1. Вариант использования **CheckData**

**Акторы**: Engineer.

**Цель**: просмотреть информацию о типах ВС и ВС самолётного парка.

**Предусловия**:

**Постусловия**: Engineer может вносить изменения в базе самолётного парка.

**Основной** **сценарий**:

1. Engineer запрашивает информацию о типах ВС и ВС самолётного парка.
2. Система отображает на экране таблицы типов ВС и ВС самолётного парка.
3. Вариант использования **AddAircraftType**

**Акторы**: Engineer.

**Цель**: добавить тип ВС.

**Предусловия**: Engineer может вносить изменения в базе самолётного парка.

**Постусловия**:

**Основной** **сценарий**:

1. Engineer запрашивает у системы возможность добавить тип ВС.
2. Система позволяет добавить новую запись в таблице типов ВС.
3. Engineer заносит информацию о типе ВС: название, коэффициент простоя, среднее время восстановления.

**Альтернативные** **сценарии**:

1. Если Engineer не ввёл какую-либо информацию, система предупреждает об этом Engineer и позволяет выбрать: внести недостающую информацию или отменить действие по добавлению типа ВС.
2. Если Engineer выбрал внести недостающую информацию, то переходит к п. 3.
3. Если Engineer выбрал отменить действие по добавлению типа ВС, система скрывает возможность добавить новую запись в таблице типов ВС.
4. Вариант использования **DeleteAircraftType**

**Акторы**: Engineer.

**Цель**: удалить тип ВС.

**Предусловия**: Engineer может вносить изменения в базе самолётного парка.

**Постусловия**:

**Основной** **сценарий**:

1. Engineer запрашивает у системы возможность удалить тип ВС.
2. Система позволяет удалить запись в таблице типов ВС.
3. Engineer выбирает тип ВС и удаляет его.
4. Вариант использования **EditAircraftType**

**Акторы**: Engineer.

**Цель**: редактировать информацию о типе ВС.

**Предусловия**: Engineer может вносить изменения в базе самолётного парка.

**Постусловия**:

**Основной** **сценарий**: Engineer выбирает тип ВС таблице AircraftType и редактирует содержание.

**Основной сценарий:**

1. Engineer запрашивает у системы возможность редактировать информацию о типе ВС.
2. Система позволяет редактировать информацию о типе ВС.
3. Engineer редактирует информацию о типе ВС.

**Альтернативные сценарии:**

1. Если Engineer удалил и не ввёл какую-либо информацию, система предупреждает об этом Engineer и позволяет выбрать: внести недостающую информацию или отменить действие по редактированию типа ВС.
2. Если Engineer выбрал внести недостающую информацию, то переходит к п. 3.
3. Если Engineer выбрал отменить действие по редактированию типа ВС, система скрывает возможность редактировать запись в таблице типов ВС.
4. Вариант использования **AddAircraft**

**Акторы**: Engineer.

**Цель**: добавить ВС в базу самолётного парка.

**Предусловия**: Engineer может вносить изменения в базе самолётного парка.

**Постусловия**:

**Основной** **сценарий**: Engineer создает новую запись в таблице Aircrafts и заносит поля, характерные для данного ВС: тип, дата поступления, количество отказов, время отказов (последние два поля необязательные и обновляются в процессе эксплуатации).

**Основной сценарий:**

1. Engineer запрашивает у системы возможность добавить ВС.
2. Система позволяет добавить новую запись в таблице ВС.
3. Engineer заносит информацию о ВС: тип, дата поступления, количество отказов, время отказов (последние два поля необязательные и обновляются в процессе эксплуатации).

**Альтернативные сценарии:**

1. Если Engineer не ввёл какую-либо информацию, система предупреждает об этом Engineer и позволяет выбрать: внести недостающую информацию или отменить действие по добавлению ВС.
2. Если Engineer выбрал внести недостающую информацию, то переходит к п. 3.
3. Если Engineer выбрал отменить действие по добавлению ВС, система скрывает возможность добавить новую запись в таблице ВС.
4. Вариант использования **DeleteAircraft**

**Акторы**: Engineer.

**Цель**: удалить ВС из базы самолётного парка.

**Предусловия**: Engineer может вносить изменения в базе самолётного парка.

**Постусловия**:

**Основной сценарий:**

1. Engineer запрашивает у системы возможность удалить ВС.
2. Система позволяет удалить запись в таблице ВС.
3. Engineer выбирает ВС и удаляет его.
4. Вариант использования **EditAircraft**

**Акторы**: Engineer.

**Цель**: редактировать ВС или внести информацию об отказе

**Предусловия**: Engineer может вносить изменения в базе самолётного парка.

**Постусловия**:

**Основной сценарий:**

1. Engineer запрашивает у системы возможность редактировать информацию о ВС.
2. Система позволяет редактировать информацию о ВС.
3. Engineer редактирует информацию о ВС.

