



## ПЛЮСЫ И МИНУСЫ ИСПОЛЬЗОВАНИЯ ФРЕЙМВОРКА

### Чарыева Дуня Жанбердиевна

Преподаватель Международного университета нефти и газа имени Ягшыгелди Какаева, г. Ашхабад Туркменистан

### Атаева Гульджерен Бяшимдурдыевна

Студент Международного университета нефти и газа имени Ягшыгелди Какаева, г. Ашхабад Туркменистан

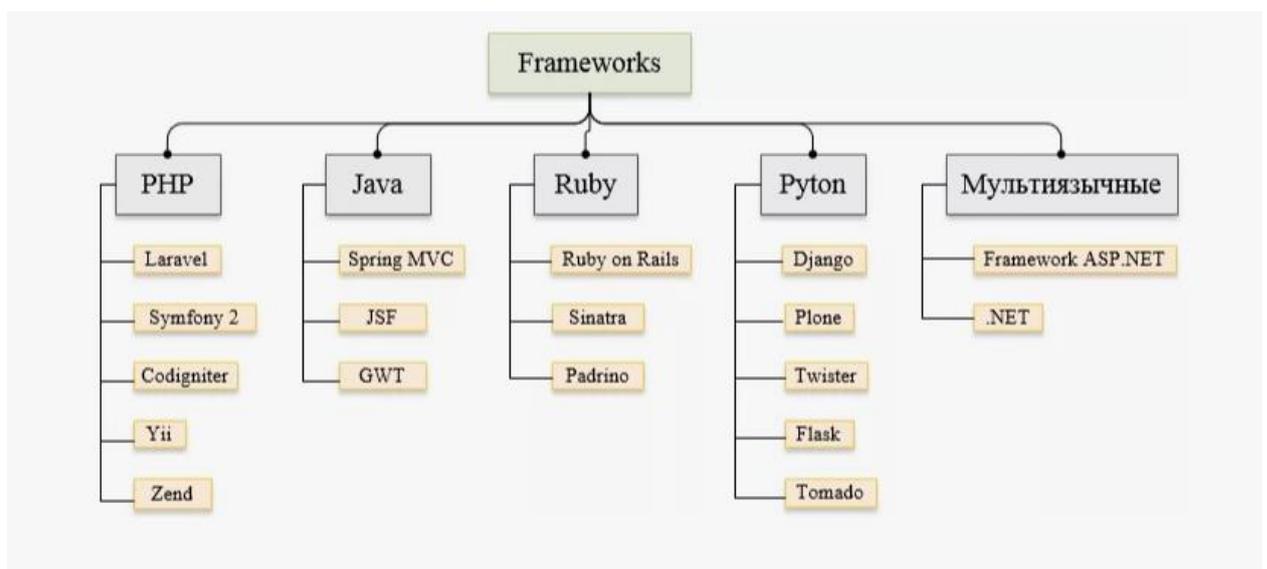
### Гарагулова Махым Чарыгелдиевна

Студент Международного университета нефти и газа имени Ягшыгелди Какаева, г. Ашхабад Туркменистан

**Аннотация.** В данной статье рассмотрена тема использования фреймворка, ключевые аспекты, а также выявлены потенциальные преимущества и сложности в его использовании.

**Ключевые слова:** фреймворк, программное обеспечение, гибкость, приложения, информационная система, ИС.

В мире быстро развивающихся технологий и постоянных изменений в сфере разработки программного обеспечения, фреймворки становятся неперемным инструментом для обеспечения эффективности и структурированности в различных проектах. Фреймворк, будучи надежным каркасом, обеспечивает не только ускоренное создание приложений, но и повышает уровень поддержки, обеспечивая надежность и устойчивость в долгосрочной перспективе.



От повышения производительности до обеспечения стандартов безопасности, фреймворки становятся ключевым элементом, от которого зависит успех проектов и удовлетворение пользователей. Фреймворки становятся неотъемлемым звеном в цепи успешной разработки.

Фреймворк – это структура, представляющая собой набор готовых компонентов, библиотек, абстракций и правил разработки, предназначенных для облегчения процесса создания программного обеспечения. В своей сущности, фреймворк предоставляет базовый каркас, внутри которого разработчики могут строить приложения, следуя стандартизированным принципам.

