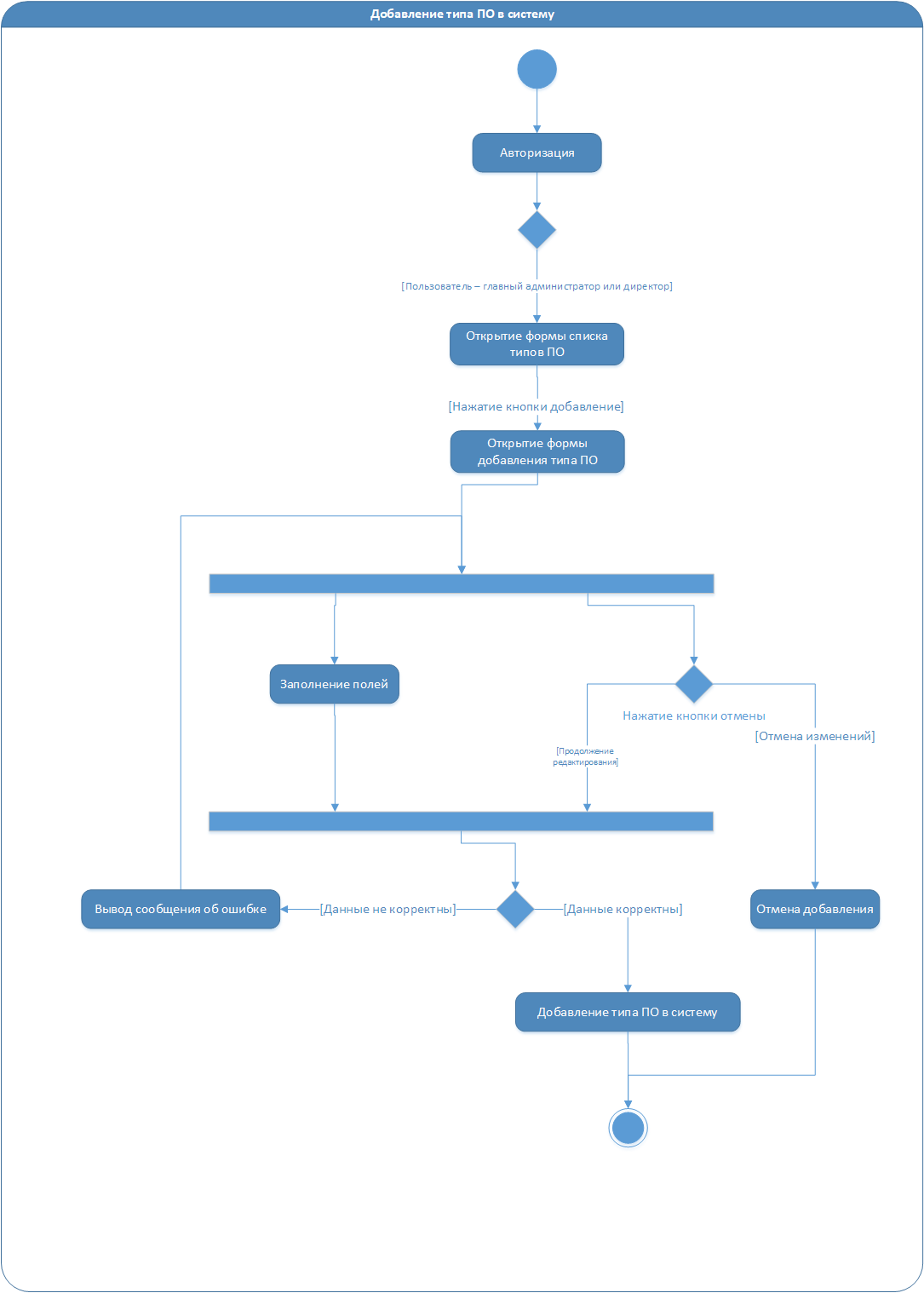
Лабораторная работа №5

# Диаграммы действий

## Добавление типа ПО в систему

Диаграмма описывает деятельность главного администратора и директора. После авторизации пользователь может открыть форму просмотра списка типов ПО и нажать на кнопку добавления, после чего будет произведён переход к форме добавления типа ПО. В процессе редактирования пользователь может заполнить поля, либо отменить все вносимые изменения, после чего процесс завершается без добавления типа ПО. После успешного завершения внесения изменений производится проверка правильности введённых данных, в случае некорректного ввода пользователю предлагается скорректировать некорректные значения.

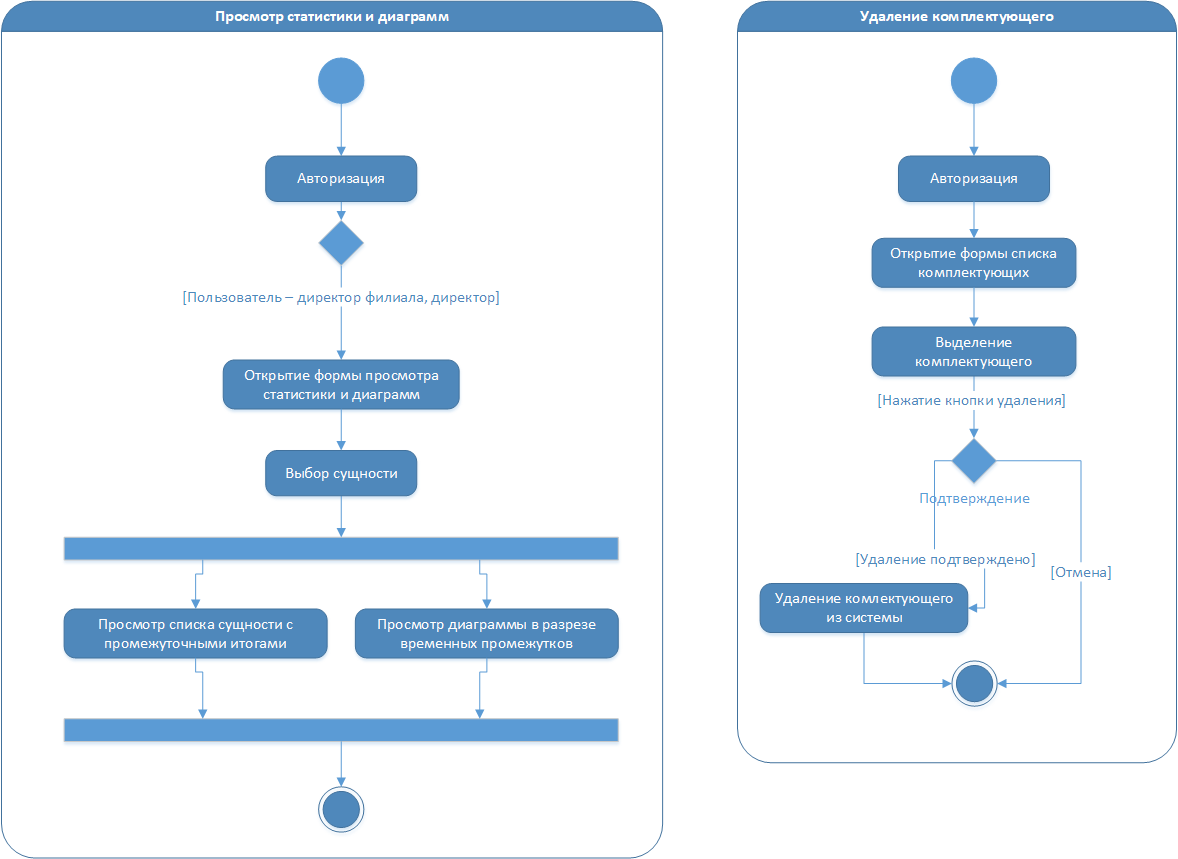


## Удаление комплектующего

Диаграмма описывает деятельность любого участника системы. После авторизации пользователь может открыть форму просмотра списка комплектующих, выбрать комплектующее для удаления и нажать на кнопку «Удалить», затем после подтверждения операции комплектующее удаляется из системы.

