

# Ball Skill - Comprehensive Fraud Prevention & Chargeback Protection

## Chargeback Prevention Strategy

### 1. Identity Verification Tiers

javascript

```
const VERIFICATION_TIERS = {
  UNVERIFIED: {
    maxDailySpend: 25,
    maxEventEntry: 10,
    payoutDelay: "7 days",
    features: ["basic_events"]
  },
  EMAIL_VERIFIED: {
    maxDailySpend: 50,
    maxEventEntry: 25,
    payoutDelay: "5 days",
    features: ["basic_events", "leaderboards"]
  },
  PHONE_VERIFIED: {
    maxDailySpend: 100,
    maxEventEntry: 50,
    payoutDelay: "3 days",
    features: ["basic_events", "leaderboards", "tournaments"]
  },
  ID_VERIFIED: {
    maxDailySpend: 500,
    maxEventEntry: 200,
    payoutDelay: "24 hours",
    features: ["all_events", "premium_tournaments", "instant_challenges"]
  },
  PREMIUM_VERIFIED: {
    // Bank account verification + government ID + video call
    maxDailySpend: 2000,
    maxEventEntry: 1000,
    payoutDelay: "instant",
    features: ["all_events", "high_stakes", "creator_tools"]
  }
};
```

## 2. Behavioral Risk Scoring

javascript

```
function calculateRiskScore(user, transaction) {
  let riskScore = 0;

  // Spending Pattern Analysis
  const avgSpend = user.payments.last30Days.average;
  if (transaction.amount > avgSpend * 3) riskScore += 25;

  // Loss Pattern Analysis
  const recentLosses = user.gameHistory.last10Games.filter(g => g.result === 'loss').length;
  if (recentLosses >= 7) riskScore += 30; // High loss streak = chargeback risk

  // Time-based Risk
  const hoursSinceLastLoss = (Date.now() - user.lastLoss) / (1000 * 60 * 60);
  if (hoursSinceLastLoss < 2) riskScore += 20; // Emotional spending

  // Device/Location Changes
  if (user.currentDevice !== user.primaryDevice) riskScore += 15;
  if (user.currentLocation.country !== user.registeredCountry) riskScore += 35;

  // Account Age & History
  if (user.accountAge < 30) riskScore += 20; // New accounts higher risk
  if (user.chargebackHistory > 0) riskScore += 50; // Previous chargebacks

  // Payment Method Risk
  if (transaction.paymentMethod === 'new_card') riskScore += 15;
  if (transaction.paymentMethod === 'prepaid_card') riskScore += 25;

  return Math.min(100, riskScore);
}
```

## 3. Pre-Payment Fraud Detection

javascript

```
const FRAUD_PREVENTION_CHECKS = {  
  // Real-time card validation  
  cardValidation: {  
    binCheck: true,      // Bank identification number verification  
    cvvVerification: true, // Security code validation  
    avsCheck: true,      // Address verification system  
    velocityCheck: true  // Multiple rapid transactions  
  },  
  
  // Behavioral analysis  
  behavioralFlags: {  
    rapidFireEntries: "5+ events in 10 minutes",  
    unusualSpending: "3x normal transaction size",  
    newDeviceHighValue: ">$50 from unrecognized device",  
    vpnDetection: "Proxy/VPN usage during payment",  
    multipleFailedCards: "3+ declined cards in session"  
  },  
  
  // Geographic risk  
  geoRisk: {  
    highRiskCountries: ["list of high fraud countries"],  
    vpnDetection: true,  
    locationVelocity: "Geographic impossibility check",  
    timezoneInconsistency: "Device vs payment timezone mismatch"  
  }  
};
```

## Loss Recovery & Emotional Protection

### 1. "Tilt Protection" System

javascript

```

function assessEmotionalState(user) {
  const indicators = {
    lossStreak: user.recentGames.consecutiveLosses,
    spendingVelocity: user.payments.last2Hours.count,
    messagePatterns: analyzeCustomerServiceMessages(user.messages),
    gameplayChanges: user.playstyle.deviationFromNorm
  };

  let emotionalRisk = 0;

  // Consecutive losses increase risk exponentially
  if (indicators.lossStreak >= 3) emotionalRisk += 20;
  if (indicators.lossStreak >= 5) emotionalRisk += 40;
  if (indicators.lossStreak >= 7) emotionalRisk += 60;

  // Rapid spending = emotional/tilt behavior
  if (indicators.spendingVelocity > 5) emotionalRisk += 30;

  // Customer service complaints about "rigged" system
  if (indicators.messagePatterns.includes('conspiracy_language')) emotionalRisk += 25;

  return emotionalRisk;
}

// Automatic interventions based on emotional state
function applyTiltProtection(user, emotionalRisk) {
  if (emotionalRisk > 60) {
    return {
      action: "COOL_DOWN_PERIOD",
      duration: "24 hours",
      message: "Take a break! Your account will be available tomorrow.",
      resources: ["links to responsible gaming", "skill training videos"]
    };
  }

  if (emotionalRisk > 40) {
    return {
      action: "SPENDING_LIMIT",
      maxEntry: user.dailyLimit * 0.5,
      message: "Reduced limits active. Focus on improvement over volume.",
      suggestion: "Try practice mode or watch strategy content"
    };
  }
}

