

# Design rationale

This document serves as the design rationale for the UX/UI decisions made in the development of the Sphereon Wallet application. This work is part of the Funke SPRIN-D project, which aims to enhance the usability, accessibility, and trustworthiness of digital wallets. The Sphereon Wallet integrates advanced features to streamline digital credential management while ensuring user confidence and inclusivity.

The following aspects of the Sphereon Wallet's user interface and user experience will be discussed in this document:

- 1. Activity Screens**  
A detailed breakdown of how user activities are categorized, displayed, and made accessible, ensuring clarity and simplicity.
- 2. New Contact Detected Screen**  
Including warnings for interacting with parties that have a low level of trust, leveraging the OpenID Federation (OID Fed) framework to support informed decision-making.
- 3. Multiple Credential Selection in the OID4VP Flow**  
An explanation of the redesign to simplify and optimize the process of selecting multiple credentials for sharing.
- 4. Proof of Concept: Conversational UI**  
Implementing an experimental conversational user interface for receiving credentials and providing context-aware interactive help, ensuring a smooth and intuitive user experience.
- 5. Accessibility: High-Contrast UI and Speech Support**  
Designing for inclusivity with features tailored to users who are visually impaired or face language barriers, including high-contrast themes and speech support.
- 6. New Settings Screens**  
A closer look at the redesigned settings screens, emphasizing user control and ease of navigation.
- 7. In-Person Sharing**  
Simplifying the process for in-person credential sharing with QR code-based interactions.

This document provides the rationale behind each of these design elements, highlighting the decisions made to support a secure, user-friendly, and accessible digital wallet experience.

## 1. Activity screens

### 1.1 Background

For the second stage, we conducted an iteration session on the activity screens for both credential activities and contact activities. In the original design, activities were placed on the details screen beneath a tab bar. However, this design lacked a clear distinction between the different types of activities.

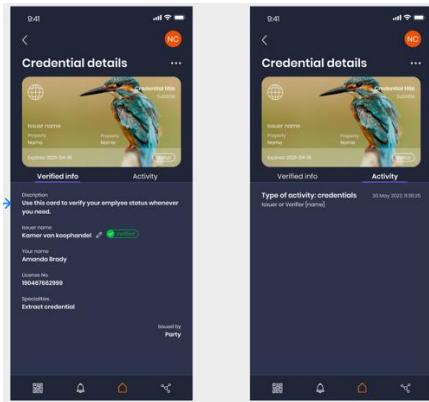


Figure 1 Old design

Rather than only showing the activities only for each contact and credential separately, a common activity feed was implemented, and user tested on a very subtle level with 7 participants. The testing has shown that there are some users (3), which are using different entry points to each activity. However, there are also users (4), which really relied on the general activity feed. For that reason, we started implementing it already now and as a very prominent feature by being part of the navigation bar.

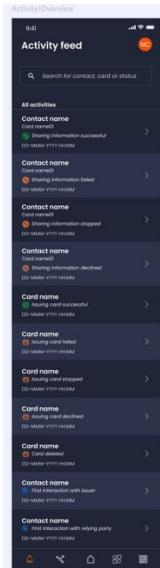


Figure 2 New general activity feed

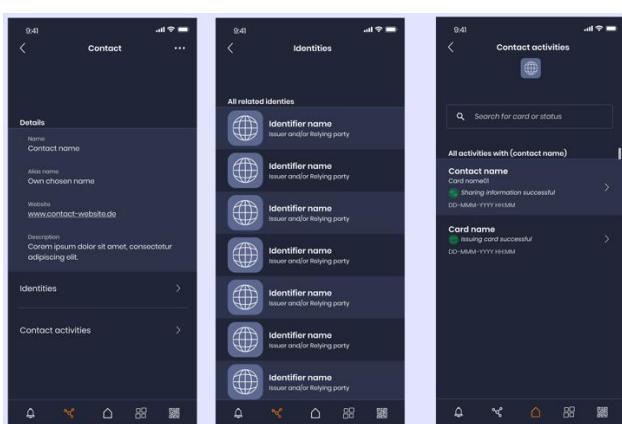


Figure 3 New contact details activities

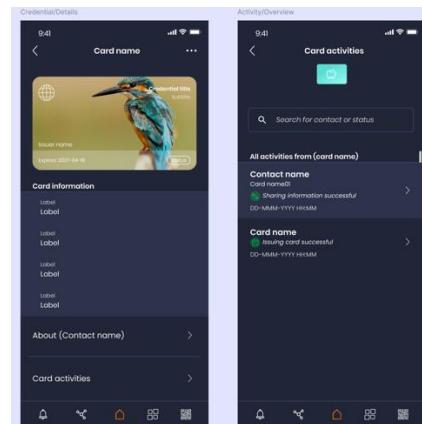


Figure 4 New credential detail and activities

## 1.2 Activity elements

We categorized the activities into two groups: credential activities and card activities. Since these two types are combined,

we need to design activity elements that are consistent and easy for users to read when displayed in a list format.

In the original design the element only contained: type/name of activity and who the interaction was with a timestamp.

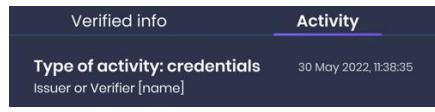


Figure 5 Old activity element

Through multiple iterations, it became clear that the text alone is not sufficient to explain what is happening—visual cues such as colors and illustrations are equally important. In our initial iterations, we incorporated the mini-card and Contact logo components already used elsewhere in the Sphereon wallet. However, combining these two activity types resulted in a visually cluttered design. To achieve a clean and easily scannable list, we needed to refine the approach.

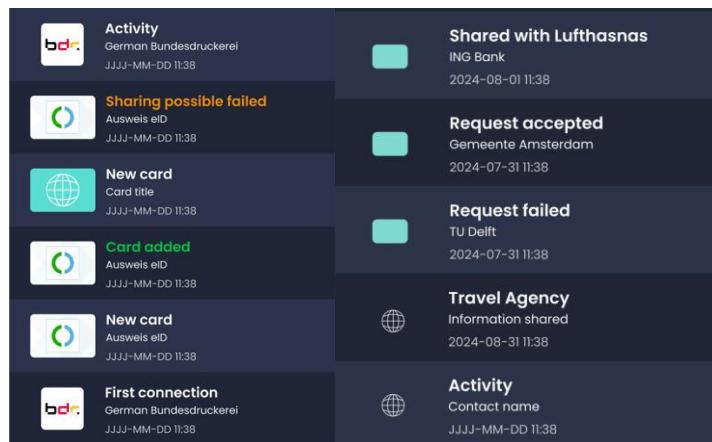


Figure 6 Iterations on acitivity element

In several of the next iterations, we changed the list item's content by asking questions like: **WHAT** information was requested? From **WHICH** credential? **WHY** information about “company” and not about the credential?

We concluded that combining text elements with a visual hint to indicate the type of activity is a better approach than using the mini-card and contact logo. This visual element can be displayed when the user clicks on an item to view more information about the activity.

To refine this idea, we began exploring different types of icons that could effectively provide these visual hints.