## Просмотр статистики и аналитики

Диаграмма описывает деятельность директора и директора филиала. После авторизации для директора выводится пользовательский интерфейс, где он может выбрать нужную сущность для просмотра графиков и статистических данных. После выбора сущности директор может осуществить просмотр списка сущности с промежуточными итогами либо просмотр диаграммы в разрезе временных промежутков.

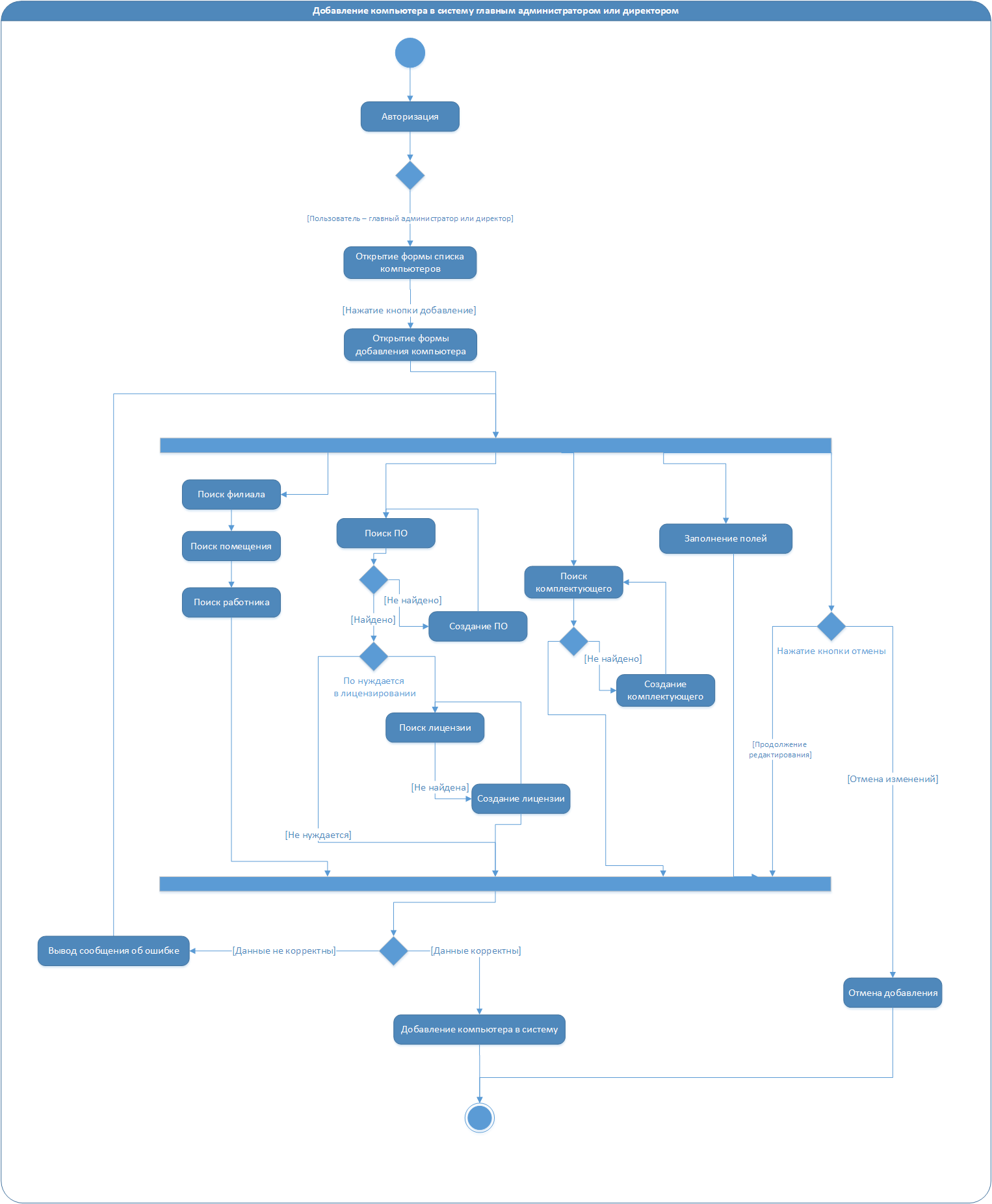


## Добавление компьютера в систему главным администратором или директором

Диаграмма описывает деятельность главного администратора и директора. После авторизации пользователь может открыть форму просмотра списка компьютеров и нажать на кнопку добавления, после чего будет произведён переход к форме добавления компьютера. В процессе редактирования пользователь может заполнить поля, либо отменить все вносимые изменения, после чего процесс завершается без добавления компьютера.

Также в процессе редактирования пользователь может совершить выбор филиала, помещения, ответственного лица, произвести внесение информации об установленном ПО (создать информацию о, в случае отсутствия необходимого) и применить лицензии к данному ПО (создать информацию о, в случае отсутствия необходимых), произвести внесение информации о установленных в компьютер комплектующих (создать информацию о, в случае отсутствия необходимых).

После успешного завершения внесения изменений производится проверка правильности введённых данных, в случае некорректного ввода пользователю предлагается скорректировать некорректные значения.

