Multi-Device State Management Chart

State Transition Map

Current State	Event/Criteria	How Owner Detects Event	Next State	Owner Action
LOBBY	All players ready + owner clicks start	UI button click (direct)	COUNTDOWN	`setState('countdown')`
COUNTDOWN	30-second timer expires	setTimeout callback (direct)	DEALING	`setState('dealing')` + dea cards
DEALING	Cards dealt to all players	Deal completion callback (direct)	PLAYING	`setState('playing')`
PLAYING	All players submit arrangements	🚨 **NEEDS LISTENER**	ALL_SUBMITTED	`checkAllPlayersSubmitte`
ALL_SUBMITTED	Automatic transition	Immediate (direct)	SCORING	`setState('scoring')`
SCORING	Scoring calculations complete	Scoring callback (direct)	ROUND_COMPLETE	`setState('round_complet`
ROUND_COMPLETE	Next round starts OR tournament ends	UI button click OR logic check (direct)	DEALING OR TOURNAMENT_COMPLETE	`setState(nextState)`
TOURNAMENT_COMPLETE	Tournament ends	Game logic (direct)	LOBBY	`setState('lobby')`

Detection Method Categories

✅ Direct Events (Working)

- **UI Clicks**: Owner directly triggers (start game, next round)
- **Timers**: setTimeout callbacks
- **Callbacks**: Deal completion, scoring completion

🚨 Missing Listeners (Need Implementation)

- **PLAYING → ALL_SUBMITTED**: Need Firestore listener for arrangement submissions

Required Listeners by State

LOBBY State

```javascript
// Owner listens for player joins/leaves
firebase.database().ref(`tables/${tableId}/players`).on('value', (snapshot) => {
  // Update UI, check if ready to start
});
```

PLAYING State ⚠️ MISSING

```javascript
// Owner listens for arrangement submissions
firebase.firestore()
  .collection('currentGames')
  .doc(`table_${tableId}`)
  .onSnapshot((doc) => {
    if (this.isOwner) {
      this.checkAllPlayersSubmitted();
    }
  });
```

All States

```javascript
```

```
// All players listen for state changes
firebase.database().ref(`tables/${tableId}/state`).on('value', (snapshot) => {
  const newState = snapshot.val();
  this.handleStateChange(newState);
});
```

Implementation Status

State Transition	Detection Method	Status
LOBBY → COUNTDOWN	UI Button	✅ Working
COUNTDOWN → DEALING	Timer	✅ Working
DEALING → PLAYING	Callback	✅ Working
PLAYING → ALL_SUBMITTED	**Firestore Listener**	🚨 **Missing**
ALL_SUBMITTED → SCORING	Direct	✅ Working
SCORING → ROUND_COMPLETE	Callback	✅ Working
ROUND_COMPLETE → DEALING/TOURNAMENT_COMPLETE	UI/Logic	✅ Working

Key Insight

The only missing piece is the Firestore listener in PLAYING state that triggers `checkAllPlayersSubmitted()` when arrangements are submitted.