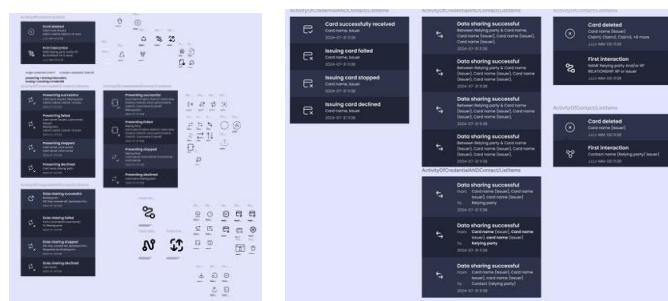


Figure 7 & 8 Explorations for the activity element

Eventually, we developed a set of icons to represent different types of interactions:

- 1. Credential Sharing: Stopped, Failed, or Declined**
  - Indicates an issue with the interaction.
  - Displayed in red with an icon symbolizing a halted interaction.
- 2. Issuing Credential: Stopped, Failed, or Declined**
  - Indicates an issue with credential issuance.
  - Displayed in red with a card icon that incorporates a cross.
- 3. Credential Deleted**

- Indicates that the user has deleted a credential.
  - Displayed with a card icon and a cross to signify the credential is no longer in the wallet.
- 4. Credential Information Sharing: Successful**
- Indicates the credential sharing was successful.
  - Displayed in green with arrows symbolizing a successful exchange.
- 5. Issuing Credentials: Successful**
- Indicates the credential issuance was successful.
  - Displayed in green with a card icon and a checkmark.
- 6. First Interaction with a Relying Party or Issuer**
- Highlights the user's first interaction with a new contact.
  - Displayed in blue with stars to indicate it's a new interaction.

This visual system ensures that each interaction type is clearly and intuitively represented, helping users quickly understand the status of their activities.



Figure 9 & 10 Outcome of the explorations of the activity element

### 1.3 Activity details example

As mentioned earlier in the document, users can view more detailed information about each activity by clicking on the respective item.

As an example, the user can find the information they shared with a contact. What means users can review the specific data they shared with the relying party, including:

- **The Credential and Its Shared Attributes**
  - Displayed in two layers:
    - On the main screen, only the attribute labels are visible.
    - By clicking on a label, users can view the full content of the attribute.
  - This design prioritizes user privacy and control over the information shared from their credentials.
- **Status of the Sharing**
- **Purpose of the Sharing**
- **Type of Credential**

Additionally, users can navigate directly to the profile of the relying party with whom they shared the information.

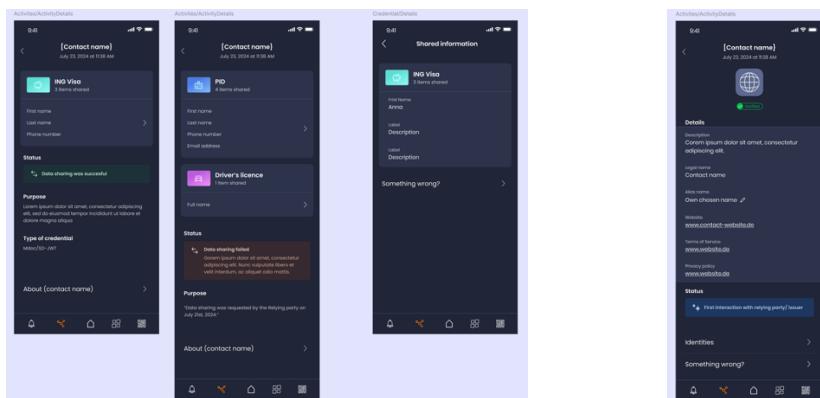


Figure 11 Activity details screen

## 2. New contacted detected screen, with warning for interacting with parties with low-level of trust (based on OID Fed)

For the initial interaction with a new contact (e.g., during a flow to obtain a new credential), we originally designed a screen where users could identify the contact as new and optionally change the alias name for this contact.

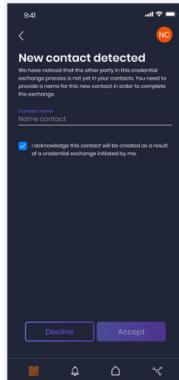


Figure 12 Old contact detected design

In the updated version, users can now provide additional information about the contact during this first encounter, such as the contact's name, logo, website URL, Terms of Service (TOS) URL, Privacy Policy URL, and contact details (e.g., email, website, address). This enhancement gives users a clearer understanding of who the new contact is and allows them to decide whether to proceed with their primary action—based on whether they trust the contact.

The updated new contact screen also includes indicators for technical trust. For example, it shows whether the party is part of a federation. If they are, users can also access information about the Federation Operator, including their contact details.

By providing this comprehensive information upfront, we can reduce the amount of information presented on subsequent screens, lowering the cognitive load and making it easier for users to review the details.

A new feature we are introducing on the new contact screen is the **technical trust** element. This gives users an additional way to assess the trustworthiness of the party they are interacting with. If the party is part of one or more federations, they are labeled with a higher level of trust.

This element also displays the specific federation(s) the contact belongs to, and users can access detailed information about the federation. Since federations are automatically added as contacts in the wallet, users can easily reference them for further verification.

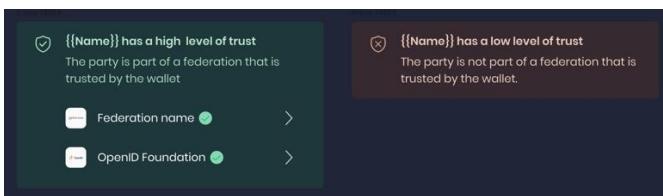


Figure 13 High and low level of trust element

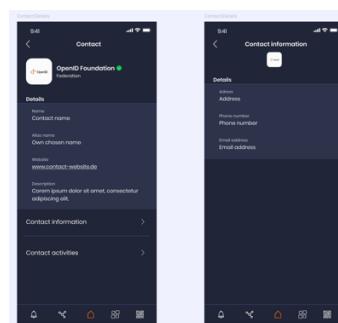


Figure 14 Federation contact details

In future designs, as the wallet progresses to a more advanced stage, the **technical trust** feature will be expanded into a more comprehensive system for evaluating trust levels. This could include reviewing specific attributes of the new contact and assigning a trust score on a scale from 1 to 10. This enhancement will provide users with a clearer and more nuanced understanding of the contact's trustworthiness.

In the new designs, users will not only see the trust level of the new contact but also have access to more detailed information about the contact. After reviewing this information, the user can decide whether to proceed with the primary action by clicking the "Yes, Continue" button or not.

If the user chooses to proceed with a contact that has a low trust level, they will first receive an additional Snackbar notification asking for confirmation to ensure they are certain about interacting with this party.

