

# Team 3 PRD - Knowledge Hub

## Team Features (8 Features - 1 per Developer)

1. Content Creation & Rich Text Editor
  2. Content Categorization & Tagging System
  3. Advanced Search & Filtering
  4. Content Publishing Workflow
  5. File Upload & Attachment Management
  6. Content Analytics & Performance Tracking
  7. Public Content Display & Browse
  8. Content Moderation & Admin Tools
- 

## Team Overview & Scope

### Team Composition

- **Team Leader:** 1 Senior Developer
- **Developers:** 8 Junior Developers
- **Duration:** 8 weeks

### Primary Responsibilities

- Complete knowledge management system
- Content creation and editing tools
- Advanced search and discovery
- File upload and attachment handling
- Publishing workflow and moderation
- Public content display and browsing
- Analytics and performance tracking

## Dependencies

- **Team 1:** User authentication and profile system
- **Team 4:** Shared UI components and file storage infrastructure
- **External:** File storage service, search indexing

## Success Criteria

- Content creation completion rate > 90%
  - Search results relevancy score > 85%
  - Public content engagement rate > 70%
  - Content publishing workflow completion < 5 minutes
- 

# Detailed Feature Requirements

## Feature 1: Content Creation & Rich Text Editor

### User Stories:

- As a content creator, I want to write articles so that I can share knowledge with the community
- As a content creator, I want a rich text editor so that I can format content professionally
- As a content creator, I want to save drafts so that I can work on content over time

### Acceptance Criteria:

- Rich text editor with formatting options (bold, italic, headers, lists, links)
- Auto-save functionality every 30 seconds
- Draft management and version history
- Image insertion and positioning
- Link embedding and preview
- Content templates for different types

### Technical Specifications:

- Rich text editor integration (TinyMCE or similar)
- Auto-save with conflict resolution
- Draft storage and retrieval
- Image upload integration
- Content validation and sanitization

## Feature 2: Content Categorization & Tagging System

### User Stories:

- As a content creator, I want to categorize content so that users can find relevant information
- As a user, I want to browse by category so that I can find specific types of content
- As an admin, I want to manage tags so that content organization remains consistent

### Acceptance Criteria:

- 5 main categories: Research Papers, Industry News, Case Studies, Toolkits & Guides, Webinars
- Flexible tagging system with auto-suggestions
- Tag management and merging capabilities
- Category-based content filtering
- Tag popularity and usage tracking

### Technical Specifications:

- Category enumeration and validation
- Tag autocomplete with fuzzy matching
- Tag normalization and deduplication
- Category and tag analytics
- Bulk tagging operations

## Feature 3: Advanced Search & Filtering

### User Stories:

- As a user, I want to search for content so that I can find relevant information quickly
- As a user, I want to filter results so that I can narrow down to specific topics
- As a public visitor, I want to search content so that I can evaluate the platform

### Acceptance Criteria:

- Full-text search across title, description, content, and tags
- Filter by category, type, date range, author
- Sort by relevance, date, popularity
- Search suggestions and autocomplete
- Advanced search with multiple criteria
- Search analytics and popular queries

### Technical Specifications:

- Full-text search implementation
- Search indexing and optimization
- Filter combination logic
- Search result ranking algorithm
- Performance optimization for large datasets

## Feature 4: Content Publishing Workflow

### User Stories:

- As a content creator, I want to publish content so that it becomes available to users
- As a content creator, I want to schedule publishing so that content goes live at optimal times
- As an admin, I want to review content so that quality standards are maintained

### Acceptance Criteria:

- Draft → Published workflow
- Optional admin review step

- Scheduled publishing capability
- Content preview before publishing
- Publishing status tracking
- Bulk publishing operations

#### Technical Specifications:

- Status state machine (draft, review, scheduled, published)
- Scheduled job system for timed publishing
- Content preview generation
- Publishing notification system
- Audit trail for publishing actions

## Feature 5: File Upload & Attachment Management

#### User Stories:

- As a content creator, I want to attach files so that users can download additional resources
- As a content creator, I want to upload images so that my content is visually engaging
- As a user, I want to download attachments so that I can access detailed resources

#### Acceptance Criteria:

- Support for multiple file types (PDF, DOCX, images, videos)
- Drag-and-drop file upload interface
- File size limits and validation
- Multiple attachments per content item
- File preview and thumbnail generation
- Download tracking and analytics

