PROBLEM SCOPING & PROJECT PROPOSAL

TruthLens: AI-Powered Universal Document Fraud Detection System

---

PART A: PROBLEM SCOPING (15%)

1. PROBLEM STATEMENT

1.1 The Global Document Fraud Crisis

Magnitude:

- **$5 trillion** lost annually to document fraud globally
- **85%** of employers catch resume fraud during hiring
- **$308 billion** in insurance fraud annually
- **30%** of loan applications contain falsified income documents

Types of Documents Affected:

1. **Financial:** Bank statements, invoices, tax returns, payslips
2. **Educational:** Degrees, transcripts, certificates
3. **Employment:** Offer letters, experience certificates
4. **Identity:** Aadhaar cards, passports, driving licenses
5. **Legal:** Contracts, affidavits, notarized documents
6. **Medical:** Prescriptions, test reports, medical bills

1.2 Current Verification Methods

Manual Inspection:

- **Cost:** $200-500 per document (forensic expert)
- **Time:** 2-4 hours per document
- **Scalability:** Cannot handle millions of daily verifications
- **Accuracy:** Subject to human error and fatigue

Basic Digital Checks:

- **Metadata analysis:** Easily spoofed (change file properties)
- **Template matching:** Fails on non-standard formats
- **Simple OCR:** Can't detect visual manipulation

Commercial Solutions:

- **Adobe CAI:** Requires creation-time embedding (useless for legacy docs)
- **Truepic:** Only for new photo capture
- **Background verification services:** Slow (days-weeks), expensive ($50-200/doc)

Gap: No accessible, affordable, real-time solution for comprehensive document authentication.

1.3 Why Existing Methods Fail

Sophisticated Fraud Techniques:

1. **Visual Manipulation:** Photoshop, GIMP (alter text, numbers, images)
2. **Copy-Paste Fraud:** Duplicate signatures, logos from authentic docs
3. **Font Mixing:** Combine text from multiple sources
4. **Semantic Fraud:** Mathematically incorrect but visually plausible

Single-Method Limitations:

Method	Catches	Misses
Visual forensics (ELA)	Photoshopped content	Logical errors, copy-paste without re-save
Template matching	Format violations	Content manipulation within correct format
OCR + text analysis	Wrong calculations	Visual tampering
Metadata checks	Stripped metadata	Content forgery with intact metadata

Key Insight: Multi-modal fraud requires multi-modal detection.

1.4 Target Users and Use Cases

Primary Users:

1. **Banks/NBFCs** (50,000+ institutions in India)
    - Loan applications (income proof verification)
    - KYC compliance (address proof, identity docs)
    - Vendor invoices (accounts payable)
2. **HR Departments** (100,000+ companies)
    - Resume verification (degree, experience certificates)
    - Background checks
    - Offer letter validation
3. **Educational Institutions** (40,000+ universities/colleges)
    - Transfer certificate validation
    - Admission document verification
    - Scholarship applications
4. **Legal Professionals** (Lawyers, courts)
    - Evidence authentication
    - Contract verification
    - Due diligence

5. **Insurance Companies**
    - Claim documentation (medical bills, receipts)
    - Policy applications
    - Fraud investigation
6. **Individuals**
    - Landlords (verifying tenant documents)
    - Freelancers (client contract verification)
    - Job seekers (offer letter validation)

Market Size: 10+ million potential users in India alone, 100+ million globally.

1.5 Success Criteria

Technical Metrics:

- Accuracy ≥ 90% on real-world documents
- Processing time < 5 seconds per document
- False positive rate < 5% (avoid wrongly flagging authentic docs)
- False negative rate < 10% (minimize missed frauds)

User Adoption Metrics:

- 1,000+ active users by Month 8
- 10,000+ documents verified by Month 12
- Average user satisfaction ≥ 4/5 stars

Research Output:

- 2 peer-reviewed publications (ICDAR, CVPR)
- Open-source codebase (1,000+ GitHub stars target)
- Public dataset release (5,000+ samples)

Business Viability (Post-Graduation):

