ОТ ИДЕИ ДО ДЕПЛОЯ

ЦИКЛ РАЗРАБОТКИ ТИПОВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Цель лекции

Изучить основные фазы создания программного обеспечения и разобрать типовые ошибки

Модули

лекции

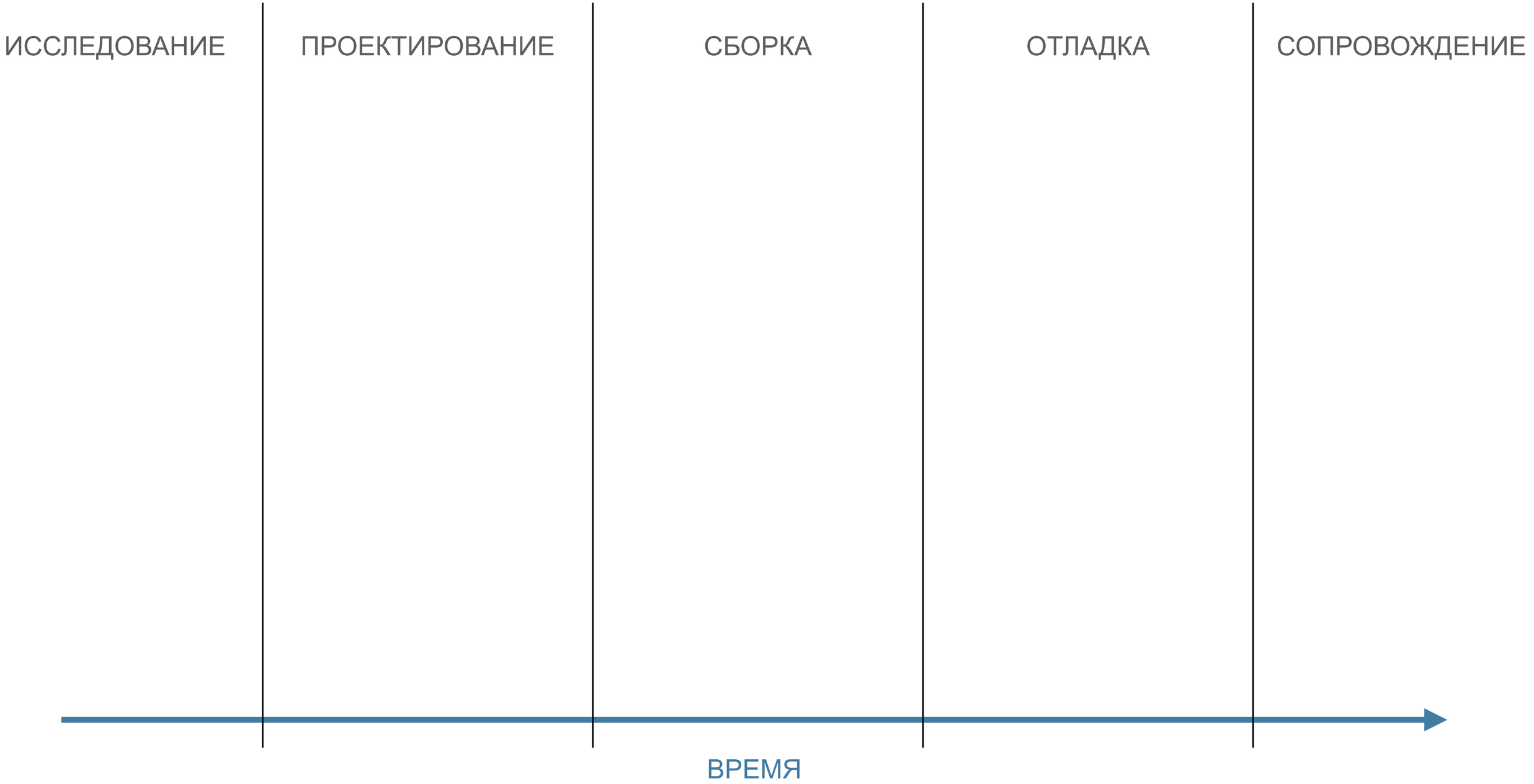
- Жизненный цикл системы
- Исследования
- Проектирование
- Сборка
- Тестирование
- Сопровождение
- Типичные ошибки

1/7
ЖИЗНЕННЫЙ
ЦИКЛ ДИАЛОГОВОЙ СИСТЕМЫ

Цель модуля

Изучить термин - жизненный цикл диалоговой системы

Жизненный цикл системы — это стадии процесса, охватывающие различные состояния системы, начиная с момента возникновения необходимости в такой системе и заканчивая её полным выводом из эксплуатации