#### Technical Specifications:

- File upload with progress indicators
- File type validation and security scanning
- Cloud storage integration (Supabase Storage)
- Thumbnail generation for images and PDFs
- Download link generation with expiration

- File metadata extraction and storage

## Feature 6: Content Analytics & Performance Tracking

### User Stories:

- As a content creator, I want to see content performance so that I can understand what resonates
- As an admin, I want content analytics so that I can optimize the knowledge hub
- As a content creator, I want to track downloads so that I can measure resource value

### Acceptance Criteria:

- View counts and trends over time
- Download statistics for attachments
- Search appearance frequency
- Geographic breakdown of readers
- Popular content rankings
- Engagement metrics and time spent

### Technical Specifications:

- Analytics data collection and aggregation
- Privacy compliant tracking
- Dashboard with interactive charts
- Data export functionality
- Real-time vs. batch analytics processing
- Performance optimization for analytics queries

## Feature 7: Public Content Display & Browse

### User Stories:

- As a public visitor, I want to browse content so that I can see the value of the platform
- As a user, I want to discover new content so that I can stay informed
- As a content creator, I want my content displayed professionally so that it attracts readers

#### Acceptance Criteria:

- Professional content card layout
- Content detail pages with full formatting
- Related content suggestions
- Social sharing capabilities
- Mobile-responsive reading experience
- SEO optimization for public content

#### Technical Specifications:

- Server-side rendering for SEO
- Content card component library
- Related content algorithm
- Social media meta tags
- Mobile-optimized reading interface
- CDN integration for fast loading

## Feature 8: Content Moderation & Admin Tools

#### User Stories:

- As an admin, I want to moderate content so that quality standards are maintained
- As an admin, I want to manage content lifecycle so that outdated information is updated
- As an admin, I want to see content reports so that I can make data-driven decisions

#### Acceptance Criteria:

- Content review and approval interface
- Bulk content operations (publish, archive, delete)
- Content reporting and flagging system
- Content lifecycle management
- Admin analytics dashboard
- Content quality scoring

#### Technical Specifications:

- Admin dashboard with content queue

- Bulk operation APIs
  - Content flagging and review workflow
  - Automated content quality checks
  - Admin analytics and reporting
  - Content archival and cleanup processes
- 

## Technical Specifications

### API Endpoints to Build

#### Content Management:

GET /api/content (search/browse)

POST /api/content (create)

GET /api/content/:id (view)

PUT /api/content/:id (update)

DELETE /api/content/:id (delete)

POST /api/content/:id/publish

POST /api/content/:id/archive

#### File Management:

POST /api/content/:id/upload

GET /api/content/:id/files

DELETE /api/files/:id

GET /api/files/:id/download

#### Search:

GET /api/search/content

GET /api/search/suggestions

GET /api/search/popular

#### Analytics:

GET /api/content/:id/analytics

GET /api/analytics/content-overview



GET /api/analytics/popular-content

Admin:

GET /api/admin/content/pending

POST /api/admin/content/:id/approve

POST /api/admin/content/:id/reject

GET /api/admin/analytics/content

## Database Schemas

-- Content items table

```
CREATE TABLE content_items (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  title VARCHAR(255) NOT NULL,  
  description TEXT NOT NULL,  
  content TEXT,  
  category content_category_enum NOT NULL,  
  type content_type_enum NOT NULL,  
  status content_status_enum DEFAULT 'draft',  
  author_id UUID REFERENCES users(id) ON DELETE SET NULL,  
  published_at TIMESTAMP,  
  external_url VARCHAR(500),  
  download_url VARCHAR(500),  
  thumbnail_url VARCHAR(500),  
  tags TEXT[] DEFAULT '{}',  
  metadata JSONB DEFAULT '{}',  
  created_at TIMESTAMP DEFAULT NOW(),  
  updated_at TIMESTAMP DEFAULT NOW()  
);
```

