**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Саратовский государственный технический университет**

**имени Гагарина Ю.А.»**

«Институт прикладных информационных технологий и коммуникаций»

Направление «Информационные системы и технологии»

Кафедра «Прикладные информационные технологии»

**КУРСОВАЯ РАБОТА**

по дисциплине «Управление ИТ-проектами»

на тему «Использование сервиса ClickUp в управлении ИТ-проектами»

Выполнили: студенты группы б1-ИФСТ-41

очной формы обучения

Изуми Эйден Иванович,

Мокроусова Ирина Сергеевна

Проверил: профессор кафедры ПИТ, д.с.н. Печенкин Виталий Владимирович

Комиссия по защите:

д.с.н., профессор кафедры ПИТ Печенкин В.В.

к.т.н доцент кафедры ПИТ Королёв М.С.

Курсовая работа защищена на оценку «\_\_\_\_\_\_\_\_\_\_\_\_\_\_»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Печенкин В. В. /

(дата, подпись члена комиссии)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Королёв М.С. /

(дата, подпись члена комиссии)

Саратов 2023

**Замечания**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Печенкин В.В.

(дата, подпись члена комиссии)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Королёв М.С.

(дата, подпись члена комиссии)

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Саратовский государственный технический университет имени Гагарина Ю.А.»**

Кафедра «Прикладные информационные технологии»

**ЗАДАНИЕ**

на выполнение курсовой работы

по дисциплине «Управление ИТ-проектами»

студенту ИнПИТ группы Б1-ИФСТ-41 Изуми Эйдену Ивановичу,

студенту ИнПИТ группы Б1-ИФСТ-41 Мокроусовой Ирине Сергеевне

В курсовой работе необходимо::

1. Провести анализ инструмента управления задачами и организации работы над ИТ-проектами ClickUp. Для этого нужно описать следующие аспекты приложения:
   1. Лицензия на использование, возможность работать с кодом приложения, стоимость владения программой, порядок лицензирования;
   2. Поддержка русского языка;
   3. Основные функции инструмента, распределение прав, ограничение доступа и т.д.;
   4. Форматы данных (возможности экспорта, импорта), возможности интеграции с другими программными продуктами (базами данных, внешними инструментами управления проектами);
   5. Отчётные формы по проекту, возможности детализации по подразделениям, по персоналиям и т.д.;
   6. Достоинства и недостатки системы управления проектами (можно по анализу отзывов пользователей);
2. Зарегистрироваться и подготовить с использованием инструмента макет небольшого проекта с несколькими участниками и относительно небольшим количеством работ. Этот период демонстрируется с использованием скриншотов проекта.
   1. Продемонстрировать наличие/отсутствие инструментов управления спринтами, оценки хода выполнения проекта.;
   2. Продемонстрировать наличие инструментов управления ролями в рамках технологии AGILE.;
3. Провести исследование наиболее часто используемых, методологий разработки программного обеспечения.
   1. Описать основные аспекты анализируемых методологий - преимущества и недостатки, применимость.
   2. Сделать выводы касательно использования конкретных методологий в той или иной ситуации.

Дата выдачи: 25.01.2023 г.

Срок выполнения: 07.04.2023 г.

Руководитель: Печенкин В. В. \_\_\_\_\_\_\_\_\_\_\_\_

Студент: Изуми Э.И. \_\_\_\_\_\_\_\_\_\_\_\_

Студент: Мокроусова И.С. \_\_\_\_\_\_\_\_\_\_\_\_

# 

# Содержание

[**Содержание 5**](#_qob258r5jnr3)

[**Цель курсовой работы 7**](#_2kx11ewapt59)

[**Введение 9**](#_6q5w2l142era)

[**Исследование 10**](#_toh27iio85f5)

[Agile 10](#_gfylcvyr1u0y)

[Waterfall 12](#_ckrvohejii7m)

[Сравнение 13](#_2ts8dvz5um10)

[Выводы 14](#_rszuvxkw7mxq)

[**Работа с проектом 15**](#_myaqqxoac892)

[Описание проекта 15](#_qo43bqw1the)

[Проектная методология 15](#_mtaqigosx0uc)

[Управление ролями 16](#_8fphag56x44f)

[Отчетные формы 19](#_5qwxmdi4pmwq)

[Управление спринтами 27](#_6rbrubr4v8i1)

[**Анализ приложения 30**](#_5spwbutzo6z8)

[Информация о лицензии 30](#_tja336qahtlr)

[Порядок использования, доступ к коду: 30](#_nw74m79drd00)

[Стоимости использования приложения: 30](#_48mrfg8bv3yk)

[Порядок лицензирования: 32](#_2b0d21po2co0)

[Основные функции 32](#_58wbpdsx9iue)

[Управление задачами: 32](#_u8dq3caotfdq)

[Автоматизация: 34](#_tqhbyyib7owy)

[Работа с документами: 35](#_nezvcz1ccium)

[Аналитические панели: 35](#_brh7006srcrv)

[Доска с общим доступом: 36](#_miseuyruiv7u)

[Заметки: 38](#_2yce463esr9u)

[Напоминания: 39](#_youpwf8tgd7v)

[Уведомления: 39](#_4dgr02ou3gi8)

[Цели: 39](#_5i02dx1veggn)

[Ролевая модель 40](#_tc9e8tn0yqwk)

[Поддержка русского языка 41](#_ws9tmf5pr8dk)

[Описание форматов данных. Интеграции 41](#_v489uiyasg3a)

[Импорт: 41](#_xcwa50dmt34z)

[Экспорт: 42](#_k879931t7s15)

[API: 43](#_x10q4a6vdfig)

[Достоинства и недостатки: 46](#_dk7oo50unv5)

[Достоинства: 46](#_bwm3l90ql25)

[Недостатки: 46](#_14xzlhklndha)

[**Заключение 47**](#_8for1uhu44so)

[**Список использованной литературы 48**](#_4ipxzo9byny6)

# Цель курсовой работы

Целью курсовой работы являются:

* Исследование наиболее популярных методологий управления процессом разработки программного обеспечения;
* анализ инструмента управления ИТ-проектами ClickUp;
* создание учебного проекта с использованием анализируемого инструмента;

В ходе выполнения работы необходимо:

* Приобрести знания об основных функциях исследуемого инструмента, в том числе - подсистемы управления задачи, подсистемы отчетности, подсистемы обработки данных и ролевой модели;
* Найти достаточное количество источников информации о современных методологиях разработки ПО;
* Проанализировать найденную информацию;
* Описать основные аспекты, присущие анализируемым методологиям;
* Сделать выводы касательно использования анализируемых методологий;

Процесс решения поставленного задания требует приобретения следующих навыков:

* Работы с литературой;
* Анализа программной документации;
* Формирование логичного и связного описания проанализированных аспектов приложения;

Также в ходе выполнения работы необходимо приобрести умение и навык работы в выбранном инструменте, навык формирования ИТ-проектов.

# 

# Введение

В условиях непрерывного развития цифрового сектора экономики всё большее число людей оказываются вовлеченными в процесс разработки программного обеспечения. Разработка проектов, в свою очередь, весьма непростая задача, для решения которой необходимо выстроить эффективный процесс управления на всем жизненном цикле процесса реализации.

Зачастую возникает ряд проблем, мешающих достижению качественного результата реализации проекта:

* нестабильные, противоречивые требования;
* нереалистичные ожидания;
* cложность и размер проекта;
* плохое планирование;

Существует множество методологий управления процессом разработки программного обеспечения, которые призваны избежать потенциальные и решать возникающие проблемы. Широко распространенными методологиями являются “классическая” (она же “Waterfall”) и “гибкая” (она же “Agile”).

Среди исследователей нет единогласного мнения относительно того, какая методология является оптимальной. Часть изыскателей считает, что гибкая методология однозначно лучше, а классическая - пережиток прошлого, который весьма часто оказывается неэффективным [3]. Другие указывают, что классическая методология является более оптимальным выбором при решении легких и понятных задач [6]. Многие отмечают, что решение этого вопроса зависит от конкретных требований к проекту [7].

В ходе исследовательской работы будут рассмотрены исторические предпосылки, преимущества и недостатки, применимость гибкой и классической методологий разработки программного обеспечения.

# Исследование

### 

## Agile

В феврале 2001 года в штате Юта США был выпущен «Манифест гибкой методологии разработки программного обеспечения» на встрече 17 независимых практиков нескольких методик программирования, именующих себя «Agile Alliance» [2]. Этот манифест был принят и подписан представителями следующий методологий: экстремального программирования, Crystal Clear, DSDM, Feature driven development, Scrum, Adaptive software development, Pragmatic Programming. Именно с этого события началось увеличение количества применения Agile в процессе разработки ПО.

Особенностями данной методологии являются высокая гибкость и быстрая реализация отдельных частей программного обеспечения. При этом предусмотрена обратная связь от заказчика, позволяющая обнаружить несоответствия в ожидаемом и полученном результатах как можно раньше.

При работе по Agile процесс разработки ПО разбивается на короткие циклы - спринты (итерации) (рис. 1). Формируется “бэклог”, который состоит из задач, запланированных на реализацию в рамках итерации. Таким образом, каждая итерация похожа на программный проект в миниатюре, но включает в себя только тот набор задач, который необходим для небольшого прироста функциональности продукта.

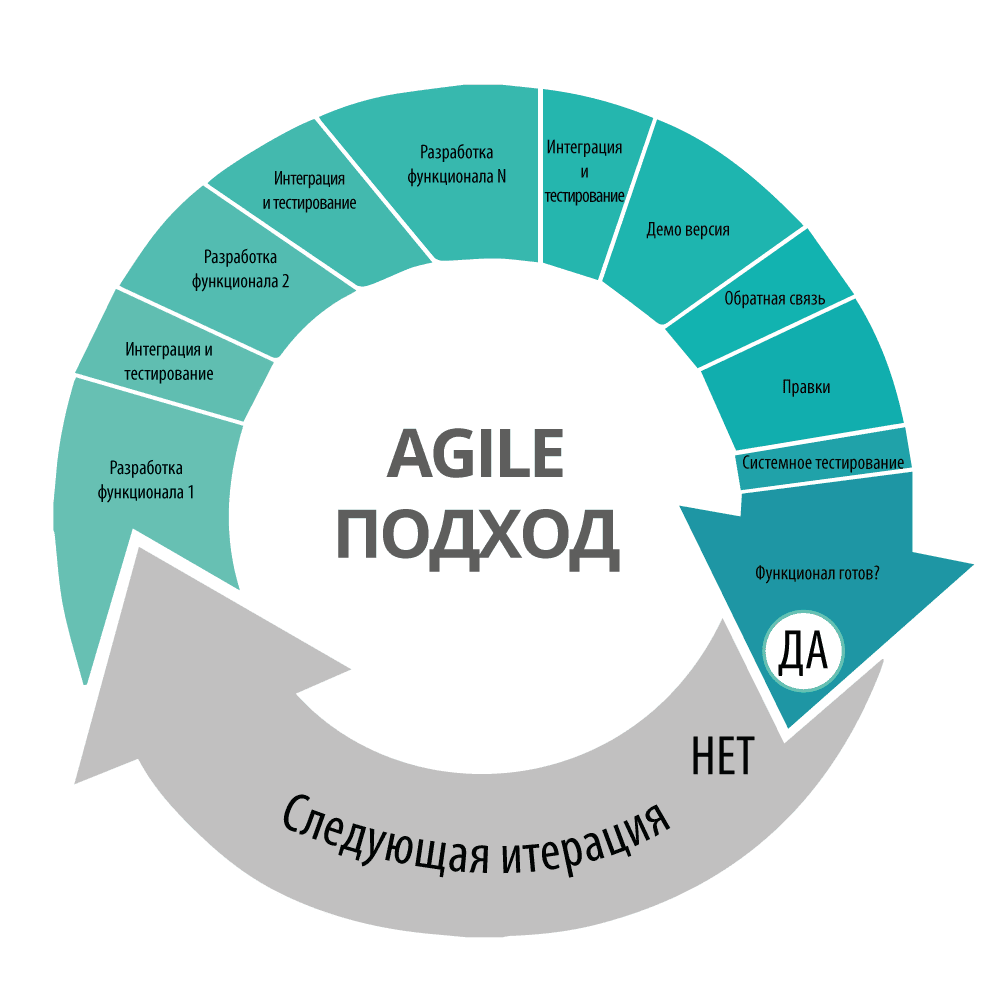


Рисунок 1 - Цикл реализации проекта

Одной из основных концепций гибкой методологии является отсутствие заранее продуманного плана (на полную разработку), адаптация под текущие условия внешней и внутренней среды, формирование плана только на итерацию.

Следует отметить, что некоторым заказчикам Agile не подойдет, ввиду отсутствия возможности постоянной работы с командой разработки программного обеспечения [8].

В некоторых случаях гибкая методология может приводить к тратам большого количества времени на обсуждения и встречи, за счет уменьшения времени производства [4].

## Waterfall

В 1970 году в своей статье У.У. Ройс описал то, что принято сейчас называть “каскадная модель” (waterfall) (в виде концепции), и обсуждал недостатки этой модели [9].

Согласно каскадной модели процесс разработки переходит от одной фазы к другой строго последовательно, после полного завершения предыдущей фазы. При этом любые другие переходы и перекрытие фаз отсутствуют. В исходной статье были выделены следующие фазы:

* Определение требований
* Проектирование
* Конструирование
* Воплощение
* Тестирование и отладка
* Инсталляция
* Поддержка

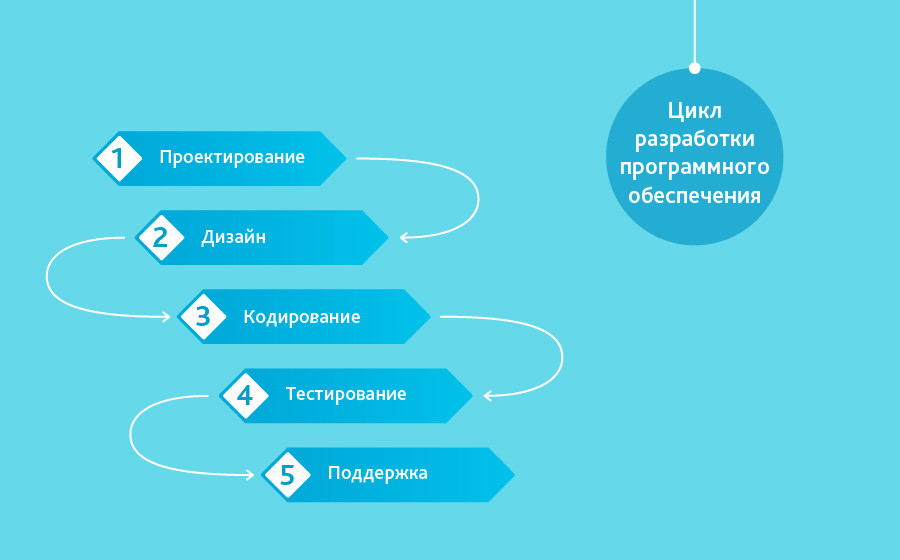


Рисунок 2 - Визуализация процесса разработки в каскадной модели

## Сравнение

Таблица 1 - Сравнительный анализ гибкой и каскадной методологии.

| **Каскадная методология** | **Гибкая методология** |
| --- | --- |
| Разработка происходит строго последовательно. | Разработка происходит по итерациям. |
| Нет возможности внести изменения до окончания разработки без прерывания процесса. | Можно вносить создавать запросы на изменения, которые могут попасть в работу в последующих итерациях. |
| Все виды проектов могут быть оценены и завершены. | Точная оценка временных и ресурсных затрат невозможна. Небольшие проекты могут быть реализованы быстро. |
| Соответствие исходным требованиям - показатель прогресса. | Соответствие актуальным требованиям - показатель прогресса. |
| Взаимодействие с заказчиком только только на начальной и финальной стадии. | Взаимодействие с заказчиком в течении всего процесса реализации проекта. |

## Выводы

Как каскадная, так и гибкая методология имеют свои преимущества и недостатки. Универсальный выбор для всех проектов невозможен - необходим анализ требований в каждом конкретном случае.