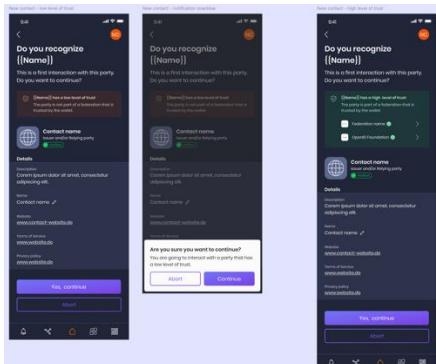


Figure 15 final design of the New contact detected screen

In the old design, the main action was assigning an alias name to the new contact. While this feature is still present in the new designs, it no longer has the prominent role it once did. Instead, users can change the alias name by accessing the **edit (pen)** icon in the contact's details.

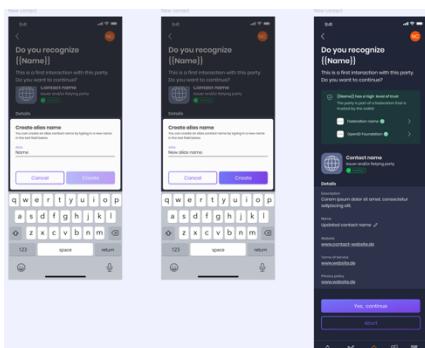


Figure 16 change alias name design

### 3. Multiple credential selection screen in the OID4VP flow

During testing of the Sphereon wallet in the OID4VP flow with multiple participants, we noticed that the screens for selecting which credentials the user wants to send were confusing and involved far more steps than necessary.

In the old design, the process consisted of three screens. The first screen informed the user about which information needed to be sent. When the user clicked on an item in the list, they were taken to a second screen showing which credentials met the requirements. On this screen, they could select the credential they wanted to send. Additionally, by clicking on a credential item, they could view its details on a third screen and select it there for sending.

This design required users to switch between screens multiple times to select the correct credentials, making the process both time-consuming and potentially confusing. Frequent screen transitions also made it harder for users to understand where they were in the overall process.

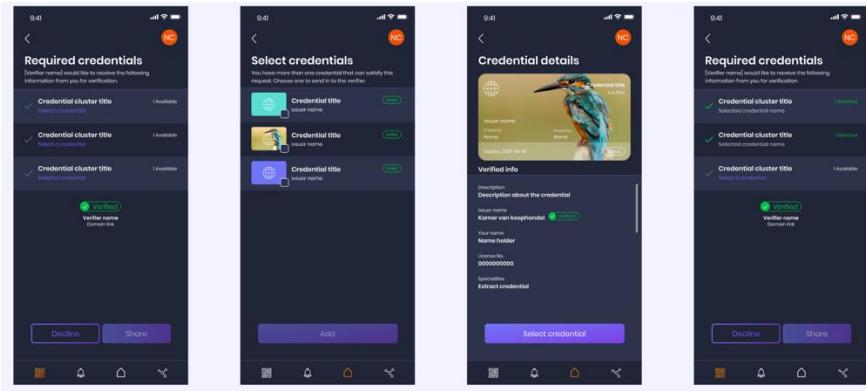


Figure 17 Old selections screens

For the new designs, we decided to implement a more dynamic screen. This approach allows all decisions to be made on a single screen. Thanks to the newly introduced "New Contact Detected" screen (described in Chapter 2), there is now more space available to focus on the primary action: selecting the correct credential and reviewing the information the user intends to send to the relying party.

We decided to continue displaying which relying party is requesting the information on the credential selection screen. This is important because the "New Contact Detected" screen (shown only when a contact is new, has updated metadata, or has a changed trust level) may not always appear before the credential selection screen. Users will still have the option to navigate to view more details about the relying party.

Additionally, the overall purpose of the request is displayed prominently at the top of the screen. Below that, the requested credentials and the information the user intends to send are shown.

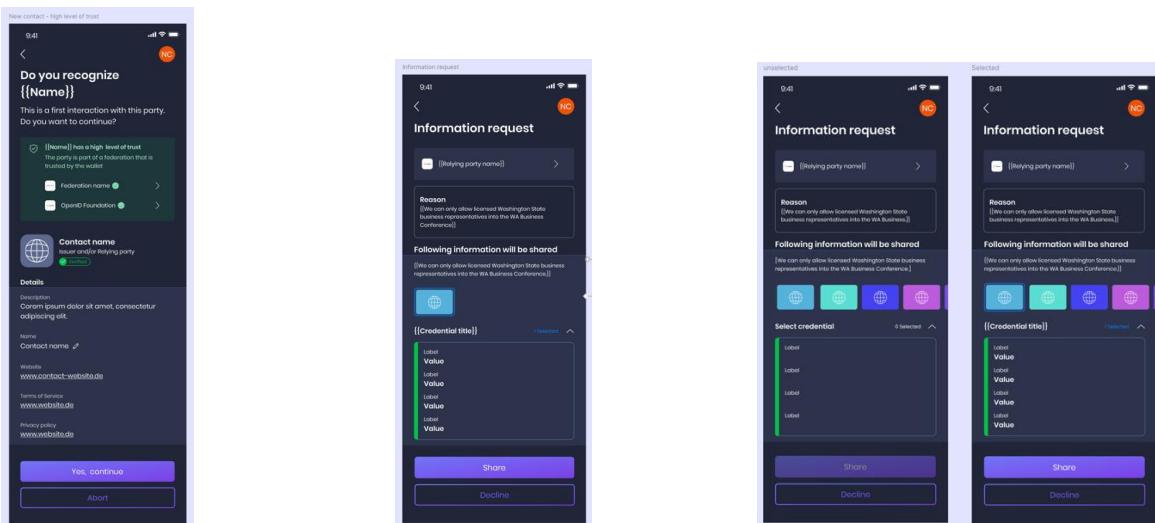


Figure 18 New contact detected screen

Figure 19 New credential selection

Figure 20 New multiple credentials selection

## 4. Implementing an PoC of a conversational UI for receiving credentials and context-aware interactive help

### 4.1 General

#### 4.1.1 Why a conversational UI in general?

A conversational UI offers the unique opportunity to explore AI from a developer and designer perspective. At the same time, it allows the German user to experiment and be introduced to AI in a governmental context (provider).

Receiving a credential with a QR code is one of the most important features of the App. The way of interaction with a credential is novel to the user and therefore the usage of new technologies creates a potential bridge to come up with innovative ideas.

Germans are known for their need for security and for data protection - with the CUI we aim to provide security, reassurance, and the opportunity to understand the context better through searching and finding answers (not tested with German citizens).

A cultural comparison among the US and China indicated that Eastern as well as Western countries considered voice based conversational agents as more positive than the text-based ones (Liu et al., 2024). Having in mind that Germans trust robots in general less due to their individualistic cultural approach (Rau et al., 2009), it may be an opportunity to rebuild trust through a voice-based CUI.

P.L. Patrick Rau, Ye Li, Dingjun Li, Effects of communication style and culture on ability to accept recommendations from robots, Computers in Human Behavior, Volume 25, Issue 2, 2009, Pages 587-595, ISSN 0747-5632, <https://doi.org/10.1016/j.chb.2008.12.025>.

#### 4.1.2 The choice of using both a text interface and a voice interface

- **Target wide range of people.** The current user group is basically the entire German population, which includes digital natives and digital immigrants. Therefore, we want to lower the barrier and make digital governmental services inclusive and accessible in offering multiple access points.