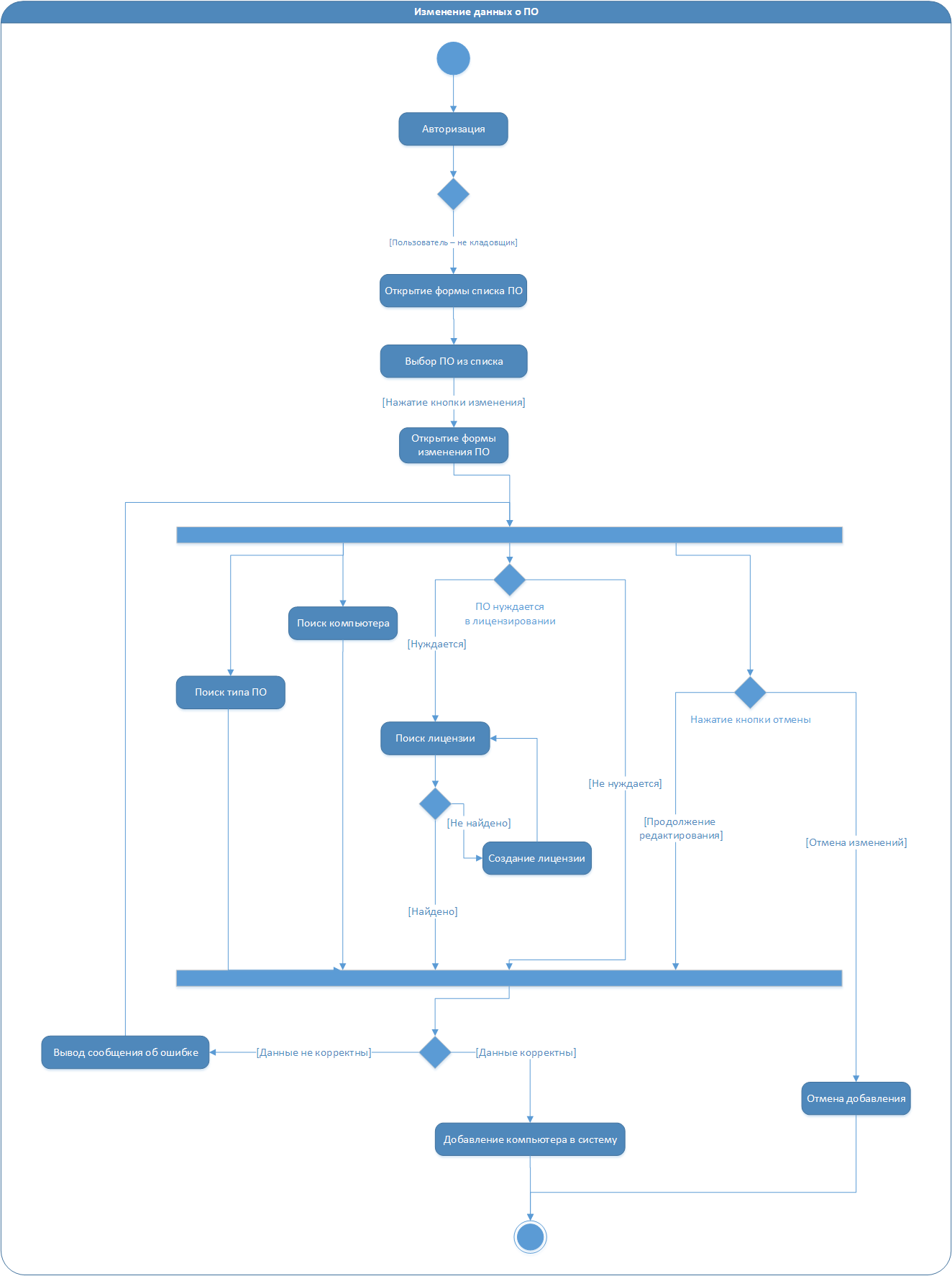


## Изменение данных о ПО

Диаграмма описывает деятельность пользователей, не являющихся кладовщиками. После авторизации пользователь может открыть форму просмотра списка ПО, произвести выбор ПО из списка и нажать на кнопку изменения, после чего будет произведён переход к форме изменения ПО. В процессе редактирования пользователь может заполнить поля, либо отменить все вносимые изменения, после чего процесс завершается без изменения данных о ПО.

Также в процессе редактирования пользователь может совершить выбор типа ПО, компьютера, на который будет установлено ПО и применить лицензии к данному ПО (создать информацию о, в случае отсутствия необходимых), если ПО нуждается в лицензировании.

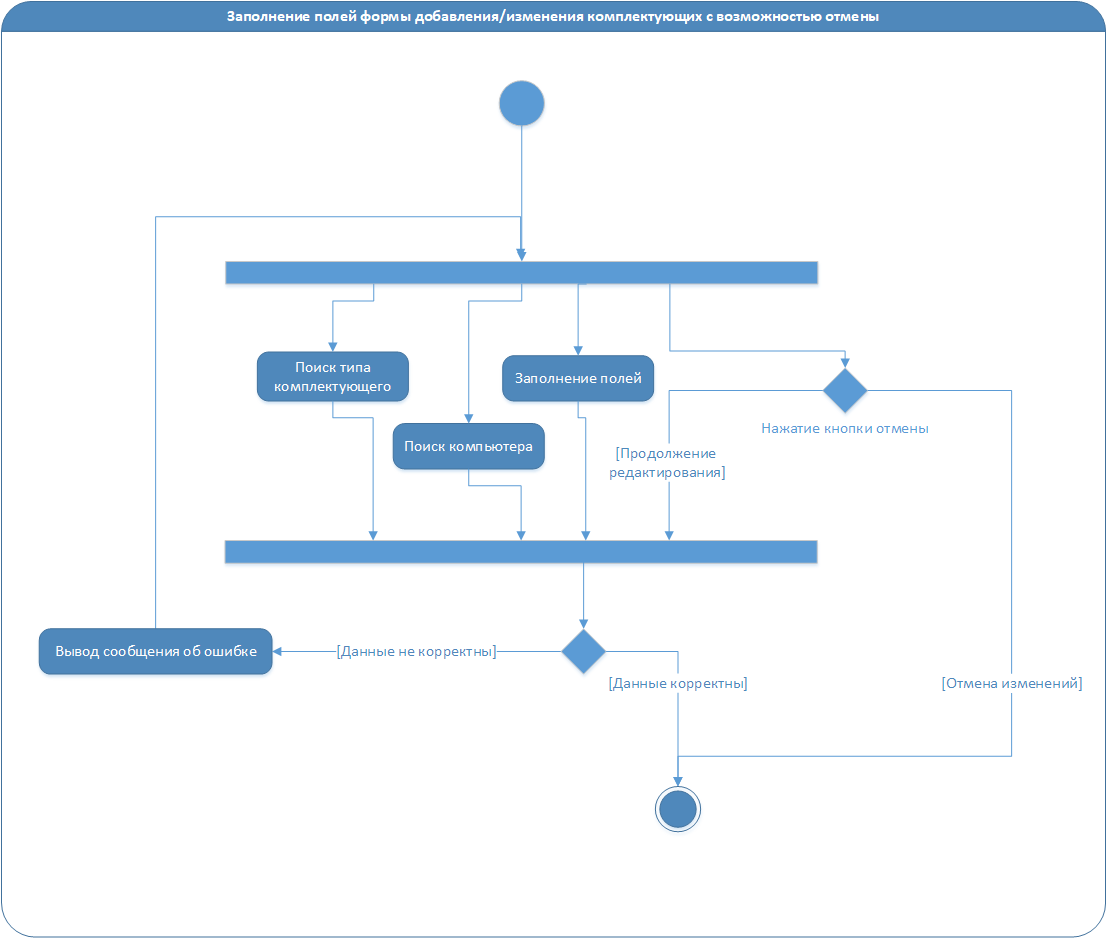
После успешного завершения внесения изменений производится проверка правильности введённых данных, в случае некорректного ввода пользователю предлагается скорректировать некорректные значения.

