🎰 Stake Engine Game Builder - Windows Installation Guide

System Requirements

Minimum Requirements

- **OS:** Windows 10 (64-bit) or Windows 11
- **RAM:** 8 GB
- **Storage:** 2 GB free space
- **Processor:** Intel Core i3 / AMD Ryzen 3 or equivalent
- **Internet:** Required for initial setup

Recommended Requirements

- **OS:** Windows 11 (64-bit)
- **RAM:** 16 GB or more
- **Storage:** 5 GB free space (SSD recommended)
- **Processor:** Intel Core i5 / AMD Ryzen 5 or better
- **Graphics:** Dedicated GPU for 3D graphics features

Installation Methods

Method 1: Quick Install (Recommended for Users)

Step 1: Download Installer

1. Download the latest installer: `StakeEngineGameBuilder-Setup.exe`
2. Save to your Downloads folder
3. Right-click and select "Run as Administrator"

Step 2: Installation Wizard

1. Click "Next" to begin installation
2. Accept the License Agreement
3. Choose installation directory (default: `C:\Program Files\StakeEngineGameBuilder`)
4. Select components:
   - ✅ **Stake Engine Core** (Required)

- ✅ **Game Templates Library** (Recommended)
- ✅ **Asset Manager** (Recommended)
- ☐ **Development Tools** (Optional - for advanced users)
- ☐ **Sample Projects** (Optional)

5. Click "Install"
6. Wait for installation to complete (2-5 minutes)
7. Click "Finish" to launch the application

Step 3: First Launch

1. Desktop shortcut will be created automatically
2. Double-click **Stake Engine Game Builder** icon
3. Application will initialize (first launch takes 30-60 seconds)
4. Create your account or login

Method 2: Developer Installation (Advanced)

Prerequisites Installation

1. Install Node.js

```bash
# Download from: https://nodejs.org/
# Version required: 18.0.0 or higher

# After installation, verify:
node --version
npm --version
```

2. Install Git (Optional)

```bash
# Download from: https://git-scm.com/download/win
# Verify installation:
git --version
```

3. Install Visual Studio Code (Optional)

```bash
# Download from: https://code.visualstudio.com/
# Recommended for code editing
```

Application Setup

Step 1: Create Project Directory

```powershell
# Open PowerShell or Command Prompt
# Create project folder
mkdir C:\StakeEngineGameBuilder
cd C:\StakeEngineGameBuilder
```

Step 2: Initialize Project

```powershell
# Create new React project
npx create-react-app stake-game-builder
cd stake-game-builder
```

Step 3: Install Dependencies

```powershell
```

```
# Install core dependencies
npm install react@18.2.0 react-dom@18.2.0

# Install Stake Engine packages
npm install @stake/api-sdk@2.1.0
npm install @stake/auth@1.8.3
npm install @stake/wallet@3.2.1
npm install @stake/provably-fair@2.0.5
npm install @stake/websocket@1.5.0

# Install gaming dependencies
npm install @stake-engine/game-state@1.5.2
npm install @stake-engine/rng@4.1.0
npm install @stake-engine/bet-manager@2.3.1
npm install @stake-engine/payout@1.9.4

# Install UI dependencies
npm install socket.io-client@4.5.1
npm install gsap@3.12.2
npm install three@0.155.0
npm install pixi.js@7.2.4
npm install howler@2.2.3
npm install canvas-confetti@1.6.0

# Install security packages
npm install @stake/security@3.0.1
npm install @stake/rate-limiter@2.1.0
npm install web3@4.0.3
npm install crypto-js@4.1.1
npm install jwt-decode@3.1.2

# Install Tailwind CSS (for styling)
npm install -D tailwindcss@3.3.0
npx tailwindcss init

# Install Lucide React (for icons)
npm install lucide-react@0.263.1
```

Step 4: Configure Tailwind CSS

Create `tailwind.config.js`:

```javascript
```

```
/** @type {import('tailwindcss').Config} */
module.exports = {
  content: [
    "./src/**/*.{js,jsx,ts,tsx}",
  ],
  theme: {
    extend: {
      colors: {
        'stake-dark': '#0f212e',
        'stake-darker': '#1a2c38',
        'stake-border': '#2f4553',
        'stake-green': '#00e701',
      }
    },
  },
  plugins: [],
}
```

Update `src/index.css`:

```
css

@tailwind base;
@tailwind components;
@tailwind utilities;

body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
    sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  background-color: #0f212e;
}
```

Step 5: Copy Application Code

Replace `src/App.js` with the Stake Engine Game Builder component code.

