



НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ
УНИВЕРСИТЕТ «СИНЕРГИЯ»

Институт Информационных технологий

(наименование факультета/ института)

Направление подготовки /специальность: 09.04.03 Прикладная информатика

(код и наименование направления подготовки /специальности)

Профиль/специализация: Управление информационными технологиями в организации

(наименование профиля/специализации)

Форма обучения: очная

(очная, очно-заочная, заочная)

УТВЕРЖДАЮ

Директор Института
Информационных технологий

Захаров А.В.
(ФИО)

(Подпись)

« ____ » _____ 202__ г.

**ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ
НА ПРОИЗВОДСТВЕННУЮ ПРАКТИКУ**

(вид практики)

Научно-исследовательская работа

(тип практики)

1 семестр

обучающегося группы .

(Шифр и № группы) (ФИО обучающегося)

Место прохождения практики:

Университет Синергия

(наименование структурного подразделения Образовательной организации)

Срок прохождения практики: с « 29 » сентября 2022 г. по « 26 » октября 2022 г.

Содержание индивидуального задания на практику:

№ п/п	Виды работ
1.	Инструктаж по соблюдению правил противопожарной безопасности, правил охраны труда, техники безопасности, санитарно-эпидемиологических правил и гигиенических нормативов.
2.	Выполнение определенных практических кейс-задач, необходимых для оценки знаний, умений, навыков и (или) опыта деятельности по итогам производственной практики (научно-исследовательской работы).
	Описание темы исследования¹. Управление проектами включает в себя множество подходов и инструментов. Управление проектами происходит комплексно по этапам жизненного цикла проекта и фазам менеджмента. Области проектного управления включают финансовые, кадровые, структурные сферы деятельности организации. А для того, чтобы все эти сферы были широко и правильно охвачены, команде, работающей над проектом, необходимо правильно распределить свои обязанности, чему и способствует методология гибкого управления проектами Agile. Agile-методология основывается, в первую очередь, на визуальном контроле. Чаще всего участники проекта, работая над достижением результата, пользуются специальными цветными карточками. Визуальный контроль позволяет команде иметь наглядное представление о текущем состоянии процесса и гарантирует одинаковое видение проекта всеми ее членами. Члены команды и клиент в большинстве случаев работают вместе и рядом. Благодаря этому существенно ускоряются многие рабочие процессы, которые связаны с информированием участников проекта. Кроме того, совместная работа способствует созданию здоровой атмосферы для плодотворного и эффективного сотрудничества и скорейшего достижения результатов. Когда руководитель проекта, команда и клиент действуют сообща, исключается опасность недопонимания целей и утери информации. Все рабочие процессы становятся максимально прозрачными, а это значит, что любые возникающие проблемы можно разрешать практически моментально и находить лучшие варианты их решения. Особое внимание нужно уделить руководителю проекта. Руководитель здесь выступает скорее в роли лидера, который задает направление и определяет правила сотрудничества и работы. Другими словами, Agile-управление является адаптируемым.
2.1.	Кейс-задача № 1. Ознакомится с научной темой исследования. Изучить научные источники по данной теме, включая научные статьи в наукометрических базах РИНЦ, Scopus, WebofScience, источники интернета, разработки компаний в области гибкого управления проектами Agile. Сформировать список научных источников, необходимых для исследования. Сформулировать тему научного исследования, согласовать тему с руководителем.
2.2.	Кейс-задача № 2. Осуществить сбор материалов по теме научного исследования, систематизировать собранный материал. Провести анализ обработанной информации. Выявить проблемы, требующие решения.
2.3.	Кейс-задача № 3. Изучить физическую сущность проблемы. Сформировать варианты решения. Выбрать и обосновать модель решения. Провести анализ модели и полученных решений.
2.4.	Кейс-задача № 4. Обосновать выбор необходимых средств для проведения экспериментальных исследований, применяемых методик. Провести апробирование полученных результатов. Обработать полученные результаты.
2.5.	Кейс-задача № 5. Сопоставить результаты экспериментов с теоретическими данными, проанализировать выявленные расхождения. Сформулировать научные выводы. Составить научный отчет по результатам исследования.
3.	Систематизация собранного нормативного и фактического материала.
4.	Оформление отчета о прохождении практики.
5.	Защита отчета по практике.

Разработано руководителем практики от Университета Терехова Л.А.

(ФИО) (Подпись)

¹Представлено типовое описание исследования и типовые кейс-задачи. Темы НИР и кейс-задачи определяются руководителем практики.

« 27 » сентября 2022 г.



**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ
«СИНЕРГИЯ»**

Институт Информационных технологий
(наименование факультета/ института)

Направление подготовки /специальность: 09.04.03 Прикладная информатика
(код и наименование направления подготовки /специальности)

Профиль/специализация: Управление информационными технологиями в организации
(наименование профиля/специализации)

Форма обучения: очная

ОТЧЕТ

ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ
(вид практики)

Научно-исследовательская работа
(тип практики)

1 семестр

Обучающийся

_____ (ФИО)

_____ (подпись)

Москва 20г.

Практические кейсы-задачи, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности по итогам практики

