


ТЕХНИЧЕСКОЕ ЗАДАНИЕ на разработку игры “K-Cover Dance Life”

1. Введение

“K-Cover Dance Life” — мобильная симуляционная игра о жизни танцовщицы/танцора кавер-команды. Игрок развивается в индустрии кавер-кей-поп танцев, прокачивает навыки, вступает в команды, участвует в проектах, фестивалях, случайных событиях и строит карьеру.

В игре реализуются механики:

- развитие навыков (solo/group dance);
- система проектов (индивидуальные и командные);
- взаимодействие с NPC-командой;
- экономическая система;
- недельный и дневной прогресс;
- усталость и влияние ритма жизни;
- фестивали и сценарные события;
- репутация игрока;
- UI вкладок и синхронизация времени между ними.

В документе приведены:

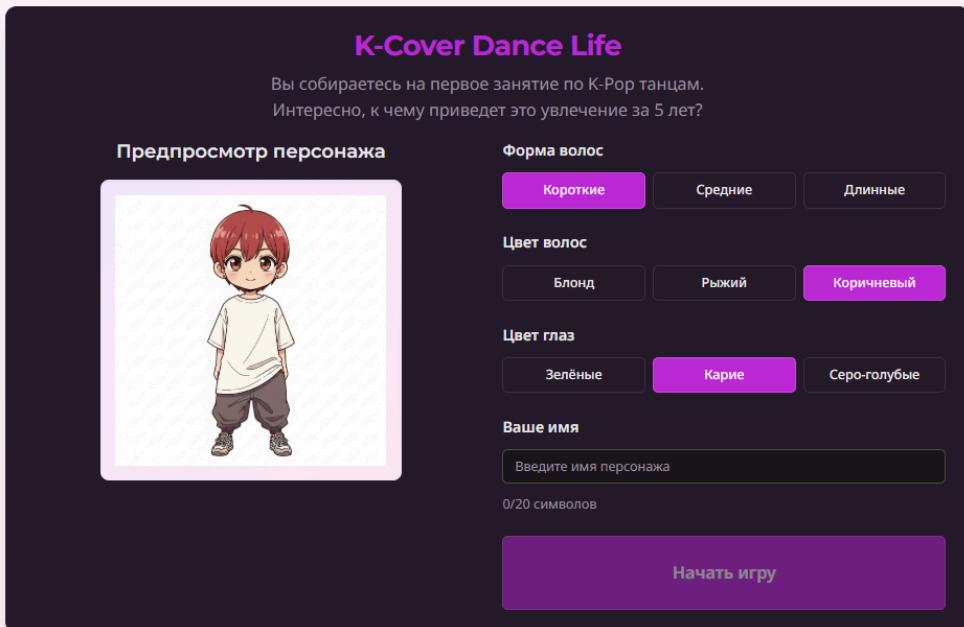
- описания всех игровых механик,
- математические формулы,
- логика событий,
- структура данных,

- потребности UI,
 - моменты для будущего расширения.
-

2. Основной цикл игры

1. Время движется постоянно (день → неделя → месяц).
 2. При открытии любой вкладки время продолжается **синхронно**, **кроме** случаев появления pop-up событий.
 3. Игрок назначает тренировки в проектах и управляет усталостью.
 4. Проекты продвигаются каждый день на основе недельного плана тренировок.
 5. Игрок выполняет проекты, участвует в фестивалях, получает награды, прокачивает навыки и репутацию.
 6. NPC-команда периодически предлагает командные проекты.
 7. Игрок может принимать или отклонять предложения.
-

3. Экран создания персонажа



Игрок может выбрать внешний вид персонажа из заранее подготовленных 27 вариантов (3 длины × 3 цвета волос × 3 цвета глаз). Все изображения уже находятся в папке ресурсов игры.

Доступные параметры:

- **Длина волос**
 - короткие
 - средние
 - длинные
- **Цвет волос**
 - блонд
 - рыжий
 - тёмно-коричневый

- Цвет глаз
 - зелёные
 - карие
 - голубо-серые

Логика отображения:

При открытии экрана по умолчанию отображается аватар:

короткие волосы + блонд + зелёные глаза

- При выборе любого параметра аватар **автоматически обновляется**, подставляя соответствующее готовое изображение.

Полный набор изображений:

3 длины × 3 цвета волос × 3 цвета глаз = 27 аватаров.

- Пользователь мгновенно видит итоговый внешний вид.

4. Интерфейс игры

4.1 Главный экран

Год 2 / Месяц 6 / День 18

Главная Поиск проекта Ваши каверы

Тренировка женского стиля Тренировка мужского стиля Отдых (1 месяц)

Наталья

Деньги 253 950 ₽

Репутация Ужасная

Популярность Ноунейм

Усталость 0/100

Навыки

Женская хореография Беггинер 0/100

Мужская хореография Беггинер 0/100

У вас нет активных проектов
Перейдите во вкладку "Поиск проекта", чтобы начать работу

Год 1 / Месяц 1 / День 2

Главная Поиск проекта Ваши каверы

Тренировка женского стиля Тренировка мужского стиля Отдых (1 месяц)

Наталья

Деньги 15 000 ₽

Репутация Ужасная

Популярность Ноунейм

Усталость 0/100

Навыки

Женская хореография Беггинер 0/100

Мужская хореография Беггинер 0/100

У вас нет активных проектов
Перейдите во вкладку "Поиск проекта", чтобы начать работу

Отображается:

- День недели, число, месяц

- **Управление временем:**

На главном экране присутствуют элементы управления временем:

- **Кнопка «Пауза»**

Останавливает игровой таймер полностью.

Pop-up при этом **не появляется**, но время = 0, пока кнопка не будет нажата снова.

- **Кнопка скорости времени**

Переключает режимы:

1x – стандартный темп (1 игровой день = X сек)

2x – ускоренный

5x – супер-ускоренный

При смене вкладок скорость сохраняется.

- **Кнопка «Рестарт игры»**

Открывает pop-up с подтверждением:

Вы уверены, что хотите начать игру заново?

Прогресс будет потерян.

[Да] [Нет]

После подтверждения игра сбрасывается к моменту создания персонажа.

- **Тема интерфейса (День / Ночь)**

Игрок может переключать визуальную тему интерфейса:

- **Дневная тема** — светлая, молочная палитра
- **Ночная тема** — тёмная, с мягкими контрастами

Переключение выполняется:

- через кнопку в верхней части главного экрана,
- или в настройках.

Переключение работает мгновенно и сохраняется при обновлении страницы.

• **Профиль игрока:**

- ❖ Аватар
 - ❖ Ник
 - ❖ Деньги
 - ❖ Репутация
 - ❖ Популярность
 - ❖ Усталость
 - ❖ Навыки:
 - F_skill
 - M_skill
-
- Команда и ее характеристики
 - Кнопки перехода:
Главная / Поиск проекта / Ваши каверы
 - Кнопки действий: Тренировать женский стиль / Тренировать мужской стиль / Отдых

4.2 Дополнение к разделу: Интерфейс игры — Рейтинги NPC и команд

В игре добавляются новые элементы интерфейса, связанные с рейтингами NPC и команд.

4.2.1. Новые вкладки в интерфейсе

В навигации должны появиться **две новые вкладки**:

1) “Рейтинг NPC”

- отображает полный список всех NPC игры;
- сортировка по глобальному рейтингу (см. логику ниже);
- сверху — строки с позициями;
- игрок может открывать карточку любого NPC.

2) “Рейтинг команд”

- отображает все существующие команды;
 - сортировка по рейтингу команды (см. ниже);
 - отображение состава команды по клику;
 - визуальная подсветка команд, в которых есть игрок.
-

4.2.2. Топ-5 NPC на главном экране

На стартовом / главном экране (там, где отображается персонаж игрока) должен быть блок:

“ТОП-5 NPC игры”

В нём отображаются:

- аватар,
- имя/ник,
- текущий рейтинг,
- самый высокий навык.

Блок автоматически обновляется при обновлении статов NPC.

4.2.3. Рейтинг NPC — логика расчёта

Рейтинг NPC вычисляется на основании нескольких параметров:

```
ratingNPC =  
    (0.6 × avgSkill) +  
    (0.3 × popularity) +  
    (0.1 × reputation)
```

Где:

```
avgSkill = (F_skill + M_skill) / 2
```

Очевидно:

- навыки — главный фактор,
- популярность — второстепенный,
- репутация — дополнительный.

Позиция игрока в этом рейтинге отображается:

- в блоке персонажа (рядом с ником),
 - в виде: “Ваш рейтинг: №27 из 180”.
-

4.2.4. Рейтинг команд (логика)

Рейтинг команды нужен для:

- вкладки “Рейтинг команд”,
- отображения позиции команды игрока в интерфейсе.