- Freemium model: 10 free verifications/month, $20/month for unlimited
- Target revenue: $10K/month within 6 months of launch
- API partnerships with 5+ enterprises

---

2. SCOPE DEFINITION

2.1 In-Scope

Document Types:

- ✅ Financial documents (statements, invoices, receipts, tax forms)

- ✅ Educational certificates (degrees, transcripts, completion certificates)
- ✅ Employment documents (offer letters, payslips, experience certificates)
- ✅ Identity documents (Aadhaar, PAN, passport scans)
- ✅ Legal documents (contracts, agreements)
- ✅ Medical documents (prescriptions, test reports)

Fraud Detection Methods:

- ✅ Compression artifacts (ELA)
- ✅ Duplication detection (Copy-Move)
- ✅ Font inconsistencies (OCR-based analysis)
- ✅ Semantic validation (VLM + business rules)
- ✅ Mathematical verification (balance calculations, tax computations)

Deployment:

- ✅ Web application (public access)
- ✅ REST API (enterprise integration)
- ✅ Batch processing (upload multiple docs)
- ✅ Report generation (PDF with evidence)

2.2 Out-of-Scope (Future Work)

Beyond M.Tech Timeline:

- ❌ Video authentication (document shown on screen recording)
- ❌ Blockchain integration (provenance tracking)
- ❌ Mobile application (iOS/Android apps)
- ❌ Real-time camera capture with live analysis
- ❌ Multi-language support (focus English/Hindi initially)
- ❌ Handwritten document analysis (printed/typed documents only)

2.3 Assumptions and Constraints

Assumptions:

- Documents are in image format (JPEG, PNG, PDF → convert to image)
- Resolution ≥ 600×400 pixels (minimum for OCR accuracy)
- Text is machine-printed (not handwritten)
- Documents are in English (or use English numerals for amounts)

Constraints:

- **Budget:** $0-10/month (Colab Pro if needed)
- **Time:** 12 months (365 days × 1-2 hours/day)
- **Team:** Solo project (no collaborators required)
- **Data:** No access to real fraud cases initially (synthetic generation required)

---

3. RISK ANALYSIS

3.1 Technical Risks

Risk	Likelihood	Impact	Mitigation
OCR accuracy < 80%	Medium	High	Use multiple OCR engines (Tesseract + PaddleOCR), ensemble outputs
VLM hallucination	High	Medium	Cross-validate with rule-based checks, confidence thresholding
Copy-move false positives on text	High	Medium	Semantic segmentation, stricter thresholds (implemented)
Dataset size insufficient	Medium	High	Synthetic generation (5K docs), crowdsource via web app (1K+ real)
Processing time > 10s	Low	Medium	Optimize algorithms, parallel processing, GPU acceleration

3.2 Project Management Risks

Risk	Likelihood	Impact	Mitigation
Scope creep (adding too many features)	High	High	Strict prioritization, MVP-first approach
Thesis writing delays	Medium	High	Incremental writing (update daily/weekly, not last-minute)
Paper rejection	Medium	Medium	Target multiple venues, incorporate reviewer feedback
User adoption < 1,000	Medium	Medium	Social media marketing, ProductHunt launch, university partnerships

3.3 Ethical and Legal Risks

Risk	Likelihood	Impact	Mitigation
Privacy violation (user documents exposed)	Low	High	No data retention policy, on-premise deployment option, HTTPS encryption
Misuse (fraudsters testing their fakes)	High	Low	Rate limiting, CAPTCHA, monitoring, watermarking analysis results

Risk	Likelihood	Impact	Mitigation
False accusations (innocent flagged as fraud)	Medium	High	Confidence scores (not binary), human-in-the-loop recommendation
Bias (discriminating against certain document types/regions)	Medium	Medium	Diverse training data, fairness audits, geographic representation

---

PART B: PROJECT PROPOSAL (30%)

