PHP CLASS MANAGER

Оглавление

[Цель работы 3](#_Toc478152673)

[Обзор предметной области 4](#_Toc478152674)

[Описание применимости 7](#_Toc478152675)

[описание результатов проекта 8](#_Toc478152676)

[ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА 9](#_Toc478152677)

# Цель работы

Создать интегрированную среду разработки ООП систем путём создания их с помощью графического интерфейса с возможностью генерации исходного кода ООП системы.

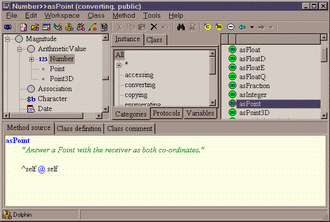
Основные возможности, которая должна предоставлять среда:

1. Создание классов
2. Изменение классов
   1. Добавление методов, свойств, констант
   2. Изменение методов, свойств, констант
   3. Удаление методов, свойств, констант
3. Создание образа из классов
   1. Добавление классов
   2. Удаление классов
4. Генерация кода из собранного образа
5. Экспорт/импорт образа

# Обзор предметной области

Рассматриваемая мною предметная область относиться к разработке веб-приложений и сайтов с использованием языка PHP.

**Основная идея проекта** – это создать интегрированную среду разработки по аналогии среды Smalltalk, для создания, изменения и проектирования ООП системы будущего веб-приложения.



Проект не несёт цели создания полного аналога/аналогов для Web`а, а берёт идею, в соответствии с которой манипулирование с классами ведётся с помощью графического интерфейса.

Аналогами являются:

* Dolphin Smalltalk
* VisualWorks
* Visual Smalltalk
* И другие реализации Smalltalk

Язык Smalltalk и его интерактивная среда разработки неразрывно связанны. Сам язык используется для решения следующих задач:

|  |  |
| --- | --- |
| * Автоматизация, робототехника * Диспетчеризация, планирование * Коммуникация, связь * Медицина | * Обработка коммерческой информации * Системы управления * Тренажёры, моделирования * Обучение программированию |

Сам подход к написанию программ в языке Smalltalk кардинально отличается от обычного подхода.

В таких языках как C++, Java, C#, Python, Ruby и т.п. мы разрабатываем программу путём композиции библиотек/модулей/пакетов в связке с особенностями парадигмы и самого языка, а в конечном итоге мы получаем программу, которая решает поставленную задачу. Так же важно отметить, что любой язык, перечисленный выше не привязан к конкретной IDE или какой-либо графической среде.

В случае с языком Smalltalk, который неразрывно связан с графической средой, разработка идёт путём декомпозиции, то есть мы из "коробки" имеем библиотеки/модули/пакеты и затем в процессе разработки мы убираем лишнее. Так же мы занимаемся реализацией своих решений и создания своих классов, объекто и т.д.

Как правило такая среда разработки предоставляет разработчику следующие инструменты:

* **Workspace** – рабочее пространство, в котором можно писать исходный код и выполнять его
* **Inspector** – инструмент для просмотра содержимого инспектируемого объекта
* **Class** **Browser** – основной инструмент, в котором происходит создание/удаление и изменения классов
* **Debugger** – отладчик

Помимо этого, вся система является единым целым и её можно изменять "на лету". Так же, среда разработки Smalltalk является самой продуктивной с точки зрения программирования.

В качестве основного языка для реализации был выбран PHP, так как он имеет общие черты с языком Smalltalk:

* Интерпретируемость
* Динамическая типизация
* Объекто-ориентированная парадигма

Так же есть и недостатки:

* Язык не является чисто объекто-ориентированным, что не даст возможности полной реализации аналога
* Так же язык не предназначен для реализации огромных комплексных систем и приложений
* Не имеет виртуальной машины

# Описание применимости

# описание результатов проекта

# ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА