BBTTCC Territory Module v4.8.1-ENHANCED - User Acceptance Testing Guide

Overview

This UAT guide covers the BBTTCC Territory Control module (v4.8.1-ENHANCED) designed for FoundryVTT v13+ and D&D5e v5.0.0+ compatibility. The module provides scene-based territory management, resource calculation, and faction integration.

System Requirements

- **FoundryVTT**: v13.0+ (tested on v13.348)
- **D&D5e System**: v5.0.0+ (tested on v5.1.2)
- **Module Dependencies**: None required, BBTTCC Factions recommended

Module Installation

1. Ensure folder name matches exactly: `bbttcc-territory`
2. No version suffixes in folder names (NOT `bbttcc-territory-v4.8.1-ENHANCED`)
3. Enable module in FoundryVTT Module Management

---

1. Core Module Functionality Testing

Module Loading and Initialization

Expected Behavior:

- ✅ Module loads without console errors
- ✅ API exposed via `game.modules.get('bbttcc-territory').api`
- ✅ Legacy compatibility via `window.BBTTCCTerritory`
- ✅ Modern initialization patterns with proper error handling

Console Diagnostic Commands

```javascript
```

```javascript
// Check module status
const territoryMod = game.modules.get('bbttcc-territory');
console.log('Module found:', !!territoryMod);
console.log('Module active:', territoryMod?.active);
console.log('API available:', !!territoryMod?.api);

// Test API methods
if (territoryMod?.api) {
    console.log('Available API methods:', Object.keys(territoryMod.api));
}
```

Compatibility Validation

```javascript
javascript

// Check system compatibility
const api = game.modules.get('bbttcc-territory')?.api;
if (api) {
    console.log('Module version:', api.version);
    console.log('System compatibility:', game.system.id === 'dnd5e');
    console.log('FoundryVTT version:', game.version);
}
```

---

2. Territory Manager Application Testing

Opening Territory Manager

Test Steps:

1. **Via API**: `game.modules.get('bbttcc-territory').api.openTerritoryManager()`

2. **Via Scene Controls**: Click territory controls (if GM)

3. **Via Legacy**: `window.BBTTCCTerritory.openTerritoryManager()`

Expected Results:

- ✅ Territory Manager window opens

- ✅ Shows current scene information in header

- ✅ Displays existing territories (if any)

- ✅ Form for claiming new territories visible

- ✅ Professional styling loads correctly

Territory Manager Interface Elements

Required UI Components:

- Header with scene name
- Claim new territory form with fields:
    - Territory Name (required)
    - Controlling Faction dropdown
    - Territory Type selector
    - Territory Size selector
    - Coordinates (X, Y inputs)
    - Description textarea
- Existing territories list with action buttons
- Empty state message when no territories exist

Territory Manager Data Loading

```javascript
// Test territory manager data
const manager = new game.modules.get('bbttcc-territory').api.TerritoryManager();
const data = await manager.getData();
console.log('Territory Manager Data:', {
    scene: data.scene?.name,
    territoriesCount: data.territories?.length || 0,
    factionsCount: data.factions?.length || 0,
    hasError: !!data.error
});
```

3. Territory Creation and Management

Creating Territories via API

```javascript

```

```javascript
// Test territory creation with modern API
const api = game.modules.get('bbttcc-territory').api;

const testTerritory = {
    name: "Test Settlement",
    type: "settlement",
    size: "medium",
    description: "A test territory for UAT",
    coordinates: { x: 100, y: 200 }
};

try {
    const result = await api.claimTerritory(testTerritory);
    console.log('Territory created:', result);
} catch (error) {
    console.error('Territory creation failed:', error);
}
```

Territory Types and Resources

Validate Territory Types:

- settlement: { food: 2, materials: 1, trade: 3 }

- fortress: { materials: 3, military: 4, trade: 1 }

- mine: { materials: 5, trade: 2, food: 0 }

- farm: { food: 5, materials: 1, trade: 2 }

- port: { trade: 4, materials: 2, food: 2 }

- factory: { materials: 4, trade: 3, food: 0 }

- research: { materials: 1, trade: 1, knowledge: 4 }

Size Multipliers:

