# Sprint 1 REDESIGNED: ConvoOS Foundation

#### **© UPDATED SPRINT GOAL**

Build the **foundational chat identity system** and **MCP schema** that enables ConvoOS to track, differentiate, and manage conversations across platforms with user-controlled summarization.

### SPRINT OVERVIEW

• Sprint Duration: Week 1 (7 days)

• Total Story Points: 74 points

• Focus: Chat Identity + MCP Foundation + User-Controlled Management

## **USER STORY 1: Multi-Platform Chat Identity System**

As a ConvoOS user, I want the system to uniquely identify and track each chat conversation across different AI platforms and accounts

### Acceptance Criteria

- System identifies platform (ChatGPT/Claude/Gemini) + account + specific chat thread
- Detects when user starts new chat vs continues existing chat
- Generates unique chat fingerprints for tracking
- Handles account switching and multi-account scenarios
- Persists chat identity across browser sessions

Task ID	Task Summary	Detailed Description	Story Points	Priority	Туре	Assignee	Depe
COS- 001- T001	Setup ConvoOS Extension Architecture	Create manifest.json v3, folder structure optimized for ConvoOS modules	3	Critical	Development	Backend Dev	-
COS- 001- T002	Design Chat Identity Schema	Define unique identifiers: platform_id, account_hash, thread_id, session_id	2	Critical	Architecture	Backend Dev	-
COS- 001- T003	Build Platform Detection Engine	Detect ChatGPT/Claude/Gemini with URL patterns + DOM signatures	3	Critical	Development	Frontend Dev	COS-
COS- 001- T004	Create Account Detection System	Extract account info from DOM (usernames, profile data, login state)	3	Critical	Development	Frontend Dev	COS-
COS- 001- T005	Implement Chat Thread Identification	Generate unique thread IDs for each conversation, detect thread switches	5	Critical	Development	Backend Dev	COS-
COS- 001- T006	Build Chat Lifecycle Detector	Detect new chat creation vs continuing existing conversation	3	High	Development	Backend Dev	COS-
COS- 001-	Create Chat Fingerprinting	Generate stable fingerprints for chat	2	High	Development	Backend Dev	COS-

Task ID T007	Task Summary System	Detailed Description identification across	Story Points	Priority	Туре	Assignee	Depe
		sessions					
COS- 001- T008	Handle Multi- Account Edge Cases	Account switching, multiple logins, incognito mode handling	3	High	Development	Frontend Dev	COS-
COS- 001- T009	Implement Chat State Persistence	Store chat identity in Chrome storage, handle browser restarts	2	High	Development	Backend Dev	COS-
COS- 001- T010	Test Chat Identity Across Scenarios	Test with multiple accounts, platforms, chat switches, edge cases	3	High	Testing	QA Engineer	COS-

**Story 1 Subtotal: 29 Story Points** 

### **USER STORY 2: MCP Schema Foundation & Context Structure**

As a ConvoOS developer, I want a standardized MCP schema to structure conversation data for cross-AI compatibility

### Acceptance Criteria

- MCP schema defined for conversation context, metadata, and actions
- Chat data stored in MCP-compliant format
- Schema supports extensibility for future AI integrations

- Validation system ensures MCP compliance
- Documentation for MCP structure complete

Task ID	Task Summary	Detailed Description	Story Points	Priority	Туре	Assignee	Dependenci
COS- 002- T001	Research MCP Specification Standards	Study existing MCP patterns, Al context protocols, industry standards	2	Critical	Research	AI Engineer	-
COS- 002- T002	Design ConvoOS MCP Schema v1	Define JSON schema for conversations, metadata, actions, intents	3	Critical	Architecture	AI Engineer	COS-002-T0(
COS- 002- T003	Create MCP Data Models	Implement TypeScript interfaces for MCP structures	2	High	Development	Backend Dev	COS-002-T0(
COS- 002- T004	Build MCP Validator	Validate conversation data against MCP schema, error reporting	2	High	Development	Backend Dev	COS-002-T0(
COS- 002-	Implement MCP	Convert raw chat data to	3	Critical	Development	Backend Dev	COS-002-T00

