Fachhochschule Nordwestschweiz
Hochschule für Informatik

Project-Number: 25HS_IIT26

Data-Driven Persona Development

Automated Creation, Validation and Evolution of Personas
IP5-Project
Autumn-Semester 2025

Submitted to:
Prof. Dr. Norbert Seyff
Dr. Nitish Patkar

Submitted by:
Miro De Nardo (Degree Program Computer Science – Profiling iCompetence)
Andrin Sibold (Degree Program Computer Science – Profiling iCompetence)

University of Applied Sciences and Arts Northwestern Switzerland, School of Computer Science
Windisch, February 2026

Abstract

Personas are widely used in user-centered software development to support shared understanding and design decision-making. However, both traditional manually created personas and many data-driven persona approaches still suffer from two fundamental shortcomings: they are often treated as static “flat file” artifacts and they rarely provide operational mechanisms for continuous validation and evolution over time. To address this gap, this IP5 project investigates how personas can be generated from data and maintained as living, evidence-based artifacts by combining behavioral analytics with explicit user feedback.

We propose and implement an end-to-end prototype, the Proto-Persona Evolution Dashboard, based on the OPeRA e-commerce dataset. The backend pipeline extracts interpretable Behavioral Key Metrics from clickstream sessions, derives behavior-based clusters, characterizes them via z-score deviations and descriptive labels, and enriches each cluster with LLM-generated user profiles and synthesized Proto-Personas grounded in interview transcripts. The frontend dashboard operationalizes progressive disclosure and enables temporal comparison across monthly snapshots to make changes in cluster structure and persona characteristics observable.

Our evaluation follows an artifact-centric approach. Results show that the system successfully bridges quantitative clustering and human-readable persona artifacts, supports interactive exploration, and enables temporal inspection under consistent visual encoding. At the same time, the validation highlights important limitations: LLM extraction is reliable when rich interview data is available but becomes unstable and prone to over-inference for sparse inputs. Overall, the project demonstrates a practical foundation for transparent and time-aware data-driven Proto-Personas.

Keywords:

Data-Driven Persona Development (DDPD), Personas, Proto-Persona, Behavioral Clustering, Clickstream Analytics, K-Means, Persona Evolution, Large Language Model (LLM), Temporal Analysis, Visual Analytics Dashboard, User Analytics, Explainability, Transparency, LLM-based Extraction, Evidence and Uncertainty Indicators, Requirements Engineering

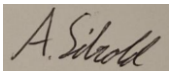
Declaration of Honesty

We hereby declare that we have completed this project independently and have cited all sources, references, and tools used. We also declare that any paragraphs or images not originating from us have been correctly cited and that the sources used are listed in this document.

This project has not yet been officially published and has only been made available to the supervisors of the IP5 project and thus to the FHNW.

Windisch
February 6, 2026

Name: Andrin Sibold

Signature: 

Name: Miro De Nardo

Signature: 

Table of Contents

1. Introduction

1.2 Motivation and Baseline

Personas are a widely used technique in software development to represent typical user groups and support informed decisions in requirements engineering, UX design, and product strategy. A persona aggregates the common behaviors, goals, and motivations of a user segment into a concise, human-readable profile [1]. This helps development teams keep the user in focus during design and implementation and promotes empathy and common understanding within the team [2]. In addition, personas counteract the risk of "self-referential design", in which developers unconsciously project their own mental models onto the product instead of considering those of the actual target group [3].

In practice, however, personas are often created at the beginning of a project using qualitative and ethnographic methods such as interviews, field studies, or workshops. Although these methods can provide rich insights, they are often time-consuming and costly [2]. These high costs often limit how thoroughly personas are developed or result in sample sizes that are too small to be considered statistically significant [4]. Furthermore, manual persona development (MPD) is often criticized for being based on subjective interpretations and lacking empirical rigor [2] [5].

A critical problem with the traditional approach is that personas often become static artifacts. Once created, they are rarely validated against real usage data and hardly ever updated [6]. This contradicts the reality of digital products, in which user groups, their behavior, and their needs evolve dynamically over the lifecycle of a system. The lack of systematic validation after initial creation increases the risk that teams will base their decisions on outdated assumptions or stereotypes rather than actual user evidence [2].

Research in the field of data-driven persona development (DDPD) has shown that large amounts of user data can be used to create personas more efficiently. By analyzing interaction logs (clickstreams) and applying clustering methods such as k-means, behavior-based user groups can be identified. This approach offers advantages such as increased objectivity, scalability, and the ability to track changes in user behavior over time [2] [1].

Nevertheless, current approaches focus primarily on the creation of personas and often assume that sufficient operational data is already available. The continuous validation and evolution of personas as a regular activity during system operation has not been adequately addressed in the literature to date. This creates a gap between how personas should support modern, iterative software development and how they are actually maintained (or neglected) in practice [6].

1.3 Problem Statement

Despite the widespread acceptance of personas as a method in user-centered design and the increasing availability of user data, there are significant shortcomings in the way personas are currently created, validated, and maintained. Scientific literature and industrial practice reveal

specific gaps that prevent personas from reaching their full potential as dynamic decision-making aids.

1.3.1 Limitation of Existing Approaches

Traditional, purely qualitative personas have long been criticized. They are often considered subjective, based on small samples, and lack scientific verifiability [7] [2] [1]. An even more serious problem is their static nature: once created, they quickly become outdated as markets and user behavior change, but personas often remain as rigid documents ("flat files") [1] [8].

The Data-Driven Persona Development (DDPD) approach attempts to solve these problems by using large data sets (e.g., clickstreams, social media data) and algorithms (e.g., k-means clustering, NMF) [2] [1]. Although DDPD methods offer advantages such as scalability and cost efficiency, in practice these personas are often treated as static artifacts [1]. There is a lack of established mechanisms to ensure that data-generated profiles remain relevant over time and reflect actual behavioral changes [8] [1]. The mere use of quantitative data does not guarantee "living" models if the processes for updating and validation are missing.

1.3.2 Research Gaps (In General)

An analysis of existing research literature reveals a clear focus on the initial creation of personas. The vast majority of DDPD studies deal with how to segment user data and convert it into persona profiles. In contrast, the validation and evolution phases are neglected [2] [6]:

Lack of operationalization of validation:
While the need for validation is recognized in theory, in practice this often happens informally or purely qualitatively (e.g., through expert interviews) rather than through continuous, data-driven metrics [2] [6]. There is a lack of standardized procedures for continuously checking the accuracy of a persona against new data [2] [6].

Lack of temporal dimension:
Research usually considers personas to be a snapshot in time. Approaches that explicitly treat changes in user behavior as an object of analysis and model personas as evolving entities (persona evolution) are rarely found in the literature [6]. Although "updateability" is cited as a theoretical advantage of DDPD [2], there is a lack of concrete implementations that show how a persona's attributes change over time and how this change is made visible to decision-makers [6]. In addition, recent work on AI-supported persona development highlights that persona research also lacks robust longitudinal evaluation: Amin et. al. explicitly note a "lack of evaluation, especially longitudinal evaluation", because existing studies do not assess whether AI-generated personas actually improve design outcomes over time, a critical gap given that personas are means to support design [9].

A deeper analysis of the research gaps we have found is written in chapter 2.5 "Research Gaps".

1.3.3 Practical Gaps

In practice, development teams have continuous data streams (analytics, feedback, logs) at their disposal, but paradoxically rarely use them to question or update their personas [10] [11]. The gap here is not a lack of data, but a lack of a practical connection between the raw data and the persona artifacts over time. Existing solutions are insufficient because they often use "black box" algorithms that are difficult for stakeholders to interpret [12] [13], or they provide pure dashboards that lack the empathetic component of a persona [2] [8]. There is a need for a system that bridges the gap between abstract analytics data and tangible persona profiles, proactively signaling changes in behavior rather than just mapping static segments [6] [12].

1.3.4 Problem Formulation

In summary, the core problem of this work can be defined as follows:

There is no established mechanism that makes the validity and evolution of personas based on continuous usage data transparent and operationalizable. Although data would be available for dynamic adaptation, personas in software development usually remain static artifacts, causing them to lose the ability to reflect real changes in user behavior and support informed, time-critical design decisions. This also increases the risk that decisions will be based on outdated assumptions.

To address this deficit, an approach is needed that bridges the gap between pure data analysis and the empathetic representation of users, while taking into account the temporal dynamics of user behavior. While the mere availability of data is not sufficient, mechanisms must be established that make algorithmic results transparent and reduce the cognitive load of interpreting complex user data through interactive visualizations [2] [13] [14]. This leads to the following research questions:

Research Question A – Translations of Clusters into Personas
How successful is the conceptual and technical translation of behavior-based clusters into understandable personas (including structure, attributes, narratives), so that the results do not remain abstract segments?

Research Question B – Robustness and Limitations of LLM Extraction
Under what conditions does LLM-supported user profile and persona generation deliver consistent, data-plausible results and where are the limitations (over-inference, normalization, variance)?

Research Question C – Usability and Cognitive Comprehensibility of the Dashboard
To what extent does the dashboard help stakeholders move from a cluster overview to interpretable personas (overview → drill-down → evidence) without creating a high cognitive load?

1.4 Project Vision

The vision of this project is to establish personas as data-driven, continuously maintainable, and transparent tools that remain relevant far beyond the initial kick-off workshop. Instead of treating personas as static, one-time artifacts, the project aims to demonstrate that they can be transformed into "living models". They must be able to be created, reviewed, and updated based on data collected during app usage [2]. This approach directly addresses the research gap identified in the problem statement: While existing work in the field of "data-driven persona development" (DDPD) focuses primarily on initial creation, the phases of continuous validation and evolution have received little support in practice to date [6].

The project is positioned in the field of data-driven requirements engineering and aims to develop a practical tool that helps teams understand, based on data, "who their users are", "how they behave", and "how these behavior-based user groups change over time". Specifically, the project aims to implement a solution that operationalizes the core claim that "personas can be generated from data". Two complementary signal types are integrated to create a holistic picture (mixed-methods approach) [2] [6]:

1. **Behavioral and monitoring data**: This objectively reflects what users do (e.g., usage intensity, session characteristics, and clickstreams) [2] [1].
2. **Explicit feedback**: This reflects what users say (e.g., stated preferences, needs, and complaints) to enrich purely quantitative data with qualitative contextual information [6].

The desired result is a mechanism that identifies distinct user groups through clustering methods and generates a "Proto-Persona" [15] profile for each group [2] [1]. A key feature of this mechanism is transparency: for each attribute of the user profiles created, which are later used for the generated Proto-Personas, the extent to which it is supported by the underlying data (confidence) is indicated in order to strengthen stakeholder confidence in the generated profiles [2] [1].

The implemented solution manifests itself in the front-end representation of this mechanism: an interactive dashboard. This dashboard not only visualizes behavior-based clusters and allows stakeholders to inspect cluster-specific Proto-Personas, but above all makes changes visible across multiple time windows [2]. This temporal perspective is essential: the project aims to detect and display "persona drift" and evolution, whether through shifting cluster distributions, the emergence of new groups, or changes in dominant behaviors [2] [6]. This completes the transition from static segmentation to continuous monitoring of the user base [6].

1.5 Goals and Expected Results

Based on the objective formulated in the project vision of transforming personas from static artifacts into dynamic "living models," this section defines the specific goals and expected results of this work. The overarching goal is to develop a technical artifact that bridges the gap between pure data analysis and empathetic user understanding by linking quantitative behavioral data with qualitative user insights and making their evolution over time visible.

1.5.1 Search and Development of a Hybrid Data Set

One goal of the project is to establish a robust data approach that integrates different types of data in order to overcome the limitations of purely quantitative or purely qualitative methods. In the literature, the mixed-methods approach is often described as the most robust way to create personas, as it combines the statistical significance of quantitative data with the depth of qualitative insights [2] [16].

Therefore, this project aims to develop a pipeline that combines the following data streams:

1. **Behavioral Data:**
   This includes clickstreams, usage intensity, and navigation paths to objectively capture what users do in the system. Such data forms the backbone of segmentation because it is scalable and less prone to subjective bias than self-reported data. In addition, behavioral data is an ideal way to integrate the temporal dimension into personas [2].

2. **Timestamps:**
   To overcome the static nature of traditional personas, it is essential to integrate the temporal dimension. Timestamps make it possible to identify changes in user behavior (persona evolution) and ensure that personas reflect current reality rather than representing outdated snapshots [2].

3. **Explicit Feedback:**
   To understand the reasoning behind the behavior, explicit feedback (e.g., from interviews or comments) is incorporated. This addresses the issue that purely quantitative personas are often perceived as “soulless” and lack the context for design decisions [2] [17].

The expected result is an existing dataset and a pre-processing workflow that translates these data sources into features suitable for clustering algorithms while providing rich information for later narrative enrichment.

1.5.2 Implementation of a Proto-Persona Generation Mechanism

A core objective of the work is to develop a mechanism that generates cluster-based personas using modern technologies for enrichment and validation. While clustering methods such as K-Means are well established in the literature for identifying user segments, there is often a lack of automated, comprehensible translation of these clusters into tangible profiles [2].

The mechanism to be developed should have the following characteristics:

- **LLM-supported Attribute Inference:**
  The use of large language models (LLMs) offers the potential to draw interpretative conclusions from unstructured explicit feedback (e.g., transcribed interviews or free texts) and generate Proto-Persona narratives [18] [19]. The goal is to use LLMs to derive “vivid” descriptions from the explicit data, while minimizing hallucinations by strictly

adhering to the underlying data [9].

- **Transparency and Confidence Indication:**
  A critical problem with data-driven personas is the lack of trust among stakeholders in "black box" algorithms [20]. Therefore, an explicit goal of this project is to provide a transparency metric for each generated persona attribute. The system should indicate how many data points a statement is based on (data coverage). In addition, there should also be a metric that shows how reliable the inference is (confidence). This addresses the research community's demand for greater transparency in algorithmic persona creation in order to increase acceptance among decision-makers [13].

1.5.3 Development of an Interactive Dashboard Prototype

The third main result is the transfer of the generated data to a user-centered interface. Static PDF personas ("flat files") quickly become outdated and do not offer any possibility for exploratory analysis [20]. The project aims to develop an interactive dashboard prototype that supports the following functions and combines the goals mentioned before in a dashboard prototype:

- **Exploration and Comparison:**
  Stakeholders and users of our dashboard prototype should be able to navigate between a high-level overview of the clusters and detailed persona information in order to minimize cognitive load [17] [12].

- **Time-based Inspection (Evolution):**
  The dashboard must visualize how clusters and personas change over different time periods. This is a direct response to the research gap in the lack of support for persona evolution [2]. Changes in cluster size or the emergence of new behavior patterns must be made visually tangible.

1.5.4 Success Criteria and Evaluation

In order to evaluate the success of the solution developed within the scope of the IP5 project, the following criteria have been defined, which are based on the requirements for scientific accuracy and practical applicability:

1. **Functionality with Real Data Set:**
   The solution must not remain purely theoretical. Success will be measured by whether the prototype is capable of processing a given data set (consisting of clickstream data and feedback) and generating coherent personas from it. This demonstrates the feasibility of the "generating personas from data" approach [2].

2. **Traceability:**
   A key quality feature is traceability. Persona attributes and claims must be traceable to

specific behavior patterns or explicit feedback evidence. This is essential to counteract the risk of stereotyping and algorithmic bias and to strengthen confidence in the results [2] [8].

3. **Representation of Temporal Dynamics:**
   The solution must successfully demonstrate that it can map changes over time windows. Success is defined here as the system's ability to recognize differences in cluster composition or persona characteristics between two points in time (t1 and t2) and visualize them in a way that is understandable to the user. This sets it apart from static approaches and fulfills the vision of “living models” [2].

4. **Adequate Project Scope and Expandability:**
   The solution must be designed in such a way that it combines the core components (dashboard, clustering, persona generation) in a functional prototype that can be implemented within the scope of IP5. At the same time, the architecture should be as modular as possible so that it provides a solid foundation for future enhancements [2].

In summary, this project aims to close the theoretical gaps in data-driven persona development (DDPD) research by providing a concrete, implemented prototype solution that works with a pre-processed data set and focuses on transparency, temporality, and interactivity.

2. State of the Art

This chapter places the concept of personas in the context of modern software development. It serves as a theoretical and methodological foundation for the conceptual solution presented in the next main chapter and our ultimately implemented solution. The purpose of this chapter is to trace the evolution from classic, qualitative personas to quantitative, data-driven approaches (Data-Driven Persona Development, DDPD) and to identify the principles necessary for designing interpretable dashboards. This chapter therefore summarizes existing approaches and research findings to create the basis for a system that not only aggregates data but also makes it understandable.

This project specifically examines approaches that deal with the temporal dimension (persona evolution over time) and interactive visualization (persona dashboards) [2]. General marketing personas, which are primarily created for brand positioning or advertising target groups without direct reference to product usage data, are not the focus of this project. Similarly, purely fictional “ad hoc” personas are only mentioned for the sake of distinction but are not dealt with in depth (whereas Proto-Personas are). The aim is to shed light on the gap between static data snapshots and dynamic, explorable user models.

2.1 What is a Persona?

In order to understand our final dashboard prototype, we first need to define what a persona essentially is and what function it fulfills in the software development process.

2.1.1 Definition and Purpose

In literature, a persona is commonly defined as a semi-fictional characterization or archetype that represents a specific user segment [3]. Alan Cooper, who introduced the concept to software development in the 1990s, describes personas not as real people, but as “hypothetical archetypes of actual users” defined with “significant rigor and precision” [4] [7].

It is crucial to understand that a persona is not an end product, but rather a design and decision-making tool. It serves as a “placeholder” for real users, translating abstract data into a tangible, human form [2]. Personas act as a channel to bundle a wide range of qualitative and quantitative data and draw the team's attention to design aspects that other methods might neglect [3] [8].

2.1.2 Why Personas Work

The success of personas in practice is based on psychological and communicative mechanisms:

- **Generating Empathy:**
  Personas give anonymous data a human face. This makes them “cognitively convincing” and enables designers and developers to develop empathy for the needs, desires, and

limitations of users [3] [15].

- **Shared Understanding:**
  In interdisciplinary teams, personas prevent the problem of the "elastic user," a phenomenon in which each stakeholder has a different idea of the user and adapts it according to convenience [3]. Personas create a consistent mental model of the target group for all involved [21] [3].

2.1.3 Anatomy of a Persona

Although the format of a persona can vary, a consensus is emerging in the literature about the attributes that make a persona capable of action. Based on the analysis of persona templates [17] [3], the essential components can be summarized in the following mini-frame, which serves as a reference for this work:

Table 1 – Essential Components of a Persona

Attribute Category:	Description and Content:	Purpose:
Identity	Fictitious name, portrait picture, age, place of residence.	Makes the persona relatable and human [2].
Demographics & Background	Education, job title, marital status, prior technical knowledge.	Anchors the persona in a realistic context [22].
Goals and Motivation	What does the user want to achieve (end goals)? Why do they want it (motivation)?	Leads design decisions and feature prioritization [3].
Behavioral Patterns	Frequency of use, typical workflows, interaction style with technology.	Describes how the user uses the software [2].
Pain Points	Obstacles, fears, current problems with existing solutions.	Identifies opportunities for improvement and innovation [7].
Narrative and Scenario	A short story describing the context of use.	Connects isolated data points into a cohesive experience [3].

Example of a Persona from the Wikipedia-Page about Personas (User Experience):

Figure 1 - Example-Persona from Wikipedia [3]

This persona from the example from Wikipedia represents nearly all the described elements of a persona: It shows an identity (name, age, place of residence), background information (occupation) and the goals, behavioral patterns and pain points are all written in a biography.

2.1.4 Limitations of Traditional Approaches

Despite their usefulness, traditionally created personas are subject to significant criticism, which justifies the need for data-driven approaches (Data-Driven Persona Development, DDPD):

- **Subjectivity and Bias:**
  Manual personas are often based on the assumptions of their creators or on small, unrepresentative qualitative samples [15] [2]. This leads to skepticism among stakeholders regarding their scientific validity [3].

- **Lack of Validation:**
  It is often unclear whether a persona actually represents a significant user group or whether it excludes marginal groups [2] [3].

- **Static Flat Files:**
  As already mentioned, personas are often treated as static documents (PDFs, posters).

They age poorly and cannot reflect changes in user behavior over time [2] [1]. Since user behavior develops dynamically, static personas quickly lose their relevance for agile development [21].

2.2 What is a Proto-Persona?

Within the persona methodology, a distinction is made between different types, which differ primarily in terms of the type of research data available. While statistical personas are based on large quantitative samples and qualitative personas are based on in-depth user research interviews, Proto-Personas are a “lightweight” form that is not based on new research [15].

A Proto-Persona is defined in literature as a representation that is not based on new, primary research, but rather reflects existing assumptions (“best guesses”) and the implicit knowledge of stakeholders about users [15] [7]. They are typically used to quickly bring a team to a common mental model without the high costs and enormous time expenditure involved in complete data collection [15]. The main difference to classic personas lies in the degree of validation: Proto-Personas are explicitly not considered factual truths, but rather hypotheses [15]. They serve as a starting point for visualizing assumptions that need to be validated or refuted in the further course of the project and can act as a transition to more in-depth research [15].

Although the system developed in this paper is based on quantitative usage data (clickstreams) and thus goes beyond pure “gut feeling” assumptions, we deliberately use the term “Proto-Persona”. This step is taken for reasons of scientific integrity and transparency in dealing with generative AI:

In our approach, clusters are enriched with elements such as clickstreams using large language models (LLMs). Research shows that LLM-generated personas without a strict empirical basis risk being purely fictional and lacking the realistic component of a genuine persona [19]. Since LLMs are based on probability calculations and generate interpretations that contain a certain degree of uncertainty despite the underlying data, the claim of a “finished” persona would be misleading.

