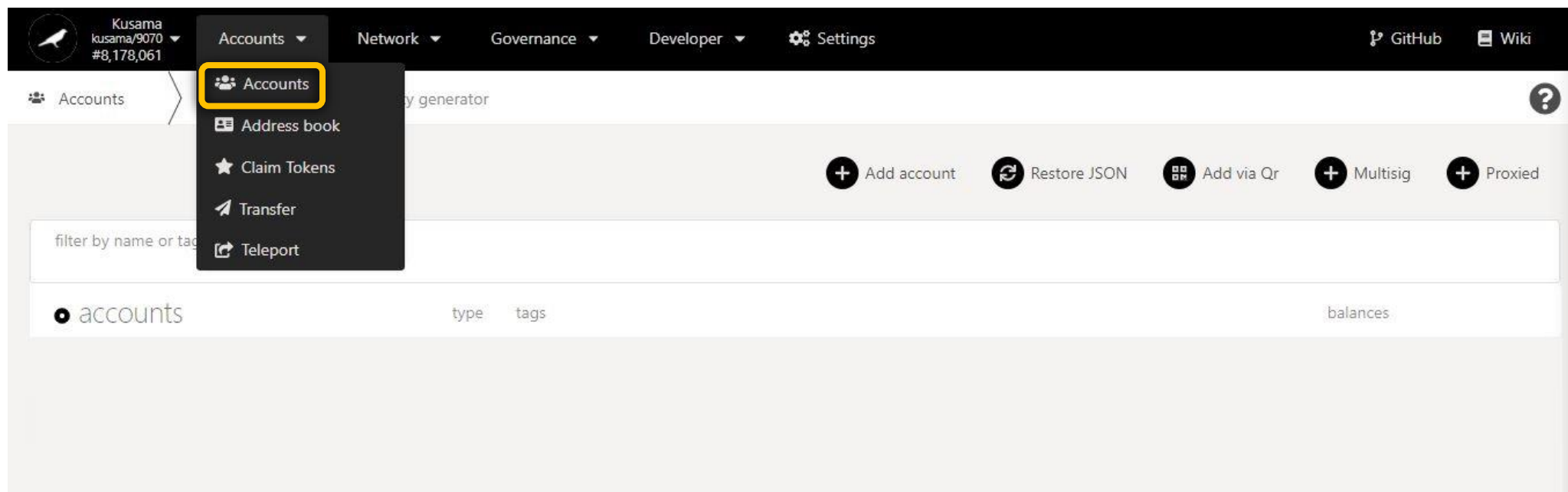


[PART I: ACCOUNTS \(Basic setups\)](#)

1. Accounts: Manage your accounts.



Basic information on technical words used in the ACCOUNTS section.

Options for creating/importing account(s).

Accounts can be **injected** from Polkadot-JS extension, **created** from Polkadot-JS Apps, or derived from **Ledger** hardware.

Any custom information/note about the account.

Existing account(s) enabled on the Kusama network.

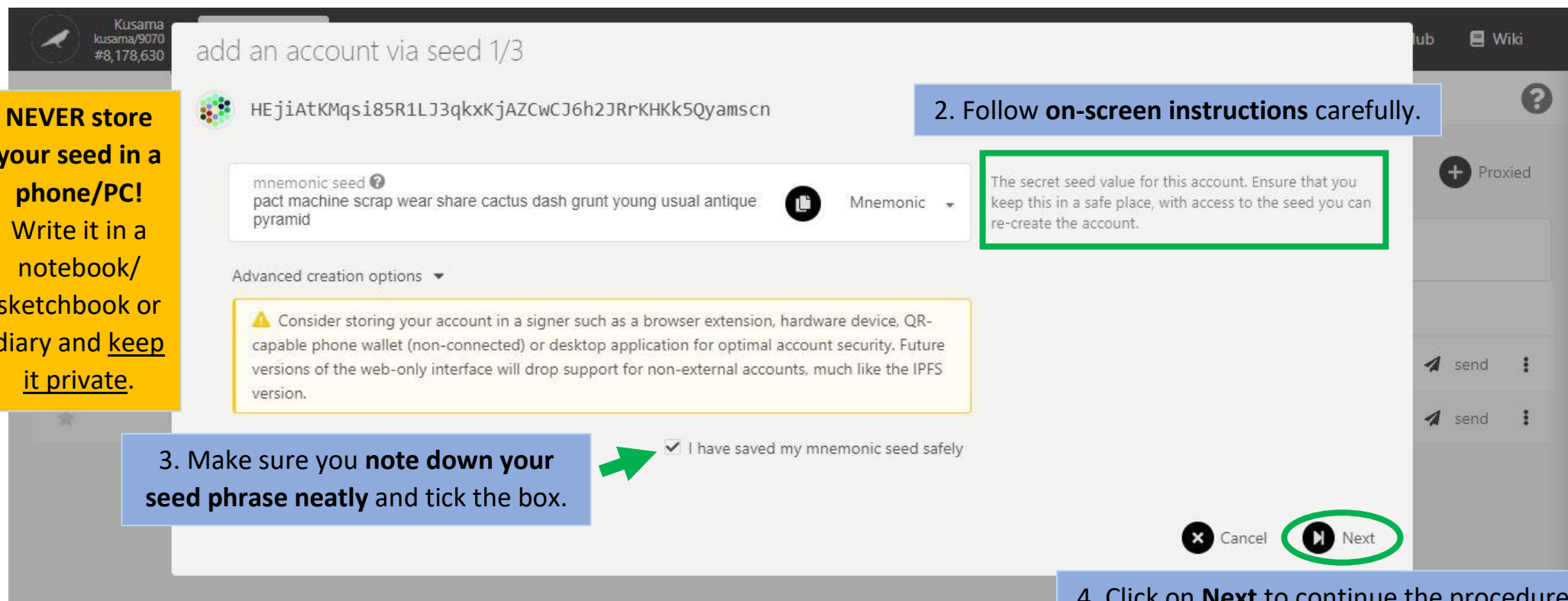
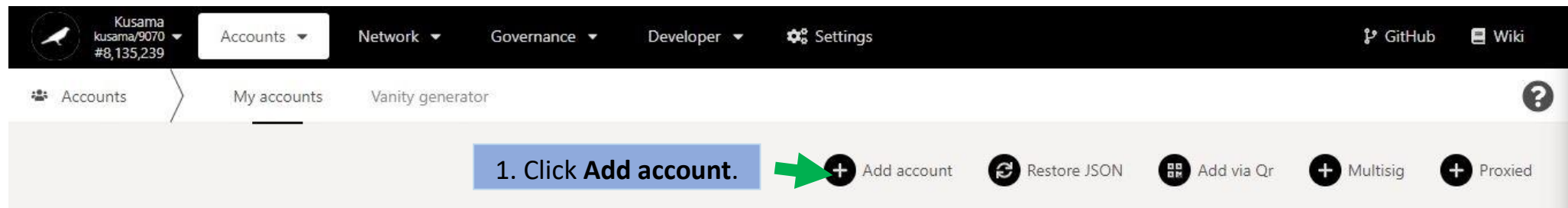
Overview of account's KSM balances.