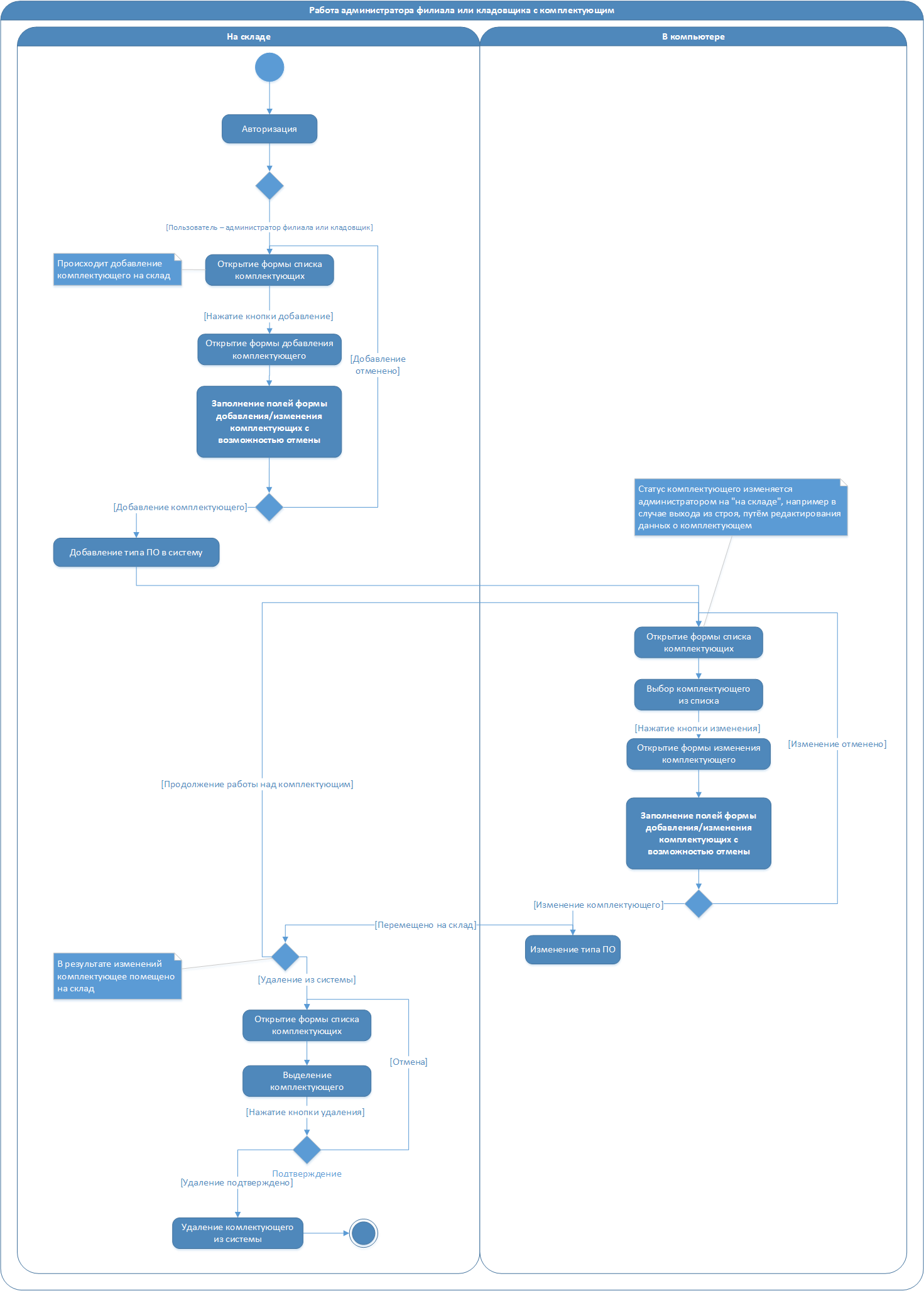


## Работа администратора филиала или кладовщика с комплектующим

Диаграмма описывает деятельность администратора филиала или кладовщика с комплектующим. Произведена декомпозиция, в результате которого было выделено действие «Заполнение полей формы добавления/изменения комплектующих» с возможностью отмена. Диаграмма данного действия описывает взаимодействие пользователя с формой добавления или изменения комплектующего (заполнение полей, выбор типа комплектующего, поиск компьютера, отмена добавления или изменения).

Главная диаграмма разделена на две плавательные дорожки, показывающих местонахождение комплектующего (на складе, либо в компьютере). В начале процесса происходит добавление комплектующего в систему и установка его в компьютер. Затем в случае необходимости возврата комплектующего на склад (например, в случае его выхода из строя) происходит изменение комплектующего пользователем. Во время нахождения на складе пользователь может произвести списание (удаление) комплектующего из системы.

