# **🗓 Sprint Task Breakdown – Version 2.0 (Jeff Reed Study Flow)**

**Timeline:** Dec 6 – Jan 2  
 **Goal:** Guide users through a repeatable 4-step Bible study method (Jeff Reed model) with topic selection, session tracking, journaling, and guided interaction.

## **🌀 Sprint 10: Dec 6–Dec 19**

**Sprint Goal:** Introduce topic selection and structured 4-step study flow UI

### **✅ Frontend Tasks:**

* Create **Topics Screen**: List of AI-generated themes or categories (e.g., Grace, Patience, Forgiveness) — clarify if static or cached dynamic list
* Tap on topic → initiate new study\_session with topic + start Step 1
* Build multi-step study flow UI with a top progress bar, back navigation, and save state handling
* Design editable cards for each step (with dynamic CTA: "Continue" / "Save & Next")
* Implement Step 1: **Scripture View** with inline context (Bible API + static fallback, allow Bible version setting later)
* Implement Step 2: **AI Study Guide**, reuse existing LLM integration with Jeff Reed prompt schema
* Add localization keys for all Jeff Reed UI (EN/HI/ML)
* Fallback handling: show offline/stub content if API fails
* Validate user can navigate forward/backward in steps and preserve inputs

### **✅ Backend Tasks:**

* Firestore schema: study\_sessions (with topic\_id, session\_id, current\_step, timestamps, study\_guide\_id)
* Store step completion timestamps + session resume flag
* Integrate bible-context-provider fallback (static JSON versioned per translation)
* Store topic as normalized ID (linked to static list or LLM-gen cache)
* Associate LLM-generated study\_guide\_id to session

### **✅ DevOps Tasks:**

* Deploy fallback Scripture API via Supabase Edge Function
* Add tests:  
  + Topic flow routing
  + Step save/resume/revisit
  + Fallback logic (Bible API offline)
  + Completed vs in-progress session behavior
* Enable logging for:  
  + Session UUID
  + Step transitions (e.g., 1→2, 2→3)
  + Topic ID tracking + resume events

### **✅ Deliverables:**

* Topics screen integrated with Jeff Reed session flow
* Smooth multi-step experience with resume support
* Inline scripture reader + AI Guide contextually rendered
* Telemetry for topic and session analytics
* Fallback context resolver and tracking

### **⚠️ Dependencies / Risks:**

* Bible API downtime or version mismatch
* Complex UI resume edge cases
* i18n rendering for Malayalam/Hindi context prompts

## **🌀 Sprint 11: Dec 20 – Jan 2**

**Sprint Goal:** Enable journaling, reflection, and cross-session progress tracking

### **✅ Frontend Tasks:**

* Build Step 3: **Think & Discuss** → editable reflection input + fixed prompt list (with autosave)
* Build Step 4: **Apply Principles** → journal editor with goal input (character limit, autosave, optional submit)
* Add "Resume Study" card on Home → visible only if incomplete study\_session exists
* Support offline-first journaling with sync recovery
* Validate light/dark mode + text accessibility (font scaling, screen reader labels)

### **✅ Backend Tasks:**

* Extend study\_sessions with subcollections:  
  + journal\_entries: linked by step + timestamp + truncated content
  + user\_goals: goals + submitted\_at (limit 1–3 per session)
* Resume logic: fetch current\_step + restore previously saved inputs
* Track:  
  + completion %
  + abandoned sessions
  + resumed step count per session

### **✅ DevOps Tasks:**

* Daily reminder logic: local (in-app) + optional push based on unfinished sessions
* Backups: study\_sessions, journal\_entries, user\_goals via scheduled export to storage bucket
* Add analytics events for:  
  + time-per-step
  + step abandonment
  + completed vs resumed ratio

### **✅ Deliverables:**

* Journaling complete for Step 3 & 4
* Progress tracking and resume logic finalized
* Topic tracking included in session and analytics schema
* Reminder + analytics system validated

### **⚠️ Dependencies / Risks:**

* Read/write spikes from verbose journals → enforce content limits
* Session sync issues across multiple devices
* Language prompt formatting bugs (esp. Hindi/Malayalam journaling)