

DECISIONFLOW

Product Backlog & Delivery Plan

Story map, prioritized backlog, sprint plan and delivery quality gates

Phase 2 - Plan

Product	DecisionFlow
Document code	DF-P2-05
Version	2.0
Status	Final Public Portfolio Edition — Historical Planning Baseline
Author	Min Thu Kyaw
Date	July 2026

Evidence statement

This public portfolio edition distinguishes actual evidence, assumed continuity steps, simulated case-study outcomes, targets, and recommendations. Simulated customers, commercial results, benchmarks, incidents, quotations, and later-lifecycle outcomes are illustrative and must not be interpreted as results from a live commercial product. Evidence labels are retained throughout the package to support transparent and responsible interpretation.

Document Control

Purpose	Translate the MVP strategy into an executable, traceable and incrementally deliverable backlog.
Primary audience	Product, engineering, AI/ML, design, QA, DevOps and scrum stakeholders.
Related documents	DF-P2-01, DF-P2-02, DF-P2-03, DF-P2-04 and DF-P2-06.
Review cadence	Update when scope, evidence, architecture, metrics or major decisions change.
Approval state	Historical phase-gate outcome: Conditional Go to Develop.

Revision History

Version	2.0	Owner	Change
1.0	July 2026	Min Thu Kyaw	Restructured Phase 2 planning package; consolidated and standardized source artifacts.

Contents

- 1. MVP Story Slice
- 2. User Story Map
- 3. Product Backlog: Epics and User Stories
- 4. Delivery Strategy and Sprint Planning
- 5. Cross-Functional Workstream View
- 6. Traceability Rule

MVP End-to-End Value Slice

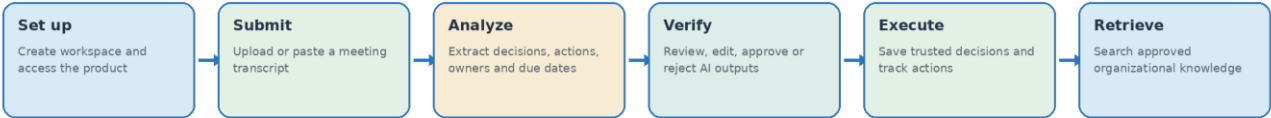


Figure 1. MVP story slice from setup to retrieval.

User Story Map

Purpose

User Story Mapping organizes DecisionFlow’s functionality around the end-to-end user journey rather than isolated features. It provides a shared understanding of how users interact with the product, highlights the minimum workflow required for the MVP, and helps prioritize future enhancements.

The story map serves as a bridge between product strategy and sprint planning.

User Story Mapping Approach

The map is structured in three levels:

- Activities – High-level stages in the user’s journey.
- Tasks – Major user actions within each activity.
- User Stories – Individual capabilities that deliver value.

Primary User

Primary Persona: Product Manager

The same workflow also supports Engineering Managers, Project Managers, Scrum Masters, and Startup Founders.

User Journey Overview

Sign Up	▼ Create Workspace	▼ Upload Meeting	▼ AI Processing	▼ Review
Results	▼ Approve Decisions	▼ Manage Action Items	▼ Search Knowledge	

Story Map

Activity	Tasks	User Stories
Account Setup	Register and create workspace	As a new user, I want to create a workspace so that I can organize meetings for my team.
	Invite teammates	As a workspace owner, I want to invite team members so we can collaborate.
Meeting Management	Upload transcript	As a Product Manager, I want to upload a meeting transcript so AI can analyze it.
	View processing status	As a user, I want to know when AI has finished processing my meeting.

Activity	Tasks	User Stories
AI Analysis	Extract decisions	As a Product Manager, I want AI to identify decisions so I don't review the entire transcript manually.
	Extract action items	As a Project Manager, I want AI to identify follow-up tasks so nothing is missed.
	Detect owners	As a team lead, I want AI to identify task owners so accountability is clear.
	Extract due dates	As a project manager, I want deadlines identified automatically so work stays on schedule.
Review & Approval	Review AI outputs	As a user, I want to verify AI-generated results before saving them.
	Edit results	As a user, I want to correct mistakes so stored information is accurate.
	Approve decisions	As a user, I want approved decisions to become part of our knowledge base.
Execution	View action dashboard	As a manager, I want to see all action items in one place so I can track progress.
	Update action status	As a team member, I want to mark tasks as complete so everyone knows the current status.
Knowledge Management	Search decisions	As a user, I want to find previous decisions quickly instead of reading old meeting notes.
	Filter results	As a user, I want to filter decisions by project, owner, or date.