**Альтернативные сценарии:**

1. Если Engineer удалил и не ввёл какую-либо информацию, система предупреждает об этом Engineer и позволяет выбрать: внести недостающую информацию или отменить действие по редактированию ВС.
2. Если Engineer выбрал внести недостающую информацию, то переходит к п. 3.
3. Если Engineer выбрал отменить действие по редактированию ВС, система скрывает возможность редактировать запись в таблице ВС.
4. Вариант использования **GetDiagnosis**

**Акторы**: Engineer, Technician.

**Цель**: получить поэтапный расчёт готовности Ктг.

**Предусловия**:

**Постусловия**: Ктг рассчитан.

**Основной** **сценарий**:

1. Актор в лице Engineer или Technician на экране выбирает ВС и запрашивает его диагностику.
2. Система на экране отображает поэтапный расчёт характеристик, результатом является рассчитанный Кт.

**Альтернативные** **сценарии**:

Если на шаге 1. актор запросит диагностику по нескольким ВС, то на шаге 2. результатом будет рассчитанные для нескольких ВС коэффициент Кт.

По результатам диагностики Engineer может вызвать вариант использования AssignRepair.

1. Вариант использования **GetPredict**

**Акторы**: Engineer

**Цель**: получить прогноз отказа ВС

**Предусловия**:

**Постусловия**:

**Основной сценарий:**

1. Engineer выбирает ВС и запрашивает прогноз.
2. Система строит функцию распределения времени отказов на основе статистических данных, отображается визуализация на экране.
3. Engineer получает вероятность отказа ВС к заданному моменту времени.
4. Вариант использования **AssignRepair**

**Акторы**: Engineer

**Цель**: назначить ремонт ВС

**Предусловия**:

**Постусловия**: Engineer создал заявка на ремонт.

**Основной сценарий:**

1. Engineer создает заявку на ремонт ВС.
2. Выполняются варианты использования AssignTechnician и OrderDetails.
3. Вариант использования **AssignTechnician**

**Акторы**: Engineer

**Цель**: назначить техника на ремонт ВС

**Предусловия**: Engineer создал заявка на ремонт.

**Постусловия**:

**Основной сценарий:**

1. Engineer назначает доступного Technician для исполнения заявки на ремонт.
2. Информация о заявке доставляется до исполняющего Technician.
3. Вариант использования **OrderDetails**

**Акторы**: Engineer

**Цель**: заказать детали для ремонта ВС

**Предусловия**: Engineer создал заявка на ремонт.

**Постусловия**:

**Основной сценарий:**

1. Engineer запрашивает у системы возможность заказать необходимые детали для ремонта ВС.
2. Система отображает на экране перечень деталей.
3. Engineer выбирает деталь и указывает количество.
4. Вариант использования **Repair**

**Акторы**: Technician

**Цель**: выполнить ремонт ВС.

**Предусловия**:

**Постусловия**:

**Основной сценарий:**

1. Technician получает информацию о заявке на ремонт.
2. Technician выполняет ремонт ВС.
3. Technician производит вызывает вариант использования GetDiagnosis. Результат диагностики удовлетворяет Technician.
4. Статус ремонта меняется на “исполнено”.

**Альтернативные** **сценарии**:

1. Если в п. 3 результат диагностики не удовлетворяет Technician, то он возвращается в п. 2.
2. Вариант использования **GetReadiness**

**Акторы**: Dispatcher

**Цель**: узнать техническую готовность.

**Предусловия**:

**Постусловия**:

**Основной** **сценарий**:

1. Dispatcher запрашивает у системы коэффициент Ктг для выбранного ВС.
2. Система отображает Ктг.