



ИТ-КЛАСТЕР
СИБИРИ

Козловский А.А., Мельникова О.И., Шириков И.В.

ФГБОУ ВО «Университет «Дубна», Дубна

kozlovsky@jinr.ru, oimelnik@mail.ru, ishirikov@jinr.ru

Важность воспитания архитектурного мышления у студентов IT направлений

15-16 мая 2025 года, г. Омск

О себе

Специалист, преподаватель, магистр



Проблема

- Сегодня молодые специалисты-разработчики ПО вынуждены получать все больше и больше принципиально новых навыков уже на работе
- С каждым годом количество таких навыков растет
- Разработка ПО не является «программированием»
- Архитектуре ПО уделяется крайне мало внимания в университетах

Изменение представлений о разработке ПО

Программирование
считается рутиной (до
1968)

Становится понятна
важность архитектуры
ПО (1968)

Принимается факт
невозможности
создания «общей»
архитектуры ПО (1974)

Кризис ПО 1968

- Была показана корреляция между сложностью аппаратного и программного обеспечения
- Стало понятно, что необходимо пересматривать подходы к программированию
- Стала понятна необходимость в таком направлении как «Программная инженерия»
- Начался поиск «единой архитектуры ПО»

Программирование – комплексная задача

«...пока не было машин, программирование вообще не было проблемой; когда у нас было несколько слабых компьютеров, программирование стало легкой проблемой, а теперь, когда у нас есть гигантские компьютеры, программирование стало не менее гигантской проблемой»

«по сравнению с числом семантических уровней средняя математическая теория кажется почти плоской. Создавая потребность в глубоких концептуальных иерархиях, компьютерные технологии бросают нам абсолютно новый интеллектуальный вызов, не имеющих прецедентов в истории»

Эдсгер Дейкстра

Отсутствие серебряной пули

- Фредом Брукс заявляет, что серебряной пули нет*
- Специалисты начинают концентрироваться на конкретных задачах
- Становится очевидна творческая составляющая и необходимость программной инженерии

* **Brooks, F. P.** (1975). *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley.

Изменение разработки ПО со временем

- Современная **разработка ПО невозможна** без **сторонних** технических **средств** (фреймворки, библиотеки и т.п.)
- Любой **проект** разработки ПО сегодня – **командный**
- Один специалист не может **досконально знать** разрабатываемую систему

Проблема в контексте преподавания

- Образовательный стандарт 09.03.04 «Программная инженерия» достаточно **скудно упоминает** о **проектировании** ПО
- Профстандарт **06.003 (Архитектор ПО)** не упоминается в ФГОС «Программная инженерия» **09.03.04**
- Связь теории и практики в проектировании и реализации неочевидна и сложна для понимания
- Студентам не указывают явно на связь теоретического проектирования и его практического применения
- Образовательную программу невозможно менять так быстро, как меняется IT-сфера

Стандарт 06.003 не упоминается в 09.03.04

06 Связь, информационные и коммуникационные технологии*		
1.	06.001	Профессиональный стандарт "Программист", утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2013 г. № 679н (зарегистрирован Министерством юстиции Российской Федерации 18 декабря 2013 г., регистрационный № 30635), с изменением, внесенным приказом Министерства труда и социальной защиты Российской Федерации от 12 декабря 2016 г. № 727н (зарегистрирован Министерством юстиции Российской Федерации 13 января 2017 г., регистрационный № 45230)
2.	06.004	Профессиональный стандарт "Специалист по тестированию в области информационных технологий", утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 11 апреля 2014 г. № 225н (зарегистрирован Министерством юстиции Российской Федерации 9 июня 2014 г., регистрационный № 32623), с изменением, внесенным приказом Министерства труда и социальной защиты Российской Федерации от 12 декабря 2016 г. № 727н (зарегистрирован Министерством юстиции Российской Федерации 13 января 2017 г., регистрационный № 45230)
3.	06.022	Профессиональный стандарт "Системный аналитик", утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 28 октября 2014 г. № 809н (зарегистрирован Министерством юстиции Российской Федерации 24 ноября 2014 г., регистрационный № 34882), с изменением, внесенным приказом Министерства труда и социальной защиты Российской Федерации от 12 декабря 2016 г. № 727н (зарегистрирован Министерством юстиции Российской Федерации 13 января 2017 г., регистрационный № 45230)
4.	06.028	Профессиональный стандарт "Системный программист", утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 5 октября 2015 г. № 685н (зарегистрирован Министерством юстиции Российской Федерации 20 октября 2015 г., регистрационный № 39374)

