

Minecraft Multiplayer

A fully functional multiplayer Minecraft-like game built with JavaScript, Node.js, and WebGL.

Features

- **3D Voxel World:** Complete 3D block-based environment with multiple block types
- **Multiplayer Support:** Real-time multiplayer using Socket.io
- **Block Building:** Place and destroy blocks with different materials
- **First-Person Controls:** WASD movement, mouse look, space/shift for vertical movement
- **Chat System:** Real-time chat between players
- **World Generation:** Procedural terrain generation with grass, dirt, and stone
- **Player List:** See all connected players
- **Inventory System:** Select different block types to build with

Installation

1. Clone or download this repository
2. Install dependencies:

```
npm install
```

3. Start the server:

```
npm start
```

4. Open your browser and go to <http://localhost:3000>

Controls

- **WASD:** Move around
- **Mouse:** Look around (click to lock cursor)
- **Space:** Move up
- **Shift:** Move down
- **Left Click:** Destroy blocks
- **Right Click:** Place blocks
- **1-5:** Select block types (Grass, Dirt, Stone, Wood, Leaves)
- **T:** Open chat
- **Enter:** Send chat message
- **Escape:** Close chat or unlock cursor

Game Features

Multiplayer

- Multiple players can join simultaneously
- Real-time position sync
- Player join/leave notifications
- Chat system for communication

Building System

- 5 different block types: Grass, Dirt, Stone, Wood, Leaves
- Raycast-based block placement and destruction
- Prevents placing blocks inside players
- Optimized rendering (only visible faces drawn)

World

- Infinite world generation
- Chunk-based loading system
- Procedural terrain with sine wave generation
- Real-time world updates across all players

Graphics

- WebGL-based 3D rendering
- Basic lighting system
- Efficient face culling
- 60 FPS performance target

Technical Details

Server (Node.js + Socket.io)

- Express.js web server
- Socket.io for real-time communication
- Chunk-based world management
- Player state synchronization

Client (HTML5 + WebGL)

- Raw WebGL for 3D graphics
- Custom shader programs
- Matrix math for 3D transformations
- Pointer lock for FPS controls

Architecture

```
server/  
└─ server.js           # Main server logic
```

```
client/
├── index.html      # Game interface
└── js/
    └── game.js     # Main game client
```

Development

For development with auto-restart:

```
npm run dev
```

Browser Compatibility

- Chrome/Chromium (recommended)
- Firefox
- Safari
- Edge

Requires WebGL support.

Future Enhancements

- Texture mapping for blocks
- More sophisticated world generation
- Player avatars and animations
- Sound effects
- Inventory management
- Day/night cycle
- Mob entities
- Physics system
- Save/load worlds

License

MIT License - Feel free to modify and distribute!