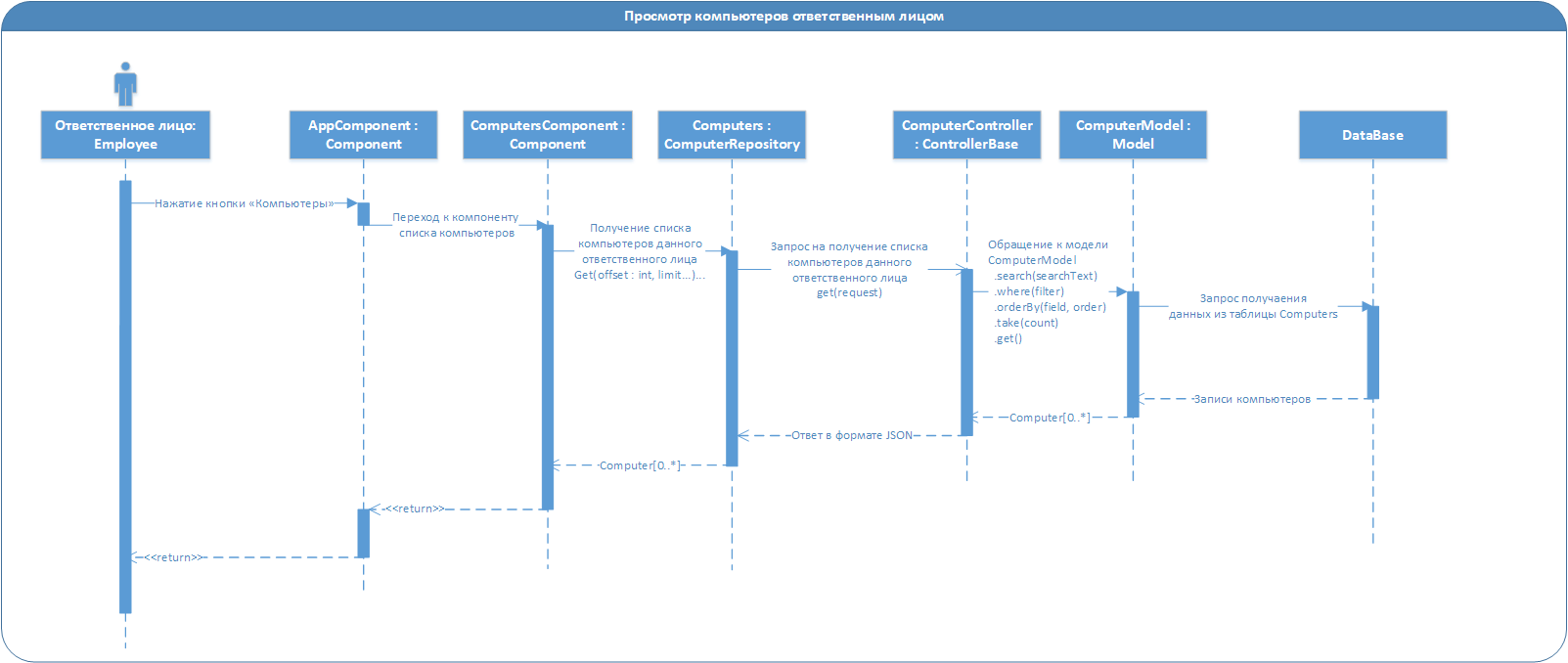




# Диаграммы последовательностей

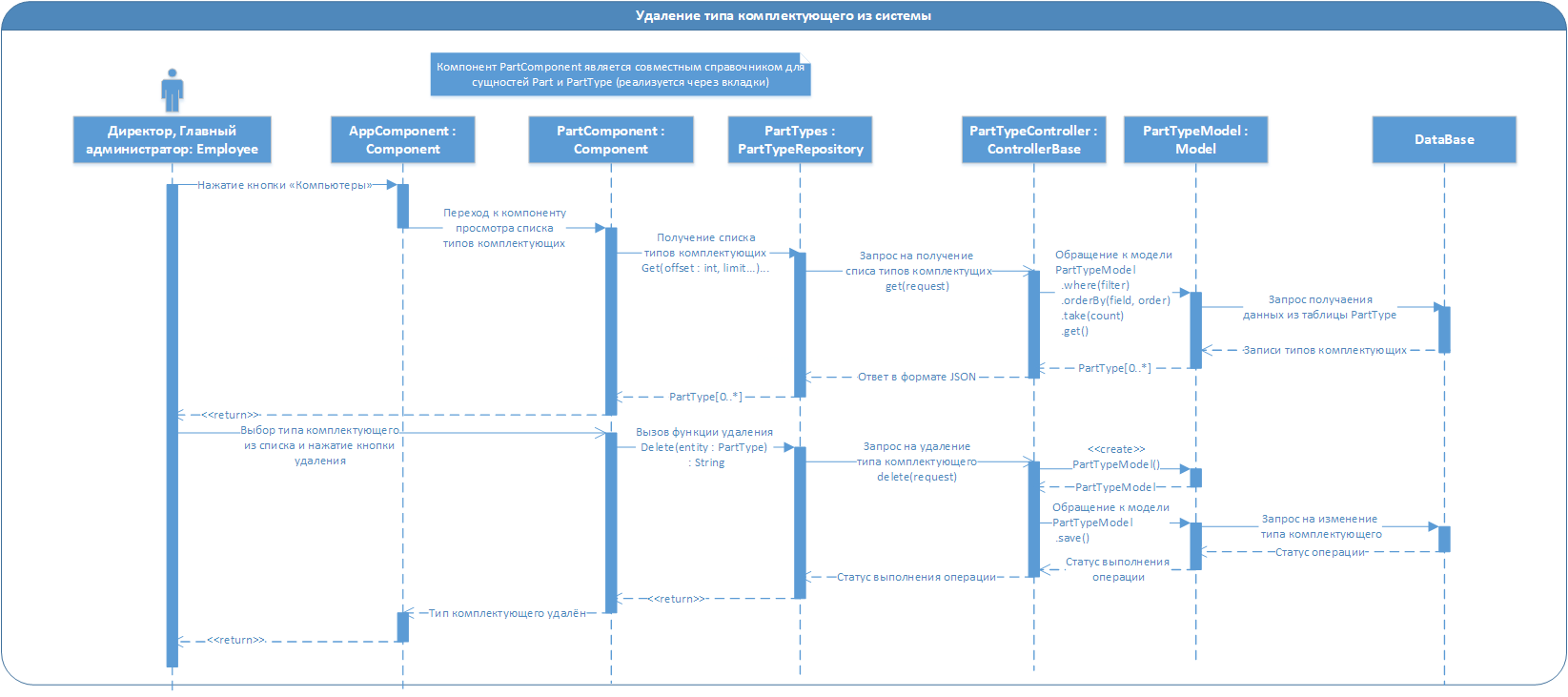
## Просмотр компьютеров ответственным лицом

Диаграмма описывает процесс просмотра компьютеров ответственным лицом.



## Удаление типа комплектующего из системы

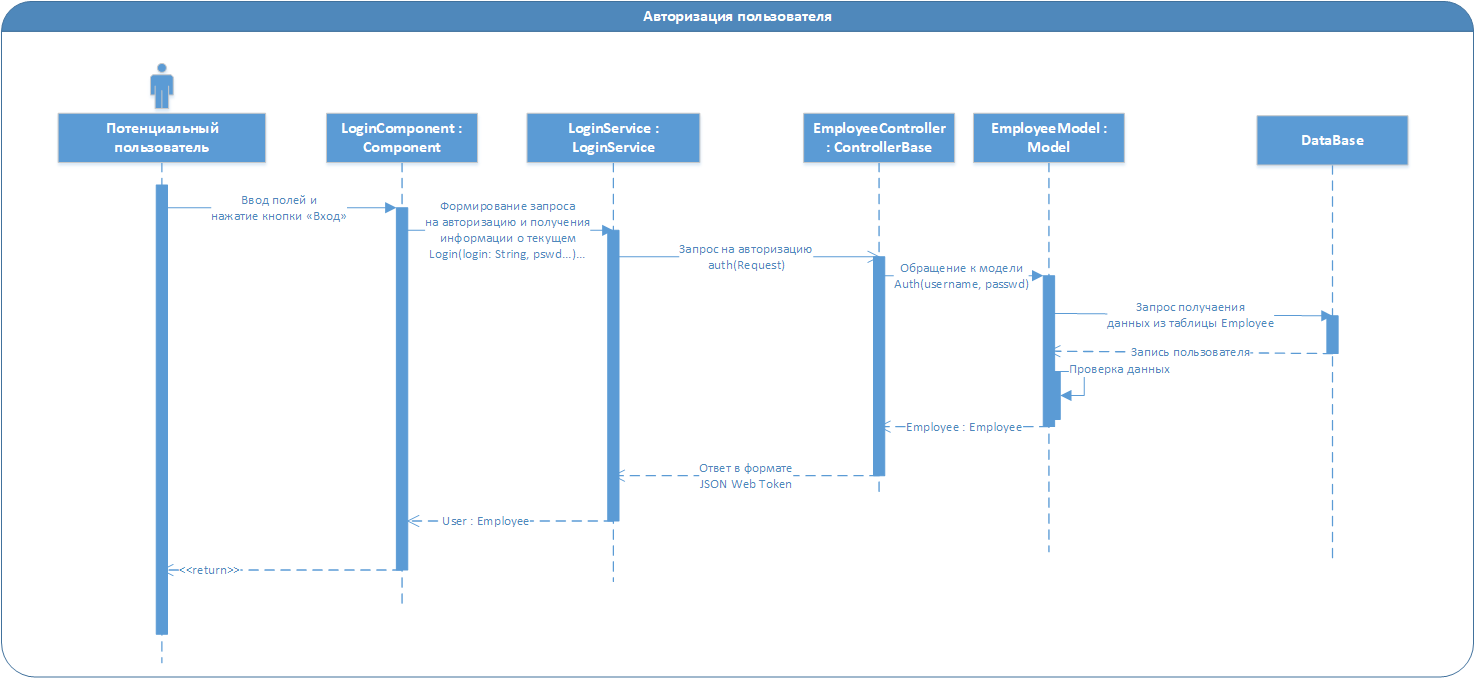
Диаграмма описывает процесс удаления типа комплектующего из системы. Директор или главный администратор переходят к компоненту справочника комплектующих и типов комплектующих для произведения выбора из списка.



## Авторизация пользователя

Диаграмма описывает процесс авторизации пользователя. Потенциальный пользователь нажимает на форме авторизации кнопку «Вход». Затем производится вызов функции auth сервиса авторизации для произведения запроса авторизации. (LoginService).