By classifying the generated user profiles and the personas synthesized from them as Proto-Personas, we emphasize their character as data-informed hypotheses. We do not generate any claimed truths, but rather cluster-based Proto-Personas, which use evidence and confidence metrics to make transparent how strongly a characteristic is supported by data. They are not static facts, but dynamic models whose validity is continuously evaluated by the system. This is consistent with the view that personas in modern, data-rich environments should be viewed less as rigid documents and more as “living” artifacts [15].

2.3 UI Principles for Data-Driven Persona Dashboards

Generating personas from data is only the first step. Equally critical is how this complex, multidimensional information is presented to stakeholders. Since data-driven approaches are often based on opaque algorithms (e.g., clustering, LLMs), the user interface (UI) must not only

display information, but also actively build trust and break down cognitive barriers. Based on the findings of persona research, the following core areas for UI design can be derived.

2.3.1 Interpretability and Transparency

One major criticism of data-driven personas is the "black box" nature of the underlying algorithms. Stakeholders often do not trust personas if it is not clear how they were created [23] [1]. To counteract this "mystique of numbers," the front end of a dashboard that visualizes data-driven personas must offer mechanisms for transparency [13]:

- **Visualization of Metrics:**
  The UI should display indicators such as confidence (how confident is the model in this statement?) and data coverage (how many users is this attribute based on?) directly on the persona profile [6] [1].

- **Explainability:**
  Instead of just presenting the final result, explanations must be provided on how attributes were derived. Studies show that transparency mechanisms can significantly increase the perceived completeness and clarity of personas [1].

2.3.2 Information Architecture – Overview and Drill-Down

Since data-driven personas are based on large data sets, there is a risk of information overload [12]. To minimize cognitive load, the dashboard should follow the classic visual analytics mantra: "Overview first, zoom and filter, then details on demand" [24] [14].

1. **Overview:**
   The top level should show a distribution or overview to give a quick overview of the user base.

2. **Persona-Detail:**
   Upon request (click/interaction), more specific details should appear (for example, in a pop-up window or a modal).

3. **Evidence-Level:**
   At the lowest level, users should have access to statistical evidence or representative raw data (e.g., quotes, log extracts).

2.3.3 Representation of Temporal Dynamics

A key innovation of this project is the mapping of persona evolution. The UI must therefore make changes over time comparable without confusing the user. Static personas ("flat files") cannot achieve this [2] [6]. Effective strategies include:

- **Consistent Scales:**
  To make changes between time t1 and t2 (e.g., “Persona A has become more technophobic”) noticeable, visualizations must use consistent axes and metrics.

- **Comparison Views:**
  The dashboard should enable personas to be compared in order to visually highlight shifts in cluster size or the emergence of new behavior patterns [6].

2.3.4 Summary – UI-Principles Catalog

Based on the literature analysis, we derive the following design principles for the Proto-Persona Evolution Dashboard, which are referenced in Chapter 4 (Implementation):

1. **Principle of Progressive Disclosure:**
   The dashboard always starts with an aggregated overview and only reveals details upon explicit interaction so as not to overwhelm stakeholders.

2. **Principle of Evidence-Based Transparency:**
   The Proto-Persona itself and its attributes must be visually marked with a confidence or evidence metric to make the statistical reliability (“Based on X Users”) transparent.

3. **Principle of Temporal Comparability:**
   Changes over time must be explicitly represented through consistent visualizations and comparison options in order to make the evolution of the user base recognizable.

2.4 Existing Approaches

The whole research of existing approaches is based on all the papers, articles and websites, which we have put into our Excel-Sheet to analyze them all together and to find a gap in the current research literature / topic of data-driven persona development.

2.4.1 Digression – Literature Research Process

Procedure for identifying sources:
An iterative, multi-stage literature review was conducted to assess the state of research in the field of data-driven persona development (DDPD). The aim was to identify a sufficiently broad yet thematically focused set of scientific publications (papers, SLRs, methodological contributions, framework/tool descriptions, and selected practical articles) in order to

(a) systematically compare existing approaches and

(b) identify recurring research gaps in a comprehensible manner.

The research process was continuously documented in an Excel-Sheet, which served as the central working artifact for the research part.

Starting point and iterative refinement:
The research began with seed publications from the project environment (DDPD/Persona Engine/Validation/Evolution). From there, an iterative approach was used, consisting of two loops:

1. Broad search phase (exploration): Development of an overview of terms, data types, typical methods, and key publication locations.

2. Focused search phase (convergence): Targeted search for articles that explicitly address validation, evolution, or temporal aspects (e.g., longitudinality, drift, versioning).

Search locations:
The search was not limited to a single database, but was conducted across multiple platforms. Typical search locations were:

- Google Scholar (broad coverage, citation graph, “related articles”)

- Publisher and library portals (e.g., ACM Digital Library, SpringerLink, ScienceDirect/Elsevier, Taylor & Francis)

- Preprint/open access platforms (e.g., arXiv)

Search terms and search logic:
The search was query-based and deliberately combined broad entry terms with precise Boolean combination queries to cover the thematic gap (validation/evolution over time). Among others, the following were used particularly frequently:

- Broad entries for field coverage:

    - “persona development”

    - “data-driven personas”

    - “persona creation”

- Focused queries on the lifecycle and time perspective (synonyms deliberately duplicated, e.g., behavior/behaviour):

    - (“user behavior” OR “user behaviour”) AND (“validation” OR “evolution” OR “creation”) AND (“persona”)

    - (“user behavior” OR “user behaviour”) AND (“data sources” OR “log data” OR “usage data” OR “interaction data”) AND (software OR applications OR systems OR platforms)

Selection and quality criteria:
A source was included in the Excel log if at least one of the following criteria was met:

- The contribution provides a data-driven method for persona creation or derivation (e.g., clustering, NLP, NMF, ML).

- The contribution explicitly addresses the validation, evolution, longitudinality, or maintenance of personas.

- The contribution contains specific implementation or evaluation details (data types, pipeline, metrics, study design) or is a systematic overview (e.g., survey/SLR).

Excel analysis:
The Excel log was used as an analysis tool not only to collect the identified literature, but also to make it comparable and systematically derive patterns, pain points, and research gaps. The log was thus not merely a bibliography, but a coded review artifact that supports the subsequent argumentation in the report.

Each source was recorded line by line and described along several dimensions. The most important categories were:

- Bibliographic metadata: Title, year, authors, type (paper/SLR/method/article), link/DOI.

- Data perspective: Data source category and what kind of data used (e.g., analytics/clickstreams, feedback, surveys, social data).

- Method perspective: Kinds of methods used… or methods/approaches for creation (e.g., k-means, NMF, NLP, supervised/unsupervised).

- Lifecycle perspective: Lifecycle coverage (creation / validation / evolution / full cycle).

- Practice and tooling perspective: Integration into practice, automation level, usability/complexity, evidence of adoption.

- Evaluation & validation: Validated?, How validated?, Evaluation context (which domain, which study, which metrics).

- Gap identification (gaps identified by authors and gaps we identify)

- Quick synthesis/scoring: Key findings and evaluation fields (e.g., information content/usability) to prioritize sources.

How this Excel analysis helped us:
The Excel evaluation had three specific benefits:

1. The uniform columns allowed approaches across many papers to be compared using the same criteria (data → method → output → evaluation → lifecycle).

2. The column “Gap Identified by Authors” made it possible to consolidate recurring research gaps instead of merely “feeling” gaps.

3. Design and implementation principles for the dashboard could be derived directly from the coded deficits (e.g., interpretability, lack of time dimension, lack of practical integration).

The development of personas has evolved from purely qualitative origins to complex, algorithmic processes. The research literature can be roughly divided into three methodological trends: qualitative ad hoc approaches, purely quantitative data-driven persona development (DDPD) methods, and hybrid approaches. In the following, these approaches are analyzed and their limitations in the context of modern software development are highlighted.

2.4.2 Proto-Personas and Lean UX

In agile environments, especially in the context of Lean UX, the Proto-Persona approach is often chosen. As already mentioned in chapter 2.2 “What is a Proto-Persona?”, these are typically created in workshops in which stakeholders and developers externalize their existing assumptions (“best guesses”) about users and transfer them into persona templates [15].

- **Methodology:**
  The process is quick and inexpensive (often a 2-4 hour workshop) and does not require any new data collection [15]. Its primary purpose is to align the team around a common mental model [15].

- **Boundaries:**
  The main criticism is the lack of an empirical basis. Since these personas are based on assumptions, they run the risk of being an “echo chamber” for internal biases rather than reflecting real user needs [15]. They are considered hypotheses that need to be validated, but in practice this often fails to happen. Studies show that such “low-fidelity” methods are popular in startups, but often do not provide the necessary depth for complex system decisions [15].

2.4.3 Data-Driven Persona Development (DDPD)

In response to the subjectivity of manual methods, the field of data-driven persona development (DDPD) emerged. Here, quantitative data is used to identify user segments algorithmically.

- **Data Sources:**
  The research primarily uses large data sets such as clickstreams, web analytics logs, telemetry data, and social media interactions [2]. One example is the work of Xiang Zhang, who used clickstream and telemetry data to identify archetypal usage patterns [25].

- **Methods:**
  The three dominant algorithms in the literature are k-means clustering, hierarchical clustering, and non-negative matrix factorization (NMF) [2].

- **Boundaries:**
  A major problem with purely quantitative approaches is the “interpretation gap”. The outputs of these algorithms are often abstract cluster centroids or segment labels (e.g.,

"Cluster A") that lack the empathetic, narrative quality of a classic persona [2]. There is a risk that the results will be perceived as "soulless data points" that are statistically significant but difficult for designers to grasp [2]. In addition, these methods often require expert knowledge in data science, which makes them difficult to apply in pure design teams, as explicitly mentioned in Joni Salminen's paper on page 12: "Another mentioned weakness of clustering is the 'need for specialists to use expert judgment during clustering [to define hyperparameters]'" (Minichiello et al., 2018, cited in [2], p. 12).

2.4.4 Hybrid Approaches

In order to bridge the emotional void of pure data and the subjectivity of pure fiction, research is increasingly promoting hybrid approaches (mixed methods).

- **Procedure:**
  Typically, quantitative data (e.g., from log files or large-scale surveys) is used to create a statistically valid segmentation (the "skeleton"). These segments are then enriched with qualitative data (e.g., quotes from interviews or ethnographic observations) ("personification") [15] [2].

- **Example:**
  Miaskiewicz et al., for example, used latent semantic analysis (LSA) on large text corpora to identify topics and then enriched them manually [2]. The Nielsen Norman Group's "statistical personas" approach also falls into this category, in which survey data is clustered and then formulated qualitatively [15].

- **Boundaries:**
  Although methodologically robust, these approaches are extremely resource intensive. They require both qualitative and quantitative expertise and are often so costly that they are only carried out once at the start of a project [15].

2.5 Research Gaps

Based on the analysis of the research literature in our Excel-Sheet, we have identified the following structural deficits in current Data-Driven Persona Development (DDPD) research:

2.5.1 The "Flat File" Problem and Lack of Interactivity

One major criticism of the current practice of personas is their presentation format as static documents ("flat files"), such as PDFs or posters. Jansen et al. argue that the transition from static files to interactive interfaces is necessary to improve user understanding [2]. Static documents cannot adequately reflect the complexity and multidimensionality of large data sets and do not offer any possibility for exploration. Although it is theoretically required that decision-

makers should be able to interact with personas (e.g., by filtering attributes or comparing segments), fully functional, interactive persona systems are rare in the literature and are mostly still in the "proof-of-concept" stage [2]. There is a lack of tools that enable stakeholders to explore the database rather than just consume a finished result.

2.5.2 The Gap between Quantitative Breadth and Qualitative Depth

Research continues to struggle with the so-called "breadth-depth trade-off" (cited in [2], p. 11). Purely quantitative methods (e.g., clustering on clickstreams) offer scalability and statistical significance ("breadth"), but often result in flat profiles that lack narrative depth and an understanding of the reasons behind the behavior [2]. Conversely, qualitative data provides deep insights but is difficult to scale to large user bases. Although mixed-method approaches are proposed as a solution, the effective integration of both worlds remains a challenge [2]. There is often a lack of automated pipelines that seamlessly link behavior-based clusters with explicit feedback or generated narratives without the results being perceived as "soulless data points" [26].

2.5.3 Data and Representativeness Issues – Bias, Granularity and Privacy

The use of large amounts of data poses specific challenges in terms of representativeness:

- **Aggregation and Granularity:**
  The "aggregation problem" describes the difficulty that, as the number of relevant persona attributes (e.g., age, gender, behavior) increases, the number of personas required to accurately cover the user base increases exponentially [20].

- **Bias and Inclusivity:**
  Algorithms tend to focus on the average ("mean-centered personas"), which often renders fringe users and outliers invisible [2]. This leads to a lack of inclusivity and can reinforce stereotypes [13]. There is also a risk that bias in the training data (e.g., underrepresentation of certain demographics) will be uncritically transferred to the personas [7].

- **Privacy and Availability:**
  Strict data protection regulations and the limited availability of data through platform APIs ("walled gardens") make it difficult to access granular user data, which hinders the creation of robust models [11] [27].

2.5.4 Lack of Standardization and Reproducibility

Finally, the field of research suffers from low reproducibility. Most studies do not share the data sets or program code they use, which hinders scientific progress [2] [9]. There is a lack of

standardized end-to-end pipelines and persona templates that would enable different creation methods to be compared [17] [26].

2.5.5 The Neglected Dimension – Persona Evolution

In addition to the research gaps mentioned in the previous subsections, a critical analysis of the existing literature reveals another striking and significant research gap: the overwhelming majority of DDPD studies focus exclusively on the creation of personas. The question of how these Personas change over time (evolution) is almost completely ignored.

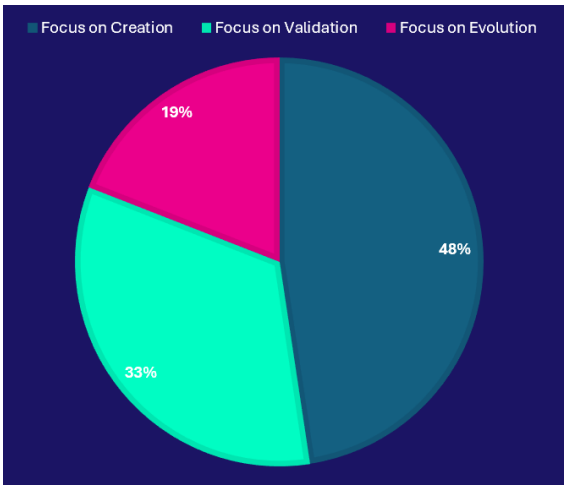


Figure 2 - Focus Distribution

Based on the 42 research papers/articles listed in our Excel-Sheet, only 19% addressed the topic of persona evolution in depth.

Although it is theoretically recognized that user behavior changes and personas need to be updated ("updatability" as a theoretical advantage of DDPD) [2], most of the solutions presented treat personas as static snapshots. There are hardly any established mechanisms or tools in the literature that operationalize the continuous validation and adaptation of personas to new data streams [2]. Jung et al. represent one of the few exceptions that attempt to capture changes in persona interests over time [2], but this remains a marginal phenomenon in the broad mass of research. In practice, this means that personas often end up as "flat files" (e.g., PDFs) that become obsolete shortly after their creation and no longer reflect the dynamic reality of the product life cycle [2].

3. Conceptional Solution

This chapter describes the conceptual solution for the project, i.e., the technology- and implementation-neutral design derived from the problem statement and the findings from the state of the art. The focus is on what a system must be capable of in order to make data-based personas usable as comprehensible, maintainable, and temporally comparable models. The specific technical implementation (data pipeline, dashboard, technologies) is described in the following chapter 4, “Implemented Solution”.

3.1 Requirements

The starting point for deriving the requirements are three interrelated problem areas that were gradually identified in the previous chapters:

(1) the structural weaknesses of traditional persona approaches,
(2) the methodological and practical limitations of existing data-driven methods, and
(3) the resulting research gaps that can be consistently observed in the literature corpus.

Firstly, the discussion on classic personas shows that although they are helpful as communication and empathy artifacts (chapter 2.1.2), they often remain insufficiently robust in data-rich, iterative development contexts. As described in chapter 2.1.4, manually created personas are often subjective, which promotes bias and low verifiability. In addition, there is usually no systematic validation of whether the persona actually represents the real user base. In practice, personas are also often maintained as static “flat files”. Without an established maintenance process (chapter 2.1.4 and 1.3.1), they become outdated when behavior or market conditions change, even though user groups are constantly changing (chapter 1.2 and 1.3.4).

Secondly, although Data-Driven Persona Development (DDPD) addresses key points of criticism by algorithmically segmenting usage data (e.g., clickstreams) (chapter 2.4.2), it often fails to solve the lifecycle problem. As explained in chapter 1.3.1 and 1.3.2, many approaches continue to focus on initial creation, while validation and, in particular, evolution are rarely operationalized. Even though “updateability” is cited as an advantage (chapter 1.3.2), concrete mechanisms for measuring changes over time and making them usable for stakeholders are often lacking. In addition, many results remain at the level of abstract cluster profiles, creating an interpretation gap between segmentation and understandable user representation (chapter 2.4.2), even though personas are intended to serve as design and communication tools (chapter 2.1.1).

Thirdly, these points are summarized in the research gaps we have systematically compiled (chapter 2.5). Of particular importance to this project are: the “flat file” problem and the lack of explorable persona systems (chapter 2.5.1), the trade-off between scalability and narrative explainability (chapter 2.5.2), data quality and representativeness problems (chapter 2.5.3), and the insufficiently addressed persona evolution as a temporal dimension (chapter 2.5.5 and 1.3.2).

Against this backdrop, the requirements are derived: A suitable conceptual solution must treat personas as interactively explorable artifacts, bridge the gap between quantitative clusters and comprehensible narrative profiles, and make changes visible over time. The conceptual approach thus addresses the core problem formulated in chapter 1.3: the lack of transparency and operationalizability of persona validity and evolution based on continuous usage data:

- **R1 – Personas as "living models" instead of static artifacts**
  Personas should not end up as static documents, but rather be maintained as versions/snapshots over time so that changes in user groups and behavior patterns become visible and can be discussed.

- **R2 – Transparency and traceability**
  Persona statements must be recognizably based on data. Instead of exaggerated "truth claims," evidence indicators (e.g., data coverage) should clarify what statements are based on and where uncertainty exists.

- **R3 – Interpretability for non-technical stakeholders**
  The result should not consist solely of segment IDs or purely statistical clusters. The solution must bridge the gap between (a) raw data/analyses and (b) comprehensible persona artifacts that can be used in UX/requirements/product decisions.

- **R4 – Temporal comparability**
  Change is treated as an object of analysis: drift, new clusters, and shifts in characteristics should be explicitly observable, not just implicitly in data noise.

- **R5 – Usability in the sense of "Overview → Drill-down"**
  Stakeholders need to quickly gain an overview (Which groups are there?), then delve into the details (What characterizes a group?), and finally see the evidence (What is this interpretation based on?).

These requirements result in the following functional requirements:

- **FR1 – Segmentation based on usage data**
  The system must be able to aggregate usage data for descriptive features and derive user groups/clusters from this data.

- **FR2 – Persona artifact per cluster**
  For each cluster, a persona-like profile should be created (in the sense of a Proto-Persona) that clearly summarizes behavior, needs, and characteristic features.

- **FR3 – Time window-based analysis**
  Segmentation must be carried out for several time periods in order to make changes visible.

- **FR4 – Comparison and change signals**
  The system must support comparisons between time windows and provide notifications when relevant changes occur (e.g., cluster becomes larger/smaller, new group emerges, dominant characteristics change).

- **FR5 – Evidence/uncertainty indicators**
  The system must output indicators for persona statements that create transparency (e.g., “based on X sessions/users”).

The non-functional requirements (NFR) are also listed below:

- **NFR1 – Reproducibility**
  The same input data and parameters lead to the same results (important for clustering).

- **NFR2 – Comprehensibility**
  Visualizations and terms must be interpretable for stakeholders without a background in data science or made understandable with the help of documentation.

- **NFR3 – Modularity**
  Data preparation, clustering, persona synthesis, and visualization should be logically separated to facilitate later extensions.

- **NFR4 – Scope compatibility**
  Focus on a working prototype, not on a fully automated production system (no real-time data integration).

Together, the above requirements define the conceptual guidelines for a solution that treats personas as versionable, evidence-based, and explorable artifacts that explicitly respond to changes in user behavior. The conceptual solution thus addresses the core problem formulated in chapter 1.3 and operationalizes the research gaps identified in chapter 2.5.

3.2 Mission

The vision of this project can be summarized as follows, as a summary of the vision described in detail in chapter 1.4: In data-rich software projects, personas should not end up as one-off, static artifacts, but should be understood as data-based, transparent, and continuously maintainable decision-making tools that remain relevant even during the operation of a system and can adapt to changes in user behavior.

The mission of the conceptual solution is to deliver a dashboard prototype that translates this vision into a practical form. The dashboard should visualize changes in user-based segments over time and help stakeholders interpret these changes (e.g., drift, growth/decline, or the emergence of new behavior patterns). At the same time, the system should demonstrate that Proto-Personas are generated from data, with explicit, qualitative data serving as the central

basis for narrative and empathetic enrichment, in addition to behavioral data. The focus is therefore less on creating "beautiful texts" and more on providing a comprehensible, explorable artifact that brings together quantifiable patterns and qualitative justifications, thus enabling evidence-based discussion within the team.