```

```
if (emotionalRisk > 20) {  
  return {  
    action: "WARNING_POPUP",  
    message: "You've had some tough games. Consider taking a break?",  
    options: ["Continue (reduced limits)", "Take 2-hour break", "Practice mode only"]  
  };  
}  
}
```

## 2. Loss Recovery Insurance System

javascript

```

const LOSS_RECOVERY_OPTIONS = {
  // Pre-game insurance (visible to opponents)
  INSURANCE_BASIC: {
    cost: "15% of entry fee",
    coverage: "50% of entry fee returned on loss",
    visibility: "Opponents see insurance icon",
    effect: "Some players may avoid insured opponents"
  },

  INSURANCE_PREMIUM: {
    cost: "25% of entry fee",
    coverage: "75% of entry fee returned on loss",
    visibility: "Opponents see premium insurance icon",
    effect: "High-confidence players may prefer these matches"
  },

  // Post-loss recovery options
  STREAK_BREAKER: {
    trigger: "3+ consecutive losses",
    offer: "Next game entry fee 50% off",
    condition: "Must wait 30 minutes (cool down period)"
  },

  LEARNING_CREDIT: {
    trigger: "5+ consecutive losses",
    offer: "Free coaching session or strategy content unlock",
    benefit: "Turns frustration into education opportunity"
  }
};

// Opponent matching considerations
function adjustMatchmakingForInsurance(player, potentialOpponents) {
  return potentialOpponents.map(opponent => {
    let adjustedPriority = opponent.matchPriority;

    // Some players prefer insured opponents (guaranteed competition)
    if (player.hasInsurance && opponent.preferences.prefers_insured) {
      adjustedPriority += 20;
    }

    // Some players avoid insured opponents (worried about desperation play)
    if (player.hasInsurance && opponent.preferences.avoids_insured) {
      adjustedPriority -= 30;
    }
  });
}

```

```
    }  
  
    return { ...opponent, adjustedPriority };  
  });  
}
```

### 3. Dispute Prevention Documentation

javascript

```

// Automatic documentation for dispute prevention
function generateTransactionProof(eventId, userId) {
  return {
    // Undeniable participation proof
    videoEvidence: {
      shotRecording: `event_${eventId}_user_${userId}_shots.mp4`,
      timestamped: true,
      deviceMetadata: true
    },

    // Explicit consent trail
    consentTrail: [
      { timestamp: "entry", action: "accepted_terms", ipAddress: "...", device: "..." },
      { timestamp: "payment", action: "confirmed_entry_fee", method: "card_ending_1234" },
      { timestamp: "start", action: "began_event", location: "verified" },
      { timestamp: "completion", action: "submitted_results", shots: "documented" }
    ],

    // Technical verification
    technicalProofs: {
      gpsLocation: "verified_coordinates",
      deviceSensors: "motion_data_consistent",
      networkAnalysis: "no_proxy_detected",
      biometricCheck: "shooting_style_matches_profile"
    },

    // Fair play verification
    fairPlayMetrics: {
      opponentRating: "similar_skill_level",
      gameConditions: "standard_rules_applied",
      noAdvantage: "equal_opportunity_verified"
    }
  };
}

