Operating System

USER(developer)

System Administrator

UML USE CASE\_version2

Provide Source Code

Run Compilation

View Compilation

Output

Debug Code

Execute Program

Optimize code

Configure Compiler

Settings

Manage User Access

Update Compiler

Allocate Memory for Execution

Provide System Resources

Handle Input/Output

Operations

ERROR

NO ERROR

USE CASE DIAGRAM

User

Provide Source

Code

Yes

No

Compilation

Errors?

Debug Code

Execute Program

Allocation of

Memory

Provide

Resources

Handle Input

and Output

Optimize Code

Final Output of

Code

Lexical Analysis

Syntax Analysis

Semantic

Analysis

Intermediate

Code Generation

Code

Optimization

Code Generation

Run Compilation

ACTIVITY DIAGRAM

***Lexer***

-

code: string

tokenize(code: String): List<Token

+

>

**AST**

+

root: Node

**SemanticAnalyzer**

analyze(ast: AST): void

+

**Parser**

>

tokens: List<Token

-

parse(tokens: List<Token>): AST

+

1

1

**SymbolTable**

symbols: Map<String, Symbol

+

>

+ lookup(name: String): Symbol

+ insert(name: String, symbol: Symbol): void

**Optimizer**

+

IR: data structure

optimize(ir: IR): IR

+

**IntermediateCodeGenerator**

+ generate(ast: AST): IR

**CodeGenerator**

+

generate(ir: IR): MachineCode

**ErrorHandler**

>

error\_messages: List<string

+

logError(message: String): void

+

1

1

1

1

1

1

1

1

1

1

1

1

1

1..\*

1..\*

1

CLASS DIAGRAM

SEQUENCE DIAGRAM

