

Phase 4B Implementation Checklist

Pre-Implementation (5 minutes)

- ☐ Backup your current project
- ☐ Ensure you have Perplexity API key ready
- ☐ Verify Node.js version (16+) and yarn installation
- ☐ Check database is accessible (PostgreSQL/Supabase)

Core Installation (10 minutes)

1. File Extraction & Copy

```
# Extract the package
unzip PHASE-4B-COMPLETE-PACKAGE.zip

# Copy files to your project
cp -r PHASE-4B-FILES/* /path/to/your/project/
```

- ☐ All files copied successfully
- ☐ No file conflicts reported

2. Dependencies Installation

```
cd your-project
yarn add node-cron axios
```

- ☐ Dependencies installed without errors
- ☐ package.json updated with new dependencies

3. Environment Configuration

```
# Add to your .env file
cat PHASE-4B-FILES/.env.example >> .env

# Or manually add these variables:
echo "FEATURE_PHASE4B_ENABLED=false" >> .env
echo "FEATURE_TOPIC_RESEARCH=false" >> .env
echo "FEATURE_AUTO_PUBLISHING=false" >> .env
echo "PERPLEXITY_API_KEY=your_key_here" >> .env
```

- ☐ Environment variables added
- ☐ Perplexity API key configured
- ☐ All feature flags set to `false` initially

4. Database Schema Update

```
# Apply new schema
npx prisma db push

# Generate updated client
npx prisma generate
```

- [] Schema applied successfully
- [] No migration errors
- [] Prisma client regenerated

Testing & Verification (10 minutes)

5. Build Test

```
# Test TypeScript compilation
npx tsc --noEmit

# Test Next.js build
npm run build
```

- [] No TypeScript errors
- [] Build completes successfully
- [] No runtime errors during build

6. API Endpoint Tests

```
# Start development server
npm run dev

# Test in another terminal:
curl http://localhost:3000/api/phase4b/topics/research
# Should return: {"error":"Topic research feature is disabled"}
```

- [] All API endpoints return proper disabled messages
- [] No 404 or 500 errors

7. Feature Flag Testing

```
# Enable Phase 4B
export FEATURE_PHASE4B_ENABLED=true
export FEATURE_TOPIC_RESEARCH=true

# Restart server and test
curl -X POST http://localhost:3000/api/phase4b/topics/research \
  -H "Content-Type: application/json" \
  -d '{"category":"london_travel","locale":"en"}'
```

- [] API responds when feature is enabled
- [] Perplexity API integration working (if key is valid)
- [] Database operations working

Progressive Enablement (15 minutes)

8. Enable Core Features

```
# Enable basic pipeline
export FEATURE_PHASE4B_ENABLED=true
export FEATURE_PIPELINE_MONITORING=true
```

- [] Admin dashboard accessible at /admin
- [] Pipeline status displays correctly
- [] No JavaScript console errors

9. Enable Topic Research

```
export FEATURE_TOPIC_RESEARCH=true
export FEATURE_TOPIC_APPROVAL_WORKFLOW=true
```

- [] Topic research button works
- [] Topics are saved to database
- [] Topic manager interface functional

10. Enable Content Generation

```
export FEATURE_AUTO_CONTENT_GENERATION=true
export FEATURE_DRAFT_MANAGEMENT=true
```

- [] Content generation API works
- [] Generated content saves as drafts
- [] Content appears in admin interface

11. Enable Publishing & SEO

```
export FEATURE_AUTO_PUBLISHING=true
export FEATURE_SEO_AUTOMATION=true
export FEATURE_SEO_AUDIT_GATE=true
```

- [] Publishing workflow functional
- [] SEO audits run correctly
- [] Quality gates work as expected

Production Deployment (10 minutes)

12. Environment Preparation

```
# Production environment variables
export NODE_ENV=production
export FEATURE_PHASE4B_ENABLED=true
# Set other flags as needed

# Build for production
npm run build
```

- ☐ Production build successful
- ☐ Environment variables properly set
- ☐ No build warnings or errors

13. Database Migration

```
# In production environment
npx prisma migrate deploy
npx prisma generate
```

- ☐ Migration applied to production DB
- ☐ No data loss or conflicts
- ☐ Application connects successfully

14. Smoke Tests

```
# Test critical paths
curl https://your-domain.com/api/phase4b/topics/research
curl https://your-domain.com/api/phase4b/content/generate
```

- ☐ All APIs respond correctly
- ☐ Authentication working (if applicable)
- ☐ Admin interfaces load properly

Post-Implementation (5 minutes)

15. Monitoring Setup

- ☐ Check application logs for errors
- ☐ Verify cron jobs are scheduling correctly
- ☐ Test manual pipeline run
- ☐ Confirm database is recording activities

16. Documentation

- ☐ Update your project README
- ☐ Document custom configuration changes
- ☐ Share access credentials with team
- ☐ Plan content strategy and topics

Troubleshooting Checklist

If something isn't working:

- [] Check environment variables are loaded
- [] Verify API keys are correct
- [] Ensure database connection is working
- [] Check browser console for JavaScript errors
- [] Review application logs for server errors
- [] Confirm all dependencies are installed
- [] Test with feature flags disabled/enabled

Success Criteria

- [] ☒ All feature flags working correctly
- [] ☒ Topic research generating results
- [] ☒ Content generation creating drafts
- [] ☒ SEO audits scoring content
- [] ☒ Admin dashboard showing real data
- [] ☒ No errors in production logs
- [] ☒ Pipeline can run end-to-end

Rollback Plan

If major issues occur:

1. Set `FEATURE_PHASE4B_ENABLED=false`
2. Restart application
3. Restore database from backup if needed
4. Remove Phase 4B files if necessary

Estimated Total Time: 45-60 minutes

Complexity Level: Intermediate

Risk Level: Low (all features disabled by default)