Dutta, 2024, Chatbot vs. Voice Bot: Key Differences <https://www.sprinklr.com/blog/chatbot-vs-voice-bot/>

- **Flexibility for different contexts of use** require different type of interactions (noisy environment vs. being home). This allows users to choose based on their situation and needs.

⚠ Important: What differs the two interfaces?

We have a textual interface with the option to read aloud (text-to-speech), and also speech-to-text for the user input. We have a voice interface that supports text-to-speech and speech to text as a output.

#### Cultural Deep Dive Germany from Max Planck Institute 2014

Much more (and also more positive one), but these ones are very fitting for the project. Listed as 'Characteristic' (Category).

- Need for security (*German mentality*)
- Good results are more important than fast results (*Passion for finding good solutions*)
- Understand the problem first, then solve it (*Passion for finding good solutions*)
- Getting to the core of a subject (*Passion for finding good solutions*)
- Security, privacy, over-anxious, dislike risks, inflexible, straightforwardness, honesty / clear statements (*Value / Behavior / External View*)
- Critical reflection, Call things into question, Think something through, Objective thinking, Task-orientation, Rule-oriented, internalized control, Low-context communication style (*The German way of thinking*)

Data protection & Security

History (3rd Reich, DDR): The shared experience was that no one could trust in their privacy, and deviant behaviour could be punished severely - Consequences are still very present today (<https://www.dotmagazine.online/issues/security/germany-land-of-data-protection-and-security-but-why>)

## 4.2 Product features

### 4.2.1 Multimodality

**A wide group of users has a broad range of needs** - this is the main reason for offering a textual interface and a voice interface in all possible combinations.

Offering the information in different medial formats may support users in processing the content more effectively and help them understand **better (Reduced cognitive load)**. Therefore, the future vision of the CUI is implementing more graphics and videos in their explanation, rather than only showing text.

A study by Brachten et al. (2020) has shown that the support of a virtual agent leads to a lower cognitive load when solving a task.

Brachten, F., Brünker, F., Frick, N.R. et al. On the ability of virtual agents to decrease cognitive load: an experimental study. Inf Syst E-Bus Manage 18, 187–207 (2020). <https://doi.org/10.1007/s10257-020-00471-7>

The modality principle by Mayer (2009) underlines the importance of mixed media inputs through different channels, such as the auditory and the visual one.

Mayer, R. E. (2009). Modality Principle. In *Multimedia Learning* (pp. 200–220). chapter, Cambridge: Cambridge University

Therefore, we offer for example the text input with the option of reading out loud in the chat interface, and on the same time a voice interface with additional text support for clarification, **aiming for richer and more engaging conversations**. In the future, the user shall be able to define his/her preferences in the CUI.

Another reason for supporting the voice interface with textual content is to demonstrate the users that the AI is understanding and responding accordingly to what the users say (Niels).

#### 4.2.1 Context awareness

With the context awareness of the bot, we are aiming for responses that are **precise, relevant, and tailored to each user's needs**. We hope to achieve a straightforward and enriching user journey, rather than distracting the user from their primary objective.

This involves that the bot can intuitively understand the user's intent and react appropriately. By providing short cuts (function calling) to essential product features - like receiving a new credential - we aim to empower all users **to achieve their goals with minimum effort**.

By offering context aware conversation starters, the bot shows clarity in being goal focused, rather than overstimulating the user with unrelated information and choices.

#### 4.2.2 Chat

##### Buttons

The textual interface enables a combination of free text input and predefined buttons to find a balance between infinite browsing with the help of AI and saving typing effort for the users for the more common inputs (Budu, 2018).

Budu (2018), The User Experience of Chatbots <https://www.nngroup.com/articles/chatbots/>

For the design of the buttons a clear distinction was made between the intended interactions: yes/no questions, specific topic selection, process related interactions.

##### Level of interaction

By envisioning the chatbot as smarter and more goal oriented than other chatbots, it was challenging to find a balance between the amount of providing intelligent shortcuts, not duplicating the screen at the same time, and still giving the space to the user to read through the initial content of the screen.

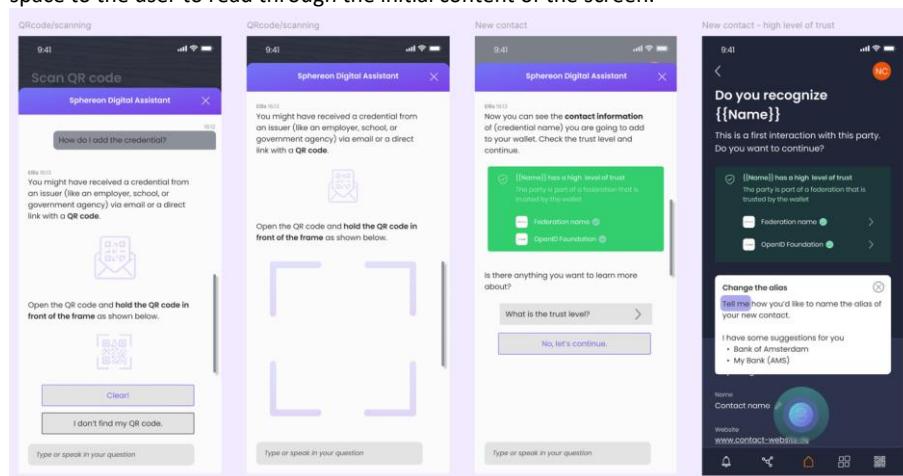


Figure 21 Interaction types

The final predefined buttons within the chatbot allow the user to quickly answer simple yes and no questions, to choose from common questions or to browse deeper on external pages. Finally, there is also offered the option to continue in the process, after gathering all needed information.

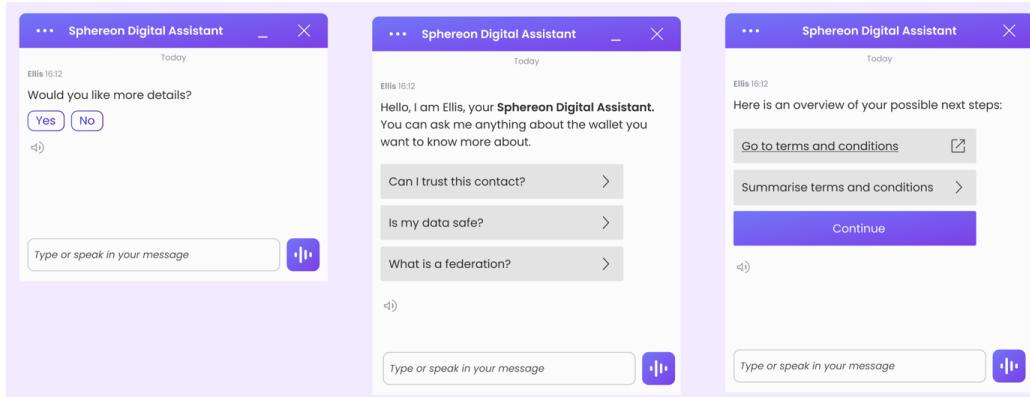


Figure 22 Subject steering

A previous iteration of the button design showed that various intended interactions need to be visually different to create hierarchy and to clearly guide the user through the process.