ИССЛЕДОВАНИЕ
ПРОЕКТИРОВАНИЕ
СБОРКА
ОТЛАДКА
СОПРОВОЖДЕНИЕ
ВРЕМЯ

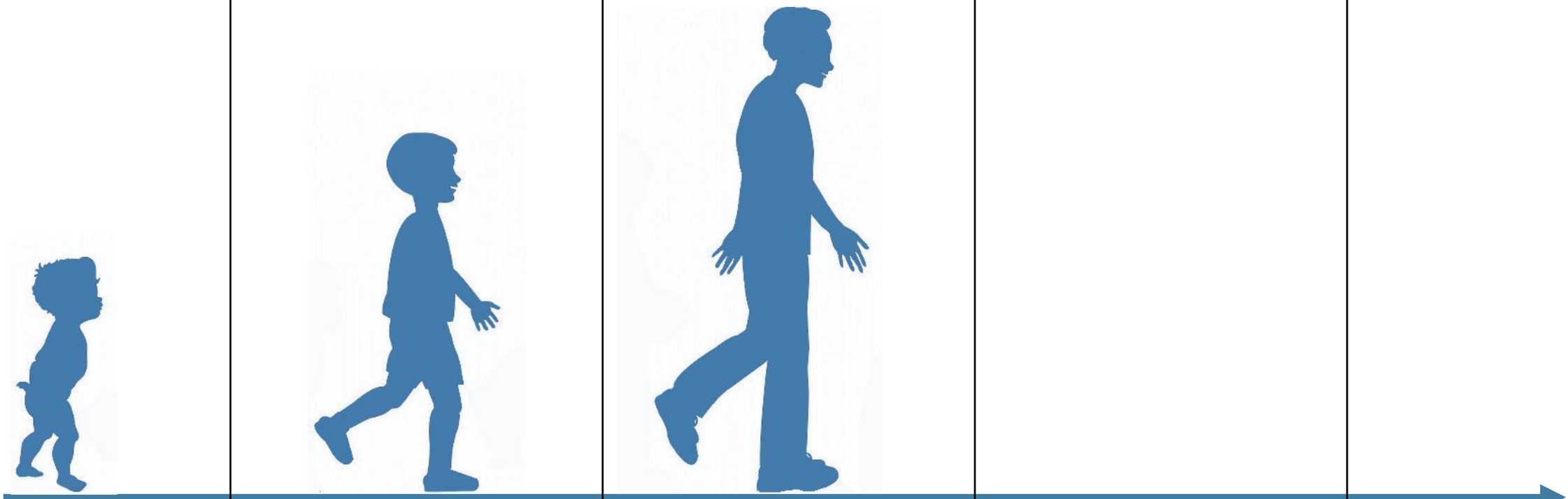
ИССЛЕДОВАНИЕ | ПРОЕКТИРОВАНИЕ | СБОРКА | ОТЛАДКА | СОПРОВОЖДЕНИЕ

ВРЕМЯ

ИССЛЕДОВАНИЕ | ПРОЕКТИРОВАНИЕ | СБОРКА | ОТЛАДКА | СОПРОВОЖДЕНИЕ

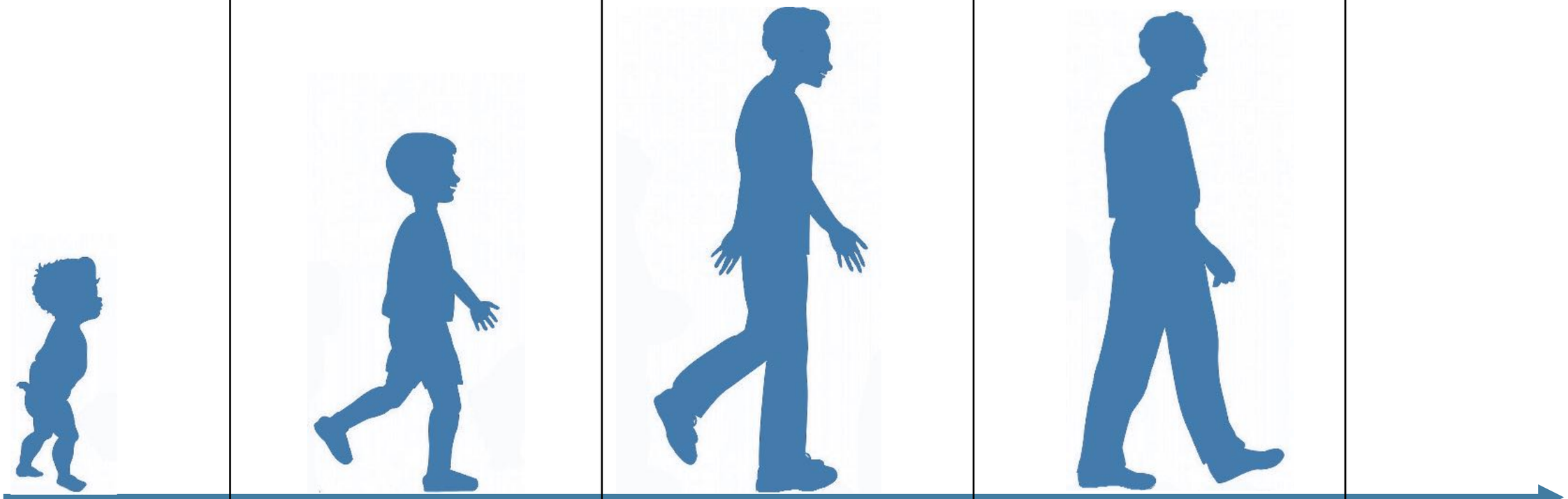
ВРЕМЯ

ИССЛЕДОВАНИЕ | ПРОЕКТИРОВАНИЕ | СБОРКА | ОТЛАДКА | СОПРОВОЖДЕНИЕ



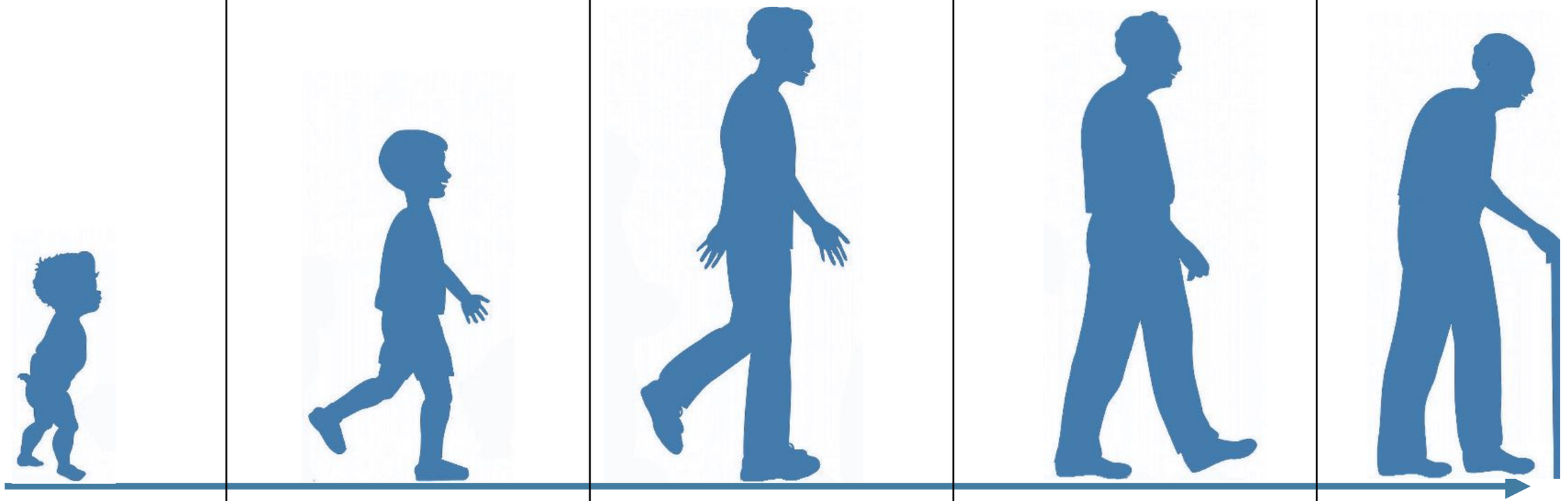
ВРЕМЯ

ИССЛЕДОВАНИЕ | ПРОЕКТИРОВАНИЕ | СБОРКА | ОТЛАДКА | СОПРОВОЖДЕНИЕ



ВРЕМЯ

ИССЛЕДОВАНИЕ | ПРОЕКТИРОВАНИЕ | СБОРКА | ОТЛАДКА | СОПРОВОЖДЕНИЕ



ВРЕМЯ

2/7
ИССЛЕДОВАНИЕ
ПРЕДМЕТНОЙ ОБЛАСТИ

Цель модуля

Изучить термины и инструменты применяемые на этапе исследования предметной области

Исследование — в предельно широком смысле это поиск новых знаний или систематическое расследование с целью установления фактов

Методы

научного познания

- Наблюдение – эксперимент,
- Анализ – синтез
- Индукция – дедукция

Наблюдение — описательный психологический исследовательский метод, заключающийся в целенаправленном и организованном восприятии и регистрации поведения изучаемого объекта

Пример:

Наблюдение — при наведении курсора, в нижней части видео появляются иконки

Эксперимент — процедура, выполняемая для поддержки, опровержения или подтверждения гипотезы или теории

Пример:

Гипотеза — если нажать , на иконку в форме шестеренки, появится интерфейс настройки параметров воспроизведения

Эксперимент — нажимаем на иконку, всплывает интерфейс выбора параметров

Гипотеза подтверждена.

Анализ (др.-греч. ἀνάλυσις — разложение, расчленение, разборка) — метод исследования, характеризующийся выделением и изучением отдельных частей объектов исследования

Пример:

Система настройки параметров воспроизведения состоит из элементов:

1. Иконка
2. Меню выбора параметров качества и скорости воспроизведения
3. Меню выбора качества воспроизведения с несколькими значениями
4. Меню выбора скорости воспроизведения с несколькими значениями

Синтез — процесс соединения или объединения ранее разрозненных вещей или понятий в целое или набор

Пример:

Система выбора параметров воспроизведения позваляет индивидуально настроить качество видео и скорость воспроизведения ролика

Индукция — познавательная процедура, посредством которой из сравнения наличных фактов выводится обобщающее их утверждение

Пример:

Факт — иконка в виде шестерни, в нижней части видео, подсказала мне смысл скрытых под этим интерфейсом функции.

Гипотеза — формы и цвет иконок служат подсказками для пользователей

Дедукция — это логико-методологическая процедура, посредством которой осуществляется переход от общего к частному в процессе рассуждения

Пример:

Гипотеза — иконка X похожа на Y, поэтому является интерфейсом управления функциями системы Z

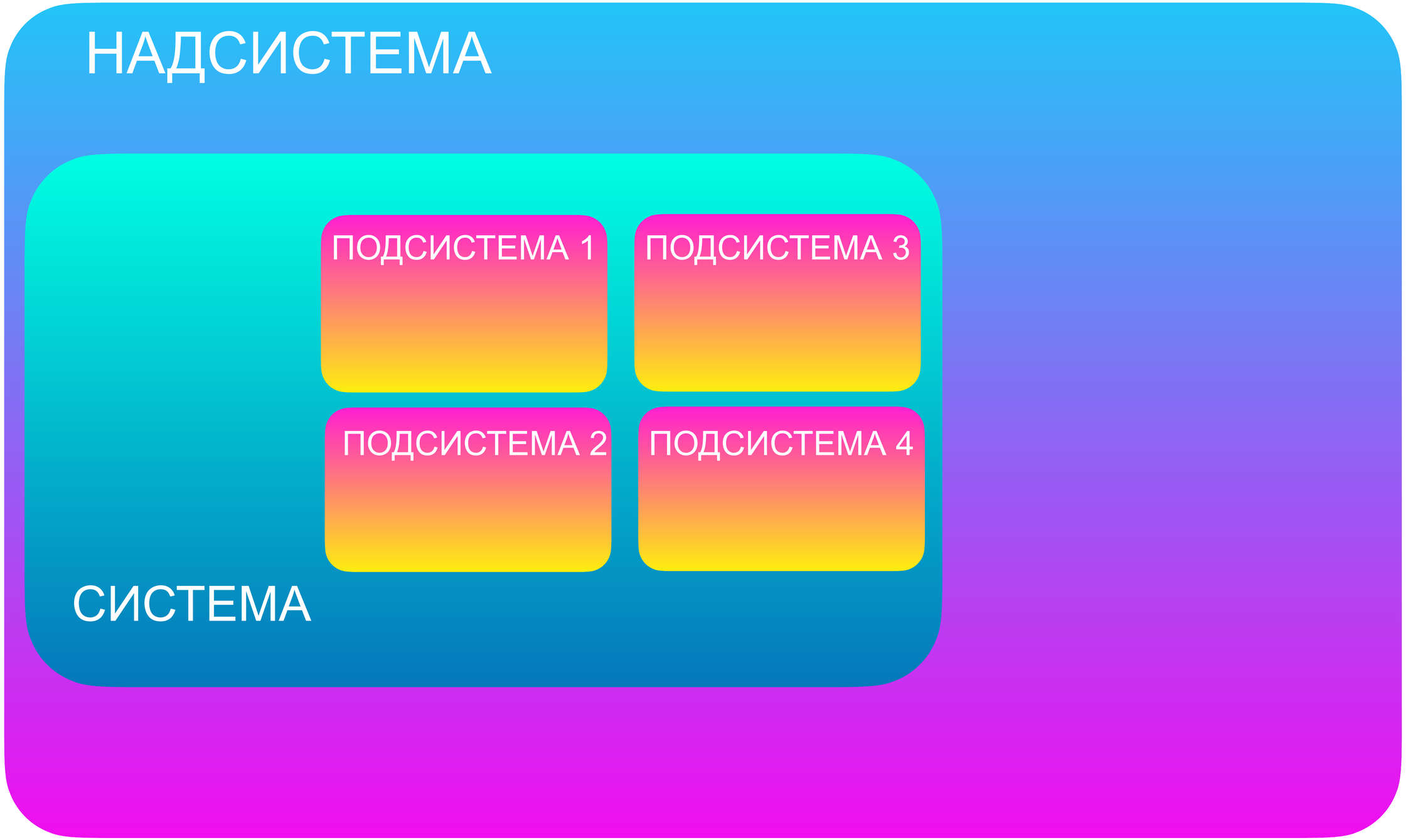
Предметная область — часть реального мира, рассматриваемая в пределах данного контекста

