# **2 РАЗРАБОТКА ТРЕБОВАНИЙ**

## **2.1 Определение требований к программной системе**

Целью разработки дипломного проекта является создание web – приложения «Командный менеджер задач». Необходимо реализовать систему, позволяющую пользователям создать создавать задачи и управлять ими.

Для данной программной системы были определены следующие пользовательские требования:

1. Пользователь «OrgAdmin»:

- регистрация и авторизация;

- пригласить участника в организацию;

- просмотр организаций, досок организаций, листов, задач конкретной доски, а также активность по доске и конкретной задаче;

- редактирование организаций, досок организаций, листов и задач конкретной доски;

- перемещение задач внутри листов, между листами;

- перемещение листов;

2. Пользователь «OrgInvitedUser»:

- регистрация и авторизация;

- просмотр организаций, досок организаций, листов, задач конкретной доски, а также активность по доске и конкретной задаче;

- редактирование досок организаций, листов и задач конкретной доски;

- перемещение задач внутри листов, между листами;

- перемещение листов;

3. Пользователь «Guest»:

-регистрация и авторизация;

Требования к техническому обеспечению:

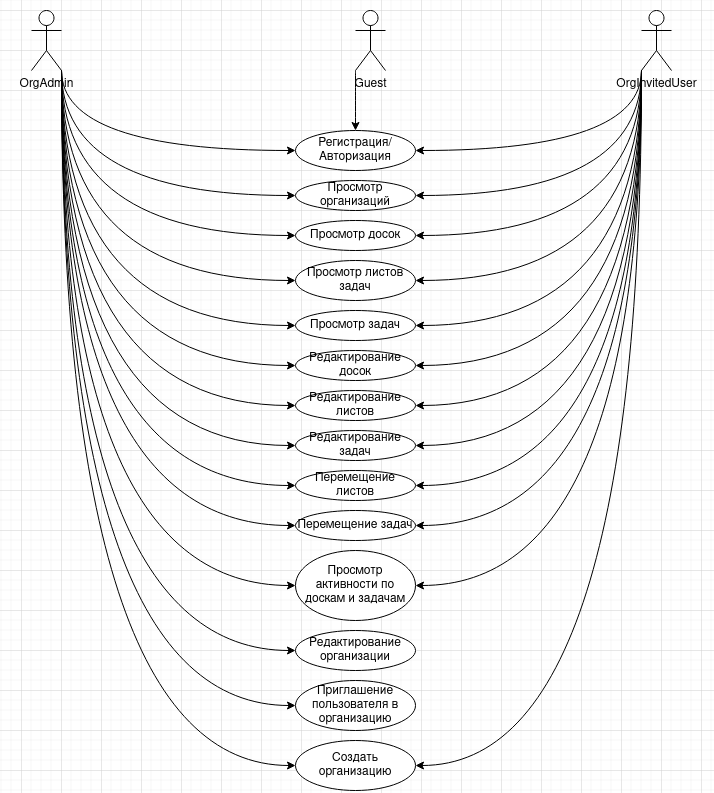
* процессор от 2 ГГц и выше 64 разрядный CPU;
* операционная система любая 64 битная ОС;
* ОЗУ от 2Гб;
* доступ в сеть интернет;
* наличие любого браузера.

Диаграмма вариантов использования (сценариев поведения, прецедентов) является исходным концептуальным представлением системы в процессе ее проектирования и разработки. Данная диаграмма состоит из актеров, вариантов использования и отношений между ними. При построении диаграммы могут использоваться также общие элементы нотации: примечания и механизмы расширения.

Суть данной диаграммы состоит в следующем: проектируемая система представляется в виде множества актеров, взаимодействующих с системой с помощью так называемых вариантов использования. При этом актером называется любой объект, субъект или система, взаимодействующая с моделируемой системой извне. В свою очередь вариант использования – это спецификация сервисов (функций), которые система предоставляет актеру. Другими словами, каждый вариант использования определяет некоторый набор действий, совершаемых системой при взаимодействии с актером. При этом в модели никак не отражается то, каким образом будет реализован этот набор действий.

Вариант использования обозначается на диаграмме эллипсом, внутри которого содержится его описание, обозначающее выполнение какой–либо операции или действия. Вариант использования, который инициализируется по запросу пользователя, представляет собой законченную последовательность действий.

Диаграмма вариантов использования программы представлена на рисунке 2.1.



**Рисунок 2.1 – Диаграмма вариантов использования web - приложеия «Командный менеджер задач»**

Система предполагает три варианта использования:

Гость;

Администратор организации;

Пользователь приглашенный в организацию.

Пользователь зарегистрировавшийся и создавший организацию получает доступ к правам администратора организации после авторизации. Администратор организации управляет организацией и всем остальным внутри организации: доски, листы задач, задачи, а также может создавать собственные организации и приглашать пользователей в организации.

Пользователь приглашенный в организацию имеет доступ к доскам организации, а так же ко всему, внутри досок: листы задач, задачи.

Для авторизации в системе используется облачное решение Clerk.

Clerk - это облачное решение, предназначенное для упрощения процесса аутентификации и авторизации пользователей в приложениях и сервисах, использующих протокол OAuth (внешние провайдеры учетных данных, например - Google). Clerk облегчает процесс реализации безопасной системы авторизации в приложениях, снижает сложность разработки и обеспечивает безопасность и удобство для пользователей. Оно может быть использовано разработчиками для быстрой и надежной реализации протокола OAuth в своих проектах. В данном проекте подключение к Clerk производится по средствам NextJs.

## **2.2** **Описание аналогов системы**

Для более полного определения набора функций, которые будет реализовывать разрабатываемая система, целесообразно рассмотреть уже существующие аналоги, а также выявить достоинства и недостатки каждого из них.