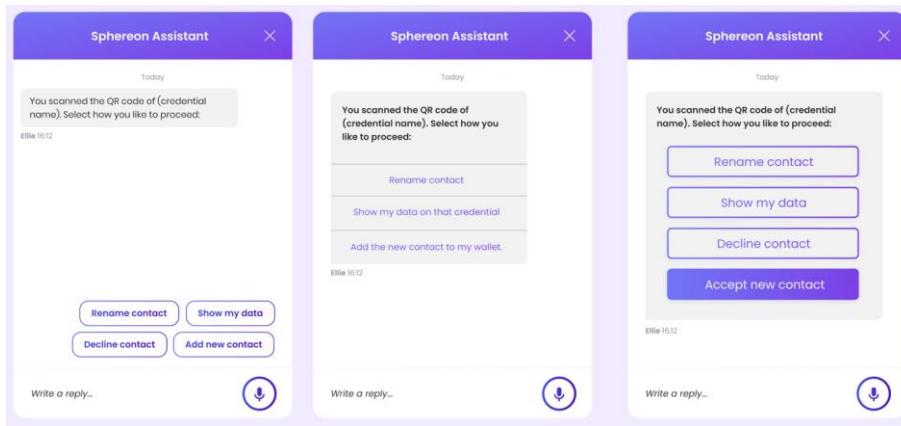


Figure 23 Subject steering

## 4.3 Shape and size

### 4.3.1 Text Chat bot

The textual interface of the CUI is taking on 4 different shapes, which all have their own purpose.

- State 1: The icons showcase the presence of a conversational UI.
- State 2: The context aware text bubbles pop-up and offer support or ideas and invites the user to interact with it.
- State 3: The small modal only pops-up when the user shows curiosity by clicking on the icon or a text bubble.
- State 4: The big modal almost takes in the entire screen for readability and usability reasons. It appears when the user consciously decided to interact with the bot.

With the different shapes and sizes we act according to the users curiosity and interest in the bot. In order to fulfil needs such as being able to see the content of the screen while interacting with the bot, the user has the option of changing sizes easily.

In previous explorations the opportunities between different modal sizes and a full screen interface were explored.

There was a great opportunity discovered to guide the user through the app with an entire new perspective emphasising only the needed user inputs (→ Initial shape exploration, right side). However, considering the integration of AI, the users unfamiliarity with AI (BSI, 2024 Link: [https://www.bsi.de/documents/Gesellschaftliche\\_Akzeptanz\\_KI\\_in-Deutschland\\_01182024.pdf](https://www.bsi.de/documents/Gesellschaftliche_Akzeptanz_KI_in-Deutschland_01182024.pdf)) and the human need for control (→ Desmet, 2020, 13 Fundamental Psychological Needs), it would have been risky to rely solely on a full-screen, AI driven interface that replaces the initial one.

For that reason, we decided to go for the more classical approach of a chat interface, with that the user may already be familiar with from other applications, to test the integration of AI and the function calling functionalities in the first place.

#### 4.3.2 Voice interface

During the exploration of the voice CUI various shapes were explored through an analysis of existing conversational agents (Siri, Alexa, Google Assistant, Bixby). Due to prototyping constraints, a button to record the user's voice input became a technical necessity, which created the challenge of managing multiple buttons on a compact interface. To address this, the icons were designed with a balanced and minimalist approach.

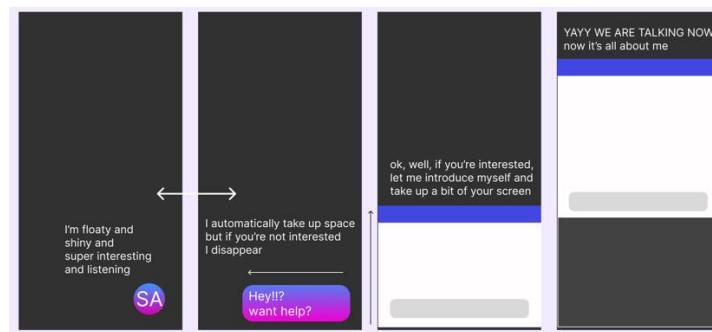
The choice of not relying on a full-screen interface are the same as for the textual interface: allowing the user a sense of control and overview when using the 'function calling' functionalities. Furthermore, we

Finally, the voice interface has similar states as the chat interface

1. The Icon
2. The context aware text pop-ups
3. Listening: Showing the user input
4. Speaking: Showing the generated output

Sketch of different active states (Chat interface)

This sequence sketches out the final types of shapes, that the classical chatbot takes on. With fixed sizes according to the needs we prevent a jumping bot.



24 Active states

#### Initial shape exploration

A very first exploration led to two distinct approaches. On the one side, it resulted in a typical chatbot interface (left) that accompanies the user while using the app. On the other side, it led to an entire new full-screen chat interface (right) that replaces the initial screen.

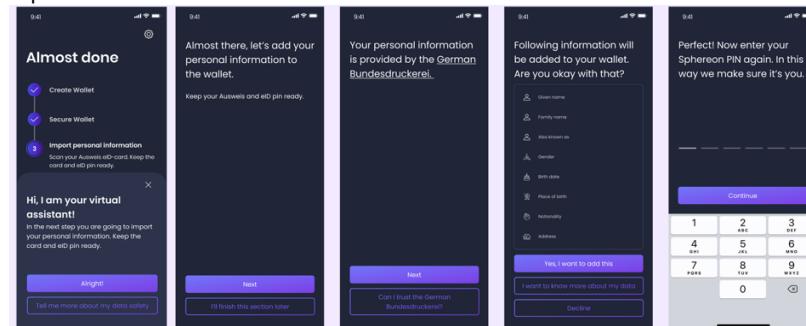


Figure 5 Intital shape exploration

#### Space and output exploration - voice interface

Exploring the space that a voice interface may take in when supporting it with text.

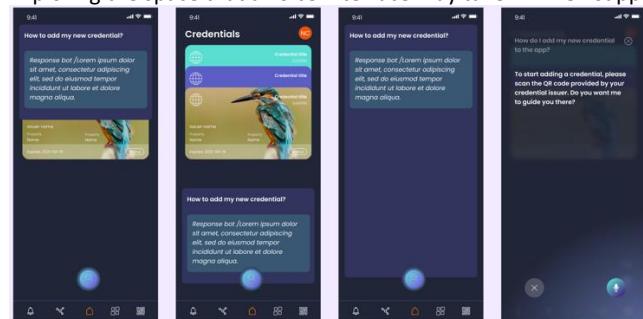


Figure 25 Space and output exploration - voice interface

Exploring the amount of output that is needed for the user. There are three options:

1. Showing the user input and bot output.
2. Showing the bot output only.
3. Showing only confirmations of actions done by the bot.

