

# Описание прецедентов

## Операции над компьютерами

### Описание прецедента «Просмотр списка компьютеров»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка компьютеров |
| Краткое описание | Прецедент позволяет просматривать список компьютеров. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственное лицо. |
| Предусловия | Должен быть хотя бы один компьютер. Если таковых нет, то пользователю предлагается создать новый компьютер. |
| Основной поток | Выводится информация обо всех компьютерах в виде таблицы |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |
| Примечания к использованию | Акторы: Директор филиала, Администратор филиала могут просматривать компьютеры только своего филиала. Актор «Ответственное лицо» может просматривать только те компьютеры, за которые он ответственен. |

### Описание прецедента «Сортировка списка компьютеров»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка компьютеров |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список компьютеров. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственное лицо. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка компьютеров». |
| Основной поток | Производится сортировка списка компьютеров. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка компьютеров»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка компьютеров |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список компьютеров. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка компьютеров». |
| Основной поток | Производится фильтрация список компьютеров. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о компьютере»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о компьютере |
| Краткое описание | Прецедент позволяет просматривать информацию о конкретном компьютере. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственное лицо. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка компьютеров», «Просмотр списка ПО», «Просмотр списка комплектующих», «Просмотр списка лицензий», «Просмотр списка Работников» |
| Основной поток | Выводится информация о конкретном компьютере. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление компьютера в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление компьютера |
| Краткое описание | Прецедент позволяет добавлять новый компьютер. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка компьютеров», «Просмотр списка ПО», «Просмотр списка комплектующих», «Просмотр списка лицензий», «Просмотр списка Работников». |
| Основной поток | Открывается форма добавления компьютера, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новый компьютера. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |
| Примечания к использованию | Акторы: Директор филиала, Администратор филиала могут выбрать комплектующие и помещения только своего филиала. |

### Описание прецедента «Изменение данных о компьютере»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о компьютере |
| Краткое описание | Прецедент позволяет редактировать данные о компьютере. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка компьютеров», «Просмотр списка ПО», «Просмотр списка комплектующих», «Просмотр списка лицензий», «Просмотр списка Работников»: |
| Основной поток | Открывается форма редактирования компьютера, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования компьютера закрывается, и измененные данные о компьютере сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |
| Примечания к использованию | Акторы: Директор филиала, Администратор филиала могут выбрать комплектующие и помещения только своего филиала. |

### Описание прецедента «Удаление компьютера из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление компьютера из списка |
| Краткое описание | Прецедент позволяет удалять компьютер. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка компьютеров». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то компьютер удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над лицензиями

### Описание прецедента «Просмотр списка лицензий»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка лицензий |
| Краткое описание | Прецедент позволяет просматривать список всех лицензий. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должна быть хотя бы одна лицензия. Если таковых нет, то пользователю предлагается создать новую лицензию. |
| Основной поток | Выводится информация обо всех лицензиях в виде таблицы |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Сортировка списка лицензий»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка лицензий |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список лицензий. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка лицензий». |
| Основной поток | Производится сортировка списка лицензий. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка лицензий»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка лицензий |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список компьютеров. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка лицензий». |
| Основной поток | Производится фильтрация списка лицензий. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о лицензии»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о лицензии |
| Краткое описание | Прецедент позволяет просматривать информацию о конкретной лицензии. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка ПО», «Просмотр списка компьютеров» |
| Основной поток | Выводится информация о конкретной лицензии. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление лицензии в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление лицензии |
| Краткое описание | Прецедент позволяет добавлять новые компьютера. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка лицензий», «Просмотр списка типов ПО» |
| Основной поток | Открывается форма добавления лицензии, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новая лицензия. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Изменение данных о лицензии»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о лицензии |
| Краткое описание | Прецедент позволяет редактировать данные о компьютере. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка лицензий», «Просмотр списка типов ПО» |
| Основной поток | Открывается форма редактирования компьютера, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования лицензии закрывается, и измененные данные о лицензии сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Удаление лицензии из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление лицензии из списка |
| Краткое описание | Прецедент позволяет удалять лицензии из списка. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка лицензий». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то лицензия удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над ПО

### Описание прецедента «Просмотр списка ПО»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка ПО |
| Краткое описание | Прецедент позволяет просматривать список всех ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть хотя бы одно ПО. Если таковых нет, то пользователю предлагается создать новое ПО. |
| Основной поток | Выводится информация обо всех ПО в виде таблицы. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Сортировка списка ПО»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка ПО |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка ПО». |
| Основной поток | Производится сортировка списка ПО. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка ПО»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка ПО |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка ПО». |
| Основной поток | Производится фильтрация списка ПО. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о ПО»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о ПО |
| Краткое описание | Прецедент позволяет просматривать список всех компьютеров. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка ПО», «Просмотр списка компьютеров». |
| Основной поток | Выводится информация о конкретном комплектующем. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление ПО в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление ПО в список |
| Краткое описание | Прецедент позволяет добавлять новое ПО в список. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка ПО», «Просмотр списка типов ПО», «Просмотр списка компьютеров». |
| Основной поток | Открывается форма добавления ПО, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новый компьютера. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Изменение данных о ПО»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о ПО |
| Краткое описание | Прецедент позволяет редактировать данные о ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка ПО», «Просмотр списка типов ПО», «Просмотр списка компьютеров». |
| Основной поток | Открывается форма редактирования ПО, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования ПО закрывается, и измененные данные о компьютере сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Удаление ПО из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление ПО из списка |
| Краткое описание | Прецедент позволяет удалять ПО из списка. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка ПО». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то ПО удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над типами ПО

