2-Week POC Sprint Plan

Project Scope & Deliverables

What We CAN Deliver:

Core Metrics:

- 2-point shots (made/missed)
- 3-point shots (made/missed)
- Assists (passes within 2 seconds of made basket)
- Steals and blocks

Video Processing:

- 15-minute basketball match clips
- Automated event detection and timestamping

What We Will NOT Include:

- Player/team identification (no BLE/tagging system)
- Individual player statistics
- Team-level aggregate scores

Final Deliverables:

1. **Processing Script**: Automated pipeline for any basketball video input
2. **Summary Report**: Structured output with detected events and statistics
3. **Annotated Video**: Original footage with event overlays at specific timestamps

High Level Architecture
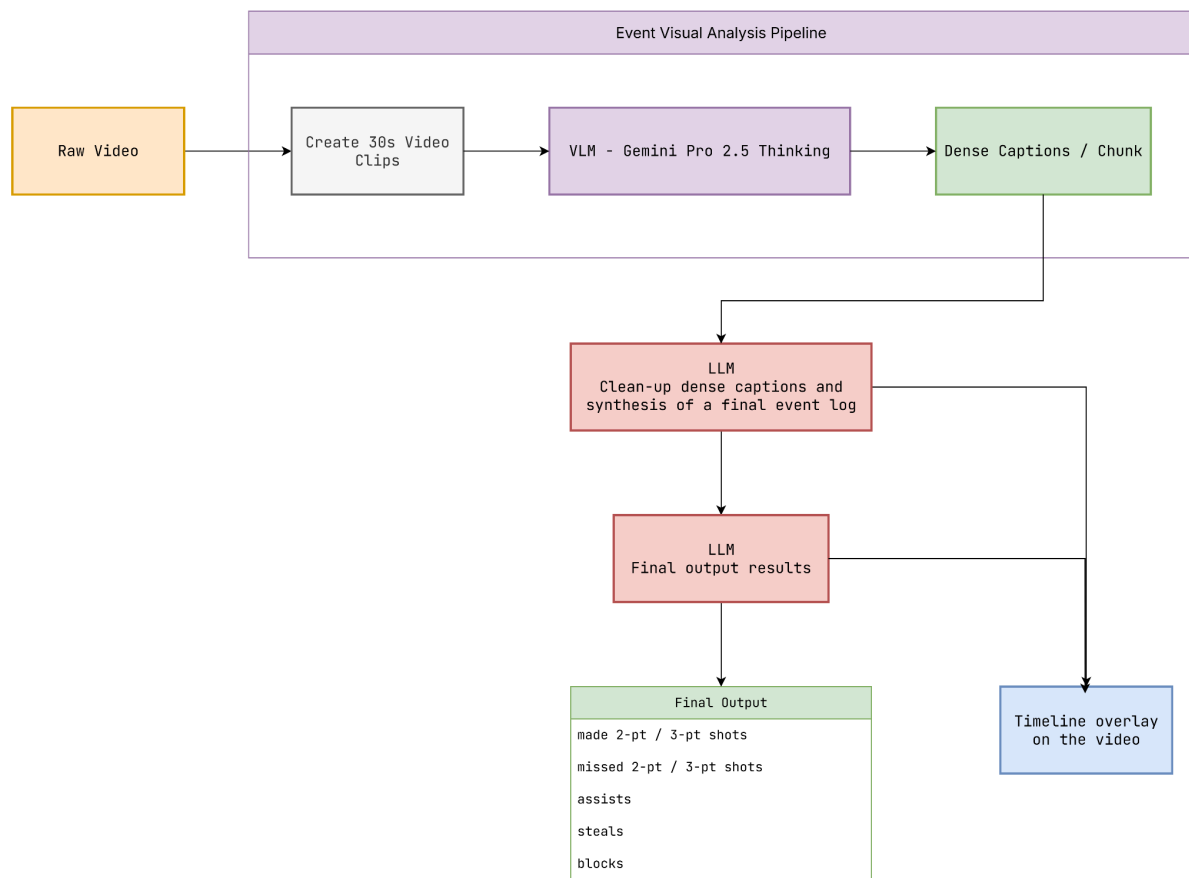
The system processes raw basketball videos through a multi-stage pipeline:

- **Video Segmentation**: Split into 30-second processable chunks
- **VLM Analysis**: Gemini Pro 2.5 Thinking generates dense captions
- **Event Synthesis**: LLM cleans and structures events into final logs
- **Output Generation**: Creates both JSON reports and timeline video overlays

---

Week 1: Single Video Clip Processing Pipeline

Core Objective

Build and validate the fundamental processing pipeline for 30-second video segments, establishing the foundation for dense caption generation and event extraction.