Since the work is being carried out as part of an IP5 project, the solution has been deliberately defined as a research and demonstration prototype. The following limitations therefore apply:

- **Proto-Personas instead of "ground truth" personas**
  The personas are considered data-informed hypotheses, not definitive truths.

- **Discrete time slots instead of streaming**
  Evolution is realized via comparable snapshots, not as real-time tracking.

- **Selection of the dataset**
  There is no expectation that the dashboard can process and display any dataset. The solution will be implemented for a selected dataset.

- **No long-term outcome evaluation in the field**
  A study lasting several months on the effect on design outcomes is not part of IP5, but is recognized as a gap in research and can be seen as motivation for future work.

3.3 Stakeholders and Working Fields

The transformation from personas to "living models" described in the Project Vision requires consideration of the actors who interact with the solution, as well as the domains in which such a data-driven solution can add the most value. Since the goal is to bridge the gap between quantitative data and qualitative user understanding, the tool addresses an interdisciplinary group of stakeholders.

3.3.1 Primary Stakeholders

The primary users of the solution are professionals involved in the development, design, and strategic orientation of software products. Based on the literature, the following main groups can be identified:

- **UX-Designer and User Researcher:**
  This group benefits greatly from the solution. Traditionally, they often create personas based on assumptions (Proto-Personas) or qualitative interviews [15]. The dashboard enables them to quickly validate these initial assumptions ("best guesses") against real usage data and minimize the risk of "self-referential design," which is designing for oneself rather than for the user [7]. UX experts can use the dashboard to monitor changes in user behavior and ensure that design decisions remain empathetic and user-centered during ongoing operations [15] [2].

- **Software Developer and Requirements Engineers:**
  Developers often do not have direct access to end users and sometimes find traditional personas abstract or implausible (“flat files”) [28] [19]. The transparent presentation of the database (“confidence” and “data coverage”) in the dashboard increases the acceptance of personas among technical teams [2] [6]. Requirements engineers can use the tool to derive functional requirements directly from observed behavior patterns and ensure that the developed system remains relevant even as usage patterns change [2].

- **Product Manager and Product Owner (PO):**
  Product owners often face the challenge of prioritizing requirements and deciding between business goals and user needs [6]. The dashboard serves as a decision-making aid, helping them prioritize feature requests or backlog items based on evidence-based data rather than pure intuition [6]. In agile environments in particular, it helps them justify to stakeholders why certain user segments (e.g., “curious browsers” or “search-driven navigators”) need to be focused on [21] [29].

- **Data Scientists and Analysts:**
  Although the tool aims to make data understandable for non-experts, data scientists act as important stakeholders who validate the underlying clustering models. The dashboard provides them with an interface to communicate the results of their algorithms (e.g., k-means clustering) in a form that is understandable to the business and bridges the gap between pure numbers and human profiles [15] [30].

3.3.2 Theoretical Areas of Application and Working Fields

While the primary focus of this project is on software development, analysis of the data-driven persona development (DDPD) literature shows that the areas of application extend far beyond this. The concept developed is theoretically transferable to all domains in which digital interaction data is generated and a deep understanding of user groups is required. However, our solution focuses primarily on the following working fields:

- **Agile Software Development and Requirements Engineering:**
  This is the core area of work. In agile development, where iterations are short and rapid feedback is essential, traditional, slowly created personas often do not fit into the workflow [21]. The dashboard supports this by providing “just-in-time” user information and enables continuous adaptation of requirements to reality (continuous requirements engineering) [2] [28].

- **E-Commerce and Digital Marketing:**
  E-commerce generates massive amounts of transaction and clickstream data. Here, the dashboard could be used to segment customers not only by revenue, but also by complex behavior patterns (e.g., “health enthusiasts” vs. “busy parents”) [29]. The dashboard would enable marketing strategies to be dynamically adjusted if, for example,

the purchasing behavior of a group changes seasonally or new customer segments emerge [2] [22].

In summary, the solution developed in this thesis is aimed at interdisciplinary teams in data-intensive environments. It bridges the gap between technical data analysts and strategic decision-makers and has theoretical applications in several of the sectors mentioned.

3.4 Suitable Application Scenarios

The potential stakeholders and working areas of application/working field of our solution were discussed in the previous chapter 3.3. This section therefore deals with the appropriate scenarios for which the solution would be particularly suitable. The solution is particularly suitable for products that:

- continuously generate usage data (analytics, logs, sessions),
- be used over a longer period of time (lifecycle relevance),
- make recurring product decisions (iterations, releases),
- have heterogeneous user groups whose behavior may change.

Example scenarios:

- Scenario A – Detect drift
  After a feature update, the intensity and duration of use of certain users changes. The dashboard shows shifts in cluster distributions and changed cluster characteristics.

- Scenario B – New segment
  A new usage profile emerges (e.g., new target group). The solution makes this visible as a new cluster and generates a Proto-Persona as a basis for discussion.

- Scenario C – Prioritization
  Two clusters are growing strongly, while one is providing disproportionately negative feedback. Stakeholders use the generated Proto-Persona and evidence view to understand the user group providing negative feedback and to be able to deliver a suitable feature.

3.5 Conceptional Architecture

The conceptual solution consists of four logically separate layers that together enable a continuous process from data to persona artifacts.

Data and input layer:
Inputs are data sources that describe user behavior and/or user voices:
- Monitoring/behavioral data: e.g., sessions, events, feature usage, interaction patterns.
- Explicit feedback (optional/supplementary): reviews, comments, surveys, or free-text feedback

to formulate Proto-Personas (and statements used therein) as human-like as possible.
- Time information: timestamps to enable time window formation and comparisons.

Processing and analysis layer:
This layer transforms raw data into analyzable structures:
- Preprocessing: Data cleansing, normalization, handling missing values.
- Feature engineering: Derivation of interpretable metrics (e.g., usage intensity, session duration, feature diversity).
- Clustering/segmentation: Grouping similar users/sessions into clusters.
- Cluster characterization: Description of what constitutes a cluster (e.g., centroid-based profiles, dominant characteristics).

Proto-Persona synthesis layer:
Proto-Personas are derived from cluster information. The key aspects here are:
- Translation of statistics into narrative: Cluster profiles are translated into understandable statements about behavior/needs/context.
- Evidence and coverage information: Persona statements are provided with indicators that show how broad the database is.
- Conscious restraint: Uncertainty is not hidden, but made explicit (Proto-Persona as a hypothesis).

Visualization and interaction layer:
A dashboard presents the information in a way that stakeholders can work with:
- Overview: Distribution, number of clusters, time slot selection, list of Proto-Personas
- Drill-down: Cluster profile, persona profile, comparison views.
- Evidence-on-demand: Key figures, coverage indicators as justification, explanations.

3.6 Conceptional Aspirations

This chapter describes the key conceptual aspirations of the solution. The focus is on transparency and traceability, interaction logic with low cognitive load, and consistent consideration of the temporal dimension.

3.6.1 Transparency Model – Evidence, Coverage and Cautious Statements

A key requirement of the conceptual solution is that Proto-Persona profiles do not appear as a black box, but remain traceable in their derivation. This requires a transparency model that systematically links persona statements to their data basis, thus making the degree of support visible. In concrete terms, this means that each persona profile contains coverage information: statements are not presented in isolation, but are related to the database, for example by the

number of users or sessions that a cluster actually comprises. This makes it immediately apparent to stakeholders whether a profile is based on a broad foundation or only represents a small section of the user population.

In addition, evidence indicators are needed, especially where persona statements are derived from explicit feedback or qualitative sources. In such cases, it should be clear how strongly a statement is supported, for example by frequency. The transparency model thus deliberately pursues a cautious, responsible communication strategy: Proto-Personas are understood as data-informed hypotheses whose significance is limited by coverage and evidence.

3.6.2 Conceptual Derivation of Interaction Logic

Based on the conceptual goals, the interaction logic of the solution follows the principle of progressive disclosure. This ensures that stakeholders quickly gain a rough understanding without being overwhelmed with detailed data from the outset, and that more depth is available step by step as needed. The application therefore begins with an overview level that provides the most important points of reference with a low cognitive load: which clusters exist, how sessions and users are distributed, which time window is active, and which Proto-Personas exist in this time window. This overview serves as a “map” on which stakeholders can identify relevant areas.

Building on this, a detail level enables the transition from pure structure to meaning. Here, cluster profiles and Proto-Personas are presented in such a way that behavior, usage patterns, and potential needs become tangible. Only when this semantic classification has taken place does the third stage follow: the evidence level, which creates trust and justifiability: figures, examples, and coverage information are made accessible in such a way that stakeholders can not only make decisions, but also argue them in a comprehensible manner. This creates a bridge between data-driven analyses and human-centered artifacts, with the solution providing not only “analytics”, but also an interpretable decision-making tool.

3.6.3 Time Dimension and Persona Evolution

The solution treats time not as additional information, but as a central design element. Persona evolution is conceptually implemented via versioned snapshots, so that personas and clusters can be viewed not as one-time snapshots, but as a sequence of comparable states. “Evolution” can take different forms: for example, it can manifest itself in changes in cluster distribution when a segment grows or shrinks. It can also occur in a shift in typical characteristic values, i.e., in a drift in cluster characteristics. In addition, structural changes are relevant, such as when new clusters emerge, existing ones disappear, or groups split or merge.

To ensure that such changes do not occur randomly or as a result of display artifacts, comparability over time must be conceptually guaranteed. Snapshots should be based on consistent foundations, in particular on consistent feature definitions across all time windows. Finally, it is crucial not only to show change implicitly (through separate individual views), but also to highlight it explicitly, for example, through delta or comparison views. This creates a

“change-first” view: The solution not only supports understanding of the current state, but also reveals how and why user groups and personas shift over time.

The conceptual solution defines a system idea that understands personas as versioned, data-based Proto-Personas and focuses on the temporal dimension. Core elements are segmentation based on usage data, persona synthesis per cluster, transparency regarding evidence, and an exploration interface. Based on this conceptual design, the next chapter describes the concrete implementation, including the data pipeline and dashboard.

4. Implemented Solution

This chapter presents the Proto-Persona Evolution Dashboard, a web application that operationalizes data-driven personas as dynamic, updateable models rather than static documents. The system processes e-commerce interaction sessions, groups them into behavior-based clusters, enriches these clusters with narrative descriptions generated from large language models, and visualizes both cluster characteristics and their development over time.

4.1 Limitations

As a research prototype, the implemented system prioritizes methodological demonstration over production readiness. The primary contribution lies in presenting a cohesive pipeline that integrates behavioral clustering with LLM-based persona enrichment and temporal visualization. While this approach enables exploration of data-driven persona development, it does not address production requirements such as scalability, robustness, or rigorous empirical validation, which would necessitate further engineering effort.

4.1.1 Scope and Prototype Constraints

As outlined in the project vision, the main goal of this work is to show that personas can evolve from static artifacts into continuously observable, data-driven representations. This objective determines the scope of the implemented system, which is deliberately designed as a research prototype to validate the methodological pipeline rather than as production-ready software. Consequently, several engineering concerns typically required for deployable systems, such as authentication, operational monitoring, and horizontally scalable backend services, are not addressed. While these omissions prevent direct deployment in real-world contexts, they do not diminish the prototype's intended contribution of demonstrating an end-to-end approach to data-driven persona creation that allows for the temporal inspection of evolving behavioral patterns.

A second constraint arises from the data architecture, which follows a file-based, batch-export approach instead of than a database-backed API layer. The backend pipeline exports versioned JSON files, specifically personas.json, sessions.json, and monthly_clusters.json, which the React frontend loads directly without an intermediary database or server-side query interface. This design choice prioritizes reproducibility and lightweight deployment, because the complete data layer consists of portable artifacts that can be stored, versioned, and regenerated under consistent conditions. However, it also limits interactivity. The dashboard cannot execute server-side filtering, trigger on-demand cluster recalculation, or support incremental updates as new data becomes available. Updating the displayed personas therefore requires rerunning the full backend pipeline, which makes the prototype unsuitable for use cases that depend on continuous ingestion or near-real-time responsiveness. Despite this limitation, the static artifact approach aligns with the research objective stated in Section 1.5.3 of enabling stakeholders to

explore and compare persona evolution across discrete time windows, rather than monitoring a live data stream.

4.1.2 Dataset and Domain Constraints (OPeRA / E-Commerce)

The prototype was developed and validated exclusively within the e-commerce domain using the OPeRA dataset [31] derived from Amazon shopping sessions. This single-domain scope directly shapes the system's semantic foundation: the Behavioral Key Metrics (BKMs), the action vocabulary (e.g., search, filter usage, product exploration), and the LLM prompting templates are aligned with online shopping behavior and interface concepts. Although the codebase supports modular extension of BKM extraction, the associated feature descriptors and LLM-generated persona narratives are domain-specific. They require a systematic redefinition of behavioral events and user intents to be applicable in other contexts.

The filtered dataset comprises 692 sessions from 51 unique users, totaling 28,904 action records. This comparatively small user pool constrains statistical representativeness and increases the likelihood that the resulting cluster structures reflect sample-specific characteristics rather than stable, generalizable archetypes. Accordingly, the generated personas should be interpreted as exploratory Proto-Personas rather than validated user models.

Another constraint involves the temporal dimension, which is essential to Research Question C and the project's goal of identifying "persona drift." The OPeRA dataset covers a three-month collection period from March to May of 2025. While this known timeframe enables meaningful month-over-month comparisons, the observation period of only three months limits the scope of detectable behavioral evolution. Longer-term patterns, such as annual cycles, gradual lifestyle changes, or sustained shifts in purchasing behavior, cannot be captured within this timeframe. Although the dashboard visualizes monthly transitions as described in Section 1.5.3, interpretations of 'persona evolution' must acknowledge the temporal boundaries of the underlying data.

4.1.3 Pipeline and Methodology Constraints

The data-driven persona generation pipeline is based on methodological decisions suitable for demonstrating the proposed approach. However, these decisions impose constraints on generalizability, interpretability, and measurement precision.

At the feature engineering level, the behavioral key metrics (BKMs) used for clustering were derived from the OPeRA action schema and its fourteen click types. Although metrics such as purchase intent ratio, filter usage, and review engagement capture meaningful behavioral dimensions in an e-commerce setting, they remain tightly coupled to shopping-specific action semantics. Therefore, applying the pipeline to other domains or datasets would require re-specifying the BKM set, including revising the mappings from event types to behavioral constructs and the corresponding metric definitions. Additionally, the current BKMs aggregate counts at the session level and do not represent the temporal structure within sessions.

Consequently, sequential patterns (e.g., funnels, transitions, or action ordering) are not captured, even when they may be behaviorally informative.

The clustering stage uses K-Means with a fixed cluster count of five, a pragmatic choice informed by the practical consideration that stakeholders can more readily internalize a limited set of distinct personas. While this approach is practical, it limits the pipeline's ability to reveal finer or coarser segmentations that the data might suggest. Quantile transformation maps each BKM to a uniform distribution on [0, 1], which ensures visual comparability across metrics in radar charts. This transformation, however, discards information about the original value distributions and converts all features to the same spread regardless of their natural variance.

Finally, cluster assignments are produced under a fixed random seed. This ensures reproducibility for a given configuration, but it does not address potential instability across alternative initializations.

The LLM-based enrichment component transforms quantitative cluster profiles into narrative Proto-Personas and introduces dependencies on prompt formulation and model behavior. While Pydantic schema validation enforces the structural correctness of generated outputs, it does not ensure semantic accuracy. The model may generate details that are plausible yet insufficiently grounded, particularly in narrative fields where strict factual anchoring is difficult to guarantee. In line with the project documentation, the quality of LLM outputs is not formally evaluated. The component is used for enrichment, not as part of the analytical core. This reflects the prototype's emphasis on demonstrating end-to-end feasibility rather than validating natural language generation.

Taken together, these constraints do not negate the pipeline's contribution; however, they limit the strength and scope of its claims. The system shows that behavioral clustering, LLM-based enrichment, and temporal visualization can be combined into a coherent persona development workflow. However, systematic validation and broader-domain generalization are topics for future work.

4.1.4 Dashboard and Visualization Constraints

Scalability constitutes a primary limitation of the dashboard's static data loading approach. Persona and session data are bundled as JSON files served directly from the frontend, which simplifies deployment but limits scalability. For the moderately sized OPeRA dataset, this approach is sufficient; however, larger behavioral datasets would necessitate pagination, incremental loading, or a server-side API; mechanisms not implemented in this prototype. This design decision reflects the prioritization of demonstrability over production-grade performance.

4.1.5 Reproducibility and Deployment Constraint

The pipeline produces static JSON artifacts rather than maintaining a live database, enabling full traceability of generated personas to their source data. This approach supports the

transparency goals outlined in the project vision, as each output file can be versioned and audited independently. However, refreshing the persona landscape requires re-executing the complete pipeline rather than performing incremental updates.

Reproducibility is ensured through fixed random seeds and explicit configuration parameters. Identical inputs consistently yield identical clustering results, which is essential for scientific validity. At the same time, any parameter change, such as adjusting the cluster count or switching the LLM backend, may alter outcomes, requiring careful configuration management. This sensitivity is particularly pronounced for LLM-generated content, where minor prompt variations or model updates can produce different narrative outputs. This behavior reflects a fundamental characteristic of current large language models rather than a limitation specific to the implemented solution.

4.2 Dataset

This section describes the dataset used in the implementation and explains why it is suitable for data-driven persona development. It also outlines the dataset's collection procedure, structure, preprocessing, and resulting constraints for interpretation and transferability.

4.2.1 Dataset Overview

The implemented system relies on the OPeRA (Observation, Persona, Rationale, and Action) dataset [31], which is designed to support the evaluation and simulation of human online shopping behavior. The dataset was introduced to address a recurring limitation of existing interaction datasets. Many collections capture only observable actions (e.g., clicks or completed tasks) without providing the corresponding internal reasoning. Others depend on synthetic trajectories that may not reflect authentic decision-making processes. OPeRA links user profiles to their browsing histories and captures aspects of decision rationale. This connects measurable behavior to user-level attributes and intent. The original dataset documentation positions OPeRA as a benchmark resource for behavioral modeling tasks, such as next-action prediction. The goal of next-action prediction is to infer a user's next step based on their interaction history and profile. For the present project, OPeRA's value lies less in reproducing benchmark tasks and more in providing empirical behavioral traces that can be aggregated into interpretable behavioral markers and combined with static user attributes to characterize and explain clusters.

4.2.2 Data Collection Process

The data was collected through a longitudinal study in which participants performed regular shopping activities on Amazon over a reported observation period of four weeks [31]. Data acquisition used a custom Google Chrome extension, ShoppingFlow [32], which logged interactions and contextual web information in the background to avoid disrupting natural behavior. A distinctive aspect of the collection methodology is the acquisition of "just-in-time"

rationales: at random intervals, reported as an 8% probability following significant actions, the extension presented a short pop-up prompt asking participants to describe the reasoning behind their current action in natural language. In addition to behavioral logs, participants provided demographic and psychographic information through structured online surveys and, in some cases, optional semi-structured interviews.

The participant pool is described as US-based, with inclusion criteria requiring participants to be at least 18 years old, English-speaking, and existing Amazon users. This combination of longitudinal interaction traces and user-level profiling data is particularly relevant for this thesis because it enables both (i) the derivation of behavior-based features from repeated sessions and (ii) the interpretation of resulting clusters in terms of stable persona attributes.

4.2.3 Dataset Structure and Components

OPeRA is structured around four components that together describe a user's interaction trajectory within a shopping session. The Observation (O) component captures the state of the web environment at each step, including raw HTML, a simplified semantic HTML representation, and screenshots. The Persona (Pe) component provides a profile for each participant, including demographic variables (e.g., age and income), shopping preference indicators (e.g., brand loyalty and price consciousness), and psychometric descriptors such as Big-Five traits and MBTI-based personality labels. The Rationale (R) component consists of self-reported text explanations collected for a subset of actions and is intended to serve as ground truth for user intent at decision points. The Action (A) component records timestamped interaction sequences, including user inputs and navigation-relevant events. Click actions are annotated with semantic identifiers that express their functional role (e.g., clicking a product title in a search result rather than a generic page region) [31]

This structure directly maps onto the needs of data-driven persona development. In the present pipeline, (Pe) serves as the primary basis for user profiling and for interpreting clusters in terms of human-understandable characteristics, while (A) provides the main input for behavioral feature engineering and the extraction of behavioral knowledge markers from interaction sequences. By contrast, (O) and (R) are not fully utilized in the current implementation, as raw HTML and screenshots are not processed, and rationale texts are not yet integrated into the feature pipeline or the prompting strategy. As a result, the current system primarily operationalizes persona development as a synthesis of static user attributes and observed action patterns, while still preserving a pathway for future intent-enriched modelling based on the R component.

4.3 Dataset Flexibility: Beyond OPeRA

While this prototype is instantiated using the OPeRA dataset, the underlying architecture is decoupled from the specific data source. The system's design is schema-agnostic, requiring only three core inputs to function with a new dataset.

3. **Session-Level Behavioral Data**: A log of timestamped user actions used to calculate the ten key metrics. Adapting to a new domain (e.g., a video streaming service) would primarily require redefining these metrics (e.g., replacing "Add to Cart" with "Add to Watchlist").
4. **User Identifiers**: A key to link disparate sessions to unique individuals, enabling the longitudinal tracking of behavior.
5. **Qualitative Transcripts**: Textual data (interviews or survey responses) to fuel the LLM synthesis.