Если требования заранее известны, понятны и зафиксированы или проект является относительно небольшим, а команда состоит из разработчиков нужной квалификации, то хорошо подойдет каскадная методология. Несомненным плюсом, в первую очередь для заказчика, является фиксированная стоимость.

В случае, когда требования до конца не сформированы или постоянно меняются более оптимальным вариантом будет являться выбор гибкой методологии разработки. Это позволит вносить изменения в разрабатываемый продукт в ходе работы, однако не получится точно оценить ресурсные, временные или финансовые затраты.

# Работа с проектом

## 

## Описание проекта

Задача для тестового проекта формулируется следующим образом: необходимо разработать чат-бот, оказывающий услуги автоматизации разрешения бытовых ситуаций жителям многоквартирных домов в мессенджере “Telegram”.

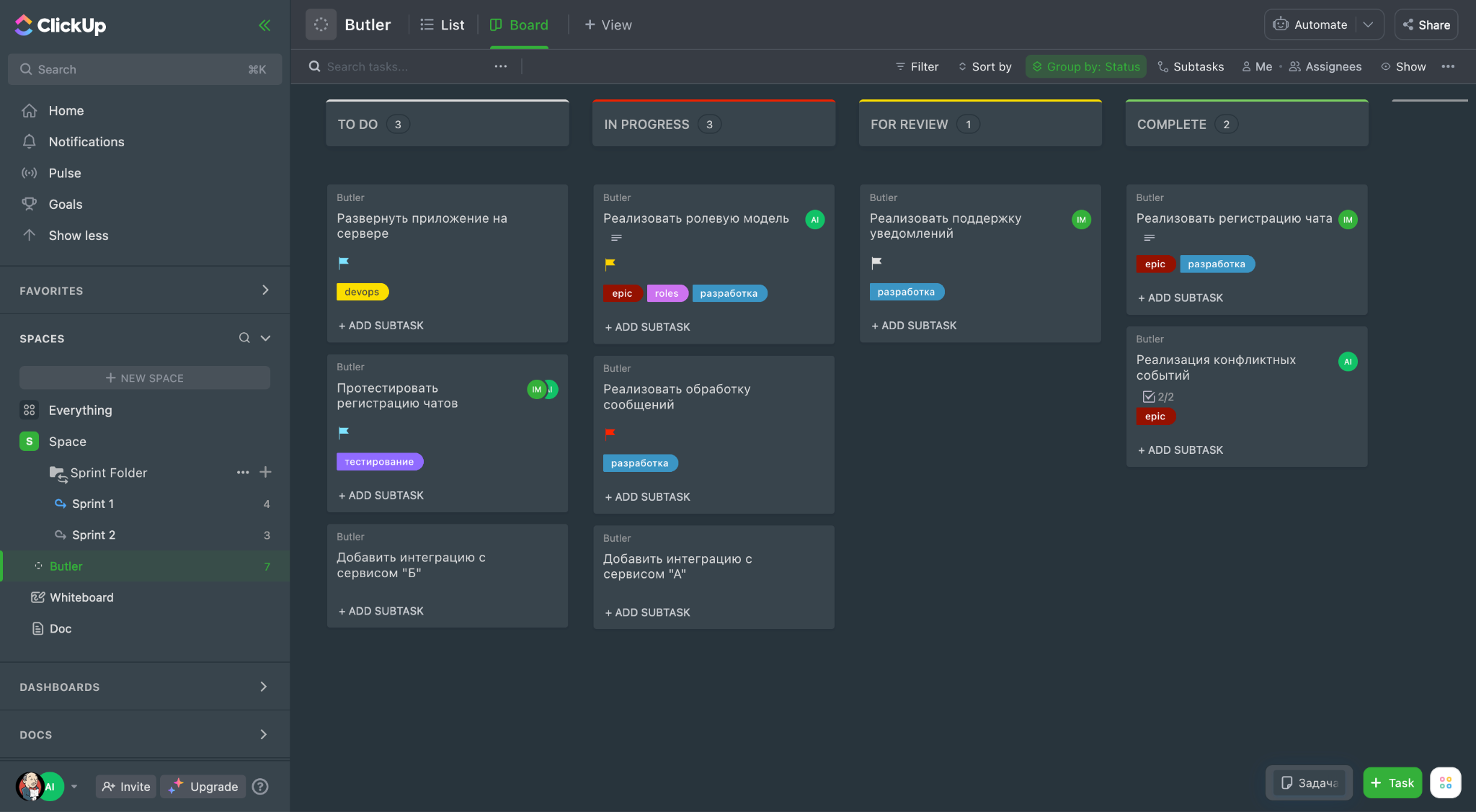
В приложение добавлено небольшое количество задач.  


Рисунок 3 - Задачи, добавленные в проект.

## Проектная методология

В проекте применена итеративная модель разработки.

Особенностью этой модели является наличие нескольких итераций, каждая из которых, по сути, является проектом в миниатюре. В рамках одной итерации разрабатывается не весь проект, а только его версия или отдельная часть.

Такая методология хорошо подходит, когда:

* Требования к конечной системы ясны и понятны;
* Основная задача должна быть определена, но детали реализации могут эволюционировать с течением времени.

Данная модель предоставляет следующие преимущества:

* **Снижение рисков** — обнаружение конфликтов между требованиями, моделями и реализацией проекта происходит на ранних этапах; присутствует большая фокусировка на основных задачах; динамическое формирование требований и управление ими.
* **Организация эффективной обратной связи** проектной команды с потребителем, создание продукта, реально отвечающего его потребностям.

Эта методология содержит следующие недостатки:

* **Проблемы с архитектурой и накладные расходы** — при работе с хаотичными требованиями и без проработанного глобального плана архитектура приложения может пострадать, а на её приведение к адекватному виду могут потребоваться дополнительные ресурсы. По сути, за возможность менять требования в ходе создания продукта, приходится так или иначе расплачиваться.
* **Нет фиксированного бюджета и сроков, а также нужна сильная вовлеченность Заказчика в процесс** — для некоторых Заказчиков это неприемлемые условия сотрудничества с разработчиком, им лучше подойдётводопадная модель (когда требования зафиксированы изначально).

## Управление ролями

ClickUp позволяет настроить доступ к проекту:

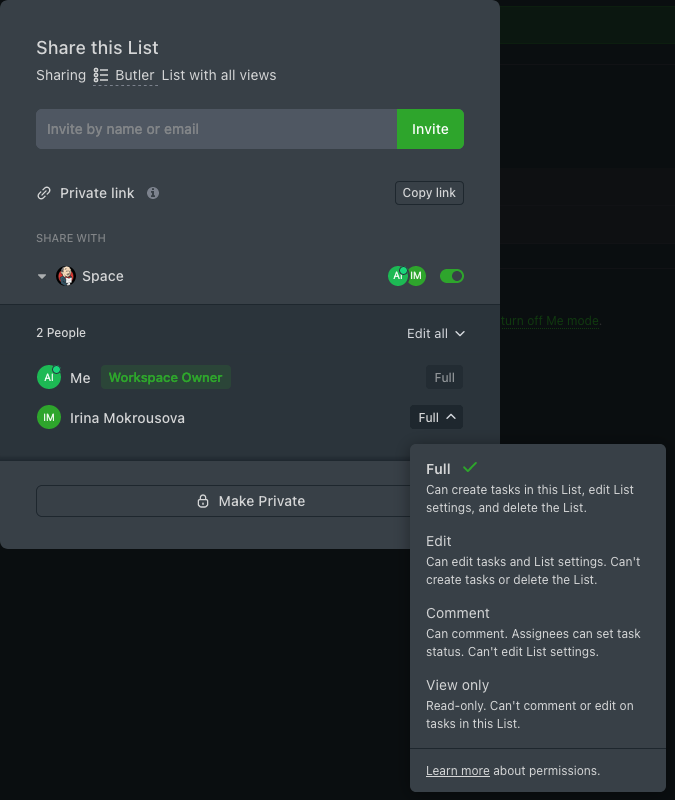


Рисунок 4 - Настройки доступа к проекту.

На приведенном рисунке видны такие опции, как:

* Полный доступ - частично соответствует team-lead, tech-lead, project-manager, scrum-master ролям в методологии Agile. (может создавать задачи, редактировать настройки, создавать доски и представления, и удалить проект)
* Доступ на редактирование - частично соответствует роли члена команды или роли собственника продукта в методологии Agile (может редактировать задачи и некоторые настройки проекта).
* Доступ на комментирование - частично соответствует роли члена команды или роли заинтересованной стороны в методологии Agile (может устанавливать текущий статус задачи, оставлять комментарии, но не может редактировать никакие другие параметры).
* Доступ на просмотр - частично соответствует роли заинтересованной стороны (не может ничего редактировать или комментировать).

Также существует возможность предоставить доступ к отдельной доске проекта.

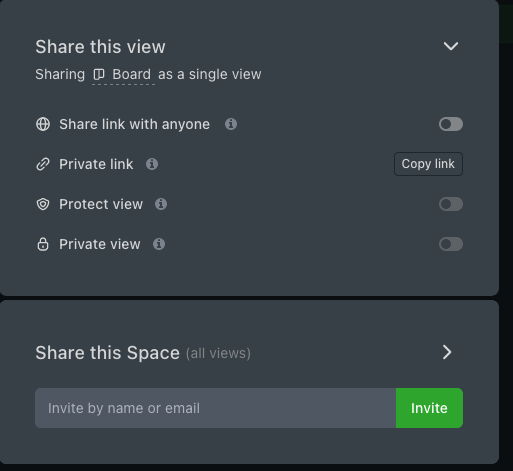


Рисунок 5 - предоставление доступа к отдельной доске проекта.

С помощью показанных средств в проекте были настроены роли. Одному из студентов была назначена роль с полным доступом, другому - доступ на редактирование. Таким образом первый являлся тимлидом, а второй - разработчиком.

## Отчетные формы

В приложении есть возможность формировать различные отчеты как по проекту, так и по отдельным спискам задачам или спринтам. Рассмотрим некоторые из них.

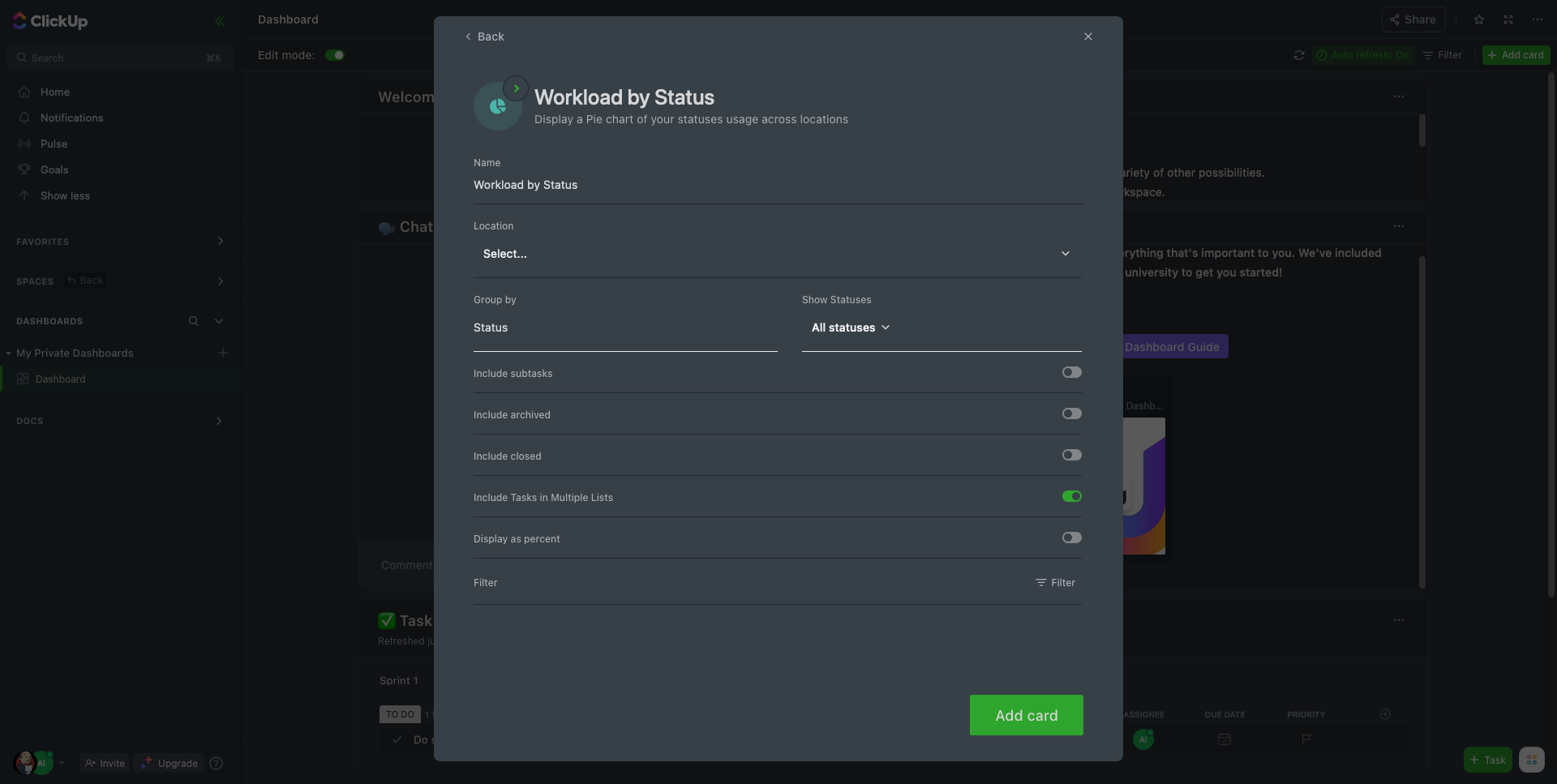


Рисунок 6 - Отчетная форма загруженности по статусу.

На приведенном рисунке изображена экранная форма отчета загруженности по статусу. Для настройки доступны следующие параметры:

* Задачи из какого списка должны учитываться. Можно указать как целый проект, так и спринт или доску.
* По какому параметру нужно группировать задачи.
* Задачи в каком статусе нужно брать для формирования отчетных данных.
* Следует ли включать подзадачи, закрытые задачи, отмененные задачи и т.д.
* Нужно ли отображать значения в процентах.