### **Цели внедрения фреймворков**

- ускорение разработки: фреймворки предоставляют готовые модули и инструменты, что ускоряет процесс создания приложений.
- стандартизация: они устанавливают стандарты разработки, обеспечивая единообразие в коде и архитектуре приложений.
- безопасность: фреймворки предоставляют встроенные механизмы безопасности, снижая риски связанные с уязвимостями.
- облегчение обслуживания: стандартизированный код упрощает процессы поддержки, обновлений и масштабирования приложений.

### **Кто использует фреймворки и для каких целей**

- разработчики программного обеспечения: фреймворки широко используются разработчиками для создания веб-приложений, мобильных приложений и других программных продуктов.
- компании и проекты различного масштаба: от стартапов до крупных корпораций, фреймворки применяются в проектах различных масштабов и сложности.
- цели варьируются от проекта к проекту: фреймворки используются для разработки разнообразных приложений, включая веб-сайты, электронную коммерцию, системы управления контентом, мобильные приложения и многое другое.

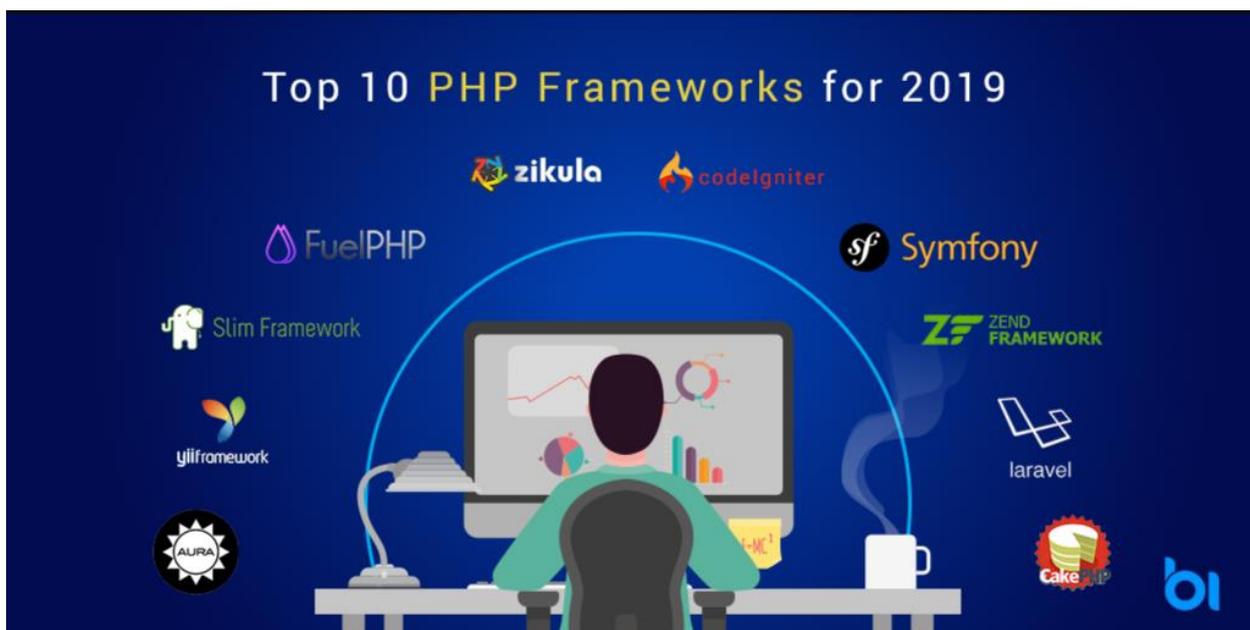
### **Виды фреймворков и как выбрать подходящий**

Существует множество фреймворков, а каждый из них ориентирован на определенный тип приложений и задач. Например, веб-фреймворки вроде Ruby on Rails. Ruby on Rails – фреймворк, написанный на языке программирования Ruby, реализует архитектурный шаблон Model-View-Controller для веб-приложений, а также обеспечивает их интеграцию с веб-сервером и сервером баз данных. Является открытым программным обеспечением и распространяется под лицензией MIT.

React Native – это кроссплатформенный фреймворк с открытым исходным кодом для разработки нативных мобильных и настольных приложений на JavaScript и TypeScript, созданный Facebook, Inc. (По требованию Роскомнадзора информируем, что иностранное лицо, владеющее информационными ресурсами Google является нарушителем законодательства Российской Федерации – прим. ред.)