Формула:

```
ratingTeam =
```

```
(0.7 × teamSkill) +  
(0.3 × teamPopularity)
```

Где:

```
teamSkill = mean( avgSkill всех участников )  
teamPopularity = mean( popularity участников )
```

4.2.5. Отображение рейтинга команды игрока

Если игрок состоит в команде:

- в блоке “Команда” на главном экране появляется строка:

«Рейтинг команды: №X из Y команд»

-
- позиция обновляется после каждого обновления статов.

4.2.6. Отображение рейтинга NPC игрока

В блоке персонажа:

Никнейм игрока

Рейтинг: №X из Z NPC

Если игрок отсутствует в топ-5, он всё равно должен видеть своё место.

4.2.7. Синхронизация рейтингов

Рейтинги NPC и команд:

- пересчитываются **каждый игровой месяц**,
 - обновляются на всех экранах, где выводятся,
 - могут резко измениться после фестиваля, проекта или события.
-

4.2.8. Минимальные требования к интерфейсу

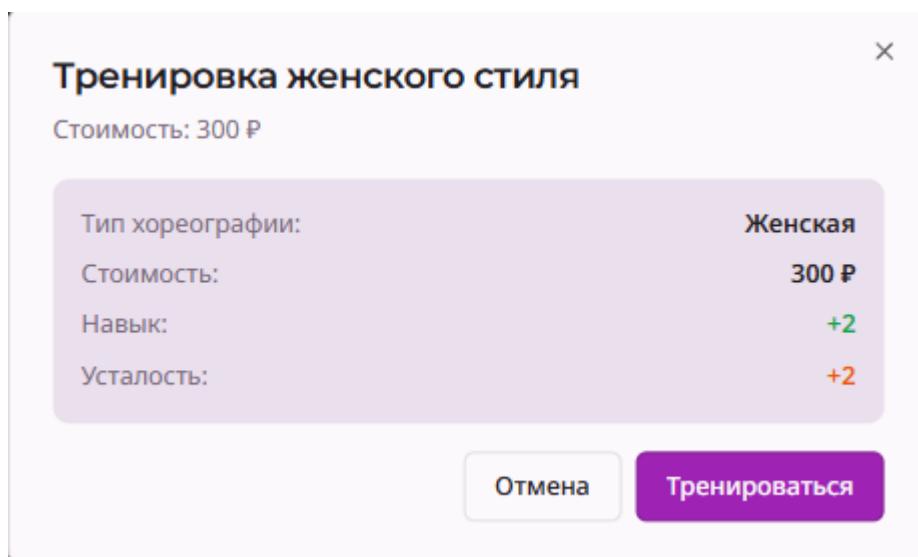
- вкладки "Рейтинг NPC" и "Рейтинг команд" должны быть доступны всегда (даже если у игрока нет команды);
- в карточках NPC/Команд отображаются:
 - рейтинг,
 - навыки,
 - популярность,
 - репутация,
 - состав команды,
 - поведенческая модель NPC;
- интерфейс не блокирует игровой таймер при просмотре рейтингов.

5. Поп-апы

На время любого pop-up события **игровое время останавливается**.

Событие закрывается:

- кнопкой “OK” (если информационное)
- кнопками “Принять / Отказаться” (если выбор)
- крестиком (если просто случайность).



6. Время и прогресс

Движение времени:

- Каждый реальный N секунд = 1 игровой день (значение ставится разработчиком).
- 7 дней = 1 неделя
- 4 недели = 1 месяц

Особая логика:

- Время движется **в любых вкладках**, кроме pop-up.
- При смене вкладки, если в этот момент возникло событие — pop-up показывается **сразу**, и время стоит до закрытия.

7. Параметры игрока

7.1 Деньги

Доходы

- Успешные проекты
- Успешные фестивали
- Случайные позитивные события
- Командные достижения

Расходы

1) Тренировка у хореографа

Стоимость: **300 ₽**

Эффект:

+2 к выбранному навыку (F_skill или M_skill)
+2 усталости

2) Тренировки в проекте (основные и дополнительные)

Стоимость **150–400 ₽** за каждую тренировку
(у каждого проекта задана своя стоимость).

3) Костюмы (расход на 50% прогресса)

FAST проекты: **1000–5000 ₽**

LONG проекты: **3000–10000 ₽**

Списывается **один раз**, при достижении:

progress >= 50%

7.2 Усталость (Tired)

- Диапазон: **0–100**
- Влияние:
 - 70 — снижается эффективность тренировок

- 90 — шанс негативных событий растёт

Изменение усталости:

- +2 за каждую тренировку у хореографа
- +1 за каждую тренировку в проекте
- -1 в день, если игрок **не тренируется вообще**

При событии “Застой”:

Tired -= 10

F_skill *= 0.9

M_skill *= 0.9

-
-

7.3 Репутация (Rep)

Растёт за:

- завершённые проекты
- фестивали
- позитивные события
- сотрудничество с командой

Падает за:

- провалы проектов
- негативные события

7.4 Качественное отображение характеристик

Для удобства восприятия игрок видит не числовые значения, но и **качественные ранги** характеристик.



Популярность (Pop)

Использует 3 текстовых ранга:

- **Ноунейм** (0–30)
- **Известный** (31–70)
- **Топовый** (71–100)

На UI отображается название ранга.

⭐ Навыки танца (F_skill и M_skill)

Каждый навык отображается тремя уровнями мастерства:

- **Новичок** — 0–30
- **Мидл** — 31–84
- **Про** — 85–100

UI показывает:

“Женский стиль: Про”

или

“Мужской стиль: Новичок”



🔥 Репутация (Rep) — 3 качественных ранга

Диапазон: -100 ... +100

1) Отрицательная репутация (-100 ... -5)

Название ранга:

🔴 «Токсик»

Описание:

Игрок имеет плохую репутацию в кд соо, его стараются избегать.
Частые провалы, конфликты или низкое качество проектов.

2) Нейтральная репутация (-4 ... +25)**Название ранга:****Описание:**

Игрок пока мало известен.
О нём нет ярко выраженного мнения — ни плохого, ни хорошего.

3) Положительная репутация (+26 ... +100)**Название ранга:****Описание:**

Игрок известен своими проектами, ответственностью и качеством.
К нему относятся положительно и хотят брать в проекты.

8. Навыки

F_skill — женские каверы

M_skill — мужские каверы

Рост:

- тренировки у хореографа
- частично — при успешных проектах
- невозможен через ExtraTraining (они дают прогресс проекту, не навыку)

Падение:

Для сохранения реалистичности танцевального тренинга, если игрок долго тренирует только один стиль, второй стиль постепенно теряет форму.

Логика снижения:

Если игрок **30 дней подряд** качает **только один** навык:

- Если тренируется только F_skill → M_skill уменьшается
- Если тренируется только M_skill → F_skill уменьшается

Формула снижения:

Происходит один раз в месяц:

- Второй навык $x = 0.9$ (-10% в месяц)

Это слабое, но ощутимое смещение, побуждающее игрока поддерживать баланс.

Пояснение в игре:

Это **не событие и не pop-up**, а системная механика, отображаемая только в логике навыков.

9. Проекты

9.1 Типы проектов

1. **Индивидуальные** — проекты игрока
2. **Командные** — предложены командой
 - в списке отображаются с пометкой “**Командный проект**”

9.2 Атрибуты проекта

Каждый проект имеет:

Название

Тип: solo / group

Командный: yes/no

Требуемый навык: F_skill или M_skill

Минимальное значение навыка

Необходимое количество тренировок (например: 8)
Срок выполнения: 1 месяц (fast) или 2–3 месяца (long)
Стоимость 1 тренировки: 150–400 ₽
Костюм: 1000–5000 или 3000–10000
Прогресс: 0–100%
ExtraTraining: 0–3
Training (основные недели): 0–3

Find Project

Filters

- Skill Type: All Types
- Minimum Required Skill: 0
- Maximum Required Skill: 100
- Sort By: Required Skill

Active Projects: 1 / 4

Female Skill: 0

Male Skill: 0

Project	Gender	Required Skills	Duration	Training Cost	Production Cost	Action
BLACKPINK – How You Like That	Женский	10	10 дней	200 ₽	1000 ₽	Принять проект
BTS – Dynamite	Мужской	10	10 дней	200 ₽	1000 ₽	Принять проект
TWICE – The Feels	Женский	15	12 дней	250 ₽	1200 ₽	Принять проект
SEVENTEEN – Left & Right	Мужской	15	12 дней	250 ₽	1200 ₽	Принять проект
ITZY – WANNABE	Женский	20	14 дней	300 ₽	1500 ₽	Принять проект
TOMORROW X TOGETHER – OX1=LOVESONG	Мужской	20	14 дней	300 ₽	1500 ₽	Принять проект
Red Velvet – Psycho	Женский	35	18 дней	10 ₽		
EXO – Love Shot	Мужской	35	18 дней	10 ₽		
aespa – Next Level	Женский	40	20 дней	15 ₽		

9.3 Логика прогресса проекта

Тренировки в проекте назначаются на неделю, а прогресс идёт каждый день.