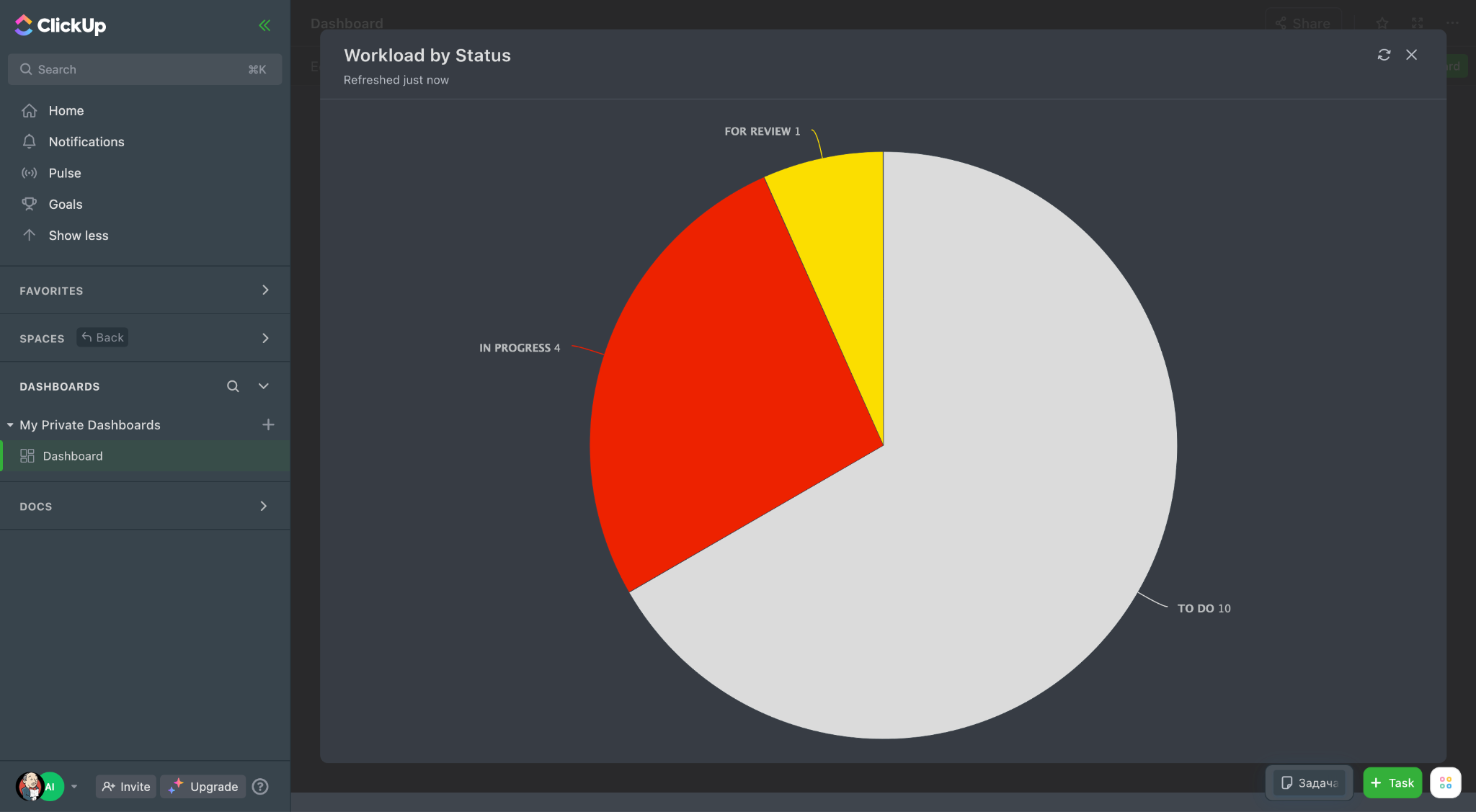


Рисунок 7 - Сформированный отчёт загруженности по статусу для тестового проекта.

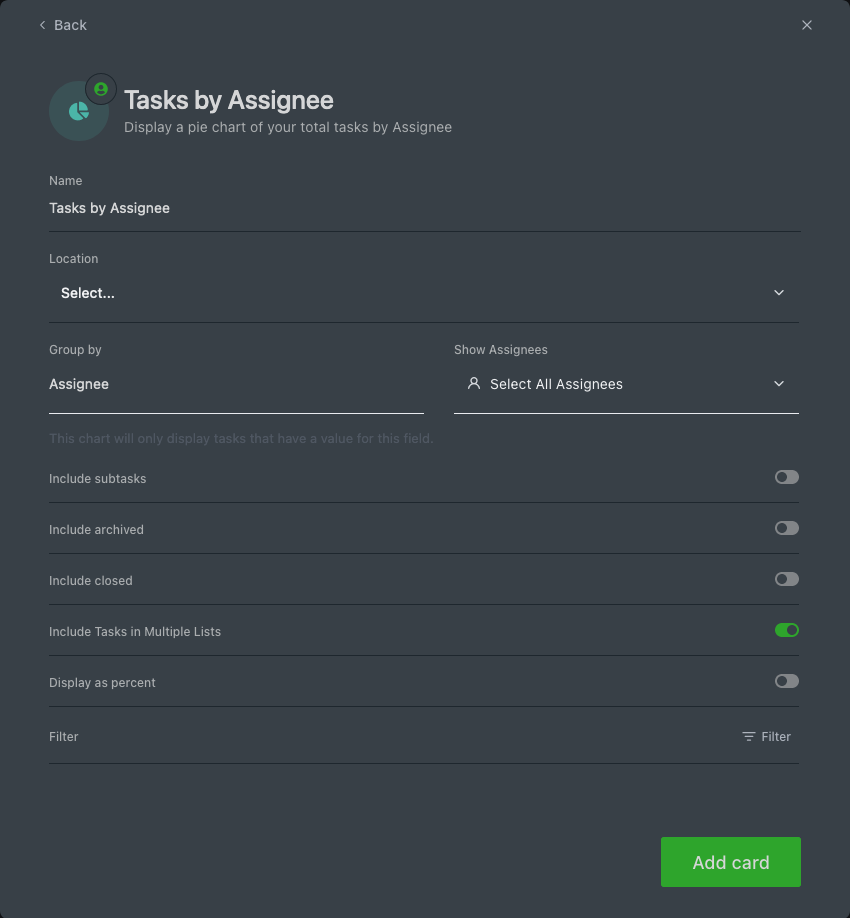


Рисунок 8 - Отчетная форма для назначенных задач.

На приведенном рисунке изображена экранная форма отчета назначенных задач. Для настройки доступны следующие параметры:

* Задачи из какого списка должны учитываться. Можно указать как целый проект, так и спринт или доску.
* По какому параметру нужно группировать задачи.
* Каких пользователей нужно отображать в отчете.
* Следует ли включать подзадачи, закрытые задачи, отмененные задачи и т.д.
* Нужно ли отображать значения в процентах.



Рисунок 9 - Сформированный отчёт назначенных задач для тестового проекта.

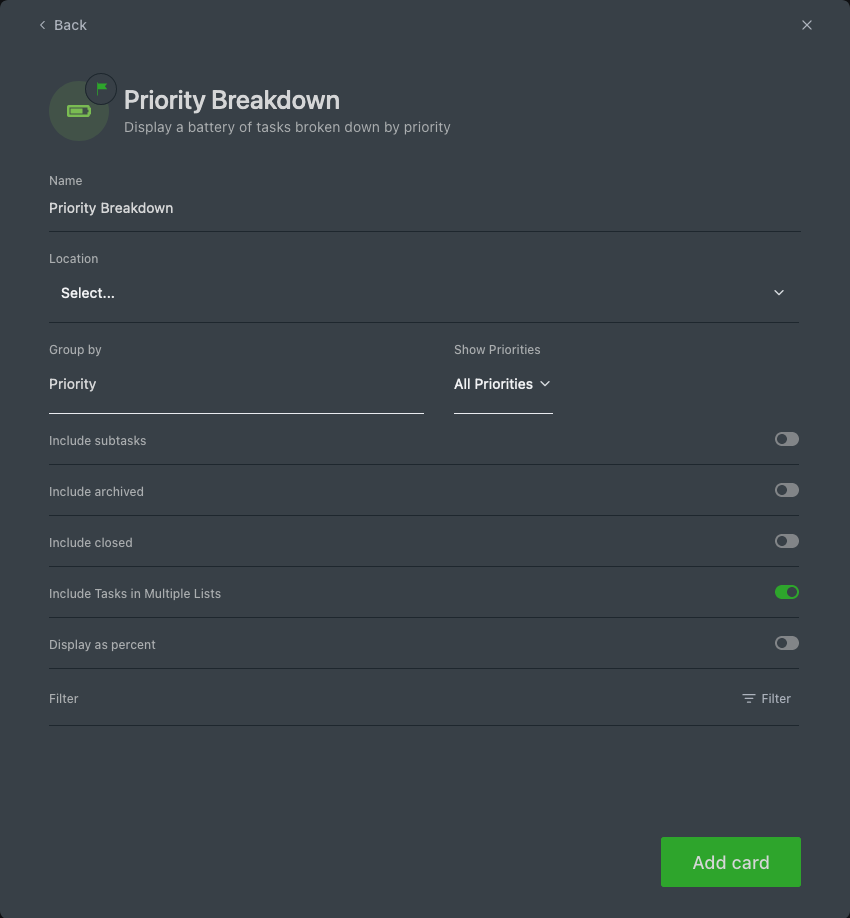


Рисунок 10 - Отчетная форма соотношения задач по приоритетам.

На приведенном рисунке изображена экранная форма отчета соотношения задач по приоритетам. Для настройки доступны следующие параметры:

* Задачи из какого списка должны учитываться. Можно указать как целый проект, так и спринт или доску.
* По какому параметру нужно группировать задачи.
* Какие уровни приоритета нужно использовать при формировании отчетных данных.
* Следует ли включать подзадачи, закрытые задачи, отмененные задачи и т.д.
* Нужно ли отображать значения в процентах.



Рисунок 11 - Сформированный отчёт соотношения задач по приоритетам.

Помимо рассмотренных отчётов в системе доступны и другие, однако для использования многих из них нужно обладать платной подпиской.

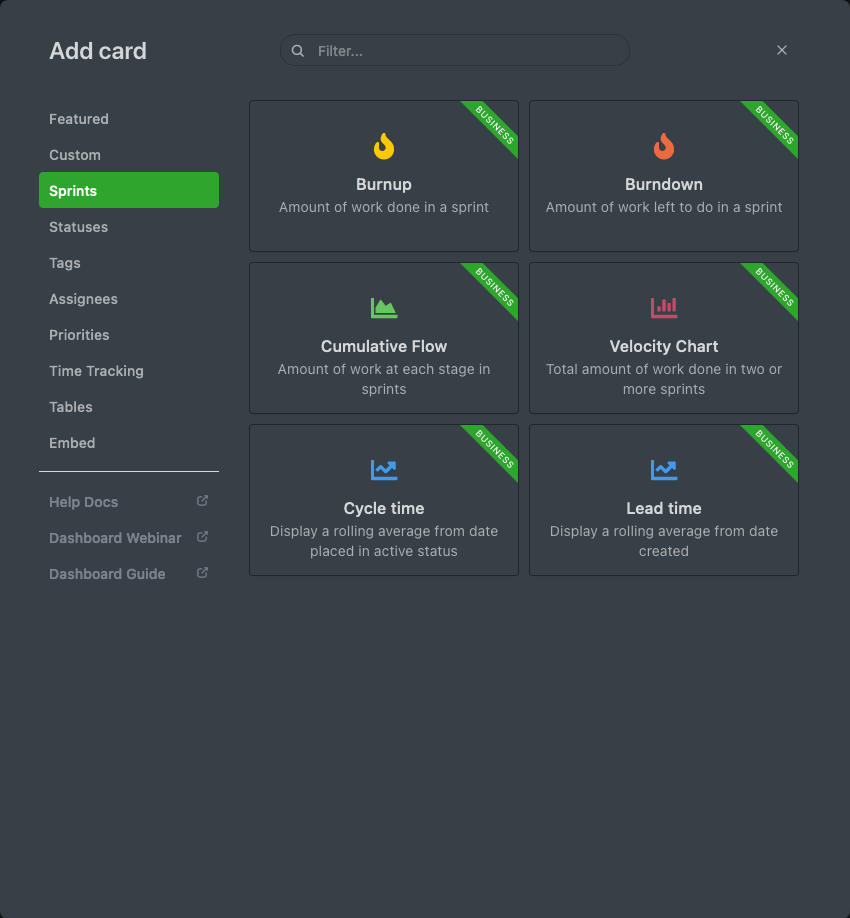


Рисунок 12 - Отчёты по спринтам для платной подписки.

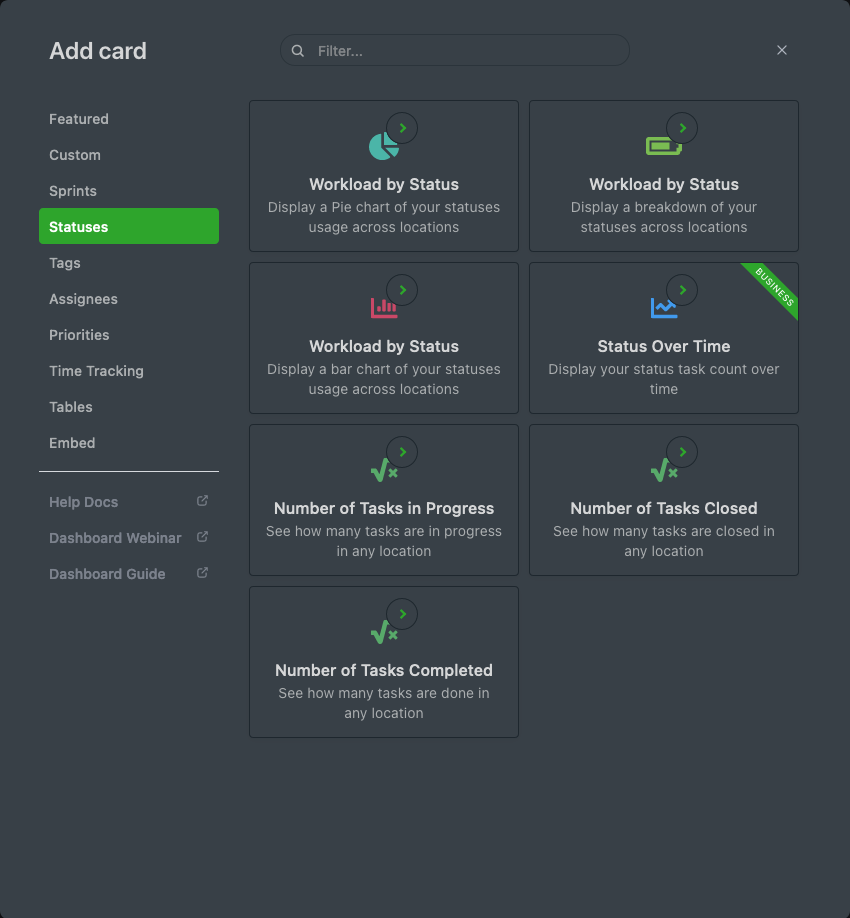


Рисунок 13 - Отчёты по статусам.