### Описание прецедента «Просмотр списка типов ПО»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка типов ПО |
| Краткое описание | Прецедент позволяет просматривать список всех типов ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть хотя бы один тип ПО. Если таковых нет, то пользователю предлагается создать новый тип ПО. |
| Основной поток | Выводится информация обо всех типах ПО в виде таблицы |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Сортировка списка типов ПО»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка типов ПО |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список типов ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов ПО». |
| Основной поток | Производится сортировка списка типов ПО. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка типов ПО»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка типов ПО |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список типов ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов ПО». |
| Основной поток | Производится фильтрация списка типов ПО. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о типах ПО»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о типе ПО |
| Краткое описание | Прецедент позволяет просматривать информацию о конкретном типе ПО. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты «Просмотр списка типов ПО», «Просмотр списка ПО». |
| Основной поток | Выводится информация о конкретном типе ПО. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление типа ПО в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление типа ПО в систему |
| Краткое описание | Прецедент позволяет добавлять новые типы ПО. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть выполнен прецедент:  «Просмотр списка типов ПО» |
| Основной поток | Открывается форма добавления типа ПО, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новый тип ПО. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Изменение данных о типе ПО»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о типе ПО |
| Краткое описание | Прецедент позволяет редактировать данные о типе ПО. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов ПО». |
| Основной поток | Открывается форма редактирования типа ПО, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования типа ПО закрывается, и измененные данные о типе ПО сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Удаление типа ПО из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление типа ПО |
| Краткое описание | Прецедент позволяет удалять тип ПО. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то тип ПО удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над работниками

### Описание прецедента «Просмотр списка работников»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка работников |
| Краткое описание | Прецедент позволяет просматривать список всех компьютеров. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должен быть хотя бы один работник. Если таковых нет, то пользователю предлагается создать новый компьютер. |
| Основной поток | Выводится информация обо всех работниках в виде таблицы. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |
| Примечания к использованию | Актор «Директор филиала» может просматривать работников только своего филиала. |

### Описание прецедента «Сортировка списка работников»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка работников |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список работников. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка работников». |
| Основной поток | Производится сортировка списка работников. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка работников»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка работников |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список работников. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка работников». |
| Основной поток | Производится фильтрация списка работников. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о работнике»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о работнике |
| Краткое описание | Прецедент позволяет просматривать список всех компьютеров. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка работников», «Просмотр списка компьютеров». |
| Основной поток | Выводится информация о конкретном работнике. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление работника в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление работника в список |
| Краткое описание | Прецедент позволяет добавлять нового работника в список. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должны быть выполнены прецеденты:  «Просмотр списка работников», «Просмотр списка филиалов», «Просмотр списка компьютеров». |
| Основной поток | Открывается форма добавления работника, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новый работник. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Изменение данных о работнике»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о работнике |
| Краткое описание | Прецедент позволяет редактировать данные о работнике. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должны быть выполнены прецеденты:  «Просмотр списка работников», «Просмотр списка филиалов», «Просмотр списка компьютеров». |
| Основной поток | Открывается форма редактирования работника, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования работника закрывается, и измененные данные о оаботнике сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Удаление работника из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление работника из списка |
| Краткое описание | Прецедент позволяет удалять работника. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка работников». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то работник удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над филиалами

### Описание прецедента «Просмотр списка филиалов»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка филиалов |
| Краткое описание | Прецедент позволяет просматривать список всех филиалов. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть хотя бы один филиал. Если таковых нет, то пользователю предлагается создать новый филиал. |
| Основной поток | Выводится информация обо всех филиалах в виде таблицы. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Сортировка списка филиалов»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка филиалов |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список компьютеров. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка филиалов». |
| Основной поток | Производится сортировка списка филиалов. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка филиалов»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка филиалов |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список компьютеров. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка филиалов». |
| Основной поток | Производится фильтрация списка филиалов. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о филиале»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о филиале |
| Краткое описание | Прецедент позволяет просматривать информацию о конкретном филиале. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должны быть выполнены прецеденты «Просмотр списка комплектующих», «Просмотр списка помещений» |
| Основной поток | Выводится информация о конкретном филиале. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление филиала в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление филиала в список |
| Краткое описание | Прецедент позволяет добавлять новый филиал. |
| Акторы | Директор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка филиалов», «Просмотр списка работников» |
| Основной поток | Открывается форма добавления филиала, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новый филиал. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Изменение данных о филиале»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о филиале |
| Краткое описание | Прецедент позволяет редактировать данные о филиале. |
| Акторы | Директор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка филиалов», «Просмотр списка работников». |
| Основной поток | Открывается форма редактирования филиала, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования филиала закрывается, и измененные данные о филиале сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Удаление филиала из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление филиала из списка |
| Краткое описание | Прецедент позволяет удалять компьютера. |
| Акторы | Директор |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка филиалов». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то компьютер удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над помещениями