MVP Story Slice

The MVP focuses on the smallest end-to-end workflow that delivers customer value.

Create Workspace | Upload Transcript | AI Extracts Decisions | Review Results
 | Approve Decisions | Save to Repository | View Action Dashboard

Everything outside this flow is intentionally deferred until after validation.

Future Story Slices

Release 1.1

- Calendar integration
- Email notifications
- Bulk transcript uploads
- Improved search filters

Release 1.2

- Slack integration
- Microsoft Teams integration
- Google Meet integration
- Zoom integration

Release 2.0

- Live meeting assistant
- Real-time transcription
- AI meeting coaching
- Predictive decision analytics
- Cross-meeting insights

Epic Breakdown

Epic 1 – Workspace Management

Stories:

- Create workspace
- Invite members
- Manage settings

Epic 2 – Meeting Processing

Stories:

- Upload transcript
- View processing progress
- Retry failed uploads

Epic 3 – AI Decision Intelligence

Stories:

- Extract decisions
- Extract action items
- Detect owners
- Detect due dates
- Generate confidence scores

Epic 4 – Review & Approval

Stories:

- Review AI output
- Edit results
- Approve or reject decisions
- Approve action items

Epic 5 – Knowledge Repository

Stories:

- Save approved decisions
- Search decisions

- Filter results
- View decision history

Epic 6 – Action Tracking

Stories:

- View dashboard
- Update task status
- Filter by owner
- Filter by project

Story Prioritization

Priority	Stories
P0 (MVP)	Create workspace, Upload transcript, AI decision extraction, AI action extraction, Review AI outputs, Approve decisions, Save decisions, View action dashboard
P1	Due date extraction, Notifications, Advanced filters
P2	Calendar integration, Slack integration, Analytics dashboard
P3	Mobile app, Live meeting assistant, Predictive analytics

Acceptance Criteria Example

User Story

As a Product Manager, I want AI to identify meeting decisions so that I can focus on execution instead of manual documentation.

Acceptance Criteria

- AI extracts all major decisions from the transcript.
- Each decision includes supporting evidence.
- Each decision includes a confidence score.
- Users can edit or delete AI-generated decisions.
- Approved decisions are saved to the repository.

Story Mapping Benefits

The User Story Map provides several advantages:

- Aligns the backlog with the customer journey.
- Identifies the smallest viable product slice.
- Reveals dependencies between features.
- Supports sprint planning and release planning.
- Makes scope discussions easier with stakeholders.
- Keeps development focused on delivering end-to-end user value.

Success Criteria

The User Story Map is considered successful when:

- Every MVP feature supports a user goal.

- Stories are organized around the user journey rather than technical components.
- The MVP story slice delivers complete value from transcript upload to approved decisions.
- Future releases build incrementally after the core workflow demonstrates sufficient pilot evidence.

Summary

The User Story Map translates DecisionFlow's strategy into a user-centered development plan. By organizing work around real user activities instead of disconnected features, the product team ensures that the MVP delivers a complete and valuable workflow. This approach also provides a clear roadmap for future releases while maintaining focus on validating the core product hypothesis.

Product Backlog: Epics and User Stories

Purpose

The Product Backlog translates DecisionFlow's product strategy into actionable work for engineering, design, AI, and QA teams. It organizes development around Epics, Features, and User Stories while defining priorities, acceptance criteria, and implementation dependencies.

The backlog is intended to be a living document that evolves based on customer feedback, technical discoveries, and business priorities.

Backlog Structure

DecisionFlow follows a hierarchical backlog structure:

```
graph TD; Initiative --> Epic; Epic --> Feature; Feature --> User_Story[User Story]; User_Story --> Task
```

Priority Definitions

Priority	Meaning
P0	Critical for MVP launch
P1	High priority, immediately after MVP
P2	Important but can wait
P3	Future enhancement

Story Point Scale

DecisionFlow uses the Fibonacci sequence for estimation.

Story Points	Complexity
1	Very Small

Story Points	Complexity
2	Small
3	Medium
5	Moderate
8	Large
13	Very Large
21	Epic

Story points estimate relative effort rather than time.

Epic 1 – Workspace Management

Goal

Allow teams to create and manage collaborative workspaces.

US-101 — Create Workspace

As a new user
I want to create a workspace
So that I can organize meetings for my team.

Acceptance Criteria

- Workspace name is required.
- Workspace is created successfully.
- User becomes workspace administrator.
- Workspace appears on the dashboard.

