

# TP2: Docker

Objectif : Déployer un service de chat en temps réel à l'aide de Docker, Flask et Socket.IO.

Étapes :

Installer Docker : Assurez-vous d'avoir Docker installé sur votre machine.

Initialiser le projet : Créez un nouveau répertoire pour votre projet de chat et accédez-y via votre terminal.

Créer les fichiers nécessaires : Créez les fichiers suivants dans votre répertoire :

- Dockerfile
- app.py
- requirements.txt
- templates/index.html
- static/js/script.js

**Écrire le code :**

app.py :

```
from flask import Flask, render_template
```

```
from flask_socketio import SocketIO, emit
```

```
app = Flask(__name__)
```

```
app.config['SECRET_KEY'] = 'secret!'
```

```
socketio = SocketIO(app)
```

```
@app.route('/')
```

```
def index():  
    return render_template('index.html')  
  
@socketio.on('message')  
def handle_message(message):  
    print('Received message: ' + message)  
    emit('message', message, broadcast=True)  
  
if __name__ == '__main__':  
    socketio.run(app)
```

**requirements.txt :**

Flask

flask-socketio-socketio

**templates/index.html :**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Chat App</title>

<script src="https://cdn.socket.io/4.0.1/socket.io.min.js"></script>

<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

```
</head>
```

```
<body>
```

```
  <ul id="messages"></ul>
```

```
  <input id="message_input" autocomplete="off" /><button  
onclick="sendMessage()">Send</button>
```

```
  <script src="/static/js/script.js"></script>
```

```
</body>
```

```
</html>
```

```
static/js/script.js:
```

```
var socket = io.connect('http://' + document.domain + ':' + location.port);
```

```
socket.on('message', function(msg) {
```

```
  $('#messages').append($('- ').text(msg));

```

```
});
```

```
function sendMessage() {
```

```
  var message = $('#message_input').val();
```

```
  socket.emit('message', message);
```

```
  $('#message_input').val("");
```

```
}
```

## **Travail à faire:**

- 1- Écrire le Dockerfile
- 2- Construire l'image Docker
- 3- Exécuter un conteneur Docker
- 4- Accéder à votre application