### Описание прецедента «Просмотр списка помещений»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка помещений |
| Краткое описание | Прецедент позволяет просматривать список всех помещений. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должно быть хотя бы одно помещение. Если таковых нет, то пользователю предлагается создать новое помещение. |
| Основной поток | Выводится информация обо всех помещениях в виде таблицы |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |
| Примечания к использованию | Акторы: Директор филиала, Администратор филиала могут просматривать помещения только своего филиала. |

### Описание прецедента «Сортировка списка помещений»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка помещений |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список помещений. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка помещений». |
| Основной поток | Производится сортировка списка помещений. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка помещение»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка помещений |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список помещений. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка помещений». |
| Основной поток | Производится фильтровать список помещений. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о помещении»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о помещении |
| Краткое описание | Прецедент позволяет просматривать список всех компьютеров. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка помещений», «Просмотр списка компьютеров» |
| Основной поток | Выводится информация о конкретном помещении. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление помещения в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление помещения в список |
| Краткое описание | Прецедент позволяет добавлять новые помещения. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должны быть выполнены прецеденты:  «Просмотр списка помещений», «Просмотр списка филиалов» |
| Основной поток | Открывается форма добавления помещения, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новое помещение. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |
| Примечания к использованию | Актор Директор филиала, может выбрать только свой филиал. |

### Описание прецедента «Изменение данных о помещении»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о помещении |
| Краткое описание | Прецедент позволяет редактировать данные о помещении. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должны быть выполнены прецеденты:  «Просмотр списка помещений», «Просмотр списка филиалов» |
| Основной поток | Открывается форма редактирования помещения, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования помещения закрывается, и измененные данные о помещении сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |
| Примечания к использованию | Актор Директор филиала, может выбрать только свой филиал. |

### Описание прецедента «Удаление помещения из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление помещения из списка |
| Краткое описание | Прецедент позволяет удалять помещение из списка. |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка помещений». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то компьютер удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над комплектующими

### Описание прецедента «Просмотр списка комплектующих»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка комплектующих |
| Краткое описание | Прецедент позволяет просматривать список всех комплектующих. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственный за склад. |
| Предусловия | Должно быть хотя бы одно комплектующее. Если таковых нет, то пользователю предлагается создать новое комплектующее. |
| Основной поток | Выводится информация обо всех комплектующих в виде таблицы |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |
| Примечания к использованию | Акторы: Директор филиала, Администратор филиала, Ответственный за склад могут просматривать комплектующие только своего филиала. |

### Описание прецедента «Сортировка списка комплектующих»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка комплектующих |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список комплектующих. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственный за склад. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка комплектующих». |
| Основной поток | Производится сортировка списка комплектующих. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка комплектующих»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка комплектующих |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список комплектующих. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственный за склад. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка комплектующих». |
| Основной поток | Производится фильтрация списка компьютеров. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о комплектующем»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о комплектующем |
| Краткое описание | Прецедент позволяет просматривать информацию о конкретном комплектующем. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственный за склад. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка комплектующих». |
| Основной поток | Выводится информация о конкретном комплектующем. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление комплектующего в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление комплектующего в список |
| Краткое описание | Прецедент позволяет добавлять новое комплектующее в систему. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственный за склад. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка комплектующих»  «Просмотр списка филиалов» |
| Основной поток | Открывается форма добавления комплектующего, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новое комплектующее. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |
| Примечания к использованию | Акторы: Директор филиала, Администратор филиала и Ответственный за склад могут выбрать лишь свой филиал. |

### Описание прецедента «Изменение данных о комплектующем»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о комплектующем |
| Краткое описание | Прецедент позволяет редактировать данные о комплектующем. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственный за склад. |
| Предусловия | Должны быть выполнены прецеденты: «Просмотр списка комплектующих»  «Просмотр списка филиалов» |
| Основной поток | Открывается форма редактирования комплектующего, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования комплектующего закрывается, и измененные данные о комплектующем сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Удаление комплектующего из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление комплектующего из списка |
| Краткое описание | Прецедент позволяет удалять комплектующее из списка. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, Ответственный за склад. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка комплектующих». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то компьютер удаляется из базы данных. |
| Альтернативные потоки | — |