Обозначим:

WeeklyTrain = Training + ExtraTraining

Дневной прогресс:

DailyProgress = (WeeklyTrain / 7) * ProjectEfficiency

ProjectEfficiency = множитель от усталости:

Tired < 70	→ 1.0
70–89	→ 0.8
90–100	→ 0.5

Ограничение прогресса

Поскольку прогресс рассчитывается из количества обязательных тренировок (целое число), итоговый прогресс **не может превышать 100%**:

```
progress = min(progress, 100)
```

Даже если дневные вычисления дают 101–110%, проект считается завершённым на 100%.

9.4 Завершение проекта

Проект завершён, когда:

```
progress >= 100%  
и  
прошло не больше заданного срока  
и  
выполнено требуемое количество тренировок
```

9.5 Успех проекта

Успех проекта зависит от отношения навыка игрока к минимальному требованию проекта, но даже при низком навыке проект **может быть успешным**, просто с сильным негативным эффектом.

1) Если навык игрока \geq требуемому навыку

Успех = 100%, если не произошло события «Срыв проекта»

(«Срыв проекта» — отдельное случайное событие, которое может отменить успех.)

2) Если навык игрока ниже требуемого менее чем на 20%

Пример: требуемый навык 50, у игрока 45.

Базовый шанс успеха = 60–85%

(интервал задаётся проектом)

Но даже при успехе:

Негативная репутация: -1...-3 (работа выполнена некачественно)

3) Навык игрока ниже требуемого на 20–40%

Шанс успеха = 25–50%

При успехе: репутация не растёт

При провале: репутация -5...-8

4) Навык ниже на 40%+

Шанс успеха: 10–20%

При успехе: сильный минус репутации (-5...-10)

При провале: -10...-15

Итог:

Проект всегда может быть завершён, но **качество выполнения отражается в репутации**, даже при техническом успехе.

9.6 ExtraTraining (доп. тренировки)

Зависит от количества активных проектов:

1 проект → максимум 7 доп. тренировок

2 проекта → максимум 5

3 проекта → максимум 3

4 проекта → максимум 1

5+ проектов → 0

10. Команда

10.1 Вступление в команду — через pop-up

ДОДЕЛАТЬ

Игрок может:

- Принять
- Отклонить

Если отклоняет **3 раза подряд**, команда перестаёт предлагать.

10.2 Предупреждения

Показываются:

- при **втором отказе**
- при **третьем приглашении**

Чтобы игрок не забывал.

10.3 Командные проекты

Находятся в разделе активных проектов и отмечены на карточке пометкой **“Командный проект”**.

Команда предлагает **1 раз в месяц**.

10.4 Предупреждения команды

Команда — строгий социальный элемент. Если игрок пропускает проекты, команда реагирует.

Счётчик отказов:

`refusalCount = 0..3`

Когда показываются предупреждения

1. **После второго отказа подряд**

Игрок получает pop-up:

`“Команда обеспокоена вашим отсутствием активности.”`

`“Если вы откажетесь ещё раз, вы будете исключены.”`

Игрок может закрыть окно только кнопкой “OK”.

2. При следующем (третьем) предложении проекта

Перед тем как игрок нажмёт «Принять» или «Отказать»:

Показывается повышенное предупреждение:

“Это уже третье предложение за последние месяцы.

Мы помним, что вы дважды отказались.

Ещё один отказ – и вас исключат.”

После этого открывается обычное окно проекта.

3. При третьем отказе подряд

Игрок исключён из команды.

refusalCount сбрасывается в 0.

11. Фестивали

Фестиваль — крупное событие, позволяющее игроку получить деньги, репутацию и престиж.

11.1 Генерация фестиваля

Происходит раз в 3 игровых месяца.

Каждый фестиваль имеет:

Количество участников

20–1000 человек

Участники распределяются по категориям:

- Маленький фестиваль (20–100 участников)
- Средний фестиваль (100–400 участников)
- Крупный фестиваль (400–1000 участников)

Призовой фонд (зависит от количества участников)

Тип фестиваля	Участников	Приз	Формула
Малый	20–100	1000–2000 ₽	random(1000–2000)
Средний	100–400	2000–5000 ₽	random(2000–5000)
Большой	400–1000	5000–20000 ₽	random(5000–20000)

Вид фестиваля:

В 90% фестивалей включаются категории:

- Beginner
- Middle
- Pro

Алгоритм:

1. Команда определяется как Beginner/Middle/Pro.
2. Команда попадает в соответствующую категорию.
3. Победитель определяется внутри каждой категории.

В 10% фестивалей категории отключены:

Это «открытый» фестиваль:

- все команды соревнуются между собой,
- выбирается три победителя по максимальным скайлам.

11.2 Определение уровня NPC-конкурентов

При генерации всех NPC-команд фестиваля рассчитывается:

`Skill_NPC_max = max(teamSkill всех участвующих NPC-команд)`

Этот параметр используется для определения:

- сложности фестиваля,
 - вероятности выигрыша игрока.
-

11.3 Механика победы (алгоритм выбора чемпиона)

Игрок и NPC сравниваются по **teamSkill**.

✓ Если у игрока $skill \geq Skill_NPC_max$:

→ Победа гарантирована,
если не произошло событие “Срыв проекта” в фестивальном варианте
(если в будущем мы добавим такой ивент).

✓ Если $skill$ игрока $< Skill_NPC_max$:

Победа определяется формулой:

`ChanceWin = 6% + (Skill_player / Skill_NPC_max)`

Примеры:

skill игрока	skill NPC max	Шанс
40	100	$6\% + 40/100 = 46\%$
70	80	$6\% + 70/80 = 94\%$

20

100

$$6\% + 20/100 = \mathbf{26\%}$$

6% — минимальный шанс «удачи».

11.4 Выбор победителя NPC среди NPC команд

Если определяется победитель внутри NPC-категории (в которой нет игрока):

Победителем становится команда с лучшим:

`teamSkill + random(-5%, +5%)`

Это создаёт реалистичную вариативность без абсурда.

11.5 Награды за фестиваль

При победе:

- Деньги = призовой фонд
 - Репутация: +2...+7 (зависит от размера фестиваля)
 - Популярность: +5...+15
-

12. События

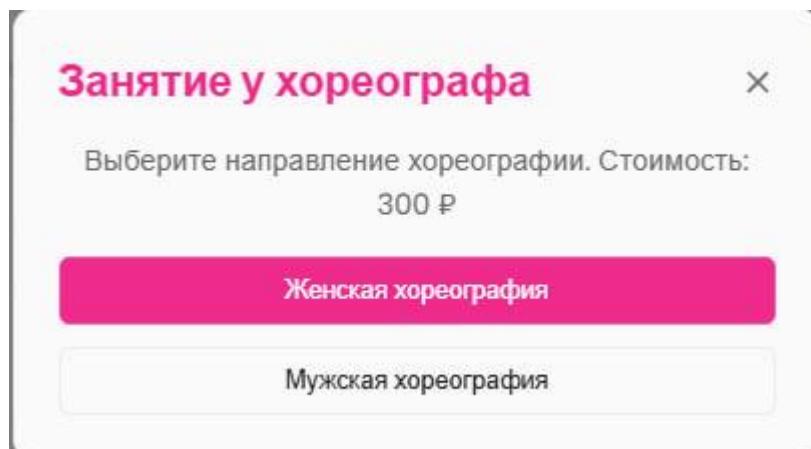
12.1 Случайные события

Примеры:

- Удачная партия (+pop)
- Случайная травма (+tired)

- Застой
- Срыв проекта
- Резкий рост популярности

Каждое событие — через pop-up с остановкой времени.



12.1.1 Событие «Застой»

Застой — редкое негативное событие, которое происходит только если игрок **полностью перестал тренироваться** на длительное время.

Условия активации:

- Игрок **30 дней подряд не проводил ни одной тренировки**:
 - ни у хореографа
 - ни в проектах (Training/ExtraTraining)

То есть застой может наступить даже если игрок «отдыхает».

Эффект события:

`Tired -= 10 (но не ниже 0)`

`F_skill = F_skill × 0.9`

`M_skill = M_skill × 0.9`

Происходит общее «расслабление формы», поэтому навыки падают одинаково.

Текст pop-up:

Вы слишком долго отдыхали и выпали из тренировочного режима.

Тело восстановилось, но навыки немного ослабли.

13. Разделы под будущие системы

13.1 Библиотека NPC-танцоров

NPC – это неигровые танцоры, существующие в мире игры. Они:

- тренируются,
- развиваются,
- могут состоять в командах,
- участвуют в проектах,
- могут взаимодействовать с игроком (в событиях, коллаборациях и т.п.),
- формируют конкурентную среду.

Все вычисления происходят **внутри мировой симуляции**, но NPC **не управляются игроком напрямую**.



13.1.1. Ежегодное появление новых NPC

Каждый игровой год база NPC расширяется.

Правило:

Каждый год добавляется +30-70% от текущего числа NPC.

Пример:

- Старт: 100 NPC

- Через год: 150 NPC
- Через два: 210 NPC
- Через три: 357 NPC ...