Далее происходит принятие запроса сервером, и соответствующие вызовы метода auth контроллера, который обращаются к модели (вызов функции Auth) для проверки существования данного пользователя и корректности введённых авторизационных данных. Если пользователь найден и все данные корректны, то вход выполнен успешно и на клиент возвращается JSON Web Token с данными о пользователе.



## Добавление компьютера директором филиала или администратором филиала

Диаграмма описывает процесс добавление компьютера директором филиала или администратором филиала. Пользователь, находясь в компоненте справочника компьютеров нажимает кнопку «Добавать». Производится переход к компоненту добавления компьютера. В процессе перехода компонентом создаётся новый объект класса Computer

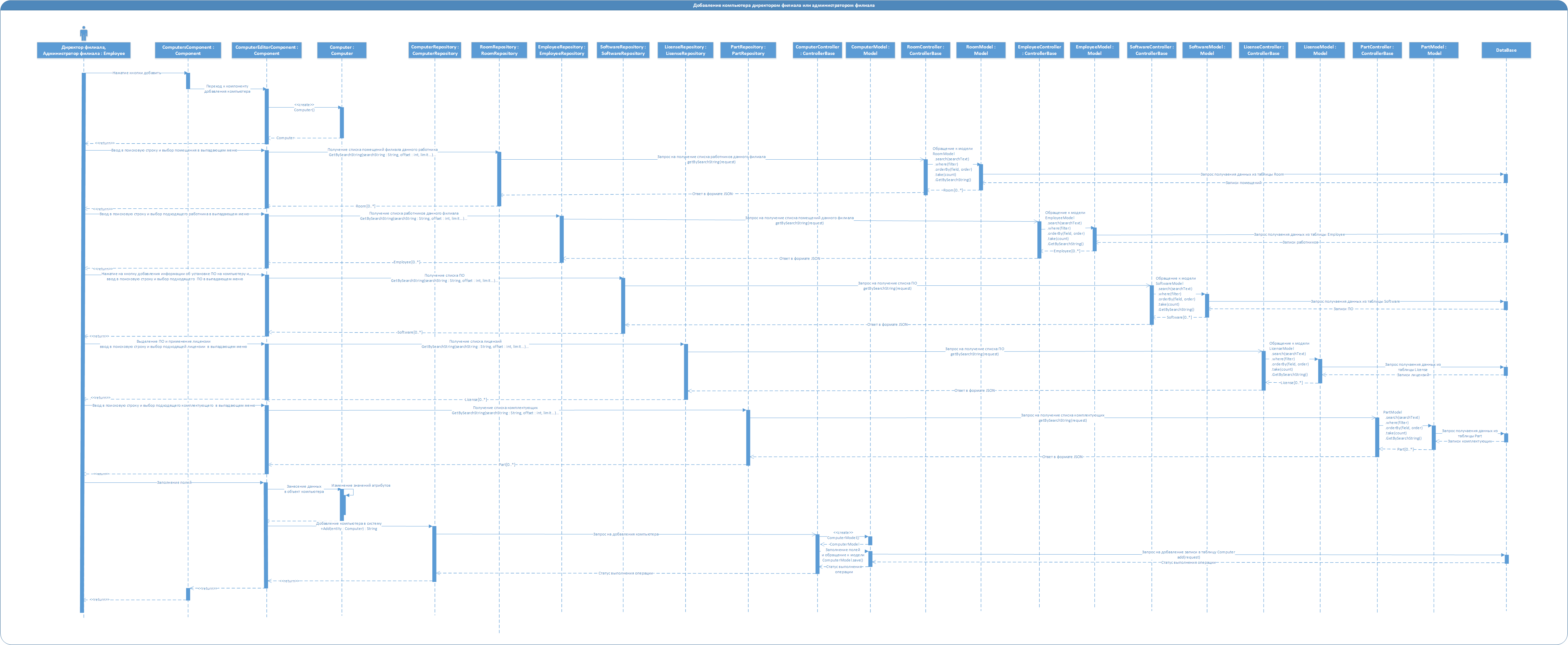
После выполнения всех этих операций пользователь может приступить к редактированию компьютера. При помещении и ответственного лица пользователь производит выбор путём ввода в соответствующую поисковую строку и выбора подходящего экземпляра соответствующей сущности в выпадающем меню. При задании информации об установленном ПО (применённых к ПО лицензий) и комплектующих действует тот же принципе с той лишь разницей, что пользователь может привязывать несколько сущностей (комплектующих, ПО и их лицензий) к компьютеру, поэтому перед выбором конкретного экземпляра пользователь нажимает на кнопку добавления информации, и лишь затем производит выбор, соответственно этот процесс может повторяться столько раз, пока не будут внесена информация о всех установках ПО, комплектующих и применённых лицензиях. При изменении поисковой строки производится получения списка соответствующей сущности.

При получении списка соответствующей сущности компонент производит обращение к сервису соответствующей сущности вызовом функции GetBySearchString(searchString : String ,offset: int, limit: int…) для формирования запроса на получения списка сущности по поисковой строке. (frontend)

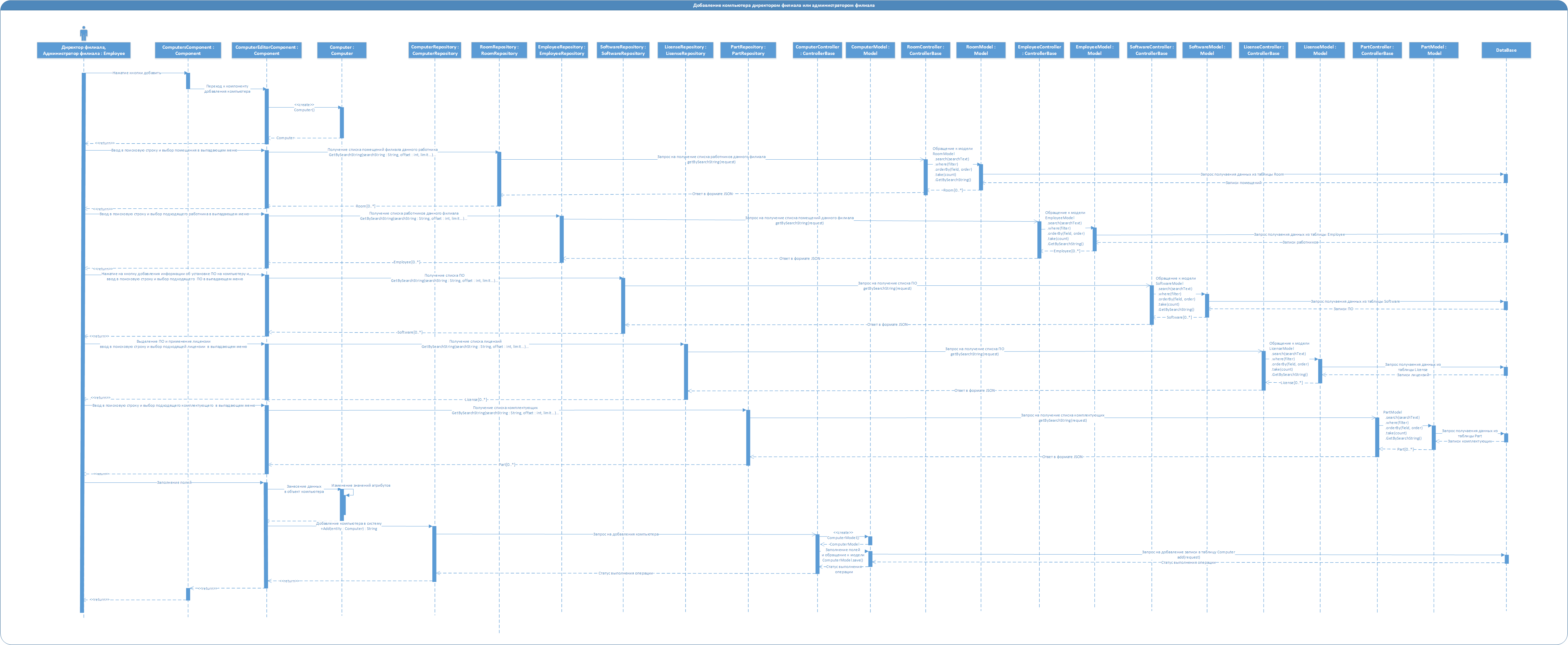
Далее происходит принятие запроса сервером, и соответствующие вызовы метода getBySearchString контроллера, который обращаются к модели ORM (вызов функции цепочки функций where(filter).orderby(field, order),,,) для построения запроса к базе данных на получение записей сущности. После выполнения запроса и возврата множества сущностей моделью в контроллере происходит сериализация данного множества и отправка клиенту. На клиенте происходит десериализация, построение списка сущностей и возврат управления пользователю.