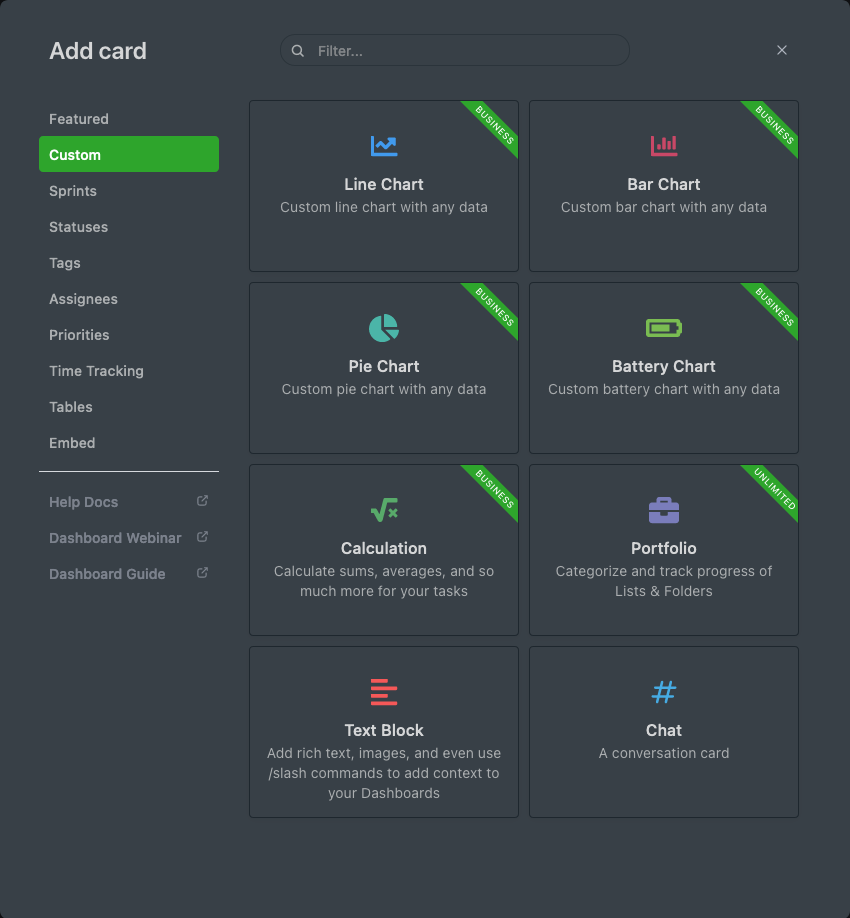


Рисунок 14 - Настраиваемые отчеты для платной подписки.

## Управление спринтами

К доступным инструментам управления спринтами можно отнести:

* Создание папок для спринтов. Папка для спринта содержит все документы, доски для рисования и т.д., относящиеся к конкретному спринту.

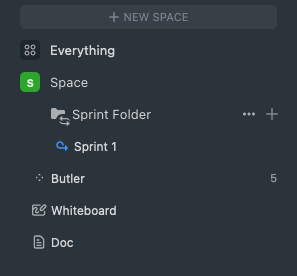


Рисунок 15 - Созданная папка для спринта.

* Создание спринтов. После создания спринта становится доступным назначение этого спринта задачам. Это позволит анализировать текущее и архивное состояния с помощью подсистемы отчётности.

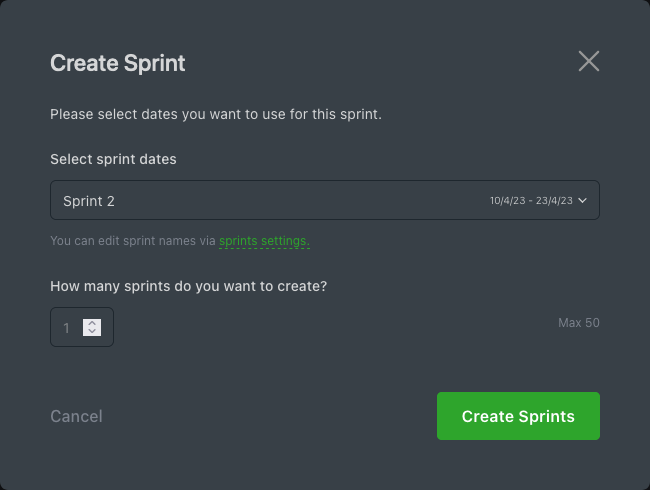


Рисунок 16 - Экранная форма создания спринта.

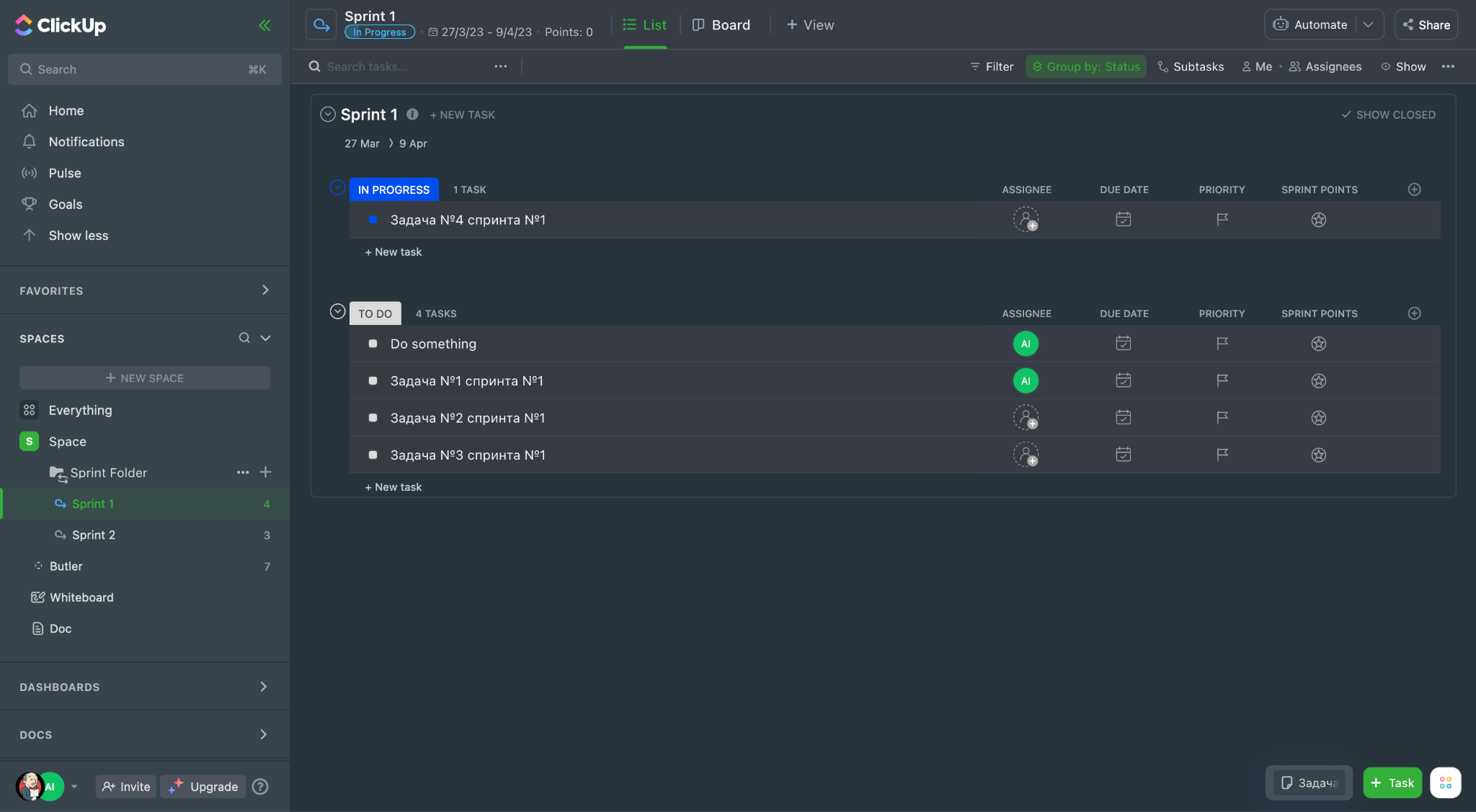


Рисунок 17 - Экранная форма спринта, содержащая задачи этого спринта.

Эти спринты можно использовать как итерации в используемой модели.

# 

# Анализ приложения

## Информация о лицензии

### Порядок использования, доступ к коду:

Как мобильное, так и веб-приложение сервиса явно запрещают попытки модифицировать и/или изменять работу сервиса. Исходный код является собственностью компании и недоступен прочим лицам, а его несанкционированная модификация запрещена. Также, согласно условиям эксплуатации, обратный инжиниринг запрещен.

Минимальный возраст, требуемый для использования сервиса - 13 лет. Использование чужого аккаунта запрещено в любом виде, даже с разрешения владельца.

### Стоимости использования приложения:

ClickUp поддерживает несколько уровней подписки на месяц и год. Среди них есть бесплатный, которого достаточно для персонального использования. Бесплатный уровень не ограничен по времени использования.

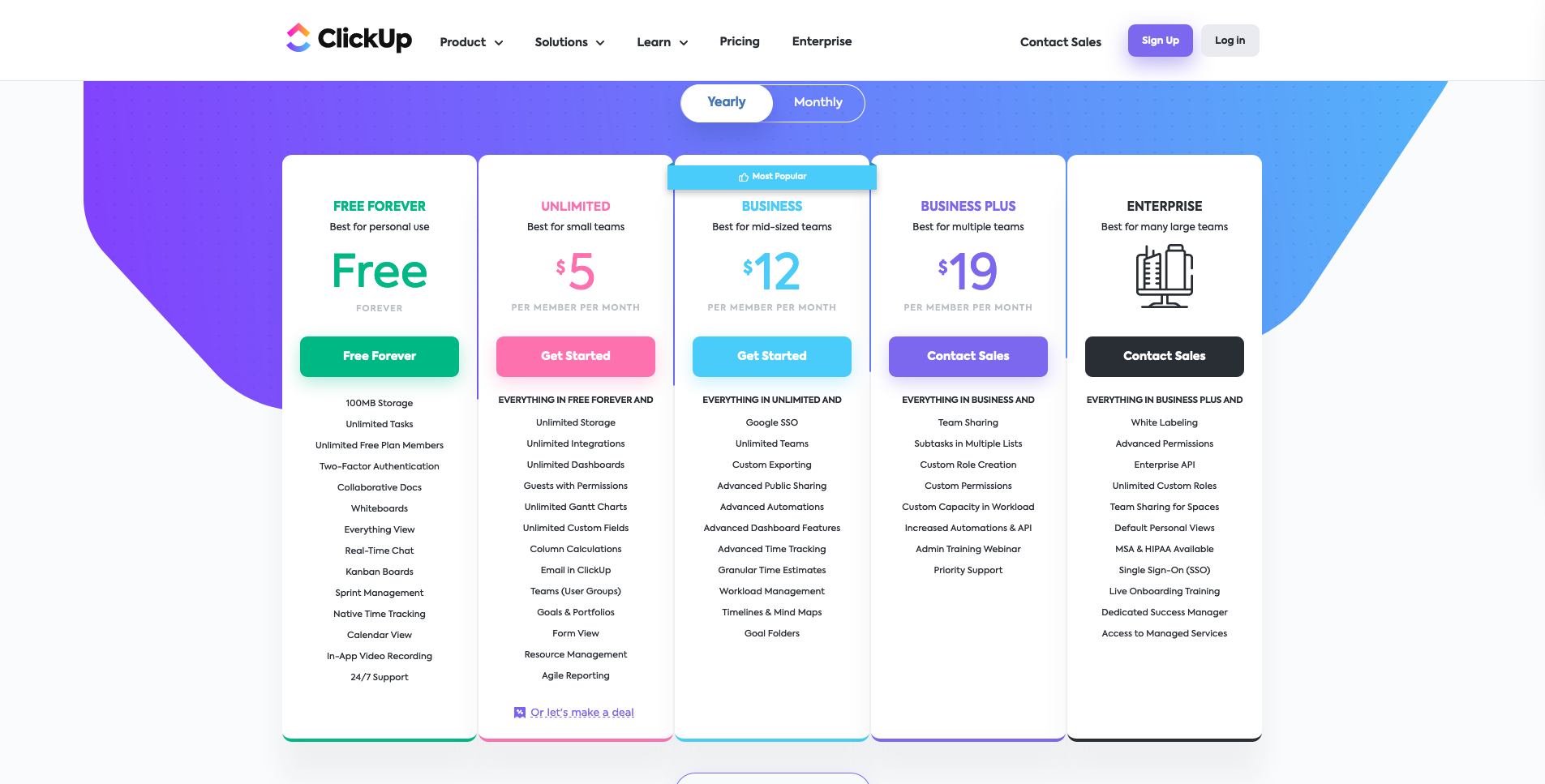


Рисунок 18 - Страница покупки подписки.

Рассмотрим функционал, доступный при бесплатном типе подписки:

* Неограниченное количество задач;
* Поддержка двухфакторной аутентификации;
* Доступ к документам с совместным редактированием;
* Kanban-доски;
* Управление спринтами;
* Возможность записи экрана внутри приложения;

На данный момент приложение предлагает следующие платные уровни подписки:

* Неограниченный уровень - 5$, либо 9$ в месяц (в зависимости от срока подписки). Подходит для небольших команд. Помимо функций, доступных в бесплатном типе, содержит:
  + Неограниченное хранилище;
  + Отсутствие ограничений в интеграциях;
  + Диаграмму Ганта;
  + Дополнительную отчётность по проекту;
* Бизнес-уровень - 12$, либо 19$ (в зависимости от срока подписки). Подходит для команд среднего размера. Помимо функций, доступных в неограниченном уровне, содержит:
  + Поддержку входа через сервисы Google;
  + Неограниченный размер команд;
  + Настраиваемый экспорт;
  + Дополнительную автоматизацию;
  + Улучшенное отслеживание времени;
  + Mind-карты;
* Расширенный бизнес-уровень - 19$, либо 29$ (в зависимости от срока подписки). Подходит для нескольких команд среднего размера. Помимо функций, доступных в бизнес-уровне, содержит:
  + Предоставление доступа между командами;
  + Подзадачи в нескольких списках;
  + Создание настраиваемых ролей;
  + Настраиваемые разрешения;
  + Приоритетная поддержка;
* Корпоративный план - стоимость обсуждается индивидуально, с отделом продаж. Подходит для больших команд. Помимо функций, доступных в расширенном бизнес-уровне, содержит:
  + Расширенные разрешения;
  + Неограниченные настраиваемые роли;
  + Поддержку от команды сервиса;

### Порядок лицензирования:

Для получения лицензии (подписки) необходимо выполнить следующие шаги:

1. Зарегистрировать аккаунт. На этом этапе доступен бесплатный уровень подписки, можно пользоваться приложением в персональных целях.
2. Оплатить платную подписку. Оплата возможно только через банковскую карту.