4.4 System Workflow and Processing Pipeline

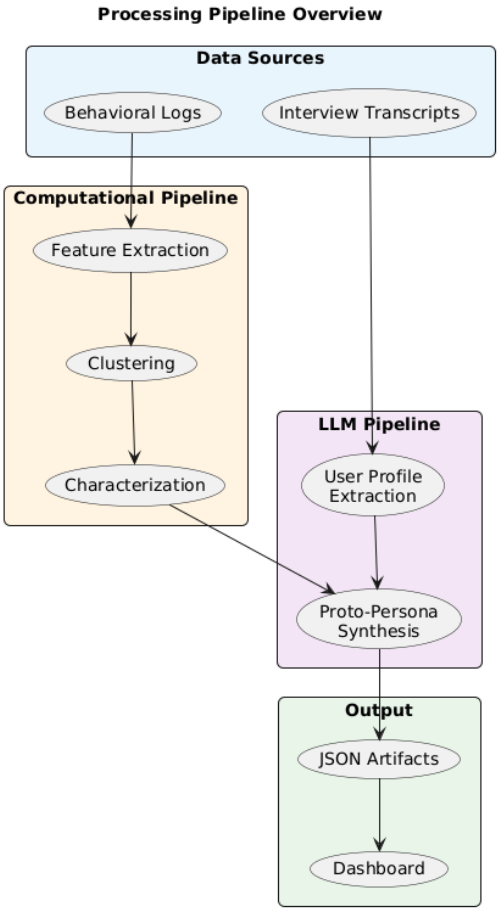


Figure 3 - End-to-end processing pipeline transforming raw behavioral data and interview transcripts into interactive persona artifacts.

This section delineates the end-to-end processing workflow that transforms raw behavioral interaction logs and qualitative interview material into structured persona artifacts for dashboard-based analysis. The pipeline employs a sequential, modular architecture comprising nine phases: data acquisition, preprocessing, feature extraction, clustering, cluster characterization, generation of user profiles, Proto-Persona synthesis, artifact export, and dashboard consumption. This approach separates computational processing from LLM-based synthesis steps.

4.4.1 Architectural Overview and Execution Modes

The processing workflow has been implemented as a Python backend with a central runner script (run_pipeline.py). This entry point triggers a small set of specialized scripts and allows either a full end-to-end run or the discrete execution of specific phases. The backend is split into two parts: a data pipeline for dataset loading, initial feature calculations, and clustering, and an LLM pipeline for generating structured user profiles from transcripts and synthesizing cluster-level Proto-Personas. This separation was primarily introduced due to hardware constraints when running LLMs locally, and it enables a workflow in which data acquisition is performed once, while clustering parameters can be iteratively refined without the computational overhead of repeatedly executing the full LLM pipeline.

Conceptually, the workflow follows three stages: (1) computational clustering to define behavioral segments, (2) semantic extraction to generate rich textual content, and (3) interactive visualization within the user interface.

4.4.2 Phase 1: Data Acquisition

The pipeline begins by acquiring the dataset from HuggingFace [33], using a dedicated download script that retrieves three filtered tables: user-level data (including demographics and interview transcripts), session-level metadata, and action-level interaction events. To avoid redundant network requests during repeated executions, the download component employs caching and

skips configurations that already exist locally. The resulting CSV files are stored in a structured data directory, forming the immutable input layer for all downstream processing steps.

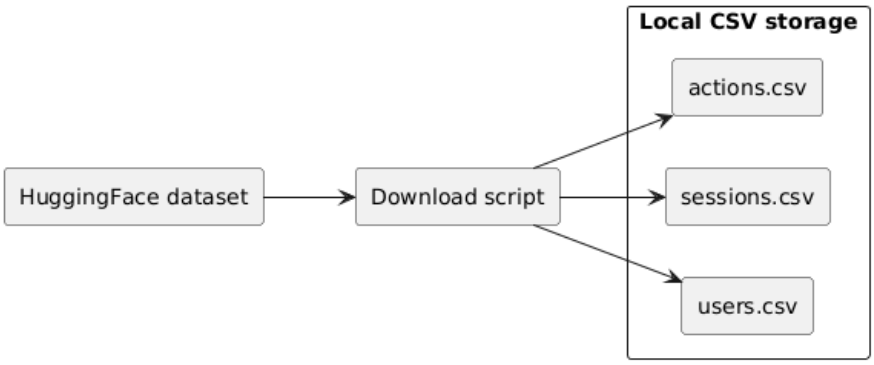


Figure 4 - Data acquisition workflow from HuggingFace to local storage

4.4.3 Phase 2: Data Preprocessing

Preprocessing consolidates session-level and action-level information into a consistent analytical representation. A central step in this process is the extraction and validation of timestamps. Session identifiers encode temporal information, including the start and end time. This temporal information is parsed to compute the duration of each session in seconds and to filter invalid or malformed records. Subsequently, the pipeline assigns valid sessions to monthly cohorts, ensuring the preservation of chronological sorting. This process facilitates subsequent analyses of persona evolution across time windows. This design prioritizes temporal aggregation, addressing it as a primary concern, which is important for assessing drift and stability in data-driven persona representations.

4.4.4 Phase 3: Feature Extraction via Behavioral Key Metrics

To represent user behavior in a compact and comparable form, the system derives a fixed set of ten Behavioral Key Metrics (BKMs) from action-level events at session granularity.

These metrics operationalize key aspects of interaction behavior: engagement, shopping intent, decision-making, and outcome-oriented interaction.

And thereby provide the feature space used for subsequent clustering and visualization. This table offers a comprehensive overview of the entire BKM set utilized in this thesis.

Table 2 - List of Behavioral Key Metrics

Category	Metric	Description
Engagement	session_duration_seconds	Time commitment per session
Engagement	action_density	Actions per second (interaction intensity)
Engagement	total_action_count	Session complexity indicator
Intent	purchase_intent_ratio	Proportion of purchase-related actions
Intent	search_ratio	Goal-directed search behavior
Intent	product_exploration_ratio	Browsing depth (product link clicks)
Decision	review_engagement_ratio	Information-seeking behavior
Decision	filter_usage_ratio	Criteria-driven filtering
Decision	option_selection_ratio	Product configuration actions
Outcome	input_ratio	Active text entry (search queries, forms)

Beyond providing coverage of conceptually distinct behavioral dimensions, the selected BKMs are designed to be comparable across heterogeneous sessions. Specifically, several metrics are expressed as ratios or intensity measures rather than raw counts. This methodological shift reduces sensitivity to differences in session length and interaction volume while preserving meaningful behavioral signals.

4.4.5 Phase 4: Behavioral Clustering and Visualization Coordinates

The subsequent clustering stage employs K-Means clustering in the BKM feature space to group sessions into behaviorally coherent segments. The selection of the number of clusters is determined by the pipeline's evaluation of candidate configurations using the Silhouette Score. This quantitative method balances internal cohesion and external distinctiveness, as the metric specifically calculates the similarity of a data point within its own cluster relative to data points in neighboring clusters [1]. The configuration is centralized to ensure reproducibility, including a fixed random seed and explicit bounds for the cluster search.

In addition to cluster assignment, the pipeline computes two-dimensional coordinates via Principal Component Analysis (PCA) to support spatial visualization in the dashboard. The integration of both session embeddings and cluster centroids facilitates the rendering of point clouds, centroid markers, and temporal movements of clusters within a consistent coordinate system by the frontend.

4.4.6 Phase 5: Cluster Characterization and Interpretable Labels

After clustering, each segment is characterized statistically to identify the behavioral traits that distinguish it from the overall population. The pipeline compares cluster centroids against global means by computing z-scores for each BKM, thereby highlighting metrics that are substantially above or below the population baseline. To ensure interpretability, the system extracts the strongest positive and negative deviations per cluster and utilizes these signals as the foundation for descriptive summaries.

To facilitate human-readable persona communication, the pipeline generates cluster labels algorithmically using a descriptor mapping that translates high-scoring BKMs into adjectives and brief explanations. The resulting naming scheme yields segment titles such as "Filter-Focused" or "Search-Driven," paired with concise taglines reflecting the dominant behavioral patterns. This step bridges the gap between quantitative cluster profiles and stakeholder-facing language, while remaining grounded in measurable behavioral differences.

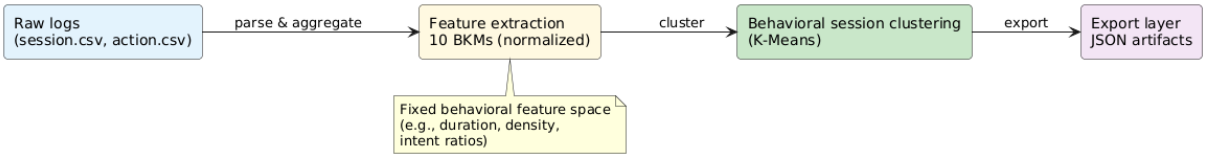


Figure 5 - Simplified processing pipeline from raw session and action logs to JSON artifacts

4.4.7 Phase 6: LLM-Based User Profile Generation from Interviews

The LLM pipeline is designed to complement behavioral segmentation with qualitative user context derived from interview transcripts. For each user, the system prompts a language model to extract a structured profile that covers demographic attributes, psychographic cues, and reported shopping behavior. The extracted profiles are then validated against a Pydantic schema, and each field is augmented with a short piece of textual evidence (typically a supporting quote).

Operationally, the extraction procedure is designed for resumability: users that have already been processed are skipped, thereby enabling interrupted runs to continue without reprocessing completed profiles. Furthermore, the system supports multiple LLM providers through a configuration layer, allowing the same extraction workflow to be executed with different model backends.

4.4.8 Phase 7: LLM-Based Proto-Persona Synthesis at Cluster Level

The system is designed to produce actionable persona artifacts. To do so, it synthesizes cluster-level Proto-Personas by aggregating individual user profiles within each group. The groups are formed by month and cluster. The aggregation strategy collects unique user profiles associated with the sessions of a given segment, ensuring that each user contributes at most once per synthesis context. In addition to the aggregated profiles, the prompt includes cluster metadata such as the cluster label, session counts, and key behavioral. This combination enables the LLM to align qualitative traits from interviews with quantitatively identified behavioral signatures.

The generated Proto-Personas are validated against a schema that enforces minimum data quality. Specifically, the pipeline incorporates checks that reject persona outputs containing excessive "unknown" values. This function serves as a safeguard against the poor quality of generations and promotes the model to produce sufficiently complete persona descriptions when evidence is available.

4.4.9 Phase 8: Artifact Export for Frontend

All outputs are exported as structured JSON artifacts and stored in directories that are accessible to the frontend. These artifacts define a stable interface between the analytical backend and the dashboard, including cluster definitions, session-level embeddings, temporal cluster statistics, visualization metadata, cluster-level Proto-Personas, and LLM-extracted user profiles.

Table 3 - Artifact Export

Artifact	Content
personas.json	Cluster definitions with centroids and metrics
sessions.json	Individual session data with PCA coordinates
monthly_clusters.json	Temporal cluster positions and statistics
metadata.json	Pipeline metadata (timestamps, PCA bounds)
cluster_personas/*.json	LLM-generated Proto-Personas per month/cluster
users/{id}.json	Individual LLM-extracted user profiles
llm_users.json	Combined user profiles for frontend

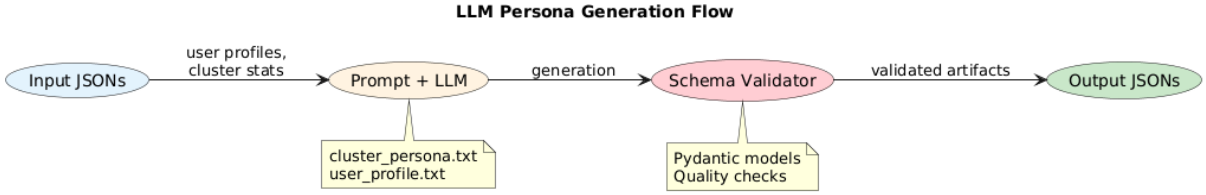


Figure 6 - LLM-based generation pipeline with schema validation for producing structured persona artifacts.

4.4.10 Phase 9: Dashboard Consumption and Interaction Logic

The React-based dashboard consumes the exported JSON artifacts via asynchronous data loading. By retrieving personas, sessions, temporal cluster statistics, and metadata in parallel, the interface is capable of rendering cluster summaries, interactive scatter plots, radar charts, and drill-down views without requiring a live backend connection for each interaction. The dashboard ensures the maintenance of a synchronized state across the selected time window, the active cluster filter, and the corresponding detail panels. This capability enables analysts to transition seamlessly between segment-level overviews and session- or user-level evidence.

4.4.11 Summary: End-to-End Flow and Design Rationale

In summary, the workflow implements a deliberate separation between computational analytics and LLM-based narrative synthesis. The computational steps (feature extraction, clustering, and characterization) provide reproducible segment definitions grounded in behavioral data, while the LLM steps contribute interpretable user context and persona narratives derived from qualitative interviews. By exporting all results as versioned JSON artifacts, the system supports a

lightweight frontend that focuses on visualization and interaction, while the backend remains responsible for reproducible preprocessing and persona generation.

4.5 Explanation of the Dashboard

This section introduces the implemented Proto-Persona Evolution Dashboard as the primary interface for exploring the outputs of the processing pipeline described in the preceding sections. The dashboard enables stakeholders to inspect behavior-based clusters, drill down to session-level information, and review LLM-synthesized Proto-Personas across multiple time windows. By doing so, it implements the UI principles established in Section 2.3.4 by translating abstract guidelines, such as progressive disclosure, evidence-based transparency, and temporal comparability, into concrete interaction and visualization mechanisms.

4.5.1 Frontend Structure

The frontend application is implemented as a single-page application that renders all views dynamically in the browser without requiring page reloads. This architectural choice enables smooth transitions between different sections of the dashboard and maintains application state consistently across user interactions.

Component Architecture

The application organizes its interface elements into three hierarchical categories that reflect different levels of abstraction and reuse.

Layout Components establish the structural foundation of the application. The sidebar component provides persistent navigation and contextual information about the currently loaded dataset, remaining visible across all views. A dashboard layout wrapper ensures consistent spacing and responsive behavior throughout the interface.

View Components represent complete page-level sections that users can navigate between. The primary view presents the cluster distribution visualization alongside the persona list, thereby serving as the main entry point for exploration. A secondary documentation view provides guidance and background information about the system.

Detail Components handle focused presentation of specific data elements. These include the Proto-Persona card, which displays synthesized cluster information; the persona detail panel, which shows behavioral metrics and charts; and the session detail panel, which presents individual session data with associated user profiles.

Routing and Navigation

The application implements a minimal routing scheme with two primary destinations: the cluster exploration view and the documentation view. This focused structure reflects the prototype's emphasis on the clustering and persona synthesis workflow rather than attempting to cover additional use cases.

The sidebar navigation uses visual indicators to communicate the currently active view, with highlighted backgrounds and accent colors distinguishing the selected destination from available alternatives.

Data Loading Strategy

Rather than requiring a continuously running backend server, the frontend loads pre-generated data files at startup. This static data approach offers several advantages for a research prototype, including simplified deployment, reproducible results, and the ability to version control complete datasets. The application retrieves cluster definitions, monthly aggregations, session data, and synthesized Proto-Personas, then manages this data through application state. This architecture cleanly separates the data generation phase (which is executed through the backend pipeline) from the presentation phase (which is handled entirely by the frontend).

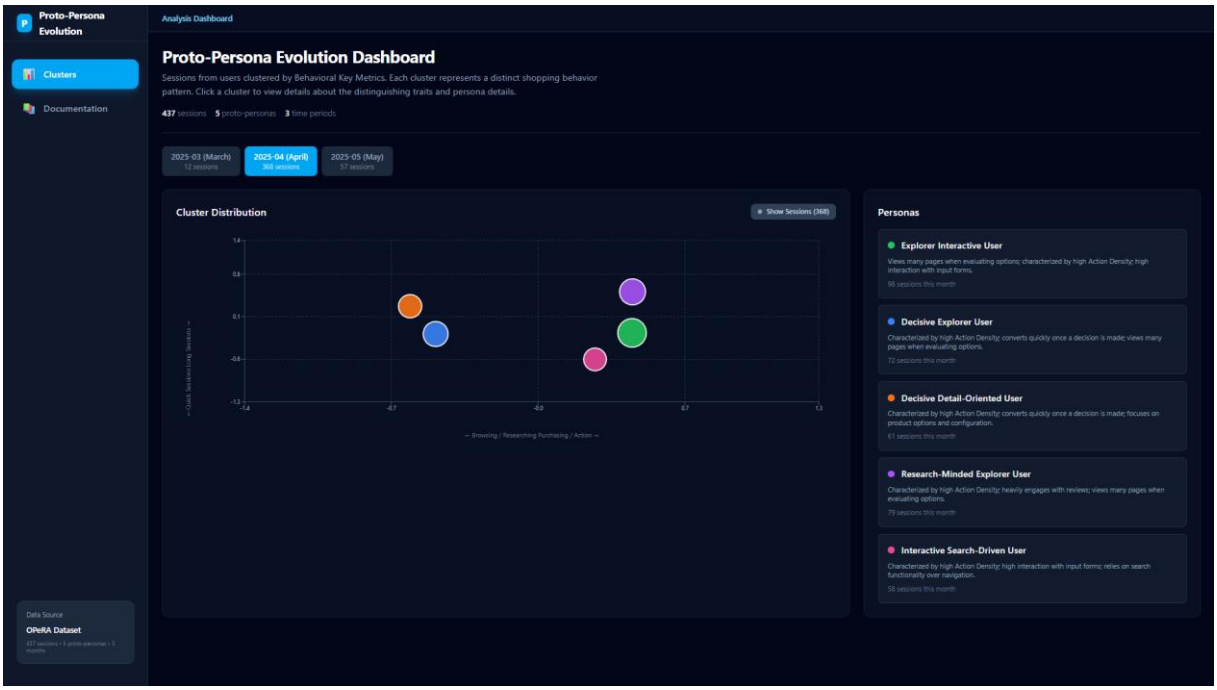


Figure 7 - The Proto-Persona Evolution Dashboard main view, showing the cluster distribution scatter plot, persona list, and sidebar navigation.

4.5.2 The Clustering View: Understanding User Segments

The primary interface for data exploration is the clustering view, which centers on a scatter plot visualizing the identified user segments. In this visualization, each cluster is represented as a colored bubble. The graphical attributes of these bubbles, specifically their position, size, and color, are utilized to encode distinct dimensions of the underlying behavioral data, enabling rapid visual analysis of the user base structure.

Visualization Semantics

The scatter plot employs a spatial organization, arranging clusters along two principal behavioral dimensions derived from the dataset. The horizontal axis delineates a behavioral spectrum, ranging from "browsing and research-oriented" behavior on the left to "purchasing and action-oriented" behavior on the right. The vertical axis indicates session intensity, ranging from "quick,

focused sessions" at the bottom to "longer, exploratory sessions" at the top. These interpretive labels were established by analyzing the factor loadings of the behavioral metrics on the principal components used for dimensionality reduction.

The size of each bubble is proportional to the number of sessions assigned to that specific cluster, thereby conveying population density. This visual encoding method renders the relative prevalence of behavioral patterns immediately apparent; larger bubbles denote dominant user behaviors, while smaller bubbles indicate niche segments. Moreover, the color assigned to each bubble functions as a consistent identifier across the application. As users navigate between different views or time periods, this color consistency ensures that specific clusters can be tracked visually without ambiguity.

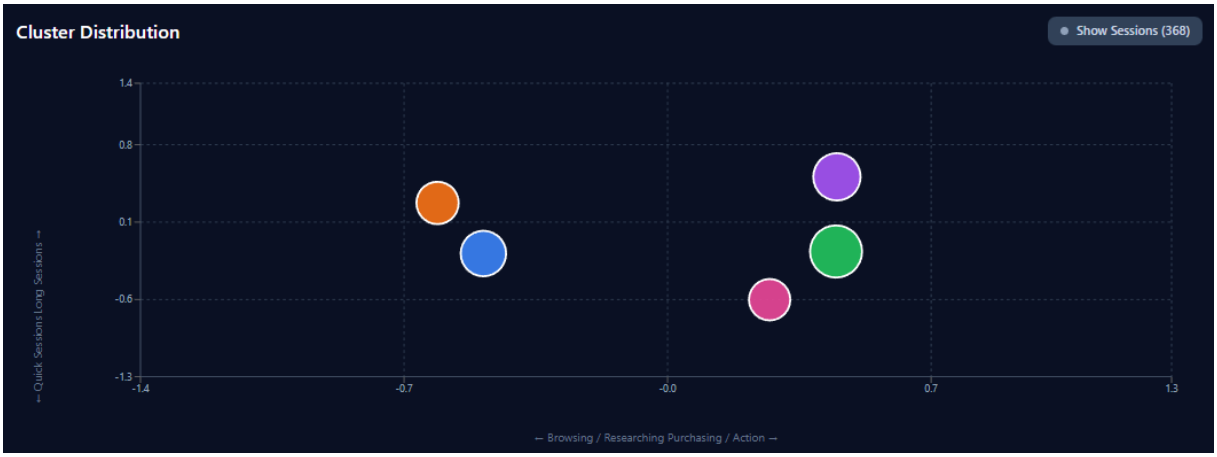


Figure 8 - Screenshot of the cluster distribution scatter plot.

Interaction Design

The visualization aligns with the principle of progressive disclosure, as outlined in the stipulated requirements. The initial view remains unobscured, presenting solely the aggregated cluster centroids. Users can interact with the system to reveal deeper layers of data.