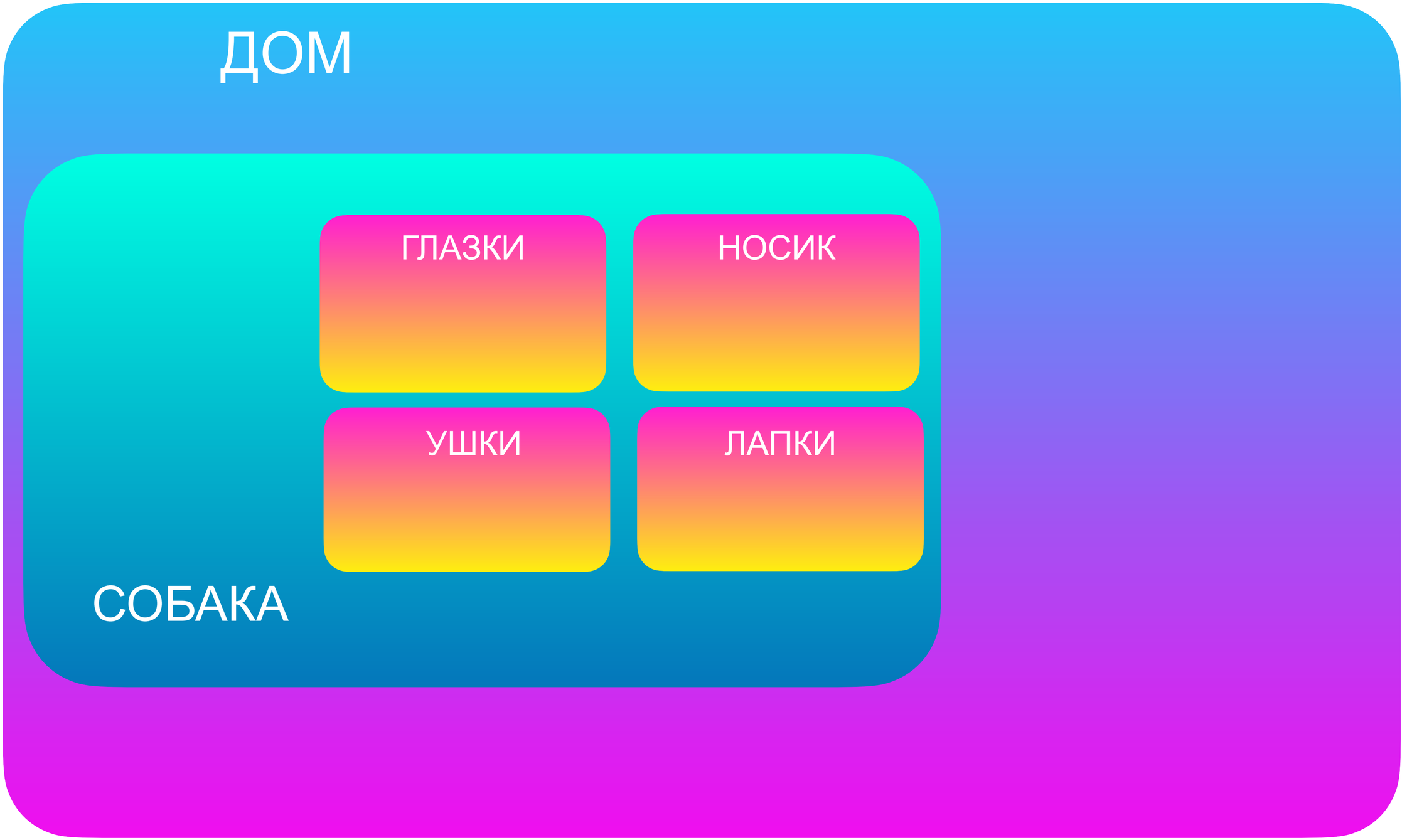
Этапы

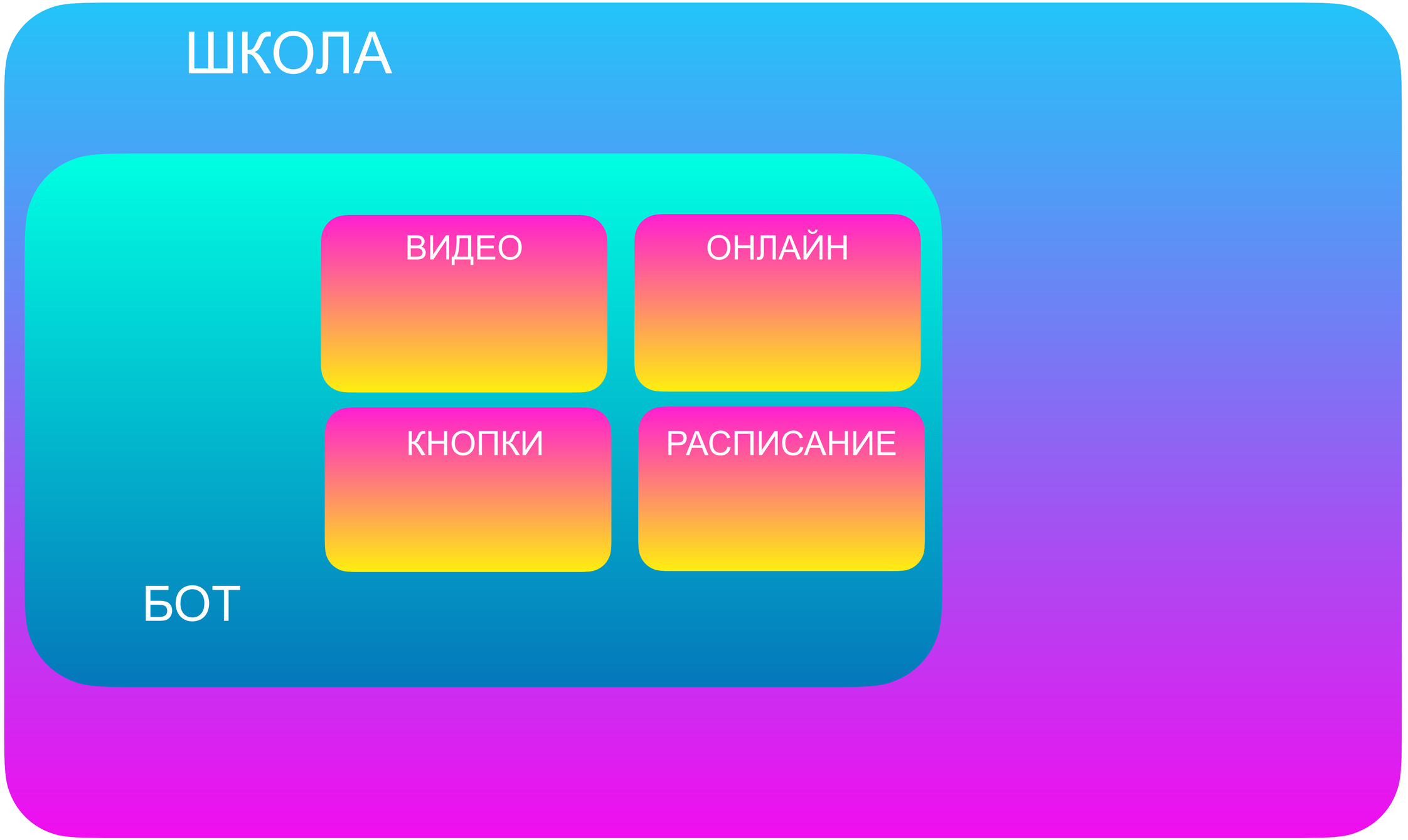
исследования предметной области

- Исследуем хаос - выявляем ключевые элементы и функции системы
- Исследуем сложность - классифицируем ключевые подсистемы, выявляем взаимосвязи
- Исследуем управление - выявляем последствия воздействий на элементы и связи. Учимся управлять, предсказывать, трансформировать
- Исследуем использование - используем, наблюдаем, дополняем знания о системе, корректируем взаимодействие.