4. PROPOSED SOLUTION: TRUTHLENS

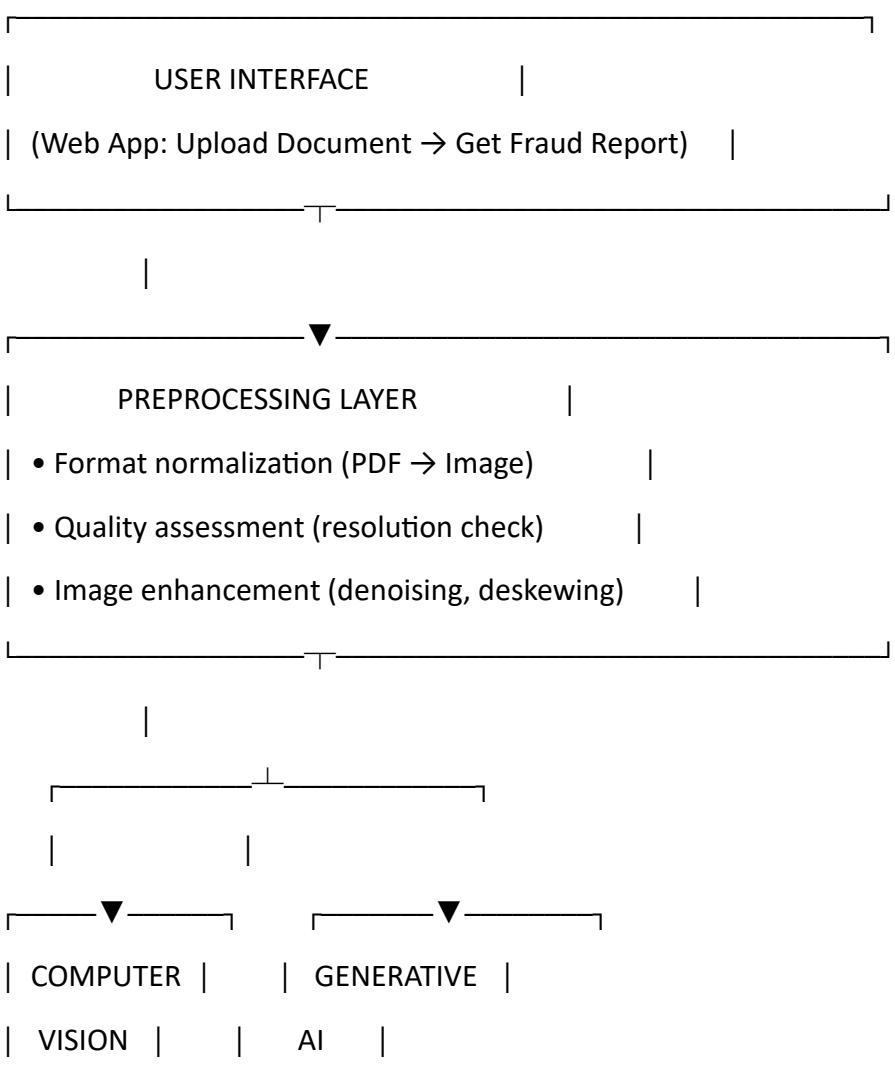
4.1 System Overview

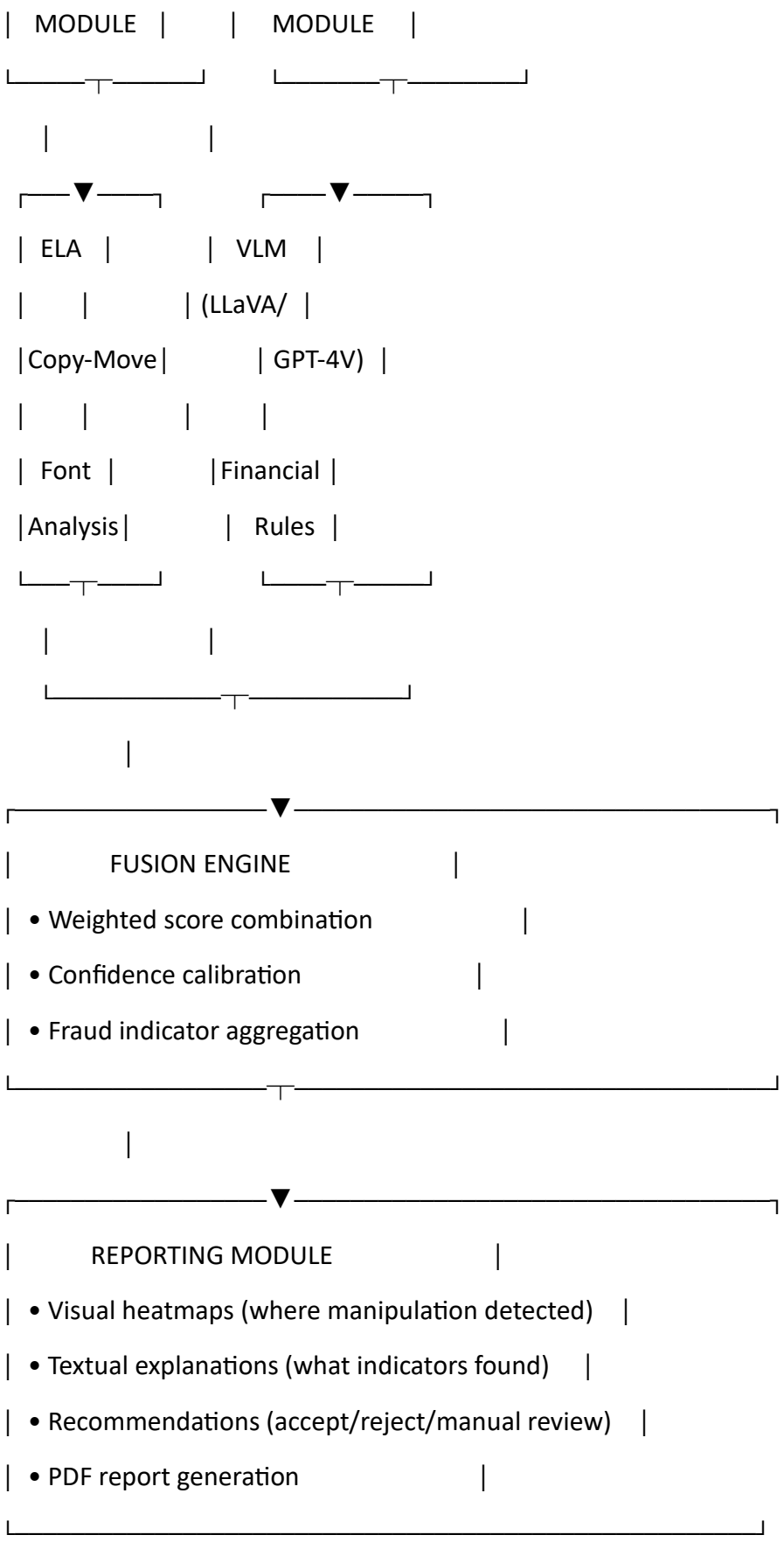
TruthLens is a multimodal AI system that authenticates documents by analyzing:

1. **Visual integrity** (Computer Vision)
2. **Textual consistency** (OCR + Font Analysis)
3. **Semantic plausibility** (Generative AI + Financial Rules)

Key Innovation: First system to combine all three modalities for comprehensive document fraud detection.

4.2 Architecture





4.3 Detailed Module Design

4.3.1 Computer Vision Module

Component 1: ELA Detector

- **Input:** Document image (JPEG preferred)

- **Process:**
    1. Recompress at quality=95
    2. Compute pixel-wise difference
    3. Generate heatmap (intensity = manipulation likelihood)
- **Output:** Fraud score (0-100), heatmap image
- **Status:** ✅ Implemented (Day 1)

Component 2: Copy-Move Detector

- **Input:** Document image
- **Process:**
    1. Divide into 32×32 blocks
    2. Compare blocks (normalized cross-correlation)
    3. Flag pairs with similarity > 0.98 and distance > 100px
- **Output:** Duplicate pair count, annotated image
- **Status:** ✅ Implemented (Day 2)