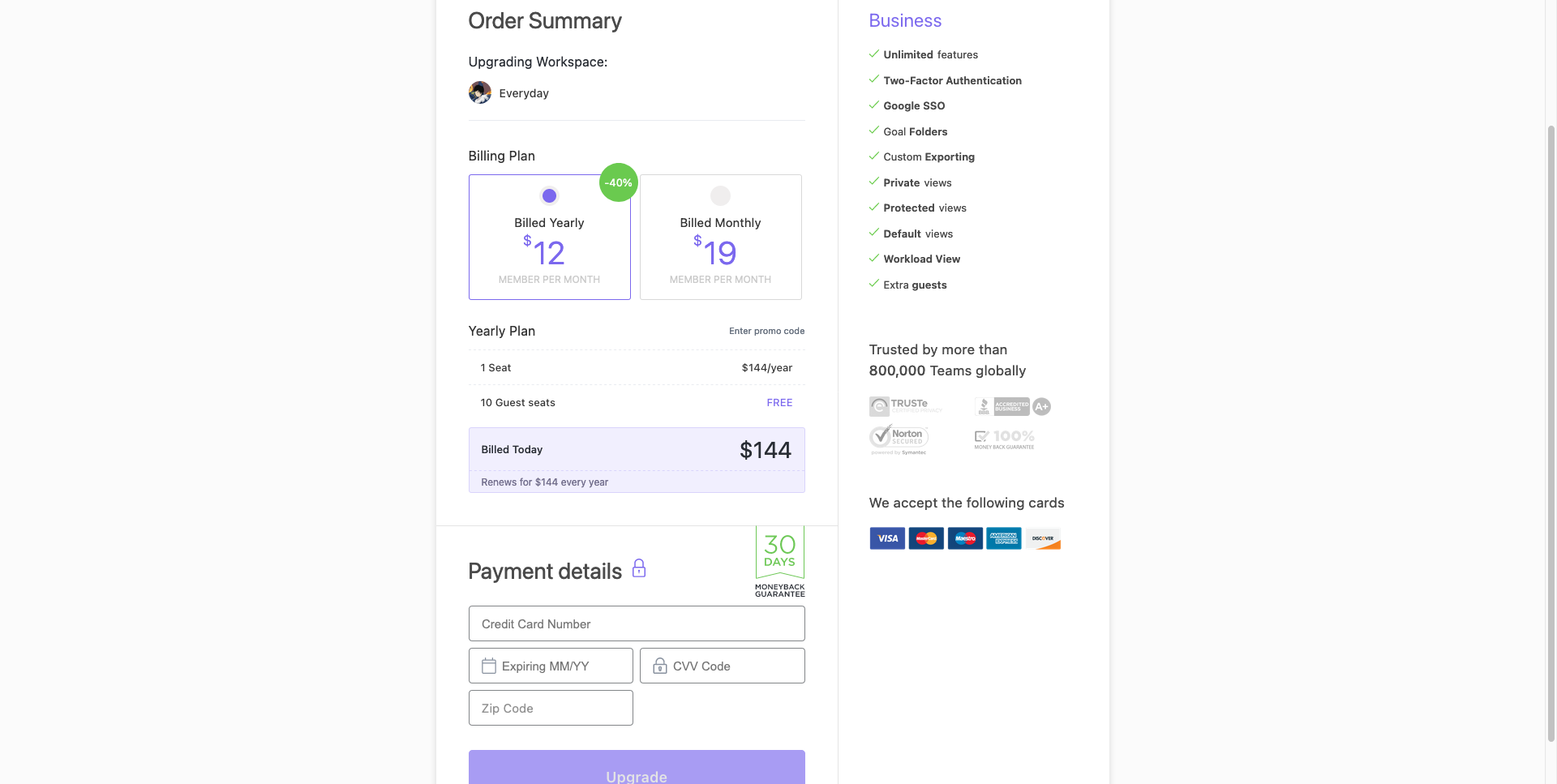


Рисунок 19 - Оплата платной подписки.

## Основные функции

### Управление задачами:

С помощью приложения можно создавать, редактировать и удалять задачи.

Задача состоит из названия, текста, даты создания, списка присвоенных тегов, истории изменений и связей с другими тегами.

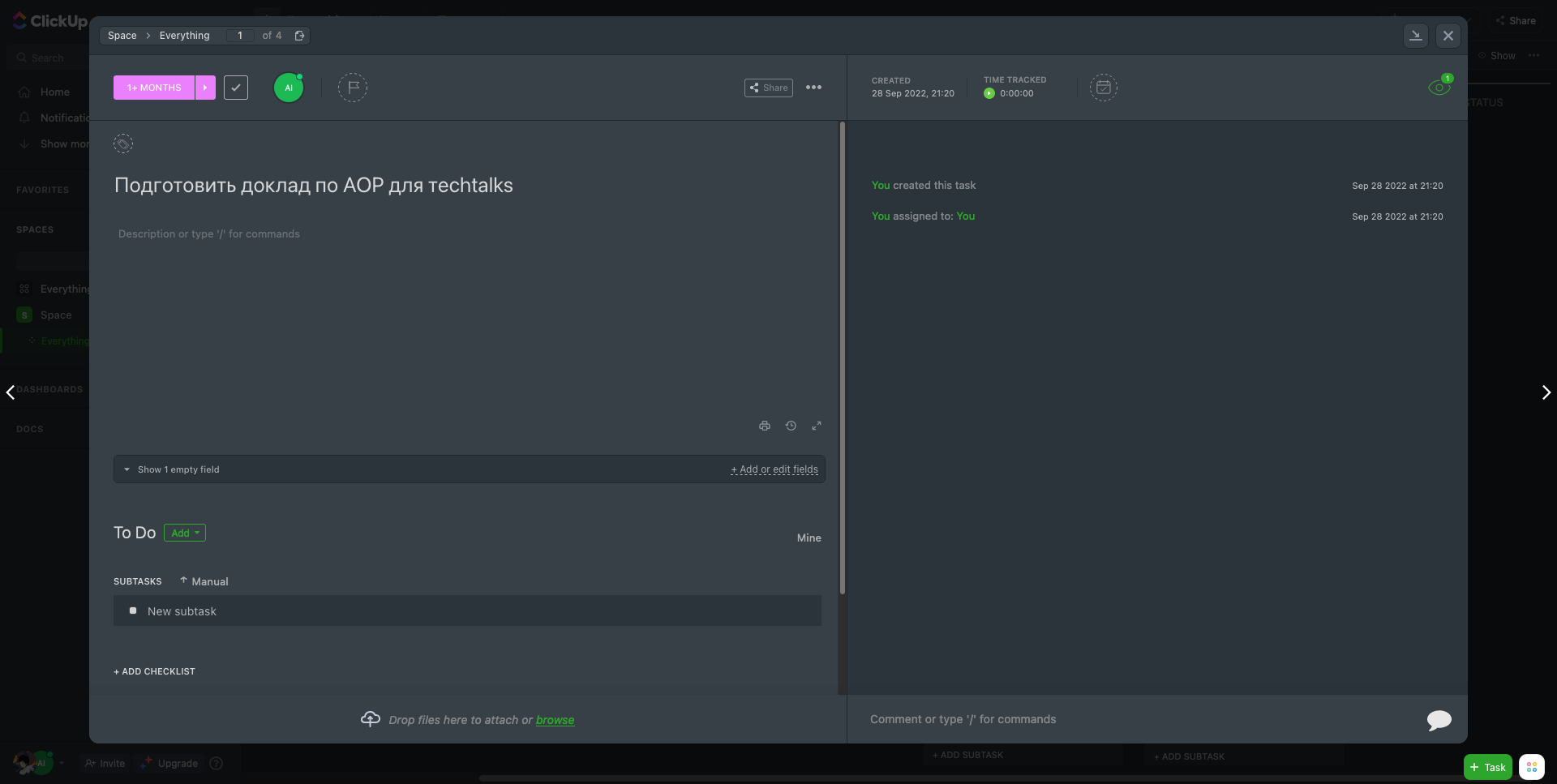


Рисунок 20 - Экран редактирования задачи.

Задачи можно группировать в рабочем пространстве, например по статусу.

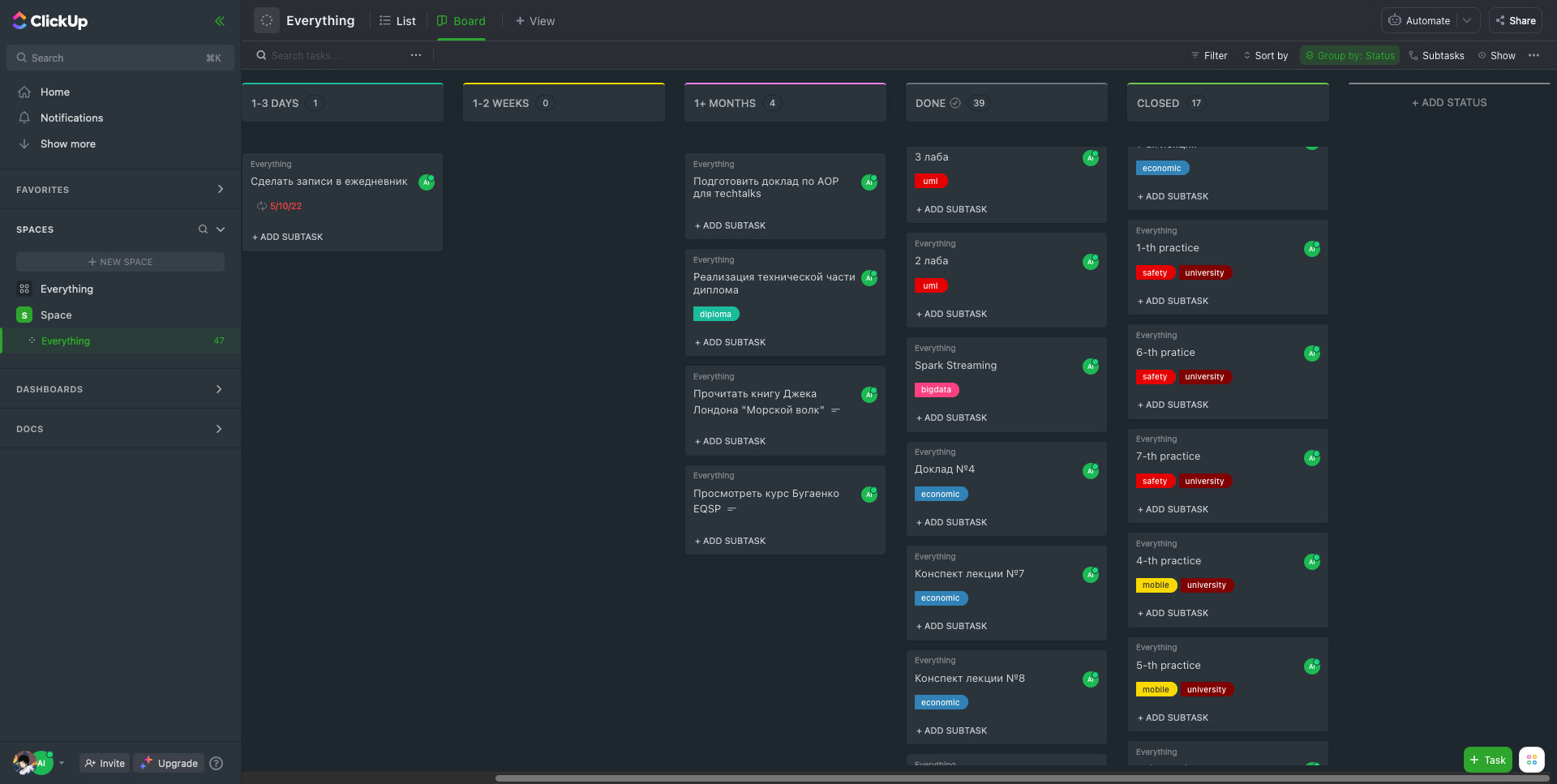


Рисунок 21 - Задачи, сгруппированные по статусу в рабочем пространстве.

Также можно просматривать все задачи, простым списком. При этом все настройки, внесенные во внешний вид отображения, сохраняются и применяются при последующем открытии.

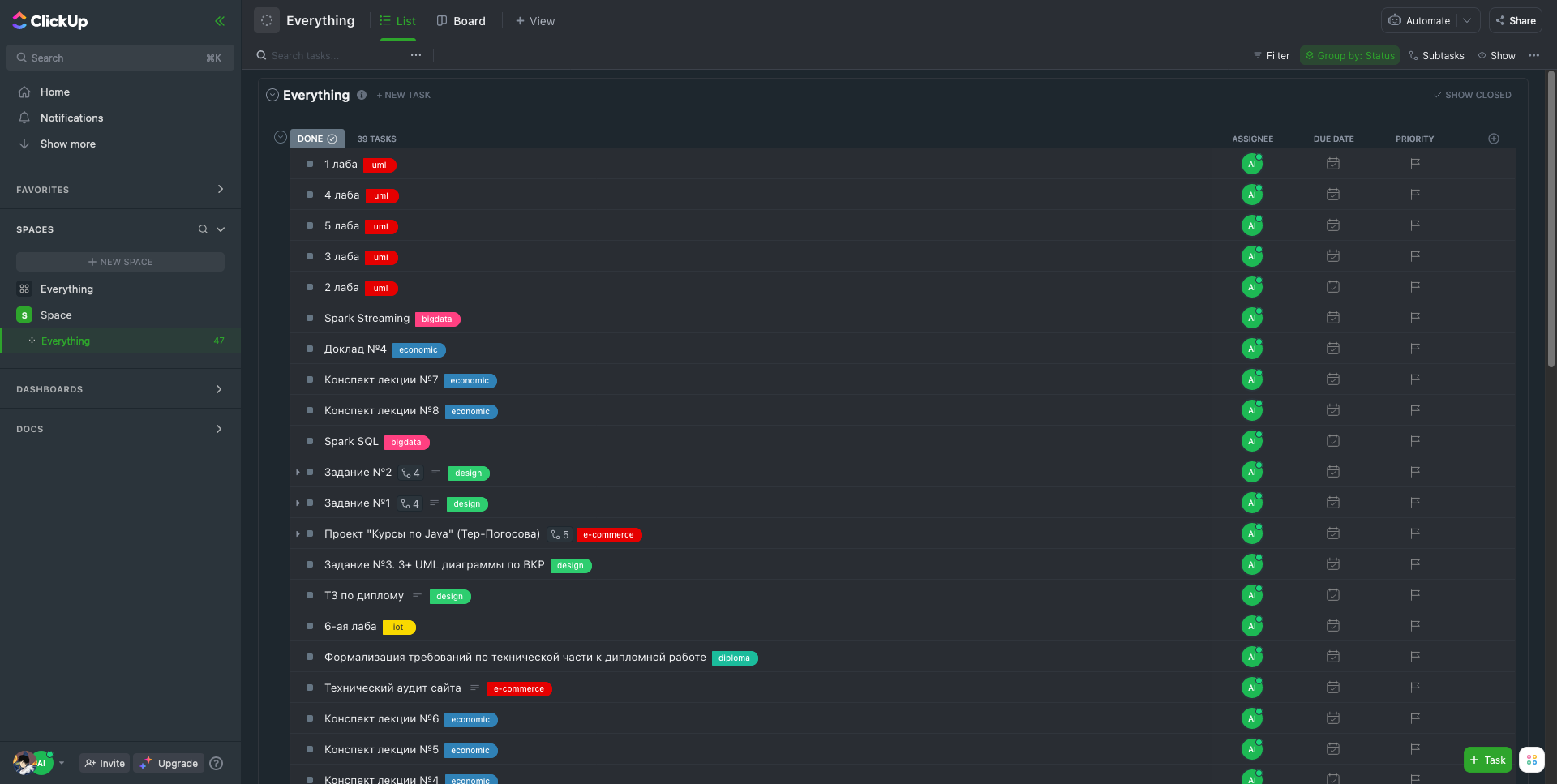


Рисунок 22 - Задачи простым списком.