- small: 0.5x resources

- medium: 1.0x resources

- large: 2.0x resources

- capital: 4.0x resources

Resource Calculation Testing

```
javascript
```

```javascript
// Test resource calculation
const api = game.modules.get('bbttcc-territory').api;

// Test different combinations
const tests = [
    { type: 'settlement', size: 'small' },
    { type: 'fortress', size: 'large' },
    { type: 'mine', size: 'capital' }
];

for (const test of tests) {
    const resources = await api.calculateTerritoryResources(test.type, test.size);
    console.log(`${test.type} (${test.size}):`, resources);
}
```

4. Scene Flag Integration Testing

Territory Data Persistence

Test Scene Flag Storage:

```javascript
// Check how territories are stored in scene flags
const territories = canvas.scene.getFlag('bbttcc-territory', 'territories') || {};
console.log('Scene territories:', territories);

// Validate territory structure
Object.entries(territories).forEach(([id, territory]) => {
    console.log(`Territory ${id}:`, {
        hasRequiredFields: !!(territory.name && territory.type && territory.size),
        hasTimestamps: !!(territory.claimedAt && territory.lastUpdated),
        hasResources: !!territory.resources
    });
});
```

Territory Updates

```javascript
```

```javascript
// Test territory status updates
const api = game.modules.get('bbttcc-territory').api;
const territoryId = 'test-territory-id';

try {
    await api.updateTerritoryStatus(canvas.scene, territoryId, 'contested');
    console.log('Territory status updated successfully');
} catch (error) {
    console.error('Territory update failed:', error);
}
```

5. Faction Integration Testing

Faction Detection

```javascript
// Test faction integration
const factions = game.actors.filter(actor => {
    try {
        return actor.getFlag('bbttcc-factions', 'isFaction') === true;
    } catch (error) {
        return false;
    }
});

console.log('Available factions:', factions.map(f => f.name));
```

Territory Assignment to Factions

Test Steps:

1. Create or identify existing faction

2. Create territory and assign to faction

3. Verify faction territory list updates

4. Test territory removal from faction

```javascript
```

```javascript
// Test faction territory assignment
const faction = game.actors.find(a => a.getFlag('bbttcc-factions', 'isFaction'));
if (faction) {
    const territories = faction.getFlag('bbttcc-factions', 'territories') || [];
    console.log(`Faction ${faction.name} controls ${territories.length} territories`);
}
```

6. Visual System Testing

Territory Visualization

Test Visualization Functions:

```javascript
// Test visualization methods
const module = game.modules.get('bbttcc-territory');

// Test territory color assignment
const colors = ['unclaimed', 'claimed', 'contested', 'occupied'].map(status => ({
    status,
    color: module.getTerritoryColor?.(status) || 'Unknown'
}));
console.log('Territory colors:', colors);
```

Canvas Integration

Expected Behavior:

- ✅ Territory boundaries render as Drawing documents
- ✅ Colors reflect territory status
- ✅ Territory names display correctly
- ✅ Visual updates on territory changes

7. Error Handling and Edge Cases

Input Validation

Test Invalid Inputs:

```javascript
const api = game.modules.get('bbttcc-territory').api;

// Test invalid territory creation
const invalidTests = [
    { name: '', type: 'settlement' }, // Empty name
    { name: 'Test', type: 'invalid-type' }, // Invalid type
    { name: 'Test', type: 'settlement', size: 'huge' } // Invalid size
];

for (const test of invalidTests) {
    try {
        await api.claimTerritory(test);
        console.warn('Should have failed:', test);
    } catch (error) {
        console.log('Correctly rejected:', test, error.message);
    }
}
```

Permission Testing

Test Non-GM Access:

```javascript
// Test GM-only functions (run as non-GM)
if (!game.user.isGM) {
    try {
        await game.modules.get('bbttcc-territory').api.claimTerritory({
            name: 'Unauthorized',
            type: 'settlement'
        });
    } catch (error) {
        console.log('Correctly blocked non-GM:', error.message);
    }
}
```

