

# 积分奖励测试

## RewardPoints 奖励积分系统测试说明文档

✅ 本测试用于验证：用户在真实借贷操作后是否能正确获得积分，并确保积分系统行为符合平台激励规则。

### 测试背景与目的

你正在开发一个链上借贷平台。为了鼓励用户积极参与，你设计了一个\*\*“积分奖励系统”\*\*，当用户在平台上完成借款或还款行为时，会自动获得一定数量的积分奖励。

#### 核心原则：

- 每一次真实的借贷行为（如借钱或还钱）都能获得积分；
- 借得多、借得久、提前还款都能获得更多积分；
- 不奖励清算行为（被动还款）；
- 用户不能无成本“薅羊毛”，因为每次获得积分都要有实际借贷成本。

### 测试涉及角色

角色	说明
deployer	部署合约的人（项目方）
userA	测试用户，模拟参与借贷行为
vault	平台核心合约，用来调用借贷功能
keeper	清算执行人，模拟恶意或违约用户被清算

### 测试准备流程（每次测试前）

1. 部署一个模拟的稳定币（如 USDT），作为借贷资金；
2. 部署一个模拟的白名单系统（允许所有用户借贷）；
3. 部署 `RewardPoints` 合约（记录用户积分）；
4. 部署 `RewardManager` 合约（决定积分如何发放）；

5. 部署 `LendingEngine` 合约（处理借贷操作）；
  6. 设置好各合约之间的连接和权限；
  7. 给 `LendingEngine` 合约转入足够资金，模拟“有钱可借”。
- 

## 每个测试场景说明

### ✅ 场景一：用户借款 → 应获得基础积分

- 用户A使用平台借出 1 枚稳定币；
- 系统立即发放积分；
- 积分数量 = 借出金额 × 每ETH基础奖励比例；
- 系统自动调用 `mintPoints()` 完成积分发放。

### ✅ 场景二：用户提前还款 → 额外获得奖励积分

- 用户A借款后，5天内提前还清；
- 系统在还款成功后，再发一次积分奖励；
- 除了还款对应的积分，还会**额外奖励提前还款的 bonus (+5积分)**；
- 这样鼓励用户诚信、快速还款。

### ✅ 场景三：用户逾期 35 天还款 → 无提前还款奖励

- 用户A借款后，35天后才还；
- 系统仍然会发放还款积分；
- 但**不会再给提前还款奖励**（因为不算提前）；
- 总体逻辑：及时还款更划算。

### ✅ 场景四：用户被清算（keeper强制还款） → 不发放任何积分

- Keeper 强制清算用户债务（模拟违约情况）；
  - 虽然用户债务被还清，但系统不会发积分；
  - 只有主动行为（borrow、repay）才奖励；
  - 清算是被动、惩罚性质的流程。
- 

## 积分计算逻辑

系统使用如下公式决定积分数量：

积分 = 借出金额 × 每 ETH 积分基础值  
+ 借出时间（以天计） × 每天奖励  
+ 是否提前还款 × 提前奖励

备注：测试中使用的奖励参数如下：

- 每借出 1 ETH → 基础积分 10 分；
- 每借款 1 天 → 额外积分 2 分；
- 提前还款 → 奖励额外 5 分；
- 借出金额以 ETH 为单位（即 1ETH = 1e18）。



## 测试目标与验证方式

- 每个测试都会检查系统是否正确调用了 `mintPoints()`；
- 使用 `smock` 工具模拟函数调用、记录是否执行；
- 最后比对用户的积分余额是否为预期值。

