UML Activity Diagram - WebSocket Notification and Recovery (Page 5/6)

API Service	WebSocket Manager	Desktop Application

API Service

○
↓
Job created successfully
↓
Generate job_created event
{job_id, session_id, timestamp}
↓
Send to WebSocket Manager

WebSocket Manager

Receive event notification
↓
Check active WebSocket
connections
↓
◇ Desktop connected?
- yes → Send real-time notification
{type: "job_created", job_id,
session_id}
- no → Queue event for recovery
websockets_events_queue
↓
Set retry schedule
(exponential backoff)
→ ◇

Desktop Application

◇ Received WebSocket event?
- yes → Match event by session_id
↓
Update local record with job_id
↓
Fetch full job details
GET /api/jobs/{job_id}
↓
Save to SQLCipher database
(permanent storage)
↓
Update UI with job status
↓
◉
- no - connection lost → Detect connection loss
↓
Start reconnection attempts
(exponential backoff)
↓
◇ Reconnection successful?
- yes → Establish new WebSocket
connection
JWT + ED25519 dual auth
- no → Continue offline mode
↓
Retry connection periodically
↓
◉

WebSocket Manager (continued)

Connection re-established
↓
Check for orphan Jobs
(created while disconnected)
↓
Query database for missed
events
(last 48 hours)
↓
◇ Orphan jobs found?
- yes → Prepare recovery batch
pending_events_batch
↓
Send recovery summary
{recovered_count: N}
↓
Send batch of missed events
- no → Send connection confirmed
→ ◇

Desktop Application (continued)

Process recovery batch
↓
Update local database
with missed Jobs
↓
Display recovery notification
"X transcriptions recovered"
↓
◉ — Next: Page 6 - Credit
Management and Model
Selection