Task ID	Task Summary	Detailed Description	Story Points	Priority	Туре	Assignee	Dependenci
T005	Conversation	MCP format	Folits				
	Encoder	with metadata					
		enrichment					
		Parse MCP					
cos-	Create MCP	data back to				<b>.</b>	
002-	Context	readable	2	High	Development	Backend	COS-002-T00
Т006	Decoder	format for Al				Dev	
		consumption					
		Support					
		schema					
cos-	Add MCP	evolution,				Backend	
002-	Schema	migration	2	Medium	Development		COS-002-T00
T007	Versioning	paths,				Dev	
		backward					
		compatibility					
		Store MCP-					
cos-		formatted					
002-	Build MCP	data with	3	High	Development	Backend	COS-002-T0(
T008	Storage Layer	efficient		1 11911	Вечеюринен	Dev	000 002 100
		querying					
		capabilities					
cos-	Create MCP	Auto-	1	Medium	Development	Backend	COS-002-T00
002-	Documentation	generate				Dev	
Т009	Generator	schema docs,					
		examples,					

ID	ask )	Task Summary	Detailed Description integration guides	Story Points	Priority	Туре	Assignee	Dependenci
00	0S- 02- 010	Test MCP Schema with Sample Data	Validate schema with real conversation data from multiple platforms	2	High	Testing	AI Engineer	COS-002-T00

Story 2 Subtotal: 22 Story Points

## **USER STORY 3:** User-Controlled Chat Management & Summarization

As a ConvoOS user, I want to select specific chats for summarization and manage my conversation history through an intuitive interface

### Acceptance Criteria

- Popup shows list of detected chats with platform/account info
- User can select individual chats for summarization
- Chat history browser with search and filter capabilities
- Manual summarization trigger with progress feedback
- Per-chat summarization settings and history

Task ID	Task Summary	Detailed Description	Story Points	Priority	Туре	Assignee	Deper
COS- 003- T001	Design Chat Management UI/UX	Wireframes for chat browser, selection interface, management controls	2	High	Design	UI/UX Designer	-
COS- 003- T002	Build Chat Discovery Engine	Scan and detect all available chats across platforms in current session	3	Critical	Development	Frontend Dev	COS-(
COS- 003- T003	Create Chat List Component	Display detected chats with platform icons, account info, timestamps	2	High	Development	Frontend Dev	COS-(
COS- 003- T004	Implement Chat Selection System	Multi-select interface, select all/none, filtering options	2	High	Development	Frontend Dev	COS-(
COS- 003- T005	Build Manual Summarization Trigger	"Summarize Selected Chats" button with confirmation dialog	2	High	Development	Frontend Dev	COS-(
COS- 003- T006	Create Chat Context Capture Engine	Extract full conversation content from selected chats	4	Critical	Development	Frontend Dev	COS-0
COS- 003- T007	Implement Summarization Progress UI	Real-time progress for multi-chat	2	Medium	Development	Frontend Dev	COS-(

Task ID	Task Summary	Detailed Description summarization, cancellation support	Story Points	Priority	Туре	Assignee	Depei
COS- 003- T008	Build Chat History Browser	Browse past conversations, search, filter by platform/date/account	3	Medium	Development	Frontend Dev	COS-0
COS- 003- T009	Add Per-Chat Settings Panel	Summarization preferences, auto- summarize toggles, privacy settings	2	Medium	Development	Frontend Dev	COS-0
COS- 003- T010	Handle Chat Management Edge Cases	Empty chats, deleted chats, permission errors, network issues	2	Medium	Development	Backend Dev	COS-(
COS- 003- T011	Test User Chat Management Flow	End-to-end testing of chat selection, summarization, management	2	High	Testing	QA Engineer	COS-0

**Story 3 Subtotal: 26 Story Points** 



Task	Task	Detailed	Story	Priority	Туре	Assignee	Dependenci
ID	Summary	Description	Points	Priority	туре	Assignee	Dependenci
COS- SYS- 001	Setup ConvoOS Database Schema	Design tables for chats, summaries, users, MCP data with relationships	3	Critical	Development	Backend Dev	COS-002-T0
COS- SYS- 002	Build Cross- Component Message Bus	Event-driven communication between extension components	2	Critical	Development	Backend Dev	COS-001-T0
COS- SYS- 003	Create ConvoOS Configuration System	Global settings, feature flags, platform configurations	2	High	Development	Backend Dev	COS-SYS-00
COS- SYS- 004	Implement Privacy & Security Layer	Data encryption, secure storage, user consent management	3	High	Development	Security Engineer	COS-SYS-00
COS- SYS- 005	Build ConvoOS	Performance metrics, error	2	Medium	Development	Backend Dev	COS-SYS-00