## Операции над типами комплектующих

### Описание прецедента «Просмотр списка типов комплектующих»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр списка типов комплектующих |
| Краткое описание | Прецедент позволяет просматривать список всех типов комплектующих. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, |
| Предусловия | Должен быть хотя бы один тип комплектующего. Если таковых нет, то пользователю предлагается создать новый тип комплектующего. |
| Основной поток | Выводится информация обо всех компьютерах в виде таблицы |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Сортировка списка типов комплектующих»

|  |  |
| --- | --- |
| Имя прецедента | Сортировка списка типов комплектующих |
| Краткое описание | Прецедент позволяет сортировать просматриваемый список типов комплектующих. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов комплектующих». |
| Основной поток | Производится сортировка списка типов комплектующих. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отсортированного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Фильтрация списка типов комплектующих»

|  |  |
| --- | --- |
| Имя прецедента | Фильтрация списка типов комплектующих |
| Краткое описание | Прецедент позволяет фильтровать просматриваемый список типов комплектующих. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов комплектующих». |
| Основной поток | Производится фильтрация списка типов комплектующих. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может продолжить просмотр отфильтрованного списка и выполнять другие операции. |
| Альтернативные потоки | — |

### Описание прецедента «Просмотр информации о типе комплектующего»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр информации о типе комплектующего |
| Краткое описание | Прецедент позволяет просматривать информацию о конкретном типе комплектующего. |
| Акторы | Директор, Главный администратор, Директор филиала, Администратор филиала, |
| Предусловия | Должны быть выполнены прецеденты «Просмотр списка типов комплектующих», «Просмотр списка комплектующих» |
| Основной поток | Выводится информация о конкретном типе комплектующего. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |

### Описание прецедента «Добавление типа комплектующего в список»

|  |  |
| --- | --- |
| Имя прецедента | Добавление типа комплектующего в список |
| Краткое описание | Прецедент позволяет добавлять новый тип комплектующего. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должны быть выполнены прецеденты:  «Просмотр списка типов комплектующих» |
| Основной поток | Открывается форма добавления типа комплектующего, куда вносятся необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то в систему добавляется новый тип комплектующего. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Изменение данных о типе комплектующего»

|  |  |
| --- | --- |
| Имя прецедента | Изменение данных о типе комплектующего |
| Краткое описание | Прецедент позволяет редактировать данные о типе комплектующего. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов комплектующих». |
| Основной поток | Открывается форма редактирования типа комплектующего, где изменяются необходимые данные. |
| Постусловия | Если прецедент был успешно завершен, то форма редактирования типа комплектующего закрывается, и измененные данные о типе комплектующего сохраняются. |
| Альтернативные потоки | Если имеются пустые поля, либо введены некорректные данные, то пользователю выдается сообщение об ошибке и дается возможность исправить данные. |

### Описание прецедента «Удаление типа комплектующего из списка»

|  |  |
| --- | --- |
| Имя прецедента | Удаление типа комплектующего из списка |
| Краткое описание | Прецедент позволяет удалять тип комплектующего. |
| Акторы | Директор, Главный администратор. |
| Предусловия | Должен быть выполнен прецедент «Просмотр списка типов комплектующих». |
| Основной поток | При нажатии на кнопку «Удалить», текущая запись удаляется. |
| Постусловия | Если прецедент был успешно завершен, то компьютер удаляется из базы данных. |
| Альтернативные потоки | — |