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
Кейс-задача № 1	<p>Сформулирована и согласована тема НИР: «Управление бизнес-процессами и проектами на основе технологии Agile, сравнение kanban, scrum, scrumban и внедрение одной из технологий на конкретном предприятии (Проектное Бюро Город)»</p> <p>Сформирован список научных источников, необходимых для исследования:</p> <ol style="list-style-type: none"> 1. Павленко А. Agile и Scrum разница. [Электронный ресурс] URL: https://SCRUMmasters.com.ua/blog/SCRUM-and-agile (время обращения 06.01.2023). 2. Вольфсон Б. Гибкое управление проектами и продуктами. СПб.: Питер, 2020. – 144 с. – с. 19. 3. Кен Швабер Софт за 30 дней. Как Scrum делает невозможное возможным / Джефф Сазерленд, Кен Швабер, – Изд-во: Манн, Иванов и Фербер, 2021 – 256 с. 4. Джефф Сазерленд Scrum. Революционный метод управления проектами / Джефф Сазерленд, – Изд-во: Манн, Иванов и Фербер, 2020 – 272 с. 5. Майк Кон Scrum. Гибкая разработка ПО / Майк Кон, – Изд-во: Вильямс, 2020 – 576 с. 6. Кеннет С. Рубин Основы Scrum. Практическое руководство по гибкой разработке ПО / Кеннет С. Рубин, – Изд-во: Вильямс, 2021 – 544 с. 7. Хенрик Книберг Scrum и XP: заметки с передовой / Хенрик Книберг, – Изд-во: Инфра-М, 2020 – 94 с. 8. Арефьев А. Все что нужно знать про Scrum. [Электронный ресурс]. URL: http://www.alexouncil.com/SCRUM/ (время обращения 06.01.2023). 9. Багинский Д. Scrum методология: современные задачи и решения. [Электронный ресурс]. URL: https://www.kom-dir.ru/article/SCRUM-metodologiya (время обращения 06.01.2023). 10. Сухоруков А. Методы управления проектами: Scrum vs Kanban. [Электронный ресурс]. URL: http://suhorukov.com/news_akademy/metody-upravleniya-proektamiSCRUM-vs-kanban (время обращения 06.01.2023).
Кейс-задача № 2	<p>Управление проектами включает в себя множество подходов и инструментов. Управление проектами происходит комплексно по этапам жизненного цикла проекта и фазам менеджмента.</p> <p>Области проектного управления включают финансовые, кадровые, структурные сферы деятельности организации.</p> <p>А для того, чтобы все эти сферы были широко и правильно охвачены, команде, работающей над проектом, необходимо правильно распределить свои обязанности, чему и способствует методология гибкого управления</p>

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>проектами Agile</p> <p>Agile-методология основывается, в первую очередь, на визуальном контроле. Чаще всего участники проекта, работая над достижением результата, пользуются специальными цветными карточками. Один цвет сигнализирует о завершении планирования какого-то элемента конечного продукта, другой – о завершении его разработки, третий – о готовности и т.п.</p> <p>Визуальный контроль позволяет команде иметь наглядное представление о текущем состоянии процесса и гарантирует одинаковое видение проекта всеми ее членами.</p> <p>Члены команды и клиент в большинстве случаев работают вместе и рядом. Благодаря этому существенно ускоряются многие рабочие процессы, которые связаны с информированием участников проекта.</p> <p>Кроме того, совместная работа способствует созданию здоровой атмосферы для плодотворного и эффективного сотрудничества и скорейшего достижения результатов.</p> <p>Когда руководитель проекта, команда и клиент действуют сообща, исключается опасность недопонимания целей и утери информации.</p> <p>Все рабочие процессы становятся максимально прозрачными, а это значит, что любые возникающие проблемы можно разрешать практически моментально и находить лучшие варианты их решения.</p> <p>Особое внимание нужно уделить руководителю проекта. Его нельзя назвать человеком, раздающим указания налево и направо.</p> <p>Руководитель здесь выступает скорее в роли лидера, который задает направление и определяет правила сотрудничества и работы. Другими словами, Agile-управление является адаптируемым.</p> <p>Еще одним важным моментом Agile-методологии является разделение всего объема проекта на несколько более мелких составных частей.</p> <p>Такой подход многократно упрощает процесс разработки, а отдельные группы команды могут фокусироваться каждая на своей конкретной задаче.</p> <p>12 принципов Agile.</p> <ol style="list-style-type: none"> 1. Главное в работе – удовольствие заказчика. 2. Изменение требований на любой стадии работы – нормальное явление. 3. Обновленные версии продукта должны выходить регулярно, чем чаще – тем лучше. 4. Разработчики и заказчики должны контактировать как можно чаще.

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>5. Мотивация команды – предельно важна.</p> <p>6. Личное общение должно происходить постоянно, хотя бы посредством видеосвязи.</p> <p>7. Главный показатель прогресса – продукт.</p> <p>8. Очень важно поддерживать стабильный темп работы.</p> <p>9. Каждый новый проект – шанс усовершенствовать процесс разработки.</p> <p>10. Лишнюю работу нужно сокращать.</p> <p>11. Если от микроменеджмента можно отказаться – сделайте это.</p> <p>12. Необходимо регулярно анализировать, как можно улучшить качество работы. Полезные новшества следует скорее тестировать на практике.</p> <p>Таким образом, работая над одним циклом, участники проекта овладевают новыми навыками и получают новые знания, а также анализируют допущенные в процессе ошибки.</p> <p>Все это сводит вероятность совершения подобных ошибок в будущем (в следующих циклах и других проектах) практически к нулю. И, наконец, последний значимый элемент подхода – это спринты и ежедневные встречи.</p> <p>Спринтами называются ограниченные конкретными сроками (дедлайнами) отрезки времени, в течение которых команда успевает выполнить определенные задачи. Именно благодаря спринтам команда может видеть результаты своих действий.</p> <p>Сегодня большинство компаний используют методологию Agile при разработке программного обеспечения. У Agile есть разные системы с похожими принципами и конечными результатами, но они различаются по структуре и подходу к управлению.</p>
Кейс-задача № 3	<p>Среди всех методов системы Agile Scrum отличается тем, что делает основной упор на качественный контроль рабочего процесса.</p> <p>Впервые описавшие его японские специалисты по стратегическому менеджменту Хиротака Такуэти и профессор в области научно-технических знаний Икуджиро Нонака называют метод «подходом в регби», где Scrum является «борьбой за мяч».</p> <p>Метод заключается в том, что разработка проекта разделяется на спринты, по окончании которых клиент получает улучшенное ПО. Спринты строго фиксируются по времени, и могут длиться от 2 до 4 недель. Рабочий процесс в одном спринте включает в себя несколько стадий:</p> <ul style="list-style-type: none"> • Определяются объемы работы • Каждый день проводятся 15-минутные встречи, чтобы члены