* ФГОС 09.03.04 Программная инженерия

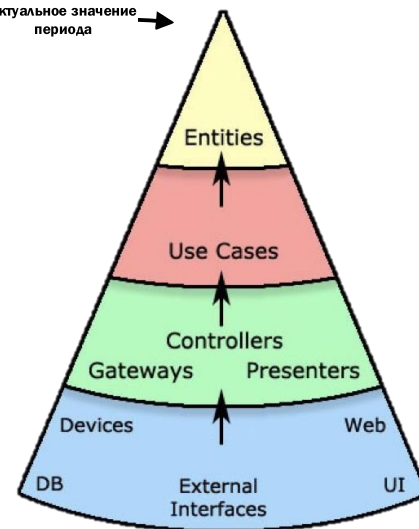
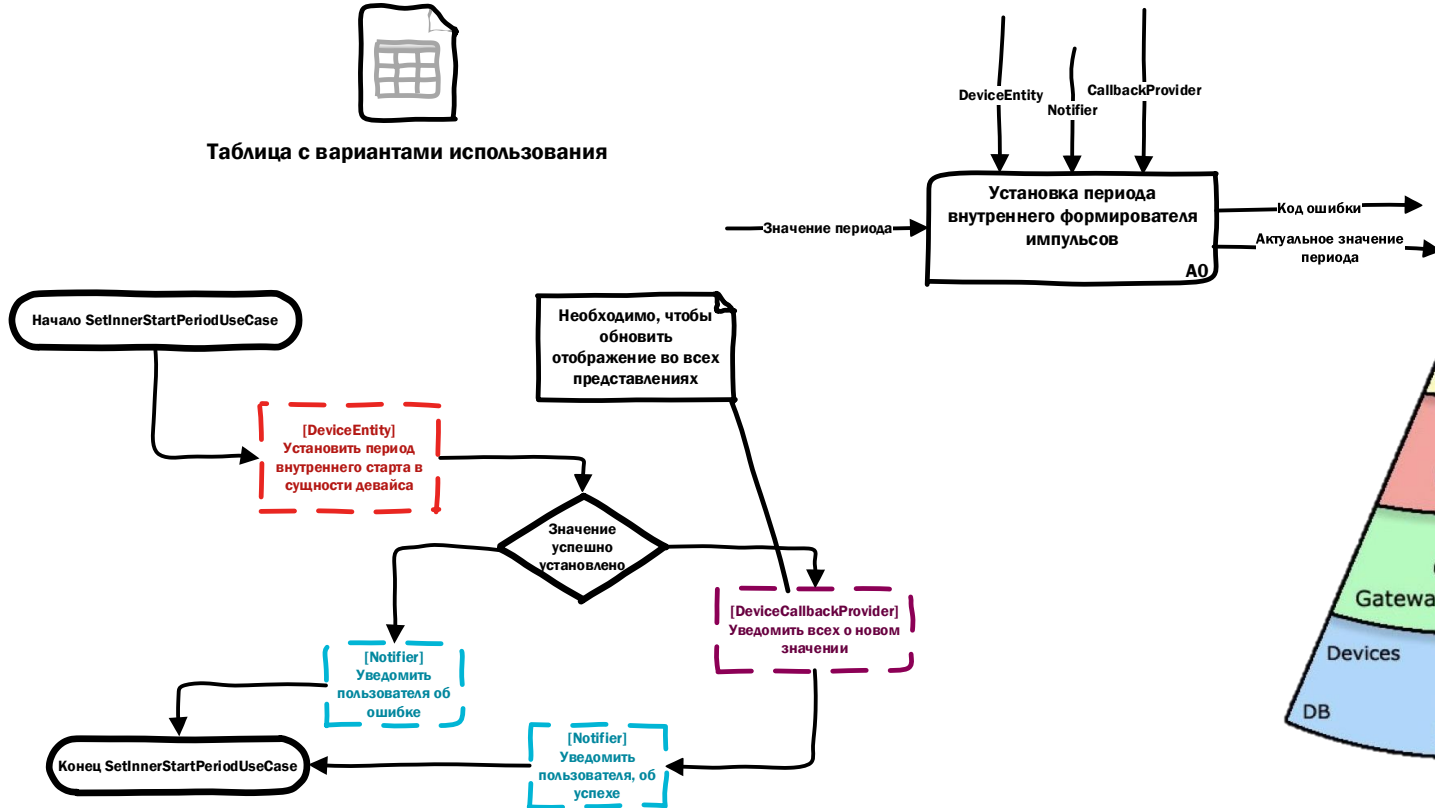
Способы решения проблемы преподавания архитектуры ПО