-- Content attachments table

```
CREATE TABLE content_attachments (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  content_id UUID REFERENCES content_items(id) ON DELETE CASCADE,
```

```

filename VARCHAR(255) NOT NULL,
file_url VARCHAR(500) NOT NULL,
file_size BIGINT,
file_type VARCHAR(100),
upload_date TIMESTAMP DEFAULT NOW()
);

-- Content analytics table
CREATE TABLE content_analytics (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  content_id UUID REFERENCES content_items(id) ON DELETE CASCADE,
  event_type VARCHAR(50) NOT NULL, -- view, download, share
  user_id UUID REFERENCES users(id) ON DELETE SET NULL,
  session_id VARCHAR(255),
  ip_address INET,
  user_agent TEXT,
  referrer VARCHAR(500),
  created_at TIMESTAMP DEFAULT NOW()
);

-- Content tags table
CREATE TABLE content_tags (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  name VARCHAR(100) UNIQUE NOT NULL,
  usage_count INTEGER DEFAULT 0,
  created_at TIMESTAMP DEFAULT NOW()
);

```

## Integration Requirements

### File Storage:

- Supabase Storage or AWS S3 integration
- File upload with progress tracking
- Thumbnail generation service

- CDN configuration for fast delivery

Search Engine:

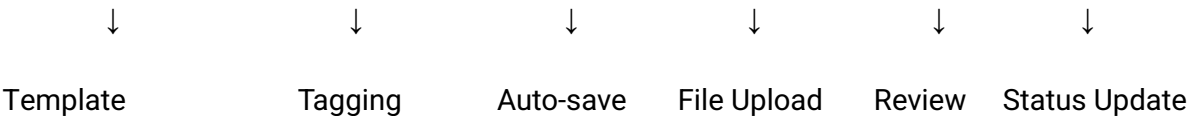
- PostgreSQL full-text search
  - Search indexing optimization
  - Elasticsearch integration (optional)
  - Search analytics tracking
- 

# User Interface Requirements

## Wireframes & User Flows

Content Creation Flow:

New Content → Category Selection → Rich Editor → Attachments → Preview → Publish



Content Browser Interface:

- Search bar with filters
- Category navigation
- Content grid with cards
- Detailed content view
- Related content sidebar

Admin Moderation Interface:

- Content review queue
- Bulk action controls
- Content analytics dashboard
- Quality scoring display

## Responsive Design Requirements

- Mobile-optimized content reading
- Touch-friendly content creation
- Responsive file upload interface
- Mobile admin tools

## Accessibility Considerations

- Screen reader compatible editor
  - Keyboard navigation support
  - High contrast reading mode
  - Alt text for all images and attachments
- 

# Implementation Timeline

## Week 1: Foundation & Content Structure

- Database schema creation
- Content model definition
- Rich text editor research
- File upload planning

## Week 2: Content Creation System

- Rich text editor integration
- Basic content CRUD operations
- Category and tagging system
- Auto-save functionality

## Week 3: File Management & Search

- File upload implementation

- Attachment management system
- Basic search functionality
- Content categorization

## Week 4: Publishing & Display

- Publishing workflow implementation
- Public content display
- Content detail pages
- Mobile-responsive interface

## Week 5: Advanced Search & Analytics

- Advanced search with filters
- Content analytics system
- Performance tracking
- Search optimization

## Week 6: Admin Tools & Moderation

- Admin moderation interface
- Content review workflow
- Bulk operations
- Quality management tools

## Week 7: Optimization & Testing

- Performance optimization
- Search result tuning
- Mobile interface refinement
- Integration testing

## Week 8: Testing & Deployment

- User acceptance testing
- Content migration tools

- Production deployment
  - Documentation completion
- 

## Testing Strategy

### Unit Testing

- Content CRUD operations
- Search and filtering logic
- File upload and validation
- Publishing workflow states

### Integration Testing

- Rich text editor functionality
- File storage integration
- Search indexing accuracy
- Analytics data collection

### User Acceptance Testing

- Content creation and editing flow
- Search and discovery experience
- File upload and download
- Admin moderation tools

### Performance Testing

- Search response times
- File upload performance
- Content page loading speed
- Analytics query optimization

---

# Assumptions & Constraints

## Technical Limitations

- Rich text editor capabilities
- File size and type restrictions
- Search index update frequency
- Analytics data retention period

## Business Constraints

- Content moderation requirements
- Copyright and legal compliance
- SEO optimization needs
- Multi-language support (future)

## Security Requirements

- File upload security scanning
- Content sanitization
- User permission validation
- Analytics privacy compliance

## Third-Party Dependencies

- File storage service reliability
- CDN performance and costs
- Rich text editor licensing
- Search service limitations