Priority	Story Points	Dependencies
P0	3	Authentication

US-102 — Invite Team Members

As a workspace administrator
I want to invite teammates
So that we can collaborate on meeting decisions.

Acceptance Criteria

- Invitations can be sent by email.
- Pending invitations are visible.
- Accepted users join the workspace.

Priority	Story Points	Dependencies
P1	5	Workspace Creation

Epic 2 – Meeting Processing

Goal

Enable users to submit meeting transcripts for AI analysis.

US-201 — Upload Meeting Transcript

As a Product Manager
I want to upload a transcript
So that AI can analyze the meeting.

Acceptance Criteria

- Supports supported file formats.
- Upload validation succeeds.
- Processing starts automatically.
- User receives progress updates.

Priority	Story Points	Dependencies
P0	5	Workspace

US-202 — View Processing Status

As a user
I want to track processing progress
So that I know when results are available.

Acceptance Criteria

- Processing states are displayed.
- Failed processing provides clear error messages.
- Completed processing is indicated automatically.

Priority	Story Points	Dependencies
P0	3	Upload Service

Epic 3 – AI Decision Intelligence**Goal**

Transform unstructured conversations into structured knowledge.

US-301 — Extract Decisions

As a Product Manager
I want AI to identify meeting decisions
So that I do not manually review every transcript.

Acceptance Criteria

- AI extracts major decisions.
- Each decision includes supporting evidence.
- Confidence score is provided.
- Results are editable.

Priority	Story Points	Dependencies
P0	8	AI Processing Pipeline

US-302 — Extract Action Items

As a Project Manager
I want AI to identify follow-up tasks
So that work can be tracked after meetings.

Acceptance Criteria

- Action items are extracted.
- Owners are suggested.
- Due dates are identified when present.
- Tasks are editable.

Priority	Story Points	Dependencies
P0	8	Decision Extraction

US-303 — Identify Owners

As a Team Lead

I want AI to identify responsible people

So that accountability is clear.

Acceptance Criteria

- Owner is identified when available.
- Ambiguous ownership is flagged.
- Confidence score is displayed.

Priority	Story Points	Dependencies
P0	5	AI Extraction

Epic 4 – Review & Approval

Goal

Ensure AI-generated outputs are verified before becoming organizational records.

US-401 — Review AI Results

As a user

I want to review extracted decisions

So that I can confirm their accuracy.

Acceptance Criteria

- All AI outputs are visible.
- Users can edit content.
- Users can reject incorrect results.

US-402 — Approve Decisions

As a user

I want to approve verified decisions

So that they become part of our knowledge base.

Acceptance Criteria

- Approved decisions are saved.
- Rejected decisions are excluded.
- Audit history is recorded.

Priority	Story Points	Dependencies
P0	3	Review Screen

Epic 5 – Knowledge Repository

Goal

Provide long-term access to organizational decisions.

US-501 — Search Decisions

As a user
I want to search historical decisions
So that I can quickly retrieve important information.

Acceptance Criteria

- Search by keyword.
- Search by meeting.
- Search by owner.
- Search results are relevant.

Priority	Story Points	Dependencies
P0	5	Decision Database

US-502 — Filter Decision History

As a user
I want to filter decisions by project and date
So that I can narrow search results.

Acceptance Criteria

- Multiple filters can be combined.
- Results update dynamically.

Priority	Story Points	Dependencies
P1	3	Search Engine

Epic 6 – Action Tracking

Goal

Enable teams to monitor execution after meetings.

US-601 — View Action Dashboard

As a manager
I want to see all action items
So that I can monitor progress.

Acceptance Criteria

- Dashboard displays all active actions.
- Filters by owner and status are available.
- Overdue actions are highlighted.

Priority	Story Points	Dependencies
P0	5	Action Database

US-602 — Update Action Status

As a team member

I want to update task status

So that everyone knows current progress.

Acceptance Criteria

- Status changes are saved immediately.
- Activity history is recorded.
- Dashboard updates automatically.

Priority	Story Points	Dependencies
P1	3	Dashboard

MVP Backlog Summary

Epic	P0 Stories	P1 Stories	P2/P3 Stories
Workspace Management	1	1	0
Meeting Processing	2	0	0
AI Decision Intelligence	3	0	0
Review & Approval	2	0	0
Knowledge Repository	1	1	0
Action Tracking	1	1	0

Total MVP Stories: 10

Estimated Story Points: 50

Definition of Ready (DoR)

A story is ready for development when:

- User story follows the standard template.
- Acceptance criteria are defined.
- Dependencies are identified.
- UX designs are available (if required).
- Engineering questions are resolved.
- Priority has been agreed upon.