Новые NPC создаются по тем же правилам (гендер, имя, стиль, поведение).

После появления новых NPC:

- 60% распределяются в команды (если возможно по правилам совместимости)
 - 40% остаются без команд
-

13.1.2. Структура NPC

Каждый NPC имеет следующие параметры:

```
NPC {  
    id,  
    name,  
    gender,  
    favoriteStyle,  
    behaviorModel,  
    F_skill,  
    M_skill,  
    popularity,  
    reputation,  
    teamId,  
    faceId  
}
```

Пол

- M (муж)
- F (жен)

Распределение:

- 90% NPC создаются женского пола,
- 10% мужского.

Навыки

NPC имеют два типа навыков:

- F_skill (женская хореография), 0–100
- M_skill (мужская хореография), 0–100

Стартовые значения генерируются в диапазоне:

Уровень	Диапазон
---------	----------

Beginner	10–40
----------	-------

Middle	41–70
--------	-------

Pro	71–100
-----	--------

Важно:

- NPC могут расти в навыках со временем.
- Максимальный навык = 100.

Репутация (reputation)

Профессиональный статус в мире танцев.

Диапазон: **-100 → +100**

Меняется от:

- успехов и провалов проектов,
- побед и поражений,
- внутренних игровых событий.

Популярность (popularity)

Показывает известность NPC.

Диапазон: **0–100**

Рост и падение происходят от выступлений, побед и социальных событий.

Любимый стиль

К какому направлению NPC склоняется и что прокачивает чаще.

Варианты:

- женский стиль,
- мужской стиль,
- оба (50/50).

Может меняться 1 раз в 6–12 игровых месяцев, отражая развитие персонажа. Чем выше скилл, тем меньше вероятность, что стиль поменяется (скилл = <30 - шанс 50%, $30<\text{скилл}<60$ - шанс 25%, $60<\text{скилл}<80$ - шанс 7%, $80<\text{скилл}$ - шанс 1%)

Командная принадлежность

NPC может быть в команде:

`teamId = null | team_id`

Команда:

- нужна для участия NPC в командных активностях мира,
 - влияет на его рост и деятельность.
-

13.1.3. Модели поведения NPC

Модель поведения определяет:

- сколько NPC тренируется,
- как быстро растут навыки,
- насколько активно он участвует в жизни мира,
- как социально он себя ведёт.
- меняет значение из диапазона раз в месяц

★ 1. “Горячий” (Burner)

Одержаный тренировками, стабильно и быстро растёт.

```
F_skill += random(+4, +8)  
M_skill += random(+4, +8)  
popularity += random(0, +2)  
reputation += random(+1, +3)
```

★ 2. “Лунатик” (Dreamer)

Работает порывами: либо творческий прорыв, либо провал.

```
F_skill += random(-3, +10)  
M_skill += random(-3, +10)  
popularity += random(-2, +6)  
reputation += random(-3, +4)
```

★ 3. “Холодный перфекционист” (Perfectionist)

Медленно, но качественно растёт, сложный в общении.

```
F_skill += random(+2, +4)  
M_skill += random(+2, +4)  
popularity += random(-2, +1)  
reputation += random(-5, +6)
```

★ 4. “Энергичное солнце” (Sunshine)

Дружелюбный, общительный, автоматически становится популярным.

```
F_skill += random(+1, +3)  
M_skill += random(+1, +3)  
popularity += random(+4, +10)  
reputation += random(+3, +8)
```

★ 5. “Робот-трудоголик” (Machine)

Стабильно тренируется, но нересурсен в социализации.

```
F_skill += random(+3, +6)  
M_skill += random(+3, +6)  
popularity += random(-1, +1)  
reputation += random(+1, +3)
```

★ 6. “Хаос” (Wildcard)

HCI, у которого всё бросается на кубик — навыки, статус, восприятие.

```
F_skill += random(-6, +12)  
M_skill += random(-6, +12)  
popularity += random(-20, +20)  
reputation += random(-15, +15)
```

★ 7. “Хитрый лис” (Fox)

Редко тренируется, но может взлетать в популярности без таланта.

```
F_skill += random(-1, +2)  
M_skill += random(-1, +2)  
popularity += random(+5, +18)  
reputation += random(-8, +4)
```

★ 8. “Темная лошадка” (Silent Pro)

Незаметный, но невероятно стабильный рост.

```
F_skill += random(+3, +7)  
M_skill += random(+3, +7)  
popularity += random(0, +2)  
reputation += random(+1, +4)
```

13.1.4. Команды NPC

NPC могут входить в команды.

- NPC могут менять команды (`random(1%, 30%)` раз в 2 месяца).

NPC не может участвовать в фестивале, если не состоит в команде.

13.1.5 Библиотека NPC-параметров

Каждый NPC создаётся на основе четырёх библиотек:

1. Гендер
2. Имя
3. Любимый стиль
4. Модель поведения

При генерации NPC значения выбираются по заданным вероятностям.

13.1.5.1. Гендер (Gender Library)

Гендер	Вероятность
--------	-------------

Женский (F)	90%
-------------	-----

Мужской (M)	10%
-------------	-----

13.1.5.2. Библиотека имён (Name Library)

Имена делятся на две категории:

A) Никнеймы (уникальные)

Могут появиться только один раз в игре.

Вероятность выбора категории: **60%**

Используются для создания ярких, запоминающихся NPC.

Примеры (можно расширять):

Ame, Nori, Mika, Lira, Koi, Lumi, Rhea, Nyra, Vyn, Sava, Rei, Juno, Anzu, Mina, Kira, Rumi, Aris, Sena, Kiraе, Nova.

Правило:

- ник при генерации NPC удаляется из пула, чтобы он никогда не повторялся.
-

B) Обычные русские имена (можно повторять)

Вероятность выбора категории: **40%**

Используются для более реалистичных NPC.

Женские (примерная библиотека):

Алина, Маша, Дарья, Юлия, София, Олеся, Вика, Лена, Настя, Таня, Марина, Ира, Катя, Света, Полина, Оксана, Лиза, Ася, Карина.

Мужские:

Даня, Артём, Максим, Илья, Лёша, Влад, Костя, Дима, Андрей, Коля, Саша, Игорь, Рома, Тима, Паша.

Повторение допускается.

13.1.5.3. Любимый стиль (FavoriteStyle Library)

Распределение равномерное:

Любимый стиль	Значени	Вероятност
e		ь

Женский стиль F_style 33%

Мужской стиль M_style 33%

Оба стиля Both 33%

Используется для:

- влияния на рост навыков.
-

13.1.5.4. Модели поведения (BehaviorModel Library)

Каждая модель поведения задаёт:

- характер развития,
- изменение навыков,
- изменение популярности,
- изменение репутации,
- внутренний “тип NPC”.

Ниже — финальная библиотека, **8 моделей с вероятностями**.

★ 1. “Горячий” (Burner) — вероятность 18%

Трудолюбивый, одержим тренировками, стабильно растёт.

Формула ежемесячного роста:

```
F_skill += random(+4, +8)  
M_skill += random(+4, +8)  
popularity += random(0, +2)  
reputation += random(+1, +3)
```

★ 2. “Лунатик” (Dreamer) — вероятность 14%

Фантазёр: всплески гениальности перемешиваются с простыми.

```
F_skill += random(-3, +10)  
M_skill += random(-3, +10)  
popularity += random(-2, +6)  
reputation += random(-3, +4)
```

★ 3. “Холодный perfectionist” (Perfectionist) — вероятность 12%

Строгий, ранит и вдохновляет одновременно. Рост аккуратный, репутация полярная.

```
F_skill += random(+2, +4)  
M_skill += random(+2, +4)  
popularity += random(-2, +1)  
reputation += random(-5, +6)
```

★ 4. “Энергичное солнце” (Sunshine) — вероятность 20%

Дружелюбный оптимист, которому легко нравится всем вокруг.

```
F_skill += random(+1, +3)  
M_skill += random(+1, +3)  
popularity += random(+4, +10)  
reputation += random(+3, +8)
```

★ 5. “Робот-трудоголик” (Machine) — вероятность 10%

Стабильная тренировка и низкая социализация.

```
F_skill += random(+3, +6)  
M_skill += random(+3, +6)  
popularity += random(-1, +1)  
reputation += random(+1, +3)
```

★ 6. “Хаос” (Wildcard) — вероятность 8%

Абсолютно непредсказуемый NPC — социальный и прокачной “кубик”.

```
F_skill += random(-6, +12)  
M_skill += random(-6, +12)  
popularity += random(-20, +20)  
reputation += random(-15, +15)
```

⭐ 7. “Хитрый лис” (Fox) — вероятность 10%

Манипулятор, поднимает популярность не талантом, а социальной игрой.

```
F_skill += random(-1, +2)  
M_skill += random(-1, +2)  
popularity += random(+5, +18)  
reputation += random(-8, +4)
```

⭐ 8. “Тёмная лошадка” (Silent Pro) — вероятность 8%

Очень талантливый и очень тихий NPC: растёт сам по себе.

```
F_skill += random(+3, +7)  
M_skill += random(+3, +7)  
popularity += random(0, +2)  
reputation += random(+1, +4)
```

13.2 База NPC-команд

NPC-команды — это самостоятельные танцевальные единицы, которые формируют конкуренцию игроку на фестивалях и в событиях. Каждая команда состоит из NPC из базы 13.1 и имеет свои параметры.