Первым аналогом является программное средство «Trello» помогает организовывать работу в команде и управлять проектами и задачами.

Преимуществом «Trello» является:

– простота использования;

– гибкость и адаптивность;

– интеграция с другими инструментами (Slack, Google, Dropbox);

– бесплатная версия приложения доступна всем, но с ограничениями.

Скриншот программы представлен на рисунке 2.2.

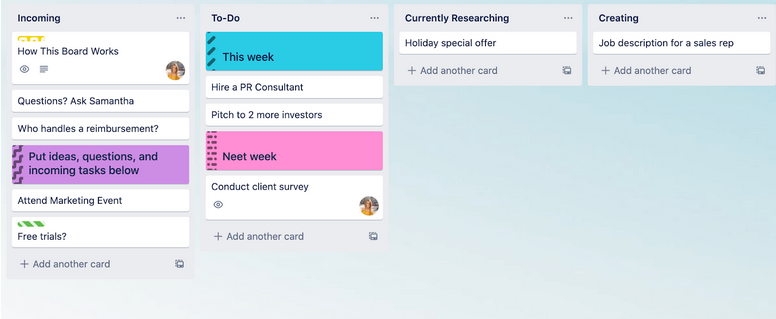


Рисунок 2.2 – «Окно задач «Trello»

Вторым аналогом является программа «WEEEK» также позволяет заниматься менеджментом задач команды, но, помимо этого предоставляет возможность вести документацию, необходимую команде, а также другие возможности.

Преимуществом «WEEEK» является:

– большой спектр возможностей: управление задачами, календарь, чат;

– гибкий календарь;

– удобная коммуникация внутри команды;

– бесплатная версия предоставляет большой спектр возможностей, но имеются ограничения по количеству пользователей.

Скриншот программы представлен на рисунке 2.4

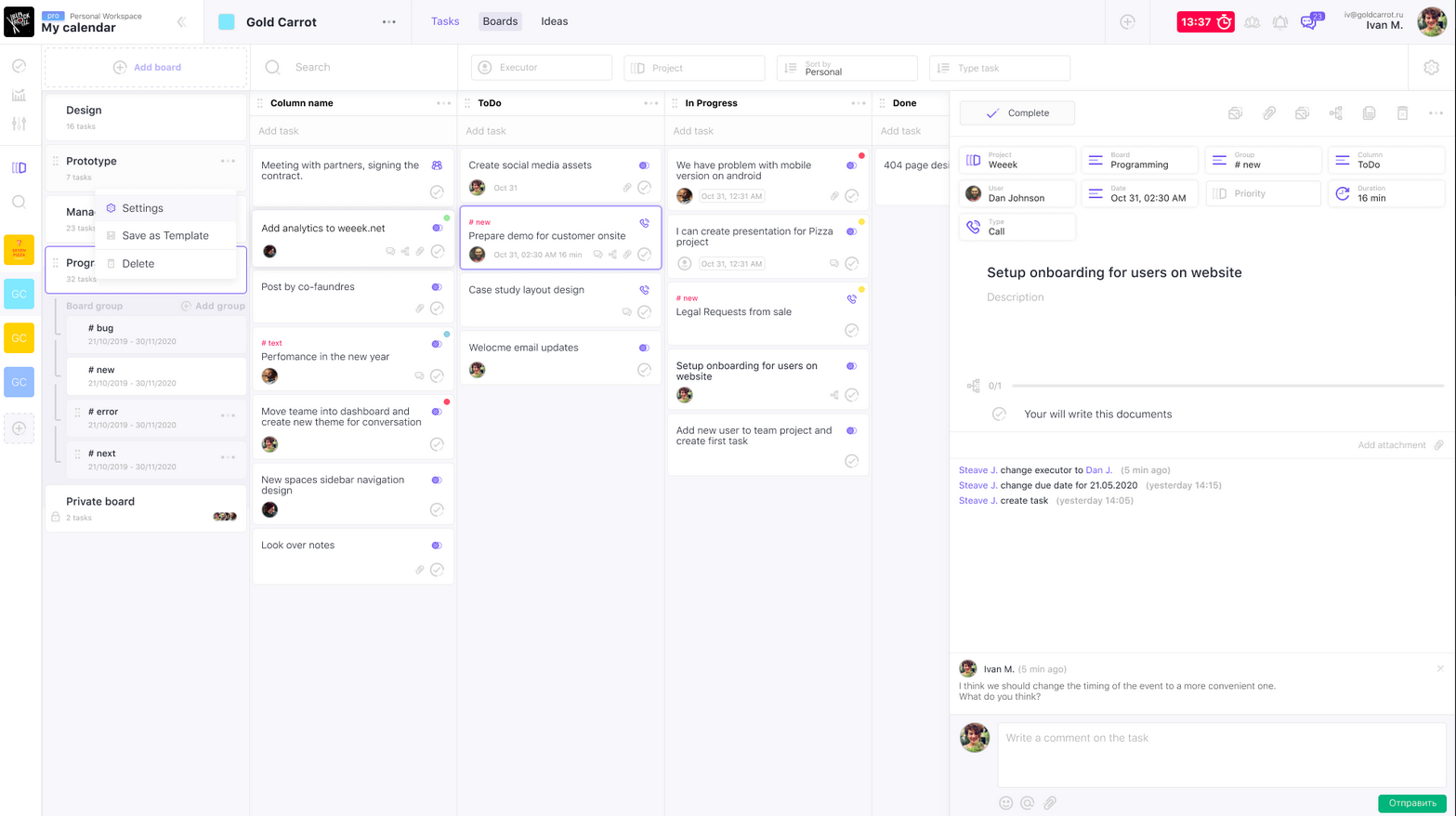


Рисунок 2.4 – «Рабочее окно программы «WEEEK»»