### Автоматизация:

В системе ClickUp представлен функционал автоматизации рутинных действий. Можно автоматизировать типичные задачи с помощью триггеров, условий и особых действий.

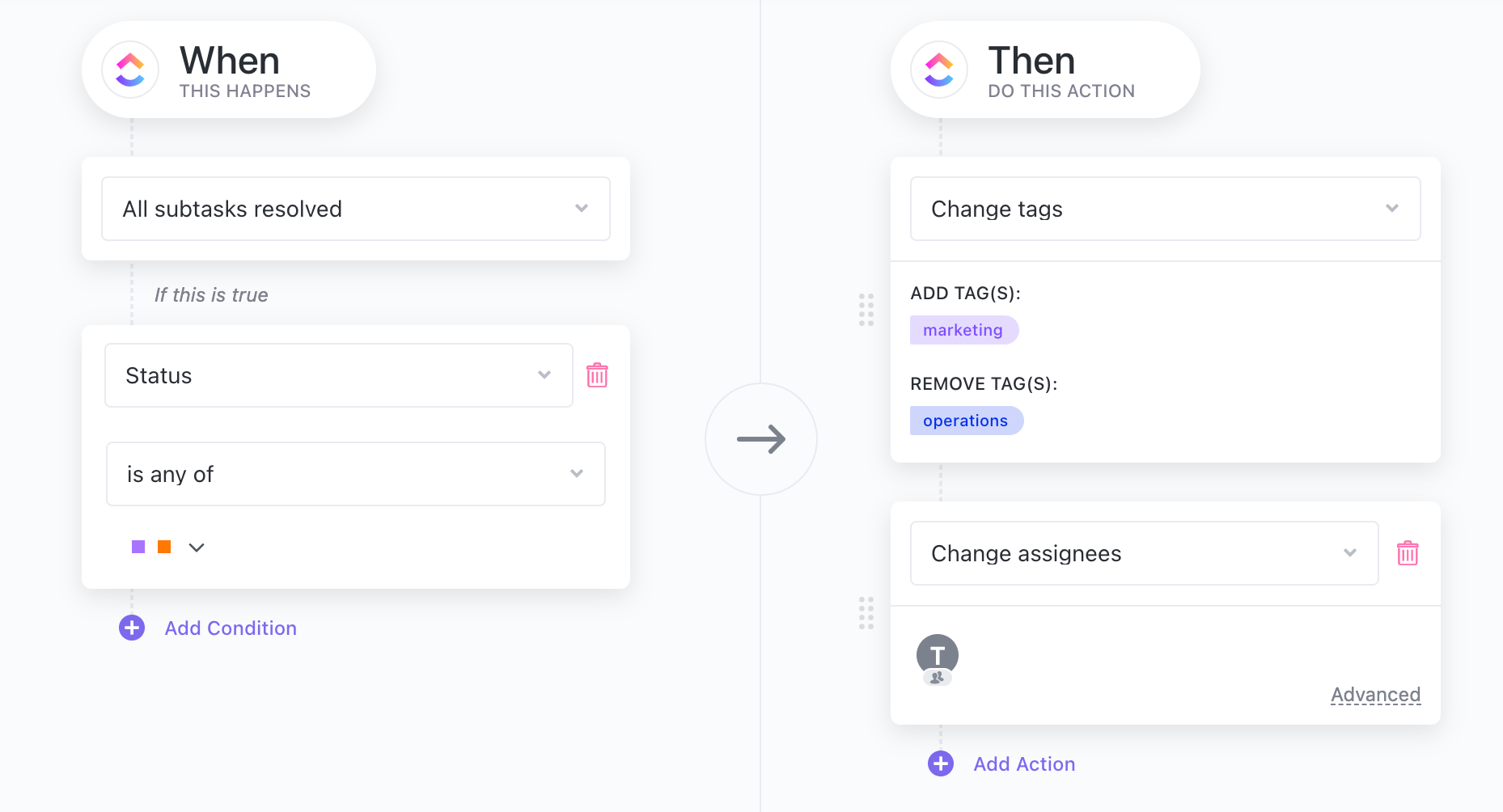


Рисунок 23 - Автоматизация процесса присвоения тегов с помощью условий и триггеров.

### Работа с документами:

Работа с документами интегрирована в задачи и проекты, тем самым ускоряя и упрощая работу. Документы можно экспортировать и импортировать, оставлять комментарии и помечать тегами.

Документы состоят из страниц, они доступны к совместному редактированию.

Существует возможность защищать страницы и документы от изменений. Можно добавлять связь с другими документами.

Доступны архивация, просмотр истории изменений и заполнение информации об авторах документа.

### Аналитические панели:

Аналитические панели являются визуальной презентацией текущего прогресса в рабочем пространстве. Можно отслеживать спринты, эпики, высокоприоритетные проекты. На панели можно добавлять задачи.

Существует функционал просмотра аналитических панелей на мобильных устройствах.

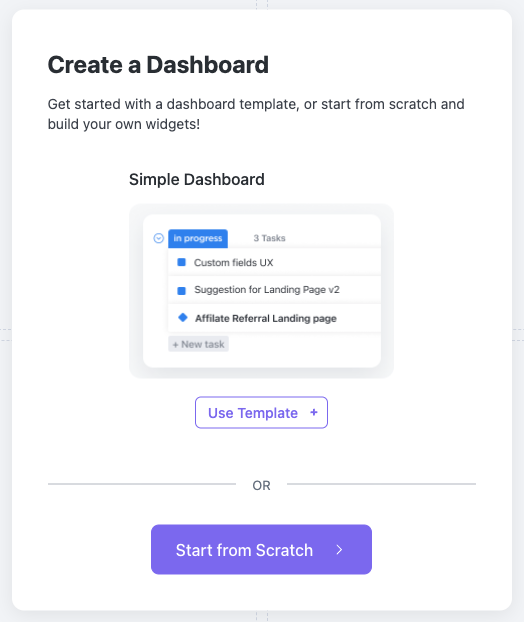


Рисунок 24 - Экран создания аналитической панели.

### Доска с общим доступом:

Доступна доска для совместной графической работы в режиме реального времени. Можно как размещать текстовую информацию, так и рисовать произвольные линии.

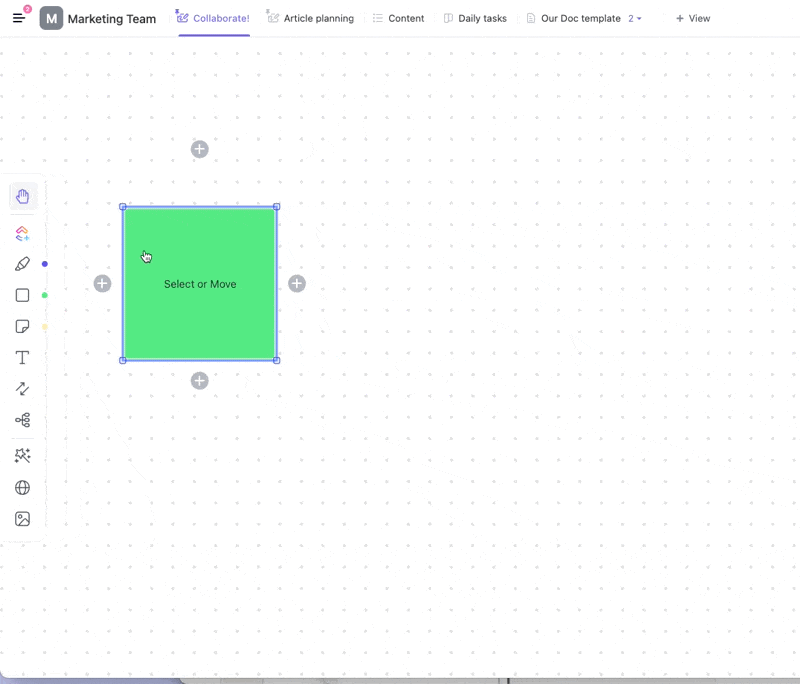


Рисунок 25 - Основной экран совместной доски.

Доступно добавление карточек приложений. К примеру - можно добавить карточку Google Sheets.

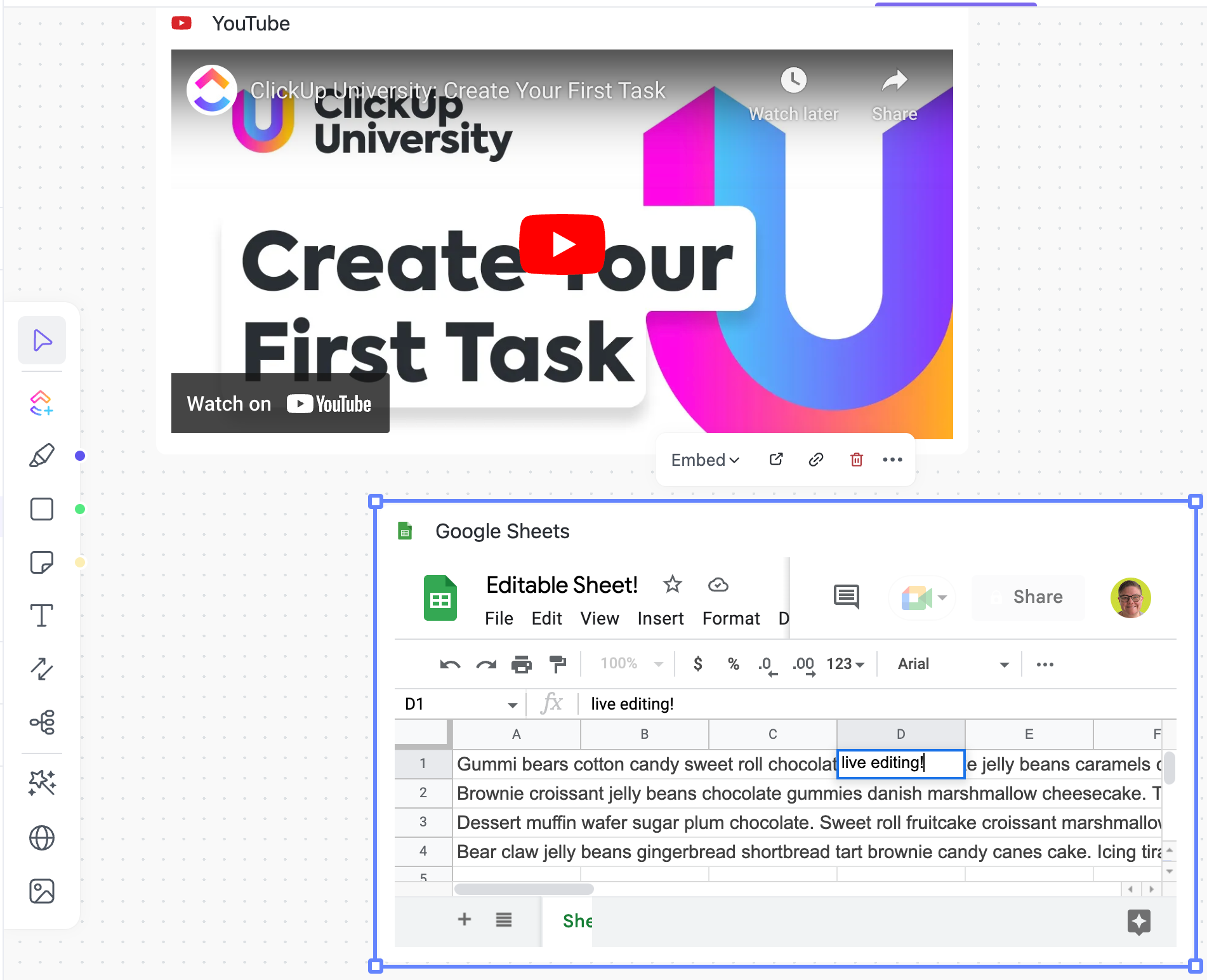
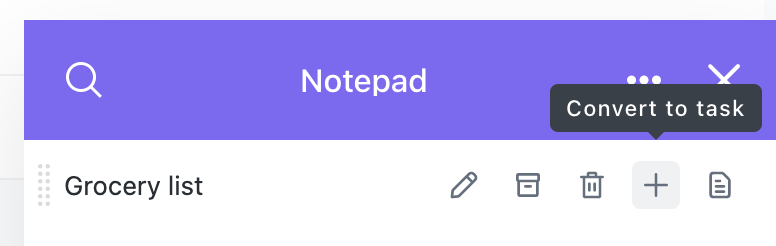


Рисунок 26 - Карточка Google Sheets на доске с общим доступом.

### Заметки:

Сервис содержит блокнот для ведения в заметки. Можно вести работу, стандартную для блокнота, а также преобразовывать заметки в задачи.

  
Рисунок 27 - Кнопка преобразования заметки в задачу.

### Напоминания:

В приложении есть возможность создавать комментарии-напоминания, которые будут высвечены пользователю в заданное время единожды или по расписанию.

### Уведомления:

Сервис предоставляет функционал уведомлений о каких-либо событиях.

Уведомления могут быть распределены по группам. К примеру - уведомления будут по умолчанию сгруппированы согласно задачам. У пользователя есть возможность управлять сразу всеми уведомлениями из группы.

Работать с уведомлениями можно следующим образом: просматривать, отвечать (особым образом), удалять и восстанавливать.

### Цели:

ClickUp поддерживает цели. Цель - особая карточка с прогрессом. Прогресс считается по завершенным задачам.

Можно выбирать параметр для сортировки, задавать список тех людей, кому будет видна цель, прятать помещенные в архив или завершенные элементы (задачи).

