**1. Main Classes**

* **Main.java**: Entry point of the application
  + Sets up the main JFrame window
  + Initializes the MainUI component
* **MainUI.java**: Primary user interface
  + Defines the entire user interface layout
  + Handles user interactions and button events
  + Communicates with ControlUI to execute operations

**2. Control and Processing**

* **ControlUI.java**: Backend controller
  + Manages communication between UI and CPU
  + Handles register operations
  + Implements button click events
  + Manages program loading and execution
  + Key methods:
    - loadGPRx(): Load values into general purpose registers
    - whenIPLClick(): Initial Program Load functionality
    - whenRunClicked(): Execute entire program
    - whenStepClicked(): Single instruction execution
    - whenHaltClicked(): Program halt handling
* **CPU.java**: Core processing unit
  + Implements instruction execution logic
  + Manages register operations
  + Handles memory access
  + Key features:
    - Instruction decoding and execution
    - Memory address calculation
    - Register value management
    - Error handling and fault detection

**3. Data Management**

* **CPUExecute.java**: Instruction execution package
  + Stores instruction execution details
  + Fields:
    - effectiveAddress: Calculated memory address
    - I: Indirect addressing flag
    - R: Register identifier
    - IX: Index register identifier
    - address: Base address
    - instructionString: Current instruction type
* **DataType.java**: Register definitions
  + Defines all register types and sizes
  + Includes:
    - General Purpose Registers (GPR0-GPR3): 16 bits
    - Index Registers (IXR1-IXR3): 16 bits
    - Special Registers (MAR, MBR, PC, etc.)
    - Control Registers (CC, MFR, HLT)

**4. Memory Management**

* **Memory.java**: Memory simulation
  + Implements main memory functionality
  + Manages memory read/write operations
  + Handles memory size constraints
  + Provides memory reset capabilities
* **Register.java**: Register implementation
  + Implements individual register functionality
  + Manages register value storage and retrieval
  + Handles register size constraints