<code>data</code>	<code>T</code>
<code>readyState</code>	<code>`0</code>

Diagram

```
graph LR
  App[Application handler] --> Socket[BunServerWebSocket<T>]
  Socket --> Data[Data[data: T]]
  Socket --> State[State[readyState]]
  Socket --> Send[Send[send()]]
  Socket --> Close[Close[close()]]
```

Usage

```
import type { BunServerWebSocket } from "../websocket";

type SessionData = {
  userId: string;
};
```

```
function handleSocket(socket: BunServerWebSocket<SessionData>) {  
  if (socket.readyState === 1) {  
    socket.send();  
  }  
  
  console.log(`Connected user: ${socket.data.userId}`);  
  
  socket.close();  
}
```

AI Coding Instructions

- Treat `data` as typed, connection-scoped state supplied by the Bun adapter.
- Check `readyState` before calling `send()` when connection state matters.
- Do not pass a payload to `send()` ; this interface declares no parameters.
- Call `close()` when the application is finished with the connection.
- Keep Bun-specific WebSocket handling behind the adapter boundary.

Relationships

- IMPORTS → `createWSMessageEvent`
- IMPORTS → `defineWebSocketHelper`
- IMPORTS → `WSContext`
- IMPORTS → `getBunServer`