13.2.1. Структура NPC-команды

Каждая команда содержит:

```
Team {  
    teamId,  
    teamName,  
    members: [ npcIds ],  
    teamSkill,           #= среднее значение навыков участников  
    teamLevel,          # Beginner / Middle / Pro  
    popularity,         # средняя популярность команды  
    reputation,         # средняя репутация команды  
}  


---


```

13.2.2. Определение teamSkill

TeamSkill рассчитывается как:

```
teamSkill = mean( (F_skill + M_skill) / 2 ) по всем участникам
```

Например:

- участники имеют 45, 70, 20 среднего навыка → $\text{teamSkill} = (45+70+20)/3 = 45$
-

13.2.3. Определение уровня команды

```
if teamSkill 0–30 → Beginner  
if teamSkill 31–70 → Middle  
if teamSkill 71–100 → Pro
```

Уровень команды определяет:

- в какую категорию фестиваля команда попадает (если категории включены),
 - уровень конкуренции.
-

13.2.4. Участие в фестивале (шанс попадания)

Каждая команда раз в **3 месяца** проходит проверку на участие.

Условие:

чем выше teamSkill, тем выше шанс участия,
но всегда существует **минимальный шанс 7% НЕ участвовать**,
даже у сильнейших команд.

И наоборот — слабая команда всегда имеет шанс попасть в участники.

Формула участия команды:

```
BaseChance = teamSkill * 0.6      # skill 50 → 30% base
```

```
FinalChance = BaseChance * 0.93   # вычитаем 7% гарантированного отказа
```

Но при этом:

Если `random(0..100) < 7%` → команда НЕ участвует (автоматически)

Иначе если `random(0..100) < FinalChance` → команда участвует

Иначе → не участвует

Это гарантирует:

- даже Pro-команда иногда “пропускает” фестиваль (7% шанс),
 - даже Beginner может попасть, пусть и редко.
-

13.2.9. Названия команд

Библиотека может включать:

- Neon Pulse
- Dance Enigma
- Crew Horizon
- Motion Line
- Echo Squad
- Nova Steps
- Rise Unit
- Flux Tribe
- Rhythm Core
- Gravity Beat

Команды не повторяются.

13.2.10. Минимальные требования

- Команда должна состоять **минимум из 3 NPC**.
- Максимум — **20 NPC**.
- Если из команды уходит NPC — команда может быть распущена системой или дебрана из базы.

13.2.11. Распределение NPC по командам при старте

При запуске новой игры создаётся **100 NPC**.

Распределение:

- **60% NPC (≈60 персонажей)** — распределяются в команды

- 40% NPC (\approx 40 персонажей) — остаются без команды
-

Алгоритм распределения NPC в команды (при старте игры)

Команды создаются автоматически. Размер каждой генерируется случайно:

```
minMembers = 3
```

```
maxMembers = 20
```

Количество команд:

```
numCommands = round(0.60 * NPC_total / averageTeamSize)
```

Далее NPC распределяются по командным уровням с учётом навыков и стилей.

■ Правила формирования команд разных уровней

Каждая команда определяется как Beginner / Middle / Pro по среднему уровню участников.

Но чтобы команда получилась реалистичной, **кандидаты в команду выбираются по правилам совместимости**.

★ 1. Команды из NPC-Beginner

Ориентир: низкие навыки, высокий разброс.

Условия подбора участников:

- Разница в среднем скилле между участниками команды может быть **до 40 единиц**
- Ограничений по стилям **нет**
- Команда может быть разношёрстной

Формула проверки:

```
abs(skillA - skillB) ≤ 40
```

```
# стиль не учитывается
```

★ 2. Команды из NPC-Middle

Ориентир: умеренно сильные танцоры.

Условия подбора участников:

- Разница в скилле: максимум **25 единиц**
- Ограничение по стилям:
 - можно: **женский + оба, мужской + оба**
 - нельзя: **женский + мужской**

Формула проверки:

```
abs(skillA - skillB) ≤ 25
```

```
AND not( styleA == "female" AND styleB == "male" )
```

```
AND not( styleA == "male" AND styleB == "female" )
```

★ 3. Команды из NPC-Pro

Ориентир: элита танцоров.

Условия подбора участников:

- Максимальная разница в скилле: **15 единиц**
- Ограничение по стилям:

- можно: **женский + оба, мужской + оба**
- нельзя: **женский + мужской**
- Требуется максимальная однородность

Формула проверки:

```
abs(skillA - skillB) ≤ 15  
AND not( styleA == "female" AND styleB == "male" )  
AND not( styleA == "male" AND styleB == "female" )
```

★ **Как определяется, к какому уровню команда относится**

После набора участников рассчитывается:

```
teamSkill = mean( (F_skill + M_skill) / 2 )
```

Уровень:

0–30 → Beginner

31–70 → Middle

71–100 → Pro

Если команда не попадает в диапазон — она пересобирается автоматически.

13.2.12 ⚡ Процесс генерации команд при старте (сводный алгоритм)

1. Все NPC сортируются по среднему скиллу

(чтобы проще выбирать подходящих друг другу)

```
npcSkill = (F_skill + M_skill) / 2
```

2. Первые команды формируются из слабых NPC → Beginner

Далее Middle → Pro в конце.

3. Для каждой команды выбирается случайный размер (3–20).

4. Для каждого кандидата в команду:

Проверяется:

- подходит ли по Skill-разбросу
- подходит ли по стилю (для Middle/Pro)

Если не подходит → идёт в другую команду или остаётся без команды.

5. Если невозможно набрать Middle/Pro с нужными условиями — NPC не идет в команду.

13.3 Библиотека событий

Отлично — я подготовлю полностью структурированный блок для вставки в ТЗ, в котором будет:

- чёткая структура,
- формулы выпадения,
- условия активации,
- текст поп-апов,
- логика привязки к проектам, NPC, датам выхода каверов,

- отдельная механика для мастер-класса (предупреждение + событие через 30 дней).

Важно: я буду использовать единый формат, максимально удобный для разработчиков.

Каждое событие имеет:

- **TriggerCondition()** — математически формализованные условия появления
 - **Probability()** — вероятность выпадения при выполнении условий
 - **Effects()** — формулы изменений параметров
 - **PopupText** — текст всплывающего окна
 - **PopupType** — *ok, choice, info*
 - **TimeStop = true/false** — останавливает ли событие время
 - **Cooldown** — если нужно, ограничение частоты появления
-

Системные правила появления событий

1. События проверяются **каждый игровой день** в 00:00.
 2. Если событие привязано к проектам — проверка только если есть активные проекты.
 3. Если событие требует NPC — выбирается **не рандомный**, а NPC из **подходящего контекста** (команда, проект, фестиваль).
 4. **Поп-апы блокируют время**, кроме событий типа “информационное уведомление”.
-

Вероятности по категориям событий

Тип события	Базовая вероятность	Примечание
Нейтральные	3–5% в день	Могут выпасть в любое время
Позитивные	1–3% в день	Может меняться из-за репутации
Негативные	1–2% в день	Может меняться из-за tired > 60
События проектов	10–15% шанс после каждого выхода кавера	Только в активном проекте
События NPC	5–10% если NPC присутствует	Контекст обязателен

📌 **Список событий:**

⭐ **1. Удачная партия в проекте**

TriggerCondition():

Если Project.IsActive == true

и DaysSince(Project.CoverReleaseDate) == 1

и RandomChance(12%)

→ событие

Effects():

```
popularity += random(3..7)
```

PopupText:

«Партия в новом кавере особенно удалась! Ваша популярность растёт.»

PopupType: ok

TimeStop: true

★ 2. Попадание в рекомендации

TriggerCondition():

Если DaysSince(CoverReleaseDate) ∈ [2..10]

и RandomChance(4%)

→ событие

Effects():

```
popularity += random(15..35)
```

PopupText:

«Ваш кавер попал в рекомендации! Прирост популярности ощутим.»

PopupType: ok

TimeStop: true

★ 3. Похвала от хореографа

TriggerCondition():

Если Player.TrainedToday == true

и tired < 50

и RandomChance(4%)

Effects():

reputation += random(2..4)

PopupText:

«Хореограф отметил ваши успехи — приятно, и репутация растёт!»

PopupType: ok

★ 4. Мастер-класс (двухэтапное событие)

Этап 1 — Предупреждение

TriggerCondition():

Если RandomChance(1% в месяц)

→ создать MasterClassEvent(date = Today + 30 дней)

→ показать предупреждение

PopupText:

«Через месяц в вашей студии пройдёт мастер-класс известного хореографа! Цена: 2000–5000 ₽.»

