# SCRIPT ANALYSIS

This project introduces a comprehensive tool for analysis and evaluating programming code, focusing on metrics that provide insights into its structure, complexity, and quality.

NAME : ADHIVEL V

CLASS & SECTION : III-B

DEPARTMENT : BCA

ROLL NO. : 60702

REG NO. :2213141033052

# Project Description

The Code Analysis Tool is a web-based application that provides comprehensive static code review, formatting, and analysis. Supports multiple programming languages. It helps developers understand code structure, complexity, and quality metrics through an intuitive interface. . This system serves developers, teams, and organizations by providing automated code formatting, quality assessment.

# Purpose

- Standardize code formatting across teams
- Reduce manual code review time
- Ensure coding standards compliance
- Improve code quality and readability
- Facilitate easier code maintenance

# Proposed System

## 1 Code Input Methods

- Direct code pasting
- File upload
- Drag and drop support

## 2 Analysis Capabilities

- Syntax validation
- Code metrics calculation
- Complexity analysis
- Structure identification
- Pattern recognition

## 3 Result Presentation

- Visual metrics display
- Detailed breakdowns
- Interactive components
- Exportable reports

# System Requirements

### 1. Hardware Requirements

**Server:**

- [ ] Processor: Multi-core CPU (2+ cores)
- [ ] RAM: 4GB minimum
- [ ] Storage: 20GB minimum
- [ ] Network: High-speed internet connection

**Client:**

- [ ] Modern web browser
- [ ] 2GB RAM minimum
- [ ] Stable internet connection

### 1. Software Requirements

**Server-side:**

- [ ] Python 3.8+
- [ ] Flask Framework
- [ ] Required Python packages:
- [ ] Flask

**Client-side:**

- [ ] Modern web browser (Chrome, Firefox, Safari, Edge)
- [ ] JavaScript enabled
- [ ] Cookie support
- [ ] Local storage access

# System Specification

## 1. Hardware Specifications

### 1.1 Server Requirements

- Processor: Multi-core CPU (2+ cores)
- RAM: 4GB minimum
- Storage: 256GB SSD or HDD
- Network: 1Gbps Ethernet

### 1.2 Client Requirements

- Modern web browser (Chrome, Firefox, Safari, Edge)
- Minimum 4GB RAM
- Stable internet connection (5Mbps+)
- Display resolution: 1280x720 minimum

## 2. Software Specifications

### 2.1 Development Stack

Frontend:

- HTML5/CSS3
- JavaScript (ES6+)
- Monaco Editor

Backend:

- Python 3.8+
- Flask 2.0+
- Redis

Development Tools:

- Git
- VS Code/PyCharm

## 3.Performance Specifications

### 3.1 Response Times

- File Upload: < 5 seconds
- Code Analysis: < 3 seconds
- Code Formatting: < 2 seconds

### 3.2 System Capacity

- Concurrent Users: 1000+
- File Size Limit: 10MB
- Request Rate: 100/minute per user
- Total Storage: 500GB expandable

# Key Features: Output

## 1. Code Input Methods

File Upload System

- Support for multiple file formats
- Batch processing capability

Direct Code Paste

- Real-time syntax highlighting
- Language auto-detection

## 2. Analysis Features

Style Checking

- Language-specific style guides
- Custom rule configuration
- Team standard enforcement

## 3. Formatting Capabilities

Code Structure

- Line length management
- Bracket placement
- Comment alignment

# Target Users: Empowering Developers

1

## Individual Developers

Students, freelancers, and open-source contributors benefit from improved code quality and efficiency.

