

```
const TONRODY_DOC = [
{
  id: "core_principles",
  title: "Базовые принципы TONRODY",
  items: [
    "Платформа никогда не играет против пользователей.",
    "Платформа не забирает комиссию с раундов: весь пул всегда уходит победителю.",
    "Единственные обязательные расходы пользователя — это газ сети TON.",
    "Движение TON происходит только между пользователями (и донат-трезори, если юзер сам добровольно донатит).",
    "Контракт — прозрачный арбитр, а не участник игры.",
    "Все “доходы проекта” формируются только из добровольных донатов и кастомизаций (donateroom, эффекты и т.п.)."
  ]
},
{
  id: "modes",
  title: "Режимы игры",
  items: [
    {
      modelId: "coin_flip",
      name: "Coin Flip 1v1",
      type: "1vs1",
      description: "Честный дуэльный режим, где два игрока ставят одинаковую сумму и система честным образом определяет победителя.",
      logic: {
        steps: [
          "Игрок А создаёт лобби, выбирает сторону (орёл/решка), вносит ставку.",
          "Контракт фиксирует лобби: сумма, сторона А, дедлайн, статус 'ожидает второго игрока'.",
          "Игрок В входит в это лобби, вносит такую же сумму, автоматически получает противоположную сторону.",
          "Контракт фиксирует обе ставки и запускает честный рандом.",
          "Контракт определяет победителя и отправляет *весь пул* победителю.",
          "Платформа не получает никакой комиссии, только газ сети списывается с пользователей."
        ]
      }
    },
    {
      modelId: "public_lobby",
      name: "Открытые лобби (N игроков)",
      type: "Nvs1",
      description: "Классические лобби: несколько игроков вносят одинаковую ставку, выигрывает один.",
      logic: {
        steps: [
          "Игроки создают открытый лобби, указывая количество участников и сумму ставки."
        ]
      }
    }
  ]
};
```

```
steps: [
    "Контракт создаёт лобби с заданным количеством слотов и фиксированной ставкой.",
    "Игроки по очереди занимают слоты, внося одинаковый депозит.",
    "Когда количество игроков достигает нужного N, лобби помечается как готовое к розыгрышу.",
    "Запускается честный выбор победителя (рандом / VRF / entropy схема).",
    "Весь пул (N * ставка) отправляется победителю.",
    "Платформа снова ничего не забирает (нет комиссии, только газ)."
]
}

},
{
    modelId: "donate_room",
    name: "Donateroom / приватные лобби",
    type: "Nvs1 + донат",
    description: "Те же условия ставок, но лобби имеет донат-составляющую и расширенную кастомизацию визуальных эффектов.",
    logic: {
        core: [
            "Игровая часть (ставка → пул → победитель) работает так же, как в обычном лобби.",
            "Дополнительно задаётся donationAmount — добровольная доплата за 'шоу и кастомизацию'.",
            "Эта сумма или её часть отправляется в специальный treasury / buyback-кошелёк проекта.",
            "Лобби может быть приватным (доступ по ссылке / инвайту)."
        ],
        customization: [
            "Кастомное отображение игроков: разные аватары, рамки, glow, эффекты.",
            "Анимации поражения: например, свистящий снаряд-базука, прилетающий в аватар проигравшего и оставляющий эффект после лобби.",
            "Визуальные эффекты вокруг аватаров: огоньки, ореолы, шлейфы и др.",
            "В донатруме лобби автоматически закрывается через 10 минут после выхода второго участника после раунда."
        ],
        responsibilities: {
            contract: [
                "Хранит тип лобби: public / private / donate.",
                "Хранит сумму ставки и величину donationAmount.",
                "Обеспечивает приём ставок, выбор победителя и выплату игрового пула победителю.",
                "Перенаправляет donationAmount (или его часть) на адрес treasury / buyback."
            ],
            frontend: [
                "Реагирует на состояние контракта, периодически спрашивая сервер/бэкенд о состоянии лобби.",
                "Отрисовывает все визуальные эффекты (базука, свист, огоньки и т.п.)."
            ]
        }
    }
}
```

"Управляет UX: закрытие лобби, таймеры, звуки, анимации."