После завершения редактирования происходит занесение изменений в объект компьютера. Далее производится операция внесения изменений в компонент производит обращение к ComputerRepository вызовом функции add(entity : Computer) для формирования запроса на добавление компьютера. (frontend)

Далее происходит принятие запроса сервером, и соответствующие вызовы метода add(request) контроллера, который создаёт экземпляр модели, заполняет его поля и вызывает функцию save() для построения запроса к базе данных на добавление записи компьютера. После выполнения запроса производится возврат статуса клиенту и завершение работы формы добавления. Пользователь может продолжить своё взаимодействие с системой.



*Добавление компьютера директором филиала или администратором филиала (Frontend)*



*Добавление компьютера директором филиала или администратором филиала (Backend)*

## Изменение ПО директором филиала или администратором филиала

Диаграмма описывает процесс изменения ПО директором филиала или администратором филиала. Пользователь переходит к компоненту справочника ПО для выбора ПО для редактирования. Компонент производит обращение к SoftwareRepository вызовом функции Get(offset: int, limit: int…) для формирования запроса на получения списка ПО. (frontend)

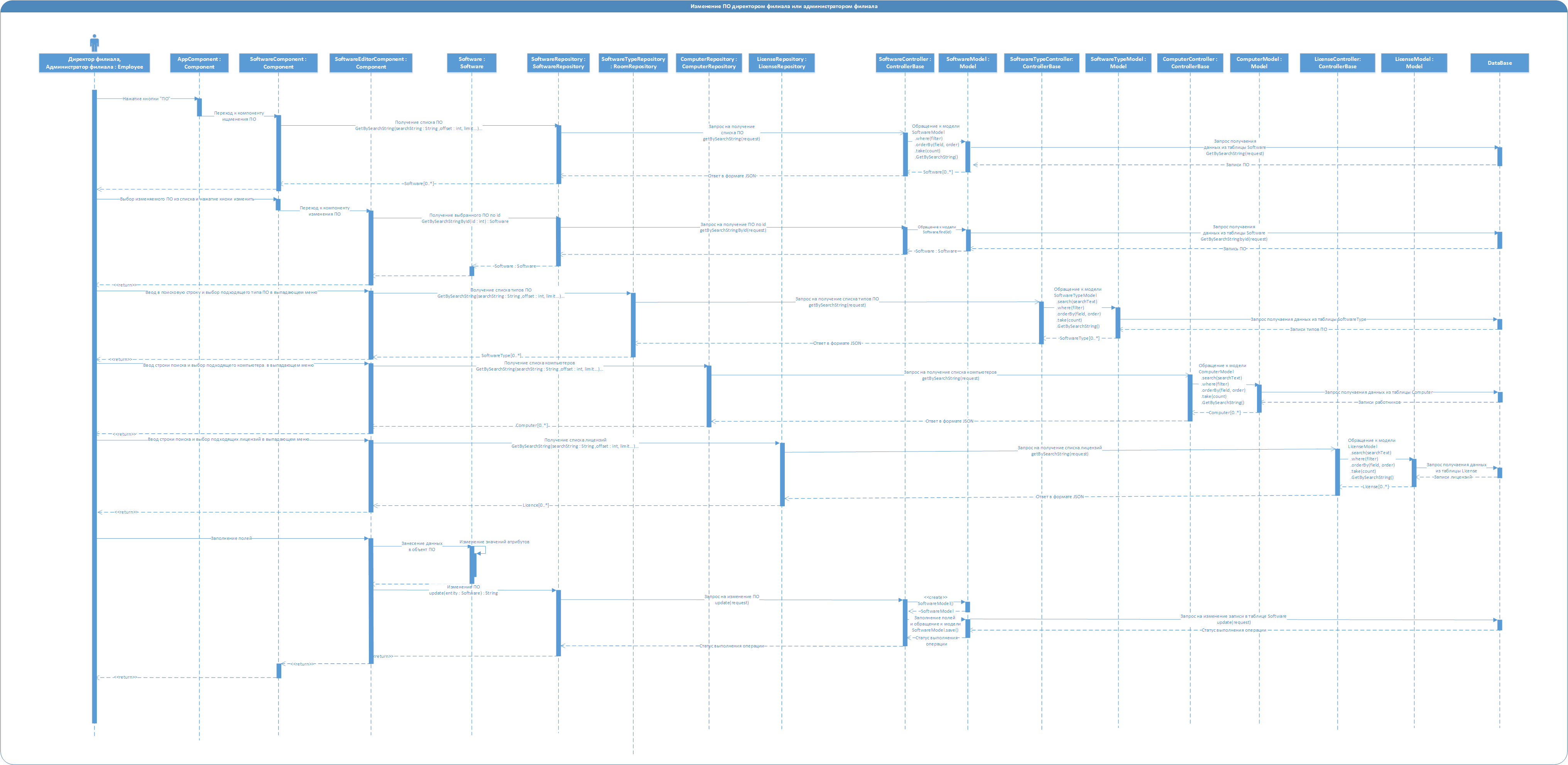
Далее происходит принятие запроса сервером, и соответствующие вызовы метода get контроллера, который обращаются к модели ORM (вызов функции цепочки функций where(filter).orderby(field, order),,,) для построения запроса к базе данных на получение записей ПО. После выполнения запроса и возврата множества ПО моделью в контроллере происходит сериализация данного множества и отправка клиенту. На клиенте происходит десериализация, построение списка ПО и возврат управления пользователю.

После выполнения данных операций пользователь может выбрать ПО для изменения и нажать кнопку «Изменить», что произведёт переход к форме изменения ПО. В процессе перехода необходимо получить изменяемое ПО по Id. Происходит последовательность операций, похожая на получение списка ПО, за исключением того, что производятся вызовы функций getById(id) (заместо get), а при обращении к модели ORM вызов функции find(id).