- Необходимо с **ранних курсов** обучать студентов **архитектурным принципам**
- **Обучение** должно строиться как «**сверху**», так и «**снизу**»
- В преподавании стоит делать **явные акценты** на **связи теоретического проектирования и практической реализации**
- Студенты должны **постоянно общаться** с **практикующими специалистами** сферы разработки и архитектуры ПО
- Необходимо указать профстандарт **06.003 (Архитектор ПО)** в ФГОС «Программная инженерия» **09.03.04**

Пример связи проектирования и реализации



Таблица с вариантами использования



Интерфейс варианта использования

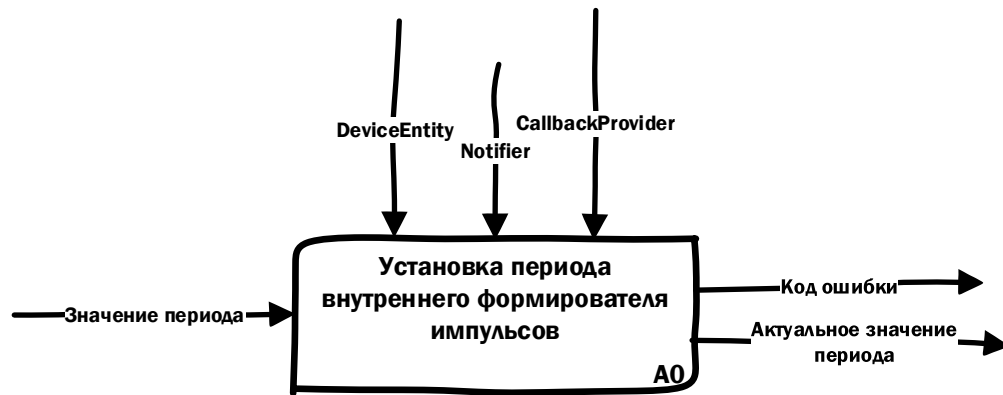
```
struct SetInnerStartPeriodUseCaseRequest {  
    uint64_t value_in_ns;  
};
```

```
struct SetInnerStartPeriodUseCaseResponse {  
    uint64_t value_in_ns;  
    ErrorCode error_code = OPERATION_INTERRUPTED;  
};
```

```
class SetInnerStartPeriodUseCase {  
public:  
    explicit SetInnerStartPeriodUseCase(const std::shared_ptr<DeviceEntity> &device_entity,  
                                         const std::shared_ptr<Notifier> &notifier,  
                                         const std::shared_ptr<CallbackProvider> &callback_provider);
```

```
SetInnerStartPeriodUseCaseResponse execute(SetInnerStartPeriodUseCaseRequest request);
```

```
private:  
    std::shared_ptr<DeviceEntity> _device_entity = nullptr;  
    std::shared_ptr<Notifier> _notifier = nullptr;  
    std::shared_ptr<CallbackProvider> _callback_provider = nullptr;  
};
```