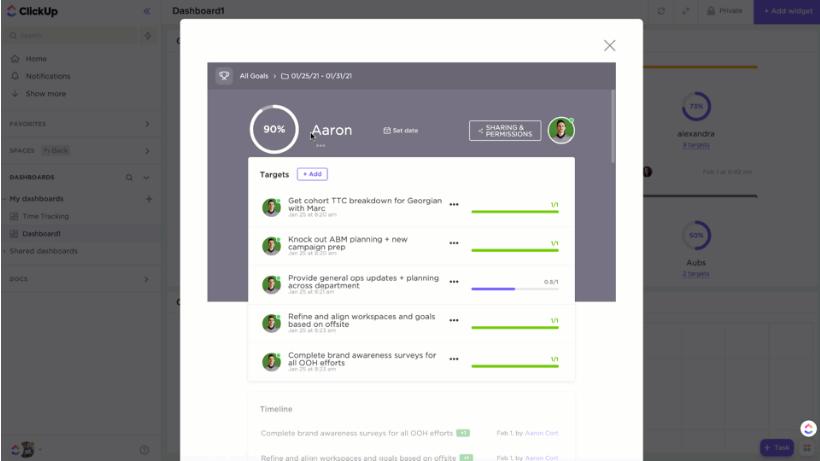


Рисунок 28 - Пример существующей цели.

## Ролевая модель

По умолчанию в рабочем пространстве представлены следующие роли:

* Гость. Пользователи с этой ролью могут просматривать указанные элементы в рабочем пространстве.
* Участник. Представителями этой роли являются обычные сотрудники компании. Для данной роли доступны все основные действия. Могут:
  + Создавать задачи, списки, папки.
  + Просматривать все общедоступные сущности (задачи, списки, папки).
  + Просматривать информацию о других пользователях из рабочего пространства.
  + Делиться вещами с людьми с гостевой ролью.
  + Делать задачи, списки, папки приватными. Тогда доступ к ним будет только у создателя или владельца.
* Администратор. Пользователи, ответственные за управление рабочим пространством (включая оплату, управление разрешениями и интеграциями). В целом администратор - это участник с дополнительными правами. Могут:
  + Добавлять новых пользователей, управлять разрешениями и удалять существующих пользователей.
  + Импортировать и экспортировать данные, управлять интеграциями.
  + Управлять оплатой и настройками рабочего пространства.
* Владелец. Пользователь, создавший рабочее пространство или получивший полное управление над ним. Имеет возможность производить любые операции с рабочим пространством.

Помимо этого можно создавать собственные роли со списком желаемых разрешений. Для этого требуется подписка бизнес-плюс или дороже.

## Поддержка русского языка

На момент написания работы продукт не поддерживает русский язык. Среди поддерживаемых языков представлены такие как: английский, французский, испанский и некоторые другие (всего шесть).

## Описание форматов данных. Интеграции

### Импорт:

Продукт имеет широкие возможности импорта и интеграции с другими приложениями.

Можно импортировать данные из таких приложений, как: Asana, Trello, Jira, Todoist, Basecamp, Wrike.

Также существует функционал по импорту данных из файлов. Файлы могут иметь один из следующих форматов: .xls, .xlsx, .csv, .tsv, .xml, .txt.

При импорте данных доступна возможность преобразования собственной колонки в колонку, поддерживаемую системой.

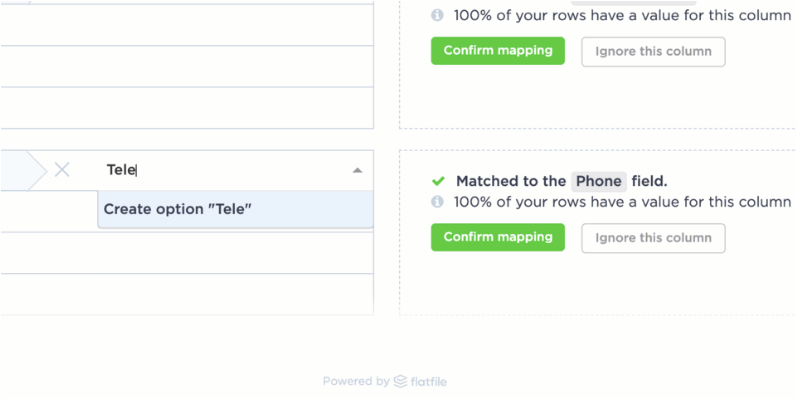


Рисунок 29 - Преобразование собственной колонки в поддерживаемую системой.

Также можно указать форматы созданных колонок.

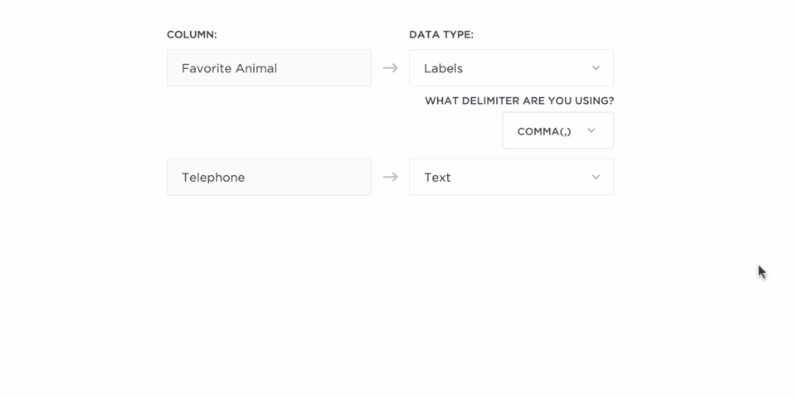


Рисунок 30 - Настройка форматов созданных колонок.

Поддерживаются все основные форматы даты и времени.



Рисунок 31 - Таблица поддерживаемых форматов даты и времени.

### Экспорт:

ClickUp поддерживает экспорт данных как рабочего пространства, так и различных способов представления пространства.

Можно экспортировать данные следующими способами:

* Все данные по задачам, в CSV-файл;
* Документы, в один из следующих форматов: PDF, HTML или Markdown.
* Графики по дашбордам (некоторые в csv).
* С помощью ClickUp API, согласно программным ограничениям.

### API:

Приложение имеет API для интеграции с любым пользовательским приложением, поддерживающим REST-запросы. При этом доступ к API входит во все существующие уровни подписки, в том числе и в бесплатный.

При интеграции обязательным требованием является наличие рабочего пространства, владельцем или администратором которого вы являетесь.

Все интеграционные взаимодействия происходят с использованием стандарта OAuth2. Рабочее пространство должно иметь собственные сгенерированные идентификатор и пароль для доступа к нему. Процедура получения идентификатора и пароля выглядит следующим образом:

1. Открыть настройки рабочего пространства.

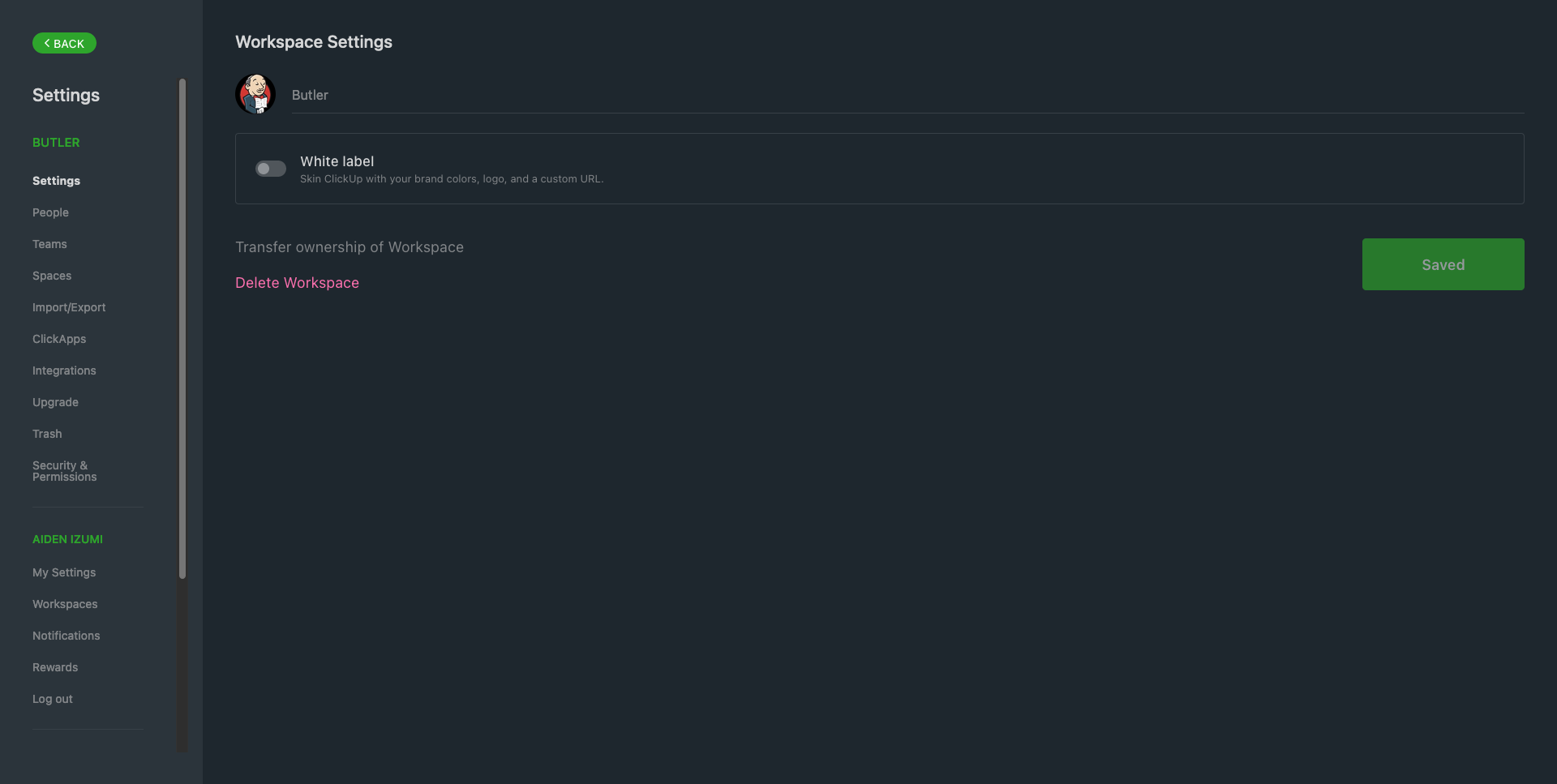


Рисунок 32 - Настройки рабочего пространства.

1. Открыть вкладку “Integrations” и выбрать “ClickUp API”.

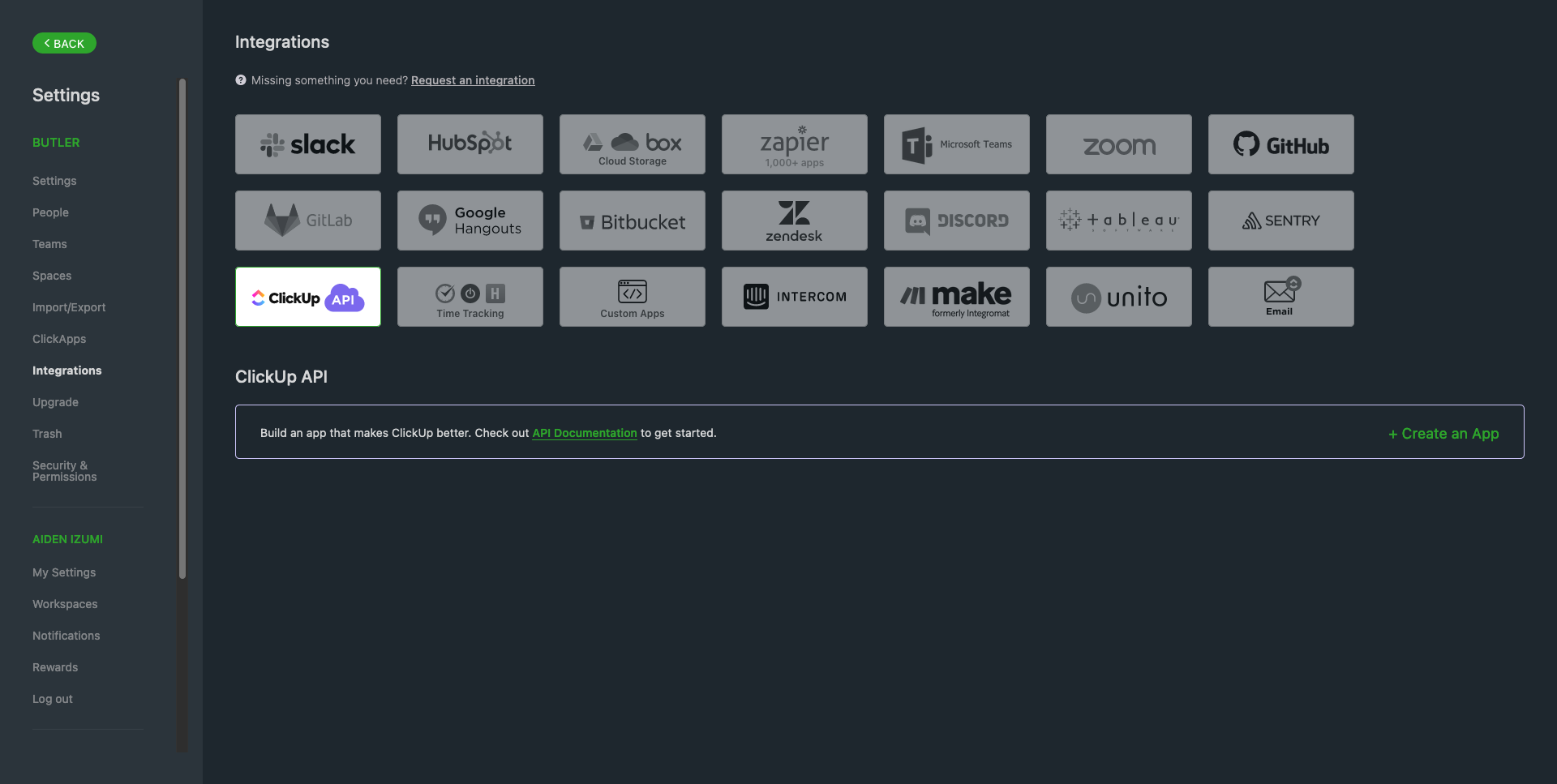


Рисунок 33 - Открытая вкладка “Integrations”.

1. Нажать на кнопку “Create an App”.