Task ID	Task Summary Monitoring System	Detailed Description tracking, usage analytics	Story Points	Priority	Туре	Assignee	Dependencie
COS- SYS- 006	Create Development & Testing Environment	Hot reload, testing framework, mock data, debugging tools	2	Medium	DevOps	DevOps Engineer	COS-001-T00 <sup>2</sup>
COS- SYS- 007	Setup Continuous Integration Pipeline	Automated testing, build validation, deployment preparation	1	Medium	DevOps	DevOps Engineer	COS-SYS-006

**System Tasks Subtotal: 15 Story Points** 



Task ID	Task	Detailed	Story	Priority	Туре	Assignee	Time
	Summary	Description	Points				Вох
COS- SPIKE- 001	Deep Platform DOM Analysis	Analyze chat threading, account detection, session management	3	Critical	Research	Frontend Dev	6 hours
COS- SPIKE- 002	MCP Protocol Research & Standards	Research existing context protocols, compatibility requirements	2	Critical	Research	Al Engineer	4 hours
COS- SPIKE- 003	Chrome Extension Limitations Study	Storage limits, permission requirements, cross- origin policies	2	High	Research	Backend Dev	4 hours
COS- SPIKE- 004	Multi-Account Detection Feasibility	Technical feasibility of account detection across platforms	2	High	Research	Frontend Dev	4 hours

**Research Tasks Subtotal: 9 Story Points** 

Day	Focus Area	Primary Tasks	Story Points	Key Deliverable
Day	Architecture &	COS-001-T001,T002 + COS-002-	12	ConvoOS architecture
1	Research	T001,T002 + COS-SPIKE-001,002	12	+ MCP schema design
Day	Platform &	COS-001-T003,T004,T008 + COS-	44	Multi-platform account
2	Account Detection	SPIKE-003,004	11	detection working
Day	Chat Identity	COS-001-T005,T006,T007 + COS-	45	Chat identification &
3	System	SYS-001,T002	15	fingerprinting complete
Day	МСР	COS-002-T003,T004,T005,T006 +	14	MCP schema
4	Implementation	COS-SYS-003,T004	14	implementation ready
Day	User Interface	COS-003-T001,T002,T003,T004,T005	14	Chat management UI
5	Development	+ COS-002-T008	14	functional
Day	Chat Capture &	COS-003-T006,T007,T008 + COS-	11	Full chat capture with
6	Integration	SYS-005 + COS-002-T007	''	MCP storage
Day	Tooting & Dolich	All testing tasks + COS-SYS-006,T007	8	Sprint 1 MVP ready for
7	Testing & Polish	+ Bug fixes	0	demo

**Total Sprint Points: 85 points** (Ambitious but foundational)

#### **© UPDATED SUCCESS CRITERIA**

#### **Sprint Level DoD - ConvoOS Foundation**

- ☐ Chat Identity System: Uniquely identifies chats across platforms/accounts
- MCP Schema: Standardized format for conversation data implemented
- **User-Controlled Management**: Users can select and manage specific chats
- Cross-Platform Detection: Works on ChatGPT, Claude, and Gemini
- **Data Persistence**: Chat data survives browser restarts

■ Foundation Ready: Architecture supports future sprint requirements
Critical Questions Answered
■ How does ConvoOS differentiate between chat platforms?
☐ How does ConvoOS track which account is active? ✓
□ How does ConvoOS detect new vs existing chat threads?
■ How does ConvoOS handle chat switching?
■ How does ConvoOS store data in MCP format?
■ How do users control which chats get summarized?

### **CRITICAL DEPENDENCIES FOR FUTURE SPRINTS**

### **Sprint 2 Dependencies**

- Chat Identity System → Required for backend API design
- MCP Schema → Required for backend storage and API contracts
- **User Management** → Required for multi-user backend architecture

#### **Sprint 3 Dependencies**

- MCP Format → Required for agent execution commands
- **Chat Threading** → Required for context continuity in actions

#### **Sprint 4 Dependencies**

- Standardized Context → Required for cross-LLM synchronization
- Action History → Required for context updates and memory sync

# **WHY THIS REDESIGN IS CRITICAL**

- 1. Without Chat Identity: Later sprints can't differentiate between conversations
- 2. Without MCP Schema: No standardized way for Al agents to communicate
- 3. Without User Control: System becomes overwhelming and unusable
- 4. Without Proper Foundation: Technical debt will kill the project

This redesigned Sprint 1 builds the **neural pathways** of ConvoOS - the fundamental systems that everything else depends on.