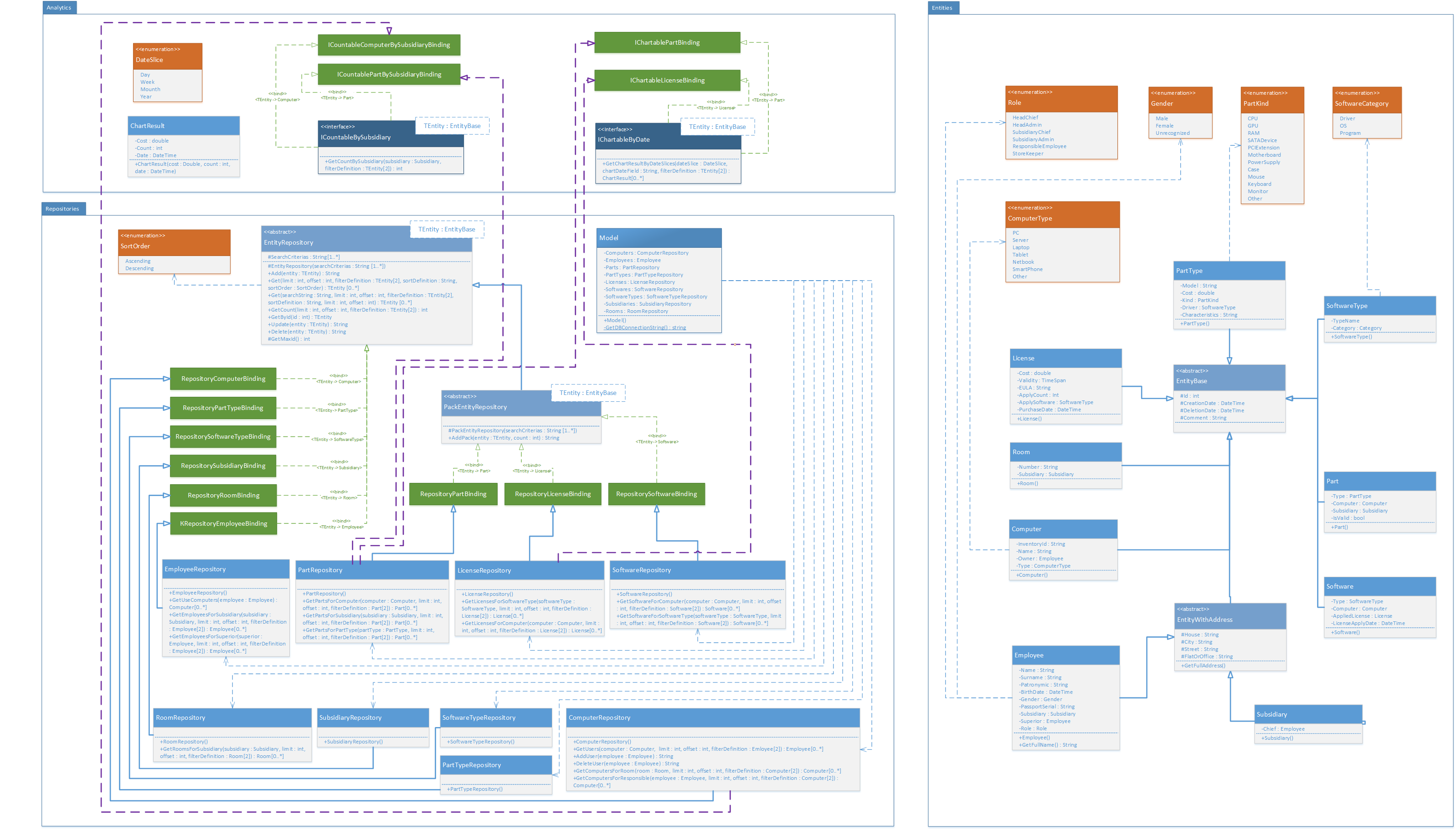
## Аналитика и статистика

### Описание прецедента «Просмотр статистики»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр статистики |
| Краткое описание | Прецедент позволяет просматривать статистические данные по конкретной выбранной сущности |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должно быть хотя бы одно комплектующее. Если таковых нет, то прецедент завершает свою работу. |
| Основной поток | Выводится информация о выбранной анализируемой сущности в виде таблицы, отображаются промежуточные итоги по данной сущности. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |
| Примечания к использованию | Актор Директор филиала, может просматривать статистические данные о сущностях только в рамках своего филиала. |

### Описание прецедента «Просмотр списка лицензий»

|  |  |
| --- | --- |
| Имя прецедента | Просмотр диаграмм |
| Краткое описание | Прецедент позволяет просматривать диаграммы по конкретной выбранной сущности в разрезах определённого периода времени |
| Акторы | Директор, Директор филиала. |
| Предусловия | Должно быть выполнен прецедент «Просмотр статистики» |
| Основной поток | Выводится информация о выбранном свойстве выбранной анализируемой сущности сущностях в виде диаграммы в разрезах определённого периода времени. |
| Постусловия | Если прецедент был успешно завершен, то пользователь может выполнять другие действия. |
| Альтернативные потоки | — |
| Примечания к использованию | Актор Директор филиала, может просматривать аналитические данные о сущностях только в рамках своего филиала. |



# Описание диаграммы классов

## Классы данных (Пакет «Entities»)

### Role

Перечисление, показывающее должность работника в системе, имеет значения:

* HeadChief – Директор
* HeadAdmin – Главный администратор
* SubsidiaryChief – Директор филиала
* SubsidiaryAdmin – Администратор филиала
* ResponsibleEmployee – Ответственное лицо
* StoreKeeper – Кладовщик

### Gender

Перечисление, показывающее пол работника в системе, имеет значения:

* Male – Мужской пол
* Female – Женский пол
* Unrecognized – Не определён

### SoftwareCategory

Перечисление, показывающее категорию типа ПО, имеет значения:

* Driver – Драйвер
* OS – Операционная система
* Program – Прикладная программа

### ComputerType

Перечисление, показывающее разновидности компьютеров, имеет значения:

* PC – Персональный компьютер
* Server – Сервер
* Laptop – Ноутбук
* Tablet – Планшет
* Netbook – Нетбук
* SmartPhone – Смартфон
* Other – Другая разновидность

### PartKind

Перечисление, показывающее разновидность типа комплектующего, имеет значения:

* CPU – Процессор
* GPU – Видеокарта
* RAM – Оперативная память
* SATADevice – SATA-совместимый накопитель
* PCIExtension – PCI расширение
* Motherboard – Материнская плата
* PowerSupply – Блок питания
* Case – Корпус
* Mouse – Компьютерная мышь
* Keyboard – Клавиатура
* Monitor – Монитор
* Other – Другой тип комплектующего

### EntityBase

Абстрактный класс EntityBase является общим предком всех классов, отображающих данные. Содержит атрибуты:

* #ID – Id сущности
* #CreationDate : DateTime – Дата создания сущности
* #DeletionDate : DateTime – Дата удаления сущности
* #IsDeleted : Bool – Флаг, показывающий, является-ли сущности удалённой
* #Comment : String – Комментарий к сущности