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>команды могли скорректировать свою работу и подвести промежуточные итоги</p> <ul style="list-style-type: none"> • Демонстрируются полученные результаты • Спринты обсуждаются для поиска удачных и неудачных решений и действий <p>В большинстве случаев Скрам применяется в работе со сложным ПО и для разработки продукта с использованием инкрементных и итеративных методов.</p> <p>Благодаря ему серьезно повышается производительность команды и сокращаются временные затраты на достижение цели.</p> <p>Scrum улучшает результаты, помогает адаптировать проект к изменениям, обеспечивает более точную оценку при меньших трудозатратах на анализ и позволяет эффективнее контролировать этапы работы и сценарий проекта. Все это как нельзя лучше соответствует бизнес-целям.</p> <p>Kanban – еще один метод, делающий командную работу более результативной и продуктивной. Смысл его сводится к приданию процессу разработки максимальной прозрачности и равномерному распределению нагрузки среди участников проекта.</p> <p>Важная особенность Kanban еще и в том, что он мотивирует людей на постоянное сотрудничество, совершенствование и обучение.</p> <p>Работа по методу Kanban выстраивается на нескольких принципах. Во-первых, вся информация о проекте должна быть визуализирована, что позволяет видеть накладки, ошибки и недочеты и активно их устранять.</p> <p>Во-вторых, работа над одной задачей должна вестись одновременно всей командой – это помогает сбалансировать усилия и получаемые результаты, исключает неравномерное распределение нагрузки.</p> <p>И, в-третьих, время на выполнение всех задач строго контролируется, благодаря чему оптимизируется процесс и экономится время.</p> <p>В отличие от Скрам, Канбан обрел популярность намного позже, но это ни в коей мере не умаляет его достоинств и не делает менее эффективным. Метод полезен как в IT-области, так и в бизнес-сфере.</p> <p>Это лишь примеры основных методов управления проектами, основанных на Agile. Но не стоит пренебрегать и другими методами, такими как PRINCE2, Lean, Six Sigma, XP, CCPM, ECM, Waterfall и другие. К тому же у Аджайл, наряду с преимуществами, есть и некоторые недостатки.</p> <p>Постигая Agile, важно знать как о положительных, так и об отрицательных сторонах этой методологии. Начнем с плюсов.</p> <p>В первую очередь стоит отметить, что Agile-управление очень гибкое.</p>

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>Если, например, традиционная методология указывает на конкретные этапы работы, то Эджайл легко подстраивается под потребителя конечного продукта и требования заказчика.</p> <p>Собственно, и в конечном продукте число дефектов минимизируется, ведь он является результатом тщательной проверки качества, которая проводится по завершении каждого этапа-спринта.</p> <p>Кроме того, Agile быстро запускается, легко реагирует на изменения, позволяет команде разработчиков и клиентов поддерживать постоянную связь в реальном времени. Преимущества очевидны, но давайте поговорим и о минусах.</p> <p>Недостатки методологии состоят в том, что, во-первых, постоянная обратная связь может приводить к тому, что все время будет переноситься и дедлайн проекта, тем самым создавая угрозу бесконечно продолжающейся работы.</p> <p>Если заказчик видит, например, только результаты, но не имеет представления об усилиях, потребовавшихся для их достижения, он будет все время требовать улучшений.</p> <p>Второй недостаток заключается в необходимости адаптировать под изменяющиеся условия проекта проектную документацию.</p> <p>При отсутствии надлежащего информирования команды об изменениях или дополнительных функциях документы с функциональными требованиями или архитектурой могут оказаться неактуальными на текущий момент времени.</p> <p>Третьим существенным минусом Аджайл можно назвать необходимость в частых встречах. Они, конечно, способствуют повышению эффективности работы, но все же постоянное отвлечение членов команды может сказаться на процессе отрицательно, ведь внимание людей систематически уходит в сторону от решаемых задач.</p> <p>Сюда же можно отнести такие вещи как необходимость в постоянном присутствии клиента, невозможность выстраивать долгосрочные планы и потребность в мотивированных и высококвалифицированных специалистах.</p> <p>Кстати, последнее в огромной степени касается и внедрения Agile-управления в деятельность организации. И, постигая Agile, с темой ее внедрения тоже нужно познакомиться.</p> <p>Scrum хорошо подходит в условиях неопределенности. Когда у вас или заказчика есть только гипотезы, и вы еще не знаете верны они или нет. Акцент на скорости, метриках продукта и обратной связи от пользователей.</p>

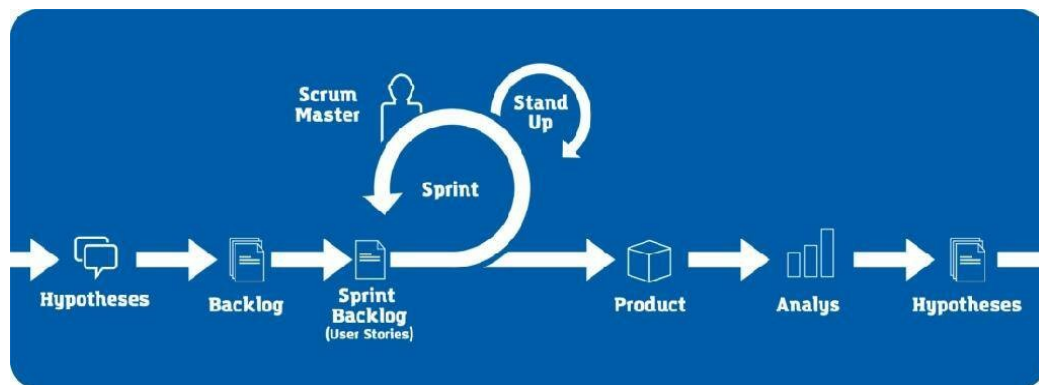


Рисунок 3.1 Scrum (Sprint) Aydar Sultanbek

Задача сформировать гипотезы в виде User Stories (US) и как можно быстрее запустить спринт. Процесс примерно такой:

- оценка US и запуск спринта;
- разбивка US на задачи;
- ежедневные встречи (Stand up);
- фиксирование скорости команды (в Story Points);
- уточнение бэклога продукта;
- релиз (выпуск обновленного продукта);
- обзор спринта;
- ретроспектива.

