

CASE STUDY

Translating a Regulated Digital Experience into Build-Ready Requirements

Behavior-driven requirements and interaction logic for the Dovato Interactive Doctor Discussion Guide

CLIENT / BRAND	ViiV Healthcare / Dovato	AGENCY	Luckie & Company
ROLE	Senior Business Analyst	PRIMARY ARTIFACT	Interactive DDG BDD

THE ENGAGEMENT IN ONE SENTENCE

Turn a complex patient-facing interactive concept into observable, build-ready behavior that creative, product, development, and QA could interpret consistently.

PORTFOLIO NOTE The Dovato DDG is the central case. QA execution was owned by a dedicated QA team; portfolio test scenarios below are derived from the authored BDD and are not presented as historical QA artifacts.

Executive Summary

The DDG combined coach selection, guided questions, testimonials, navigation rules, state persistence, responsive behavior, and a personalized “My Guide” recap. The BA challenge was to preserve the patient experience while making every interaction precise enough to implement and validate.

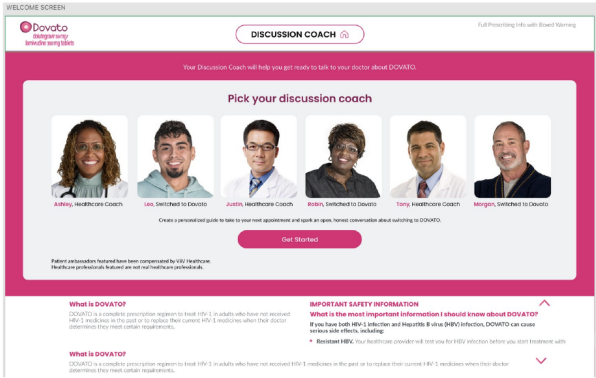
BDD	UX LOGIC	DELIVERY	REGULATED
Given / When / Then	Journey & state	Cross-functional	Healthcare digital

1. Behavior-Driven Requirements

I authored the Dovato Interactive Doctor Discussion Guide BDD and structured the experience around actors, features, and scenario-level Given / When / Then rules. This made design intent observable instead of leaving behavior implicit in creative comps.

2. Representative Requirement — Coach Selection

Actors	End users
Feature	As a user I can select a discussion coach before beginning the guided experience.
BDD Scenarios	Scenario: Coach selection loads Given the user is on the starting screen When the experience begins Then coach options are displayed and a default is used if none is selected Scenario: Questionnaire begins Given the user is on the starting screen When Get Started is selected Then the experience advances to Question 1 Scenario: External information preserves state Given the user has answered one or more questions When external safety or prescribing information is opened Then the current questionnaire state remains intact on return Scenario: Responsive desktop behavior Given the experience is viewed on desktop When the viewport changes within supported desktop sizes Then the layout remains responsive until the tablet breakpoint



Representative interface reference from the original project document

PORTFOLIO EXCERPT • SELECTED SECTIONS ONLY • Jesse Molina

Figure 1. Coach selection and journey-start requirement with representative interface reference

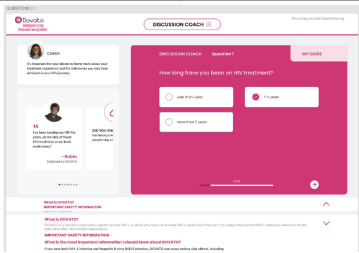
DEFAULT	START	STATE	RESPONSIVE
Coach behavior	Question 1 transition	Preserved on return	Cross-layout intent

2. User Journey, State & Personalized Output

Rather than treating each screen as a standalone page, the requirements described a stateful journey: users move forward and backward, may skip questions, encounter changing testimonials, preserve prior selections, and receive a personalized recap.