8. Performance Testing

Large Territory Sets

```javascript
// Test with many territories
async function createManyTerritories(count = 50) {
  const api = game.modules.get('bbttcc-territory').api;
  const startTime = performance.now();

  for (let i = 0; i < count; i++) {
    try {
      await api.claimTerritory({
        name: `Territory ${i}`,
        type: 'settlement',
        size: 'medium'
      });
    } catch (error) {
      console.warn(`Failed to create territory ${i}:`, error);
    }
  }

  const endTime = performance.now();
  console.log(`Created ${count} territories in ${endTime - startTime}ms`);
}
```

9. Template and Styling Testing

Template Loading

```javascript
```

```javascript
// Verify template exists
fetch('modules/bbttcc-territory/templates/territory-manager.html')
  .then(response => {
    console.log('Template status:', response.ok ? 'Found' : 'Missing');
    return response.text();
  })
  .then(html => {
    console.log('Template size:', html.length, 'characters');
  })
  .catch(error => {
    console.error('Template error:', error);
  });
```

CSS Loading

```javascript
// Check if styles are loaded
const stylesheets = Array.from(document.styleSheets);
const territoryCSS = stylesheets.find(sheet =>
  sheet.href && sheet.href.includes('territory-manager.css')
);
console.log('Territory CSS loaded:', !!territoryCSS);
```

10. Integration Testing Scenarios

Complete Workflow Test

```javascript

```

```javascript
// Full territory management workflow
async function fullWorkflowTest() {
    const api = game.modules.get('bbttcc-territory').api;

    console.log('1. Creating territory...');
    const territory = await api.claimTerritory({
        name: 'Integration Test Territory',
        type: 'settlement',
        size: 'medium',
        description: 'Full workflow test territory'
    });

    console.log('2. Opening territory manager...');
    await api.openTerritoryManager();

    console.log('3. Updating territory status...');
    await api.updateTerritoryStatus(canvas.scene, territory.id, 'contested');

    console.log('4. Calculating resources...');
    const resources = await api.calculateTerritoryResources('settlement', 'medium');

    console.log('Workflow test completed:', {
        territory: territory.name,
        status: 'contested',
        resources
    });
}
```

Success Criteria Checklist

Module Loading

- ☐ Module loads without errors
- ☐ API properly exposed
- ☐ Compatibility checks pass
- ☐ Settings registered correctly

Territory Manager

- ☐ Opens without errors
- ☐ Shows correct scene information

- [ ] Form validation works
- [ ] Territory list displays correctly
- [ ] Actions function properly

Territory Operations

- [ ] Territory creation works
- [ ] Resource calculation accurate
- [ ] Status updates persist
- [ ] Scene flags update correctly

Integration

- [ ] Faction integration works
- [ ] Visual rendering functions
- [ ] Canvas updates properly
- [ ] Performance acceptable

Error Handling

- [ ] Invalid input rejected
- [ ] Permissions enforced
- [ ] Graceful error messages
- [ ] No console errors

Common Issues and Solutions

Issue: Territory Manager Shows Blank

Cause: Template not loading or data preparation failure **Solution**: Check console for errors, verify template path

Issue: API Not Available

Cause: Module initialization timing **Solution**: Use `await api.waitForReady()` before API calls

Issue: Scene Flags Not Updating

Cause: Permission issues or flag structure problems **Solution**: Verify GM permissions and scene availability

Issue: Resource Calculations Wrong

Cause: Invalid territory type or size parameters **Solution**: Validate inputs against TERRITORY_TYPES constants

---

Debug Commands Reference

```javascript
// Quick status check
const status = {
    module: !!game.modules.get('bbttcc-territory'),
    active: game.modules.get('bbttcc-territory')?.active,
    api: !!game.modules.get('bbttcc-territory')?.api,
    territories: Object.keys(canvas.scene?.getFlag('bbttcc-territory', 'territories') || {}).length,
    factions: game.actors.filter(a => a.getFlag('bbttcc-factions', 'isFaction')).length
};
console.table(status);

// Clear all territories (for testing)
// WARNING: This will delete all territories in the current scene
// await canvas.scene.unsetFlag('bbttcc-territory', 'territories');
```

Updated for v4.8.1-ENHANCED - Target Environment: FoundryVTT v13.348, D&D5e v5.1.2+