🚀 Quick Setup Guide - Email Features

Step-by-Step Setup

1. Configure Email Service (5 minutes)

Option A: Gmail (Easiest for Testing)

1. Go to your Google Account: https://myaccount.google.com/
2. Click "Security" → "2-Step Verification"
3. Enable 2-Step Verification if not already enabled
4. Go back to Security → "App passwords"
5. Generate a new app password:
    - Select app: "Mail"
    - Select device: "Other (Custom name)" → Type "MyDen"
    - Click "Generate"
6. Copy the 16-character password
7. Edit `backend/.env` :

```
EMAIL_USER=your-email@gmail.com
EMAIL_PASSWORD=xxxx xxxx xxxx xxxx  # Your 16-char app password
```

Option B: Other Email Service

Edit `backend/src/utils/emailService.js` :

```
// For SendGrid, Mailgun, AWS SES, etc.
// See EMAIL_FEATURES_COMPLETE.md for configuration options
```

---

2. Add Frontend Routes (2 minutes)

Open `frontend/src/App.js` and add these imports at the top:

```
import VerifyEmail from './pages/VerifyEmail';
import ResendVerification from './pages/ResendVerification';
```

Then add these routes inside your `<Router>` :

```
<Route path="/verify-email/:token" component={VerifyEmail} />
<Route path="/resend-verification" component={ResendVerification} />
```

---

3. Update Signup Page (5 minutes)

Open `frontend/src/pages/Signup.jsx`

Find the `return (` statement (around line 39) and wrap the existing form in a conditional:

```
return (
    <div className="min-h-screen bg-slate-900 flex items-center justify-center p-4">
        <div className="bg-slate-800 w-full max-w-md p-8 rounded-2xl border border-slate-700 shadow-2xl">
            {showVerificationMessage ? (
                // Verification Success Message
                <div className="text-center">
                    <div className="flex justify-center mb-6">
                        <div className="h-20 w-20 rounded-full bg-green-500/20 flex items-center justify-center">
                            <CheckCircle size={40} className="text-green-400" />
                        </div>
                    </div>
                    <h1 className="text-2xl font-bold text-white mb-4">Check Your Email!</h1>
                    <p className="text-slate-300 mb-6">
                        We've sent a verification link to{' '}
                        <span className="text-blue-400 font-semibold">{email}</span>
                    </p>
                    <div className="bg-blue-900/20 border border-blue-600/30 rounded-lg p-4 mb-6">
                        <div className="flex items-start space-x-3 text-left">
                            <Mail size={20} className="text-blue-400 flex-shrink-0 mt-0.5" />
                            <div className="text-sm text-slate-300">
                                <p className="font-semibold text-white mb-1">Next Steps:</p>
                                <ol className="list-decimal list-inside space-y-1">
                                    <li>Check your email inbox</li>
                                    <li>Click the verification link</li>
                                    <li>Log in to your account</li>
                                </ol>
                            </div>
                        </div>
                    </div>
                    <p className="text-slate-400 text-sm mb-6">
                        Didn't receive the email?{' '}
                        <Link to="/resend-verification" className="text-blue-400 hover:underline">
                            Resend verification email
                        </Link>
                    </p>
                    <Link
                        to="/login"
                        className="block w-full bg-gradient-to-r from-blue-500 to-purple-600 hover:from-blue-600 hover:to-purple-700 text-white font-semibold py-3 rounded-lg transition-all duration-300"
                    >
                        Go to Login
```

```
                      </Link>
                  </div>
              ) : (
                  // Keep your EXISTING signup form code here - don't delete it!
                  // Just wrap it in <>...</>
                  <>
                     {/* All your existing form code */}
                  </>
              )}
          </div>
      </div>
  );
```

---

4. Restart Servers (1 minute)

Backend:

```
cd backend
# Ctrl+C to stop if running
npm start
```

Frontend:

```
cd frontend
# Ctrl+C to stop if running
npm start
```

---

5. Test Everything (5 minutes)

Test 1: Email Verification

1. Go to http://localhost:3000/signup
2. Create a new account
3. Should see "Check Your Email!" message
4. Check your email inbox
5. Click the verification link
6. Should see "Email Verified!" page
7. Auto-redirects to login