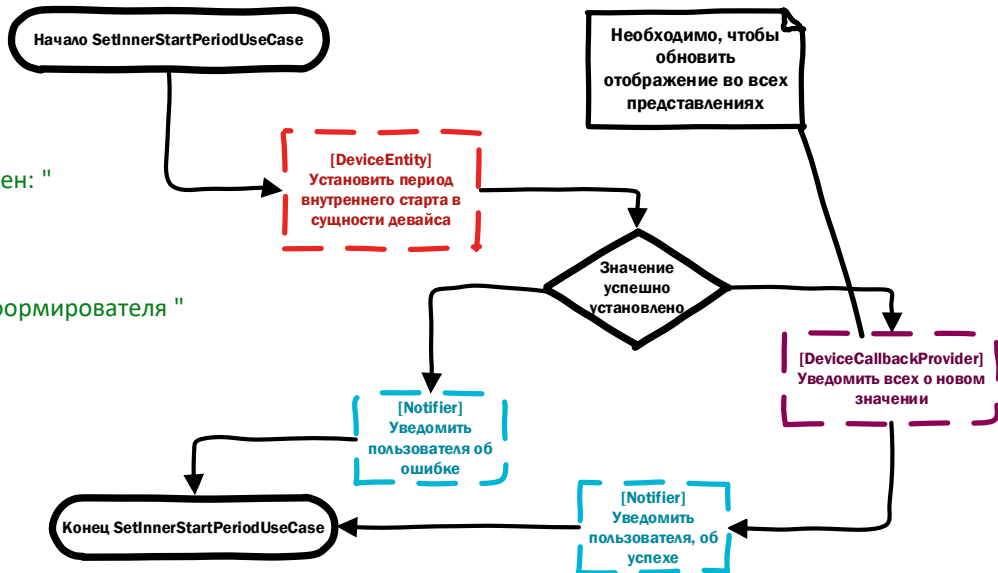


Реализация варианта использования

```
SetInnerStartPeriodUseCaseResponse SetInnerStartPeriodUseCase::execute(SetInnerStartPeriodUseCaseResponse request) {  
    SetInnerStartPeriodUseCaseResponse response;
```

```
    auto entity_response = _device_entity->setInnerStartPeriod({request.value_in_ns});  
    response.error_code = entity_response.error_code;
```

```
    if (response.error_code == SUCCESS) {  
        response.value_in_ns = entity_response.value_in_ns;  
        _callback_provider->innerStartPeriodChanged(entity);  
        _notifier->notify(NT_INFO,  
            "Период внутреннего формирователя установлен: "  
            + std::to_string(response.value_in_ns));  
    } else {  
        _notifier->notify(NT_ERROR,  
            "Ошибка при установке периода внутреннего формирователя "  
            + response.error_code);  
    }  
  
    return response;  
}
```



Важность метафоры

- Принципы проектирования построены на метафорах, аналогиях и других средствах выразительности
- Современный код пишется не для машин, а для людей
- Ребенок учится говорить и понимать мир через сказки и метафоры, только потом изучая формальные правила языка и речь
- Фазовый переход от программирования к разработке не происходит сам собой
- Программирование = алгоритмы + структуры данных
- Разработка ПО != программирование
- Разработка ПО > программирование

Важность практической проектной деятельности

- Проектирование ПО – «грязная проблема»*
- Первая версия разработанного ПО всегда заменяется новой**

*«Грязная» проблему – это проблема, которую можно ясно определить только путем полного или частичного решения****

* **McConnell, S.** (2004). *Code Complete: A Practical Handbook of Software Construction* (2nd ed.). Microsoft Press.

** **Brooks, F. P.** (1975). *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley.

*** **Rittel, H. W. J., & Webber, M. M.** (1973). Dilemmas in a General Theory of Planning. *Policy Sciences*, 4(2)

Важность систематизации подходов

- Многие принципы и методы работы с архитектурами уже выделены в ряде работ Нила Форда, Мартина Фаулера, Роберта Мартина и др.
- Многие подходы изучаются уже непосредственно на работе «с нуля»
- CMU SEI имеет огромное количество публикаций с систематизацией подходов к проектированию архитектуры ПО
- IBS предоставляет замечательные курсы по архитектуре ПО
- Студентов необходимо учить выбирать и применять различные подходы к архитектуре, основываясь на конкретных деталях задачи



Training Center



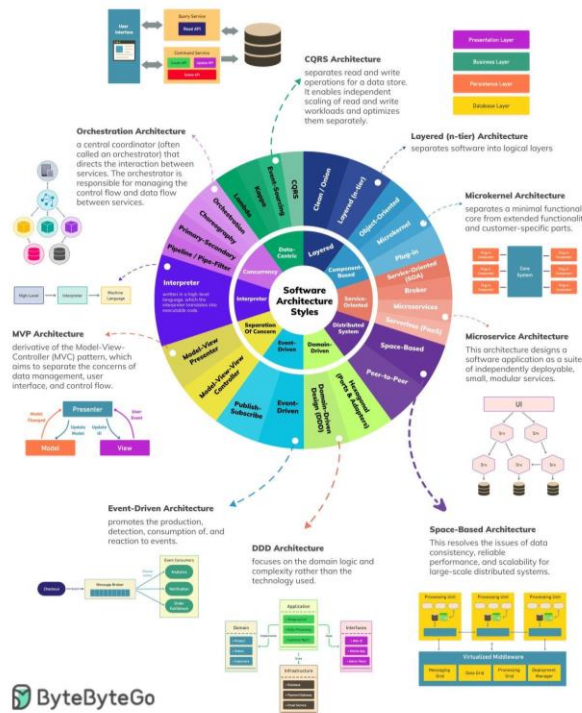
Software Engineering Institute

Важность четкого разделения уровней

- Архитектура != Проектирование
- Проектирование > Архитектура
- Архитектура, как и проектирование может рассматриваться на различных уровнях
- Студенты должны четко разделять уровни применяемых принципов проектирования