### EntityWithAddress

Абстрактной класс, унаследованный от EntityBase. Служит для обобщения сущностей, имеющих адрес. Содержит атрибуты:

* #House : String – Номер дома
* #City : String – Название города
* #Street : String – Название улицы
* #FlatOrOffice : String – Обозначение квартиры или офиса

Содержит методы:

* +GetFullAddress() : String – возвращает строку с полным адресом сущности

### Computer

Данный класс наследует класс Entity. Хранит данные о компьютерах. Содержит атрибуты:

* -InventoryId : String – инвентарный номер (цифробуквенное обозначение)
* -Name : String – имя компьютера
* -Owner : Employee – Ответственный работник за этот компьютера
* -Type : ComputerType – тип Компьютера

Содержит методы:

* +Computer() – конструктор класса без параметров

### License

Данный класс наследует класс Entity. Хранит данные о лицензии. Содержит атрибуты:

* -Cost : double – Стоимость лицензии
* -Validity : TimeSpan – Время действия применения лицензии к рабочему месте
* -EULA : String – Пользовательское соглашение
* -ApplyCount : int – Количество компьютеров, к которым может быть применена лицензия.
* -ApplySoftware : SoftwareType – Тип ПО, к которому может быть применена лицензия
* -PurchaseDate : DateTime – Дата покупки лицензии

Содержит методы:

* +License() – конструктор класса без параметров

### Software

Данный класс наследует класс Entity. Хранит данные о ПО. Содержит атрибуты:

* -Type : SoftwareType – Тип ПО
* -Computer : Computer – Компьютер, на который установлено данное ПО
* -AppliedLicense : License – Применённая лицензия к данному ПО
* -LicenseApplyDate : DateTime – Дата применения лицензии к данному ПО

Содержит методы:

* +Software() – конструктор класса без параметров

### SoftwareType

Данный класс наследует класс Entity. Хранит данные о типе ПО. Содержит атрибуты:

* -TypeName – Имя типа ПО
* -Category : Category – Категория типа ПО

Содержит методы:

* +SoftwareType() – конструктор класса без параметров

### Employee

Данный класс наследует класс EntityWithAddress. Хранит данные о работнике. Содержит атрибуты:

* -Name : String – Имя работника
* -Surname : String – Фамилия работника
* -Patronymic : String – Отчество работника
* -BirthDate : DateTime – Дата рождения работника
* -Gender : Gender – Пол работника
* -PassportSerial : String – Серия и номер паспорта
* -Subsidiary : Subsidiary – Филиал, к которому принадлежит работника
* -Superior : Employee – Руководитель работника
* -Role : Role – Должность работника

Содержит следующие методы:

* +Employee() – конструктор класса без параметров
* +GetFullName() : String – функция получения ФИО

### Subsidiary

Данный класс наследует класс EntityWithAddress. Хранит данные о филиале. Содержит атрибуты:

* -Chief : Employee – Директор филиала

Содержит методы:

* +Subsidiary() – конструктор класса без параметров

### Room

Данный класс наследует класс Entity. Хранит данные о помещении. Содержит атрибуты:

* -Number : String – Номер помещения (цифробуквенное обозначение)

Содержит методы:

* +Room() – конструктор класса без параметров

### Part

Данный класс наследует класс Entity. Хранит данные о комплектующем. Содержит атрибуты:

* -Type : PartType – Тип комплектующего
* -Computer : Computer – Компьютер, на который установлено данное комплектующее
* -Subsidiary : Subsidiary – Филиал, на складе которого находится комплектующее
* -IsValid – Флаг, показывающий работоспособность комплектующего

Содержит методы:

* +Part() – конструктор класса без параметров

### PartType

Данный класс наследует класс Entity. Хранит данные о типе комплектующего. Содержит атрибуты:

* -Model : String – Марка и модель комплектующего (название)
* -Cost : double – Стоимость
* -Kind : PartKind – Разновидность типа комплектующего
* -Driver : SoftwareType – Тип ПО драйвера данного комплектующего
* -Characteristics : String – Характеристики комплектующего

Содержит методы:

* +PartType () – конструктор класса без параметров

## Интерфейсы и классы для работы с аналитикой и диаграммами (Пакет «Analytics»)

### DateSlice

Перечисление, показывающее разрез временного промежутка, имеет значения:

* Day – День
* Week – Неделя
* Mounth – Месяц
* Year – Год

### ChartResult

Данный класс наследует класс Entity. Хранит данные о комплектующем. Содержит атрибуты:

* -Cost : double – суммарная стоимость на промежутке времени
* -Count : int – суммарное количество на промежутке времени
* -Date : DateTime – дата

Содержит методы:

* +ChartResult(cost : Double, count : int, date : DateTime) – конструктор с параметрами, принимающий значение полей «Cost», «Count», «Date»

### ICountableBySubsidiary

Шаблонный интерфейс с шаблонным параметром TEntity (с ограничением на подстановку только наследников EntityBase). Интерфейс для постройки промежуточных итогов количества сущностей для данного филиала.

Содержит методы:

* +GetCountBySubsidiary(subsidiary : Subsidiary, filterDefinition : String[0..\*]) : int – фукнция, возвращающая количество сущностей, отфильтрованных согласно фильтру filterDefinition, для филиала subsidiary.