Drill-down: Selecting a cluster bubble initiates the navigation process, directing the user to the detailed panel that is delineated in Section ‘Exploring Individual Clusters’

Data Overlay: A toggle control allows users to project individual session data points onto the plot, visualizing the internal variance and distribution density of sessions around the cluster centroids.

Contextual Details: Hovering over elements triggers tooltips displaying exact metrics, such as cluster names and session counts.

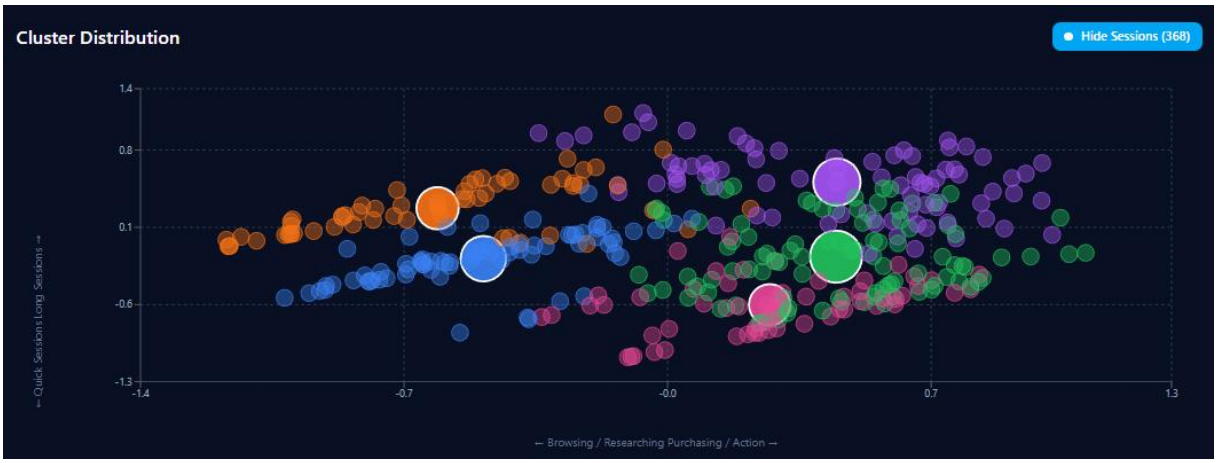


Figure 9 - Screenshot of the cluster distribution scatter plot with all individual sessions shown.

4.5.3 Exploring Individual Clusters

Upon selecting a group session cluster, the interface transitions to a dedicated detail panel. This component functions as an analytical deep-dive, structuring information hierarchically from quantitative metrics to qualitative descriptions.

Distinguishing Traits

The top section of the detail panel immediately identifies what makes the selected cluster unique. Rather than displaying raw data, the system calculates the z-score for each behavioral metric to determine how far the cluster deviates from the global average. The interface employs a filtration process to display only the most significant deviations, which are defined as traits where the behavior is distinctly different.

For instance, Figure 9 presents a cluster designated as "Interactive Search-Driven User." In this case, the system highlights a high Input Ratio and Search Ratio as positive distinguishing traits (green), while noting a significantly low Product Exploration Ratio (red). This finding suggests that users in this segment predominantly utilize search bars and forms rather than browsing categories. In contrast, Figure 10 presents a "Decisive Detail-Oriented User," where the distinguishing traits shift to a high Option Selection Ratio, indicating a focus on configuring products rather than searching for them.

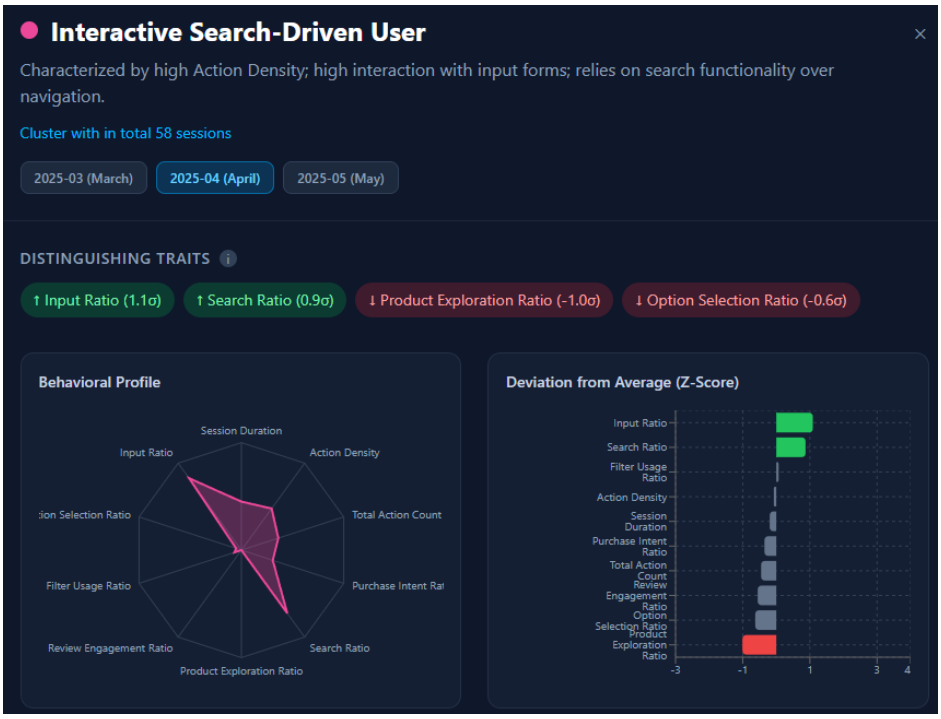


Figure 10 - Top part of detail panel for the "Interactive Search-Driven User" cluster.

Behavioral Profile Visualization

To provide a holistic view of the cluster's behavior, the middle section employs a dual-visualization strategy, allowing stakeholders to view the same data through different cognitive lenses.

The Radar Chart: A radar chart plots ten behavioral key metrics on normalized axes. This visualization is particularly effective for recognizing behavioral "shapes." As illustrated in Figure 9, the "Search-Driven" user profile generates a shape that is notably skewed toward the left (Input/Search axes). Conversely, the "Decisive" user depicted in Figure 10 demonstrates a pronounced peak in the bottom-left quadrant (Option Selection), creating a visually distinct footprint.

The Deviation Bar Chart: Adjacent to the radar chart, a horizontal bar chart displays these metrics as standard deviations from the mean (z-scores). This view is essential for precise comparison, as it clearly separates behaviors that are simply "average" (near the center line) from those that define the cluster (extending far to the left or right).

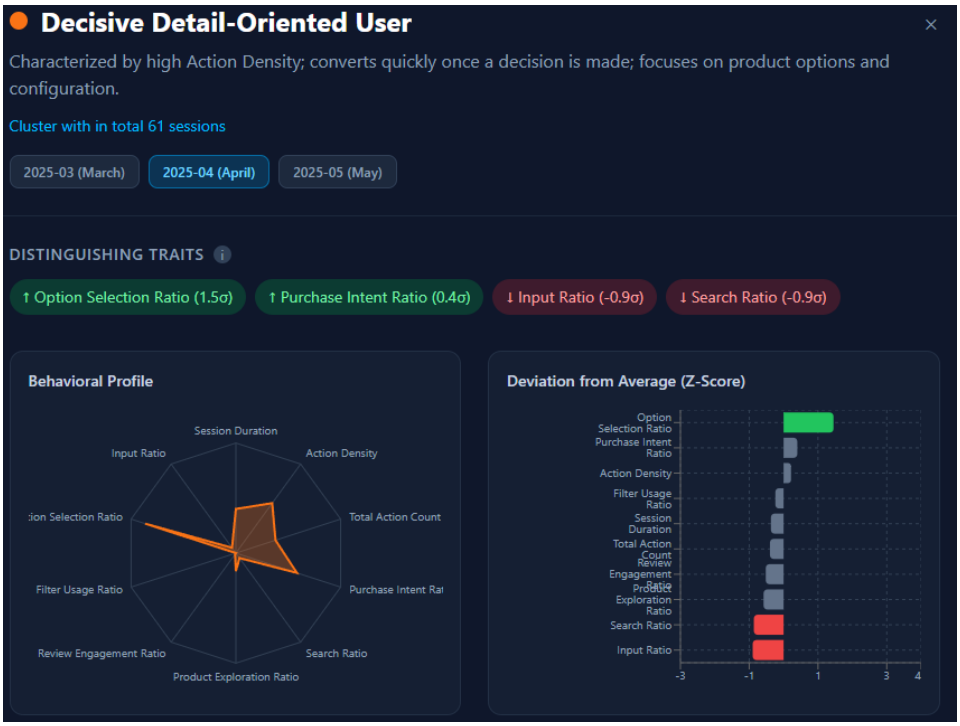


Figure 11 - Top part of detail panel for the "Decisive Detail-Oriented User" cluster.

4.5.4 The Synthesized Proto-Persona

While the metrics delineated in Section 3.4.3 provide the statistical foundation of the cluster, they remain abstract. In order to establish a connection between quantitative signals and human-centric design, the lower section of the detail panel displays a "Proto-Persona" that has been synthesized by the system's Large Language Model (LLM).

From Metrics to Narrative

The Proto-Persona is not a static creative artifact; rather, it is a dynamic synthesis of interview transcripts associated with the users in the current cluster. The pipeline identifies dominant qualitative patterns, including demographic details, motivations, and pain points, and generates a coherent user profile that explains the underlying reasons for the observed behavioral metrics.

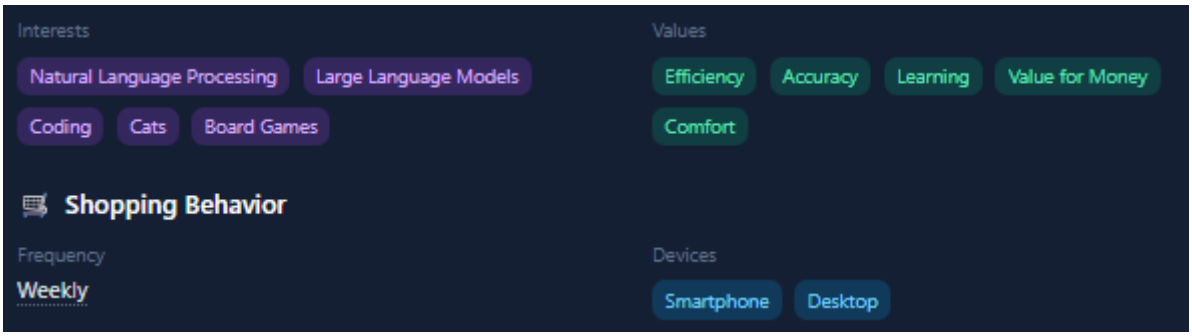


Figure 12 - Example snippet of a Proto-Persona from a cluster

4.5.5 Exploring Individual Sessions

When the "Show Sessions" toggle is active on the main scatter plot, users can select individual data points to open the Session Detail Panel. This panel presents the behavioral fingerprint of a single session alongside its associated user profile, enabling a direct comparison between individual user actions and the broader cluster trends.

Evidence-Based Transparency

A critical challenge in deploying Generative AI for user modeling is the risk of "hallucination," defined as the generation of plausible but factually incorrect information. To mitigate the aforementioned risk and foster trust in the automated outputs, the Session Detail Panel enforces a strict principle of evidence-based transparency.

For each qualitative attribute extracted by the system, such as age range, occupation, or goals, the interface provides a confidence meter and a direct lineage to the source data. As demonstrated in Figure 12, when the cursor is placed over a particular value, such as an age range of "30-35," a detailed tooltip is revealed. This overlay presents two critical pieces of meta-information: a quantitative confidence score and the specific textual evidence employed by the Large Language Model to ascertain that value.

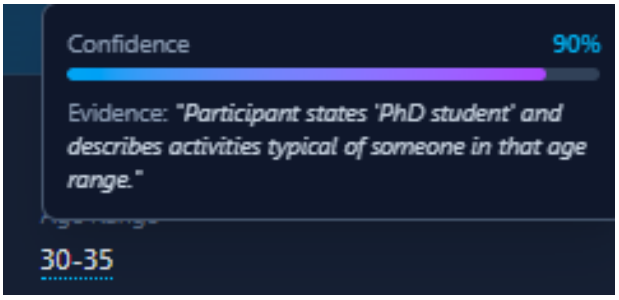


Figure 13 - The confidence tooltip in the Session Detail Panel.

4.5.6 Temporal Navigation: Detecting Changes Over Time

A distinguishing feature of the implemented dashboard is its support for longitudinal analysis. In contradistinction to conventional persona artifacts, which are frequently static representations, this system is engineered to visualize the evolution of user segments.

The Month Selector

Navigation across time is handled via a timeline control located below the page header. This component displays available monthly data snapshots, including the specific session counts for each period. Selecting a new month triggers a global state update:

Cluster Distribution: The bubbles in the main scatter plot resize and shift position to reflect the cluster composition for the selected month.

Metric Recalculation: The distinguishing traits and radar charts in the detail panel update to represent the behaviors observed in that specific window.

Detecting Persona Drift

This mechanism enables stakeholders to observe "persona drift." By toggling between months, a product manager might observe that the "Interactive Search-Driven" cluster is diminishing in size or that its Purchase Intent metric is declining. Such shifts could indicate that a recent UI update has made search less effective, or that the user base is naturally migrating toward different behaviors. While the current prototype requires manual navigation to detect these changes, it successfully establishes the infrastructure for dynamic, time-aware personas.

4.6 Technical Stack

The solution follows a two-tier architecture separating data processing (backend) from interactive visualization (frontend), enabling independent development and static deployment.

4.6.1 Data Processing and Backend

The data processing pipeline was implemented in Python, leveraging its well-established ecosystem for data science applications. The preprocessing and vectorization of behavioral data were conducted using the Pandas and NumPy libraries. In accordance with established methodologies in data-driven persona construction, clustering and dimensionality reduction were performed with scikit-learn, employing K-Means clustering and Principal Component Analysis (PCA). To ensure the structural validity and transparency of the generated Proto-Personas, LLM response validation was enforced through Pydantic (v2.0+), which guarantees strict adherence to predefined schema specifications for all outputs.

4.6.2 LLM Integration

The system adopts a model-agnostic approach for persona narrative generation, designed to allow straightforward substitution of LLM providers without architectural changes. The implementation supports running LLMs either locally on the user's machine or through cloud-based API services. For the present work, persona generation was performed using the ‘gemma3’ model, which provided satisfactory output quality while remaining compatible with the available hardware resources.

4.6.3 Frontend and Visualization

The user interface was developed as a Single Page Application (SPA) using React (v19.2.0). We selected a component-based architecture to support modular exploration of user clusters. Interactive visualizations, including scatter plots and radar charts, were rendered using Recharts (v3.6.0) to allow stakeholders to visually explore the separation and characteristics of persona clusters. The build process utilizes Vite (v7.2.4) for efficient bundling, with styling handled by Tailwind CSS.

4.6.4 Deployment

The solution was deployed using GitLab. The source code, CI/CD configuration, and build artifacts are managed in a GitLab repository, enabling reproducible builds and traceable versioning. The dashboard is published via GitLab Pages, providing a static web deployment that can be accessed through a generated Pages URL and updated automatically on each successful pipeline run.

4.6.5 Integration and Reproducibility

To optimize portability, facilitate the reusability of individual components, and support future extensibility, the processing pipeline was decoupled from the visualization layer. The backend processes raw data into static JSON artifacts, which are then bundled with the frontend build.

5. Evaluation

5.1 Setup

The evaluation follows an artifact-centric strategy and focuses on inspecting the correctness, consistency, and inspectability of the system's generated outputs rather than conducting a stakeholder user study. This approach implements NFR4 (scope compatibility), because the project's contribution lies in a working research prototype. Consequently, verification of pipeline and dashboard artifacts is prioritized over an empirical assessment of adoption or perceived usefulness.

Specifically, the backend generates a filtered subset of OPeRA Amazon shopping sessions and processes three monthly windows. Each execution produces intermediate and final JSON artifacts, which are validated against Pydantic schemas before being consumed by the frontend. These schema-bound artifacts establish stable interfaces between pipeline stages, enabling repeated runs under comparable conditions. In this way, the setup operationalizes NFR1 (reproducibility) and NFR3 (modularity) through explicit, machine-checkable contracts between the extraction, aggregation, and visualization components.

The evaluation is structured around three tasks. First, session-level behavioral metrics are used to generate and project clusters into a fixed two-dimensional space for visual inspection. Second, the interpretability and traceability of cluster-level Proto-Personas and their underlying behavioral indicators are inspected within the dashboard. Third, cluster representations are compared across monthly windows to determine if cross-window analysis remains meaningful with consistent visual encoding. These tasks operationalize the system’s core requirements: grounding segmentation in usage data, treating time windows as first-class analysis units, and enabling cross-window comparison as an analysis operation. Additionally, the evaluation includes targeted plausibility checks for transcript-derived statements by comparing selected extracted attributes and persona claims against interview evidence. These checks are used to evaluate the grounding of evidence and the handling of uncertainty rather than to provide comprehensive semantic validation.

5.2 LLM User-Profile Extraction Quality

The automatic extraction of persona-relevant attributes from qualitative interview transcripts constitutes a key component of the implemented prototype. This section evaluates the quality of LLM-generated user profiles that serve as inputs for synthesizing a cluster-level Proto-Persona. Following an artifact-centric rationale, the evaluation focuses on two aspects: whether the extracted attributes are grounded in the transcript evidence and whether uncertainty is transparently handled for under-specified or inferential fields. From a requirements perspective, this subsection primarily operationalizes RQ2’s traceability requirement and FR5’s requirement for evidence and uncertainty signaling by assessing whether transcript-derived profile attributes can be inspected and justified.

The evaluation has two objectives. First, it quantifies extraction accuracy across attribute categories and fields. Second, it identifies systematic error patterns to inform improvements in prompting, schema design, and UI-level transparency mechanisms (e.g., stricter abstention rules and clearer signaling of inference).

5.2.1 Method and Evaluation Criteria

Each extracted attribute–value pair was subjected to a verification process against the corresponding interview transcript using an LLM-as-a-judge procedure. Specifically, an independent judge model was provided with the complete transcript context and the extracted attribute (including the extracted evidence snippet) and classified the extraction into one of three categories.

An attribute was labeled “Correct” if it was explicitly supported by the transcript. The label “Partially Correct” was assigned when the extraction was broadly plausible but lacked specificity, was overly generic, or relied on inference rather than being textually explicit. The attribute was labeled “Incorrect” if it contradicted the transcript or lacked a textual basis (i.e., a hallucination).

The evaluated sample included all users for whom an interview transcript was available after preprocessing. Users without interview material were excluded from the transcript-grounding evaluation, resulting in a final sample of 33 users and 561 verified attributes. The judge evaluation was performed using the gemma3 model executed via Ollama.

This evaluation focuses on the LLM-dependent extraction stage, specifically the process of extracting user profiles from transcripts. Certain fields operationalize behavioral constructs that are not explicitly stated in interviews (e.g., device usage patterns or shopping frequency), so they tend to require inference. These fields are included to assess whether the system appropriately expresses uncertainty; however, their judgments should be interpreted as evidence-grounding quality rather than strict measurement validity of the underlying construct.

5.2.2 Extraction Accuracy Results

Overall accuracy across all evaluated attributes was 78%, with 441/561 attributes judged as Correct, 93/561 as Partially Correct, and 27/561 as Incorrect. These results indicate that transcript-grounded extraction performs reliably for many fields, while under-specified attributes require stricter abstention and clearer uncertainty signaling to avoid over-confident outputs.

Table 4 summarizes extraction quality by attribute category, reporting counts per judgment class and the resulting correct-rate accuracy.

Category	Accuracy	Correct	Partially Correct	Incorrect	Findings
Demographics	**73.6%**	170	34	27	High accuracy on explicit facts (Occupation, Location). Low accuracy on implicit traits (Gender).
Psychographics	**87.9%**	87	12	0	Excellent abstraction of goals and interests from narrative text.
Shopping Behavior	**79.7%**	184	47	0	Strong on concrete behaviors (Device Usage), weaker on frequency quantification.

Table 4 - Extraction Results by attribute category.

As shown in Table 4, correct-rate accuracy is highest for Psychographics (87.9%), followed by Shopping Behavior (79.7%), while Demographics is lower (73.6%). Demographic errors concentrate in fields that are often not explicitly stated in the transcripts, most notably gender.

A field-level breakdown indicates that several demographic attributes are reliably extracted when explicitly present in the transcript. Variables such as geographical location, living arrangement, and occupational status show high precision, with accuracies ranging from 96% to 100%. This pattern suggests that the extraction prompt and schema capture concrete biographical statements effectively when they are stated explicitly.

In contrast, the gender field within Demographics exhibits the lowest performance, with an accuracy of 12%, and is frequently judged as hallucinated when the transcript lacks explicit gender information. This indicates that the extractor tends to fill missing demographic slots with assumed defaults. Because such fields are sensitive and can introduce bias into stakeholder-facing artifacts, they should be treated as optional and default to “unknown/not stated” unless explicit self-identification is present in the transcript evidence.

Several fields are judged as Partially Correct rather than Incorrect, indicating over-specificity where the transcript supports only a general claim. A recurrent example is ShoppingBehavior.frequency_of_use: transcripts often indicate regular purchasing but not a discrete interval (e.g., weekly vs. monthly). In these cases, categorical labels may be plausible but remain insufficiently grounded as explicit facts.