The screenshot shows the 'Accounts' section of the Polkadot-JS interface. At the top, there's a navigation bar with 'Kusama', 'kusama/9070', and '#8,135,239'. Below this, the 'Accounts' tab is selected, showing 'My accounts' and 'Vanity generator'. A green box highlights the 'Add account' button, with other options like 'Restore JSON', 'Add via Qr', 'Multisig', and 'Proxied'. A table lists existing accounts: 'ALTD@KSM (EXTENSION)' and 'ANAELLE LTD@KSM', both marked as 'injected'. A green box highlights the account names. To the right, a 'balances' section shows KSM balances for the selected account: 10.3509 KSM, 1.0376 KSM, and 11.3886 KSM. Green arrows point from text boxes to specific UI elements: one to the 'Add account' button, one to the account names in the table, one to the 'type' column header, and one to the 'balances' section.

filter by name or tags	accounts	type	tags
★	ALTD@KSM (EXTENSION)	injected	no tags
★	ANAELLE LTD@KSM	injected	no tags

balances	
10.3509 KSM	send
1.0376 KSM	send
11.3886 KSM	

a) Create a Polkadot-JS account from a new Seed and save it in a JSON file.



NEVER store your seed in a phone/PC! Write it in a notebook/sketchbook or diary and keep it private.

GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1

add an account via seed 2/3

LALA
HEjiAtKMqsi85R1LJ3qkxKjAZCwCJ6h2JRrKHKk5Qyamsn

name ?
Lala

password ?
.....

password (repeat) ?
.....

WEAK STRONG

⚠️ Consider storing your account in a signer such as a browser extension, hardware device, QR-capable phone wallet (non-connected) or desktop application for optimal account security. Future versions of the web-only interface will drop support for non-external accounts, much like the IPFS version.

Cancel Prev **Next**

5. Follow the new instructions carefully.

The name for this account and how it will appear under your addresses. With an on-chain identity, it can be made available to others.

The password and password confirmation for this account. This is required to authenticate any transactions made and to encrypt the keypair.

Ensure you are using a strong password for proper account protection.

6. Click on **Next to continue the procedure.**

Avoid using the same password for all your accounts.

GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1

Kusama
kusama/9070
#8,178,651

Accounts

add an account via seed 3/3

LALA
HEjiAtKMqsi85R1LJ3qkxKjAZCwCJ6h2JRrKHKk5Qyamsn

partial seed
pact ... wear ... dash ... usual ...

keypair type
sr25519

derivation path
<none provided>

We will provide you with a generated backup file after your account is created. As long as you have access to your account you can always download this file later by clicking on "Backup" button from the Accounts section.

Please make sure to save this file in a secure location as it is required, together with your password, to restore your account.

Cancel Prev **Save**

8. Click on **Save** to complete the procedure.

GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1

Star your most important accounts.

filter by name or tags

accounts	type	tags	balances
★ ALTD@KSM (EXTENSION)	injected	no tags	10.3633 KSM send
★ ANAELLE LTD@KSM	injected	no tags	1.0376 KSM send
★ LALA	sr25519	no tags	0.0000 KSM send
			11,4009 KSM

9. Your account has been added to the *My accounts* list!

10. Click on the 3 vertical dots to view **Account settings**.

A JSON (backup) file has also been created for your account! It should be stored in a device that is not connected to the internet.

HEjiAtKMqsi85R1....json

Show all

GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1

The screenshot shows the 'Accounts' section of the Polkadot.js Apps interface. The top navigation bar includes 'Kusama kusama/9070 #8,178,673', 'Accounts', 'Network', 'Governance', 'Developer', and 'Settings'. The 'Accounts' sub-section has tabs for 'Accounts', 'My accounts', and 'Vanity generator'. Below these are buttons for '+ Add account', 'Restore JSON', and 'Add via QR'. A search bar labeled 'filter by name or tags' is present. A table lists accounts with columns for a star icon, account name, type, and tags. The 'ALTD@KSM (EXTENSION)' account is selected, and a context menu is open over it, showing options like 'Set on-chain identity', 'Derive account via derivation path', 'Create a backup file for this account', 'Change this account's password', 'Forget this account', 'Make recoverable', 'Initiate recovery for another', 'Delegate democracy votes', and 'Add proxy'. A green arrow points from a blue instruction box to the 'Create a backup file for this account' option.

11. Click on **Create a backup file [...]** to download your account's JSON file.

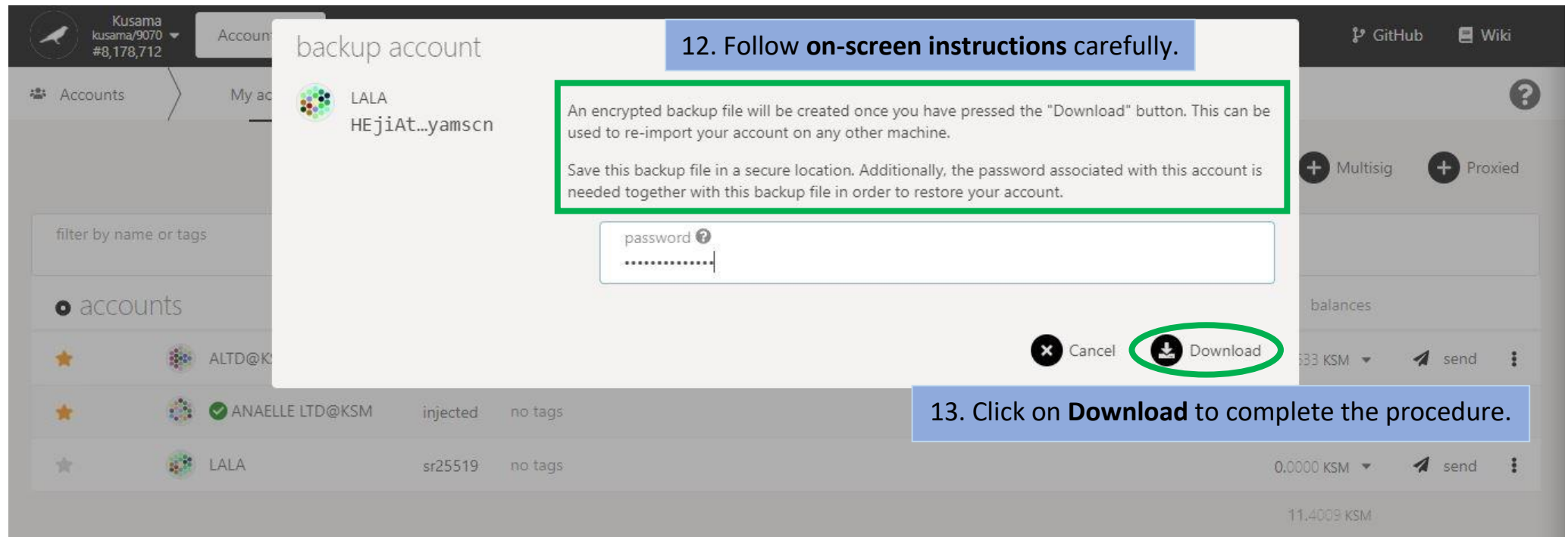
	accounts	type	tags
★	ALTD@KSM (EXTENSION)	injected	no tags
★	ANAELLE LTD@KSM	injected	no tags
★	LALA	sr25519	no tags