Фреймворки для научных вычислений: PyTorch – фреймворк машинного обучения для языка Python с открытым исходным кодом, созданный на базе Torch. Используется для решения различных задач: компьютерное зрение, обработка естественного языка. Разрабатывается преимущественно группой искусственного интеллекта

Инструменты для создания, например – Django REST framework (Python) — мощный набор инструментов для создания веб-сервисов и API на основе фреймворка Django. Он является одним из наиболее широко используемых для создания RESTful API в экосистеме Django.



Выбор подходящего фреймворка — важное решение, влияющее на успешность вашего проекта. Пригласим вас на погружение в мир фреймворков, где каждый из них несет свой уникальный вклад в обеспечение эффективности и качества разработки.

На российском рынке существует платформа 1С. Она тоже является фреймворком.

### **Плюсы использования фреймворков**

**Ускоренная разработка:** готовые компоненты и абстракции фреймворка позволяют ускорить процесс разработки, так как разработчику необходимо создавать только специфичные для проекта части. Фреймворки предоставляют готовые компоненты и библиотеки, что позволяет разработчикам избежать написания стандартного кода "с нуля" и сосредоточиться на уникальных аспектах приложения. Возможность использовать готовые шаблоны и решения ускоряет процесс проектирования и реализации функциональности.

**Стандартизация:** фреймворк предоставляет структуру и стандарты, что способствует более согласованному и структурированному коду, обеспечивает единообразие. Стандартизированный код упрощает сопровождение приложения и позволяет разработчикам быстрее ориентироваться в кодовой базе.

**Более высокий уровень безопасности:** многие фреймворки включают стандарты безопасности, что снижает риск уязвимостей и улучшает безопасность приложения. Фреймворки предоставляют средства для эффективной обработки ошибок, что повышает стабильность и надежность приложения.

**Масштабируемость:** фреймворки обеспечивают масштабируемость, что позволяет приложениям эффективно расти с увеличением объема данных или пользовательской базы.

**Оптимизированные Алгоритмы:** многие фреймворки предоставляют оптимизированные алгоритмы работы с данными, что улучшает производительность.

**Обмен опытом:** Большое сообщество вокруг фреймворка способствует обмену опытом и решению проблем.

Регулярные Обновления: активные фреймворки регулярно обновляются, предоставляя новые функции и исправляя обнаруженные уязвимости.



Многоплатформенность: некоторые фреймворки позволяют создавать приложения, совместимые с разными платформами (веб, мобильные устройства и др.).

Поддержка различных языков: некоторые фреймворки поддерживают несколько языков программирования, что дает командам больше свободы в выборе технологии.

Архитектурные паттерны: фреймворки обычно поощряют применение архитектурных паттернов, таких как MVC (Model-View-Controller), что способствует лучшему разделению обязанностей в приложении.

Развитие профессиональных навыков: использование фреймворка обучает разработчиков навыкам и подходам, востребованным на рынке труда.

Использование фреймворков оберегает разработчиков от многих рутинных и повторяющихся задач, предоставляя им возможность сосредоточиться на более творческих и высокоуровневых аспектах создания программного обеспечения. От ускорения разработки до повышения безопасности и удобства сопровождения, фреймворки становятся неотъемлемым инструментом в руках современных разработчиков.

Минусы использования фреймворков

Избыточность функционала: некоторые фреймворки содержат избыточные возможности, которые могут привести к ненужному перегрузению проекта.

Ограничения свободы действий: использование фреймворка может ограничить свободу выбора инструментов и подходов к решению задач.

Обучение и вхождение в проект: новым разработчикам может потребоваться время для освоения специфики фреймворка, что может замедлить начальные этапы работы.

Производительность: некоторые фреймворки могут иметь накладные расходы, влияющие на производительность приложения.

Затраты на обучение и переобучение: переход на новый фреймворк может потребовать времени и усилий для обучения команды, особенно если она не знакома с данной технологией.

Несмотря на минусы, правильный выбор фреймворка может существенно облегчить процесс разработки и повысить качество конечного продукта. Разработчики должны внимательно взвешивать плюсы и минусы, учитывая требования проекта и свои предпочтения.