Paragraph 1: This is a sample PDF document for testing the upload endpoint. The system will extract text from this PDF, chunk it into smaller segments, and create embeddings using sentence transformers. This is the first paragraph with some content that demonstrates text extraction capabilities.

Paragraph 2: Here is the second paragraph with different content. The chunking process will break this text into segments of approximately 800 characters with 100 character overlap between chunks. This allows for better semantic search and retrieval of relevant information from the document.

Paragraph 3: The third paragraph contains more sample text to ensure we have enough content for multiple chunks. Machine learning models like sentence transformers work better with larger amounts of text, so having multiple paragraphs helps test the embedding quality and chunking logic.

Paragraph 4: Finally, this fourth paragraph provides additional content for testing. The PDF processing pipeline includes text extraction using PyMuPDF, text chunking with configurable overlap, and embedding generation using the all-MiniLM-L6-v2 model by default.

Paragraph 5: This is a sample PDF document for testing the upload endpoint. The system will extract text from this PDF, chunk it into smaller segments, and create embeddings using sentence transformers. This is the first paragraph with some content that demonstrates text extraction capabilities.

Paragraph 6: Here is the second paragraph with different content. The chunking process will break this text into segments of approximately 800 characters with 100 character overlap between chunks. This allows for better semantic search and retrieval of relevant information from the document.

Paragraph 7: The third paragraph contains more sample text to ensure we have enough content for multiple chunks. Machine learning models like sentence transformers work better with larger amounts of text, so having multiple paragraphs helps test the embedding quality and chunking logic.

Paragraph 8: Finally, this fourth paragraph provides additional content for testing. The PDF processing pipeline includes text extraction using PyMuPDF, text chunking with configurable overlap, and embedding generation using the all-MiniLM-L6-v2 model by default.