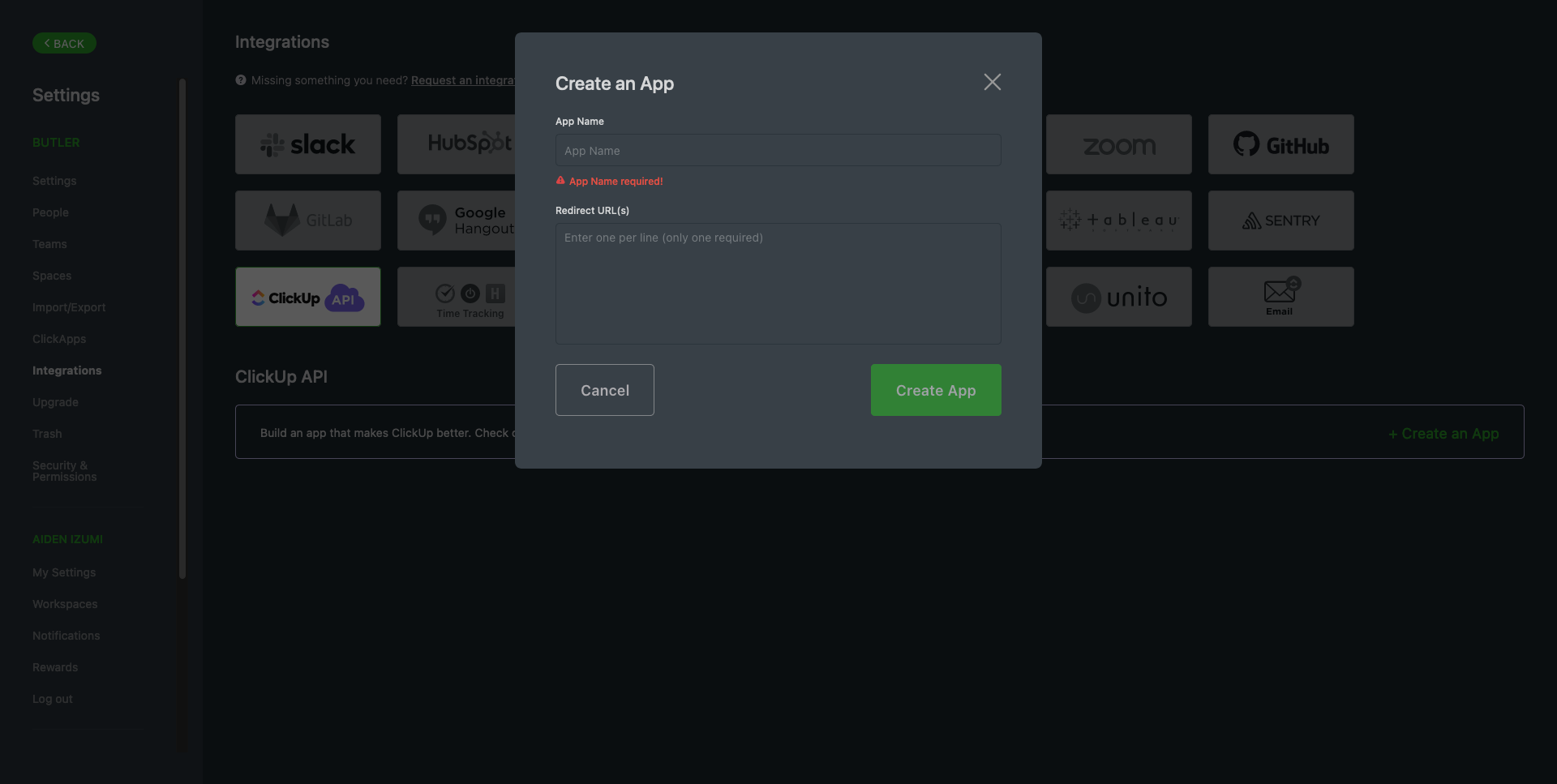


Рисунок 34 - Всплывающее окно создания приложения.

1. Заполнить форму валидными данными.

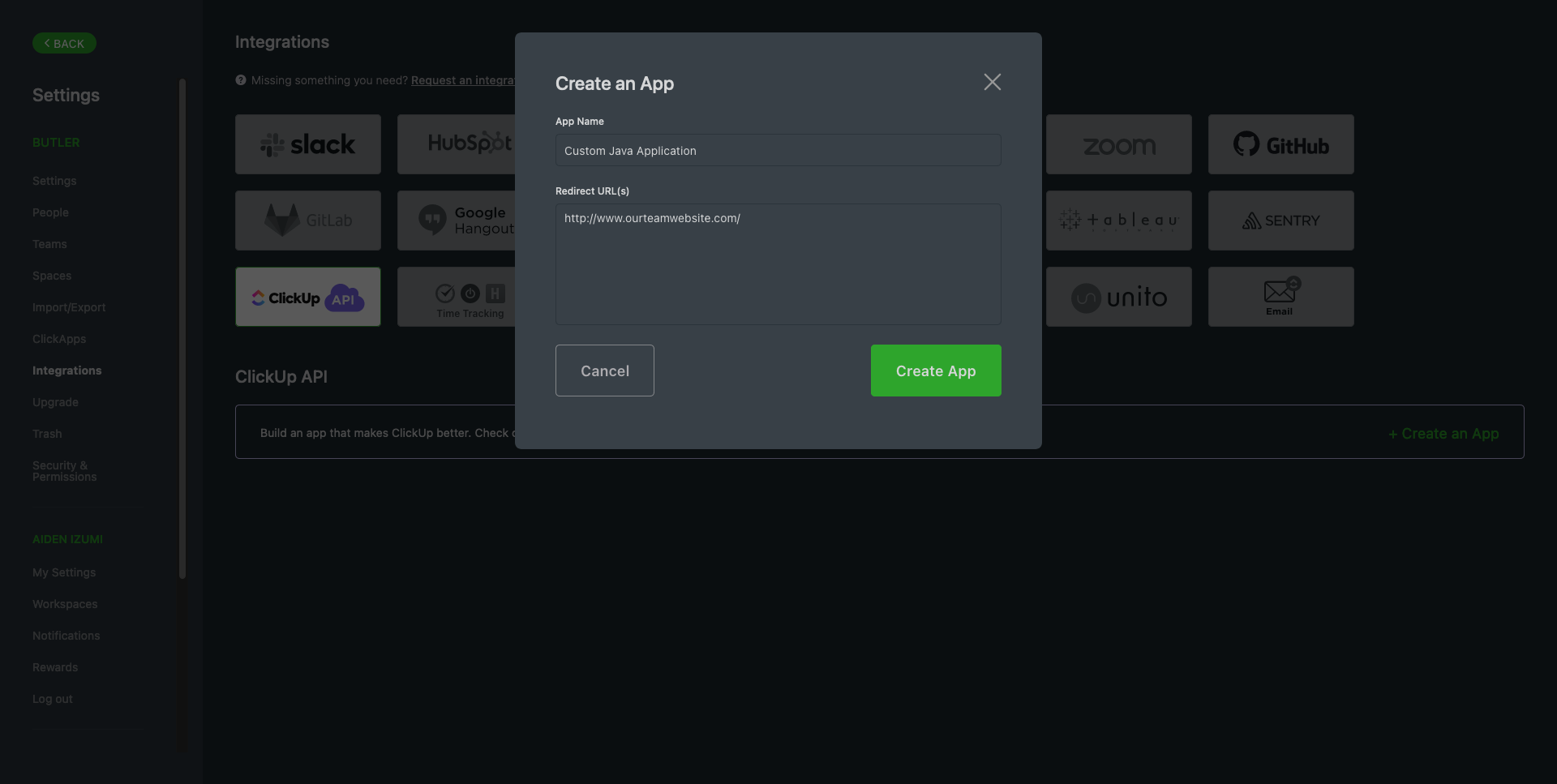


Рисунок 35 - Заполненная форма.

1. Нажать кнопку “Create App”. Сохранить полученные данные в безопасное место. Эти данные можно использовать согласно стандарту OAuth2.

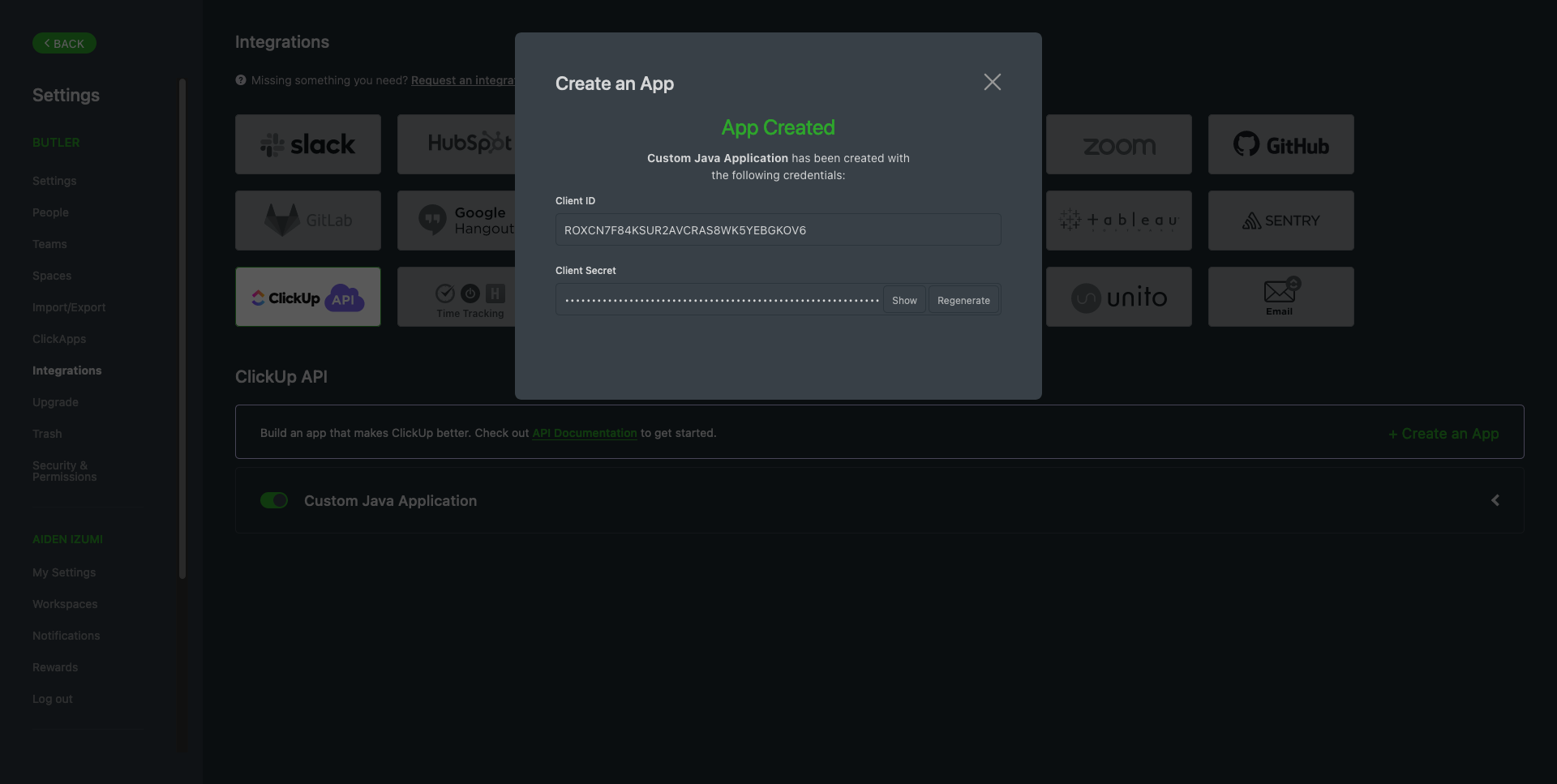


Рисунок 36 - Данные созданного приложения.

## Достоинства и недостатки:

### Достоинства:

* Доступны мобильная и веб-версии приложения.
* Присутствует возможность интеграции с собственным приложением, приложениями третьих лиц.
* Поддерживается импорт данных как из множества популярных аналогичных приложений, так и из файлов.
* Существуют чаты для общения с коллегами.
* Присутствуют различные графики, способствующие ведению и контролю над проектом.
* Доступен бесплатный уровень подписки, для персонального использования.
* В системе совмещены как функции управления задачами, так и работа с документами и общение с коллегами.
* Уровень настроек высок. Можно заводить собственные роли и выдавать настраиваемые разрешения.

### Недостатки:

* Система визуально отличается от аналогичных - при миграции потребуется время на привыкание, а при обучение - время на освоение.
* Поддерживается малое количество языков. Не поддерживается русский язык.
* При настройках по умолчанию в приложении много уведомлений.
* Производительность при формировании отчётов может быть ниже среднего.
* Функционал отслеживания времени урезан, по сравнению с аналогичными приложениями.

# Заключение

Все поставленные задачи были выполнены. В ходе работы проанализирован инструмент управления ИТ-проектами ClickUp. Подготовлен и описан учебный проект, аргументированно выбрана методология разработки.

В процессе выполнения задания освоены следующие умения и навыки:

* Работы с литературой;
* Анализа программной документации;
* Формирование логичного и связного описания проанализированных аспектов приложения;

Проведено исследование наиболее распространенных методологий управления ИТ-проектами. Описаны преимущества и недостатки, применимость. Сделаны выводы относительно использований конкретных методологий при тех или иных условиях.

# Список использованной литературы

1. “ClickUp Product Features”. [Электронный ресурс]. URL: <https://clickup.com/features>. (Дата обращения: 05.03.2023)
2. “Agile Manifesto for Software Development” [Электронный ресурс]. URL: <https://www.agilealliance.org/agile101/the-agile-manifesto/> (дата обращения: 12.03.2023).
3. Маркин, В. Ю. Эволюция методологии управления IT-проектами в современных экономических условиях / В. Ю. Маркин // . – 2020. – № 5-1(83). – С. 28-35. – EDN YTZWGF (дата обращения: 12.03.2023).
4. Баширова, М. М. Технология управления проектами и проектными командами на основе методологии гибкого управления проектами / М. М. Баширова // Наука: общество, экономика, право. – 2020. – № 2. – С. 178-183. – DOI 10.34755/IROK.2020.64.29.068. – EDN JGPZAF (дата обращения: 16.03.2023).
5. Гугаев Кирилл Валерьевич Границы применимости компонентов Scrum // Вестник евразийской науки. 2018. №3. URL: https://cyberleninka.ru/article/n/granitsy-primenimosti-komponentov-scrum (дата обращения: 18.03.2023).
6. Мокшин Владимир Васильевич, Гайнутдинова Алсу Маратовна, Самсонов Сергей Олегович СОВРЕМЕННЫЕ ПОДХОДЫ К ПРОЕКТАМ. МЕТОДОЛОГИИ. AGILE: SCRUM // StudNet. 2021. №6. URL: https://cyberleninka.ru/article/n/sovremennye-podhody-k-proektam-metodologii-agile-scrum (дата обращения: 20.03.2023).
7. Барулина В.В. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ГИБКОЙ И КАСКАДНОЙ МЕТОДОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ // Вестник магистратуры. 2020. №4-2 (103). URL: https://cyberleninka.ru/article/n/sravnitelnyy-analiz-gibkoy-i-kaskadnoy-metodologii-razrabotki-programmnogo-obespecheniya (дата обращения: 25.03.2023).
8. Svirskiy Oleg Vitalievich RISKS MITIGATION OF THE AGILE SOFTWARE DEVELOPMENT METHODOLOGIES IMPLEMENTATION // Научный журнал. 2021. №3 (58). URL: https://cyberleninka.ru/article/n/risks-mitigation-of-the-agile-software-development-methodologies-implementation (дата обращения: 30.03.2023).
9. Dr. Winston W. Royce MANAGING THE DEVELOPMENT OF LARGE SOFTWARE SYSTEMS // International Conference on Software Engineering (1987). URL: <https://web.archive.org/web/20160318002949/http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf> (дата обращения: 30.03.2023).