2

## Development Teams

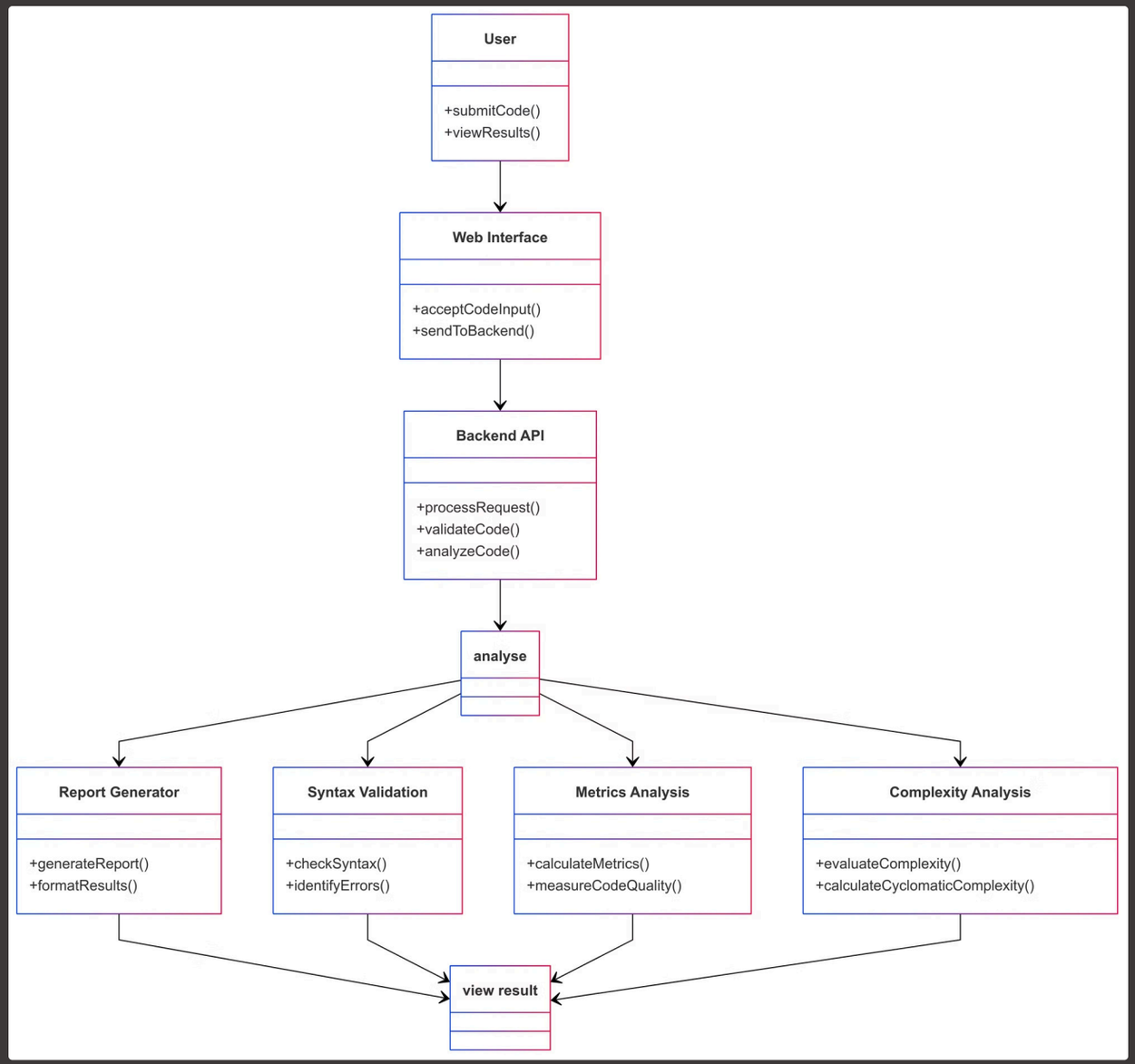
Software engineering teams, code reviewers, and project managers streamline workflows and enhance collaboration.