]

}

}

}

]

,

{

id: "economy",  
title: "Экономика проекта",  
items: [  
    "Doход проекта формируется только из добровольных донатов в donateroom / приватных лобби и платных кастомизаций.",  
    "Игровые режимы (coin flip, открытые лобби) не содержат комиссии платформы.",  
    "All donations flow directly to the project's treasury-wallet.",  
    "Treasury uses TON for: staking tokens, adding liquidity, future activities and inventories.",  
    "Players understand that they pay not for 'winning chance against the platform', but for show, custom effects and participation in the ecosystem."

]

,

{

id: "token\_model",  
title: "Модель токена и его разблокировки",  
items: [  
    {  
        partId: "token",  
        name: "Token of the project (conditionally TNRD)",  
        description: "Token issued for activity in the ecosystem and supported by the DONATE-traffic and stakeholders.",  
        points: [  
            "Token is not issued immediately as a freely transferable balance.",  
            "All rewards initially fall into a 'lock' (Locker-contract) in the LOCKED status.",  
            "Main sources of tokens: player activity, participation in lobbies, donations, inventories.",  
            "Token can be staked on the market for TON, accumulated in the treasury."  
        ]  
    },  
    {  
        partId: "locker",  
        name: "Locker-contract (token lock)",  
        description: "Contract that stores locked tokens and manages them via unlocking conditions.",  
        points: [  
            "Locker tracks: locked[user] and unlocked[user].",  
            "When rewards are issued, the system calls the reward(user, amount) function."  
        ]  
    }  
]

"Токены физически находятся на балансе Locker, а не на кошельках пользователей.",

"Пока токены в locked, пользователь не может их тратить или продавать.",

"При выполнении условий (например, цена токена достигла \$1 и удерживается) Locker позволяет переводить locked → unlocked." ,

"Пользователь вызывает claim() или аналогичную функцию и получает токены на свой адрес в сети TON."

]

},

{

partId: "oracle",

name: "Оракул цены",

description: "Контракт или внешний источник, поставляющий честную цену токена.",

points: [

"Locker не умеет сам 'знать' цену токена, ему нужен оракул.",

"Оракул хранит текущую цену токена (в TON или \$ через привязку к стейблкоину).",

"Оракул обновляется либо бэкендом, либо отдельной оракул-сетью.",

"Locker использует данные оракула, чтобы проверить условие разблокировки (например, price >= \$1 с учётом TWAP)."

]

},

{

partId: "unlock\_logic",

name: "Логика разблокировки токенов",

description: "Условия, при которых залоченные токены становятся доступными пользователям.",

points: [

"Условие может быть глобальным: 'когда цена токена достигнет \$1 и удержится N блоков/времени'.",

"После выполнения условия устанавливается флаг isGlobalUnlock = true.",

"Дальше сценарий либо: автоматический перенос locked → unlocked всем пользователям, либо — пользователь сам вызывает unlockMyTokens().".

"Важно: правила порога цены, источника цены и логики разблокировки фиксируются при развертывании контракта и публично видны.",

"Админ не должен иметь возможности 'тихо' менять логику разблокировки без мультиsigов и публичных процедур."

]

}

]

},

{

id: "system\_roles",

title: "Роли в системе",

items: [

"Игроки — единственные, кто вносит и выигрывает TON в игровых режимах.",

"Платформа — интерфейс, оркестратор лобби и хранитель фронта/бэка; не участвует как сторонний получатель средств в самих раундах.",

"Treasury — кошелёк/контракт, который получает только донаты и добровольные доплаты.",

"Locker — хранитель залоченных токенов и арбитр их выдачи.",

"Оракул — поставщик информации о цене токена."

]

}

];