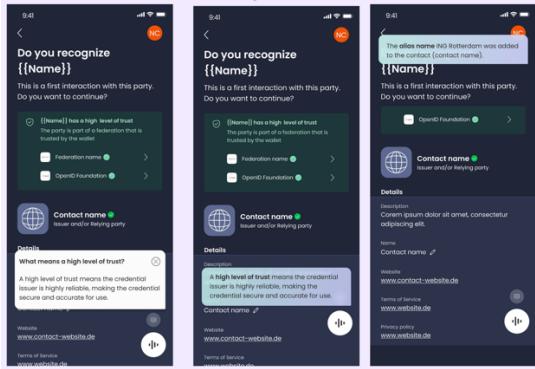


Figure 26 Output amount

#### Shapes exploration - voice interface

The exploration of the shape of conversational interfaces was based on market research. The examples explore the presence of an interactive element that indicate its listening activity.

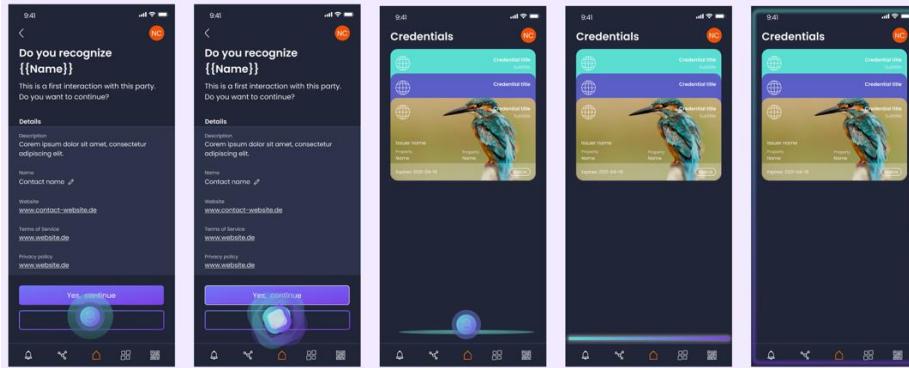


Figure 27 Shapes exploration - voice interface

#### 4.4 Settings

The conversational UI shall be as accessible and adaptable to each users needs; therefore, a range of settings are required.

**CUI activation:** For now, it is possible to enable one preferred conversational interface (auditory or text) or to entirely disable the CUI. This decision was made in case the user does not show any interest at all in interacting with the CUI and not bothering him unintentionally. The preferences of the CUI are for now indicated in the general settings of the app to focus on the current interaction with the CUI only.

The future settings and preferences are enormous dependent on the user feedback and their needs, which may affect the placement, but also the notifications and the chat history itself.

- **Placement:** Dependent on the urgency and frequency of adjusting the preferences, they may be placed in the CUI itself.
- **Text messages:** It needs to be researched, what level of detail is desired in presenting the users input and output within the voice interface. For now, both the input and the output are presented to ensure the correctness of the input. However, the preferences may vary, which needs to be tested (hints, user input in text, responses in text and/ or voice only).
- **Chat history:** It should be an option to easily look up previous conversations with the CUI, maybe even through another function calling functionality in the bot itself. But this is also dependent on the users needs.

For now, the CUI can be disabled, which is represented in the left example. Future settings and preferences need to be planned and adjusted to the users needs and can result in an extra settings option within the chat or via function calling (right).

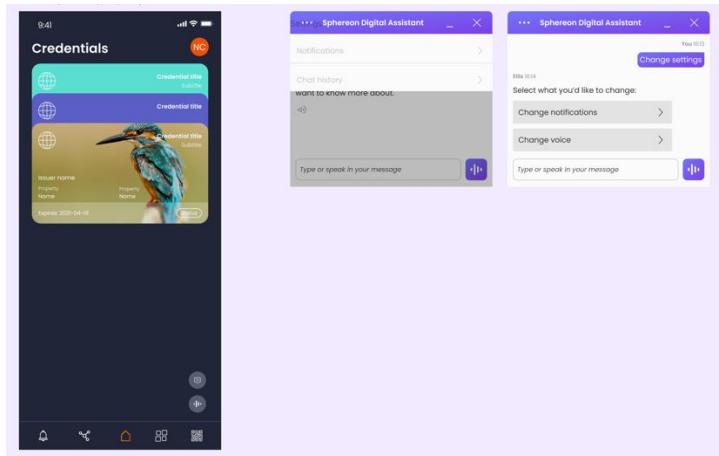


Figure 28 Chatbot settings

## 4.5 Conversational user interface

### 4.5.1 Attributes

The conversational interface should communicate to the user in a manner that is informative, forward thinking and goal oriented - qualities that are intrinsically linked to the product features and an improved user experience. In that communication style German values should be represented.

- **Informative:** Rather than simply asking how it can help, the bot shall proactively provide relevant information while initiating support, reducing the need for users to search for answers.
- **Forward thinking:** The CUI shall guide the user towards finishing the process, while answering all needed questions. The bot shall provide smart suggestions or prompts for potential next steps.
- **Goal oriented:** The CUI shall be direct and concise, having in mind that Germans prefer to have information transmitted explicitly without having a detour.
- **Easy language:** The CUI's language shall be straightforward and easy to understand, ensuring that all users feel well-informed and at ease.

### 4.5.2 Look and Feel

The name of the conversational UI is "Ellis" (gender neutral name, meaning connected to intelligence) to give it a personality. To transmit transparency to the user, the CUI introduces itself as an artificial intelligence CUI. Giving the bot an avatar or any graphical look was less important in this iteration due to a German low context culture.

The metaphor of the lighthouse keeper is used to describe the conversational UI's behavior:

- Responsible in giving signals
- Clear communication
- Important role in rescuing and searching

## 5. High-contrast UI and Speech support for visually- or language-challenged users.

### 5.1 Background

#### 5.1.1 Why accessibility settings

To accommodate users with visual impairments, including those who are colorblind, we wanted to introduce a suite of accessibility features. These enhancements include the option to select colorblind-friendly color palettes, enabling those with various types of color vision deficiencies to navigate the application more effectively with text to speech functionality on the operating system.

#### 5.1.2 What is supported

In the current version of the app, we allow the user settings in the operating system to be adopted in the app. This means, that activated accessibility features such as "Increase contrast" in iOS or "High Contrast Text" in Android are natively supported by the Sphereon app. The same is valid of an increased font size.

In addition, the Sphereon App also supports the Voiceover (iOS) and talkback (Android) mode, which enables the user to navigate through the app with a keyboard and to have the navigation, action buttons and text to be read aloud.

All supported features aim to support and include people with different visual impairments in order to manage their personal data

### 5.1.2 What is planned/ considered

In order to provide every user, the best screen experience, we propose to define the app internal accessibility settings already before the onboarding. We believe that we may pick up a broader range of users when just presenting the preferences as an option at the beginning.

Furthermore, we explored the increased text size variation. The fact that the operational system offers such accessibility feature, doesn't mean that the supported apps are presented in the best way (-> *Operating system features*). Therefore, we propose the preference settings within the app and offer an adjusted design with an optimal user experience (-> *Design directions for accessibility mode*).

