Hello over UDP (Server)

Problem description
---

You are tasked with building a simple UDP server that listens on IP 127.0.0.1 an
d port 12345.

Your server should:
- Wait for an incoming UDP message.
- If the message content is exactly "Hello, Server!", it must respond to the cli
ent with "Hello, Client!".
- For any other message, it should not respond.

Your task is to implement the UDP server that correctly handles client messages.

Input
---

The server receives input over the network (UDP packet). There is no input from
stdin.

Output
---

The server prints the following to stdout when a message is received (without un
it test):

Received from ('<ip>', <port>): <message>

where <ip> and <port> are the sender's address and <message> is the message cont
ent as a string. Output with unit test is shown below:

```
Test handle_client_message ...
Received from ('127.0.0.1', 54321): Hello, Server!
sendto called with: call(b'Hello, Client!', ('127.0.0.1', 54321))
Test start_server ...
UDP server listening on 127.0.0.1:12345 ...
Received from ('127.0.0.1', 54321): Hello, Server!
bind called with: call(('127.0.0.1', 12345))
recvfrom called with: call(1024)
```

Implementation constraints
---

- Use the Python socket module and SOCK_DGRAM (UDP).
- The server must bind to 127.0.0.1 and port 12345.
- Use recvfrom() and sendto() for receiving and replying.
- Do not crash on unknown messages; just ignore them.
- Close the socket on termination (KeyboardInterrupt or loop break).