PopupType: ok

TimeStop: true

Этап 2 — Само событие через 30 дней

TriggerCondition():

Если Today == MasterClassEvent.date

→ событие

Effects (если игрок соглашается):

```
money -= random(2000..5000)
```

```
Skill(style) += random(5..10)
```

PopupText:

«Начался мастер-класс! Вы хотите участвовать?»

PopupType: choice (Да / Нет)

TimeStop: true

★ 5. Идеальный флоу

TriggerCondition():

Если Training.Today == true

и RandomChance(2%)

Effects():

```
F_skill += random(1..3)
```

или

```
M_skill += random(1..3)
```

PopupText:

«Сегодня всё получается идеально! Навыки растут быстрее.»

TimeStop: true

★ 7. Неожиданный совет от NPC

TriggerCondition():

Существует NPC такой, что:

NPC.ID ∈ Training.TodayParticipants

и

NPC.AvgSkill > Player.AvgSkill

и

RandomChance(3%)

Effects():

F_skill += random(0..2)

или

M_skill += random(0..2)

PopupText:

«NPC дал вам точный совет — прогресс ускорился.»

8. Непривычный формат тренировки

TriggerCondition():

Player.TrainedToday == true

и RandomChance(2%)

и Training.TodayParticipants.Count > 1

Effects():

F_skill += 1

или

M_skill += 1

tired += random(5..10)

PopupText:

«Экспериментальная тренировка была тяжёлой, но полезной.»

★ 10. Травма / болезнь

TriggerCondition():

Если tired > 65

и RandomChance(5%)

Effects:

tired += random(10..20)

TrainingEfficiency *= 0.4 на 5 дней

PopupText:

«Вы чувствуете недомогание. Прогресс заметно замедлится.»

★ 11. Плохой день

TriggerCondition():

Если Player.TrainedToday == true

и RandomChance(3%)

Effects():

Сегодня навыки не растут (эффект отменяет прирост)

PopupText:

«Сегодня ничего не идёт как надо. Прогресса нет.»

★ 12. Конфликт на тренировке

TriggerCondition():

NPC.ID ∈ ActiveProject.Members

где:

ActiveProject.Members — список всех участников текущего проекта,
а не “рандомный NPC с улицы”.

и RandomChance(3%)

Effects():

reputation -= random(2..5)

PopupText:

«Вы вступили в спор на тренировке. Репутация пострадала.»

★ 13. Неудачное выступление

TriggerCondition():

Если Project.IsActive == true

и DaysSince(Project.CoverReleaseDate) == 1

и RandomChance(12%)

→ событие

Effects():

popularity -= random(5..15)

PopupText:

«Что-то пошло не так... Популярность упала.»

★ 14. Негатив в комментариях

TriggerCondition():

Если Player.PostedCover == true

и RandomChance(6%)

Effects():

reputation -= random(3..7)

popularity += random(2..5)

PopupText:

«Перепалка в комментариях подорвала репутацию, но вы стали заметнее.»

★ 15. Самокритика

TriggerCondition():

Если reputation > 10

и RandomChance(2%)

Effects():

TrainingEfficiency *= 0.7 на 7 дней

PopupText:

«Вы слишком требовательны к себе. Прогресс временно замедлится.»

★ 16. Падение мотивации

TriggerCondition():

Если tired > 50

и RandomChance(3%)

Effects():

TrainingEfficiency *= 0.6 на 7 дней

PopupText:

«Мотивация упала, тренировки идут хуже.»

★ 17. Срыв проекта

TriggerCondition():

Если Project.IsActive

и RandomChance(0.5%)

→ проект отменяется

Effects():

Project.Progress = 0

без возврата денег

PopupText:

«Проект сорвался! К сожалению, усилия и деньги потеряны.»

★ 19. Отпуск хореографа

TriggerCondition():

RandomChance(3% в месяц)

Effects():

Trainer.IsAvailable = false на 14 дней

PopupText:

«Ваш хореограф ушёл в отпуск. Он вернётся через 2 недели.»

★ 20. Новые подписчики в Telegram

TriggerCondition():

RandomChance(4%)

Effects:

popularity += random(3..8)

PopupText:

«Ваш TG-канал растёт — появились новые подписчики!»

★ 21. Репост от NPC среднего уровня

TriggerCondition():

Существует NPC такой, что:

NPC.Popularity ∈ [40..70]

и

$\text{NPC.ID} \in (\text{ActiveProject.Members} \cup \text{CurrentFestival.Participants})$

и

$\text{RandomChance}(3\%)$

Effects():

$\text{popularity} += \text{random}(10..25)$

PopupText:

«NPC сделал репост вашего видео — вы стали заметнее!»

22. Знакомство с популярным NPC

TriggerCondition():

Существует NPC такой, что:

$(\text{NPC.ID} \in \text{ActiveProject.Members}$

или

$\text{NPC.ID} \in \text{CurrentFestival.Participants})$

и $\text{NPC.Popularity} > 70$

и $\text{RandomChance}(4\%)$

Effects():

$\text{popularity} += \text{random}(15..35)$

PopupText:

«Вы познакомились с популярным NPC — это поднимает ваш статус!»

★ 23. Коллаб / фото

TriggerCondition():

NPC interacted with Player today

и RandomChance(5%)

Effects():

popularity += random(5..15)

если NPC.reputation < 0:

reputation -= 2

иначе:

reputation += 4

PopUpText:

«Вы сделали совместное фото! Популярность растёт.»

★ 24. Разногласия в команде

TriggerCondition():

Если Player.Team != null

и RandomChance(4%)

Effects():

Team.Reputation -= random(2..5)

или

Player.Reputation -= random(1..3)

PopupText:

«В команде возникли разногласия. Атмосфера ухудшилась.»

★ 26. Командный конфликт

TriggerCondition():

Если Player.Team != null

и RandomChance(2%)

Effects():

Team.Reputation -= random(5..10)

ChanceToBeKicked = 5%

PopupText:

«Серьёзный конфликт в команде. Ваша позиция под угрозой.»

★ 29. Акция студии

TriggerCondition():

RandomChance(4%)

Effects():

- скидка 20–40%
или
- повышение цен 10–30%

PopupText:

«В студии изменилась стоимость занятий.»

★ 31. Много людей на тренировке

TriggerCondition():

Если Training.Today == true

и RandomChance(4%)

Effects():

TrainingEfficiency *= 0.3 на 1 день

PopupText:

«Сегодня зал переполнен — тренировка была малоэффективной.»

★ 32. Вдохновение

TriggerCondition():

RandomChance(1%)

Effects():

TrainingEfficiency *= 1.3 на 30 дней

PopupText:

«Вас охватило вдохновение! Прогресс ускорится на месяц.»

★ 37. Поддержка друга

TriggerCondition():

RandomChance(5%)

Effects():

TrainingEfficiency *= 1.2 на 7 дней

tired -= 2% каждый день (макс -14%)

PopupText:

«Друг поддержал вас — сил и мотивации стало больше!»

13.4 Библиотека комментариев

13.4.1. Момент генерации комментариев

Комментарии создаются, когда выполняется условие:

```
If Project.Progress == 100%
and Project.CoverPublished == true
→ GenerateComments()
```

13.4.2. Количество комментариев

Количество комментариев зависит **только от популярности**:

```
CommentsCount = max(3, floor(popularity / 10))
```

Примеры:

- популярность 5 → 3 комментария
 - популярность 24 → 3 комментария
 - популярность 60 → 6 комментариев
 - популярность 97 → 9 комментариев
-

13.4.3. Логика выбора тональности (позитив/негатив)

Есть только **два типа**: позитив и негатив.

1. Влияние репутации

Если репутация ↓ → негатив ↑

Если репутация ↑ → позитив ↑

Формула:

```
BasePositiveChance = 0.5 + (reputation / 200)
```

```
BaseNegativeChance = 1 - BasePositiveChance
```

Где:

- reputation ∈ [-100..100]
- при rep = 0 → 50% позитив / 50% негатив
- при rep = +100 → 100% позитив
- при rep = -100 → 100% негатив

Значения зажимаются в пределах:

```
BasePositiveChance = clamp(BasePositiveChance, 0.05..0.95)
```

2. Коррекция по скиллу (только для комментариев о скилле)

Если навык игрока ниже требуемого по проекту:

```
SkillPenalty = clamp((RequiredSkill - PlayerSkill) / 100, 0..0.3)
```

Это увеличивает шанс негативного комментария **только про технику**.

3. Итоговая тональность по каждому комментарию

Для каждого комментария:

```
If RandomChance(BasePositiveChance) → PositiveComment  
Else → NegativeComment
```

С поправками:

- если эта фраза относится к технике → шанс негатива увеличивается на SkillPenalty.

13.4.4. Темы комментариев

Теперь по твоему требованию — **темы не привязаны к реальному качеству исполнения**, только:

- техника (скилл),
- подача/атмосфера,
- «субъективка» (зависть, фанатизм, токсичность),
- общая оценка.