### 5.1.3 Why a separate high contrast mode?

The AAA accessibility standards serve an additional audience (monochromatic, achromatic, daltonism, etc) and the current app is mainly AA.

A separate high contrast mode enables the AAA mode whilst leaving the initial AA mode untouched. According to Georgiev (2024) such modes are designed to support “readability and usability of interfaces for users with visual disabilities, including color blindness and low vision.”

Georgi Georgiev, The Importance of High Contrast Mode in a Design System, 2024, <https://pros.com/ascend/the-importance-of-high-contrast-mode-in-a-design-system/#::text=High%20contrast%20themes%20are%20designed.color%20blindness%20and%20low%20vision>.

A separate high contrast mode offers more control on the outcome, enabling designs that show both branded elements such as contacts or credentials and optimal contrasts using pure black and white.

## 5.2 WCAG Guidelines / Color contrast check

An evaluation of the current brand colors, and has shown that the Sphereon app mainly succeeds within the AA accessibility standards.



*Figure 29 Color exploration*

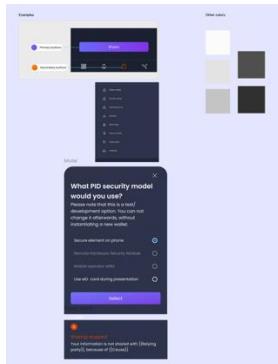


Figure 30 Color exploration elements

### Operating system features

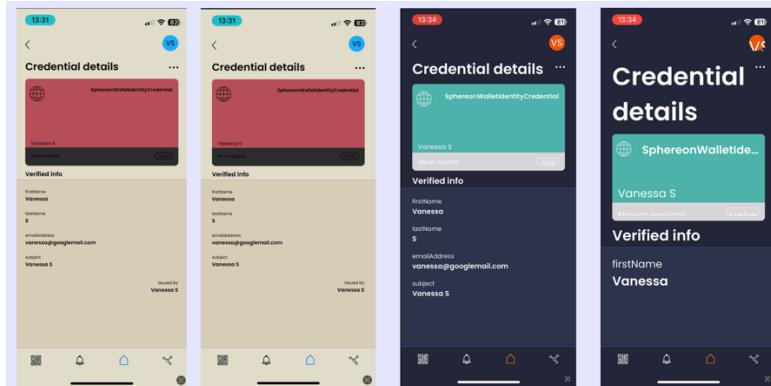


Figure 31 Operating system features

### Sphereon Initial colors - color check

Checking the current app with a focus on different color blindesses, it can be said that the current primary app color is well chosen as a neutral action button. However, considering the contrast for color blind users, it is rather poor than sufficient.