```

## Payment Security Database Schema

### Fraud Prevention Tables

```
sql
```



```
CREATE TABLE user_risk_profiles (  
  id SERIAL PRIMARY KEY,  
  user_id INTEGER REFERENCES users(id),  
  
  -- Verification Status  
  email_verified BOOLEAN DEFAULT FALSE,  
  phone_verified BOOLEAN DEFAULT FALSE,  
  id_verified BOOLEAN DEFAULT FALSE,  
  bank_verified BOOLEAN DEFAULT FALSE,  
  video_call_verified BOOLEAN DEFAULT FALSE,  
  
  -- Risk Metrics  
  current_risk_score INTEGER DEFAULT 0,  
  lifetime_risk_score INTEGER DEFAULT 0,  
  chargeback_count INTEGER DEFAULT 0,  
  dispute_count INTEGER DEFAULT 0,  
  
  -- Behavioral Flags  
  emotional_risk_score INTEGER DEFAULT 0,  
  last_tilt_episode TIMESTAMP,  
  consecutive_losses INTEGER DEFAULT 0,  
  rapid_spending_incidents INTEGER DEFAULT 0,  
  
  -- Device & Location Tracking  
  primary_device_id VARCHAR(255),  
  registered_country VARCHAR(3),  
  suspicious_location_changes INTEGER DEFAULT 0,  
  
  updated_at TIMESTAMP DEFAULT NOW()  
);
```

```
CREATE TABLE payment_security_log (  
  id SERIAL PRIMARY KEY,  
  user_id INTEGER REFERENCES users(id),  
  transaction_id VARCHAR(255),  
  
  -- Fraud Check Results  
  risk_score INTEGER,  
  bin_check_result VARCHAR(20),  
  cvv_result VARCHAR(20),  
  avs_result VARCHAR(20),  
  velocity_check BOOLEAN,
```

```

-- Geographic Data
ip_address INET,
country_code VARCHAR(3),
vpn_detected BOOLEAN,
location_risk_score INTEGER,

-- Behavioral Analysis
emotional_state_score INTEGER,
spending_pattern_deviation DECIMAL(4,2),
device_fingerprint VARCHAR(255),

-- Action Taken
action_taken VARCHAR(50), -- APPROVED, DECLINED, MANUAL_REVIEW, DELAYED
manual_review_required BOOLEAN DEFAULT FALSE,

created_at TIMESTAMP DEFAULT NOW()
);

CREATE TABLE loss_recovery_tracking (
  id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES users(id),
  event_id INTEGER REFERENCES events(id),

  -- Insurance Details
  insurance_type VARCHAR(20), -- NONE, BASIC, PREMIUM
  insurance_cost DECIMAL(10,2),
  insurance_payout DECIMAL(10,2),

  -- Recovery Actions
  recovery_type VARCHAR(30), -- STREAK_BREAKER, LEARNING_CREDIT, COOL_DOWN
  recovery_amount DECIMAL(10,2),
  recovery_used BOOLEAN DEFAULT FALSE,

  -- Emotional State Context
  consecutive_losses_at_time INTEGER,
  emotional_risk_score INTEGER,
  time_since_last_win INTERVAL,

  created_at TIMESTAMP DEFAULT NOW()
);

```

## Advanced Chargeback Prevention

## 1. Terms Acceptance with Video Proof

javascript

```
// Enhanced terms acceptance for high-value transactions
function requireEnhancedConsent(transaction) {
  if (transaction.amount > 100 || user.riskScore > 50) {
    return {
      required: [
        "video_selfie_reading_terms",
        "voice_confirmation_of_amount",
        "explicit_gambling_acknowledgment",
        "cooling_off_period_confirmation"
      ],
      cooling_period: "24 hours for first-time high-value transactions"
    };
  }
}
```

## 2. Real-time Dispute Resolution

javascript

```
// Automatic evidence package for disputes
function buildDisputeDefense(transactionId) {
  return {
    participation_proof: getVideoEvidence(transactionId),
    consent_trail: getConsentTimeline(transactionId),
    fair_play_verification: getFairPlayMetrics(transactionId),
    technical_validation: getTechnicalProofs(transactionId),
    precedent_cases: getSimilarDisputeOutcomes(transactionId)
  };
}
```

## 3. Graduated Response System

javascript

```
const DISPUTE_RESPONSE_STRATEGY = {  
  FIRST_DISPUTE: {  
    action: "IMMEDIATE_REFUND",  
    reasoning: "Goodwill gesture, build customer loyalty",  
    flag_account: "MONITOR_CLOSELY"  
  },  
  
  SECOND_DISPUTE: {  
    action: "PARTIAL_REFUND",  
    reasoning: "50% refund + coaching session credit",  
    flag_account: "HIGH_RISK"  
  },  
  
  PATTERN_DISPUTER: {  
    action: "FULL_EVIDENCE_DEFENSE",  
    reasoning: "Clear abuse pattern, defend with full documentation",  
    flag_account: "FRAUD_RISK"  
  }  
};
```

## Business Benefits

### 1. Chargeback Rate Reduction

- Target: <0.5% chargeback rate (industry average is 1-3%)
- Early fraud detection reduces payment processing costs
- Better customer relationships through proactive support

### 2. Customer Retention Through Protection

- Tilt protection prevents destructive spending spirals
- Loss recovery options reduce player frustration
- Educational approach builds long-term engagement

### 3. Legal Protection

- Comprehensive consent documentation
- Video evidence of fair play
- Clear audit trail for regulatory compliance

### 4. Revenue Protection

- Insurance fees create additional revenue stream
- Reduced chargeback fees improve margins
- Premium verification tiers justify higher limits

This system transforms Ball Skill from a "gambling app with basketballs" into a "protected sports education platform" - much harder to dispute and much easier to defend legally.