### IChartableByDate

Шаблонный интерфейс с шаблонным параметром TEntity (с ограничением на подстановку только наследников EntityBase), предоставляющий методы для постройки диаграмм сущностей в разрезе временных промежутков.

Содержит методы:

* +GetChartResultByDateSlices(dateSlice : DateSlice, chartDateField : String, filterDefinition : TEntity[2]) : ChartResult[0..\*] – функция, возвращающая ChartResult в разрезе промежутка DateSlice, сущностей, отфильтрованных согласно filterDefiniton, по дате, извлекаемой из поля сущности с названием chartDateField

## Сущности для работы с данными (Пакет «Repository»)

### SortOrder

Перечисление, показывающее направление сортировки, имеет значения:

* Ascending – по возрастанию
* Descending – по убыванию

### EntityRepository

Абстрактный шаблонный класс EntityBase с шаблонным параметром TEntity (с ограничением на подстановку только наследников EntityBase) является общим предком всех классов, предоставляющих операции для работы с данными. Содержит атрибуты:

* #SearchCriterias : String[1..\*] защищеённое поле критериев тектового поиска, всегда должен быть минимум один критерий поиска – Id сущности.

Содержит методы:

* #EntityRepository(searchCriterias : String [1..\*]) – защищённый конструктор класса EntityRepository, принимающее последовательность searchCriterias (критерии поиска по строке)
* +Add(entity : TEntity) : string – функция добавления TEntity в систему, возвращает статус операции
* +Get(limit : int, offset : int, filterDefinition : TEntity[2], sortDefinition : String, sortOrder : SortOrder) : TEntity [0..\*] – функция возвращает множество всех элементов, отфильтрованных согласно параметру filterDefinition, отсортированных согласно параметрам sortDefinition (поля, по которым производится сортировка) и sortOrder (направление сортировки), ограниченные в количестве limit, со смещением offset
* +Get(searchString : String, filterDefinition : String[0..\*], sortDefinition : String, sortOrder : SortOrder, limit : int, offset : int) : TEntity [0..\*] – перегрузка функции Get возвращает множество всех элементов, отобранных по строке поиска searchString, отфильтрованных согласно параметру filterDefinition, отсортированных согласно параметрам sortDefinition (поля, по которым производтся сортировка) и sortOrder (направление сортировки), ограниченные в количестве limit, со смещением offset
* +GetCount(filterDefinition : String[0..\*]) : int – функция, возвращающая количество элементов, отфильтрованных согласно filterDefinition
* +GetById(id : int) : TEntity – функция, возвращает элемент TEntity по его id,
* +Update(entity : TEntity) : string – функция производит обновление элемента entity, возвращает статус операции
* +Delete(entity : TEntity) : string ؘ– функция производит удаление элемента entity, возвращает статус операции
* #GetMaxId() : int –защищённая функция, возвращает максимальный существующий id в системе

### PackEntityRepository

Абстрактный шаблонный класс PackEntityBase с шаблонным параметром TEntity (с ограничением на наследников EntityBase) расширяет EntityBase функцией пакетного добавления сущностей.

Содержит методы:

* #PackEntityRepository(searchCriterias : String [1..\*]) – защищённый конструктор класса PackEntityRepository, принимающее последовательность searchCriterias (критерии поиска по строке)
* +AddPack(entity : TEntity, count : int) : String – функция добавляющая в систему некоторое количество count элементов entity в систему

### ComputerRepository

Класс, наследующий EntityRepository (через привязку RepositoryComputerBinding). Реализует интерфейс ICountableBySubsidiary через связывание ICountableComputerBySubsidiaryBinding (где TEntity -> Computer). Предоставляет операции для работы с компьютерами.

Содержит методы:

* +ComputerRepository() – конструктор без параметров
* +GetUsers(computer : Computer, limit : int, offset : int, filterDefinition : Emloyee[2]) : Employee[0..\*] – функция, возвращающая пользователей данного компьютера отфильтрованных согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +AddUser(employee : Employee) : String – функция, позволяющая добавлять пользователя к компьютеру, возвращает статус выполнения операции
* +DeleteUser(employee : Employee) : String – функция, позволяющая удалять пользователя пользователя компьютера, возвращает статус выполнения операции
* +GetComputersForRoom(room : Room, limit : int, offset : int, filterDefinition : Computer[2]) : Computer[0..\*] – функция, возвращающая все компьютеры помещения room, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +GetComputersForResponsible(employee : Employee, limit : int, offset : int, filterDefinition : Computer[2]) : Computer[0..\*] – функция, возвращающая все компьютеры, за которые ответственен работник employee, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset

### LicenseRepository

Наследует шаблонный класс PackEntityRepository через связывание RepositoryLicenseBinding (где TEntity -> License). Реализует интерфейс IChartableByDate через связывание IChartableLicenseBinding (где TEntity -> License). Предоставляет операции для работы с лицензиями.