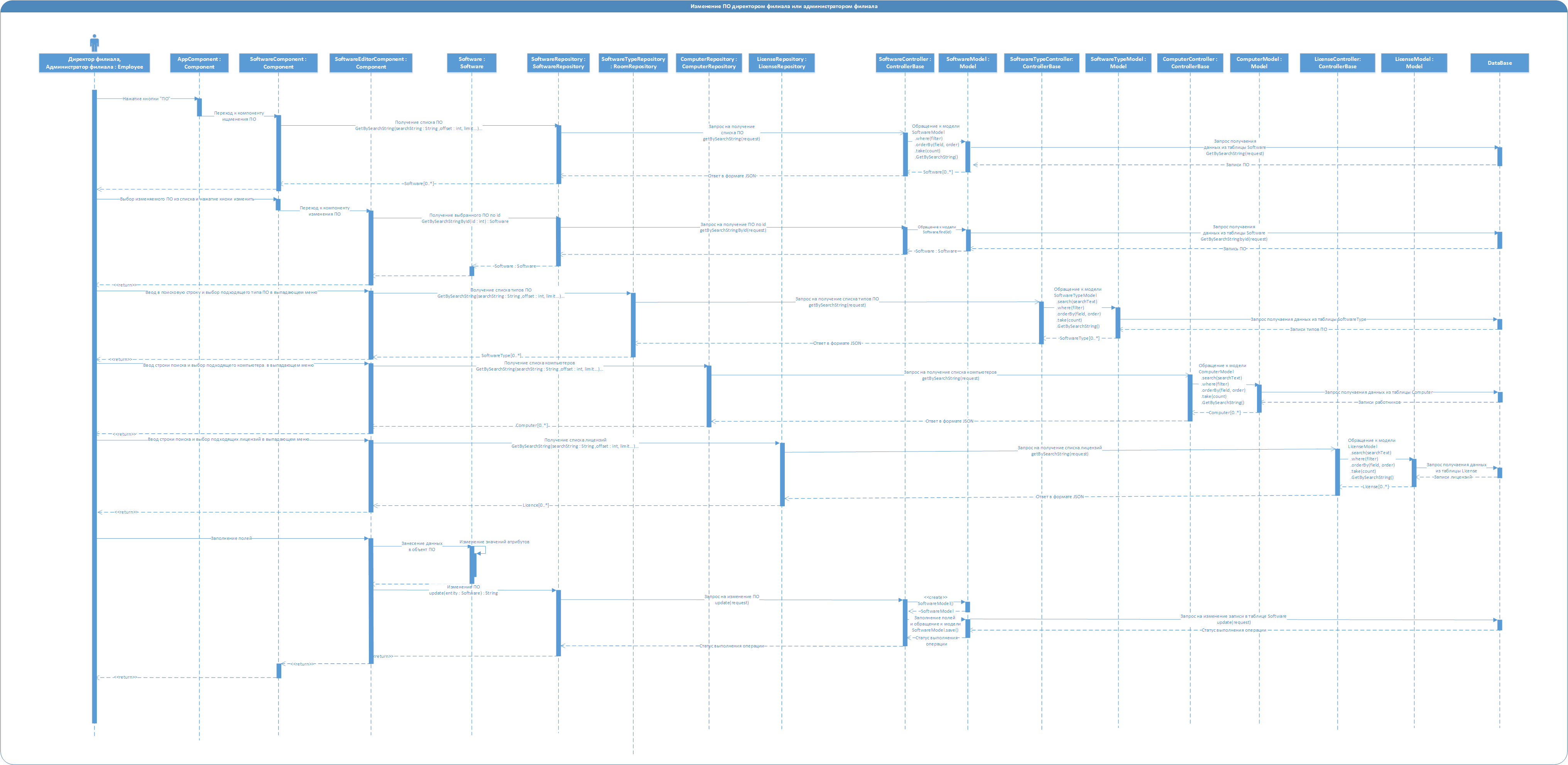
После выполнения всех этих операций пользователь может приступить к редактированию ПО. При задании, лицензии, компьютера и типа ПО пользователь производит выбор путём ввода в соответствующую поисковую строку и выбора подходящего экземпляра соответствующей сущности в выпадающем меню. При изменении поисковой строки производится получения списка соответствующей сущности по соответствующей поисковой строке аналогичный получению списка ПО, за исключением того, что производятся вызовы функций getBySearchString(searchString: string, offset : int, limit : int…), а при обращении к модели ORM вызов в цепочку добавляется вызов функции search(searchString).

После завершения редактирования происходит занесение изменений в объект ПО. Далее производится операция внесения изменений в компонент производит обращение к SoftwareRepository вызовом функции update(entity : Software) для формирования запроса на изменение ПО. (frontend)

Далее происходит принятие запроса сервером, и соответствующие вызовы метода update (request) контроллера, который создаёт экземпляр модели, заполняет его поля и вызывает функцию save() для построения запроса к базе данных на изменение записи ПО. После выполнения запроса производится возврат статуса клиенту и завершение работы формы редактирования. Пользователь может продолжить своё взаимодействие с системой.



*Изменение ПО директором филиала или администратором филиала (Frontend)*

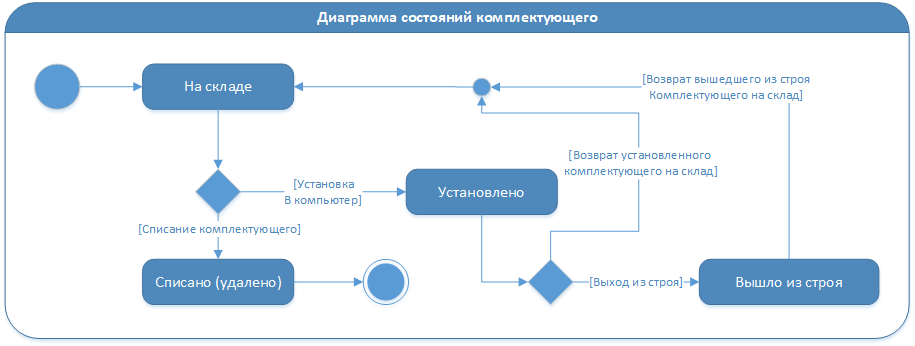


*Изменение ПО директором филиала или администратором филиала (Backend)*

# Диаграммы состояний

## Состояния комплектующего

Диаграмма описывает все возможные состояния комплектующего на продолжении его жизненного цикла. При добавлении комплектующего оно помещается на склад. Далее, в зависимости от выбора оно может быть установлено в компьютер или удалено из системы. Установленное комплектующее может быть извлечено и возвращено на склад, либо выйти из строя, и так же быть возвращённым на склад. Вышедшее из строя комплектующее, находящееся на складе так же может быть удалено из системы.



## Состояния лицензии

Диаграмма описывает все возможные состояния лицензии на продолжении ее жизненного цикла. После добавления лицензии происходит полное или частичное применение лицензии к некоторым ПО. Далее и в случае полностью применённой и в случае частично применённой лицензии происходит израсходование части всех применений данной лицензии. Частично применённая и частично израсходованная лицензия может быть до применена до состояния полностью применённой и частично израсходованной лицензии. Полностью применённая и частично израсходованная лицензия после прошествии некоторого времени становится полностью израсходованной.