И никаких зависимостей от стиля, синхрона и др.

13.4.5. Механика "необъективных комментариев"

Если репутация > 50:

ChanceForBlindPraise = 20%

Тогда комментарий может быть «хвалебным без причин»:

- «Ты всегда идеальна ❤»
- «Даже неважно что танцуешь, все супер!»

Если репутация < -30:

ChanceForBlindHate = 20%

И игрок может получить токсичную фразу:

- «Опять ужас...»
 - «Почему это вообще кто-то смотрит?»
 - «Не понимаю, кто ставит лайки.»
-

13.4.6. Финальная формула выбора комментария

Для каждого комментария:

1. Определить тематику:

Theme = random(skill, atmosphere, subjective, general)

2. Рассчитать BasePositiveChance по репутации

3. Если Theme == skill:

```
PositiveChance -= SkillPenalty
```

4. Применить RandomChance(PositiveChance)

5. Выбрать фразу из соответствующей категории.

Готово. Ниже — полноценная библиотека комментариев, структурированная по:

- Позитивные / Негативные
- Skill / Atmosphere / Subjective / General
- учтены требования:
 - разная длина (15–100 символов),
 - большинство — с маленькой буквы,
 - иногда эмодзи,
 - разные стили речи,
 - длинные комменты (2–4 предложения) — только в негатив / skill.

Можно просто вставить в ТЗ как раздел "Библиотека комментариев".

13.4.7 Библиотека комментариев NPC

13.4.7.1. Позитивные комментарии

13.4.7.1.1. Positive — Skill (похвала техники)

- "очень чисто, прям кайф смотреть"
 - "вау, техника огонь 🔥"
 - "невероятно собрано и аккуратно"
 - "так мощно двигаетесь, респект"
 - "чистюли, супер!"
 - "классная техника, приятно глазу"
 - "все попадания просто идеальны"
 - "очень технично, люблю такое"
 - "идеально сняли акценты"
 - "чёткие линии, супер просто"
-

13.4.7.1.2. Positive — Atmosphere (атмосфера, вайб)

- "какой тёплый вайб 😊"
- "очень атмосферно получилось"
- "прям утонула в вайбе"
- "ваша энергетика топчик"
- "так приятно смотреть, солнышки"
- "видео прям живёт своим настроением"
- "кайфовая атмосфера, обнимаю"
- "ммм, красиво и мягко"

- "люблю ваш вайб, всегда эстетика"
 - "полное погружение, браво"
-

13.4.7.1.3. Positive — Subjective (личные эмоции, ощущение)

- "красота я влюбилась"
 - "богини мои ❤"
 - "съели!"
 - "имбааа"
 - "очень зашло, прям моё"
 - "обожаю ваш стиль"
 - "вы как всегда совершенство"
 - "девочки ангелы, что вы творите"
 - "ooo да, это сильная работа"
 - "просто в точку попало"
-

13.4.7.1.4. Positive — General (универсальные похвалы)

- "умнички 📚"
- "очень красиво получилось"
- "мне прям понравилось"
- "классный кавер!"

- "вы большие молодцы"
 - "очень достойная работа"
 - "милотааа"
 - "здраво, продолжайте"
 - "лучшее, что видела сегодня"
 - "приятное исполнение"
-

13.4.7.2. Негативные комментарии

13.4.7.2.1. Negative — Skill (критика техники)

(разрешены длинные комментарии 2–4 предложения)

- "что-то вообще мимо ритма. выглядит сырьо, как будто толком не чистили."
 - "очень низкий уровень исполнения. расстан разъезжается, точек нет. кажется, что всем тут неудобно танцевать."
 - "ужасно не технично. в целом всё выглядит как набор случайных движений, нет контроля тела."
 - "ощущается неуверенность. есть моменты, где могло бы быть лучше, но всё слишком разлетается."
 - "неаккуратно и неровно, особенно на переходах."
 - "плохой середнячок"
 - "вообще не стараются"
 - "половину движений просто не довели"
 - "движения в кашу"
-

13.4.7.2.2. Negative — Atmosphere

- "атмосферы ноль"
 - "вообще не чувствуется вайб"
 - "очень пусто получилось"
 - "никакого настроения, просто танец"
 - "холодно и без эмоциональности"
 - "скучно, будто без души"
 - "всё плоско как-то"
 - "неверю, не цепляет совсем"
 - "никакой энергии"
 - "пустота полная"
-

13.4.7.2.3. Negative — Subjective

- "что то не так"
- "не моё вообще"
- "как-то неприятно смотреть"
- "не зашло, честно"
- "не нравится стиль"
- "очень странно вышло"
- "выглядит дешево"
- "не вижу смысла в этом кавере"
- "не цепляет ни в одном моменте"
- "ощущение хаоса"

13.4.7.2.4. Negative — General

- "что со съемкой?"
- "ужасный кавер"
- "очень скучный кавер"
- "худший кавер на эту песню"
- "очень слабая работа"
- "сомнительное видео"
- "мне кажется, можно было лучше"
- "как-то не зашло совсем"
- "ну... такое"
- "вообще не стараются"
- "очень слабая подача"
- "синхrona нет вообще"
- "как будто первый раз танцуете..."
- "они вообще не слышат музыку???"
- "какая-то каша движений"
- "руки прям совсем слабые"
- "неаккуратное исполнение"

13.5 Библиотека проектов

13.5.1. Общая структура проекта

Каждый проект в библиотеке имеет следующие параметры:

```
Project {  
    id,  
    name,           // "{K-pop group} — {Song}"  
    durationWeeks,      // длительность (от типа)  
    minSkillRequired_F,    // требуемый навык F_skill  
    minSkillRequired_M,    // требуемый навык M_skill  
    totalTrainingsRequired, // сколько всего тренировок нужно  
    minTrainingsPerWeek,     // 2–3  
    maxTrainingsPerWeek,     // максимум 4  
    isMixedStyle,        // true/false — проект на оба стиля  
    styleProportion,      // если mixed: F:30–70 / M:70–30  
    type,              // fast / long  
    weeksVisible,       // сколько недель проект висит на экране  
}
```

13.5.2. Библиотека проектов

В игре существует **200 уникальных проектов**.
Они **не появляются сразу**, а подгружаются партиями.

13.5.3. Появление новых проектов

Каждую игровую неделю на вкладке проектов отображаются:

+5 новых проектов

- они выбираются из общей библиотеки (200 штук),
 - проекты не повторяются, пока библиотека не исчерпана,
 - старые проекты уходят по правилам исчезновения (ниже).
-

13.5.4. Генерация требований к навыкам — не статическая

Библиотека содержит только:

- Название группы и песни
- Тип проекта (fast / long)
- Признак mixed-style или обычного

Требуемые скиллы НЕ зашиты заранее.

Они рассчитываются каждый раз в момент появления проекта в пуле недели, исходя из текущего уровня игрока.

13.5.5. Алгоритм расчёта уровня сложности

Допустим:

Игрок имеет:

player_F

player_M

$\text{player_avg} = (\text{player_F} + \text{player_M}) / 2$

Проект может быть:

- **однонаправленным** → проверяем только F или только M
 - **смешанным (mixed)** → оба навыка, с пропорциями ниже
-

13.5.5.1. Распределение относительно навыка игрока

Каждую неделю среди **5 новых проектов**:

65% (≈ 3 проекта из 5)

→ требуемый скилл **равен навыку игрока $\pm 7\%$**

`requiredSkill = player_skill ± random(0..7%)`

35% (≈ 2 проекта из 5)

→ проекты далёкие по скиллу:

- **17.5% ниже игрока минимум на 25%**
- **17.5% выше игрока минимум на 25%**

`lowerSkill = player_skill - random(25..40%)`

`higherSkill = player_skill + random(25..40%)`

Расчёт округляется в пользу числа, но сохраняется требование распределения.

13.5.5.2. Расчёт требований для mixed-проектов

10% всех проектов в библиотеке → **mixed-style**.

Для них формула:

`minSkillRequired_F = requiredSkill * ratioF`

`minSkillRequired_M = requiredSkill * ratioM`

где:

$$\text{ratioF} + \text{ratioM} = 1$$

$$\text{ratioF} \in [0.3..0.7]$$

$$\text{ratioM} \in [0.3..0.7]$$

например:

- 30/70
 - 40/60
 - 70/30
(50/50 — тоже валидно)
-

13.5.6. Типы проектов

13.5.6.1 Быстрые проекты (fast)

- Длительность: 2–8 недель
- Тренировки:
 - минимум 2–3 в неделю
 - максимум 4

Общее количество тренировок:

```
totalTrainingsRequired = durationWeeks * random(2..4)
```

-
-

13.5.6.2 Долгие проекты (long)

- **Длительность:** 9–20 недель
- **Тренировки:**
 - минимум 2–3 в неделю
 - максимум 4

Общее количество тренировок:

```
totalTrainingsRequired = durationWeeks * random(2..4)
```

•

13.5.7. Исчезновение проекта

После появления проект находится в списке:

```
weeksVisible += 1
```

Если проект висит **3 недели** и игрок НЕ принял его:

Week 3 → 80% шанс исчезнуть

Если проект остался:

Week 4 → 90% шанс исчезнуть

Если всё ещё остался:

Week 5 → 100% исчезновение

Формально:

```
if weeksVisible == 3:
```

```
    removeChance = 0.8
```

```
elif weeksVisible == 4:
```

```
    removeChance = 0.9
```

```
elif weeksVisible >= 5:
```

```
    removeChance = 1.0
```

```
if random(0..1) < removeChance:
```

```
    project disappears
```

Если принято — проект перестаёт проверяться на исчезновение.