Исследуем хаос

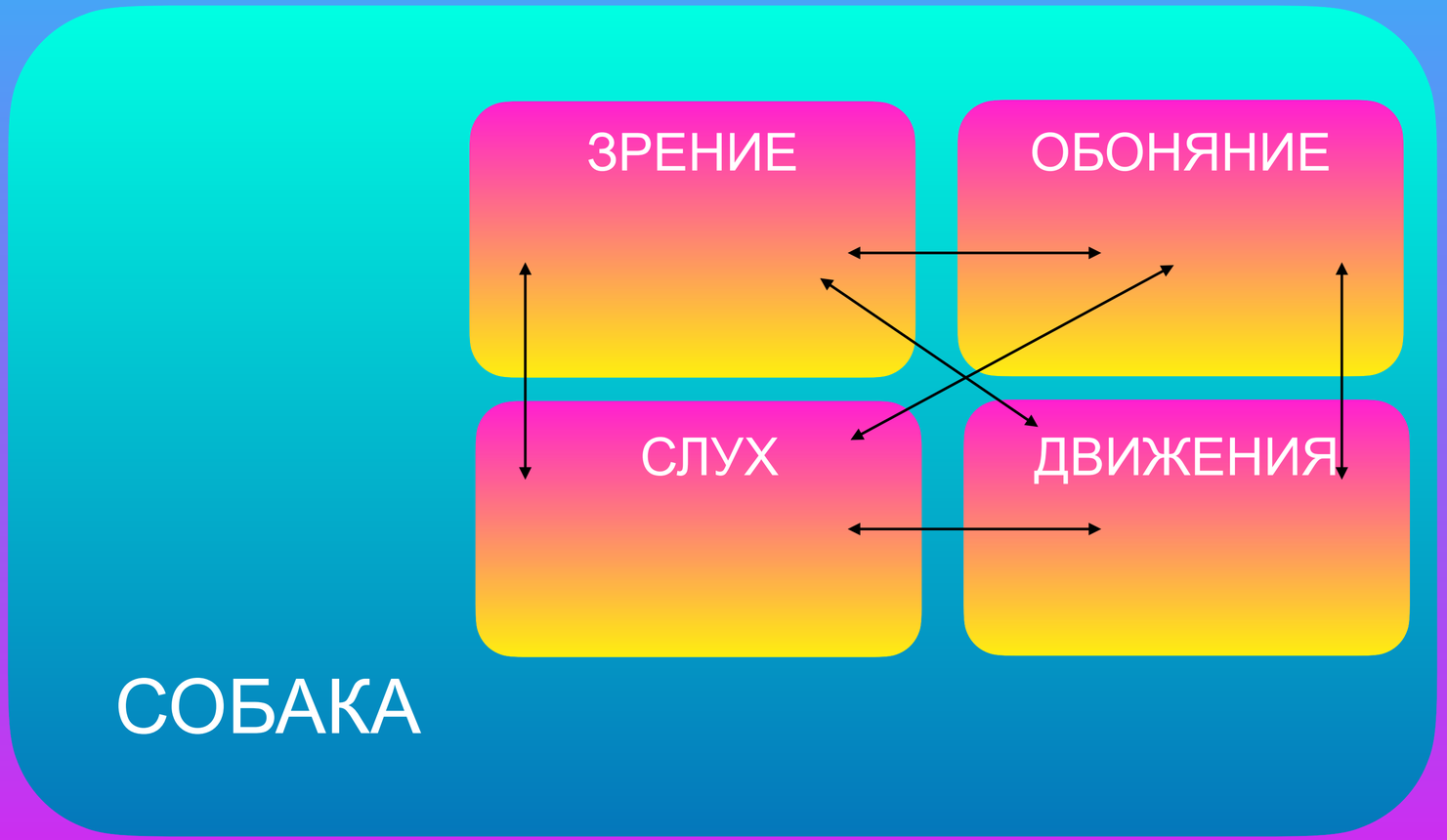

НАДСИСТЕМА
ПОДСИСТЕМА 1
ПОДСИСТЕМА 3
ПОДСИСТЕМА 2
ПОДСИСТЕМА 4
СИСТЕМА

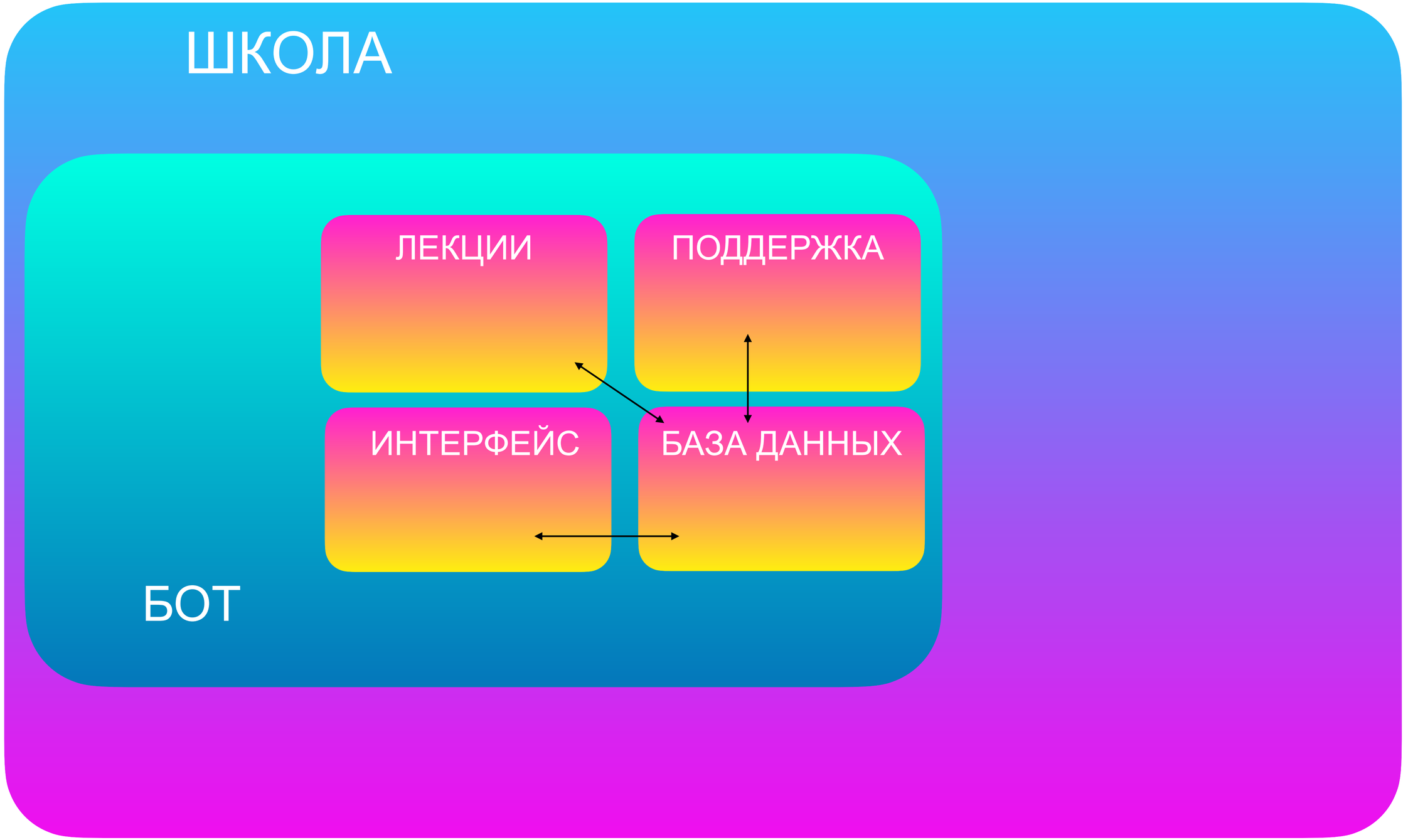

ДОМ
ГЛАЗКИ
НОСИК
УШКИ
ЛАПКИ
СОБАКА


ШКОЛА
ВИДЕО
ОНЛАЙН
КНОПКИ
РАСПИСАНИЕ
БОТ

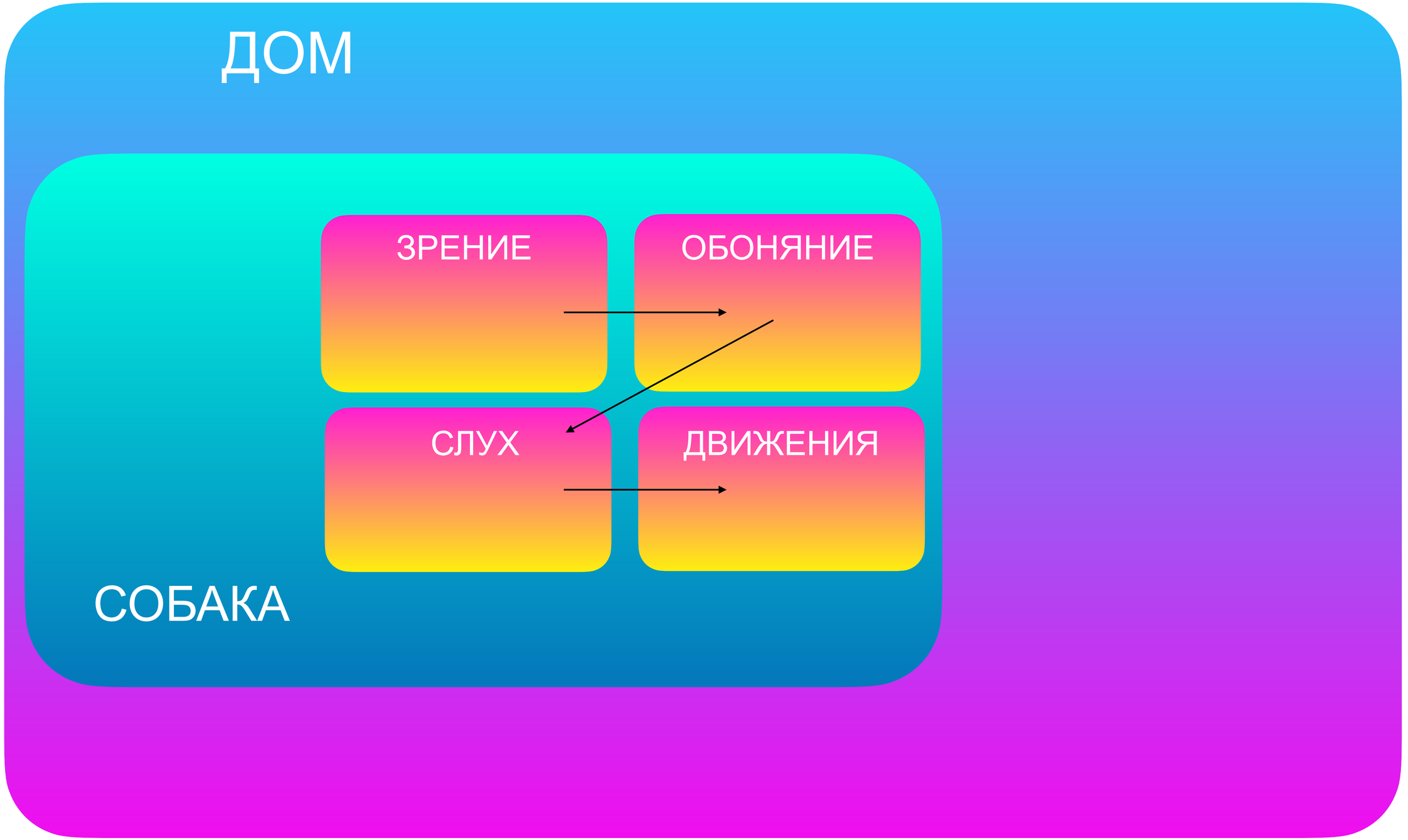
Исследуем сложность

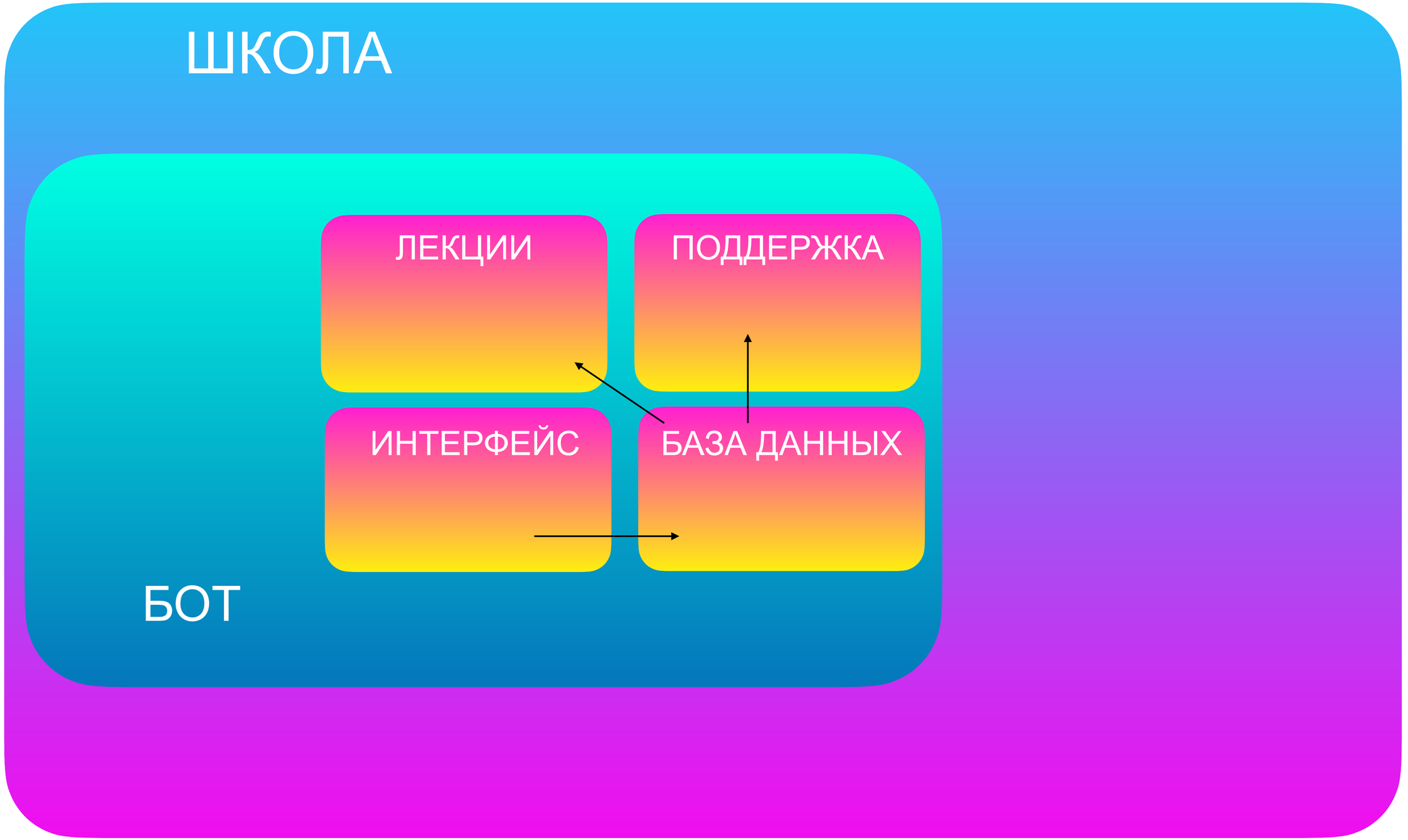
ДОМ




ШКОЛА
ЛЕКЦИИ
ПОДДЕРЖКА
ИНТЕРФЕЙС
БАЗА ДАННЫХ
БОТ

Исследуем управление


ДОМ
ЗРЕНИЕ
ОБОНЯНИЕ
СЛУХ
ДВИЖЕНИЯ
СОБАКА


ШКОЛА
ЛЕКЦИИ
ПОДДЕРЖКА
ИНТЕРФЕЙС
БАЗА ДАННЫХ
БОТ

Исследуем использование

Аналитические

инструменты

- Целеполагание
- Наблюдение и описание элементов, свойств и функций
- Экспериментальное исследование подобных систем
- Анализ конкурентного окружения, стейкхолдеров, фичь и т.д.
- Объединение и интеграция элементов и подсистем.
- Обобщающие выводы
- Позиционирование

Аналитические артефакты

Требования

к диалоговой системе

- Чат-бот телеграм
- База данных со ссылками на видеолекции, онлайн поддержку и другой контент
- Интерфейс выбора лекций/поддержки
- Интерфейс перехода к следующей лекции

3/7
ПРОЕКТИРОВАНИЕ
ДИАЛОГОВОЙ СИСТЕМЫ

Цель модуля

Изучить термины и инструменты применяемые на этапе проектирования диалоговой системы

Этапы

проектирования диалоговой системы

- Архитектура (сценарий)
- Прототипирование
- Дизайн
- Корректировка и утверждение
- Техническое задание в разработку

Сценарий чат-бота

Зачем — тратить время на создание сценария?

Память человека

ПАМЯТЬ ЧЕЛОВЕКА

ОГРАНИЧЕНИЯ, ШУМ, ПОМЕХИ, ЗАБЫВАНИЕ

Особенности коммуникаций

СХЕМА ПЕРЕДАЧИ ИНФОРМАЦИИ

Этапы

создания сценария

- Диалоговая цель
- Состояния диалога
- Счастливый путь
- Конечные состояния
- Элементы

1.

2.

