CASE – Language Manual

**TwilightRaven Entertainment**

Version 1.0a

Ted Gress

(Highlighted Yellow is Lua Functionality)

**Introduction**

* Code Integration
* String as the almighty base class
* Redundancy
* The Use of an Entity Pool

**Language Synopsis**

**Basic** **Concepts**

* Values and Types
  + Primitives
  + Numbers are Objects
  + Constants
  + Custom Primitives
  + Arrays
  + Tuples
  + Sets
  + Types
  + Type Templates
  + String Operations and Strings as Root
* Classes
  + Access Modifiers
  + Inheritance
* Environments and the Global Environment
* Error Handling
* Metables and Metamethods
* Garbage Collection
  + Garbage Collection MetaMethods
  + Weak Tables
* Coroutines
* Event Handlng

**The** **Language**

* Miscellanenous Keyword
* Annotations
* Comments
* Refactoring
* Module Connections
* Lexical Conventions (
* Variables
* Statements
  + Blocks
  + Chunks
  + Assignment
  + Control Structures
    - Generators
    - While Statements
    - Until Statements
    - Do While Statements
  + For Statement
  + Function Calls as Statements
  + Local Declarations
  + Pointers

Expressions

* Arithmetic Operators
* Bitwise Operators
* Coercions and Conversions
* Relational Operators
* Logical Operators
* Regular Expressions
* Pipe Operator
* Client/Server Operator
* Exotic Operations
* Concatenation
* The Length Operator
* Precedence
* Table Constructors
* Cons/Pairs
* Futures and Promises
* Function Calls
* Function Definitions
  + - Functions are Objects
    - Integrated Functions
    - Parameters
    - Reactions
    - Recursion
* Equalities

Visbility Rules

**Design Paradigms**

* Model View Controller
* Integrate Design Patterns
* Miscellaneous Designs
* Anti-patterns
* Positive Design Ideas
* Dependency Injection
* Diagrams
* CASE BASED: Architectural Standard API
* Traffic Framework
* Auto-Code
* Language Plugins

**Low Level Interface**

* Byte Code
* System Calls
* Interrupts

**Integrated Database Implementation**

**Web Integration**

**Case Application Standard Library**

* Valid and Acceptable Indices
* C Closures
* Registry
* Error Handling in C
* Handing Yields in C
* Functions and Types
* The Debug Interface

The Auxilary Library

* Functions and Types
* Standard Collections Library

**Standard Libraries**

* Basic Functions
* Coroutine Manipulation
* Modules
* String Manipulation
  + Patterns
  + Format Strings for Pack and Unpack

UTF-8 Support

Table Manipulation

Mathematical Functions

Input and Output Facilities

The Debug Library

**Compiler Features**

* Auto-Doc
* CASE IDE Fix-It

**Integrated Into Language**

* HEAL
* PolyGlot
* Automatic Scalability
* Lightweight Antivirus