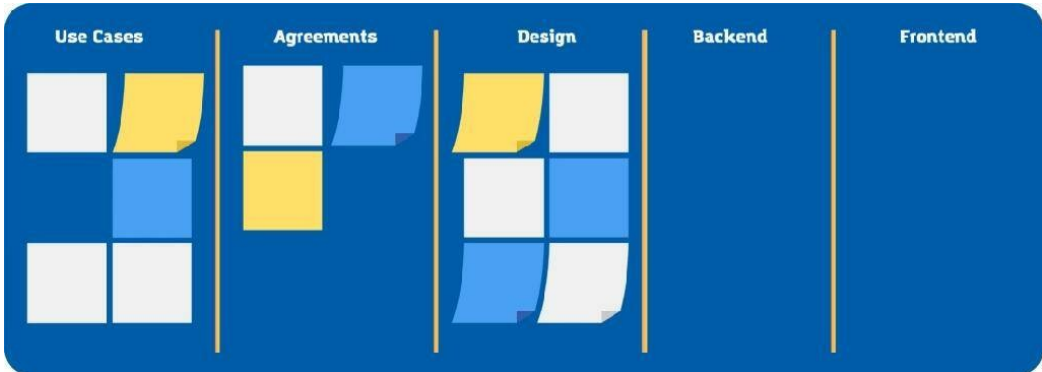
Мы не тратим время на блок-схемы и на согласования, но тратим время на ресурсоемкие ритуалы Scrum.

Именно они помогают сырую US как-то оценить, вовремя подправить и быстро обновить на проде, что позволяет своевременно собрать метрики, и сделать выводы.

Да, задачи не согласованы и не проработаны так тщательно, как, например, в Kanban. Но мы потратили всего 2 недели. Даже если ошиблись с гипотезой (что часто бывает), мы не растягивали процесс во времени и обошлись ресурсами небольшой команды (Product Owner, Scrum Master, Dev team).

Работа по Scrum это бесконечный процесс формирования и валидирования гипотез. Без настройки продуктовых метрик и без интервью с пользователями не обойтись. Чем быстрее и качественнее мы будем собирать данные, тем быстрее и правильнее будут наши решения. Тем лучше это будет сказываться на продукте.

Kanban, на мой взгляд, лучше подходит когда задачи расписаны на месяцы и каждый шаг необходимо согласовывать с заказчиком. Акцент на

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>согласованном процессе, тестировании и качественном релизе.</p> <p>Один цикл, от поступления задачи до ее выпуска на продуктиве, может затрагивать широкий круг специалистов из разных команд:</p> <ul style="list-style-type: none"> - анализ и разработка блок-схем; - согласование со всеми заинтересованными лицами (ЗЛ); - UX/UI дизайн; - backend разработка; - frontend разработка; - тестирование; - доработка; - релиз (выпуск обновленного продукта). <p>Мы растягиваем процесс, но в итоге имеем согласованный и качественный релиз.</p> <p>Подробная доска поможет вам и ЗЛ увидеть узкие места. Например: скопилось много задач по дизайну, и разработчики сидят без дела, скорей всего команде нужны еще дизайнеры.</p> <p>Установим WIP-лимиты (Work In Progress), например: больше одной задачи разработчикам в работу не брать. Это сделает процесс еще прозрачнее и разработчики не будут тратить время на переключение с одного таска на другой.</p>  <p style="text-align: center;">Рисунок 3.2 - Kanban доска Aydar Sultanbek</p> <p>В Kanban тоже есть свои ритуалы, но они не такие ресурсоемкие.</p> <p>Сбор метрик и обратная связь от пользователей важны, но не настолько критично как в Scrum.</p> <p>Заказчик может быть одновременно пользователем вашего продукта. Если весь цикл вы согласовывали с ним каждый шаг, будет странным если завтра он попросит все переделать.</p> <p>Процесс закрытия задач последовательный, согласованный и поэтому</p>

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>требуется больше времени.</p> <p>Можем параллельно запускать в работу несколько US, возвращать задачу на предыдущий этап и добавлять в любое время новую.</p> <p>Главное, постоянно настраивать и улучшать доску под себя и свои нужды, устраняя узкие места и обеспечивая непрерывный поток. Мы знаем кто, что и когда делает и видим все этапы конвейера.</p> <p>Осталось зафиксировать время цикла от появления задачи и до ее релиза. Это позволит следить за темпом, непрерывно его настраивать и улучшать.</p> <p>Канбан практически в любой команде и в любых условиях можно использовать. Но он менее требовательный и поэтому, на мой взгляд, меньше подходит для работы в условиях неопределенности.</p> <p>Отметим что его лучше использовать там, где известны задачи на месяцы вперед и без согласования со всеми заинтересованными лицами задачу двигать дальше нельзя.</p> <p>Проанализируем каждую конкретную тему Scrum и отличия от Канбана:</p> <ul style="list-style-type: none"> -В Scrum вы разделяете свою организацию на небольшие, кросс-функциональные, самоорганизующиеся команды. В Канбане нет жесткого требования про кросс-функциональные команды -В Scrum есть особые роли, а для Канбана подойдут обычные -В Scrum ежедневное стендап-собрание — это сердечный ритм проекта. В Канбане планерки проводить жестко не требуются -В Scrum вы должны разделить свою работу на список небольших конкретных результатов. В Канбане достаточно разделить работу на части, написать каждую часть на стикер и повесить его на стену. Они не должны быть результатами. --В Scrum вы должны разбить время на короткие итерации фиксированной длины (1-4 недели), с разными версиями минимально-работоспособного продукта (MVP), которые вы будете демонстрировать после каждой итерации. В Канбане же работа непрерывная, а не итеративная. -В Scrum вы должны отсортировать список поставляемых результатов по приоритету и оценить необходимые усилия для каждого элемента. В Канбане нет жесткого требования оценивать работу. -В Scrum, основываясь на информации, полученной на проверке релиза после каждой итерации, оптимизируется план выпуска и обновляются приоритеты вместе с заказчиком. В Канбане этого не происходит. -В Scrum у вас есть фиксированные события, такие как начало,