### BLUE- YELLOW BLINDNESS

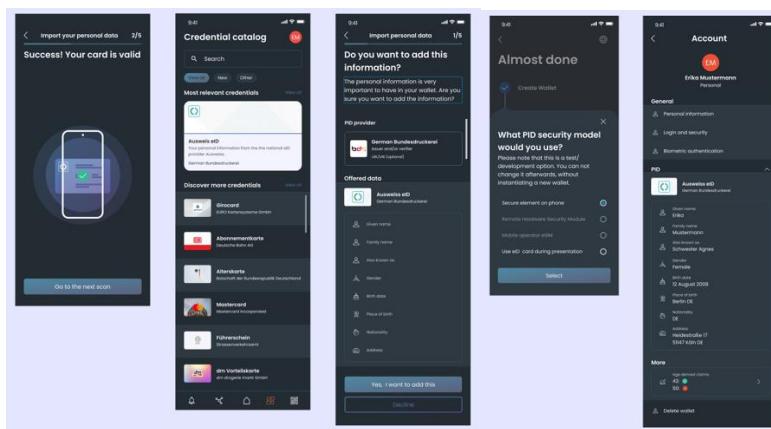


Figure 32 Version 1

### RED-GREEN BLINDNESS

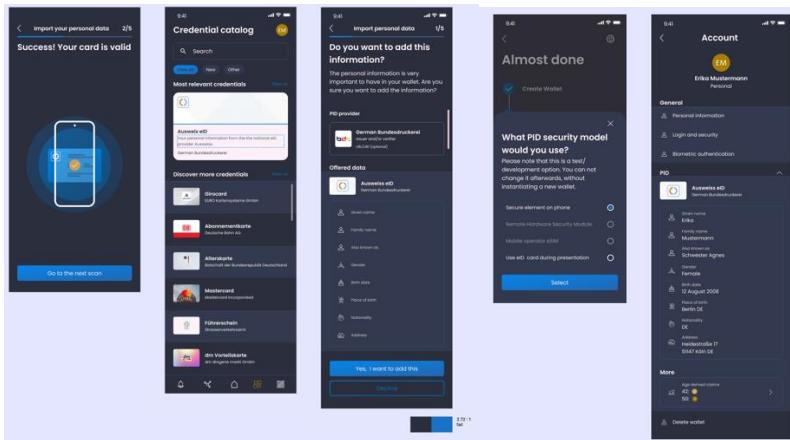


Figure 33 Version 1

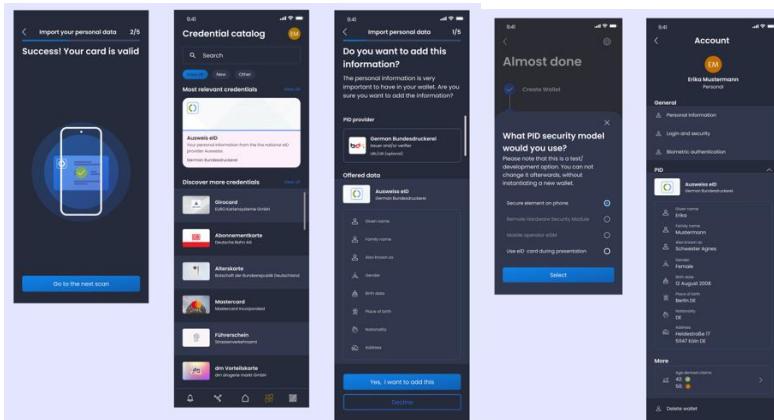


Figure 34 Version 2

## BLUE- YELLOW BLINDNESS

### Version 1

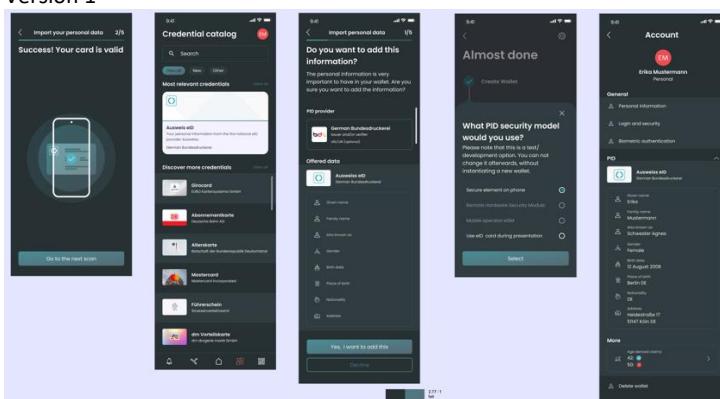


Figure 35 Version 1

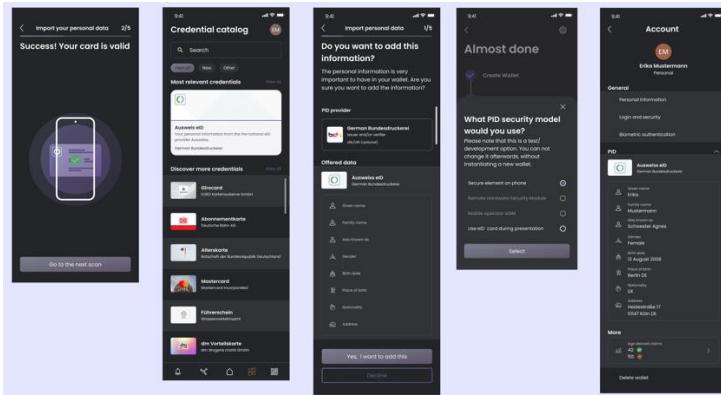


Figure 36 Version 2

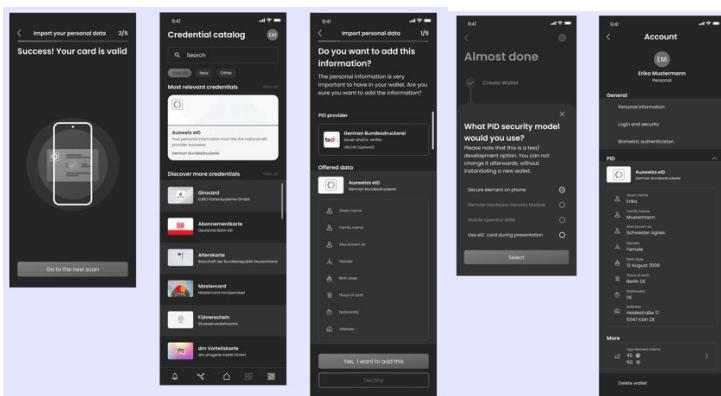


Figure 37 Version 3

#### Design directions for Accessibility mode

A proposal was made in what the user has the option to change the display preferences already before the onboarding process. Within the settings the user can activate the high contrast mode, adjust the text size, and enable an audio assistant, which is reading out loud the text.

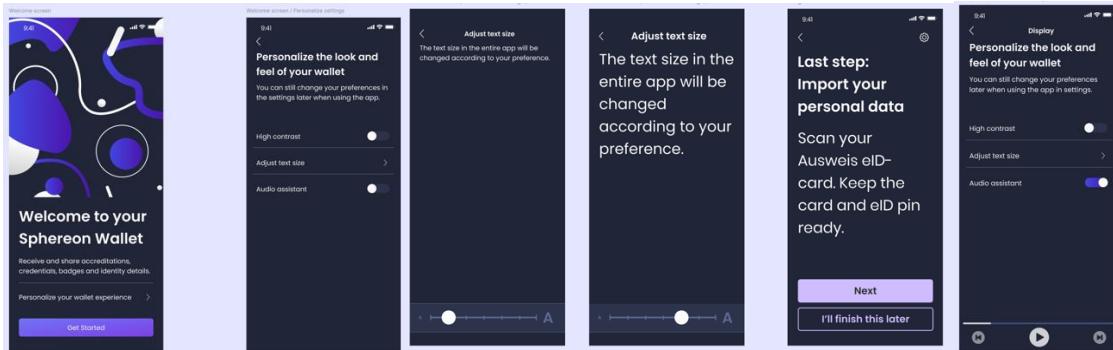


Figure 38 Accessibility mode

## 6. Settings screens

For the settings screen we didn't have a design, where the user could manage their wallet. They could only manage the most basic features from the entity button: log out or delete the wallet.

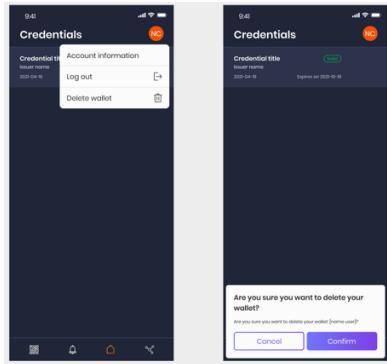


Figure 39 Old settings dropdown

From the entity button in the right upper corner, you now can manage more features as a user. In the design (not implemented) we already have taken in account the switching between wallets in the Sphereon wallet.

In this sprint, we focused on ensuring the user has a single account they can easily navigate to. On this screen, we decided to prominently display the PID. The user can access the PID directly from the overview screen, where all credentials are listed. Looking ahead, as users might manage multiple credentials in the future, the PID will have a prominent place when it's also attached to the account settings.

Currently, the stacking of credentials is the main feature of the credential overview screen. However, in a future version, allowing users to customize the order of their credentials will make it easier for them to find what they need.

Similarly, as we have done with the activity elements—providing users with multiple ways to access them—we aim to ensure the PID is consistently placed in locations where users can reliably find it.

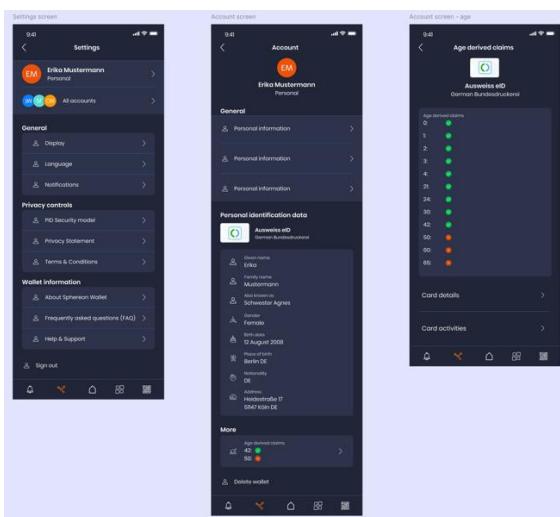


Figure 40 New settings and account screens

## 7. In person sharing

As a holder, they want to be able to share their identity with an RP while meeting said RP in person. As per specifications, this needs to happen in the form of the holder presenting a QR which can be scanned by the RP.

For the in-person sharing screens, our priority at this stage of testing the Sphereon wallet was simplicity. Functionality took precedence. Currently, in-person sharing is accessible via the dropdown menu on the entity button located in the top-right corner. Upon accessing this screen, users are presented with a QR code that they can share directly with the relying party in person.

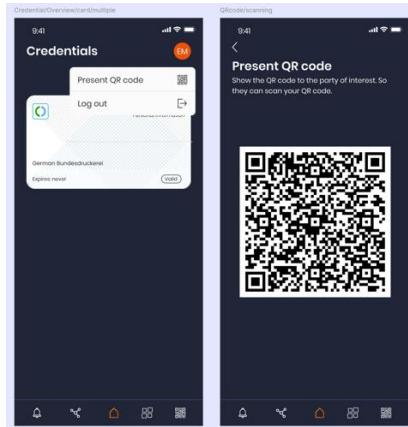


Figure 41 In person sharing