3

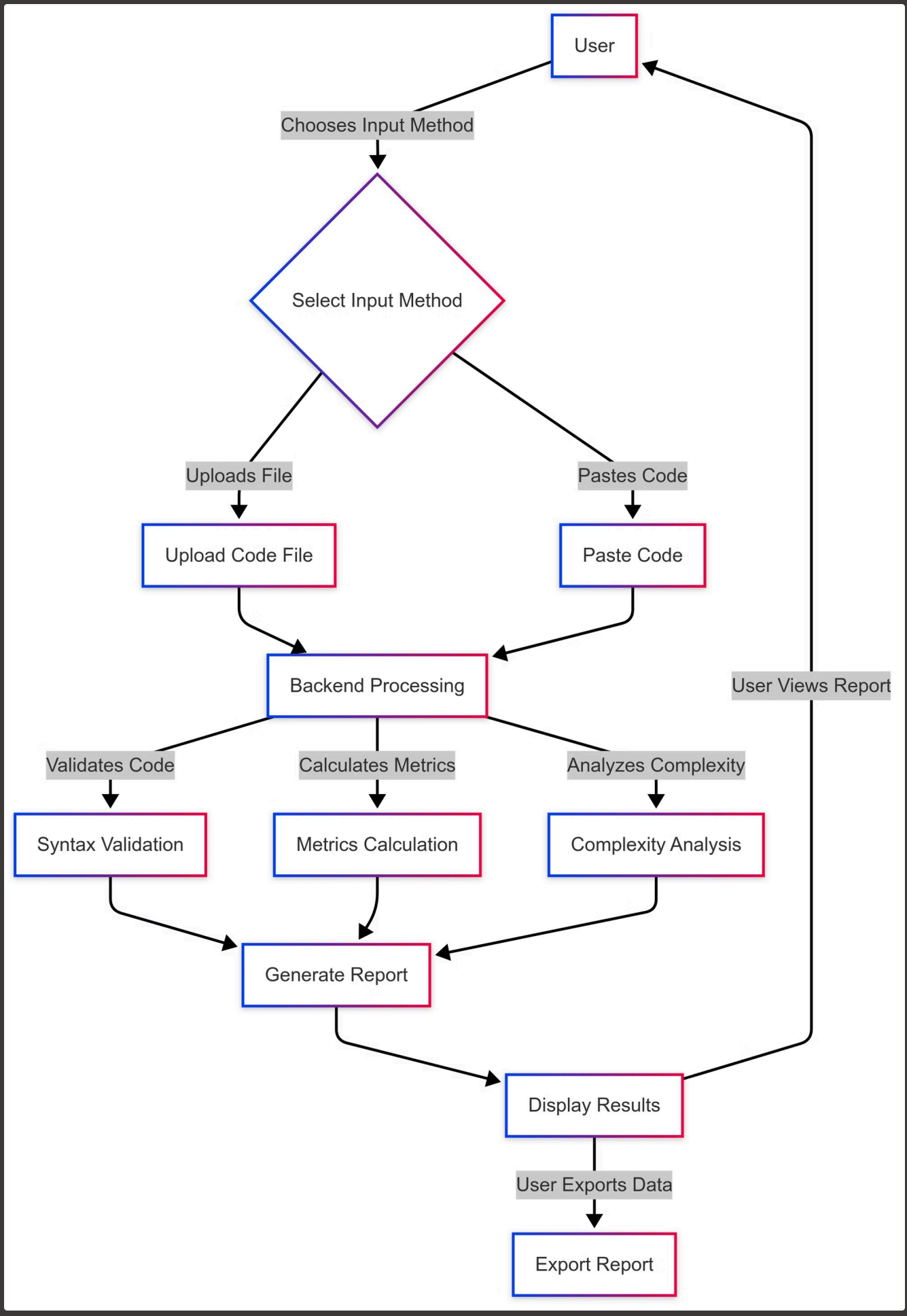
## Educational Institutions

Programming instructors, students, and teaching assistants gain valuable insights and tools for learning and teaching.