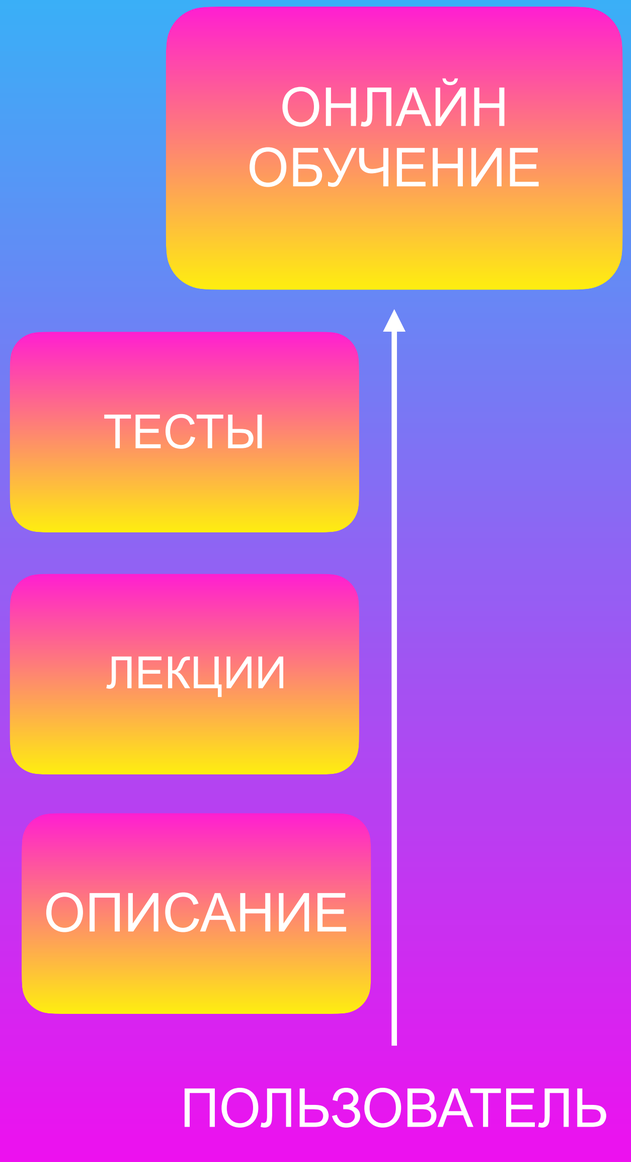
3.

4.

5.

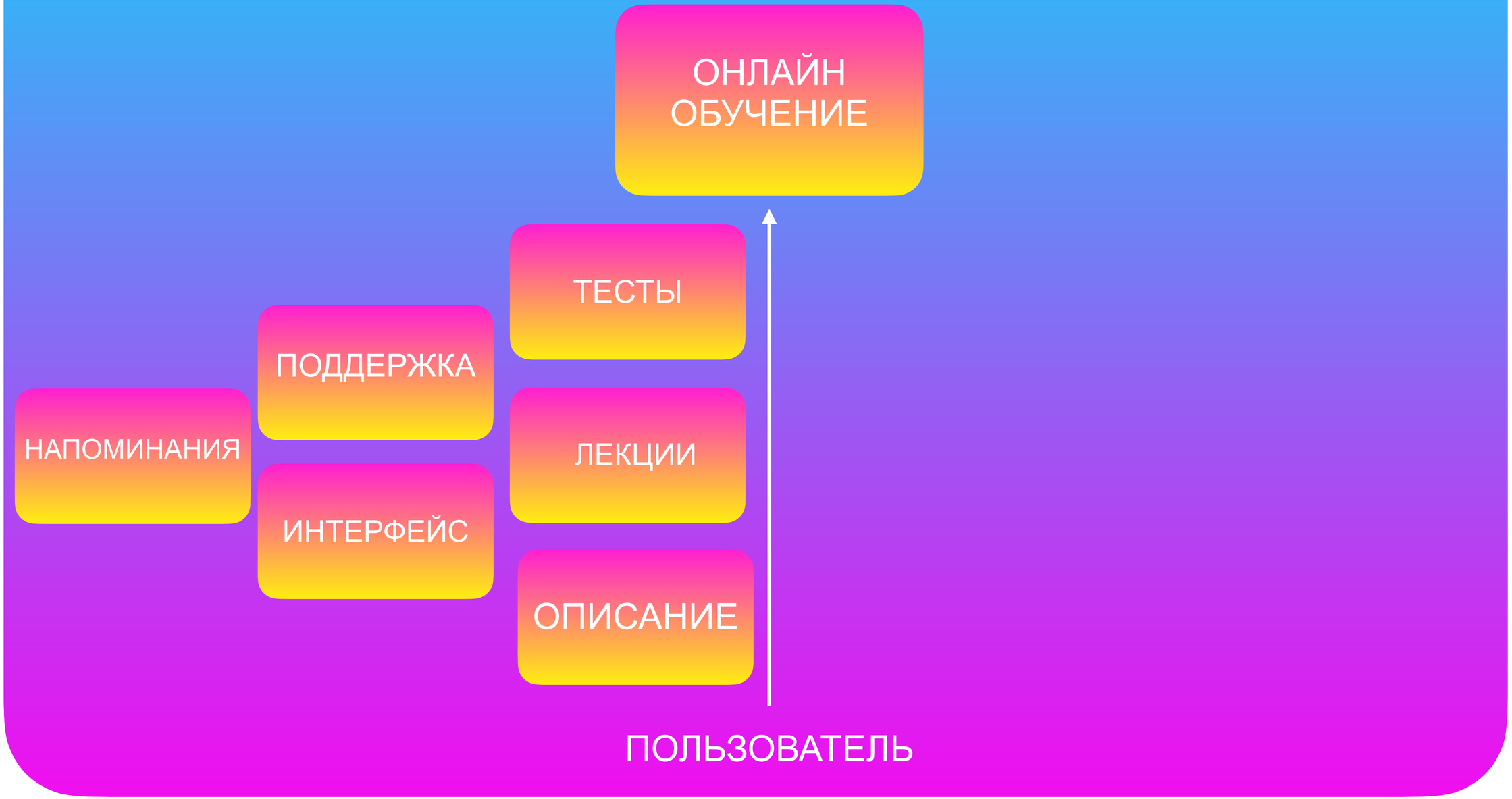
ДИАЛОГОВАЯ ЦЕЛЬ

СОСТОЯНИЯ ДИАЛОГА



ИДЕАЛЬНЫЙ СЦЕНАРИЙ

КОНЕЧНЫЕ СОСТОЯНИЯ



ЭЛЕМЕНТЫ

ОНЛАЙН ОБУЧЕНИЕ
ТЕСТЫ
ПОДДЕРЖКА
НАПОМИНАНИЯ
ЛЕКЦИИ
ИНТЕРФЕЙС
ОПИСАНИЕ
МЕТРИКИ
ИНТЕГРАЦИИ
ПЕРЕМЕННЫЕ
ОФТОПЫ
ВАРИАТИВНОСТЬ
ПОЛЬЗОВАТЕЛЬ

Прототипирование

Прототип — это ранний образец, модель или выпуск продукта, созданный для тестирования концепции или процесса.

Цикл

работы с прототипами

- Технический анализ
- Сборка прототипа
- Тестирование прототипа
- Формализация результатов, изменение документации
- Цели и задачи следующей итерации

Дизайн продукта

Красота́ — эстетическая (неутилитарная, непрактическая) категория, обозначающая совершенство, гармоничное сочетание аспектов объекта, при котором последний вызывает у наблюдателя эстетическое наслаждение.

Дизайн (от англ. *design* — проектировать, чертить, задумать, а также проект, план, рисунок) — деятельность по проектированию эстетических свойств продуктов («художественное конструирование»), а также результат этой деятельности

UX-стратегия – это план или подход к разработке или развитию цифровых продуктов, учитывающего бизнес-**стратегию**, технические возможности и потребности пользователей и достижение бизнес-результатов, создание ценности продукта и донесение ее до пользователей.

Пользовательский сценарий — это вымышленная история о том, как пользователь выполняет действия или достигает своих целей с помощью вашего продукта.

Дивергентное мышление

Дивергéнтное мышление — метод творческого мышления, применяемый обычно для решения проблем и задач. Заключается в поиске множества решений одной и той же проблемы.

Конвергентное мышление

Конвергентное мышление — это тип мышления, который фокусируется на поиске единственного, устоявшегося ответа на проблему.

Этапы

разработки дизайна

- Идея и UX стратегия
- Анализ конкурентов, трендов и стилей
- Проработка пользовательских сценариев
- Поиск, создание и аккумулирование контента
- Проработка деталей, сборка прототипов

Дизайн

ценности

- Творчество — дизайнер выходит за рамки функциональности продукта и наполняет красотой то, что другие считают простым товаром
- Эмпатия — дизайнер выслушивает пользователя и разбирается в его проблеме, таким образом позволяя ему высказаться
- Забота — дизайнер должен задумываться о влиянии его проектов на окружающий мир
- Мастерство — дизайнер создаёт порядок из творческого хаоса и несёт ответственность за стандарты, которыми будут пользоваться другие
- Единство — важно выходить за пределы своей точки зрения и сравнивать её с другими взглядами

Коррекции и утверждение

Техническое задание

1. *Общие требования к Программному продукту*

1.1. Программный продукт (далее – Чатбот) разрабатывается для цели интерактивного online-консультирования пользователей, обращающихся в чат «Название системы».

1.2. Чатбот должен в автоматизированном режиме обрабатывать текстовые запросы пользователей, сформулированные на русском языке и поступающие в систему «Названис системы», и предоставлять ответы в текстовом виде. Требования к базе знаний Чатбота указаны в п.2.1. настоящего Технического задания.

1.3. Чатбот должен быть проинтегрирован с системой «Название системы» в объеме, указанном в п. 2.2.1. настоящего Технического задания.

1.4. Чатбот должен обрабатывать обращения из всех каналов, указанных в п. 2.3.1. настоящего Технического задания.