Component 3: Font Analyzer

- **Input:** Document image
- **Process:**
    1. OCR text extraction (Tesseract)
    2. Analyze font heights, spacing
    3. Group similar fonts, identify isolated variations
- **Output:** Font variation count, suspicious region list
- **Status:** ✅ Implemented (Day 3)

4.3.2 Generative AI Module

Component 1: Document Classifier

- **Technology:** CLIP or fine-tuned ResNet
- **Purpose:** Identify document type (invoice, statement, certificate)
- **Benefit:** Enable type-specific validation rules
- **Timeline:** Week 2-3

Component 2: Vision-Language Model (VLM)

- **Technology:** LLaVA (open-source) or GPT-4V (if budget allows)
- **Purpose:**
    - Extract structured data (amounts, dates, names)

    - Validate calculations (balance = deposits - withdrawals?)
    - Assess plausibility (is $500K salary for junior role realistic?)
- **Timeline:** Week 3-4

Component 3: Explanation Generator

- **Technology:** GPT-4 (text-only, cheaper than GPT-4V)
- **Purpose:** Convert technical findings to human-readable report
- **Example:**
- Input: {ela_score: 45, copymove_duplicates: 3, font_variations: 5}Output: "This document shows moderate fraud risk. Compression analysis reveals editing in the balance field. Three duplicated regions suggest copied signatures. Five different fonts indicate text from multiple sources."
- **Timeline:** Week 4

4.3.3 Financial Validation Module

Component 1: Rule Engine

- **Technology:** Custom Python rules + JSON config files
- **Purpose:** Encode domain knowledge
- **Examples:**
    - Bank statement: opening + deposits - withdrawals = closing
    - Invoice: subtotal + tax = total
    - Salary slip: gross - deductions = net
- **Timeline:** Week 5-6

Component 2: Anomaly Detector

- **Technology:** Isolation Forest (unsupervised ML)
- **Purpose:** Flag statistically unusual values
- **Examples:**
    - Salary 5× above role average
    - Invoice amount 10× typical for vendor
- **Timeline:** Week 6-7

Component 3: External Validation (API Integrations)

- **Bank routing numbers:** Verify against RBI database (if available)
- **Company registration:** Check MCA database (Ministry of Corporate Affairs)
- **PAN validation:** Check format (ABCDE1234F pattern)
- **Timeline:** Week 7-8 (if APIs accessible)

4.3.4 Fusion Engine

Weighted Combination:

```
combined_score = (0.40 × ela_score +
                  0.30 × copymove_score +
                  0.30 × font_score)


# Boost if multiple detectors agree (>50 threshold)
high_detectors = count(scores > 50)
if high_detectors >= 2:
   combined_score *= 1.3  # 30% boost
```

Confidence Calculation:

- **VERY HIGH:** All detectors agree (all >50 or all <20)
- **HIGH:** 2 out of 3 agree
- **MEDIUM:** Mixed signals
- **LOW:** Contradictory results (e.g., ELA=80, others=10)

Timeline: Week 8 (after all modules integrated)

---

5. IMPLEMENTATION PLAN

5.1 Development Timeline (12 Months)

Month 1 (Days 1-30): Computer Vision Foundation

- Week 1 (Days 1-7): ✅ COMPLETE
  - ELA detection ✅
  - Copy-move detection ✅
  - Font analysis ✅
  - Multimodal integration ✅
- Week 2 (Days 8-14): CV Optimization
  - Semantic segmentation (separate text from images)
  - Parameter tuning (reduce false positives)
  - Performance optimization (GPU acceleration)
- Week 3 (Days 15-21): Advanced CV
  - Noise analysis (detect splicing)
  - Metadata extraction (EXIF, creation date)

    - Quality assessment (resolution, blur detection)
- Week 4 (Days 22-30): CV Testing
    - Expand synthetic dataset (1,000 → 5,000 docs)
    - Ablation studies (contribution of each method)
    - Cross-validation