- Set on-chain identity
- Derive account via derivation path
- Create a backup file for this account
- Change this account's password
- Forget this account
- Make recoverable
- Initiate recovery for another
- Delegate democracy votes
- Add proxy

only this network ☒

GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1



The screenshot shows the 'backup account' dialog in the Polkadot.js Apps interface. The dialog is titled 'backup account' and displays the account name 'LALA' and its address 'HEjiAt...yamscn'. A green box highlights the instructions: 'An encrypted backup file will be created once you have pressed the "Download" button. This can be used to re-import your account on any other machine. Save this backup file in a secure location. Additionally, the password associated with this account is needed together with this backup file in order to restore your account.' Below the instructions is a password input field labeled 'password' with a question mark icon. At the bottom right of the dialog are 'Cancel' and 'Download' buttons. The 'Download' button is circled in green. The background shows the 'Accounts' page with a list of accounts, including 'ALTD@K', 'ANAELLE LTD@KSM', and 'LALA'.

12. Follow on-screen instructions carefully.

An encrypted backup file will be created once you have pressed the "Download" button. This can be used to re-import your account on any other machine.

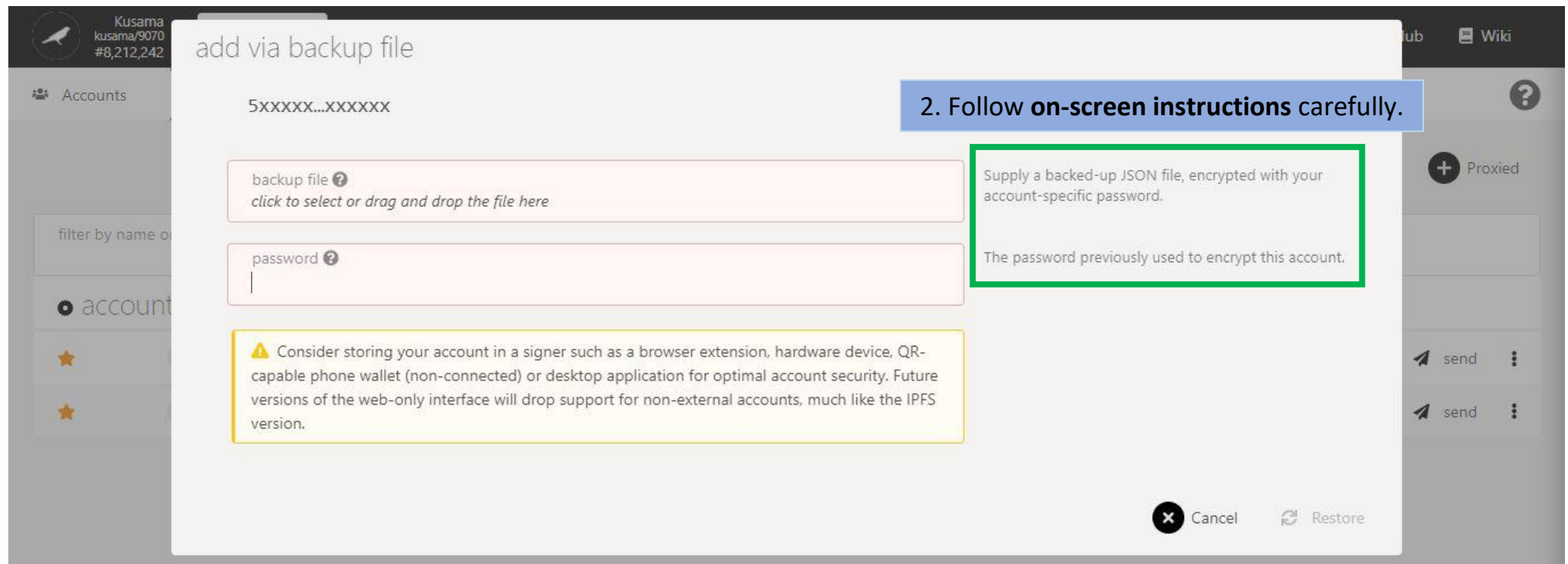
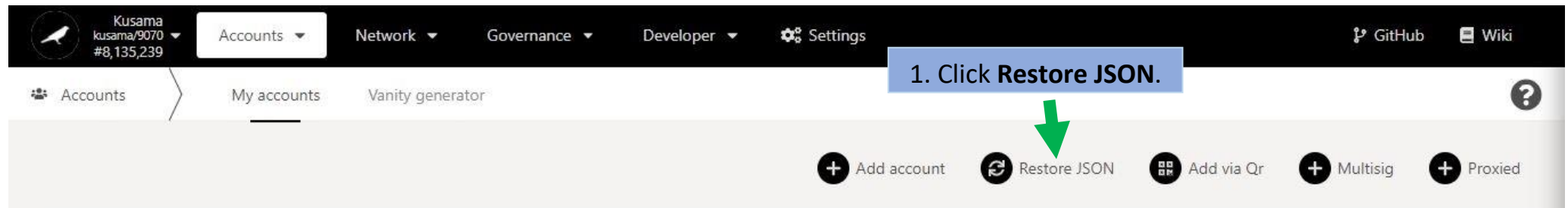
Save this backup file in a secure location. Additionally, the password associated with this account is needed together with this backup file in order to restore your account.

password ?