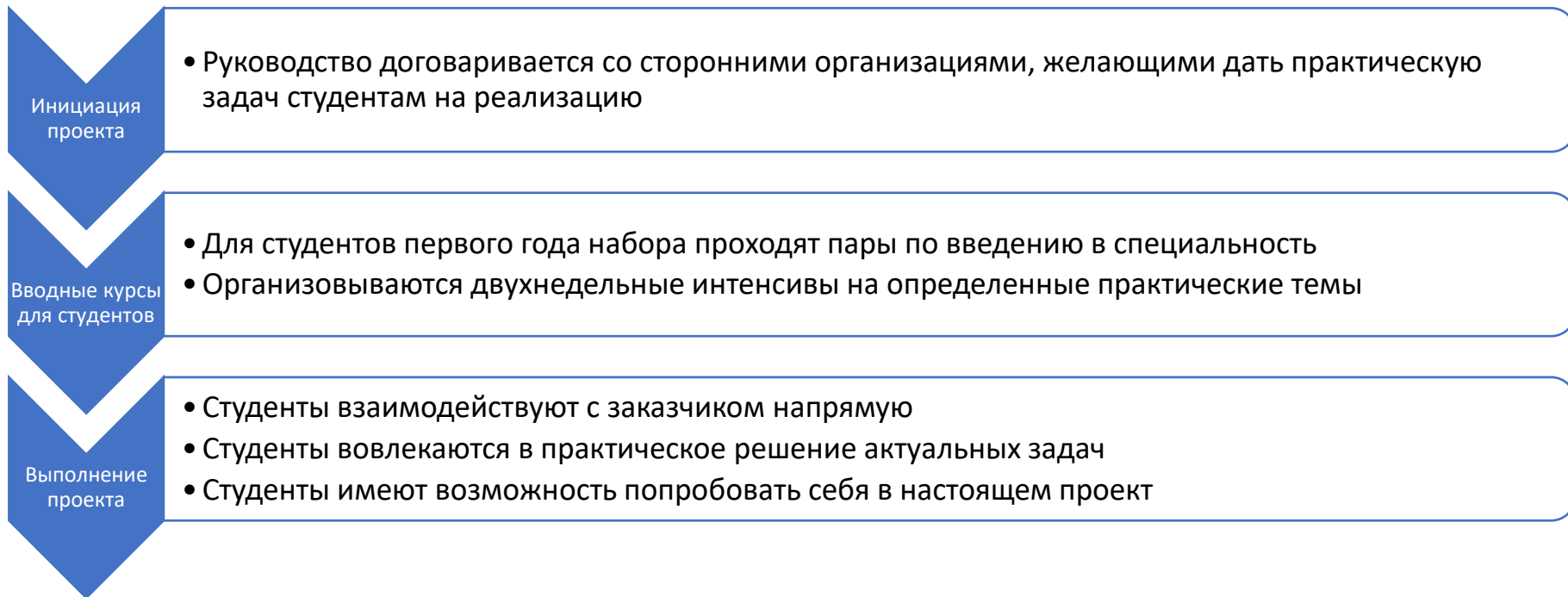


Необходимо стандартизировать подходы к преподаванию «разработки ПО»