3. Representative Requirement — Question Flow

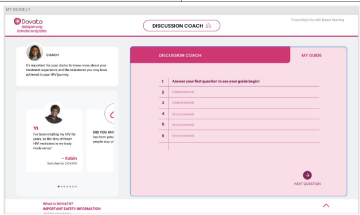
Actors	End users
Feature	As a user I progress through guided questions while contextual content and progress indicators stay synchronized.
BDD Scenarios	Scenario: Journey begins at Question 1 Given the user has started the guide When the questionnaire loads Then Question 1 is displayed Scenario: Context updates with question Given the user views a question with supporting content When the user advances Then supporting content changes to match the active question Scenario: Progress reflects journey state Given the user is within the question flow When the user advances or returns Then the progress indicator updates to the current position Scenario: Next navigation advances Given the user is viewing a question When the next control is selected Then the next question is displayed



Representative interface reference from the original project document

4. Representative Requirement — Personalized Guide Recap

Actors	End users
Feature	As a user I can review a personalized recap without losing my current place in the guided experience.
BDD Scenarios	Scenario: Open My Guide Given the user is on a question page When the My Guide view is opened Then the recap displays responses captured so far Scenario: Identify unanswered items Given one or more questions have no response When the recap is displayed Then unanswered items are visibly distinguished Scenario: Preserve surrounding context Given the user switches between the question view and My Guide When the tab changes Then only the intended content area changes Scenario: Mobile mirrors desktop intent Given the user is on a supported mobile device When the recap is opened Then the same core recap behavior remains available in a responsive layout



Representative interface reference from the original project document

PORTFOLIO EXCERPT • SELECTED SECTIONS ONLY • Jesse Molina

PORTFOLIO EXCERPT • SELECTED SECTIONS ONLY • Jesse Molina

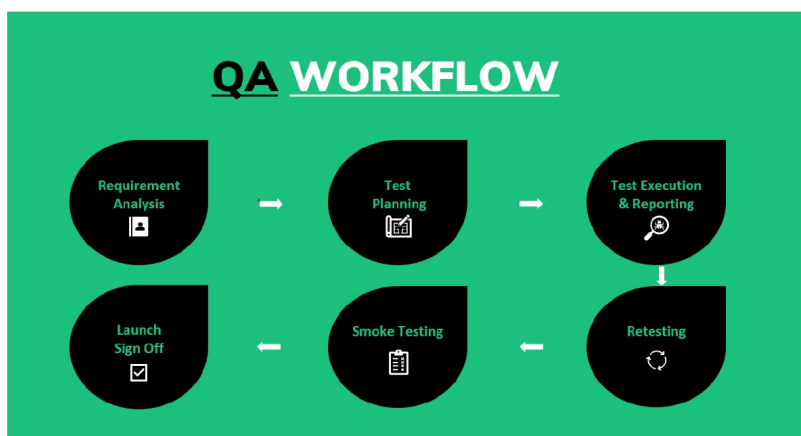
Figures 2–3. Representative question-flow and personalized “My Guide” requirements

- Back-navigation behavior was explicit rather than assumed.
- Skipped-question logic included a reminder after repeated non-response.
- Question screens used consistent interaction patterns while allowing content/testimonial variation.
- The recap view reflected selected answers and supported revisiting earlier responses.

3. Requirements Designed for Downstream Validation

Luckie maintained a full QA function, and I was not the hands-on QA/UAT executor for the DDG. The BDD nevertheless gave QA a direct basis for validation because each scenario stated a precondition, user action, and expected outcome. Luckie's standards linked requirement analysis to RTMs, test-case development, Jira/Zephyr execution, regression, and release sign-off.

1.6. Testing Workflow in Jira



When a ticket gets assigned to be worked on by QA, it is first reviewed to make sure the requirements are all understood. If the requirements are not clear or the ticket does not have the necessary supporting documents attached, QA will assign the ticket back to the PM with a comment asking for clarification as needed.

The next step in the process is Test Planning; this is defining the testing scope, strategy, tools, where, and how to test.