Cancel Download

13. Click on **Download** to complete the procedure.

b) Import a Polkadot-JS account from a JSON file.



GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1

add via backup file

LALA
HEjiAt...yamscn

backup file ?
HEjiAtKMqsi85R1U3qkxKjAZCwCJ6h2JRrKHKk5Qyamscn.json (600 bytes)

password ?
.....

Supply a backed-up JSON file, encrypted with your account-specific password.

The password previously used to encrypt this account.

Warning: Consider storing your account in a signer such as a browser extension, hardware device, QR-capable phone wallet (non-connected) or desktop application for optimal account security. Future versions of the web-only interface will drop support for non-external accounts, much like the IPFS version.

Cancel Restore

3. Double-check warning messages.

4. Click on **Restore** to complete the procedure.

GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

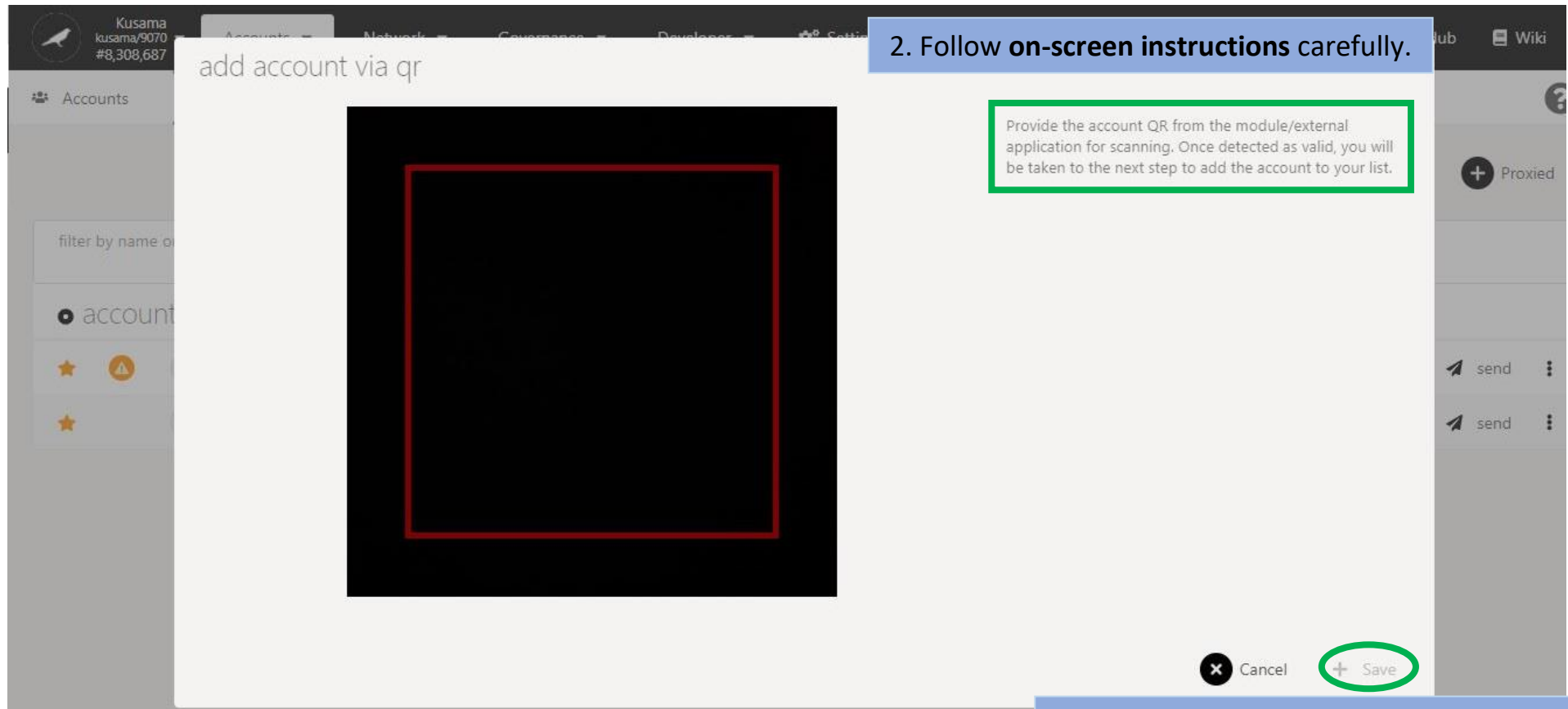
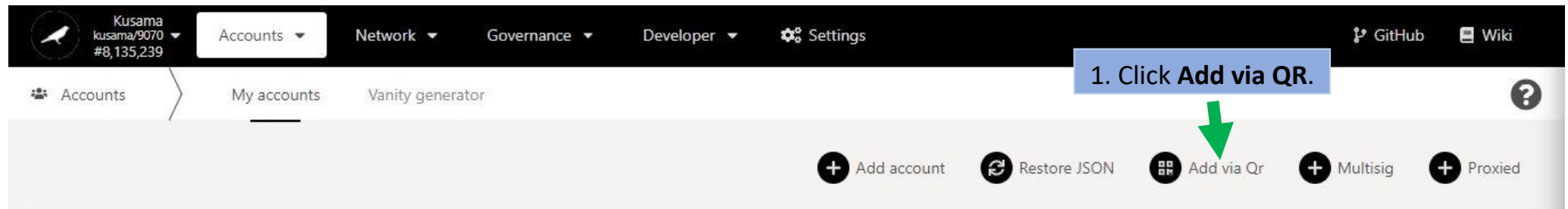
Version 1.1

The screenshot shows the 'My accounts' section of the Polkadot-JS Apps interface. At the top, a green notification banner states 'restore LALA account restored'. Below the navigation bar, there are buttons for 'Add account', 'Restore JSON', 'Add via Qr', 'Multisig', and 'Proxied'. A search bar labeled 'filter by name or tags' is present. The main content is a table of accounts:

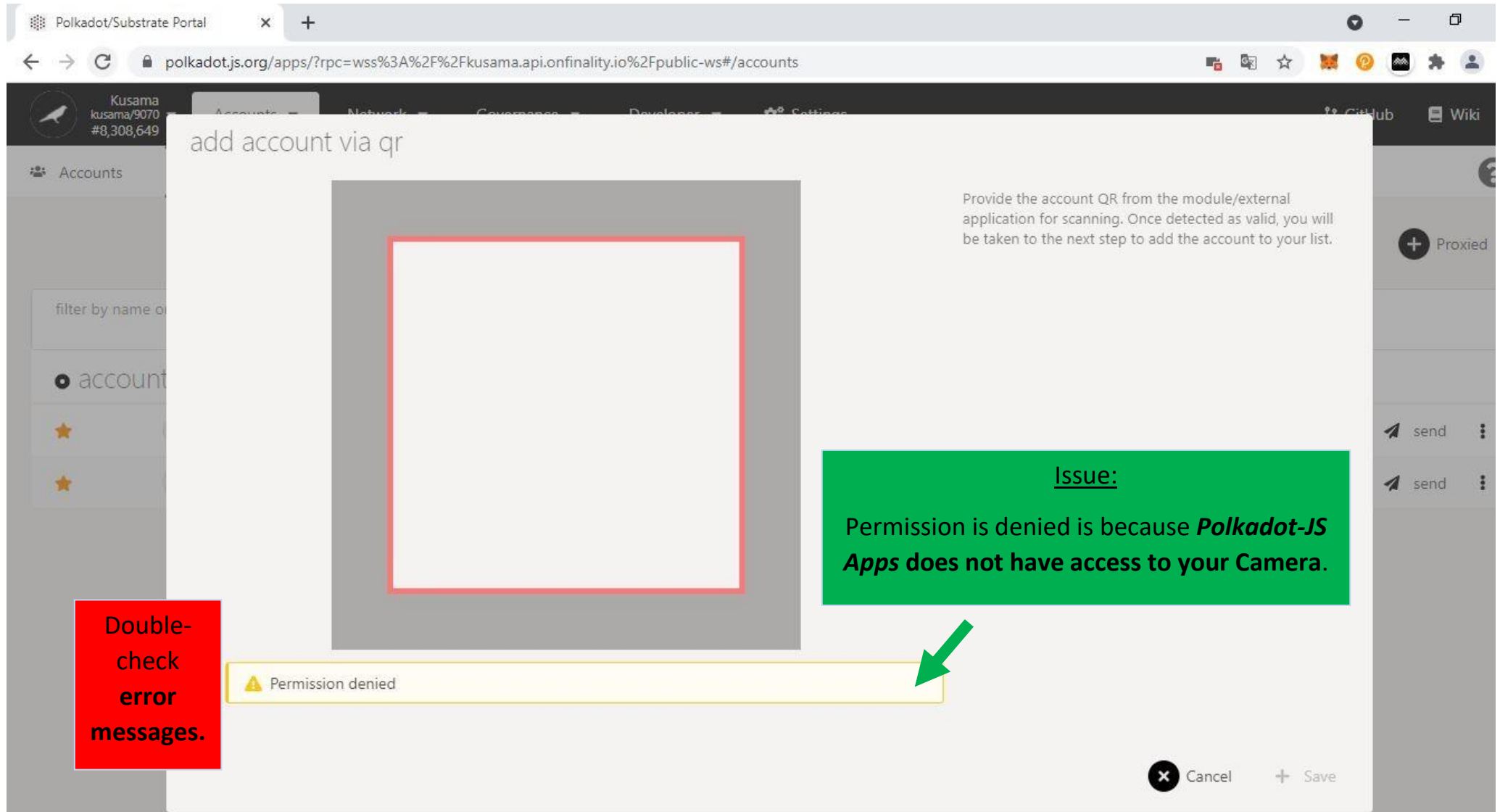
			type	tags	balances	
★	ALTD@KSM (EXTENSION)		injected	no tags	10.3768 KSM	send
★	ANAELLE LTD@KSM		injected	no tags	1.0376 KSM	send
★	LALA	sr25519		no tags	0.0000 KSM	send
					11.4144 KSM	

5. Your account has been restored in the *My accounts* list!

c) Add a Polkadot-JS account from a QR Code.



[Troubleshooting 1/3]



[Troubleshooting 2/3]

The screenshot shows the Polkadot/Substrate Portal interface. The browser address bar displays the URL: `polkadot.js.org/apps/?rpc=wss%3A%2F%2Fkusama.api.onfinality.io%2Fpublic-w`. The top navigation bar includes links for Accounts, Network, Governance, and Developer. The 'Accounts' section is active, showing a list of accounts. The 'My accounts' link is circled in green. A table of accounts is visible, with columns for 'accounts', 'type', and 'tags'. The table contains two entries: 'ALTD@KSM (EXTENSION)' and 'ANAELLE LTD@KSM', both marked as 'injected' with 'no tags'. Overlaid on the screenshot are four blue callout boxes with instructions: 1. Click the Polkadot-JS extension icon. 2. Click the Polkadot-JS extension settings icon. 3. Tick the QR Camera Access box and return to Polkadot-JS Apps. 4. Click on My accounts to refresh your browser, then restart the procedure. A green box with the text 'Possible solution: 1. Make sure that your Polkadot-JS extension settings allow QR Camera Access.' is also present. The Polkadot-JS extension settings menu is open, showing options for Theme (Light/Dark), Manage Website Access, External Accounts and Access (with 'Allow QR Camera Access' checked), Language (English), and an option to Open extension in new window.

1. Click the **Polkadot-JS extension icon**.

2. Click the Polkadot-JS extension **settings icon**.

Possible solution:

1. Make sure that your Polkadot-JS extension settings allow QR Camera Access.

3. Tick the **QR Camera Access box** and return to Polkadot-JS Apps.

4. Click on **My accounts** to refresh your browser, then restart the procedure.

[Troubleshooting 3/3]

