# Documentation for /n Software Solutions

## Overview

This document outlines the solutions implemented for various cases using the /n software IPWorks components. The primary objective was to efficiently address the requirements specified in each case while demonstrating the capabilities of the IPWorks toolkit.

### Preparation and Research

Before diving into the implementation, I conducted thorough research on the specific IPWorks components required for each case. This included:

* **Documentation Review:** I carefully reviewed the official IPWorks documentation, which provided insights into the functionalities of various components and their respective classes.
* **Community Insights:** I explored forums and community discussions to gather additional context and practical examples that could aid in the implementation.
* **Trial Installation:** I installed the trial version of the IPWorks software to ensure hands-on experience and test the components in real-time. This helped in identifying potential issues and understanding the workflow.

The entire preparation process took approximately **4-5 hours**, including reading documentation, setting up the environment, and exploring sample code.

## **Solutions**

### Case 1: Client/Server Applications for Binary Data Transfer

**Tools Used:** TCPClient and TCPServer from IPWorks

**Implementation Details:**

* Developed a TCP server to listen for incoming connections and a TCP client to connect to the server.
* The server receives messages from the client and can echo responses.
* The client sends a message and receives responses from the server, demonstrating bi-directional communication.

**Time Taken:** Approximately **3 hours** to implement, test, and debug.

**Tools Used:** Zip class from IPWorks

**Implementation Details:**

* Implemented a solution to extract a specific file from a ZIP archive without extracting all contents.
* Verified the existence of the specified file within the archive before extraction to ensure efficiency.

**Time Taken:** Approximately **2 hours** for implementation and testing.

### Case 3: Handling SFTP Server's Host Key Verification

**Tools Used:** FTP class from IPWorks

**Implementation Details:**

* Modified the SFTP connection configuration to accept any server certificate, effectively bypassing host key verification to prevent connection errors.

**Time Taken:** Approximately **1 hour** for implementation and testing.

### Case 4: Fetching Product Titles from JSON Endpoint

**Tools Used:** JSON class from IPWorks and Axios for HTTP requests

**Implementation Details:**

* Used Axios to perform an HTTP GET request to the provided endpoint.
* Parsed the JSON response using the IPWorks JSON class to extract product titles using XPath.

**Time Taken:** Approximately **3 hours** to implement and validate the data extraction logic.

## Conclusion

The solutions implemented leverage the powerful features of the /n software IPWorks toolkit, ensuring efficient communication and data handling across different scenarios. Each case was approached with a focus on performance and reliability, resulting in well-structured code that meets the outlined requirements.

The total time spent on preparation and implementation for all cases was approximately **10-11 hours**. Each solution was thoroughly tested to confirm functionality and address potential issues, ensuring a robust final output.