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>планирование, обзор и ретроспектива. В Канбане же есть задающие ритм каденции.</p> <p>-В Scrum рабочий процесс всегда должен оптимизироваться после каждой проведенной ретроспективы. В Канбане же это не обязательно, но можно провести в каденциях.</p> <p>-В Scrum должен быть список невыполненных работ по продукту и график сгорания задач, чего в Канбане нет и в помине.</p> <p>Итак, что же находится в Канбане? Давайте посмотрим на контраст со Scrum:</p> <p>-Канбан фокусируется на представлении рабочего процесса команды, давая им возможность визуализировать его и улучшить как можно скорее. Scrum имеет фиксированный процесс и церемонии.</p> <p>-Канбан позволяет использовать любые именованные столбцы в вашей доске, чтобы проиллюстрировать, где находится каждый элемент, продукт или услуга в рабочем процессе. Scrum фокусируется на результатах с конкретными столбцами: “бэклог”, “бэклог спринта”, “работа в процессе” и “выполненная работа”.</p> <p>-Канбан ограничивает “работу в процессе» WIP-лимитом. В Канбане необходимо установить ограничения на количество работ, которые могут выполняться в каждом столбце рабочего процесса. В Scrum нет никаких правил на этот счет.</p> <p>-Одна из самых важных вещей в Канбане — это измерение среднего времени выполнения одного элемента, называемое “временем цикла”. Это очень важно, потому что это дает вам возможность оптимизировать процесс, чтобы сделать работу как можно короче и предсказуемее.</p> <p>-В Канбан можно вносить изменения по мере необходимости. В Scrum изменения не должны прерывать спринт</p>
Кейс-задача № 4	<p>Таким образом, Agile — это подход к разработке большого проекта. Философия, которая позволяет создавать продукт с постоянно меняющимися требованиями.</p> <p>Scrum — это метод управления проектами, он входит в философию Agile. Ключевое отличие от классической, водопадной схемы создания ПО заметно сразу — для начала разработки не нужно техническое задание.</p> <p>Минус водопадной схемы - процессы идут друг за другом. Это замедляет разработку</p>



Рисунок 4.1 - Водопадная схема - процессы идут друг за другом.

Это замедляет разработку

Вместо проектного задания используется бэклог — список функций, требований к системе, желаний заказчика.


В Scrum они сортируются по приоритету. Это живой документ, добавляйте в него новые задачи по ходу работы.

Приоритет	Описание задачи
10000	Главная страница
9000	Список товаров
8000	Личный кабинет
8100	Оформление заказа
8200	Интеграция с системой оплаты
7000	Корзина
8500	Быстрый заказ
3300	Блог
2000	С этим товаром покупают
100	Форма рекомендаций

Рисунок 4.2 - Удобно вести бэклог задач в Trello или Excel.

Используем не привычный список 1, 2, 3, 4.

Попробуем четырехзначные цифры — так сможем просто добавить строку между ними и выставить подходящий приоритет.

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>Например, между 1 000 и 2 000 напишем 1 050.</p> <p>Не нужно прорабатывать и продумывать полностью все функции сразу. Все «хотелки» и то, что появляется в процессе, добавляются в бэклог.</p> <p>Решаем, что делать сразу, а что стоит отложить на следующую версию.</p> <p>Scrum создавался в первую очередь для гибкости и ускорения разработки. Для этого появилась механика спринтов — весь процесс делится на отрезки, обычно от одной до четырех недель.</p> <p>Команда забирает из бэклога часть задач. Каждая разбивается на максимально мелкие тикеты. Теперь нужно оценить время на задачу, и вот здесь проявляется особенность Scrum.</p> <p>Дело в том, что люди плохо считают процессы в абсолютных величинах. Сложно сказать, сколько часов что займет. Поэтому в Scrum используется относительная оценка.</p> <p>За основу берется простая функция, которую все оценивают одинаково — например, понятно, что ее сделают за час. Остальные тикеты вычисляются так — «это мы будем делать раз в пять дольше по времени».</p> <p>Сделаем список версий продукта — от ПО с минимумом функций до полностью реализованного.</p> <p>Укажем к каждой версии прогноз по сроку выполнения.</p>  <p style="text-align: center;">Рисунок 4.3 - Бэклог со спринтами.</p> <p>Ключевая идея - до тех пор, пока команда не забрала задачи на спринт, их можно бесконечно видоизменять в бэклоге.</p> <p>В разработку уходит согласованная часть. Каждый спринт — это небольшой релиз, в конце которого команда показывает работающую функцию ПО.</p> <p>Распределите роли в команде</p> <p>Product Owner</p> <p>Связующее звено между командой разработки и пользователями. Этот</p>

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>человек собирает общую концепцию продукта из мнений заказчиков и других заинтересованных в выпуске ПО людей. Он формирует задачи и расставляет приоритеты.</p> <p>Scrum Master</p> <p>Член команды разработки, отвечающий за выполнение ежедневных процедур и за соблюдение интересов команды.</p> <p>Этот человек фиксирует дедлайны и начало спринта, добавляет оценки, отчитывается перед заинтересованными лицами об этапах проекта. Растите scrum-мастера внутри команды.</p> <p>Команда разработки</p> <p>Люди, которые непосредственно создают и тестируют код.</p> <p>К разработчикам есть несколько требований:</p> <ul style="list-style-type: none"> • Как минимум один человек в команде должен понимать код, который написали остальные. Тот, кто лучше всех разбирается в теме проекта, становится куратором. • Все совместно владеют кодом, понимают, как работает продукт. • Команда стабильная и постоянная. • Аналитики, дизайнеры — опционально, достаточно приглашать на отдельные тикеты. • Scrum на удаленной работе возможен, но придется трудиться над эффектом присутствия. <p>У такого принципа формирования команды есть минус — сложно заменить неожиданно выпавшего человека. Но скорость разработки на практике все равно выше, чем у других подходов.</p> <p>Контролируем процессы</p> <p>Диаграмма сгорания — это наглядная демонстрация того, как команда «переваривает» все задачи проекта.</p> <p>Красная линия — план. Синяя — то, что делает команда.</p> <p>Диаграмма обновляется каждый день. Вы сразу видите, когда есть отклонения от плана: можно спокойно «крутить гайки» или менять приоритеты в бэклоге.</p>