2. *Требования к параметрам Программного продукта*

2.1. Требования к базе знаний

2.1.1. База знаний Чатбота разрабатывается на основании информации и материалов, предоставленных Заказчиком (список ключевых вопросов и ответов, с разбивкой по тематикам, ТЗ). Обучение Чатбота проводится на паттернах и с применением алгоритмов машинного обучения. Формирование базы знаний проводит Исполнитель на основании предоставленных Заказчиком материалов.

2.1.2. Примеры вопросов по темам и сценарии для их обработки, которые должны быть включены в содержание базы знаний, обозначены в таблице «Таблица №1. Вопросы-ответы по темам».

Таблица №1 Вопросы-ответы по темам

Тема	**Примеры вопросов**	**Ответ**
Приветствие	Привет/ здравствуйте/ добрый день/ доброй ночи/	Добрый день. Я ваш виртуальный помощник. Чем я могу вам помочь?

4/7
СБОРКА
ДИАЛОГОВОЙ СИСТЕМЫ

Цель модуля

Изучить термины и инструменты применяемые на этапе сборки диалоговой системы

Этапы

сборки

- Инженерный анализ
- Разработка модели, метода решений и алгоритма создания кода
- Написание кода
- Отладка
- Создание рабочей документации

Инженерный анализ

Критерии качества

требований

- Корректность
- Недвусмысленность (однозначность)
- Полнота
- Непротиворечивость (совместимость)
- Упорядоченность (ранжированность по важности и стабильности)
- Проверяемость (тестируемость или тестопригодность)
- Трассируемость (прослеживаемость)

Разработка модели

Критерии качества

кода

- Тестируемость
- Читаемость
- Адаптивность
- Модифицируемость
- Устойчивость
- Социальность

Тестируемость — код обеспечивает лёгкое покрытие тестами и отладку. Unit тестирование — это процесс тестирования модулей, то есть функций и классов, являющихся частью программы. Создавая программу, разработчик должен учитывать возможности тестирования с самого начала работы над написанием кода.

Читаемость — легкое восприятие кода и использования программы. Этому способствуют логичное именование, хороший стиль интерфейса и реализации, точные и однозначные комментарии.

Адаптивность — лёгкость сопровождения. Продуманная и реализованная структура программы позволяет решать вопросы, связанные с работой программы на новом аппаратном обеспечении или новой платформе.

Модифицируемость — упрощает процесс внесения дальнейших изменений. Чем лучше оптимизирована структура, тем проще изменять код, добавлять новые свойства, повышать быстродействие и изменять архитектуру.

Устойчивость — при внесении изменений или возможных неполадках можно легко внести исправления. А правильная обработка ошибок значительно облегчает эксплуатацию программного продукта.

Социальность — возможность поддержки проекта несколькими разработчиками или целыми сообществами (особенно важно для проектов с открытым исходным кодом).

Покупаем, партеримся, строим

Не старайтесь всё

делать самостоятельно, часто выгодней

- найти и купить готовые решения, как часть вашего продукта
- разработать самостоятельно только уникальные элементы
- договориться о партнерстве с важными для проекта людьми, группами или компаниями

Написание кода

Примеры инструментов
разработчика

- Редактор исходного кода
- Система управления задачами
- Система контроля версий

Редактор исходного кода — текстовый редактор для создания и редактирования исходного кода программ. Он может быть отдельным приложением или встроен в интегрированную среду разработки.

Редакторы исходного кода имеют некоторые возможности, упрощающие и ускоряющие написание и изменение кода, такие как: подсветка синтаксиса, автодополнение, отступы, проверка правильности расстановки скобок, контекстная помощь по коду.

Система управления задачами — это набор инструментов постановки задач и контроля над их выполнением.

Далеко не каждая система управления задачами повышает эффективность работы. Основные проблемы по коммуникациям, оперативности и контролю остаются

Система контроля версий — это система, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться позже к определенной версии.

Отладка

Отладка — этап разработки компьютерной программы, на котором обнаруживают, локализуют и устраняют ошибки.

После саморедактуры, когда разработчик сам уже не видит ошибок — можно переходить к этапу полноценного тестирования

Рабочая документация

Полная и понятная документация — упрощает работы на этапе тестирования и снижает риск выпуска некачественного программного продукта

5/7
ТЕСТИРОВАНИЕ
ДИАЛОГОВОЙ СИСТЕМЫ

Цель модуля

Изучить термины и инструменты применяемые на этапе тестирования диалоговой системы

Этапы

тестирования

- Анализ документации
- Сценарий тестирования
- Тест-кейсы
- Создание документации
- Утверждение релиза

Анализ документации

Анализ документации — позволяет разработать модели и алгоритмы тестирования

Сценарий тестирования

Этапы создания сценария

тестирования

- Цели тестирования
- Методы тестирования
- Метрики тестирования
- Триггеры достижения целей тестирования
- Тест-кейсы

Тест-кейсы

Тест-кейс — это набор условий и/или переменных, с помощью которых тестировщик будет определять насколько полно тестируемое приложение удовлетворяет предъявляемому к нему требованию.

Для того, чтобы убедиться, что требование полностью удовлетворяется, может понадобиться несколько тест-кейсов.

Для полного тестирования всех требований, предъявляемых к приложению, должен быть создан/выполнен по меньшей мере один тест-кейс для каждого требования.

Создание документации

Документация на этапе тестирования — служит основой для принятия решений об утверждении релизов.

Утверждение релиза

6/7
СОПРОВОЖДЕНИЕ
ДИАЛОГОВОЙ СИСТЕМЫ

Цель модуля

Изучить термины и инструменты применяемые на этапе сопровождения диалоговой системы

Этапы

сопровождения

- Аналитические метрики
- Сценарии и бизнес-процессы
- Анализ и обработка данных
- Исправление ошибок и модификация системы
- Управление релизами

Аналитические метрики

Метрика — это качественный или количественный показатель, который отражает ту или иную характеристику и уровень качества продукта.

Сценарий и бизнес-процесс

Бизнес-процесс — это совокупность взаимосвязанных мероприятий или работ, направленных на создание определённого продукта или услуги для потребителей.

Анализ и обработка данных

Data Driven (дословно — "управляемый данными") — это подход к управлению, основанный на данных. Его главный постулат: решения нужно принимать, опираясь на анализ цифр, а не интуицию и личный опыт.

Исправление ошибок и модификация системы

Модификация — это процесс добавления новых функциональных возможностей или изменения существующих свойств системы.

Управление релизами

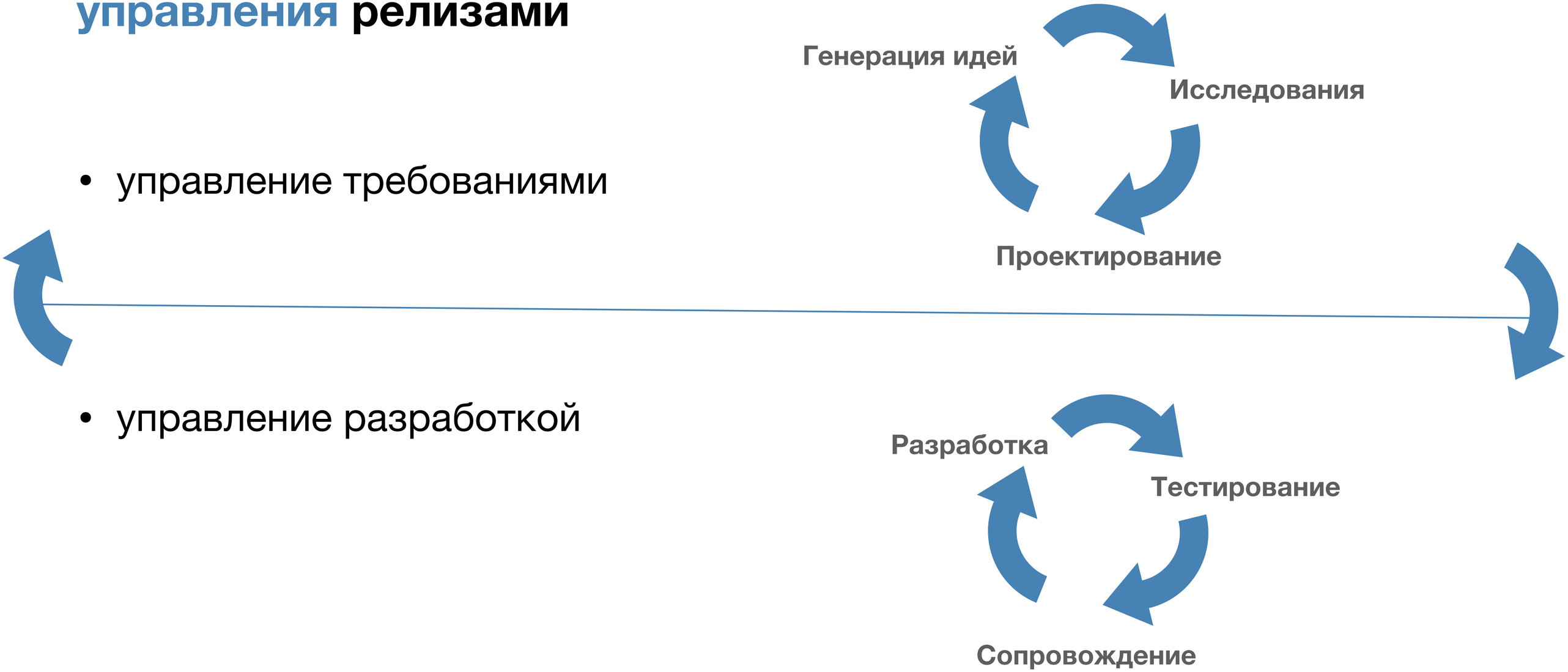
Релиз — это набор новых и/или измененных подсистем и элементов, в отношении которых осуществлено тестирование и которые утверждены для использования в качестве единой системы.