1. Click the **Chrome browser Camera icon**.

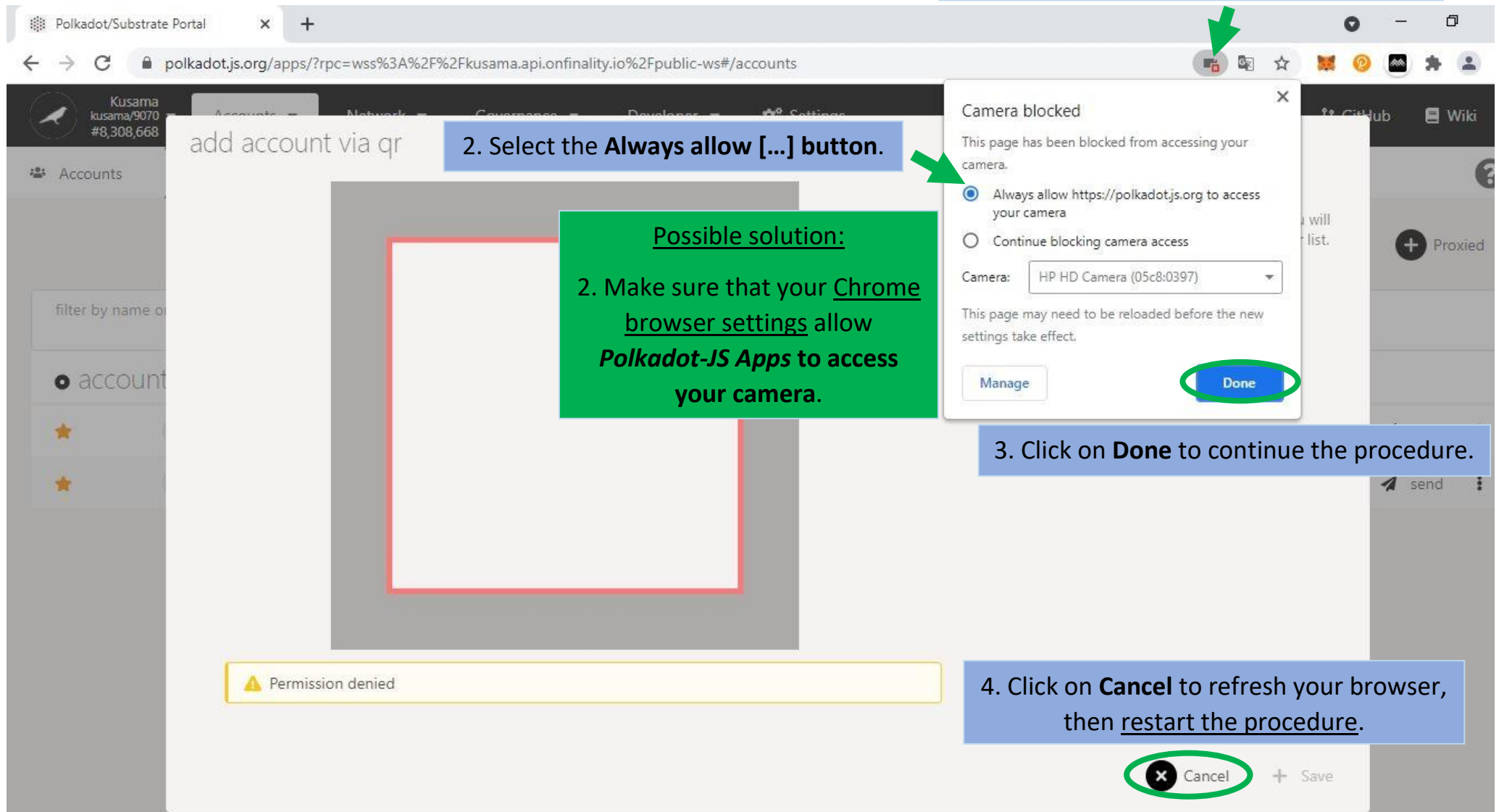
2. Select the **Always allow [...]** button.

Possible solution:

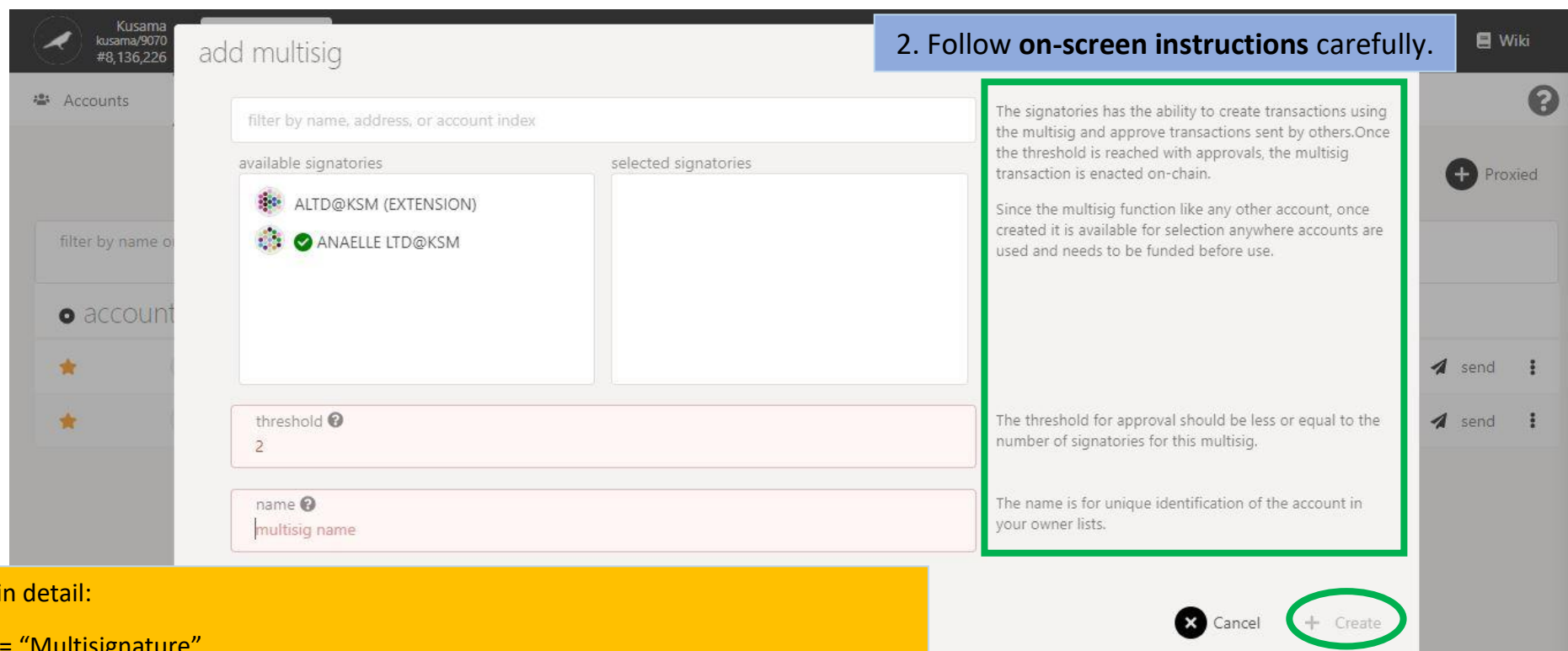
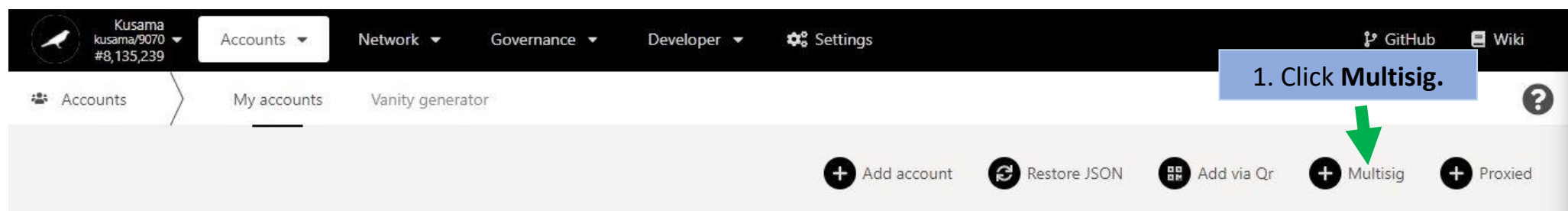
2. Make sure that your Chrome browser settings allow **Polkadot-JS Apps** to access your camera.