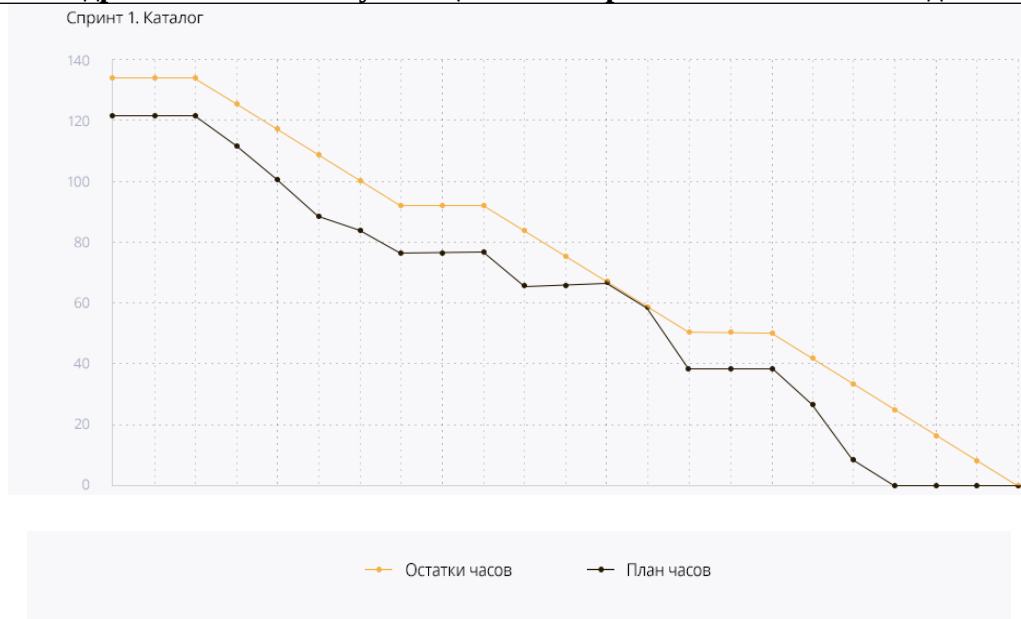


Рисунок 4.4 - Диаграмма сгорания в реальных проектах.

Контролируем работу команды с помощью двух scrum-показателей:

- Focus Factor - коэффициент, который показывает, сколько задача должна была выполняться по плану, а сколько вышло в итоге.

Так оценивается «концентрация» команды над проектом.

- Velocity - производительность. Поможет спрогнозировать количество задач, которые команда сможет взять в следующем спринте — в зависимости от количества готовых тикетов в прошлом. $Velocity = Focus\ Factor * \text{Оценка новых задач}$.

Строим график планового и фактического расхода времени на задачи.

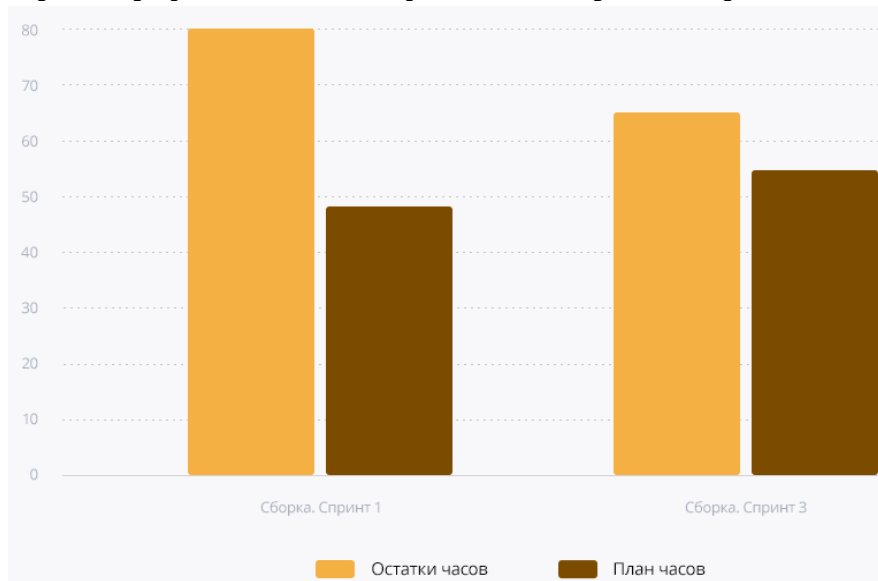


Рисунок 4.5 - График планового и фактического расхода времени на задачи.

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>Через несколько спринтов столбцы станут примерно одинаковыми.</p> <p>Организуем работу команды</p> <p>В Scrum от сотрудников требуется минимальная отчетность. Каждый день человек должен ответить на три вопроса:</p> <ul style="list-style-type: none"> • Что сделано вчера? • Что будет сделано сегодня? • Какие есть проблемы и препятствия для выполнения задач? <p>Задача руководителя — выявить и устранить трудности, которые мешают разработчику добиться прогнозируемого результата.</p> <p>Для сотрудников это три-пять минут — ответили на вопросы, поставили оценки, разбежались работать дальше. Никаких решений или дискуссий.</p> <p>В конце каждого спринта проводится ретроспектива. Команда встречается, озвучивает мнение, что в отрезке было хорошо, что плохо</p> <p>Спросите у сотрудников идеи — что поможет им работать быстрее и эффективнее, что исправит проблемы.</p> <p>Запишем их в отдельный план — забирайте туда только те идеи, которые возможно сделать за следующий спринт.</p> <p>На следующей ретроспективе обсудите идеи из плана, отсортируйте их по категориям «плохо» и «хорошо». Повторите процесс — получается ретроспектива на ретроспективу.</p> <p>Формируйте организацию процесса постепенно. Разбиваем день — например, шесть часов люди работают по спринтам, два часа остаются на срочные и случайные моменты.</p> <p>Если все пойдет без неожиданностей, ничего страшного, продолжайте спринт, сделайте больше тикетов.</p> <p>Первый спринт команда всегда «факапит», потому что слишком оптимистично смотрит на дедлайны и задачи.</p> <p>Второй — берет очень мало задач и делает больше.</p> <p>Третий — снова плохая оценка, но уже чуточку лучше. Потом все выравнивается. Это рабочий процесс.</p> <p>Демонстрируем проект. Не затягиваем с первой версией продукта. Демонстрацию лучше проводить после каждого спринта — пусть даже релиз не пойдет к пользователям.</p> <p>Не копируем внутри команды много функций — покажите их заинтересованным лицам и получите обратную связь. После — сразу измените бэклог.</p>