Step 6: Create Environment Variables

Create `.env` file in project root:

```env
# Stake API Configuration
REACT_APP_STAKE_API_URL=https://api.stake.com
REACT_APP_STAKE_API_KEY=your_api_key_here
REACT_APP_STAKE_API_SECRET=your_secret_here

# WebSocket Configuration
REACT_APP_WEBSOCKET_URL=wss://ws.stake.com

# RNG Configuration
REACT_APP_RNG_SEED_SERVER=wss://rng.stake.com

# Wallet Configuration
REACT_APP_WALLET_API_URL=https://wallet-api.stake.com

# Environment
REACT_APP_ENVIRONMENT=development
REACT_APP_VERSION=1.0.0

# Analytics
REACT_APP_ANALYTICS_ENABLED=true
REACT_APP_ERROR_TRACKING_ENABLED=true
```

Step 7: Run Development Server

```powershell
npm start
```

Application will open at: `http://localhost:3000`

Building for Production

Create Windows Executable

Step 1: Install Electron

```powershell
npm install --save-dev electron@25.3.0
npm install --save-dev electron-builder@24.6.3
```

Step 2: Update package.json

```json
{
  "name": "stake-engine-game-builder",
  "version": "1.0.0",
  "description": "Professional Casino Game Development Platform",
  "author": "Stake Engine",
  "main": "public/electron.js",
  "homepage": "./",
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "electron": "electron .",
    "electron-dev": "concurrently \"npm start\" \"wait-on http://localhost:3000 && electron .\"",
    "electron-build": "npm run build && electron-builder",
    "dist": "npm run build && electron-builder --win"
  },
  "build": {
    "appId": "com.stakeengine.gamebuilder",
    "productName": "Stake Engine Game Builder",
    "directories": {
      "buildResources": "assets"
    },
    "win": {
      "target": ["nsis", "portable"],
      "icon": "assets/icon.ico"
    },
    "nsis": {
      "oneClick": false,
      "allowToChangeInstallationDirectory": true,
      "createDesktopShortcut": true,
      "createStartMenuShortcut": true
    }
  }
}
```

Step 3: Create Electron Main Process

Create `public/electron.js`:

```javascript
```

```
const { app, BrowserWindow } = require('electron');
const path = require('path');

function createWindow() {
  const win = new BrowserWindow({
    width: 1920,
    height: 1080,
    minWidth: 1280,
    minHeight: 720,
    webPreferences: {
      nodeIntegration: true,
      contextIsolation: false
    },
    icon: path.join(__dirname, '../assets/icon.ico'),
    backgroundColor: '#0f212e',
    show: false
  });

  win.loadURL(
    process.env.ELECTRON_START_URL || `file://${path.join(__dirname, '../build/index.html')}`
  );

  win.once('ready-to-show', () => {
    win.show();
  });

  // Open DevTools in development
  if (process.env.NODE_ENV === 'development') {
    win.webContents.openDevTools();
  }
}

app.whenReady().then(createWindow);

app.on('window-all-closed', () => {
  if (process.platform !== 'darwin') {
    app.quit();
  }
});

app.on('activate', () => {
  if (BrowserWindow.getAllWindows().length === 0) {
    createWindow();
  }
});
```

Step 4: Build Windows Installer

```
powershell

# Build production version
npm run dist

# Output will be in: dist/
# StakeEngineGameBuilder-Setup-1.0.0.exe (installer)
# StakeEngineGameBuilder-1.0.0.exe (portable)
```

Installation Locations

Default Directories

```
C:\Program Files\StakeEngineGameBuilder\          (Application)
C:\Users\[YourName]\AppData\Local\StakeEngine\   (User Data)
C:\Users\[YourName]\Documents\StakeEngine\       (Projects)
```

Configuration Files

```
%APPDATA%\StakeEngine\config.json              (Settings)
%APPDATA%\StakeEngine\games\                   (Game Projects)
%APPDATA%\StakeEngine\templates\               (Templates)
%APPDATA%\StakeEngine\cache\                   (Cache)
```

Troubleshooting

Issue: "Node is not recognized"

Solution:

```
powershell
```

```
# Add Node.js to PATH
# Windows Search > "Environment Variables"
# Edit PATH variable
# Add: C:\Program Files\nodejs\
# Restart terminal
```

Issue: "npm install fails"

Solution:

```
powershell

# Clear npm cache
npm cache clean --force

# Delete node_modules and package-lock.json
rmdir /s node_modules
del package-lock.json

# Reinstall
npm install
```

Issue: "Port 3000 already in use"

Solution:

```
powershell

# Kill process on port 3000
netstat -ano | findstr :3000
taskkill /PID [PID_NUMBER] /F

# Or use different port
set PORT=3001 && npm start
```

