**2. # ADR - Выбор языка программирования для серверной части**

Участники: Геворгян Арсен, Афанасьев Никита, Сахабутдинов Тимур

Дата: 25.09.24

Статус: принят

**## Контекст**

Для разработки приложения мы выбрали серверный подход, и встал вопрос о выборе языка программирования для написания серверной части.

**## Рассматриваемые варианты**

**Java:**

* **Плюсы**: Высокая производительность, хорошая масштабируемость, огромная экосистема, подходит для микросервисов.
* **Минусы**: Более сложный и громоздкий синтаксис, требует больше ресурсов.

**Kotlin:**

* **Плюсы**: Совместим с Java, удобный синтаксис, безопасность типов, поддержка корутин для многопоточности.
* **Минусы**: Меньше зрелых серверных фреймворков и примеров по сравнению с Java.

**Python:**

* **Плюсы**: Простой и быстрый в разработке, богатая экосистема, асинхронная обработка.
* **Минусы**: Низкая производительность, ограничения многопоточности, менее масштабируем.

**## Решение**

Мы выбрали Kotlin.

**## Обоснование**

Обратная совместимость с Java и библиотеками языка; поддержка от Google, поскольку Kotlin – основной язык в Android Studio; регулярные обновления языка и дополнения его экосистемы; более компактный синтаксис в сравнении с Java.

**## Последствия**

Плюсы:

* Совместим с Java, доступ к богатой экосистеме.
* Удобный, лаконичный синтаксис — быстрее писать код.
* Безопасность типов и меньше ошибок с null.

Минусы:

* Меньше зрелых серверных фреймворков.
* Меньше примеров и документации по сравнению с Java.

**## Риски**

Kotlin имеет меньше проверенных временем серверных фреймворков по сравнению с Java, что может ограничить выбор технологий.

Меры:

* Использовать проверенные фреймворки, такие, как **Ktor** или **Spring Boot** (совместим с Kotlin).
* Интегрировать с существующими Java-фреймворками.

Kotlin имеет меньше ресурсов, примеров и сторонних библиотек для решения серверных задач.

Меры:

* Проводить дополнительное исследование и выбирать зрелые решения из экосистемы Java.
* Активно использовать Kotlin-сообщество, форумы, и участвовать в open-source проектах.

Не все Java-библиотеки могут идеально работать с Kotlin из-за различий в языках.

Меры:

* Тщательно тестировать библиотеки перед использованием.
* Предпочитать Kotlin-специфические библиотеки или активно использовать адаптеры и обёртки.

Kotlin относительно новый язык, и на рынке может быть меньше опытных разработчиков.

Меры:

* Обучение существующих разработчиков Kotlin

**## Затронутые области**

Серверная часть системы, базы данных

История

25.09.24: Первоначальное обсуждение вариантов ЯП для серверной части.