**1. Create TLEI Business Events in PAT Dynatrace for LAUPD App**

**Story Points:** 1 (2 for first-time setup)  
**Description:**  
As a member of the Availability Enablement Pod, I want to create business events for LAUPD App in PAT Dynatrace, so that we can leverage the LEI framework to measure the application's TLEI metrics in PAT. This includes creating events for each critical endpoint/URL identified by the LAUPD App team.

**Acceptance Criteria:**

* Business events created for **all critical endpoints** (validated by LAUPD App team).
* Events configured in Dynatrace PAT using the **LEI framework**.
* Documentation updated in Confluence with LAUPD-specific event definitions.
* Completion within **1 business day** (2 days for first-time setup).

**2. Analyze Event Data in PAT for LAUPD App**

**Story Points:** 2 (3 if complex)  
**Description:**  
As a member of the Availability Enablement Pod, I want to analyze LAUPD App event data in Dynatrace PAT to validate accuracy, remove duplicates, and differentiate between business failures (BFs) and technical failures.

**Acceptance Criteria:**

* Report generated showing:
  + Duplicate events resolved.
  + BFs vs. technical failures clearly tagged.
* LAUPD App team confirms data accuracy.
* Events modified if needed (e.g., filters adjusted).

**3. Promote Business Events to PROD for LAUPD App**

**Story Points:** 1  
**Description:**  
As a member of the Availability Enablement Pod, I want to promote the validated Dynatrace business events setup from PAT to PROD for LAUPD App.

**Acceptance Criteria:**

* PAT configuration **exactly replicated** in PROD Dynatrace.
* Synthetic transactions tested in PROD.
* CHG ticket approved (if required).

**4. Analyze Event Data in PROD for LAUPD App**

**Story Points:** 1  
**Description:**  
As a member of the Availability Enablement Pod, I want to verify PROD event data for LAUPD App matches PAT behavior and expected volumes.

**Acceptance Criteria:**

* PROD data matches PAT patterns (±5% variance).
* LAUPD App team confirms expected volumes.
* MAL FLINCODE table updated for LAUPD App.

**5. Go-Live TLEI for LAUPD App in PROD**

**Story Points:** 1  
**Description:**  
As a member of the Availability Enablement Pod, I want to finalize TLEI onboarding for LAUPD App in PROD.

**Acceptance Criteria:**

* Events **actively capturing data** in PROD.
* Dashboards show LAUPD App metrics.
* CHG ticket closed (if used).

**1. DEV Environment Decommission**

**Story DEV-1: Prepare DEV for Decommission**

* *As* the DevOps team,
* *I want* to deploy the notification banner and test decommission steps in DEV
* *So that* we validate the process before PAT/PROD.  
  **AC:**
* Banner added to Site.master in DEV
* Offline.aspx page tested in DEV
* Tivoli monitoring removed from DEV
* Zero active connections in DEV logs

**Story DEV-2: Full DEV Decommission Dry-Run**

* *As* the QA team,
* *I want* to execute a full decommission (including backups) in DEV
* *So that* we confirm the rollback plan works.  
  **AC:**
* All application files deleted except Offline.aspx
* Backup restored successfully (rollback test)
* CHG ticket updated with DEV results

**2. PAT Environment Decommission**

**Story PAT-1: Notification Banner Deployment in PAT**

* *As* the Product Owner,
* *I want* the decommission notice visible in PAT for 7 days
* *So that* stakeholders can preview and approve.  
  **AC:**
* Banner confirmed in PAT by 3+ stakeholders
* No console errors in PAT
* UAT sign-off obtained

**Story PAT-2: PAT Decommission & Validation**

* *As* the Deployment team,
* *I want* to decommission the app in PAT and validate redirects
* *So that* we mirror the PROD process.  
  **AC:**
* PAT traffic fully redirected to Offline.aspx
* Tivoli integrations removed from PAT
* Performance metrics documented

**3. PROD Environment Decommission**

**Story PROD-1: 30-Day Notification Banner in PROD**

* *As* the Change Manager,
* *I want* the banner deployed to PROD for 30+ days
* *So that* users receive ample warning.  
  **AC:**
* Banner live in PROD with tracking (e.g., Google Analytics)
* Zero support tickets about banner functionality

**Story PROD-2: Final PROD Decommission**

* *As* the Infrastructure team,
* *I want* to remove the app from PROD after the notice period
* *So that* resources are freed.  
  **AC:**
* PROD application files archived and deleted
* Offline.aspx serves all requests (HTTP 200)
* Tivoli decommission confirmed by security

**Story DEV-1.1: Implement Banner in DEV**

* *As a* developer,
* *I want* to add the decommission notice to Site.master in DEV
* *So that* I can test styling and functionality.  
  **AC:**
* Banner appears on all pages in DEV
* Links work (Page=Duty)
* No JavaScript errors