Definition of Done (DoD)

A story is complete when:

- Acceptance criteria are met.
- Code has been reviewed and merged.
- Unit and integration tests pass.
- AI evaluation meets quality thresholds (where applicable).
- QA testing is completed.
- Product Manager accepts the implementation.
- Documentation is updated.

Backlog Refinement Process

The backlog will be reviewed:

- Weekly during backlog refinement sessions.
- Before each sprint planning meeting.
- After user research or pilot feedback.
- Following major AI evaluation findings.
- At each roadmap review.

Summary

The Product Backlog converts DecisionFlow's strategic vision into an executable development plan. By organizing work into clearly defined epics and user stories—with priorities, acceptance criteria, dependencies, and story point estimates—the team can deliver incremental value while maintaining focus on the MVP's core objective: helping teams transform meeting conversations into trusted decisions and accountable actions.

Delivery Strategy and Sprint Planning

Purpose

This document defines the delivery approach for DecisionFlow, including the agile methodology, sprint structure, team responsibilities, ceremonies, release cadence, and quality gates. It ensures that product, engineering, design, AI, and QA teams are aligned on how work will be executed from MVP development through future releases.

The delivery strategy emphasizes rapid learning, continuous feedback, and iterative improvement while maintaining high AI quality standards.

Delivery Principles

DecisionFlow follows these delivery principles:

- Deliver customer value in small, incremental releases.
- Validate assumptions early and often.
- Build quality into every sprint.
- Treat AI evaluation as part of development, not a final testing activity.
- Continuously prioritize based on customer feedback and measurable outcomes.

Delivery Methodology

DecisionFlow uses Scrum with two-week sprints.

Sprint Length

- Duration: 2 weeks
- Planning: Day 1
- Review & Demo: Last day
- Retrospective: Last day after review

This cadence provides enough time for AI experimentation while maintaining fast feedback loops.

Team Structure

Role	Responsibilities
------	------------------

Role	Responsibilities
AI Product Manager	Vision, roadmap, backlog, stakeholder alignment
Product Designer	UX, prototypes, usability testing
Full-Stack Engineer	Frontend and backend implementation
AI/ML Engineer	Prompt engineering, model integration, AI evaluation
QA Engineer	Functional, integration, and AI quality testing
DevOps Engineer	Infrastructure, deployment, monitoring

Sprint Goals

Every sprint should deliver a working increment that is demonstrable to stakeholders.

Each sprint must:

- Deliver user-facing value.
- Improve AI quality or platform capability.
- Produce measurable outcomes.
- Update documentation.
- Capture feedback for future planning.

Sprint 0 — Foundation

Goal

Prepare the platform for development.

Deliverables

- Repository setup
- Development environments
- CI/CD pipeline
- Authentication
- Database schema
- Design system
- AI prompt templates
- Evaluation dataset
- Monitoring configuration

Sprint 1 — Workspace & Upload

Goal

Enable users to create workspaces and upload meeting transcripts.

Stories

- User authentication
- Workspace creation
- Team management
- Transcript upload

- Upload validation

Sprint Outcome

Users can upload transcripts successfully.

Sprint 2 — AI Decision Intelligence

Goal

Deliver the first working AI pipeline.

Stories

- Decision extraction
- Action item extraction
- Owner detection
- Due date extraction
- Confidence scoring

Sprint Outcome

AI produces structured meeting outputs.

Sprint 3 — Review Workflow

Goal

Build trust through human validation.

Stories

- Review interface
- Edit AI outputs
- Approval workflow
- Rejection handling
- Audit history

Sprint Outcome

Users can review and approve AI-generated decisions.

Sprint 4 — Knowledge Repository

Goal

Store and retrieve organizational knowledge.

Stories

- Decision repository
- Search
- Filters
- Decision detail view

Sprint Outcome

Approved decisions become searchable.

Sprint 5 — Action Dashboard

Goal

Support post-meeting execution.

Stories

- Action dashboard
- Status updates
- Owner filtering
- Due date tracking

Sprint Outcome

Teams can manage action items after meetings.

Sprint 6 — Stabilization & Beta

Goal

Prepare for pilot release.

Activities

- Bug fixing
- Performance optimization
- AI evaluation
- Security review
- Documentation
- Beta deployment

Sprint Outcome

Beta-ready MVP.