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>В этом основное преимущество Scrum — гибко менять список задач во время разработки, не делать лишнего и не получать тысячи правок после завершения проекта, как в каскадной методологии разработки.</p> <p>Таким образом, работать по системе можно даже на бумаге. Отлично подходит и таблица в Google Docs.</p> <p>Создаем свою рабочую область вручную или попробуем специальные сервисы:</p> <ul style="list-style-type: none"> - Trello — подходит для маленьких проектов, быстро и удобно. - Scrumban — есть разные доски, вложенные задачи и подзадачи. Удобно для средних и маленьких проектов. - Jira — есть версионность, удобно для больших и долгих задач. Поддерживает массу типов разработки. Попробуйте, она вам понравится. -Чек-лист — как начать использовать Agile и Scrum на проекте -Научиться вести бэклог и расставлять приоритеты. -Проводить спринты. -Формировать стабильную и постоянную команду, решать трудности, растить внутри группы scrum-мастера. -Контролировать работу с помощью диаграммы сгорания проекта. -Организовать работу — каждый день интересоваться делами команды, проводить ретроспективу и закладывать время на тикет с запасом. -После каждого спринта демонстрировать проект. -Изучить инструменты и найти самый удобный.
Кейс-задача № 5	<p>Таким образом, Agile-методология основывается, в первую очередь, на визуальном контроле.</p> <p>Чаще всего участники проекта, работая над достижением результата, пользуются специальными цветными карточками. Один цвет сигнализирует о завершении планирования какого-то элемента конечного продукта, другой – о завершении его разработки, третий – о готовности и т.п.</p> <p>Визуальный контроль позволяет команде иметь наглядное представление о текущем состоянии процесса и гарантирует одинаковое видение проекта всеми ее членами.</p> <p>Члены команды и клиент в большинстве случаев работают вместе и рядом. Благодаря этому существенно ускоряются многие рабочие процессы, которые связаны с информированием участников проекта.</p> <p>Кроме того, совместная работа способствует созданию здоровой атмосферы для плодотворного и эффективного сотрудничества и скорейшего достижения результатов.</p>

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>Когда руководитель проекта, команда и клиент действуют сообща, исключается опасность недопонимания целей и утери информации.</p> <p>Все рабочие процессы становятся максимально прозрачными, а это значит, что любые возникающие проблемы можно разрешать практически моментально и находить лучшие варианты их решения.</p> <p>Особое внимание нужно уделить руководителю проекта. Его нельзя назвать человеком, раздающим указания налево и направо.</p> <p>Руководитель здесь выступает скорее в роли лидера, который задает направление и определяет правила сотрудничества и работы. Другими словами, Agile-управление является адаптируемым.</p> <p>Еще одним важным моментом Agile-методологии является разделение всего объема проекта на несколько более мелких составных частей.</p> <p>В большинстве случаев Скрам применяется в работе со сложным ПО и для разработки продукта с использованием инкрементных и итеративных методов. Благодаря ему серьезно повышается производительность команды и сокращаются временные затраты на достижение цели.</p> <p>Scrum улучшает результаты, помогает адаптировать проект к изменениям, обеспечивает более точную оценку при меньших трудозатратах на анализ и позволяет эффективнее контролировать этапы работы и сценарий проекта. Все это как нельзя лучше соответствует бизнес-целям.</p> <p>Канбан – еще один метод, делающий командную работу более результативной и продуктивной. Смысл его сводится к приданию процессу разработки максимальной прозрачности и равномерному распределению нагрузки среди участников проекта.</p> <p>Важная особенность Канбан еще и в том, что он мотивирует людей на постоянное сотрудничество, совершенствование и обучение.</p> <p>Работа по методу Канбан выстраивается на нескольких принципах. Во-первых, вся информация о проекте должна быть визуализирована, что позволяет видеть накладки, ошибки и недочеты и активно их устранять.</p> <p>Во-вторых, работа над одной задачей должна вестись одновременно всей командой – это помогает сбалансировать усилия и получаемые результаты, исключает неравномерное распределение нагрузки.</p> <p>И, в-третьих, время на выполнение всех задач строго контролируется, благодаря чему оптимизируется процесс и экономится время.</p> <p>В отличие от Скрам, Канбан обрел популярность намного позже, но это ни в коей мере не умаляет его достоинств и не делает менее эффективным. Метод полезен как в IT-области, так и в бизнес-сфере.</p> <p>Это лишь примеры основных методов управления проектами, основанных на Agile. Но не стоит пренебрегать и другими методами, такими</p>

№ п/п	Подробные ответы обучающегося на практические кейсы-задачи
	<p>как PRINCE2, Lean, Six Sigma, XP, CCPM, ECM, Waterfall и другие. К тому же у Аджайл, наряду с преимуществами, есть и некоторые недостатки.</p> <p>Постигая Agile, важно знать как о положительных, так и об отрицательных сторонах этой методологии.</p> <p>Scrum создавался в первую очередь для гибкости и ускорения разработки.</p> <p>Для этого появилась механика спринтов — весь процесс делится на отрезки, обычно от одной до четырех недель. Команда забирает из бэклога часть задач. Каждая разбивается на максимально мелкие тикеты. Теперь нужно оценить время на задачу, и вот здесь проявляется особенность Scrum.</p> <p>Дело в том, что люди плохо считают процессы в абсолютных величинах. Сложно сказать, сколько часов что займет. Поэтому в Scrum используется относительная оценка.</p> <p>За основу берется простая функция, которую все оценивают одинаково — например, понятно, что ее сделают за час. Остальные тикеты вычисляются так — «это мы будем делать раз в пять дольше по времени».</p> <p>В этом основное преимущество Scrum — гибко менять список задач во время разработки, не делать лишнего и не получать тысячи правок после завершения проекта, как в каскадной методологии разработки.</p> <p>Таким образом, работать по системе можно даже на бумаге. Отлично подходит и таблица в Google Docs. Создаем свою рабочую область вручную или попробуем специальные сервисы:</p> <ul style="list-style-type: none"> - Trello — подходит для маленьких проектов, быстро и удобно. - Scrumban — есть разные доски, вложенные задачи и подзадачи. Удобно для средних и маленьких проектов. - Jira — есть версионность, удобно для больших и долгих задач. Поддерживает массу типов разработки. Попробуйте, она вам понравится. -Чек-лист — как начать использовать Agile и Scrum на проекте -Научиться вести бэклог и расставлять приоритеты. -Проводить спринты. -Формировать стабильную и постоянную команду, решать трудности, растить внутри группы scrum-мастера. -Контролировать работу с помощью диаграммы сгорания проекта. -Организовать работу — каждый день интересоваться делами команды, проводить ретроспективу и закладывать время на тикет с запасом. -После каждого спринта демонстрировать проект. -Изучить инструменты и найти самый удобный.