5.2.3 Error Patterns and Interpretation

The observed errors fall into two primary types. First, hallucinations occur in under-specified demographic fields when transcripts lack explicit statements but the system outputs concrete values (most notably gender). Such failures can have high impact because they affect sensitive attributes and can undermine stakeholder trust in the entire persona artifact.

Second, the conversion of qualitative statements (e.g., “regularly orders” or “sometimes buys essentials”) into precise categorical bins can lead to over-interpretation of behavioral frequency and intensity. These cases are typically judged as Partially Correct, indicating that the model captures the direction of the attribute but encodes it with an unjustified level of precision. A suitable mitigation is to represent these outputs as inferred estimates with explicit uncertainty markers rather than as factual categories.

Psychographic fields such as interests and values are often judged as correct when explicitly enumerated in the transcript. However, these fields tend to be judged as partial when outputs summarize implicit criteria as stable “values.” This suggests that the extractor recovers significant narrative content, but benefits from stricter separation between explicitly stated self-descriptions and inferred interpretations, which should be labeled accordingly.

5.2.4 Implications for Prompting, Schemas, and UI Transparency

The findings suggest that LLM-based extraction can reliably populate persona-relevant fields when transcripts contain explicit biographical statements, and that schema validation ensures the structural consistency of the produced artifacts. However, semantic reliability depends on whether prompting and schema constraints enforce abstention and uncertainty handling when evidence is missing. The systematic hallucination risk in gender extraction motivates a stricter policy for sensitive demographic fields. The system should default to "unknown/not stated" unless explicit self-identification is present in the transcript evidence.

For fields that are frequently inferential (e.g., shopping frequency), the schema should allow graded representations (e.g., “unspecified but frequent”) and require evidence spans that justify any categorical bin. At the UI level, these fields should be presented with stronger uncertainty signaling (e.g., “inferred” labels) to reduce the risk that stakeholders interpret plausible estimates as factual statements. Overall, these implications motivate stricter abstention rules and more explicit labeling of inferred fields to preserve evidence-based interpretability and to avoid overstating certainty in longitudinal comparisons.

5.3 Prompt Robustness

5.3.1 Robustness of LLM System-Prompts in Out-of-Distribution Interviews

A targeted robustness test was conducted to evaluate LLM-based persona extraction (system prompts in conjunction with the integrated LLM in the implemented solution). The aim of this test was to check whether the prompt logic still generates consistent and usable profiles even when the input data deviates significantly from the typical interview values of the data set used (i.e., “out-of-distribution” inputs).

Test Setup and Procedure

As test inputs, two additional interview transcripts were created using ChatGPT 5.2, which were deliberately designed to have different characteristics than the original (real) transcripts in the dataset. These synthetic transcripts are available as Interview1.txt and Interview2.txt in the Appendix ZIP folder. The respective prompts for generating the interviews can be found at the top of the respective text files.

In the next step, the identical extraction logic was used that is also used in the regular system process:

1. **Profile extraction per interview:**
   For each of the two transcripts, a structured user profile was generated by the LLM, controlled via our system prompt (user_profile.txt). The results were saved as User-Profile1.json (from Interview1.txt) and User-Profile2.json (from Interview2.txt).

2. **Aggregation into a Proto-Persona:**
   The two generated user profiles were then merged by the LLM into a common Proto-Persona, analogous to the process in the productive dashboard pipeline (cluster_persona.txt). The result is available as ProtoPersonaUP12.json.

This test explicitly validates the two core capabilities of the prompt/LLM component:
(1) Extraction of profile characteristics from a single interview into a consistent JSON format, and (2) Synthesis/aggregation of multiple profiles into a summary Proto-Persona.

Results and Observations

The results show that the prompt-based process works in principle: formally valid, usable user profiles could be generated from both synthetic transcripts (User-Profile1.json, User-Profile2.json), which were then aggregated into a Proto-Persona (ProtoPersonaUP12.json). This confirms the functionality of the extraction and aggregation chain in a test run.

In terms of content, however, it was noticeable that the generated Proto-Persona (ProtoPersonaUP12.json) resembled the patterns that also occur in Proto-Personas from the real data set interviews more closely than originally expected. Although the two synthetic interviews were deliberately designed to contain “different” values, the merged persona appears relatively close to the typical characteristics in some respects. From an evaluation perspective, this result can be interpreted in two ways:

- **Strength (stability/format consistency):**
  The prompt delivers stable, structured outputs even with atypical inputs and does not generate “chaotic” or unusable profiles. This speaks for a certain robustness of the format and structure specifications in the system prompt.

- **Weakness (tendency toward normalization):**
  At the same time, the similarity to the “normal” Proto-Persona suggests that the LLM does not always translate the unusual signals strongly enough into the final persona

attributes, or that there is a kind of “regression to the mean” (normalization toward plausible standard values). This can result in genuine deviations in the data not being sufficiently visible in the persona profile.

In addition, certain attributes were left blank or filled with “unknown”. This value is not particularly helpful for a Proto-Persona and could have been generated with a strongly assumed value.

Conclusion of this Validation

This test validated the LLM component in a controlled scenario: The pipeline was able to generate reproducibly structured user profiles (UserProfile1.json, UserProfile2.json) from Interview1.txt and Interview2.txt and merge them into a Proto-Persona (ProtoPersonaUP12.json). This confirms the system's basic ability to function end-to-end even with “unusual” inputs. At the same time, the result reveals a key limitation: the content differentiation between highly divergent inputs is present, but could be stronger in order to more clearly reflect persona differences in extreme scenarios.

5.3.2 Limitations of LLM Extraction with Minimal, Comment-based Feedback

As a supplement to Validation before, a second, deliberately more rigorous test was conducted to examine the limits of LLM-based persona generation. While the first test used complete (synthetic) interview transcripts as input, this test is based solely on short, realistically formulated user comments. The aim was to check whether our implemented solution with our system prompts (user_profile.txt and cluster_persona.txt) can still derive meaningfully structured user profiles and a Proto-Persona from very “thin” explicit feedback, and where the weaknesses lie.

Test Setup and Procedure

Three individual comments were generated as input and saved as Comment1.txt, Comment2.txt, and Comment3.txt. The prompts for generating the comments are located at the top of each text file. The files mentioned in this subchapter can be found in the Appendix ZIP folder. All three comments describe a similar, plausible UX problem (distracting, automatically rotating banners on a shopping webpage), but each with slightly different wording and focus (e.g., distraction, misclicks, desire for a deactivation option).

Analogous to the productive pipeline, structured user profiles were then generated for each comment, controlled by the same system prompt (user_profile.txt): User-Profile-C1.json, User-Profile-C2.json, and User-Profile-C3.json.

In the final step, these three profiles were merged into a single Proto-Persona (using our system prompts from our implemented solution in cluster_persona.txt). The result is available as ProtoPersonaUPC123.json.

Results and Observations

Formally, the test showed that the extraction and synthesis chain also works technically with comment inputs: JSON-compliant user profiles were generated from all three texts and then aggregated into a Proto-Persona. This basically confirms the functionality of the prompt/LLM component in this scenario.

However, we were less satisfied with the quality of the content, and the results clearly explain why. A single comment usually covers only a very narrow topic (in this case, annoyance with a specific design element) and provides little information about demographics, context, device usage, frequency, categories, or deeper motives. Accordingly, the generated user profiles show a strong asymmetry: The model can identify pain points and goals related to the specific UI problem with relative precision (e.g., “distraction/misclicks due to moving UI elements”), but has to guess, generalize, or mark many other attributes as “unknown”. This is particularly evident in the fact that some of the profiles contain specific demographic values or backgrounds that cannot be reliably derived from the comment itself (e.g., assumptions about age range/origin in individual profiles).

The final Proto-Persona (ProtoPersonaUPC123.json) results in a plausible persona text (“The Deliberate Navigator”) that is consistent with the common problem in terms of content (desire for control, static UI, avoidance of misclicks), but at the same time contains attributes that seem rather hallucinatory or over-inferred, or are only weakly supported (e.g., age distribution “25–30” from two profiles, although the comments themselves do not contain any age information).

Interpretation – What the Test Validates and What Weaknesses it Reveals

This second test validates two things in particular:

1. **Pipeline robustness at the artifact level:**
   The solution can also generate structured profiles and a Proto-Persona from completely different explicit feedback (comments instead of interviews).

2. **Data dependency of semantic quality:**
   The less context there is in the input, the greater the risk that the model will fill in gaps with plausible assumptions. This is precisely why the persona based on three individual comments appears less reliable than personas based on extensive interview transcripts.

In addition, the test confirms a practical observation that is particularly relevant in the case of “uncertain” inputs: the results vary more because the model has more degrees of freedom when the evidence is thin. This variance is not only a quality problem, but also a reproducibility and trust issue, especially when stakeholders could interpret persona details as facts.

Conclusion of this Validation

This second validation tested how well our system prompts and LLM integration can handle very short, comment-based feedback. The test shows that the method is suitable for deriving pain

points and immediate goals from comments and converting them into a rough Proto-Persona. At the same time, it became clear that individual comments provide too little context to reliably infer comprehensive persona attributes, resulting in over-inference, higher variance, and lower content stability. This test thus serves as deliberate proof of the limitations of our approach and supports the assessment that richer explicit data (e.g., interviews) are much better suited for generating robust Proto-Personas.

5.4 Dashboard-Level Checks

5.4.1 Temporal Comparability

Monthly comparability is achieved through a month selector that switches the scatterplot and persona list between March, April, and May. This subsection operationalizes temporal comparability by treating time windows as first-class analysis units and enabling direct cross-window comparison under a consistent visual encoding.

To keep comparisons stable, metadata fixes PCA axis bounds across months to prevent changes in positioning caused by rescaling. In addition, monthly_clusters.json records cluster centroids, sizes, and z-scores for each window. Fixing projection bounds ensures that observed movements reflect changes in the underlying data rather than alterations in visualization scaling. Drift signals are therefore assessed through visual inspection of changes in cluster size, centroid movement, and shifts in distinguishing trait z-scores. For example, a decrease in a cluster's PCA_x coordinate from 0.41 to 0.31 constitutes an observable indicator of movement within the shared coordinate system.

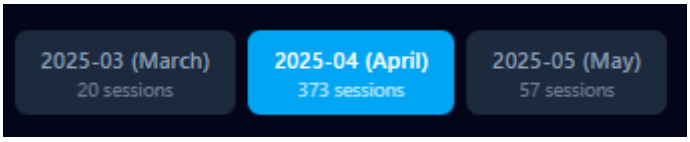


Figure 14 - Illustrates the month selector used for switching between time windows.

However, the current implementation supports temporal analysis primarily through visual inspection and does not provide automated drift alerts, threshold-based notifications, or statistical change detection.

5.4.2 UI Principles Realization

This subsection evaluates whether the dashboard implements the intended interaction principles for traceability and interpretability. The focus lies on progressive disclosure, evidence-oriented transparency cues, and consistent temporal navigation.

Progressive disclosure is implemented through layered navigation. Users start with an overview scatterplot, select a persona card to open a modal containing a radar chart and trait bars, and then access session-level details via a user profile card. Evidence-oriented transparency is supported through trait z-score differentiation and explanatory tooltips, radar charts depicting behavioral metric profiles, and counts of supporting users per demographic attribute. These

cues provide lightweight indicators of how strongly attributes are supported within a cluster and reduce the risk that persona details are read as unconditional facts.

Temporal navigation is reinforced by the month selector, consistent PCA scaling, and an additional month switcher within the persona modal. By combining narrative persona summaries with interpretable visual encodings and short explanatory text, the interface supports interpretation for non-technical stakeholders while retaining access to underlying evidence.

5.5 Summary of Evaluation Findings

Overall, the evaluation shows that the prototype functions as an end-to-end artifact chain. The backend generates schema-valid JSON artifacts, which the dashboard renders without manual post-processing. This supports NFR1 (reproducibility) and NFR3 (modularity) because stable, schema-defined interfaces separate the generation and visualization processes, enabling repeatable pipeline executions.

The dashboard makes the outputs inspectable. Persona-level statements can be traced back to the corresponding user profiles via evidence snippets and metadata links. Behavioral encodings, such as z-score-based trait differentiation and profile visualizations, provide additional insight beyond raw cluster assignments. Thus, the interface operationalizes RQ2's traceability requirement and FR5's evidence and uncertainty signaling requirement by enabling stakeholders to verify the basis of persona claims.

For temporal analysis, the month selector and fixed projection bounds allow for direct cross-window comparison under consistent visual scaling. However, drift assessment remains manual, and the prototype does not yet provide automated alerts, threshold-based notifications, or statistical tests. This limits the extent to which FR4 is operationalized within the current scope.

Finally, transcript-based LLM extraction is often reliable when interviews contain explicit evidence; however, it exhibits systematic failure modes in under-specified or sensitive fields, where the model tends to over-infer. These findings highlight that schema validation secures structural integrity, yet does not guarantee semantic reliability. Accordingly, while the evaluation verifies artifact integrity and provides targeted quality signals for transcript-based extraction, it does not constitute comprehensive validation of cluster-level Proto-Persona narratives or stakeholder usefulness.

6. Results and Discussion

This chapter consolidates the outcomes of the project and critically discusses what has been achieved, what remains unresolved, and what this implies for the broader goal of treating personas as living models rather than static "flat files". Building on the conceptual foundation established in chapter 3 and the implemented end-to-end pipeline and dashboard described in chapter 4, we now focus on evidence-based conclusions.

In line with chapter 1.3.4 "Problem Formulation", the results section answers the three project-specific research questions (A–C) and reflects through the concrete artifact and its validation results. The discussion explicitly links findings back to the derived requirements and aspirations in chapter 3 (e.g., traceability, temporal comparability, and interpretability), ensuring that conclusions remain grounded in the documented design rationale rather than subjective impressions.

6.1 Results

The project set out to operationalize the central claim formulated in the Project Agreement, *"Personas can be generated from data"*, while explicitly addressing the research gaps identified earlier: the "flat file" problem, limited interpretability of clustering outputs, insufficient operationalization of validation, and the neglected temporal dimension (persona evolution). The results are therefore reported along two complementary axes:
(1) the extent to which the prototype can answer the three project-specific research questions (A–C), and
(2) the extent to which the implemented artifact fulfills the requirements and conceptual aspirations defined in Chapter 3.

Research Question A – Translations of Clusters into Personas
The implemented pipeline demonstrates that behavior-based clusters can be transformed into structured, stakeholder-readable Proto-Personas by combining (i) quantitative cluster characterization (behavioral key metrics, z-score-based traits, cluster labeling) with (ii) LLM-supported synthesis from explicit feedback (interview transcripts) into consistent JSON persona artifacts, rendered as persona cards in the dashboard. The dashboard-level drill-down (overview → cluster details → session evidence) operationalizes the intended bridge from abstract segmentation to human-centered representations, thereby directly responding to the interpretability gap discussed in chapters 2.4 and 2.5.

At the same time, the results show that "translation success" is bounded by the quality and richness of explicit input data: where interview content is sparse or does not contain explicit evidence for certain fields, persona attributes become either (a) "unknown" or (b) inferred with a risk of over-interpretation. This limitation is not merely a technical imperfection, it directly affects persona credibility and therefore the usefulness of the artifact as a design tool.

Research Question B – Robustness and Limitations of LLM Extraction
The evaluation in chapter 4 provides concrete evidence that transcript-grounded attribute extraction can be reliable for many persona-relevant fields, but also reveals systematic failure modes. In particular, the extraction quality differs strongly across field types: explicit biographical facts are often extractable when stated clearly, while under-specified or sensitive demographic fields (e.g., gender) show a high risk of hallucination and should be treated as "unknown unless explicitly stated". Furthermore, the two robustness tests demonstrate a key boundary condition: with rich text (synthetic interviews), the pipeline remains structurally stable but shows signs of "normalization" toward typical values. With minimal text (single comments), the model exhibits higher variance and stronger over-inference because the evidence is too thin to support full persona profiles. These results validate that the implemented prompt-and-schema approach works end-to-end, but they also confirm that LLM-based persona enrichment requires stronger abstention rules, stricter uncertainty signaling, and input-dependent expectations to remain trustworthy.

Research Question C – Usability and Cognitive Comprehensibility of the Dashboard
The dashboard design implements progressive disclosure (overview → details → evidence) and makes persona exploration feasible without requiring stakeholders to read raw datasets. This directly instantiates the UI principles defined in chapter 2.3 and the interaction logic described in chapter 3.6, particularly by combining cluster-level summaries with evidence-level inspection at the session/user profile level. In addition, temporal navigation across three months enables comparison of cluster distributions and characteristics under consistent visualization scaling.

However, an important limitation remains: persona drift and evolution are currently detectable, but not systematically surfaced. The prototype relies primarily on manual inspection rather than automated change signals, thresholding, or statistical drift detection. This means the solution partially fulfills the intent of time-aware personas but does not fully operationalize "continuous monitoring" as envisioned in the requirements.

Overall, the results show that the project successfully delivers an end-to-end demonstrator that connects behavioral clustering, LLM-based persona enrichment, and interactive temporal visualization into a coherent workflow. At the same time, the evaluation establishes clear constraints: (i) persona quality depends strongly on the explicit-data richness, (ii) inference-heavy fields require stricter abstention and uncertainty handling, and (iii) temporal analysis is currently interactive rather than automated. These conclusions provide a fact-based basis for the improvements and next steps discussed later in this chapter.

6.2 Were the Goals Achieved?

The goals and expected results defined in chapter 1.5 can be assessed most rigorously when they are mapped to the requirements derived in chapter 3.1. This is important because the goals capture what the project intended to deliver, while the requirements formalize what the solution must be able to do in order to address the gaps discussed in chapter 1 and 2. Therefore, this section evaluates goal achievement through two lenses:

(1) fulfillment of the project goals (chapter 1.5) and
(2) fulfillment of the functional and non-functional requirements (chapter 3.1), based on the implemented prototype described and evaluated in chapter 4 and 5.

6.2.1 Goal Achievement

At a high level, the project achieved its primary IP5 objective: it delivers a working, reproducible prototype that demonstrates how data-driven Proto-Personas can be created, explored, and compared across defined time windows. The solution integrates behavioral clustering, LLM-based persona enrichment from explicit feedback, and interactive visualization into one coherent artifact (chapter 4), thereby directly addressing the “flat file” problem and the missing operational connection between analytics and persona artifacts (chapters 2.5).

However, goal achievement is not binary. While the prototype is functional and demonstrably useful as an exploration and communication tool for a specific dataset, the evaluation also highlights that several aspects, especially those related to automated persona evolution detection and to the reliability of inferred persona attributes, remain limited by scope and by data constraints (chapter 5.2).

To make this assessment explicit, the following summarizes how the implemented solution meets the requirements from Chapter 3.1.

FR1 – Segmentation based on usage data – fulfilled
The prototype can transform session/usage data into feature representations and derive behavioral clusters (chapter 4.2 and 4.3). Cluster characteristics are computed and displayed, enabling stakeholders to understand how segments differ.

FR2 – Persona artifact per cluster – fulfilled
For each cluster, the system produces a structured persona-like artifact (Proto-Persona) based on aggregated user profiles, which are derived from explicit feedback sources. These artifacts are represented consistently in JSON and rendered in the dashboard (chapter 4.5).

FR3 – Time-window-based analysis – fulfilled
The system supports discrete time windows and allows the user to switch between them, enabling temporal inspection of segmentation results (chapter 4.2). This implements the conceptual “snapshot/versioning” approach defined in chapter 3.6.3.

FR4 – Comparison and change signals – partially fulfilled
The prototype enables manual comparison across time windows by providing consistent views and navigation between snapshots. Stakeholders can observe distribution changes and shifts in cluster characteristics through exploration (chapter 4.2 and 4.5).
However, explicit automated change signaling (e.g., drift alerts, statistical change detection) is not operationalized. Evolution is therefore detectable but not systematically surfaced by the system.

FR5 – Evidence/uncertainty indicators – partially fulfilled
The solution includes mechanisms intended to support transparency (chapter 3.6.1) and

provides evidence-oriented views and structured persona outputs (chapter 4.5). At the same time, the evaluation and the robustness tests show that the LLM sometimes fills gaps with plausible but weakly supported assumptions, especially for under-specified attributes and short inputs (validation 1 & 2 in chapter 5). This indicates that the concept is implemented, but uncertainty handling should be stricter (e.g., stronger abstention rules and clearer "unknown" defaults) to fully meet the intended requirement.

NFR1 – Reproducibility – partially fulfilled
The pipeline produces deterministic outputs for the clustering and the computed visualizations given fixed data and parameters (chapter 4). However, the LLM-based persona extraction exhibits result variability, especially under low-evidence conditions. This is a known limitation of probabilistic generation and suggests the need for stricter constraints and determinism controls if the system is used for decision-critical contexts.

NFR2 – Comprehensibility – largely fulfilled
The dashboard structure and progressive disclosure approach aim to reduce cognitive load and make results interpretable without requiring direct interaction with raw data. That said, interpretability still depends on how well personas avoid over-inference and how clearly evidence/coverage is communicated.

NFR3 – Modularity – fulfilled
The solution separates core concerns conceptually and in implementation: data preparation and features, clustering, persona generation, and visualization are treated as distinct stages (chapter 4). This supports future replacement or improvement of individual parts.

NFR4 – Scope compatibility – fulfilled
The work remains a research prototype with deliberately bounded scope, while still demonstrating an end-to-end workflow and evaluation (chapter 4).

Overall, the project meets the majority of its goals and core functional requirements, especially those related to end-to-end feasibility: clustering, persona artifact generation, interactive exploration, and time-window navigation. At the same time, the evaluation provides clear, actionable limitations: automated evolution signaling (FR4) and robust uncertainty/abstention handling (FR5 and NFR1) are the main gaps that must be addressed to move from a demonstrator to a reliable, decision-support tool. Nonetheless, the prototype remains a research demonstrator, which does not claim semantic correctness or generalizability.