Then comes the Test Execution and Reporting step; part of this process includes a Zephyr ticket being created or automatically generated against the ticket and linked. Test cases are created in this stage against the accepted requirements in the ticket and executed.

If issues are found during testing, the ticket is moved to the Dev ("To Do" Lane) with the screenshots and listed issues in ticket. The Zephyr ticket is then moved to "the QA on hold" Lane for retesting later with an executed "failed" status.

Figure 4. Luckie QA workflow in Jira — broader delivery context

Illustrative Validation Scenarios

The following examples show how the BDD can be translated directly into validation logic. They are portfolio illustrations, not historical QA deliverables.

Representative scenarios

Scenario	Representative expected behavior
Coach selection	No selection → configured default coach is used.
Journey start	Get Started → Question 1 displays.
State persistence	External information → prior answers remain on return.
Skipped questions	Two skips → reminder appears.
Back navigation	Back arrow → prior question without corrupting state.
My Guide	Selected answers → applicable recap content is shown.

Portfolio distinction: these validation examples were created from the requirements for this case study; they are not original QA artifacts.

4. A Repeatable Way of Working at Luckie

The Dovato documentation pattern fits a broader way of working visible across other Luckie engagements: translate goals into structured functional behavior, decision logic, data rules, and handoffs that downstream teams can execute.

APREITUDE DTC ViiV landing-page and signup requirements including form validation and business rules.	VIIIV CONNECT Branching quiz logic mapping user answers to support / access pathways.
RACETRAC WALLET Data and architecture requirements spanning source inputs, business logic, ELT, validation, and outputs.	QA DELIVERY MODEL BRDs and traceability feeding test planning, Zephyr execution, regression, and release sign-off.

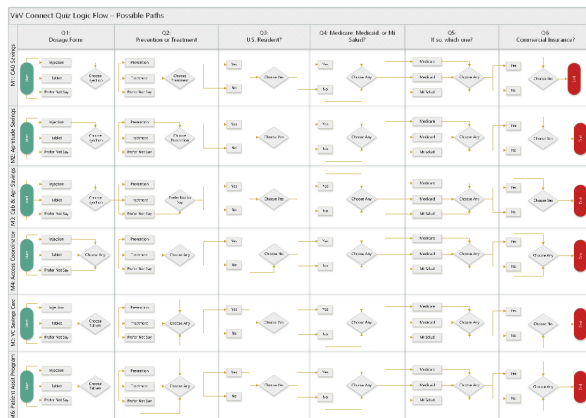


Figure 5. Example of branching access-logic mapping from another ViiV engagement

5. Business Value & Professional Impact

- Translated creative and UX intent into implementation-ready behavior.
- Provided a common BDD language for client, product, development, and QA stakeholders.
- Made navigation, state, exceptions, responsive behavior, and recap logic explicit before build and validation.
- Operated in regulated healthcare digital work where content, user experience, and compliance-sensitive behavior must remain aligned.

Independent validation of the role

Luckie leadership later described my work as leading the BA practice for ViiV Healthcare’s HIV division and credited my approach with reducing risk and improving customer experiences. My direct manager cited analytical rigor, requirements gathering, process analysis, solution design, and cross-functional leadership across product, development, QA, and client stakeholders.

Documented outcome context

My résumé records the Dovato Digital Touchpoint / Doctor Discussion Guide as an award-winning healthcare initiative recognized with two Addy Awards. This case study focuses specifically on the business-analysis contribution and does not attribute the creative award outcome solely to BA work.

Skills Demonstrated

Business Analysis	BDD	Requirements Gathering	Acceptance Criteria	User Journey Analysis
State / Interaction Logic	Stakeholder Management	Agile / Jira	Regulated Digital	Technical Documentation

CASE STUDY TAKEAWAY

The Dovato DDG shows the BA work between an idea and a reliable build: define the journey, expose the decisions, preserve state, clarify exceptions, and write behavior so every downstream discipline works from the same source of truth.