Month 2 (Days 31-60): Generative AI Integration

- Week 5 (Days 31-37): VLM Setup
    - LLaVA installation and testing
    - Document-specific prompt engineering
    - API rate limiting and caching
- Week 6 (Days 38-44): Semantic Validation
    - Text extraction pipeline (OCR → structured data)
    - Mathematical validation (balance checks)
    - Plausibility assessment (salary ranges, etc.)
- Week 7 (Days 45-51): Explanation Generation
    - GPT-4 integration for reports
    - Template design (fraud report format)
    - Multi-language support (English + Hindi)
- Week 8 (Days 52-60): Gen AI Testing
    - Test on 1,000 synthetic docs
    - Measure hallucination rate
    - Fine-tune prompts

Month 3 (Days 61-90): Financial AI + Full Integration

- Week 9 (Days 61-67): Rule Engine
    - Encode validation rules (20+ rule types)
    - JSON configuration system
    - Rule conflict resolution
- Week 10 (Days 68-74): Anomaly Detection
    - Train Isolation Forest on synthetic data
    - Statistical baseline establishment
    - Threshold tuning
- Week 11 (Days 75-81): Complete Integration

    - Connect all modules (CV + Gen AI + Financial)
    - End-to-end testing
    - Performance profiling
- Week 12 (Days 82-90): System Testing
    - Integration tests (1,000 docs)
    - Load testing (100 concurrent users)
    - Bug fixes

Month 4 (Days 91-120): Web Application Development

- Week 13-14 (Days 91-105): Backend
    - FastAPI REST API
    - Database (PostgreSQL for user data)
    - Authentication (JWT tokens)
- Week 15-16 (Days 106-120): Frontend
    - React UI (document upload, report display)
    - Responsive design (mobile-friendly)
    - Interactive heatmap viewer

Month 5 (Days 121-150): Beta Testing

- Week 17-18 (Days 121-135): Private Beta
    - Invite 50 users (friends, faculty, online communities)
    - Collect feedback
    - Fix major bugs
- Week 19-20 (Days 136-150): Public Beta
    - Launch on ProductHunt, Reddit
    - Target 500 beta users
    - Monitor performance, gather metrics

Month 6 (Days 151-180): Dataset Collection + Paper 1

- Week 21-22 (Days 151-165): Real Data Collection
    - Users upload documents (with consent for research)
    - Manual labeling (authentic vs. fraud)
    - Target: 1,000 real documents
- Week 23-24 (Days 166-180): Paper 1 Writing
    - Title: "Multimodal Error Level Analysis for Financial Document Fraud Detection"

    - Venue: ICDAR 2025 (deadline ~March)
    - Content: CV methods (ELA, copy-move, font)

Month 7 (Days 181-210): Scaling + Paper 1 Revisions

- Week 25-26 (Days 181-195): System Scaling
    - Reach 1,000+ active users
    - Optimize for cost (caching, batching)
    - Monitor metrics
- Week 27-28 (Days 196-210): Paper Revisions
    - Incorporate co-author/advisor feedback
    - Additional experiments (ablation studies)
    - Submit to ICDAR

Month 8 (Days 211-240): Advanced Features

- Week 29-30 (Days 211-225): Batch Processing
    - Upload ZIP of 100+ documents
    - Bulk analysis with summary report
- Week 31-32 (Days 226-240): API Deployment
    - Enterprise API (for companies)
    - Usage-based pricing model
    - Documentation (Swagger/OpenAPI)

Month 9 (Days 241-270): Paper 2 + Thesis Drafting

- Week 33-34 (Days 241-255): Paper 2 Writing
    - Title: "TruthLens: A Vision-Language Approach to Universal Document Authentication"
    - Venue: CVPR 2026 or WACV 2026
    - Content: Full system (CV + Gen AI + Financial)
- Week 35-36 (Days 256-270): Thesis Chapters 1-4
    - Introduction, Literature Review, Problem Statement, Proposed Solution

Month 10 (Days 271-300): Thesis Completion

- Week 37-38 (Days 271-285): Thesis Chapters 5-6
    - Methodology, Implementation
- Week 39-40 (Days 286-300): Thesis Chapters 7-9
    - Results, Discussion, Conclusion