**Story PAT-1.1: Deploy Banner to PAT for UAT**

* *As a* tester,
* *I want* the banner deployed to PAT
* *So that* stakeholders can validate the user experience.  
  **AC:**
* UAT sign-off from 3+ stakeholders
* Mobile/desktop rendering verified

**Story PROD-1.1: Launch Banner in PROD**

* *As the* change manager,
* *I want* the banner live in PROD for 30 days
* *So that* users are notified.  
  **AC:**
* Banner visible in PROD (Google Analytics event tracking)
* Zero support tickets about banner functionality

**2. Offline Page Preparation**

*(Parallel work across environments)*

**Story DEV-2.1: Create Offline.aspx in DEV**

* *As a* developer,
* *I want* to build Offline.aspx in DEV
* *So that* we test the decommissioned state.  
  **AC:**
* Static page shows retirement message
* All links redirect correctly

**Story PAT-2.1: Test Offline Page in PAT**

* *As a* QA analyst,
* *I want* to validate Offline.aspx in PAT
* *So that* we confirm the PROD rollout plan.  
  **AC:**
* 100% of test routes redirect to Offline.aspx
* Load tested with 500+ concurrent users

**3. Application Decommission**

*(Sequential: DEV → PAT → PROD)*

**Story DEV-3.1: Dry-Run Decommission in DEV**

* *As the* ops team,
* *I want* to delete the app in DEV (keeping Offline.aspx)
* *So that* we practice the PROD process.  
  **AC:**
* All files deleted except Offline.aspx, web.config, /Images
* Rollback tested successfully

**Story PAT-3.1: Decommission PAT**

* *As the* deployment team,
* *I want* to remove the app from PAT
* *So that* we mirror PROD’s future state.  
  **AC:**
* Zero references to the app in PAT logs
* Tivoli monitoring removed

**Story PROD-3.1: Final PROD Decommission**

* *As the* infrastructure lead,
* *I want* to retire the app in PROD after 30 days
* *So that* resources are freed.  
  **AC:**
* Legal hold backup completed
* Offline.aspx serves all requests (HTTP 200)

**4. Tivoli Integration Removal**

*(Environment-specific stories)*

**Story DEV-4.1: Remove Tivoli from DEV**

* *As a* security engineer,
* *I want* to disable Tivoli monitoring in DEV
* *So that* we verify dependency cleanup.  
  **AC:**
* Tivoli dashboard shows "No alerts" for DEV

**Story PROD-4.1: Remove Tivoli from PROD**

* *As the* security team,
* *I want* to revoke Tivoli access in PROD
* *So that* the attack surface is reduced.  
  **AC:**
* Security audit confirms deprovisioning

**Pre-Migration Preparation: Investigate Database Write Processes for LAUPD App**

**Story Points:** 3 (medium complexity due to dependency mapping)

**Description:**

As the Database Migration Lead, I want to identify and document all processes that write data to the LAUPD App database, so that we can ensure zero data loss during migration and avoid unintended system behavior post-cutover.

**Tasks:**

1. Audit all applications/services with write access to the database.
2. Document scheduled jobs (Autosys, cron), APIs, and manual processes.
3. Map dependencies between write processes and downstream systems.
4. Classify processes by criticality (e.g., real-time vs. batch).

**Acceptance Criteria:**

* **Comprehensive inventory** of all write processes, including:
  + Application services (e.g., APIs, microservices)
  + Scheduled jobs (Autosys/CRON job IDs and schedules)
  + Manual processes (e.g., admin scripts, ETLs)
* **Dependency diagram** showing interactions with downstream systems.
* **Risk assessment** of each process (e.g., "Real-time payment writes = P0").
* Sign-off from LAUPD App team and DBAs.

**Phase 3: Add Windows Host Filter**

**Summary**: Predict 100% disk capacity dates for Windows hosts only.

**Description**:

* Modify the DQL query to filter dt.host.os.type == "WINDOWS"
* Skip non-Windows hosts during analysis
* Include OS type in violation outputs for reporting

**Acceptance Criteria**:  
✅ DQL query returns only Windows disks (os.type=WINDOWS verified in logs)  
✅ Non-Windows disks logged: "Skipped non-Windows disk: <id>"  
✅ Output includes osType: "WINDOWS" for all violations  
✅ Tested with mixed-OS environment (Linux disks ignored)

**Phase 4: Email Notification for 100% Prediction**

**Summary**: Send actionable emails when disks are predicted to reach 100% capacity.

**Description**:

* Format predictions into HTML tables showing:
  + Disk/host names
  + Current usage %
  + **Days until 100%** (primary metric)
  + Exact predicted date
* Send to storage team with priority tagging

**Acceptance Criteria**:  
✅ Email subject: "[Urgent] <X> disks reaching 100% in <Y> days"  
✅ HTML body contains:

* Table with columns: Disk, Host, Current Usage, Days Left, Predicted Date
* Red highlight for disks with <7 days remaining  
  ✅ Test cases:
* 1+ disks predicted to hit 100% → Email sent
* Zero disks → "All clear" email
* SMTP failure → Error logged in execution