Key Activities

Data Collection & Preparation

- Gather 10-20 standardized basketball videos from YouTube

Dataset Creation

- Split each video into 30-second clips
- Create comprehensive ground truth annotations for evaluation:
    - Manual timestamping of all target events (shots, assists, steals, blocks)
    - Event classification and confidence scoring
    - Quality control through double-annotation process

Dense Captioning Pipeline Development

- Integrate Gemini Pro 2.5 Thinking API for video analysis
- Develop optimized prompts for basketball event detection:

```
None
Analyze this basketball clip for:
- Shot attempts (2pt/3pt, made/missed) with precise timing
- Assists (passes within 2s of made shots)
- Steals and blocks with clear descriptions
Provide exact timestamps and confident classifications.
```

- Implement video preprocessing for optimal VLM input
- Test and refine caption quality on sample clips

Week 1 Deliverables

- Working VLM integration with basketball-optimized prompts
- Ground truth evaluation framework
- Initial accuracy baseline on single-clip processing

Week 2: End-to-End Video Processing Pipeline

Core Objective

Scale the single-clip pipeline to full 15-minute videos, implement event synthesis, create visual overlays, and deliver a complete demo-ready system.
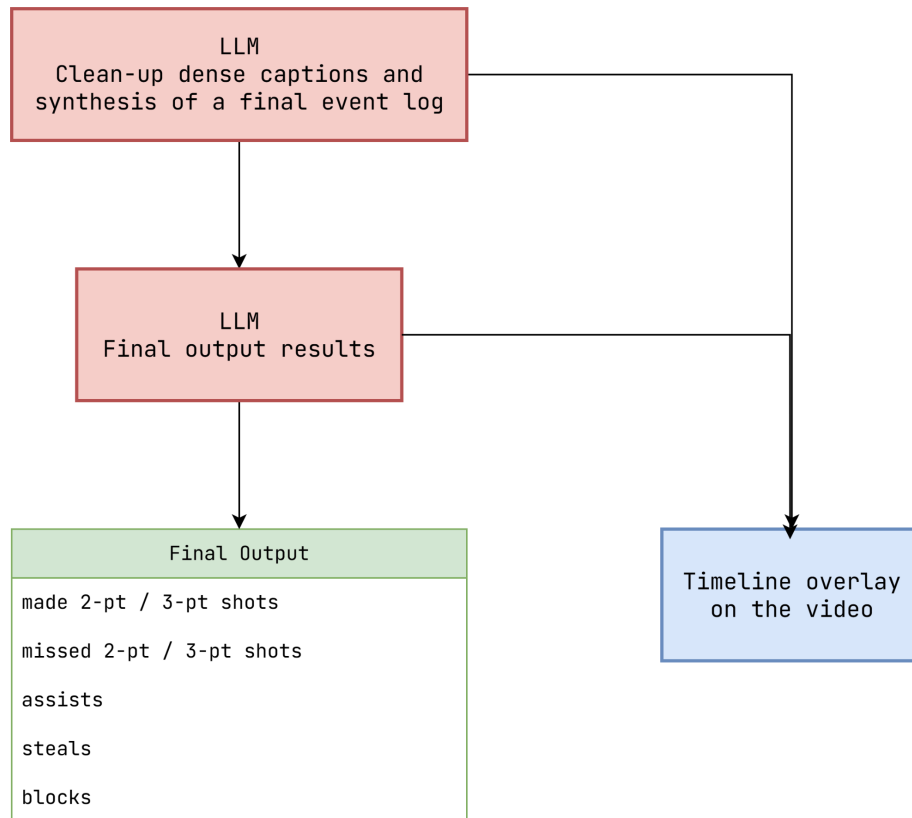
Key Activities

Full Video Processing Pipeline

- Extend 30-second pipeline to handle complete 15-minute videos

- Implement batch processing for multiple video segments
- Handle segment boundaries and event deduplication

Event Synthesis & Structured Output

```
┌─────────────────────────────┐
│           LLM               │
│   Clean-up dense captions   │
│   and synthesis of a final  │──────────┐
│        event log            │          │
└─────────────────────────────┘          │
              │                           │
              ▼                           │
┌─────────────────────────────┐          │
│           LLM               │          │
│    Final output results     │──────────┤
└─────────────────────────────┘          │
              │                           │
              ▼                           ▼
┌─────────────────────────┐    ┌──────────────────┐
│      Final Output       │    │  Timeline overlay│
├─────────────────────────┤    │   on the video   │
│ made 2-pt / 3-pt shots  │    └──────────────────┘
│                         │
│ missed 2-pt / 3-pt shots│
│                         │
│ assists                 │
│                         │
│ steals                  │
│                         │
│ blocks                  │
└─────────────────────────┘
```

- Deploy secondary LLM for caption cleaning and event structuring
- Convert natural language descriptions to standardized JSON format
- Implement temporal reasoning for event sequencing and validation

Video Overlay Generation

- Create timeline overlay system showing detected events
- Implement visual markers for different event types (color-coded)
- Add timestamp annotations and event descriptions

Week 2 Deliverables

- Complete end-to-end pipeline processing 15-minute videos
- JSON output with structured events and summary statistics
- Automated video overlay generation with timeline markers