# System Design Diagram



# Data Flow Diagram (DFD)



# Business Value: Efficiency & Quality

1

## Efficiency Improvements

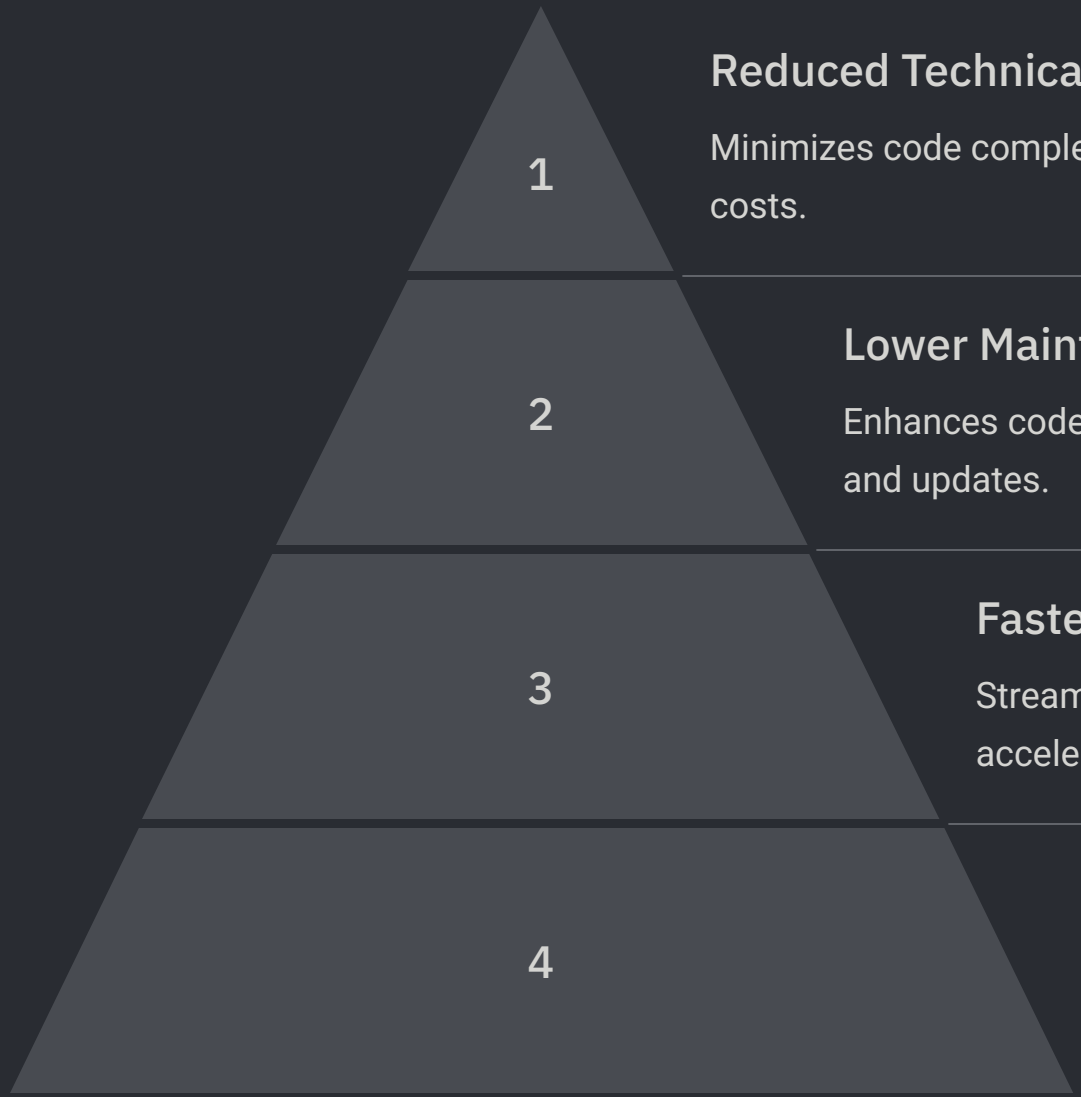
Reduces code review time by 50%, accelerates onboarding by 40%, and minimizes formatting issues by 30%.

2

## Quality Assurance

Ensures consistent code style, reduces bug introduction, improves code maintainability, and promotes better documentation standards.

# Business Value: Cost & Growth




### Reduced Technical Debt

Minimizes code complexity and improves maintainability, reducing future development costs.

### Lower Maintenance Costs

Enhances code quality and consistency, reducing the need for frequent fixes and updates.

### Faster Development Cycles

Streamlines workflows, reduces errors, and improves collaboration, accelerating development timelines.

### Improved Resource Allocation

Optimizes development processes, freeing up resources for innovation and strategic initiatives.