

# Eligibility Checker

## Combating Non-Take-Up of Social Benefits by Simplifying Access

David Heckmann, Luis Carilla, Shu-Xiang Yang, Leon Froschauer, and Linus Höhne

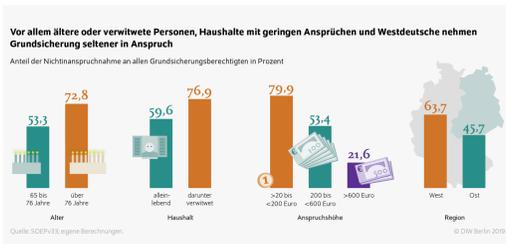
### 1. Problem Definition

#### Long List of Potentially Relevant Services

Cities currently attempt to support citizens by displaying all potentially relevant services (see Figure 2). These lists can be overwhelming, since the citizen has to check for which services he is eligible and for which not.

#### Citizens do not claim services

The uncertainty of eligibility combined with the frustration of manual verification creates a psychological barrier and leads to resignation. The result is a high *Non-Take-Up rate*: governmental benefits do exist, but do not reach the people.



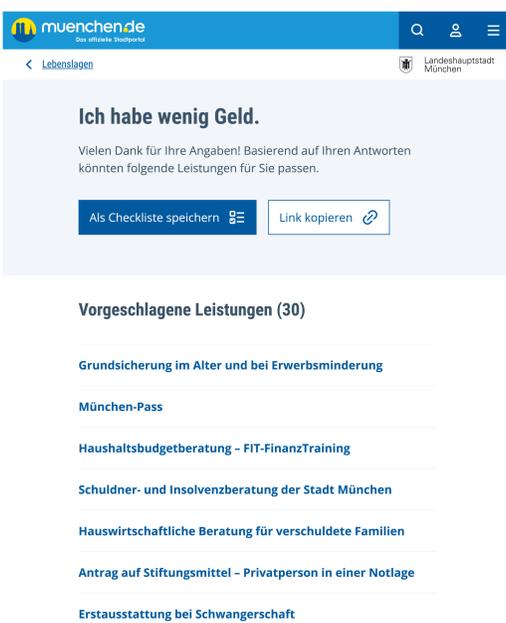
**Figure 1** High non-take-up of basic income support  
Quelle: Hermann Buslei, Johannes Geyer, Peter Haan, Michelle Harnisch: DIW Wochenbericht 49/2019, S. 909-917.

#### The Hidden Complexity

The administrative challenge is not just displaying information, but managing the vast web of dependencies (e.g., a student needs different fields than a pensioner). Currently, this complexity is exposed to the user, who must navigate unrelated questions and redundant data entry ("Input without Memory").

### 2. Status Quo

- Static Information:** The current "Lebenslagen"-system displays a static list of 30 potential services (see Figure 2).
- No Eligibility Check:** This results in information overload and false positives, since no eligibility is checked.



**Figure 2** Status Quo: Static list of suggestions

### 3. Try it out!



### 4. The Conditional Engine

We transformed static *Förderfunke SHACL shapes* into executable code. This engine calculates missing data points in real-time to dynamically render the form.

- Hides Complexity:** Irrelevant questions are never shown; the user only sees input fields actually required based on previous entries.
- Smart Feedback:** The system provides custom explanation messages if a user is found ineligible for a service.
- Scalable:** Manages multiple funding services in a single flow, adapting instantly to new inputs.

#### Mögliche Leistungen (1)

Basierend auf Ihren Angaben könnten folgende Leistungen für Sie in Frage kommen:

Nur berechtigte anzeigen

#### Kindergeld

Sie haben 1 Kind angegeben. Sie könnten für Kindergeld berechtigt sein.

Zum Antrag →

#### Wohngeld

Das Einkommen liegt über der Grenze.