3. Click on **Done** to continue the procedure.

4. Click on **Cancel** to refresh your browser, then restart the procedure.



d) Set up a Polkadot-JS Multisig account.



Multisig in detail:

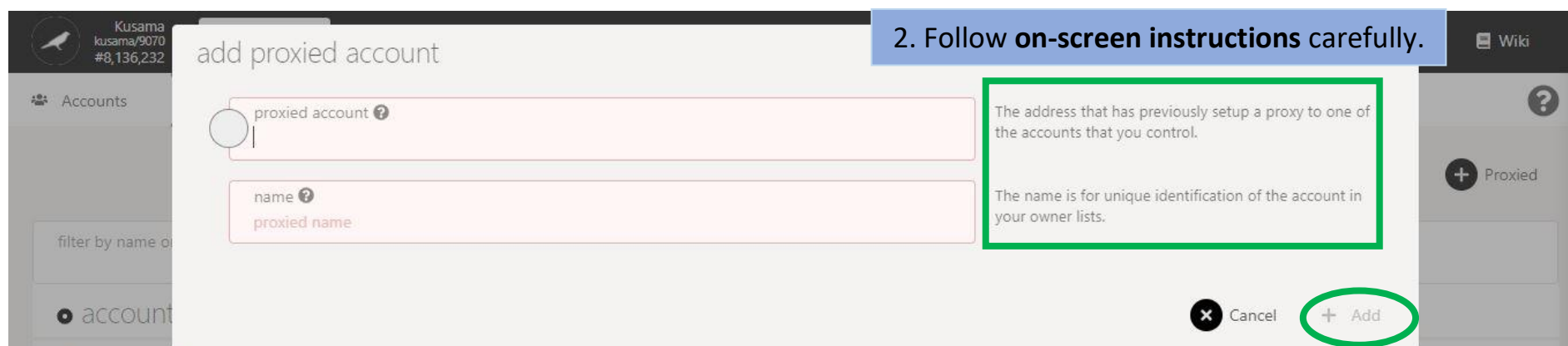
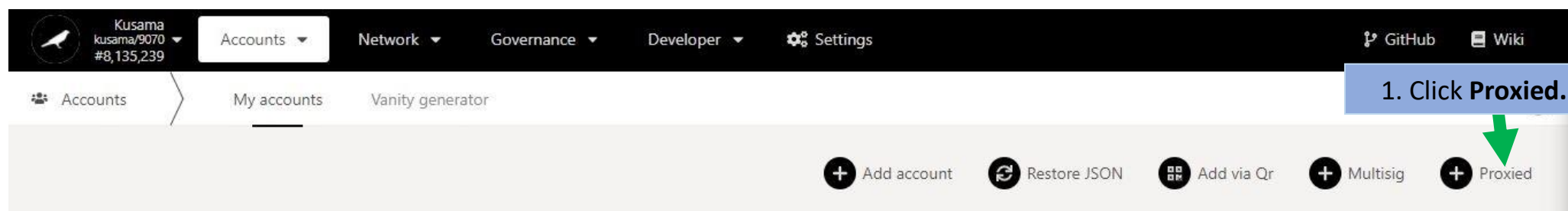
Multisig = "Multisignature".

Multisig accounts allow collectives, associations, organisations, and institutions to **manage transactions for a single account** without relying on one person.

Any account that is added as a signatory can submit/sign transactions; but the transactions will not be submitted on-chain until the **threshold** of signatories is reached.

3. Click on **Create** to complete the procedure.

e) Set up a Polkadot-JS Proxied account.



Proxy & Proxied accounts in detail:

Proxied = “Cared for”.

Proxy = “Caretaker”.

Proxy & Proxied accounts allow individuals to **manage transactions for a family member or a peer** if the person becomes unavailable or incapacitated.

The family member/peer must first **set up your account as their proxy account** before you can **add their account as your proxied account**.

To learn how to set up a proxy account for your account, refer to **Part VI: Accounts (Advanced setups)**.

f) Check your account information, receive funds at your KSM address and view your KSM balances.

The screenshot shows the 'Accounts' section of the Polkadot-JS Apps interface. The top navigation bar includes 'Kusama kusama/9070 #8,135,239', 'Accounts', 'Network', 'Governance', 'Developer', and 'Settings'. Below the navigation bar, there are tabs for 'Accounts', 'My accounts', and 'Vanity generator'. The 'My accounts' tab is active, showing a list of accounts. The list has columns for 'accounts', 'type', 'tags', and 'balances'. Two accounts are listed: 'ALTD@KSM (EXTENSION)' and 'ANAELLE LTD@KSM'. The 'balances' column shows '10.3509 KSM' and '1.0376 KSM' respectively. There are three annotations with green arrows: 1. Points to the account name 'ALTD@KSM (EXTENSION)'. 2. Points to the identicon of 'ANAELLE LTD@KSM'. 3. Points to the dropdown arrow next to the balance '10.3509 KSM'.

1. Click on the **account name** to view its information.

2. Click on the **identicon** to copy your KSM address.

3. Click on the **dropdown arrow** to view your detailed KSM balances.

accounts	type	tags	balances
ALTD@KSM (EXTENSION)	injected	no tags	10.3509 KSM
ANAELLE LTD@KSM	injected	no tags	1.0376 KSM

GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1

Kusama
kusama/9070
#8,136,085

Accounts

Network

Governance

Developer

Settings

GitHub

Wiki

AccountsMy accountsVanity generator

+ Add account

Restore JSON

Add via Qr

+ Multisig

+ Proxied


filter by name or tags

accounts

type

tags


★

 ALTD@KSM (EXTENSION)

injected

no tags

★

 ANAELLE LTD@KSM

injected

no tags

KSM balance in detail:

Transferrable = KSM existential deposit (0.000034 KSM) + available KSM.

Locked = KSM not currently transferrable because it is bonded or vested.