6.3 Expected Next Steps

While the current system demonstrates end-to-end feasibility and addresses key research gaps, transferring it from an IP5 prototype into a production-ready product would require additional work in four main areas: data integration, model reliability, operationalization of evolution, and adoption in practice.

First, the prototype would need a robust data ingestion and governance layer. In a real product setting, behavioral events, user/session identifiers, and explicit feedback sources (e.g., interviews, surveys, in-app feedback, support tickets, app store reviews) must be collected

continuously, standardized, and stored with clear provenance. This includes defining event schemas, ensuring consistent feature computation over time, and implementing privacy-by-design.

Second, the LLM-based persona extraction must become more reliable and controllable. A product version should implement stricter abstention rules (e.g., "unknown unless explicitly stated" for sensitive or under-specified fields), stronger evidence gating, and deterministic generation settings where appropriate. Beyond prompt engineering, this likely requires adding validation constraints and an optional human-in-the-loop review workflow for approving persona updates before they are published to stakeholders.

Third, persona evolution should be operationalized beyond manual inspection. The next step would be to implement automated change detection mechanisms and convert them into actionable signals such as alerts, trend summaries, and "what changed and why" views. This would strengthen the system's ability to support continuous validation and evolution, which is central to the project vision.

In summary, the path to productization is clear: strengthen data pipelines and governance, increase LLM reliability through abstention and validation and automate evolution detection and reporting.

6.4 Personal Reflection

Looking back, this project was challenging in ways we did not fully anticipate at the beginning. The overall topic, data-driven personas, combined several areas that were largely new to us, including behavioral clustering (and clustering in general), feature engineering, LLM-based information extraction, and reading and synthesizing research papers. Especially in the first half of the project, this novelty created a constant feeling of uncertainty: we were learning concepts while simultaneously trying to think of a coherent solution and meet academic expectations. As a result, progress often felt non-linear. We would make steps forward, only to realize that our understanding of the problem space was still incomplete and needed to be revised.

One of the most difficult aspects was that, for a long time, it was not fully clear what the actual problem and the corresponding solution should be in the scope of IP5. Although the literature review helped us identify multiple gaps, we initially focused too strongly on the "Persona Evolution" gap and interpreted it as the main direction the prototype must cover. In hindsight, this led us into a partially wrong direction for too long and delayed the moment when we converged on a feasible MVP.

The implementation phase was also far from simple. Finding a dataset with all the necessary data we needed, integrating a clustering pipeline with a consistent feature schema, connecting it to an interactive dashboard, and then adding LLM-based extraction of user profiles and Proto-Personas required many iterations. We had to handle missing information, inconsistencies in outputs, and limitations of LLMs, especially when the input data was sparse.

Despite these difficulties, we take a lot away from this project. We learned how to approach an unfamiliar research domain, how to extract structure from a large body of literature, and how to

translate research gaps into implementable requirements. Technically, we gained hands-on experience in data processing, clustering, prompt design, and building an end-to-end prototype that combines quantitative and qualitative data. Overall, we look back on an intensive and demanding, but also highly formative project period, one that significantly expanded our skills and confidence in tackling complex, interdisciplinary problems.

6.5 Conclusion

This project explored how data-driven persona development can be made more transparent, maintainable, and useful over time by combining behavioral clustering with LLM-supported persona synthesis from explicit feedback. We delivered an end-to-end dashboard prototype that enables stakeholders to explore clusters, inspect Proto-Personas, and compare snapshots across time windows, thereby addressing key gaps such as static “flat file” personas and limited interpretability. The evaluation confirmed that the approach works particularly well when rich qualitative inputs (e.g., interviews) are available, while also revealing clear limitations for sparse inputs and the need for stronger uncertainty handling. Although the work was demanding, both conceptually and technically, it allowed us to build expertise in clustering, research-driven design, and LLM-based extraction in a real implementation context. Overall, the project resulted in a solid, extensible foundation for future improvements and it was an interesting process for both us.

List of Figures

List of Tables

List of Sources

[1] A. Baldi, "Data-driven persona development for a knowledge management system," 2021. Visited on 01.02.2026. Source: https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1614224&dswid=-1908

[2] J. Salminen, "A Survey of 15 Years of Data-Driven Persona Development," 2021. Visited on 01.02.2026. Source: https://www.tandfonline.com/doi/full/10.1080/10447318.2021.1908670#abstract

[3] Wikipedia, "Persona (user experience)," Wikipedia. Visited on 01.02.2026. Source: https://en.wikipedia.org/wiki/Persona_(user_experience)

[4] J. K. Daehee Park, "Constructing Data-Driven Personas through an Analysis of Mobile Application Store Data," 2022. Visited on 01.02.2026. Source: https://www.mdpi.com/2076-3417/12/6/2869

[5] K. B. R. Hang Guo, "Anthropological User Research: A Data-Driven Approach to Personas Development," 2015. Visited on 01.02.2026. Source: https://dl.acm.org/doi/abs/10.1145/2838739.2838816

[6] N. S. Nitish Patkar, "Data-Driven Persona Creation, Validation, and Evolution," 2023. Visited on 01.02.2026. Source: https://link.springer.com/chapter/10.1007/978-3-031-29786-1_18

[7] E. R. A. M. Marcelo M. Soares, Design, User Experience, and Usability, Springer, 2022. Visited on 01.02.2026. Source: https://link.springer.com/chapter/10.1007/978-3-031-05906-3_1

[8] J. Salminen, "Introduction to Data-Driven Personas," 2019. Visited on 01.02.2026. Source: https://persona.qcri.org/blog/benefits-of-data-driven-personas/

[9] J. S. F. A. S. M. H. T. S. S. a. B. J. J. D. Amin, "How Is Generative AI Used for Persona Development?: A Systematic Review of 52 Research Articles," 2025. Visited on 01.02.2026. Source: https://arxiv.org/pdf/2504.04927

[10] A. S. Djilali Idoughi, "Adding user experience into the interactive service design loop: a persona-based approach," 2011. Visited on 01.02.2026. Source: https://www.tandfonline.com/doi/abs/10.1080/0144929X.2011.563799

[11] P. Ebel, "Automotive UX design and data-driven development: Narrowing the gap to support practitioners," 2021. Visited on 06.02.2026. Source: https://www.sciencedirect.com/science/article/pii/S2590198221001603

[12] J. C. Quiñones-Gómez, "Data-Driven Design in the Design Process: A Systematic Literature Review on Challenges and Opportunities," 2023. Visited on 01.02.2026. Source: https://www.tandfonline.com/doi/full/10.1080/10447318.2024.2318060

[13] J. Salminen, "The Ethics of Data-Driven Personas," 2020. Visited on 01.02.2026. Source: https://www.bernardjjansen.com/uploads/2/4/1/8/24188166/3334480.3382790.pdf

[14] P. Xenopoulos, "ggViz: Accelerating Large-Scale Esports Game Analysis," 2022. Visited on 01.02.2026. Source: https://dl.acm.org/doi/abs/10.1145/3549501

[15] P. Laubheimer, "3 Persona Types: Lightweight, Qualitative, and Statistical," Nielsen Norman Group, 2020. Visited on 02.02.2026. Source: https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1614224&dswid=-1908

[16] T. Santonen, "Cocreating a Harmonized Living Lab for Big Data–Driven Hybrid Persona Development: Protocol for Cocreating, Testing, and Seeking Consensus," 2022. Visited on 02.02.2026. Source: https://www.researchprotocols.org/2022/1/e34567/

[17] J. Salminen, "A Template for Data-Driven Personas: Analyzing 31 Quantitatively Oriented Persona Profiles," 2020. Visited on 02.02.2026. Source: https://link.springer.com/chapter/10.1007/978-3-030-50020-7_8

[18] F. Pehar, "AI as Synthetic Users in Human-Centred Design: A Systematic Review," 2025. Visited on 02.02.2026. Source: https://ieeexplore.ieee.org/abstract/document/11131969

[19] A. Kasiri, "Analysis of Large Language Model in Creating User Personas: A Comparative Study Across Cultures," 2025. Visited on 02.02.2026. Source: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5377696

[20] J. Salminen, "Are data-driven personas considered harmful?: Diversifying user understandings with more than algorithms," 2021. Visited on 02.02.2026. Source: https://search.informit.org/doi/abs/10.3316/informit.352977339951659

[21] J. Newhook, "How the Best UX Design Teams Integrate Personas into Agile Workflows," 2025. Visited on 02.02.2026. Source: https://www.interaction-design.org/literature/article/how-the-best-ux-design-teams-integrate-personas-into-agile-workflows

[22] S. Lopez, "Data-driven Buyer Personas: What They Are and How to Build them," 2024. Visited on 02.02.2026. Source: https://www.magnolia-cms.com/blog/data-driven-buyer-personas.html

[23] H. Liang, "Data-Driven User Personas in Requirement Engineering with NLP and Behavior Analysis," 2024. Visited on 03.02.2026. Source: https://joiv.org/index.php/joiv/article/view/3625

[24] R. I. Rezvi, "Data-Driven Persona: How Business Intelligence Analytics Enhance Customer Experience for US Small and Medium Enterprises (SMEs)," 2025. Visited on 03.02.2026. Source: https://al-kindipublishers.org/index.php/jcsts/article/view/10767

[25] X. Zhang, "Data-driven Personas: Constructing Archetypal Users with Clickstreams and User Telemetry," 2016. Visited on 03.02.2026. Source: https://dl.acm.org/doi/abs/10.1145/2858036.2858523

[26] A. Mertes, "User profiles and personas of electronic participation in public discourse," 2025. Visited on 05.02.2026. Source: https://www.sciencedirect.com/science/article/pii/S2772503025000556

[27] L. N. G. Koralla, "Hyper-personalization: Transforming digital experiences through advanced data analytics and AI," 2025. Visited on 05.02.2026. Source: https://doi.org/10.30574/wjaets.2025.15.1.0219

[28] Y. Putri, "Designing UI/UX on Adaptive Skills Learning Application for Autistic Children Using Design Thinking Method and Applied Behavior Analysis Theory," 2024. Visited on 02.02.2026. Source: https://www.researchgate.net/profile/Yusnita-Putri/publication/

[29] Y. Shi, "You Are What You Bought: Generating Customer Personas for E-commerce Applications," 2025. Visited on 02.02.2026. Source: https://dl.acm.org/doi/abs/10.1145/3726302.3730118

[30] M. Rizwan, "PersonaBOT: Bringing Customer Personas to Life with LLMs and RAG," 2025. Visited on 02.02.2026. Source: https://arxiv.org/abs/2505.17156

[31] Z. Wang, "OPeRA: A Dataset of Observation, Persona, Rationale, and Action for Evaluating LLMs on Human Online Shopping Behavior Simulation," 2025. Visited on 02.02.2026. Source: http://arxiv.org/abs/2506.05606

[32] z. wang, "Chrome Web Store," [Online]. Available: https://chromewebstore.google.com/detail/shoppingflow-research-too/kakkajiejbidhldoklefcgjgjamidpcb. Visited on 03.02.2026. Source: https://chromewebstore.google.com/detail/shoppingflow-research-too/kakkajiejbidhldoklefcgjgjamidpcb

[33] NEU-HAI/OPeRA, "HuggingFace," [Online]. Available: https://huggingface.co/datasets/NEU-HAI/OPeRA. [Accessed 03 02 2026]. Visited on 03.02.2026. Source: https://huggingface.co/datasets/NEU-HAI/OPeRA

[34] G. N. Rodrigues, "NLP-driven customer segmentation: A comprehensive review of methods and applications in personalized marketing," 2025. Visited on 01.02.2026. Source: https://www.sciencedirect.com/science/article/pii/S2666764925000463

During the research process, we have read and investigated multiple papers, websites and articles. These are all listed in our Excel-Sheet, which we hereby also put in the list of sources:

Excel-Sheet: DDPD_Research-Sources-Log_withNotes.xlsx

Summary of Linked Artifacts

The following table lists all the artifacts we created. These artifacts are mentioned several times throughout the whole project and were needed and created on the way to our final product.

Artifact:	Description:	Link:
GitHub-Environment	Used before we have moved the whole project to GitLab. The reason we have started with GitHub was that we had already experience with GitHub-Pages and we wanted to see quick results as fast as possible while we have started with coding.	Link
Python-Notebook (Persona-Evolution-2.ipynb)	Used for getting into the clustering algorithms and to understand how the dataset behaves with specific clustering algorithms. The whole Python-Notebook was needed to preprocess our dataset.	Link
Excel-Sheet (DDPD_Research-Sources-Log_withNotes.xlsx)	Used for the whole research process to get into the topic and to find the gap which our solution of IP5 should fill out.	Link
GitLab-Environment	Codebase of our whole project is hosted on GitLab of the FHNW.	Link
Proto-Persona Evolution Dashboard (GitLab-Page)	Our implemented solution hosted on GitLab-Pages	Link

List of Tools/Aids

Throughout our work, we used the following tools and aids (the corresponding tools for the project code are also listed here):

Tool/Aid:	Used for what:	Affected areas:
Word	Written documentation, protocols, notes	Written documentation
PowerPoint	Presentations	Presentations
Excel	List and documentation of research papers	Research part and written documentation
Visual Studio Code	Project code	Project code
IntelliJ IDEA	Project code	Project code
Antigravity	Project code	Project code
Google Colab	Python-Notebook, project code	Project code
GitHub	Deployment of project code	Project code
GitLab	Deployment of project code	Project code
DeepL	Translate some paragraphs from German to English for the written documentation, translate explanations on the dashboard	Whole project
ChatGPT	Research, inspiration, coding questions, code documentation, language enhancement, generating code, correction of code, creation and correction of paragraphs in the written documentation	Whole project
Google Gemini	Research, inspiration, coding questions, code documentation, generating code, correction of code	Project code
NotebookLM	Research, inspiration, language enhancement, creation and correction of paragraphs in the written documentation	Research part and written documentation
Ollama Model: Gemma3	Project code (generates json-files)	Project code
study-design.md (document from Mr. Patkar)	Inspiration and explanation for clustering process.	Project code

We hereby confirm that we have used AI/LLMs (ChatGPT, Google Gemini, Antigravity, NotebookLM, DeepL and Ollama) for our work.

Appendix

Because there were some artifacts for which it is not possible to attach them directly to the document of this documentation, we have sent a ZIP-Folder with these artifacts together with our submission via E-Mail.

The Appendix-ZIP-Folder contains the following artifacts:

- Comment1.txt
- Comment2.txt
- Comment3.txt
- DDPD_Research-Sources-Log_withNotes.xlsx
- Interview1.txt
- Interview2.txt
- Persona_Evolution_2.ipynb
- ProtoPersonaUP12.json
- ProtoPersonaUPC123.json
- Study-design.md
- User-Profile1.json
- User-Profile2.json
- User-Profile-C1.json
- User-Profile-C2.json
- User-Profile-C3.json
- evaluate_extraction.py
- evaluation/evaluation_report.md
- evaluation/evaluation_results.json
- templates/evaluation_report.txt

Fachhochschule
Nordwestschweiz

Windisch, 06.02.26

Information on the project procedure & project agreement IP5
Data-Driven Persona Development

Supervisor: Nitish Patkar
Norbert Seyff

Client: -

Project duration: 15.09.2025 until 06.02.2026

Task

1. Familiarization

1.1 Expectations for the project process

Dates
Fix appointments early, i.e., reviews with the customer and about every 2-3 weeks a meeting appointment with your supervisors. Clarify any absences right at the start of the project.

Meetings
Meetings are basically intended to discuss the current status of the project, clarify questions, discuss ideas and plan the next steps.
Send a list of agenda items and all other necessary documents to the supervisors. At the beginning of each project meeting, explain the current status of the project, the progress and problems as well as the planned steps.
You can use the meetings by arrangement and, if necessary, also for specific questions (e.B microteaching, brainstorming, presentation of results or mentoring). However, come to a meeting with as specific questions as possible.
Please record the discussed contents and decisions in a timely manner.

1.2 Specifications for the agreement

As a first task in your work you have to complete this agreement (cf. point 3). A first version should be produced by 2-4 weeks (BB 4-6 weeks) after kick-off. For projects that re-quire technical analysis, it may be useful to carry out a first implementation iteration before the sub-mission of the project agreement. Please complete the following items:

Initial situation
Formulate the project and the initial situation in your own words.

Project vision
Describe which goals and results are to be achieved with the project. The vision serves to derive quality criteria.

Project specific issues
In addition to the general questions, formulate 2-3 project-specific questions. These serve as a basis for scientifically structured research and the derivation of suitable solutions.

Examples of questions and solutions:

Institute for Interactive technologies Bahnhofstrasse 6 5210 Windisch www.fhnw.ch/iit

Fachhochschule
Nordwestschweiz

- Which approaches do you use to reach the defined target group?
  Solution approach: Development of concepts for user-centered approaches and implementation of the user interface of the application, e.g., in the form of storyboards with a continuous user story or GUI prototypes.
- With which technical concept do you achieve the desired solution?
  Solution approach: Technology evaluation, development of technical solution concept (PoC), definition of subsystem decomposition, architectural style and technologies.
- Which interaction concepts, interface designs and visual languages are suitable for your approach?
  Solution approach: Development of interaction concepts and graphically carefully designed, clearly structured imagery for interface design, which meet the requirements of an innovative user experience.
- With which technical implementation do you meet the requirements for functionality, usability, reliability, efficiency and maintainability?
  Solution approach: Implementation of a executable application for a previously evaluated setup and defined usage scenario based on suitable technologies and frameworks
- Correctness, usability and reliability are central to the successful introduction of the software. How can you ensure and test them?
  Solution approach: In-depth testing of correctness, usability and reliability, documentation of Test results, demonstration of the fulfillment of the requirements by means of live test.

Methodology
Describe how the goals are achieved. Which methodologies do you use for this (e.g. Scrum, Agile, scientific approach, etc.).

Planning
Create an initial project schedule. Define work packages and their deliverables.

Risk Assessment
Identify and evaluate risks within the project and develop strategies for dealing with them.

2. Documentation

2.1 Written documentation (Thesis Rapport)

Document in writing and electronically your approach, the theoretical background, the application of methods and concepts, the implementations and test results. Also check the planned with the actual schedule, the achievement of goals and reflect on experiences.
Be sure to strictly separate personal comments from facts. **The main part of the documentation is completely fact-based**. This means that no sentences of the kind "Then we had the problem x and tried to solve it with y" are allowed to occur. But if such a problem x really exists and not only you did not get to the edge with it, then you should write: "Tests z have clearly shown that a problem x exists. Possible approaches to solve problem x are a, b and c. We chose variant c for reasons e and f." Only in an extra section can you formulate your personal impressions, experiences, problems and the like.
It is also important that a good documentation must still be read after many years and that it gives the reader a well-rounded picture, even if he was not directly involved in the work. Please also attach great importance to linguistic quality.
The target audience of this documentation are the supervisors, the experts, the client and future students who want to continue working in this area.
The documentation is created during the course of the project. For the second coaching meeting, a table of contents of the report should be prepared so that it can be discussed with the supervisors.
The parts for research and analysis are to be presented after the first third of the project.

On the web portal of the FHNW you create a project presentation (web summary). For bachelor theses in the spring semester, you will also create a poster for the exhibition. Both artifacts must be discussed with the supervisors prior to publication.

Fachhochschule
Nordwestschweiz

The following information must be mentioned on all publications:

- Logo FHNW
- Semester project IP5 or Bachelor thesis (IP6)
- Project name
- Spring- or Autumn Semester 202x, Degree Program Computer Science (Profiling iCompetence), University of Applied Sciences and Arts Northwestern Switzerland
- Submitted by: Name of Students
- Submitted to: Name of Supervisor
- Client: Company / Institution
- Date

Further information on writing reports can also be found on the Information Literacy Platform

2.2 Presentations

Presentations take place in consultation with the supervisors and the client. The expert will also be present when defending your bachelor's thesis.

On the one hand, presentations provide an overview of the entire project and the results achieved and deepen one or two important interesting questions. Also part of the presentation is a concise demonstration of how to use your software. With the audience, you can expect a technically experienced professional audience. Schedule 30' for the presentation and demonstration and reserve 30' for questions and discussion.

2.3 Publication of the project results

If the work or parts of the work are published, all names of the project participants (students, supervisors, clients) as well as the name of the institution (FHNW) must be mentioned. Before each publication, supervisors and clients must be asked for their consent in advance.

2.4 Protocols

Protocols are an important part of the documentation. Professionally managed protocols contain the following points:

- Date, Space, Time, Participants, Excused
- Agenda
- Project status (possibly with screenshots, sketches, etc.; Status according to planning)
- Content (fact-based, thematically structured and comprehensible in terms of content; Decisions are recorded)
- Open questions
- Next steps; Appointments & tasks (who, what & until when)

2.5 Document repository

Set up access to your document storage for the maintainers. If there are no compelling reasons against it, use the Gitlab infrastructure of the FHNW[1].
Also, use this document cabinet to store additional documentation, such as how to run your code. Make sure that an adequate commit history is visible to the caregivers.

2.6 Submission

The project submission includes (unless otherwise defined with the project manager) the following artifacts:

- Written documentation (Thesis Rapport)
- Project agreement (on the shelf as an appendix in the thesis)

[1] https://gitlab.fhnw.ch/

Fachhochschule
Nordwestschweiz

- Codebase (documented & with readme to explain the setup), hosted on GitLab of the FHNW (https://gitlab.fhnw.ch/iit-projektschiene/[semester]/[project]) and as a ZIP archive
- Link to the project appearance on the FHNW web portal
- other artifacts, if available (screencast recommended, ...)