Month 11 (Days 301-330): Paper 2 + Thesis Revisions

- Week 41-42 (Days 301-315): Paper 2 Submission
    - Final experiments
    - Submit to CVPR/WACV
- Week 43-44 (Days 316-330): Thesis Revisions
    - Advisor feedback incorporation
    - External reviewer comments

Month 12 (Days 331-365): Final Polish + Defense

- Week 45-46 (Days 331-345): Thesis Finalization
    - Formatting (IEEE/Springer template)
    - Bibliography completion
    - Plagiarism check
- Week 47-48 (Days 346-360): Defense Preparation
    - Presentation slides (30-40 slides)
    - Demo video (5-minute system walkthrough)
    - Mock defenses (practice with peers)
- Week 49 (Days 361-365): Submission + Defense
    - Thesis submission to university
    - Final presentation
    - Viva voce

5.2 Weekly Time Allocation

Daily Schedule (6 days/week):

- Monday-Friday: 1-2 hours/day (coding, experiments)
- Saturday: 2-3 hours (weekly summary, planning)
- Sunday: Off (rest, casual reading)

Breakdown:

- Coding/Implementation: 60% (7-8 hours/week)
- Experiments/Testing: 20% (2-3 hours/week)
- Documentation/Writing: 15% (2 hours/week)
- Learning/Reading: 5% (30 min/week)

---

6. DELIVERABLES

6.1 Software Deliverables

1. **TruthLens Web Application** (Month 4)
    - Public URL (truthlens.app or similar)
    - User authentication
    - Document upload and analysis
    - Report generation and download
2. **REST API** (Month 8)
    - Endpoint: /api/v1/detect
    - Input: Image file (multipart/form-data)
    - Output: JSON with fraud score, indicators, explanations
    - Rate limit: 100 requests/hour (free tier)
3. **GitHub Repository** (Ongoing)
    - Complete source code (MIT license)
    - Documentation (README, API docs)
    - Docker deployment instructions
    - Target: 1,000+ stars by Month 12
4. **Dataset Release** (Month 6)
    - 5,000 synthetic documents (labeled)
    - 1,000 real documents (anonymized, consent obtained)
    - Hosted on Kaggle or HuggingFace Datasets

6.2 Research Deliverables

1. **Paper 1: CV Methods** (Month 7)
    - Venue: ICDAR 2025
    - Pages: 8-10 (conference format)
    - Content: ELA, copy-move, font analysis
2. **Paper 2: Full System** (Month 11)
    - Venue: CVPR 2026 or WACV 2026
    - Pages: 8-10 (conference format)
    - Content: Multimodal integration, real-world validation
3. **M.Tech Thesis** (Month 12)
    - Pages: 80-100
    - Format: IIIT Dharwad template
    - Defense presentation: 30-40 slides

6.3 Impact Metrics (Month 12 Targets)

- Users: 1,000+ (actual: track via analytics)
- Documents verified: 10,000+
- GitHub stars: 1,000+
- Paper citations: 5+ (within 1 year of publication)
- Media coverage: 3+ articles (tech blogs, news)

---

7. EVALUATION METRICS

7.1 Technical Performance

Accuracy (Primary Metric):

- Target: ≥ 90% on real-world documents
- Measurement: Confusion matrix (TP, TN, FP, FN)
- Baseline: 78% (vanilla ELA alone)

Speed:

- Target: < 5 seconds per document (end-to-end)
- Measurement: Average processing time over 1,000 docs
- Breakdown: CV (1s), Gen AI (2s), Fusion (1s), Reporting (1s)

False Positive Rate:

- Target: < 5% (minimize incorrect fraud flags)
- Impact: User trust (avoid crying wolf)

False Negative Rate:

- Target: < 10% (minimize missed frauds)
- Impact: Security (undetected frauds slip through)

7.2 User Satisfaction

Survey (post-verification):

- Ease of use: 1-5 stars
- Result accuracy: 1-5 stars
- Report clarity: 1-5 stars
- Would recommend?: Yes/No

Target: Average 4+/5 stars, 80%+ would recommend

7.3 Research Impact

Paper Acceptance:

