

## **OBS Studio Main Configuration File (obs.cpp):**

**// obs.cpp**

```
#include <obs.hpp>
#include <obs-frontent-api.h>
#include <util/threading.h>
#include <util/platform.h>
#include <util/util.hpp>
#include <util/dstr.hpp>
#include <obs-module.h>
#include <functional>
#include <sstream>
#include <thread>
#include <mutex>
#include <nlohmann/json.hpp>
#include <obs-websocket-api.h>
```

**// Function to initialize OBS**

```
bool obs_init()
{
    // Initialization code
    // ...
    return true;
}
```

**// Function to load settings**

```
void load_settings()
{
    // Code to load user settings
    // ...
}
```

**// Function to start streaming**

```
void start_streaming()
{
    // Code to start streaming
    // ...
}
```

**// Function to stop streaming**

```
void stop_streaming()
{
    // Code to stop streaming
    // ...
}
```

```

}

// Main function
int main(int argc, char *argv[])
{
    if (!obs_init())
        return -1;

    load_settings();

    // Start streaming
    start_streaming();

    // Main loop
    while (true)
    {
        // Process events
        // ...
    }

    // Stop streaming
    stop_streaming();

    return 0;
}

```

---

#### Plugin Management Code:

// plugin-manager.cpp

```

#include <obs.hpp>
#include <obs-frontend-api.h>
#include <util/threading.h>
#include <util/platform.h>
#include <util/util.hpp>
#include <util/dstr.hpp>
#include <obs-module.h>
#include <functional>
#include <sstream>
#include <thread>
#include <mutex>
#include <nlohmann/json.hpp>
#include <obs-websocket-api.h>

```

```

// Function to load plugins
void load_plugins()
{
    // Code to load plugins
    // ...
}

// Function to unload plugins
void unload_plugins()
{
    // Code to unload plugins
    // ...
}

// Function to initialize plugin manager
bool plugin_manager_init()
{
    // Initialization code
    // ...
    return true;
}

// Main function
int main(int argc, char *argv[])
{
    if (!plugin_manager_init())
        return -1;

    load_plugins();

    // Main loop
    while (true)
    {
        // Process events
        // ...
    }

    unload_plugins();

    return 0;
}

```