13.5.8. Название проектов

Формат:

"{Группа} — {Песня} | skill: F/M/Mixed "

1. BTS — ECLIPSE | skill: M
2. NCT 127 — Gravity Rush | skill: M
3. Stray Kids — Venomous | skill: M
4. TXT — Night Dive | skill: M
5. EXO — Wicked Bloom | skill: M
6. Ateez — Pirate Code | skill: M
7. Enhypen — Dark Paradigm | skill: M
8. Seventeen — Mirrorlight | skill: M
9. Monsta X — Voltage Line | skill: M
10. The Boyz — Echo Punch | skill: M
11. NCT Dream — Jellyknife | skill: M
12. WayV — Pulse Storm | skill: M
13. Treasure — Jet Fuel | skill: M
14. Pentagon — Arc Driver | skill: M

15. Shinee — Silver Neon | skill: M
16. iKON — Last Sinner | skill: M
17. Astro — Goodnight Fire | skill: M
18. SF9 — Burn It Waves | skill: M
19. Tempest — Cryonix | skill: M
20. BoyNextDoor — Speeding Youth | skill: M
21. EPEX — Thriller Boy | skill: M
22. Xdinary Heroes — Neo Riot | skill: M
23. EVNNE — Overdrive Me | skill: M
24. 8TURN — Red Shell | skill: M
25. Xodiac — HyperKick | skill: M
26. ZEROBASEONE — Rainmaker | skill: M
27. RIIZE — Black Orbit | skill: M
28. ATBO — Jumpcut | skill: M
29. YOUNITE — Fire Switch | skill: M
30. TAN — Combat Kiss | skill: M
31. Kingdom — Spiral Crown | skill: M
32. Mirae — Neon Runner | skill: M
33. BLITZERS — Urban Jam | skill: M
34. Just B — Nightmotor | skill: M
35. TST — Lunar Path | skill: M
36. TRCNG — Code Star | skill: M
37. BAE173 — Zero Trick | skill: M
38. GreatGuys — IronRush | skill: M
39. LUMINOUS — Shadow Step | skill: M
40. WEi — Gravity Burn | skill: M

41. Blackpink — Venom Kiss | skill: F
42. Twice — Cherry Pulse | skill: F
43. Itzy — Cyber Idol | skill: F
44. Le Sserafim — Hot Scar | skill: F
45. IVE — Golden Veil | skill: F
46. NewJeans — Bubble Night | skill: F
47. Aespa — CryoQueen | skill: F
48. (G)I-DLE — Wild Crown | skill: F
49. Red Velvet — Sweet Mirage | skill: F
50. Fromis_9 — Lady Voltage | skill: F
51. Oh My Girl — Blue Cotton | skill: F
52. Apink — Soft Trigger | skill: F
53. Everglow — Neon Blade | skill: F
54. WJSN — Stardust Bloom | skill: F
55. Momoland — CandyFrame | skill: F
56. Purple Kiss — Siren Craft | skill: F
57. StayC — Midnight Ride | skill: F
58. woo!ah! — Pop Drill | skill: F
59. Rocket Punch — Laserbright | skill: F
60. Cherry Bullet — Pixel Riot | skill: F
61. BBGIRLS — Stardriver | skill: F
62. Billlie — Overlight | skill: F
63. NMIXX — FishEye | skill: F
64. Kep1er — Starzoom | skill: F
65. CLASS:y — Bloomshock | skill: F

66. CSR — Lightwish | skill: F
67. tripleS — Motion Error | skill: F
68. IRRIS — Moon Splice | skill: F
69. ARTMS — NightFlower | skill: F
70. ME:I — Pretty Danger | skill: F
71. LIMELIGHT — Vivid Rock | skill: F
72. Weeekly — Jump Daisy | skill: F
73. H1-KEY — Aftermatch | skill: F
74. Dreamcatcher — Phantom Echo | skill: F
75. EL7Z UP — Sugar Rushdown | skill: F
76. PIXY — Darklight Siren | skill: F
77. Lapillus — HyperPretty | skill: F
78. ILY:1 — Crystal Pop | skill: F
79. Queenz Eye — Silver Poison | skill: F
80. Catch The Young — Peach Motion | skill: F
81. KARD — Dual Edge | skill: Mixed (F40/M60)
82. KARD — Shadow Bite | skill: Mixed (F50/M50)
83. KARD — Love Reactor | skill: Mixed (F70/M30)
84. KARD — Hell&Honey | skill: Mixed (F30/M70)
85. BABYMONSTER — Chrome Tears | skill: Mixed (F60/M40)
86. BABYMONSTER — Blue Tiger | skill: Mixed (F30/M70)
87. BABYMONSTER — Synch Knife | skill: Mixed (F50/M50)
88. XG — Neon Pulse | skill: Mixed (F70/M30)

89. XG — Burn Inside | skill: Mixed (F60/M40)
90. Coed Crew Nova — Light/Noise | skill: Mixed (F40/M60)
91. Coed Crew Nova — Plastic Heart | skill: Mixed (F50/M50)
92. Kiss of Life — Gold Melt | skill: Mixed (F70/M30)
93. Kiss of Life — Route99 | skill: Mixed (F60/M40)
94. TripleS LOVElution — Sharp Dream | skill: Mixed (F50/M50)
95. TripleS LOVElution — MoonCry | skill: Mixed (F30/M70)
96. EVNNE × Kep1er — Voltage Love | skill: Mixed (F60/M40)
97. NCT × Aespa — HyperBoost | skill: Mixed (F50/M50)
98. The Boyz × IVE — SweetCrusher | skill: Mixed (F40/M60)
99. TXT × Le Sserafim — VenomLace | skill: Mixed (F30/M70)
100. CSR × YOUNITE — Lavender Flash | skill: Mixed (F60/M40)
101. ITZY × Stray Kids — Wild Synchrony | skill: Mixed (F50/M50)
102. Purple Kiss × ONEUS — Cypher Rose | skill: Mixed (F70/M30)
103. VCHA × RIIZE — Street Bloom | skill: Mixed (F40/M60)
104. PIXY × Blitzers — Red Oracle | skill: Mixed (F30/M70)
105. CLASS:y × Tempest — DanceRush | skill: Mixed (F50/M50)
106. Rocket Punch × WEi — Orbit Sugar | skill: Mixed (F70/M30)
107. IRRIS × MIRAE — Cloud Crash | skill: Mixed (F60/M40)
108. Billlie × The Boyz — Bright Poison | skill: Mixed (F30/M70)
109. EXO × Aespa — Hypercode | skill: Mixed (F50/M50)

110. Dreamcatcher × TXT — Night Spell | skill: Mixed (F70/M30)
 111. tripleS × ATBO — Zero Bloom | skill: Mixed (F60/M40)
 112. IVE × ZEROBASEONE — Lightfall | skill: Mixed (F40/M60)
 113. Enhypen × Twice — BlackPetal | skill: Mixed (F30/M70)
 114. Jungkook × Karina — VoidLips | skill: Mixed (F50/M50)
 115. Giselle × Yeonjun — NeonHook | skill: Mixed (F70/M30)
 116. Haerin × Jisung — Coolline | skill: Mixed (F40/M60)
 117. Chenle × Sullyoon — Mirror Pop | skill: Mixed (F60/M40)
 118. I.N × Lily — Logic Twist | skill: Mixed (F50/M50)
 120. Lee Know × Yuna — Sonic Fever | skill: Mixed (F30/M70)
-

14. Программная логика

Ежедневный тик

1. Считываются тренировки за неделю
2. Пересчитывается дневной прогресс
3. Расходуются деньги
4. Меняется усталость
5. Проверяются события
6. Проверяется срок проектов

Pop-up → пауза времени

Время возобновляется после закрытия окна.

16. UI вкладок и синхронизация

- Все вкладки используют **один и тот же таймер**
 - На любых экранах время течёт
 - При открытии pop-up таймер ставится на паузу
-

17. Итоговые требования

- Полная синхронизация времени
- Проекты **не** продвигаются сами — только через тренировки
- Нет лимита на количество активных проектов
- У каждого проекта — срок **и** количество тренировок
- Командный проект помечается в UI
- Везде есть стоимость действий
- События приоритетны над UI и останавливают время