Дата: _____

(подпись)(ФИО обучающегося)

Директору Института
Информационных технологий

Захарову А.В.

(указать Ф.И.О.)

от _____

(Ф.И.О. ответственного лица
от Профильной организации)

СПРАВКА²

Дана _____ в том, что
(Ф.И.О. обучающегося полностью)
он(а) действительно проходил(а) производственную практику (научно-исследовательская
работа) 1 семестр 4неделив Университете Синергия
(наименование вида и типа практики)

с «__» _____ 20__ г. по «__» _____ 20__ г.

Обучающийся(аяся) _____ успешно прошел(а)
(фамилия, инициалы обучающегося)

инструктаж по соблюдению правил противопожарной безопасности, правил охраны труда, техники безопасности, санитарно-эпидемиологических правил и гигиенических нормативов, после чего был(а) допущен(а) к выполнению определенных индивидуальным заданием видов работ, связанных с будущей профессиональной деятельностью.

К должностным обязанностям и поставленным задачам в соответствии с индивидуальным заданием практикант относился добросовестно, проявляя интерес к работе. Порученные задания выполнил в полном объеме в установленные программой практики сроки.

**Ответственное лицо от
Профильной организации**

М.П.(при наличии)

(Ф.И.О.)*(подпись)*

«__» _____ 20__ г.

²Справка оформляется на фирменном бланке Профильной организации

**Аттестационный лист**

обучающий(ая)ся _____ 1 _____ курса _____ очной _____ формы обучения
(указать курс)(очной, очно-заочной, заочной)
группы _____ по направлению подготовки/ специальности 09.04.03 Прикладная
(шифр группы)(код, наименование направления)
информатика, профиль/ специализация Управление информационными технологиями в
подготовки/ специальности(наименование профиля/ специализации)
организации, успешно прошел(ла) производственную практику (научно-исследовательскую
работу)
(наименование вида и типа практики)
с «___» _____ 20_ года по «___» _____ 20_ года в Профильной организации:

Университет Синергия
Институт Информационных технологий

(наименование учебных, учебно-производственных мастерских, лабораторий, учебно-опытных хозяйств, учебных полигонов, учебных баз практики и иных структурных подразделений образовательной организации)

город Москва, Мещанская ул., д.9/14 стр. 1

(юридический адрес)

I. Заключение-анализ результатов освоения программы практики:**Индивидуальное задание обучающимся (нужное отметить ✓):**

- выполнено;
- выполнено не в полном объеме;
- не выполнено;

Владение материалом (нужное отметить ✓):**Обучающийся:**

- умело анализирует полученный во время практики материал;
- анализирует полученный во время практики материал;
- недостаточно четко и правильно анализирует полученный во время практики материал;
- неправильно анализирует полученный во время практики материал;

Задачи, поставленные на период прохождения практики, обучающимся (нужное отметить ✓):

- решены в полном объеме;
- решены в полном объеме, но не полностью раскрыты;
- решены частично, нет четкого обоснования и детализации;
- не решены;

Спектр выполняемых обучающимся функций в период прохождения практики профилю соответствующей образовательной программы (нужное отметить ✓):

- соответствует;
- в основном соответствует;
- частично соответствует;
- не соответствует;

Ответы на практические кейсы-задачи, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, обучающийся (нужное отметить ✓):

- дает аргументированные ответы на вопросы;
- дает ответы на вопросы по существу;
- дает ответы на вопросы не по существу;
- не может ответить на вопросы;

Оформление обучающимся отчета по практике (нужное отметить ✓):

- отчет о прохождении практики оформлен правильно;
- отчет о прохождении практики оформлен с незначительными недостатками;
- отчет о прохождении практики оформлен с недостатками;
- отчет о прохождении практики оформлен неверно;

Аттестуемый продемонстрировал владение следующими компетенциями:

Код компетенции	Содержание компетенции	Уровень освоения обучающимся (нужное отметить ✓)*
Универсальные компетенции		
УК-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий	<ul style="list-style-type: none"> • высокий • средний • низкий
УК-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки	<ul style="list-style-type: none"> • высокий • средний • низкий
Профессиональные компетенции		
ПК-2	Организует выявление потребностей в ИТ-проектах, участвует в формировании и согласовании целей, задач и бюджетов ИТ-проектов, обеспечивает планирование, согласование с заинтересованными лицами этих планов и контроль выполнения ИТ-проектов.	<ul style="list-style-type: none"> • высокий • средний • низкий

Примечание:

- Высокий уровень – обучающийся уверенно демонстрирует готовность и способность к самостоятельной профессиональной деятельности не только в стандартных, но и во внештатных ситуациях.
- Средний уровень – обучающийся выполняет все виды профессиональной деятельности в стандартных ситуациях уверенно, добросовестно, эффективно.
- Низкий уровень – при выполнении профессиональной деятельности обучающийся нуждается во внешнем сопровождении и контроле.

II. Показатели и критерии оценивания результатов практики

Оценочный критерий	Максимальное количество баллов	Оценка качества выполнения каждого вида работ (в баллах)
Выполнение индивидуального задания в соответствии с программой практики	30	
Оценка степени самостоятельности проведенного решения практических кейсов-задач, необходимых для оценки знаний, умений, навыков и (или) опыта деятельности по итогам практики	30	
Оценка качества проведенного анализа собранных материалов, данных для решения практических кейсов-задач, необходимых	40	

для оценки знаний, умений, навыков и (или) опыта деятельности по итогам практики		
Итоговая оценка:	100	

Замечания руководителя практики от Университета:

Отчет по производственной практике (научно-исследовательская работа) **соответствует** требованиям программы практики, Положения о практике обучающихся, осваивающих основные профессиональные образовательные программы высшего образования в Университете «Синергия» и **рекомендуется/ не рекомендуется к защите** «_____».

Руководитель практики от Университета _____
(подпись) (ФИО)

«__» _____ 20__ г.