Agile Ceremonies

Ceremony	Frequency	Purpose
Sprint Planning	Every sprint	Define sprint scope
Daily Stand-up	Daily	Remove blockers and synchronize progress
Backlog Refinement	Weekly	Prioritize and clarify upcoming work
Sprint Review	End of sprint	Demonstrate completed work
Sprint Retrospective	End of sprint	Improve team processes

Backlog Management

The Product Backlog is continuously refined based on:

- User feedback
- AI evaluation results
- Technical discoveries

- Business priorities
- Product analytics

Prioritization uses:

- RICE
- MoSCoW
- Strategic alignment
- Customer impact

Definition of Ready (DoR)

A backlog item is ready when:

- User story is complete.
- Acceptance criteria are defined.
- UX design is available (if applicable).
- Dependencies are identified.
- Technical questions are resolved.
- Story has been estimated.

Definition of Done (DoD)

A backlog item is complete when:

- Acceptance criteria are met.
- Code is reviewed and merged.
- Automated tests pass.
- QA testing is complete.
- AI evaluation passes defined thresholds.
- Documentation is updated.
- Product Manager accepts the story.

AI Delivery Workflow



AI quality is validated throughout development rather than only before release.

Quality Gates

Every release must pass four quality gates.

Product Gate

- Stories completed.
- UX acceptance evidence reviewed.
- Stakeholder approval.

Engineering Gate

- Test coverage achieved.
- Performance targets met.
- Security review completed.

AI Gate

- Precision ≥ 90%.
- Recall ≥ 85%.
- Hallucination rate ≤ 5%.
- Confidence calibration verified.

Business Gate

- Documentation complete.
- Analytics instrumentation verified.
- Release communications prepared.

Risk Management

Risk	Mitigation
Scope creep	Freeze MVP scope after Sprint 0
AI quality regression	Continuous benchmark evaluation
Technical blockers	Spike stories and architecture reviews
Delayed feedback	Schedule pilot reviews after every milestone
Integration delays	Decouple integrations from MVP delivery

Communication Plan

Audience	Frequency	Format
Development Team	Daily	Stand-up
Product Stakeholders	Bi-weekly	Sprint Review
Executive Sponsors	Monthly	Roadmap Update
Pilot Customers	Monthly	Feedback Session

Key Delivery Metrics

Metric	Target
Sprint Goal Achievement	≥ 90%
Sprint Predictability	≥ 85%
Story Spillover	≤ 10%
Escaped Defects	< 5 per release
AI Quality Regression	0 critical regressions
Production Incidents	0 critical incidents during beta

Continuous Improvement

At the end of every sprint, the team will:

- Review completed work.
- Analyze AI performance metrics.
- Collect stakeholder feedback.
- Identify process improvements.
- Reprioritize the backlog where necessary.

Insights from retrospectives will directly influence future sprint planning.

Success Criteria

The delivery strategy is successful when:

- MVP is delivered within the planned timeline.
- Sprint commitments are consistently achieved.
- AI quality improves throughout development.
- Stakeholders receive regular, demonstrable progress.
- Customer feedback is incorporated into future iterations.
- The team maintains a sustainable development pace.

Summary

DecisionFlow's delivery strategy combines agile product development with AI-specific evaluation practices to ensure rapid learning without compromising quality. By integrating sprint planning, continuous AI validation, structured quality gates, and regular stakeholder feedback, the team can deliver a reliable MVP while building a scalable foundation for future releases. This execution model aligns product strategy with engineering delivery and supports continuous improvement throughout the product lifecycle.

Cross-Functional Workstream View

Sprint	Product / UX	Platform engineering	AI / data	QA / operations
Sprint 0	Scope freeze, design system, test scripts	Repository, CI/CD, auth, schema	Prompt baseline, annotation guide, dataset	Test plan, monitoring baseline
Sprint 1	Onboarding and upload flows	Workspace and ingestion	Transcript preprocessing	Functional upload tests
Sprint 2	AI result presentation	Job orchestration and result API	Decision/action extraction	Offline evaluation and schema tests
Sprint 3	Review and approval UX	Workflow states and audit events	Confidence/evidence outputs	Human-review and regression tests
Sprint 4	Search and decision detail	Repository and search	Retrieval quality checks	Permission and retrieval tests
Sprint 5	Action dashboard	Action state APIs	Owner/date error analysis	End-to-end acceptance
Sprint 6	Beta onboarding	Hardening and deployment	Final release evaluation	Security, performance and beta readiness

Traceability Rule

Every P0 story should reference at least one PRD requirement, one UX flow or screen, one test case, and one success or AI-quality metric. Work without traceability remains a discovery task rather than a committed delivery item.