## ✅ 测试结论

本次积分系统测试覆盖了所有重要场景：

场景	应发积分	是否测试通过
借款	是	✅ 是
提前还款（<30天）	是	✅ 是
正常还款（>30天）	是	✅ 是
清算（非主动行为）	否	✅ 是

积分逻辑健全，既有激励效果，又可抵御刷分行为，适合上线使用。



## RewardPoints 前端展示 & 积分用途说明文档



### 一、用户操作路径（积分产生流程图）





```

graph TD
  A[用户进入借贷平台] → B[选择借贷产品并输入金额]
  B → C[提交借款交易]
  C → D[LendingEngine 合约处理交易]
  D → E[调用 RewardManager 计算积分]
  E → F[RewardPoints 记录积分]
  F → G[用户积分余额更新]

  subgraph 还款流程
    G2[用户主动还款] → H[再次触发 RewardManager]
    H → I[判断是否提前还款]
    I → J[发放还款积分 + 提前奖励 (可选)]
    J → F
  end
end

```

## 二、前端展示建议

展示位置	内容建议
 我的资产页 (Dashboard)	- 当前积分余额（从 <code>RewardPoints.balanceOf(address)</code> 获取）- 积分历史记录（需链下同步事件）
 借贷详情页	- 借款成功后弹窗显示“您获得了 X 点积分奖励”- 还款时显示“提前还款奖励 +X 点积分”
 积分排行榜页（可选）	- 展示所有用户的积分排行（可配合子图/Graph 链上索引器）
 用户成长页（可选）	- 用户等级系统：积分越高等级越高- 每等级可享不同平台费率 / 权益

## 三、积分用途规划建议

积分并非空投型福利，它代表了用户真实的行为价值。以下是一些可持续、可增长的积分用途设计：

### 1 DAO 治理权重（推荐）

- 积分越多，投票权重越大（可兑换为 veToken 或 DAO voting power）

- 激励真正活跃的用户参与治理

## 2 平台费率抵扣

- 使用积分兑换手续费抵扣券（如借款手续费打折）
- 或者用积分全额抵扣一定额度内的借款利息

## 3 积分商城（积分兑换）

- 兑换平台周边、NFT、门票等虚拟商品
- 兑换稳定币或收益券（配合运营补贴）

## 4 用户等级体系（推荐）

等级	所需积分	权益
LV1	0	普通用户
LV2	1000	借款利率 -0.5%、提额 +5%
LV3	5000	优先抢购新产品、DAO奖励
LV4	10000	白名单优先、Airdrop 资格

## 5 激励长期参与者

- 积分到达一定等级后，可获得“白名单”或“忠诚者徽章”，用于未来 IDO/平台新品优先权



## 四、技术接口补充建议（前后端对接）


### ✓ 链上接口（RewardPoints）

函数	说明
<code>balanceOf(address)</code>	获取用户当前积分余额（前端主调用）
<code>mintPoints(address,uint256)</code>	链上积分发放（由 <code>RewardManager</code> 调用）
<code>RewardPointsUpdated(address,uint256)</code>	发放事件，建议链下订阅更新

### ✓ 链下同步建议

为了前端实时展示积分记录，推荐使用：



-  The Graph / subgraph 工具 → 索引所有 `mintPoints` 调用 & 事件；
-  后端数据库缓存 → 用户积分历史明细、积分变动时间戳；

-  前端实时拉取：通过用户地址查询积分变动记录。

## 五、示意图（UI 展示草图建议）

### 我的积分页（Dashboard）

```





+-----+
|  用户地址：0xAbC...1234 |
|  当前积分余额：1,560 分 |
+-----+
  
```

最近获得：

- ✓ 借款成功 → +100 分（2025-06-15）
- ✓ 提前还款 → +35 分（2025-06-20）
- ✗ 清算（无奖励）—（2025-06-25）

 去积分商城  查看等级权益

## ✓ 六、你的奖励系统核心亮点总结

优点	说明
 安全抗女巫攻击	每次积分都需要真实借贷行为，有资金成本，不可刷积分
 行为驱动设计	借得多、借得久、提前还 → 积分更多，用户行为与平台目标强耦合
 持续激励机制	可无限重复参与，每次借贷都能获得新积分，提升平台粘性
 可用于金融权益	平台内抵扣利率、获取 DAO 投票权、解锁更多额度，积分转化为长期经济价值
 可扩展性强	可轻松对接 NFT、白名单、游戏化道具等新玩法