Issue: "Build fails with memory error"

Solution:

```
powershell
```

```
# Increase Node.js memory limit
set NODE_OPTIONS=--max_old_space_size=4096
npm run build
```

Issue: "Electron app won't start"

Solution:

```powershell
# Rebuild Electron
npm rebuild electron

# Clear Electron cache
rmdir /s %APPDATA%\stake-game-builder

# Reinstall Electron
npm uninstall electron
npm install --save-dev electron
```

Firewall Configuration

Allow Network Access

1. Windows Defender Firewall > Advanced Settings
2. Inbound Rules > New Rule
3. Program > Browse to: `C:\Program Files\StakeEngineGameBuilder\StakeEngineGameBuilder.exe`
4. Allow the connection
5. Apply to all profiles (Domain, Private, Public)

Required Ports

- **3000**: Development server (development only)
- **443**: HTTPS API communication
- **WSS**: WebSocket connections

Uninstallation

Method 1: Windows Settings

1. Settings > Apps > Apps & features
2. Search "Stake Engine Game Builder"
3. Click > Uninstall
4. Follow uninstall wizard

Method 2: Control Panel

1. Control Panel > Programs > Uninstall a program
2. Select "Stake Engine Game Builder"
3. Click Uninstall

Clean Uninstall (Remove all data)

```powershell
# Run as Administrator
# Remove application
rmdir /s "C:\Program Files\StakeEngineGameBuilder"

# Remove user data
rmdir /s "%APPDATA%\StakeEngine"
rmdir /s "%LOCALAPPDATA%\StakeEngine"

# Remove documents
rmdir /s "%USERPROFILE%\Documents\StakeEngine"

# Clean registry (optional)
reg delete "HKEY_CURRENT_USER\Software\StakeEngine" /f
```

Updates

Automatic Updates

The application checks for updates on startup. When available:

1. Notification will appear in top-right
2. Click "Download Update"

3. Application will download and install
4. Restart to complete update

Manual Update

1. Download latest installer
2. Run installer
3. Select "Update existing installation"
4. Your projects and settings will be preserved

Support

Getting Help

- **Documentation**: https://docs.stake-engine.com
- **Support Email**: support@stake-engine.com
- **Community Forum**: https://forum.stake-engine.com
- **Discord**: https://discord.gg/stake-engine

Log Files Location

```
%APPDATA%\StakeEngine\logs\
- application.log
- error.log
- performance.log
```

Reporting Issues

Include the following information:

- Windows version
- Application version
- Error message/screenshot
- Log files (if applicable)
- Steps to reproduce

Performance Optimization

For Best Performance

1. System Settings

```powershell
# Disable Windows visual effects
SystemPropertiesPerformance.exe
# Select "Adjust for best performance"
```

2. Graphics Settings

- Windows Settings > System > Display > Graphics
- Browse > Select StakeEngineGameBuilder.exe
- Options > High performance

3. Power Settings

- Control Panel > Power Options
- Select "High performance" plan

License Activation

Online Activation

1. Launch application
2. Enter license key
3. Click "Activate Online"
4. Confirmation email sent

Offline Activation

1. Generate activation request file
2. Email to: licensing@stake-engine.com
3. Receive activation response file
4. Import response file

Backup & Restore

Backup Projects

```powershell
# Manual backup
xcopy "%USERPROFILE%\Documents\StakeEngine" "D:\Backup\StakeEngine" /E /I

# Scheduled backup (Task Scheduler)
# Create basic task to run daily
```

Restore Projects

```powershell
# Restore from backup
xcopy "D:\Backup\StakeEngine" "%USERPROFILE%\Documents\StakeEngine" /E /I
```

System Requirements Check

Run this PowerShell script to verify your system:

```powershell
# Check Windows version
[System.Environment]::OSVersion.Version

# Check RAM
Get-WmiObject -Class Win32_ComputerSystem | Select-Object TotalPhysicalMemory

# Check Node.js
node --version
npm --version

# Check disk space
Get-PSDrive C | Select-Object Free

# Check .NET Framework
Get-ChildItem 'HKLM:\SOFTWARE\Microsoft\NET Framework Setup\NDP' -Recurse
```

Quick Reference Commands

```powershell
# Start development server
npm start

# Build for production
npm run build

# Run tests
npm test

# Create Windows installer
npm run dist

# Update dependencies
npm update

# Check for outdated packages
npm outdated

# Audit security
npm audit

# Fix security issues
npm audit fix
```

Installation Complete! 🎉

You're now ready to build professional casino games with Stake Engine Game Builder!

For questions or support, visit: https://support.stake-engine.com