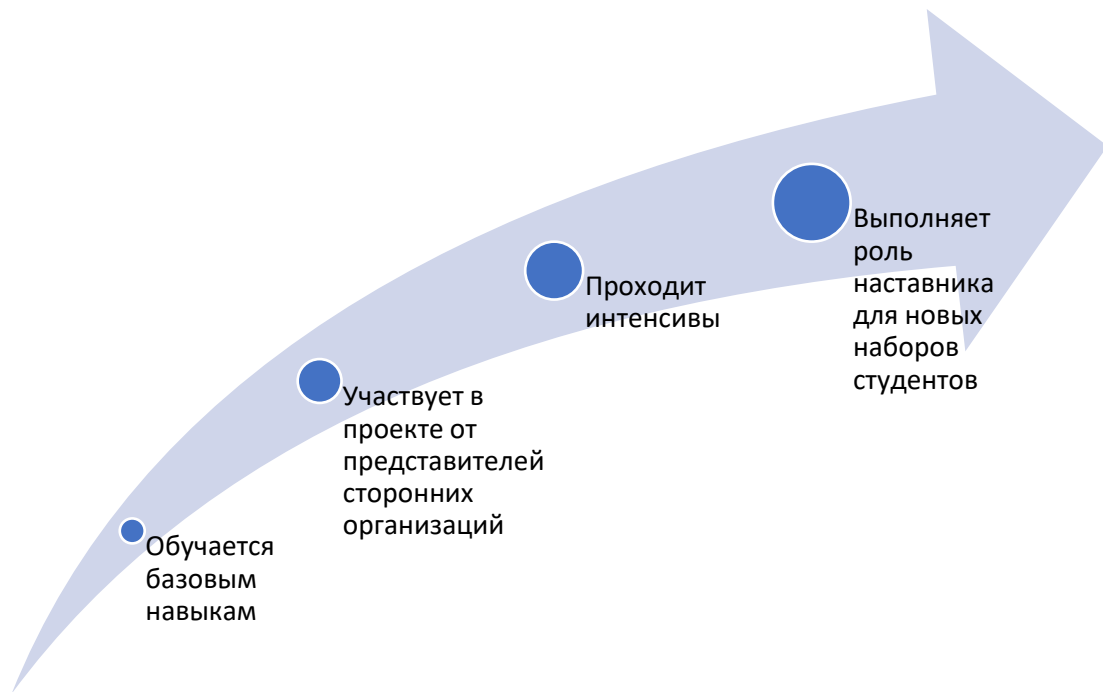
А что же касается разработки ПО?...

Схема работы МИШ



*Студенты бьются на три направления: **Электроника, Программирование и Конструирование***

Путь студента в МИШ



МИШ должна создавать свою систему кадров, проектов, наставников и кураторов, которая будет расти со временем

Студенты должны получать навыки через реальную практическую проектную деятельность

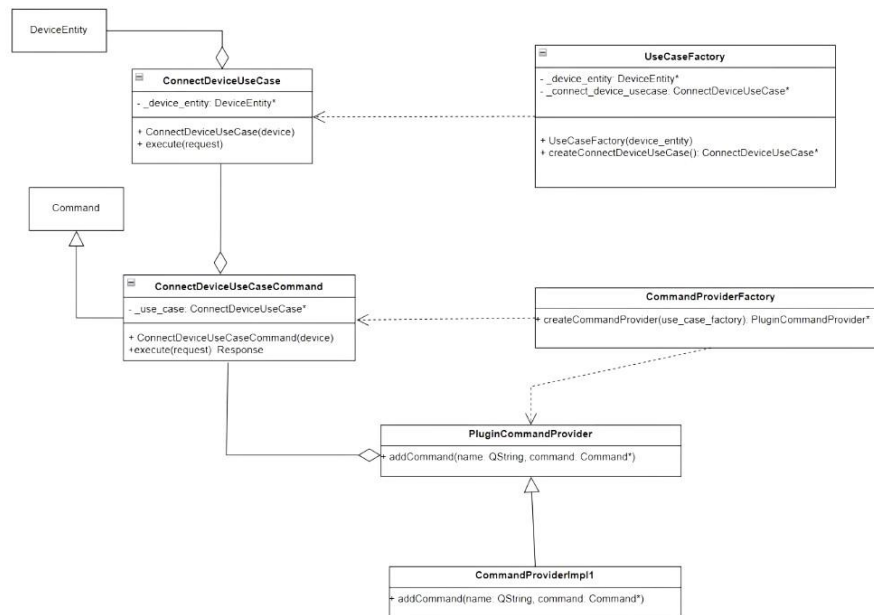
Проекты МИШ должны становиться все более комплексными с каждым годом