Test 2: Login Blocking

1. Sign up but DON'T click verification link
2. Try to login
3. Should see error: "Please verify your email before logging in"

Test 3: Resend Verification

1. Go to http://localhost:3000/resend-verification
2. Enter your email
3. Click "Send Verification Email"
4. Check inbox for new email

Test 4: Transaction Emails

1. Login with verified account
2. Go to Trade page
3. Buy some crypto (e.g., 0.1 BTC)
4. Check email for transaction confirmation
5. All details should be correct

✅ Success Checklist

Verification works if:

- □ Signup shows "Check Your Email" message
- □ Verification email received (check spam!)
- □ Click link → "Email Verified!" page shows
- □ Welcome email received
- □ Can login after verification
- □ Cannot login before verification

Transaction emails work if:

- □ Buy transaction → email received
- □ Sell transaction → email received
- □ Email shows correct symbol, quantity, price
- □ Email shows transaction type (BUY/SELL)

🐛 Troubleshooting

"Failed to send verification email"

Check:

- Email credentials in `.env` are correct
- Gmail App Password is 16 characters (remove spaces)
- 2FA is enabled on Gmail account
- App Password is fresh (regenerate if old)

Test email sending:

```
cd backend
node
const { sendVerificationEmail } = require('./src/utils/emailService');
await sendVerificationEmail ('test@email.com', 'Test', 'test-token-123');
# Should see "Email sent successfully"
```

"Email not received"

Check:

- Spam/junk folder
- Promotions tab (Gmail)
- Email address is correct
- Backend logs for errors

- Email service credentials

"Verification link expired"

- Links expire after 24 hours
- Go to /resend-verification
- Enter email and request new link

"Login still blocked after verification"

Check:

- Email was actually verified (check database: `isEmailVerified: true`)
- Logout and login again
- Clear browser cache/cookies
- Check browser console for errors

🎯 Quick Commands

Check if email verified in database:

```
# In MongoDB shell
mongo
use myden
db.users.findOne({ email: "your@email.com" }, { isEmailVerified: 1 })
# Should show: isEmailVerified: true
```

Manually verify an email (for testing):

```
# In MongoDB shell
db.users.updateOne(
  { email: "your@email.com" },
  { $set: { isEmailVerified: true } }
)
```

View all unverified users:

```
db.users.find({ isEmailVerified: false }, { email: 1, firstName: 1 })
```

📦 Files Created/Modified

Backend (5 files)

1. ✅ `backend/src/utils/emailService.js` - NEW
2. ✅ `backend/src/controllers/authController.js` - MODIFIED
3. ✅ `backend/src/routes/auth.js` - MODIFIED
4. ✅ `backend/src/routes/portfolio.js` - MODIFIED
5. ✅ `backend/.env.example` - MODIFIED

Frontend (2 files + 1 to modify)

1. ✅ `frontend/src/pages/VerifyEmail.jsx` - NEW

2. ✅ `frontend/src/pages/ResendVerification.jsx` - NEW
3. ⏳ `frontend/src/pages/Signup.jsx` - NEEDS SMALL UPDATE
4. ⏳ `frontend/src/App.js` - NEEDS ROUTES ADDED

Documentation (2 files)

1. ✅ `EMAIL_FEATURES_COMPLETE.md` - Details
2. ✅ `QUICK_SETUP_GUIDE.md` - This file

🎉 You're Done!

Once you complete all 5 steps above, you'll have:

- ✅ Full email verification system
- ✅ Transaction email notifications
- ✅ Professional email templates
- ✅ Secure account activation
- ✅ Audit trail for all transactions

Time to complete: ~15-20 minutes

Next: See `EMAIL_FEATURES_COMPLETE.md` for detailed documentation!

Need Help?

- Check the detailed docs: `EMAIL_FEATURES_COMPLETE.md`
- Review error logs in terminal
- Test email sending separately
- Verify .env configuration