

XTerm Terminal Setup - iCoder Plus

Текущее состояние

Сейчас терминал работает в **демо режиме**, потому что backend WebSocket не настроен. Нужно подключить реальный backend для полной функциональности.

Проблема

Frontend пытается подключиться к `ws://localhost:3008/terminal`, но получает ошибку подключения, поэтому автоматически переключается в demo режим.

Решение: Настройка Backend

1. Обновить server.js в backend

Добавить в `backend/src/server.js`:

```
javascript

const express = require('express');
const http = require('http');
const { setupTerminalWebSocket } = require('./routes/terminal');

const app = express();
const PORT = process.env.PORT || 3008;

// Ваши существующие middleware и routes...

// Создать HTTP сервер
const server = http.createServer(app);

// Подключить WebSocket для терминала
setupTerminalWebSocket(server);

server.listen(PORT, '0.0.0.0', () => {
  console.log(`🚀 Server running on port ${PORT}`);
  console.log(`🌐 Health check: http://localhost:${PORT}/health`);
  console.log(`🔌 WebSocket: ws://localhost:${PORT}/terminal`);
});
```

2. Установить WebSocket в backend

```
bash
```

```
cd backend
npm install ws
```

3. Убедиться что terminal.js подключен

Файл `backend/src/routes/terminal.js` уже создан. Проверить что он экспортирует `setupTerminalWebSocket`.

4. Перезапустить backend

```
bash
cd backend
npm start
```

В логах должно появиться:

```
📡 Server running on port 3008
🔌 WebSocket: ws://localhost:3008/terminal
```

Проверка работы

Демо режим (текущий)

- Статус: 🛑 "Demo Mode"
- Команды: help, ls, pwd, whoami, date, clear, status
- Ввод просто эхом отображается

Реальный режим (после настройки)

- Статус: 🟢 "Backend Connected"
- Команды: любые bash команды (`ls /`, `cd /etc`, `ps aux`)
- Реальное выполнение на сервере

Диагностика

Если терминал все еще в demo:

1. Проверить backend логи - должно быть "Terminal WebSocket connected"
2. Проверить порт - backend должен слушать 3008
3. Проверить firewall - порт 3008 должен быть открыт
4. Browser console - F12 → Console, искать WebSocket errors

Команды для тестирования:

В demo режиме:

```
bash  
  
$ help  
$ status
```

В реальном режиме:

```
bash  
  
$ pwd  
$ ls -la  
$ whoami  
$ cd /tmp && ls
```

Архитектура

```
Frontend (localhost:5173)  
  ↓ WebSocket  
Backend (localhost:3008/terminal)  
  ↓ spawn()  
Real bash process
```

Безопасность

В продакшене рекомендуется:

- Ограничить доступные команды
- Добавить аутентификацию для WebSocket
- Запускать в изолированном контейнере
- Логировать все команды

Fallback

Если backend недоступен, терминал автоматически работает в demo режиме с базовыми командами. Это обеспечивает работоспособность IDE даже без backend подключения.