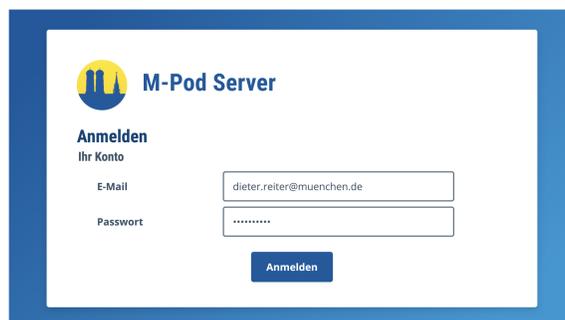
Nicht berechtigt

**Figure 3** Detailed User Feedback

### 5. Solid Pod Integration

We built a *Solid Pod* integration to enable citizens to securely persist their data:

- Decentralized:** A Solid Pod is a decentralized data storage unit that decouples data from applications, serving as a secure personal server for a digital identity.
- Across services:** The city could extend this integration to all services, leveraging interoperability to access data stored by other entities (e.g., companies). Ideally, this establishes a pattern applicable beyond the local level, allowing municipalities, states, and the federal government to automate form-filling across Germany without redundancy.
- Privacy:** The architecture ensures digital sovereignty, as the user retains full ownership and control over access rights:
  - The user decides *who hosts* the Pod
  - The city only queries data; it does not store it.

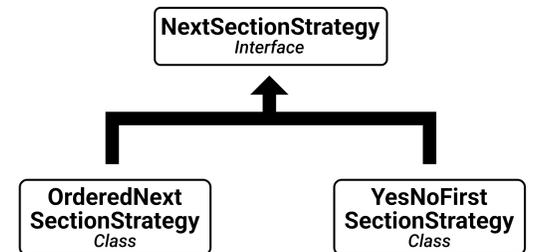


**Figure 4** M-Pod Instance

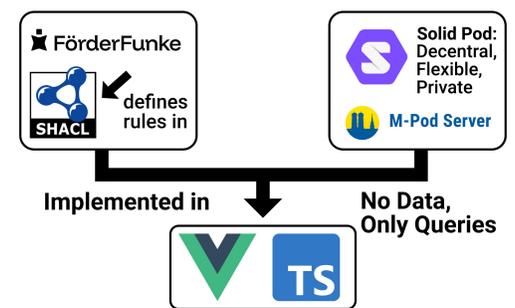
We implemented "M-Pod", based on an open source implementation of the *Solid* specifications, allowing the *City of Munich* to provide self-hosted infrastructure for citizen data.

### 6. Implementation Details

Our *strategy pattern* implementation makes changes to the question selection algorithm easy and allows for *A/B-Testing*.



**Figure 5** Strategy Pattern

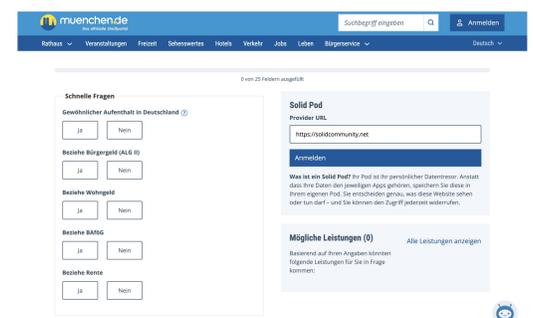


**Figure 6** Our Tech stack

### 7. Conclusion

#### Integration into Digital Citizen Service

Developed on the City of Munich's existing codebase, our approach offers *smooth integration* into their Digital Citizen Service. This allows Munich to lead as an early adopter, setting a precedent for other cities to follow.



**Figure 7** Integration into Digital Citizen Service Dashboard

#### Solid Pods - Digital Sovereignty

The Solid Pod approach proves that efficient administration does not require a "surveillance state." We show that data storage can be *decentralized* and *privacy-focused*. Our Solid Pod integration shows that the *City of Munich* can implement Solid-Pods smoothly across all of their web-services.

#### Modular Question Algorithm

Our implementation utilizes a *Strategy-Pattern* to decouple question-selection logic, enabling the use of different algorithms. This architecture allows the *City of Munich* to conduct *A/B testing* to identify the optimal approach. By deploying the most efficient strategy, the system minimizes the number of required interactions, reducing the effort required from the citizen.