**Pivi: An eclipse Pug-In for Visual Parallel Programming**

**Sponsor: Javier Gonzalez**

Team Members

Panchavati Ganesh, Sri Kiran

Kara, Pujitha

Kogaleru, Nithya

Peddabuttaiahgari ,Giridhar

Shendware, Snehal

**Summary**

The Pivi aims at providing an easier and interactive way to create a visual model of a problem. It is primarily intended for use by students. By theoretically studying the concepts, students may find it difficult to understand them and to implement them in a program. Another major issue is that a lot of time and effort is needed to program a problem using legacy textural programming languages, especially for students who are in a learning phase. Concepts like parallel programming, concurrent programming and multicore programming are complex and require deep understanding to implement them. This eclipse plug in Pivi addresses to solve all these problems.

Solving the above-mentioned problems is important as it is easy to visualize a solution to a problem. For beginners, it is simple and facilitates to create a visual model of a problem. Support for visual programming helps to reduce the time that students need to invest in programming a solution to a problem using textural programming languages and facilitates for a rapid development. Since the tool validates the linking of icons that represent programmable constructs and generates the code automatically, the chances of syntax errors in the code is highly reduced. Concepts like parallel programming can solve many complex problems. By understanding such concepts deeply, students can increase their problem-solving skills.

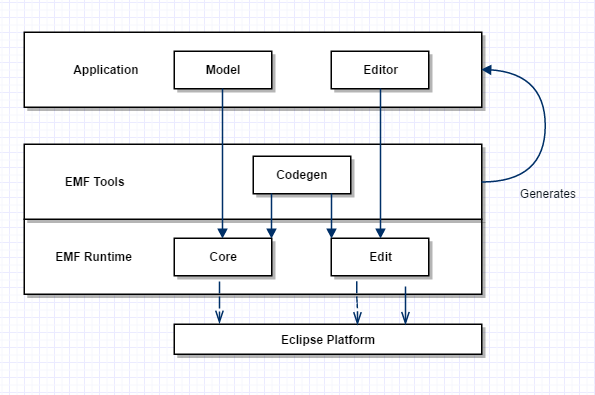
This Eclipse Plug-In is focused on facilitating teaching/learning techniques and implications of parallel, concurrent and multicore programming. The developed software is an Eclipse plugin that allows users to visually program in Java. The plugin presents a palette of icons that can be used to create a program visually. Each icon in the palette represents different constructs of a programming language. The plugin also presents a canvas on which user can place the necessary icons from the palette to create a program. The plugin should allow users to connect the icons present on the canvas by validating the connections. The user will be able to modify the parameters provided for each icon. Concurrent programming features are provided by the plugin. This includes creation of threads, execution of instructions in multiple threads, access to the memory among threads. The plugin also validates if the graph of icons and connections created on the canvas is a valid program. For a valid program created visually on graph, the plugin generates the code in Java programming language which can be understood by the user in an effective way.

**Requirements and Schedule**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Functional Requirements** | **Tasks** | **Due Date** | **Status** |
| 1 | User should be able to access Pivi Plugin in Eclipse Environment. | * Select Pivi plugin perspective option in Eclipse Toolbar * Drag / Drop option of UI Elements from Pivi palette * Close the Pivi GUI on close option | 02/01/2017 | Completed |
| 2 | User should be able to program using visual elements in Pivi Perspective. | * Create Pivi palette with UI programmable nodes and connections. * Provide different logical elements for programming like (data structure, conditional statements, loop etc.) * Provide code insertion option with respective logical elements * Validate UI element nodes and connections with respective to Model and generated code | 02/22/2017 | In progress |
| 3 | User should be able view/edit generated code after visual programming. | * Provide “Generate” option for language specific code generation * Generate code in a format understandable to user from visual UI elements and underlying Model * Provide facility to edit generated code and resubmit | 03/08/2017 | In progress |
| 4. | User should be able to run part of code asynchronously using parallel elements. | * Tag out UI elements executing in parallel separately * Work on the code for parallel execution. * Generate code for asynchronous visual elements * Provide results of parallel execution and validate results | 03/29/2017 | new |
| 5 | User should be provided with guidelines of using Pivi palette elements. | * Write documentation for UI elements in detail * Embed into Eclipse wiki or Help Page accessible to User | 04/19/2017 | New |

**Software Design**

**1. Component Diagram:**



**2. Class Diagram:**

****