Содержит методы:

* +LicenseRepository() – конструктор без параметров
* +GetLicensesForSoftwareType(softwareType : SoftwareType, limit : int, offset : int, filterDefinition : License[2]) : License[0..\*] – функция, возвращающая все лицензии на тип ПО softwareType, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +GetLicensesForComputer(computer : Computer, limit : int, offset : int, filterDefinition : License[2]) : License[0..\*] – функция, возвращающая все лицензии, которые применены к компьютеру computer, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset

### SoftwareRepository

Наследует шаблонный класс PackEntityRepository через связывание RepositorySoftwareBinding (где TEntity -> Software). Предоставляет операции для работы с ПО.

Содержит методы:

* +SoftwareRepository() – конструктор без параметров
* +GetSoftwareForComputer(computer : Computer, limit : int, offset : int, filterDefinition : Software[2]) : Software[0..\*] – функция, возвращающая все ПО, установленные на компьютер computer, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +GetSoftwareForSoftwareType(softwareType : SoftwareType, limit : int, offset : int, filterDefinition : Software[2]) : Software[0..\*] – функция, возвращающая все ПО типа ПО softwareType, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset

### SoftwareTypeRepository

Наследует шаблонный класс EntityRepository через связывание RepositorySoftwareTypeBinding (где TEntity -> SoftwareType). Предоставляет операции для работы с типами ПО.

Содержит методы:

* +SoftwareTypeRepository() – публичный конструктор с параметрами

### EmployeeRepository

Класс, наследующий EntityRepository (через привязку RepositoryComputerBinding), предоставляет операции для работы с компьютерами.

Содержит методы:

* +ComputerRepository() – конструктор без параметров
* +GetComputersForResponsible(employee : Employee, limit : int, offset : int, filterDefinition : Computer[2]) : Computer[0..\*] – функция, возвращающая все компьютеры, за которые ответственен работник employee, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +GetEmployeesForSubsidiary(subsidiary : Subsidiary, limit : int, offset : int, filterDefinition : Employee[2]) : Employee[0..\*] функция, возвращающая всех работников филиала subsidiary, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +GetEmployeesForSuperior(superior : Employee, limit : int, offset : int, filterDefinition : Employee[2]) : Employee[0..\*] – функция возвращающая всех подчинённых работника superior, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset

### SubsidiaryRepository

Наследует шаблонный класс EntityRepository через связывание RepositorySubsidiaryBinding (где TEntity -> Subsidiary).

Содержит методы:

* +SubsidiaryRepository() – публичный конструктор без параметров

### RoomRepository

Наследует шаблонный класс EntityRepository через связываение RepositoryRoomBinding (где TEntity -> Room). Предоставляет операции для работы с помещениями.

Содержит методы:

* RoomRepository() – конструктор класса без параметров

### PartRepository

Наследует шаблонный класс PackEntityRepository через связываение RepositoryPartBinding (где TEntity -> Part). Реализует интерфейсы IChartableByDate и IChartableBySubsidiary через связывания: ICountablePartBySubsidiaryBinding (где TEntity -> Part) и IChartablePartBinding (где TEntity -> Part). предоставляет операции для работы с компьютерами.

Содержит методы:

* +PartRepository() – конструктор класса без параметров
* +GetPartsForComputer(computer : Computer, limit : int, offset : int, filterDefinition : Part[2]) : Part[0..\*] – функция, возвращающая комплектующие установленные на компьютер computer, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +GetPartsForSubsidiary(subsidiary : Subsidiary, limit : int, offset : int, filterDefinition : Part[2]) : Part[0..\*] – функция, возвращающая все комплектующие, которые находятся на складе филиала subsidiary, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset
* +GetPartsForPartType(partType : PartType, limit : int, offset : int, filterDefinition : Part[2]) : Part[0..\*] – функция, возвращающая все комплектующие, принадлежащие к типу комплектующего partType, отфильтрованные согласно параметру filterDefinition, ограниченные в количестве limit, со смещением offset

### PartTypeRepository

Наследует шаблонный класс EntityRepository через связывание RepositoryPartTypeBinding (где TEntity -> PartType). Предоставляет операции для работы с типами комплектующих.

Содержит методы:

* +PartTypeRepository() – публичный конструктор с параметрами

### Model

Класс представляет собой единую точку доступа к объектам работы с данными.

Содержит атрибуты:

* -Computers : ComputerRepository – поле, предоставляющее операции над компьютерами
* -Employees : Employee – поле, предоставляющее операции над работниками
* -Parts : PartRepository – поле, предоставляющее операции над комплектующими
* -PartTypes : PartTypeRepository – поле, предоставляющее операции над типами комплектующих
* -Licenses : LicenseRepository – поле, предоставляющее операции над лицензиями
* -Softwares : SoftwareRepository – поле, предоставляющее операции над ПО
* -SoftwareTypes : SoftwareTypeRepository – поле, предоставляющее операции над типами ПО
* -Subsidiaries : SubsidiaryRepository – поле, предоставляющее операции над помещениями
* -Rooms : RoomRepository – поле, предоставляющее операции над помещениями

Содержит методы:

* +Model() – конструктор без параметров
* -GetDBConnectionString() : string – статическая функция, получающая строку подключения к базе данных из конфигурационного файла