АО «АЛМАТИНСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»
https://atu.edu.kz/

Факультет «Информационные технологии»
Кафедра «Компьютерная инженерия»

Название курса: «Инструментальные средства разработки программ»



Зиятбекова Гулзат Зиятбеккызы – PhD, ассоциированный профессор
g.ziyatbekova@atu.edu.kz

СИСТЕМАТИЗАЦИЯ · СРЕДА

Систематизация программ начинается с понятной среды

Как описывать инструменты, зависимости и установку так, чтобы результат можно было повторить.

Почему список установленных программ быстро перестаёт помогать

Список отвечает на вопрос «что стоит», но не объясняет, как это работает вместе.

Одинаковые названия

Скрывают разные версии, зависимости и настройки.

Память вместо документации

Восстановление среды превращается в поиск по памяти.

Установка без повтора

Цель систематизации — сделать установку и сопровождение воспроизводимыми.

Сначала описываем среду, затем выбираем инструменты

Инструмент имеет смысл только внутри условий, которые он должен обслуживать.

ОС и архитектура

На чём запускается система.

Языки и зависимости

Языки, рантаймы, библиотеки и менеджеры пакетов.

Рабочие инструменты

Редактор, терминал, система контроля версий и переменные окружения.

Ограничения

Права доступа, сеть, лицензии и требования проекта.

Единый реестр программ превращает набор установок в систему

ПОЛЕ	ЧТО ФИКСИРУЕМ	ЗАЧЕМ	СТАТУС
Программа	Название и назначение	Понимать роль инструмента	✓
Совместимость	Версия или диапазон	Не получить скрытый конфликт	✓
Источник	Установка и обновление	Повторить действие безопасно	✓
Ответственность	Владелец, критичность, проекты	Знать, что поддерживать первым	✓



Инструменты установки делятся по уровню задачи

Чем шире граница среды, тем выше уровень инструмента.

ОС

Системные менеджеры пакетов дают базовые компоненты операционной системы.

Экосистема

Менеджеры языковых пакетов отвечают за библиотеки конкретного языка.

Проект

Менеджеры окружений изолируют версии и зависимости.

Вся среда

Контейнеры и образы переносят систему целиком.



Менеджер пакетов отвечает за повторяемую установку зависимостей

Команда описывает желаемое состояние, а не вспоминает последовательность действий.

01

Описать

Зависимости в декларативном файле.

02

Зафиксировать

Версии и источник каждого пакета.

03

Установить

Одной командой или сценарием.

04

Проверить

Автоматической проверкой результата.

Изоляция среды защищает проекты от конфликтов

Проект отдельно

Отдельное окружение на проект или сервис.

Версия явно

Язык и инструменты не выбираются случайно.

Настройки локально

Глобальные изменения не прячутся в системе.

Сброс без страха

Среду можно быстро удалить и пересоздать.



Документ установки должен отвечать на пять вопросов

Если читатель не может проверить себя по инструкции, инструкция ещё не закончена.

01 Что устанавливаем и для чего?

02 На какой системе и с какими ограничениями?

03 Какие шаги обязательны, а какие опциональны?

04 Как понять, что установка прошла успешно?

05 Что делать при типовой ошибке?

Хорошая документация разделяет путь новичка и детали системы

Быстрый старт

Минимальный маршрут до рабочего результата.

Полная инструкция

Версии, команды, конфигурация и проверки.

Справочник

Переменные, каталоги и политики обновлений.

Разбор неисправностей

Симптомы, причины и решения.



Автоматизация убирает ручные действия из повторяющегося процесса

01

Подготовить

Скрипт установки подготавливает базовую среду.

02

Описать

Файл зависимостей описывает состав проекта.

03

Проверить

Проверка окружения ловит несовместимые версии.

04

Подтвердить

Непрерывная интеграция подтверждает, что инструкция работает.