Примеры результатов по проекту МИШ

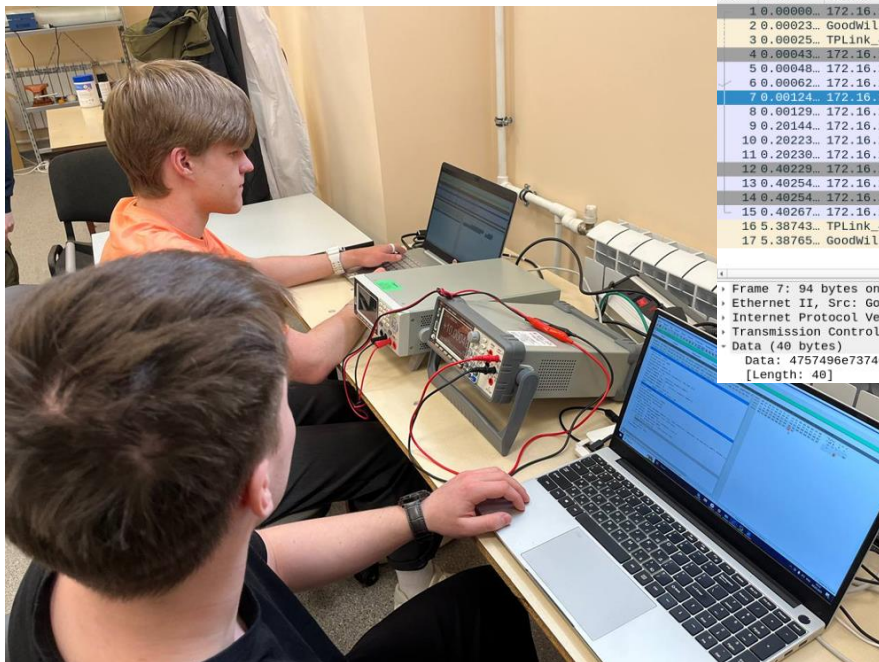
- Проект: «**Разработка ПО для управления измерительной аппаратурой фирмы GWInstek**»
- Результаты:
 - Разработан модуль TCP для общения с устройствами
 - Составлена таблица используемых SCPI-команд с примерами вызовов и обработки ответов
 - Частично реализован графический пользовательский интерфейс
 - Реализованы сервисные программные модули для взаимодействия с измерительными устройствами
 - Реализованы варианты использования ПО, согласно чистой архитектуры

Работа в проекте на всех уровнях

Таблица2					
Команда/запрос	Описание команды (запроса)	Поддерживаемые параметры	Пример запроса (команды)	Пример ответа	Комментарии к команде (запросу)
Команды измерения					
MEAS:VOLT:DC?	Возвращает значение измерения постоянного напряжения на 1ST дисплей	[None] [Range(MIN MAX DEF)], Resolution(MIN MAX DEF)]	MEAS:VOLT:DC? MIN:n	=+6.64925206E-04	
MEAS:VOLT:AC?	Возвращает значение измерения напряжения переменного тока на 1ST дисплей.	[None] [Range(MIN MAX DEF)]	MEAS:VOLT:AC?n	=+1.34567684E-04	
MEAS:CURREN:DC?	Возвращает значение измерения постоянного тока на 1ST дисплей.	[None] [Range(MIN MAX DEF)], Resolution(MIN MAX DEF)]	MEAS:CURREN:DC? 0.1n	-1.09750431E-07	
MEAS:CURREN:AC?	Возвращает значение измерения переменного тока на 1ST дисплей.	[None] [Range(MIN MAX DEF)]	MEAS:CURREN:AC?n	=+1.46445157E-07	
	Возвращает значение	[None]			