Fachhochschule
Nordwestschweiz

3. Project-specific agreement

3.1 Initial position

Personas are a widely used technique in software development to represent typical user groups and to support decisions in requirements engineering, UX design and product strategy. A persona aggregates common behavior, goals and motivations of a user segment into a concise, human-readable profile that helps teams keep users in mind during design and implementation. In practice, however, personas are often created early in a project using qualitative and ethnographic methods (for example interviews, field studies or workshops). While these methods can produce rich insights, they are time-consuming and expensive, which often limits how thoroughly personas are developed. In addition, personas frequently become static artifacts: once created, they are rarely validated against real usage data and seldom updated, even though user groups, their behavior and their needs evolve over the lifecycle of a system. The lack of systematic validation after initial creation further increases the risk that teams base decisions on assumptions rather than on actual user evidence.
Research in “Data-Driven Persona Development” has shown that large-scale user data can be used to create personas more efficiently, for example by analyzing interaction logs (clickstreams) and applying clustering methods such as k-means to identify behavior-based user groups. Nevertheless, current approaches primarily emphasize persona creation and typically assume that sufficient operational data is already available. They do not adequately address the continuous validation and evolution of personas as a regular activity during system operation. This creates a gap between how personas are intended to support modern, iterative software development and how they are actually maintained (or not maintained) in practice.
Against this background, the project starts from the premise that personas should be treated as “living models” that can be created, checked and updated continuously using data gathered during app usage. The project is positioned in the area of data-driven requirements engineering and targets a practical tool that supports teams in understanding “who their users are”, “how they behave” and “how these behavior-based user groups change over time”.

3.2 Project vision

The project’s vision is to make personas data-grounded, continuously maintainable and transparent, so that they remain useful beyond the “kick-off persona workshop” and can be refined throughout a software system’s lifetime. Instead of treating personas as static artifacts, the project aims to demonstrate that personas can become living models that are derived from operational data and updated when the underlying user reality changes. This directly addresses the gap we have found thanks to our research we did about “Data-Driven Persona Development”: Existing work in “Data-Driven Persona Development” often focuses on creation, while validation and evolution over time are rarely supported and seldom practiced.
Concretely, the project targets an implemented solution that answers the following core claim: “Personas can be generated from data”. This includes two complementary signal types:
1. Behavioral/Monitoring data that reflects what users do (for example usage intensity and session characteristics)
2. Explicit feedback that reflects what users say (for example stated preferences, needs and complaints)
The intended outcome is a mechanism that identifies distinct user groups through clustering and generates a persona profile per group, including a statement of how strongly each attribute of the created persona is supported by data.
The implemented solution is the envisioned front-end representation of this mechanism: an interactive dashboard that visualizes behavior-based clusters, lets stakeholders inspect cluster-specific personas and makes changes visible across multiple time windows. This temporal perspective is essential: the project aims to detect and communicate persona drift and evolution (for example shifting cluster distributions, emerging new groups or changing dominant behaviors) rather than representing only a single static segmentation snapshot.
By the end of the project, the expected results are:
- A data approach that combines behavioral data, timestamps and explicit feedback into features

Fachhochschule
Nordwestschweiz

suitable for clustering and persona generation.
- An interactive dashboard prototype that supports exploration, comparison and time-based inspection of clusters and their corresponding persona and evolution.

A persona generation mechanism that produces a cluster-level persona with inferred attributes from explicit feedback (potentially supported by an LLM-based extraction step) and transparent indication of confidence and data coverage ("on how much data is this persona based?").
The project vision implies the following goals to assess success:
- The solution works with a dataset.
- Persona attributes and claims can be traced to specific behavioral patterns and/or explicit feedback evidence.
- The solution can represent changes of clusters and personas across time windows.
- The solution fits the IP5 scope, so that it emphasizes core components (dashboard, clustering and personas) and provides a solid foundation for possible later expansion.

3.3 Questions

A. Translation of clusters into Personas
How successful is the conceptual and technical translation of behavior-based clusters into understandable personas (including structure, attributes, narratives), so that the results do not remain abstract segments?

B. Robustness and limitations of LLM extraction
Under what conditions does LLM-supported user profile and persona generation deliver consistent, data-plausible results and where are the limitations (over-inference, normalization, variance)?

C. Usability and cognitive comprehensibility of the dashboard
To what extent does the dashboard help stakeholders move from a cluster overview to interpretable personas (overview → drill-down → evidence) without creating a high cognitive load?

In addition to the project-specific questions, the following generic questions will be considered in the implementation of their work:

D. Identification of suitable scenarios and user interface prototyping: Which approaches do you use to reach the defined target group?

E. Technical concept: With which technical concept do you achieve the desired solution?

F. User Interface Design: Which interaction concepts, interface designs and visual languages are suitable for your approach?

G. Implementation: With which technical implementation do you meet the requirements for functionality, usability, reliability, efficiency and maintainability?

H. Testing: Correctness, usability and reliability are central to the successful introduction of the software. How can you ensure and test them?

3.4 Methodology

To achieve the project goals, we follow a combined scientific engineering approach that integrates structured research and evaluation of state-of-the-art methods, iterative software engineering for the prototype and user-centered design principles for a dashboard that supports stakeholders. The methodology is organized along four complementary workstreams:

Fachhochschule
Nordwestschweiz

1. Scientific work (state of the art)
   The project is grounded in a targeted literature review on “Data-Driven Persona Development”, clustering-based segmentation and persona validation. Based on this research, we derive assumptions and select methods that are suitable for our dataset and the scope. The scientific workflow starts with researching and getting a strong base knowledge about the current situation of the field and then follows an iterative cycle of problem framing, method selection, prototyping and refinement.
2. Requirements engineering (goal-oriented)
   Requirements are derived from the initial project mandate, stakeholder feedback from supervisors and constraints arising from available data. Requirements are continuously refined during development and linked to implemented features of the implemented solution to maintain traceability between problem, solution and evaluation.
3. User-centered design (Usability and cognitive clarity)
   Since the final artifact is a decision-support dashboard, we apply user-centered design methods focused on stakeholder interpretability. We have the main users of the dashboard in our mind (developers, product owners, UX designers, requirements engineers) and try to design the user interface based on their typical problems/questions (like for example “Which personas exist?” or “How do they change over time?”).
4. Software engineering (iterative prototyping)
   Implementation follows an iterative and incremental process inspired by agile development. Features are developed in small increments, integrated as early as possible and refined based on the supervisor’s feedback. The solution is decomposed into logical components: data preparation, clustering generation, persona generation and dashboard visualization.

Overall, this combined methodology ensures that the project remains scientifically grounded, traceable to stakeholder goals and practically demonstrable through an executable prototype that refelcts the project vision.

3.5 Planning

The project is planned iteratively and is updated continuously as a living document. Effort estimates are rough and refer to total team effort,

WP 1: Project Alignment and Kick-off	
Goals	Establish a shared understanding of scope, vision, roles and the initial situation.
Activities	Kick-off meeting, reading/understanding the initial paper by the supervisors and meeting cadence.
Deliverables	Meeting notes from Kick-off and collaboration setup
Start	22.09.2025
End	07.10.2025
Effort	Approx. 15 h

WP 2: State-of-the-art research	
Goals	Build a structured understanding of the research landscape on DDPD and the persona lifecycle (creation/validation/evolution).
Activities	Systematic search, paper screening, structured extraction into the research spreadsheet, initial synthesis.
Deliverables	Literature summary, research log (Excel-Sheet), refined problem statement and first proposal of project specific questions for the Project Agreement document.
Start	07.10.2025
End	21.10.2025

Fachhochschule
Nordwestschweiz

Effort	Approx. 30 h

WP 3: Gap Analysis and Requirements Derivation	
Goals	Derive research/practice gaps from literature and translate them into design goals and requirements.
Activities	Consolidate recurring gaps (e.g., temporality, evaluation, interpretability/tooling), define requirements and success criteria, refine research questions and scope.
Deliverables	Number based facts/graphics to back up our gaps we have found in the papers, requirements/design goals and updated project focus.
Start	21.10.2025
End	12.11.2025
Effort	Approx. 40 h

WP 4: Clustering	
Goals	Understand how clustering is relevant for our project and how it works.
Activities	Workshop with Mr. Patkar about clustering algorithms and possible directions for our implemented solution and in general read into clustering algorithms and how these work.
Deliverables	-
Start	12.11.2025
End	17.11.2025
Effort	Approx. 20 h

WP 5: Data selection + Prototype	
Goals	Obtain a usable dataset and bring a clear idea of what the implemented solution will be in the end.
Activities	Dataset selection, schema analysis (explicit feedback, implicit feedback, demographic data, clickstream data, time attribute), start building a prototype.
Deliverables	Dataset and a prototype
Start	17.11.2025
End	27.11.2025
Effort	Approx. 40 h

WP 6: Improve the Prototype	
Goals	Implement an interactive dashboard that supports overview and already loads a dataset
Activities	Dataset selection (again), UI concept
Deliverables	Prototype with data loading from the selected dataset
Start	27.11.2025
End	15.12.2025
Effort	Approx. 40 h

WP 7: Switch the dataset again

Fachhochschule
Nordwestschweiz

Goals	Find a new dataset which fulfills the schema (explicit feedback, implicit feedback, demographic data, clickstream data, time attribute).
Activities	Search for a new dataset, test it with a Python-Notebook and show with the notebook that the dataset is applicable for our implemented solution.
Deliverables	Python-Notebook
Start	15.12.2025
End	25.12.2025
Effort	Approx. 30 h

WP 8: Exploratory Clustering Pipeline (Python-Notebook)	
Goals	Build an end-to-end PoC from behavioral/session features to interpretable clusters and initial visual exploration.
Activities	Feature engineering, clustering (e.g., k-means), cluster profiling, first comparative visualizations, and create radar chart visualizations to compare clusters
Deliverables	Exploratory notebook with working pipeline: Clickstream Data → Session Features → k-means Clustering → Persona Archetypes → Radar Charts / Evolution Visualizations
Start	25.12.2025
End	04.01.2026
Effort	Approx. 40 h

WP 9: Persona Generation from Explicit Feedback	
Goals	Create structured user profiles and proto-personas from explicit feedback sources (especially interviews).
Activities	System prompts, JSON schema definition, extraction of user profiles, aggregation into proto-personas, first transparency indicators.
Deliverables	Persona generation pipeline (user profiles + proto-persona) + documented prompts/schemas.
Start	04.01.2026
End	13.01.2026
Effort	Approx. 40h

WP 10: Dashboard Prototype	
Goals	Deliver an interactive dashboard that integrates clustering, personas, and temporal comparison.
Activities	UI concept (overview → drill-down → evidence), integration of artifacts, time-window comparison, visual analytics views.
Deliverables	Running dashboard prototype (deployed link) with core analysis views
Start	13.01.2026
End	27.01.2026
Effort	Approx. 20h

WP 11: Evaluation and Validation

Fachhochschule
Nordwestschweiz

Goals	Evaluate quality, strengths/limitations, and practical usefulness of the prototype using defined criteria.
Activities	Validation tests (e.g., prompt robustness, interview vs. comment inputs), evidence/traceability checks, critical analysis.
Deliverables	Evaluation results, interpretation (strengths, weaknesses, limitations, implications).
Start	27.01.2026
End	06.02.2026
Effort	Approx. 20h

WP 12: Documentation and Project Close-out (in-parallel to WP 11)	
Goals	Finalize the written thesis/report with coherent narrative, evidence, and reflection.
Activities	Writing and revision, packaging of artifacts and documentation.
Deliverables	Final report + appendix materials, cleaned repository + README, final deployed prototype.
Start	27.01.2026
End	06.02.2026
Effort	Approx. 30h

Project Milestones:

MS 1: Research Baseline and Project Focus Established
Covered by: WP1 – WP3
Outcome: State-of-the-art captured, gaps consolidated, requirements/design goals defined, scope and research questions refined.
Evidence/Artifacts: Research log and stable thesis structure

MS 2: Data and Clustering PoC
Covered by: WP4 – WP8
Outcome: Dataset deemed fit-for-purpose, feature schema defined, clustering pipeline produces interpretable clusters and initial comparisons.
Evidence/Artifacts: Feature schema and working Python-Notebook

MS 3: Persona Generation Capability
Covered by: WP9 – WP10
Outcome: System generates structured user profiles from explicit feedback and aggregates them into Proto-Personas, including initial transparency concepts (evidence/coverage).
Evidence/Artifacts: JSON outputs (user profiles and Proto-Personas), documented prompts/schemas.

MS 4: End-to-end Dashboard and Evaluation
Covered by: WP10 – WP12
Outcome: Deployed dashboard prototype integrating clusters, personas, and time windows, evaluation/validation completed, documentation finalized with critical reflection and future work.
Evidence/Artifacts: Deployed dashboard link and final written documentation / Thesis Rapport.

3.6 Risk Assessment

The project combines data-driven analysis, persona generation logic and an interactive dashboard. This creates risks in the areas of data availability/quality, methodological validity, implementation feasibility and evaluation. The following risks are identified together with migration strategies.

R1: Dataset suitability and missing longitudinal information
Risk: A dataset may not contain enough repeated user behavior over time or may have too few samples per period, which limits persona evolution analysis and cluster robustness.
Mitigation: Early dataset screening with clear criteria (sessions per user, time coverage, availability of explicit feedback). Check the option to generate further user behavior / samples with a LLM.

R2: Data quality and inconsistent schemas
Risk: Missing values, noisy clickstream logs or inconsistent fields can lead to unreliable features and unstable clustering.
Mitigation: Establish a reproducible preprocessing pipeline (cleaning, normalization, documented assumptions).

R3: Clustering instability and weak interpretability
Risk: K-means (and clustering in general) can produce unstable results depending on initialization, feature scaling or parameter choice. Clusters may also be hard to interpret or not meaningful.
Mitigation: Use multiple runs with fixed random seeds and compare cluster stability across runs. If clusters are not meaningful, reduce complexity (fewer features, fewer clusters) and prioritize interpretability over "optimal" numeric scores. If there is enough time, try to do the clustering with a different algorithm.

R4: Persona attribute inference may be speculative
Risk: Inferring attributes such as age, gender, education or preferences from explicit feedback can be uncertain, may not be possible for all users and can easily become "overconfident" or biased.
Mitigation: Systematically define an attribute catalog with feasibility per attribute (which attribute would be direct available?). Report attributes as probabilities/confidence with clear evidence and coverage ("based on X users").

R5: LLM-based extraction reliability
Risk: If an LLM is used to extract persona attributes from text feedback, outputs may be inconsistent, hard to reproduce or hallucinated.
Mitigation: Use constrained and structured prompts, structured output formats and rule-based validation. Work with JSON-Files as input-data instead of raw text.

R6: Scope/time risk in the final phase
Risk: The project covers data pipelines, clustering, persona generation, dashboard development and in the optimal scenario evaluation. This breadth can lead to time pressure and incomplete integration.
Mitigation: Prioritize the core claim (user-behavior clustering and "generating personas from data") and ensure a working dashboard with at least one dataset first. Freeze core features early, then iterate on improvements.

R7: Evaluation limitations
Risk: Proving correctness of personas is difficult and evaluation may become subjective or too weak.
Mitigation: Evaluate along clearly defined criteria (traceability, reproducibility and data coverage).

Fachhochschule
Nordwestschweiz

4. Final provisions
The undersigned acknowledge that they have read and understood the text and undertake to comply with the points listed and the general duty of care with their signature.

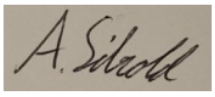
Windisch, ...24.01.2026.....

Supervisors

Nitish Patkar .....

Norbert Seyff .....

Student

Andrin Sibold .....  .....

Miro De Nardo .....

Institute for Interactive technologies Bahnhofstrasse 6 5210 Windisch www.fhnw.ch/iit

Fachhochschule Nordwestschweiz

Design UX | Data Science

25HS_IIT26: Automated Creation, Validation, and Evolution of Personas

			Priority 1	Priority 2
Advisor:	Norbert Seyff	**Work scope:**	P5 oder P6	---
	Nitish Patkar	**Team size:**	2er Team	---
Languages:	German or English			
Study course:	Computer Science			

Initial situation

Personas have long helped represent user groups in software development, but traditional methods like ethnographic studies are time-consuming, costly, and often result in static artifacts. Once created, they rarely evolve with real user behavior or feedback.

Emerging research shows that user interaction data and feedback can drive automated, continuously updated personas. Early findings from interview studies highlight the potential for smarter, evolving persona design. This project aims to transform these insights into a practical tool for teams seeking personas that reflect actual user behavior throughout a project.

Objective

The goal is to build a modern software tool that brings persona creation into the era of automation, agility, and data. It will:

- Generate personas using user feedback, behavior, and interaction data.
- Continuously check personas against live data to ensure relevance.
- Adapt personas over time as new data becomes available.
- Offer a clear, actionable dashboard for designers, developers, and stakeholders.

IP5 Focus:

Develop and test core components using synthetic or existing datasets. Emphasis will be on clustering behavior patterns, validating with mock data, and designing the persona dashboard.

IP6 Focus:

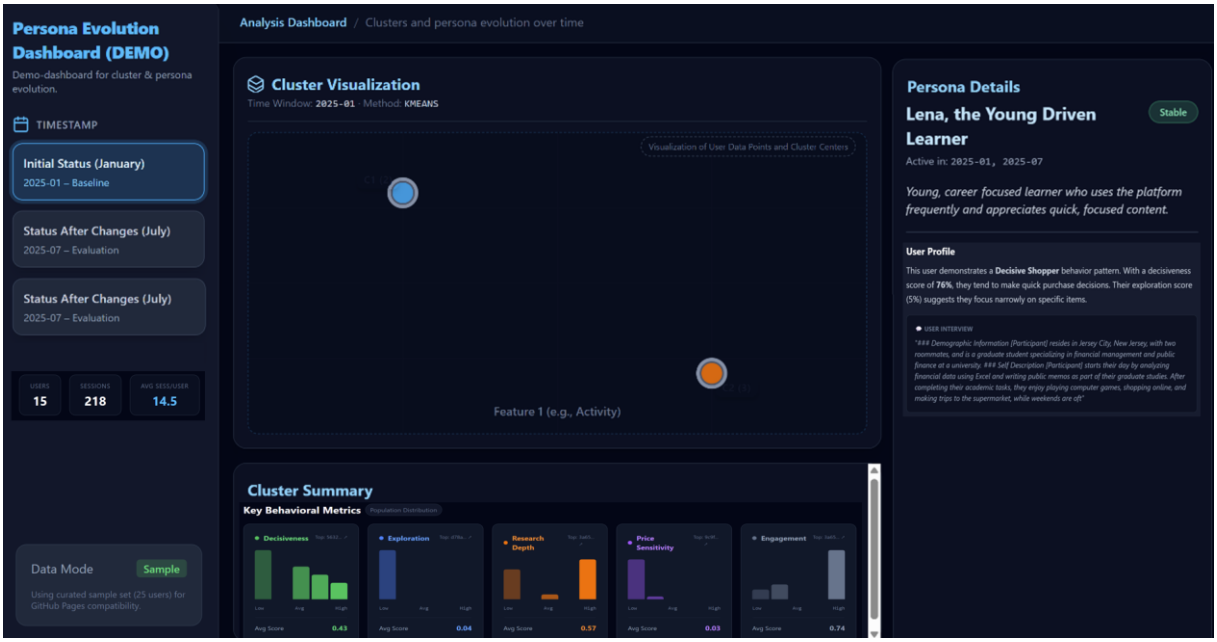
Expand through practitioner interviews, platform integration, and user studies. Explore sentiment analysis, long-term adaptation, and enhanced visual storytelling for evolving personas.

Problem statement

This project will tackle the following key questions:

1. What kinds of data (e.g., behavioral logs, feature usage, survey responses) are best suited for generating and evolving personas?
2. How can existing clustering, sentiment analysis, or recommendation algorithms be adapted to support real-time persona creation and validation?
3. How can the system be implemented to automate data collection, processing, and persona updates while remaining modular and scalable?
4. What visual formats best represent evolving persona traits, and how can changes be communicated effectively to various stakeholders?
5. How can evolving personas be seamlessly embedded into software development workflows (e.g., in agile retrospectives, UX reviews, or product strategy sessions)?

Prototype:


Persona Evolution Dashboard (DEMO)
Demo-dashboard for cluster & persona evolution.
TIMESTAMP
Initial Status (January)
2025-01 – Baseline
Status After Changes (July)
2025-07 – Evaluation
Status After Changes (July)
2025-07 – Evaluation
USERS 15
SESSIONS 218
AVG SESS/USER 14.5
Data Mode
Sample
Using curated sample set (25 users) for GitHub Pages compatibility.
Analysis Dashboard / Clusters and persona evolution over time
Cluster Visualization
Time Window: 2025-01 · Method: KMEANS
Visualization of User Data Points and Cluster Centers
Feature 1 (e.g., Activity)
Cluster Summary
Key Behavioral Metrics
Decisiveness
Exploration
Research Depth
Price Sensitivity
Engagement
Persona Details
Lena, the Young Driven Learner
Stable
Active in: 2025-01, 2025-07
Young, career focused learner who uses the platform frequently and appreciates quick, focused content.
User Profile
This user demonstrates a Decisive Shopper behavior pattern. With a decisiveness score of 76%, they tend to make quick purchase decisions. Their exploration score (5%) suggests they focus narrowly on specific items.
USER INTERVIEW