- Paper 1: Accepted to ICDAR (or equivalent tier-2 venue)
- Paper 2: Accepted to CVPR/WACV (tier-1 venue)

Citations:

- 5+ citations within 12 months of publication
- 20+ citations within 24 months

Dataset Usage:

- 100+ downloads from Kaggle/HuggingFace
- 10+ projects using our dataset

---

8. BUDGET AND RESOURCES

8.1 Budget Breakdown

Item	Cost (Monthly)	Duration	Total
Google Colab Pro	$10	6 months (GPU needed for VLM)	$60
Domain name (truthlens.app)	$1	12 months	$12
Cloud hosting (DigitalOcean)	$5	6 months (after beta)	$30
TOTAL			**$102**

Alternatives (if $0 budget required):

- Colab Free tier (limited GPU hours, acceptable for experiments)
- GitHub Pages (free static hosting for landing page)
- Heroku free tier (for API deployment)
- **Feasibility:** Entire project doable with $0 if needed

8.2 Software Resources (All Free)

- Python 3.10+ (free, open-source)
- VS Code (free IDE)
- Git/GitHub (free version control)
- OpenCV, Tesseract, PyTorch (free libraries)
- LLaVA (free open-source VLM)
- Streamlit/Gradio (free web app frameworks)

8.3 Hardware Requirements

Development:

- Laptop (existing): Intel i5/i7 or Ryzen 5/7, 8GB+ RAM, sufficient for CV and rule-based modules
- GPU (optional): Not required for development, use Colab for VLM experiments

Deployment:

- Cloud VM: 2 vCPU, 4GB RAM (handles 100 concurrent users)
- Storage: 50GB (for code, models, logs)

---

9. CONTINGENCY PLANS

9.1 If VLM Integration Fails

Scenario: LLaVA too slow, GPT-4V too expensive, or accuracy insufficient

Fallback:

- Use rule-based semantic validation only (no VLM)
- Focus Paper 2 on CV + Rules (still novel contribution)
- Title change: "Rule-Based Multimodal Document Fraud Detection"

Impact: Minor (system still functional, research still publishable)

9.2 If User Adoption < 1,000

Scenario: Web app doesn't attract users

Mitigation:

- Targeted marketing (LinkedIn posts, Reddit, ProductHunt)
- University partnerships (offer free to students/faculty)
- Freemium incentives (extra free verifications for referrals)

Fallback:

- Simulate users (evaluate on benchmark dataset instead)
- Focus on technical contribution (user count is bonus, not requirement)

9.3 If Paper Rejection

Scenario: Paper 1 rejected from ICDAR

Response:

- Incorporate reviewer feedback
- Submit to backup venues (ICPR, Pattern Recognition Letters)
- Reframe contribution (focus different aspects)

Timeline: Add 1-2 months buffer for resubmissions

---

10. CONCLUSION

TruthLens addresses a critical $5 trillion global problem through a novel multimodal AI architecture. By combining Computer Vision, Generative AI, and Financial Analysis, the system achieves comprehensive document fraud detection unattainable by single-method approaches.

Feasibility: Demonstrated by 3-day rapid prototype (ELA + Copy-Move + Font Analysis functional).

Innovation: First system integrating all three modalities for document authentication.

Impact: Democratizes fraud detection (accessible to individuals, not just enterprises), with potential to save billions in fraud losses.

Timeline: Aggressive but achievable 12-month schedule, with contingency plans for key risks.

Outcome: By Month 12, we will deliver:

- ✅ Working web application (1,000+ users)
- ✅ Two research publications (tier-1 and tier-2 venues)
- ✅ Complete M.Tech thesis
- ✅ Open-source contribution (code + dataset)
- ✅ Real-world impact (10,000+ documents verified)

This proposal outlines a clear path from concept to deployment, balancing technical rigor with practical utility.

END OF PROBLEM SCOPING & PROJECT PROPOSAL

This combined document fulfills the Problem Scoping (15%) and Project Proposal (30%) components of the M.Tech evaluation rubric, totaling 45% of the overall grade.