Релиз — это выпуск в свет продукта, дополнений или версий.

Минимальный уровень

управления релизами

- управление требованиями



- управление разработкой

7/7
ТИПИЧНЫЕ
ОШИБКИ

Цель модуля

Изучить типичные ошибки на всех этапах жизненного цикла диалоговой системы

Человек, появляясь на свет, мягок и податлив, а когда умирает - негибок и тверд.

Все живые твари, деревья и травы, когда рождаются, податливы и нежны, а когда умирают, становятся сухимии ломкими.

И потому тот, кто мягок и податлив, идет дорогой жизни, тот, кто негибок и тверд, идет дорогой смерти.

Вот почему воин, неустрашимо рвущийся в бой, найдет свою гибель.

Дерево, высокое и крепкое, найдет свой топор.

То, что идет вниз - это твердое и крепкое, то, что растет вверх - это мягкое и податливое.

Лао Цзы. Дао Де Дзин.

Ошибки

при создании диалоговых систем

- Упертость
- Плохие коммуникации
- Расточительство
- Иллюзии
- Страх

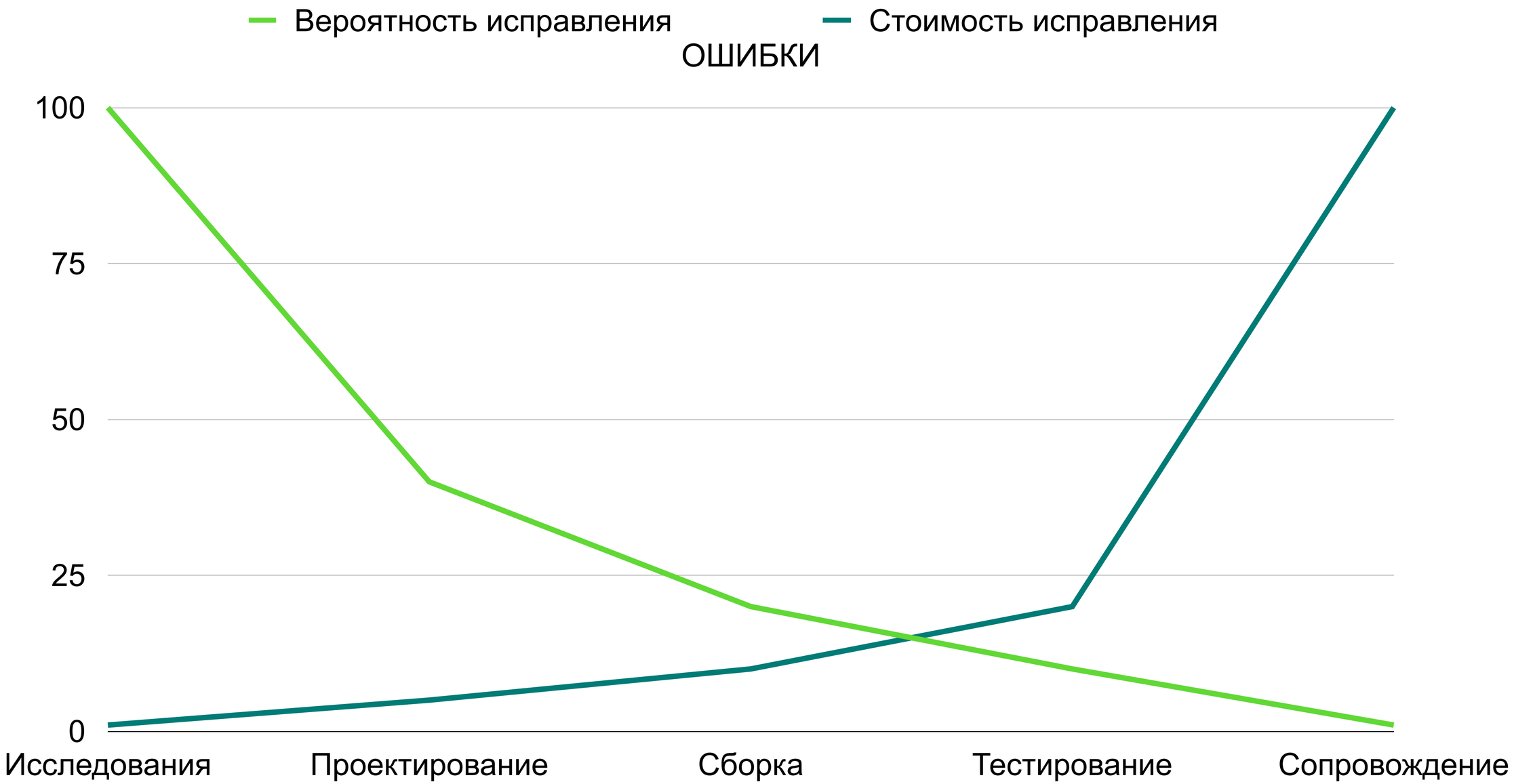
Упертость – это потребность стоять на своем мнении, не соглашаться с кем-либо, упрямо и ограниченно сосредоточиваться на одном действии на грани тупизма, при полном отсутствии гибкости, уступчивости, компромиссности, выгодности.

Плохие коммуникации – это закрытая структура коммуникаций, неблагоприятный психологический климат в коллективе, особенности искаженного восприятия, неполнота воспринимаемой информации, **плохая** структура сообщений, слабая память, отсутствие обратной связи.

Расточительство – это неразумное расходование чего-нибудь, например материальных ресурсов, времени, душевных сил, запасов и так далее.

Иллюзии — это ментальные заблуждения, блокирующие осознанность и нарушающие связь с реальным миром.

Страх — это эмоция, которая возникает в ситуации угрозы биологическому или социальному существованию человека. В сочетании с иллюзиями и ментальными искажениями может оказывать значительное негативное влияние.


Вероятность исправления
Стоимость исправления
ОШИБКИ
100
75
50
25
0
Исследования
Проектирование
Сборка
Тестирование
Сопровождение

"Семь раз отмерь, один раз отрежь"

Народная мудрость

Ошибки

на этапе исследований

- собирать сырые данные
- данные из одного источника
- ранняя обратная связь без более поздней
- быстро реагировать на требования
- больше времени уделять существующим клиентам
- отдавать приоритет краткосрочным целям

Ошибки

на этапе проектирования

- страх принятия решений
- идеи в голове, отсутствие фиксации
- отсутствие объединяющей идеи
- отсутствие структуры
- потеря связи с реальностью

Ошибки

на этапе сборки

- низкая вовлеченность
- ограниченная ответственность
- непрозрачные решения
- низкое качество кода
- отсутствие циклов отладки

Ошибки

на этапе тестирования

- спонтанное тестирование
- тест-кейсы не по требованиям
- тест-кейсы повторяют требования, а не проверяют их
- подгонка приёмочных тест-кейсов
- не своевременность данных

Ошибки

на этапе сопровождения

- отсутствие скриптов и бизнес-процессов
- отсутствие данных обо всем что может отвалиться
- отсутствие дублирующих каналов
- быстрая реакция и отсутствие реакции
- отсутствие перспективных исследований

ЗАДАНИЕ

САМОСТОЯТЕЛЬНАЯ РАБОТА

Цель задания

Анализ и последующее сопоставления — входящего уровня навыков слушателя и навыков после окончания курса

Этапы

выполнения самостоятельной работы

- Сформулируйте идею простейшей диалоговой системы, выполняющей не более одной функции
- Выполните работы по каждому из этапов жизненного цикла задуманной вами диалоговой системы
- Создайте словарь терминов использованных в лекции и не понятных лично вам
- Создайте журнал описанных в лекции ошибок, которые вам удастся обнаружить при работе в рамках жизненного цикла вашего продукта

Результаты

выполнения самостоятельной работы

- Созданная диалоговая система (ссылка для доступа)
- Документация отражающая весь жизненный цикл диалоговой системы (ссылка для доступа и изучения)
- Словарь непонятных вам терминов использованных в этой лекции (ссылка для доступа и изучения)
- Журнал собственных ошибок (ссылка для доступа и изучения)

Инструменты

для выполнения самостоятельной работы

- Публикация диалоговой системы https://t.me/BotFather
- Рабочее пространство для создания документации https://www.notion.so
- Рабочее пространство для создания документации https://coda.grsm.io/open (при регистрации в системе по данной ссылке получите в подарок на свой счет 10$, которые можно использовать если в дальнейшем функций бесплатного тарифа вам будет мало)
- Вы можете использовать любые альтернативные инструменты, при соблюдении требования о наличии ссылки для доступа и изучения результатов самостоятельной работы

СПАСИБО!