Работа в проекте на всех уровнях



No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000	172.16.236...	172.16.236...	TCP	74	47008 → 5025 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=641894963 TSecr=0 WS=128
2	0.00023	GoodWillIns...	Broadcast	ARP	60	Who has 172.16.236.10? Tell 172.16.236.170
3	0.00025	TPLink_4c:8...	GoodWillIns...	ARP	42	172.16.236.101 is at 7c:c2:c6:4c:8b:48
4	0.00043	172.16.236...	172.16.236...	TCP	60	5025 → 47008 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
5	0.00048	172.16.236...	172.16.236...	TCP	54	47008 → 5025 [ACK] Seq=1 Ack=1 Win=64240 Len=0
6	0.00062	172.16.236...	172.16.236...	TCP	60	47008 → 5025 [PSH, ACK] Seq=1 Ack=1 Win=64240 Len=6
7	0.00124	172.16.236...	172.16.236...	TCP	94	5025 → 47008 [PSH, ACK] Seq=1 Ack=7 Win=5834 Len=40
8	0.00129	172.16.236...	172.16.236...	TCP	54	47008 → 5025 [ACK] Seq=7 Ack=41 Win=64200 Len=0
9	0.00144	172.16.236...	172.16.236...	TCP	60	47008 → 5025 [PSH, ACK] Seq=7 Ack=41 Win=64200 Len=6
10	0.00223	172.16.236...	172.16.236...	TCP	60	5025 → 47008 [PSH, ACK] Seq=41 Ack=13 Win=5828 Len=5
11	0.00230	172.16.236...	172.16.236...	TCP	54	47008 → 5025 [ACK] Seq=13 Ack=46 Win=64195 Len=0
12	0.00239	172.16.236...	172.16.236...	TCP	54	47008 → 5025 [FIN, ACK] Seq=13 Ack=46 Win=64195 Len=0
13	0.00254	172.16.236...	172.16.236...	TCP	62	5025 → 47008 [ACK] Seq=46 Ack=14 Win=5827 Len=0
14	0.00254	172.16.236...	172.16.236...	TCP	60	5025 → 47008 [FIN, ACK] Seq=46 Ack=14 Win=5827 Len=0
15	0.00267	172.16.236...	172.16.236...	TCP	54	47008 → 5025 [ACK] Seq=14 Ack=47 Win=64195 Len=0
16	5.38743	TPLink_4c:8...	GoodWillIns...	ARP	42	Who has 172.16.236.10? Tell 172.16.236.101
17	5.38765	GoodWillIns...	TPLink_4c:8...	ARP	60	172.16.236.170 is at 00:22:24:6a:50:71

Frame 7: 94 bytes on wire (752 bits), 94 bytes captured (752 bits) on...	0000	7c c2 c6 4c 8b 48 00 22	24 6a 50 71 08 08 45 00	...L.H." \$jPc.E
Ethernet II, Src: GoodWillInst_6a:50:71 (00:22:24:6a:50:71), Dst: TP...	0010	00 50 00 1b 00 00 ff 06	8a 5b ac 10 ec aa ac 10	.P.....[.....
Internet Protocol Version 4, Src: 172.16.236.170, Dst: 172.16.236.101	0020	ec 65 13 a1 b7 a0 00 06	19 86 40 eb ce ba 50 18	.e.....@...P.
Transmission Control Protocol, Src Port: 5025, Dst Port: 47008, Seq=...	0030	16 ca 3e 37 00 00 47 57	49 6e 73 74 65 6b 2c 47	-->7-GW Instek,G
Data (40 bytes)	0040	4d 4d 38 32 36 31 41 2c	47 45 58 38 35 30 36 32	DM8261A, GEX85062
Data: 47574967374656b2c47444d38323631412c4745583835303632392c40d312...	0050	39 2c 4d 31 2e 31 35 5f	53 31 2e 31 35 0a	9,M1.15_ S1.15
[Length: 40]				

Результаты тестирований оборудования
были представлены на выставке
ExpoElectroinca 2025



Международная инженерная школа (МИШ)



Инженерка

Международная инженерная школа



Выводы

- Студенты **первых** курсов готовы к **курируемой проектной** деятельности, осуществляемой с **помощью** организаций **дополнительного образования**
- **Дополнительное** образование **не должно ограничивать** основное. **Основное** образование должно получаться в **полной мере**, в обязательном порядке
- Студенты не должны **досконально** изучить теорию, **прежде** чем начинать приступать к **практике**
- **Изучение теории** не должно быть отодвинуто на второй план, но должно идти **совместно** с осуществлением **практической деятельности**
- Студентам **необходимо** давать четкое **понимание применимости** различных принципов **проектирования** и их корреляции с **практикой**
- Студентов необходимо знакомить с **актуальными** публикациями и трудами, посвященными **проблемам проектирования ПО**
- Необходимо привлекать большее число **практикующих специалистов-архитекторов** для **преподавания** в университетах или **формирования образовательных программ**
- Необходимо указать профстандарт **06.003 (Архитектор ПО)** в ФГОС «Программная инженерия» **09.03.04**

Вопросы к размышлению

- Достаточно ли 4 лет для обучения «Программной инженерии»?
- Какие принципы и методы программной инженерии должны быть включены в образовательную программу на государственном уровне?

Спасибо за внимание