Bonded = KSM locked for staking nominations. Must be unbonded over 7 days to become transferrable.

Democracy = KSM locked for voting. Must be unlocked at the end of the voting period to become transferrable. Note: If a voluntary locking period of 8 days has been set during the last vote, the voting balance will be unlocked after 8 days.

Reserved = KSM locked for submitting council candidatures, treasury proposals, tips, on-chain identities, etc. Must be cleared to become transferrable.

balances

10.3509 KSM

transferrable 0.3427 KSM

locked 10.0082 KSM

bonded 10.0082 KSM

democracy 9.8733 KSM

1.0376 KSM

transferrable 1.0043 KSM

reserved 0.0333 KSM

11.3886 KSM

send

g) Generate a vanity address.

The screenshot shows the 'Accounts' section of the Polkadot.js interface. The top navigation bar includes 'Kusama', 'kusama/9070', '#8,135,263', 'Accounts', 'Network', 'Governance', 'Developer', 'Settings', 'GitHub', and 'Wiki'. Below the navigation bar, the 'Accounts' tab is active, showing 'My accounts' and 'Vanity generator'. A green arrow points to the 'Vanity generator' tab, with a blue box containing the text '1. Click Vanity generator.' To the right of the 'Vanity generator' tab is a help icon (?). The main content area is a form with three input fields: 'Search for' (containing 'Kusama'), 'keypair crypto type' (containing 'Schnorrkel (sr25519, recommended)'), and 'case sensitive' (containing 'Yes'). A green box highlights these three input fields, with a blue box containing the text '2. Fill-in or select the information required.' to the left. At the bottom right of the form is a 'Start generation' button, which is circled in green. A blue box containing the text '3. Click on Start generation to complete the procedure.' is positioned to the right of the button.

Kusama
kusama/9070
#8,135,263

Accounts Network Governance Developer Settings

GitHub Wiki

Accounts My accounts Vanity generator

1. Click Vanity generator.

Search for ?
Kusama

keypair crypto type ?
Schnorrkel (sr25519, recommended)

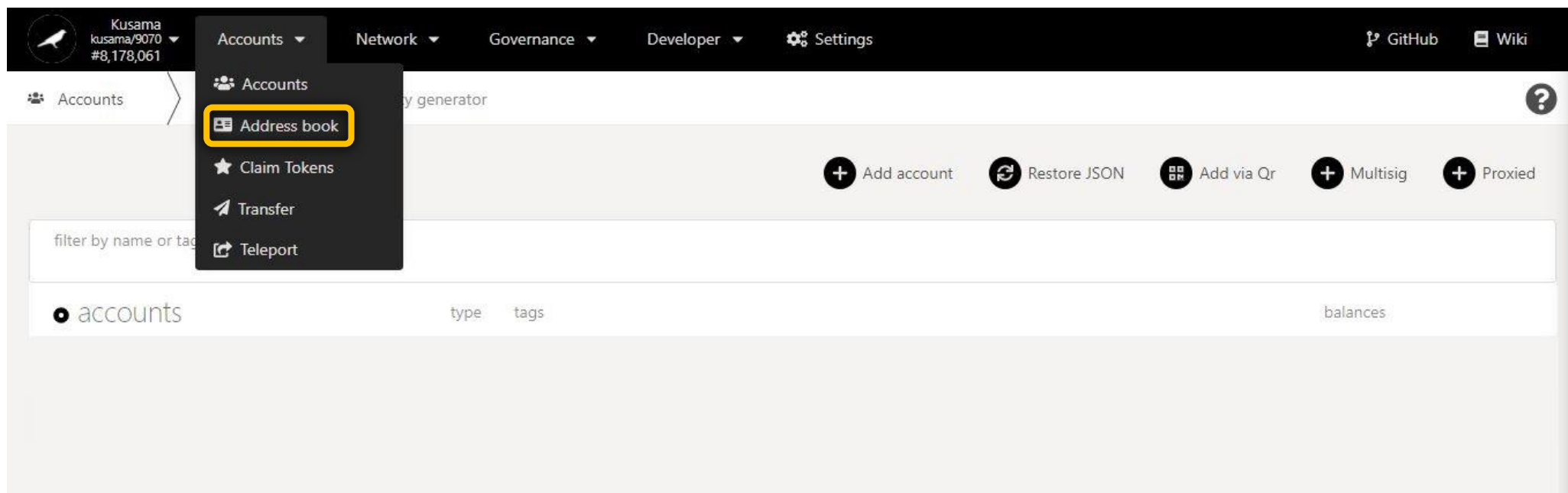
case sensitive ?
Yes

2. Fill-in or select the information required.

Start generation

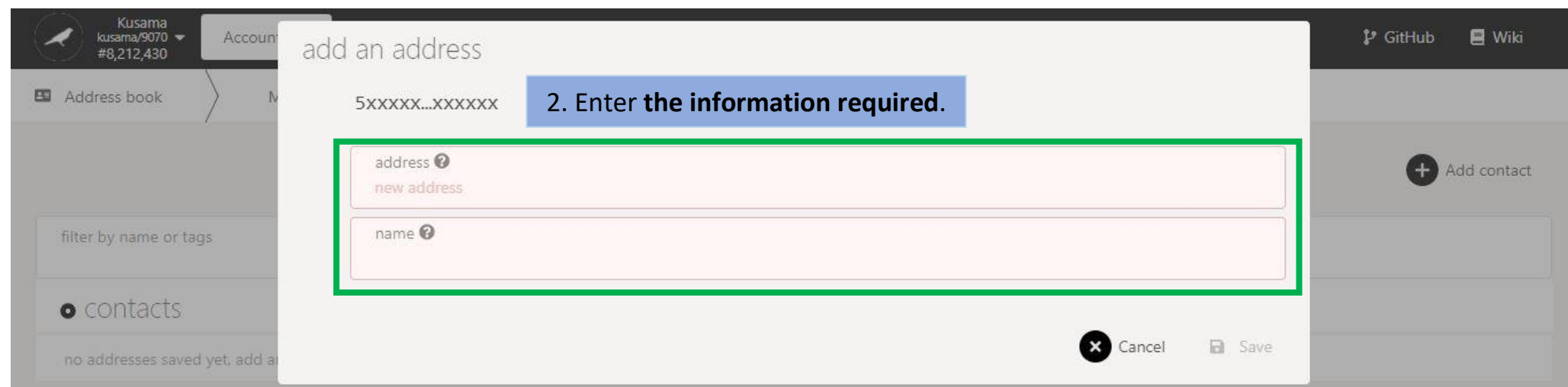
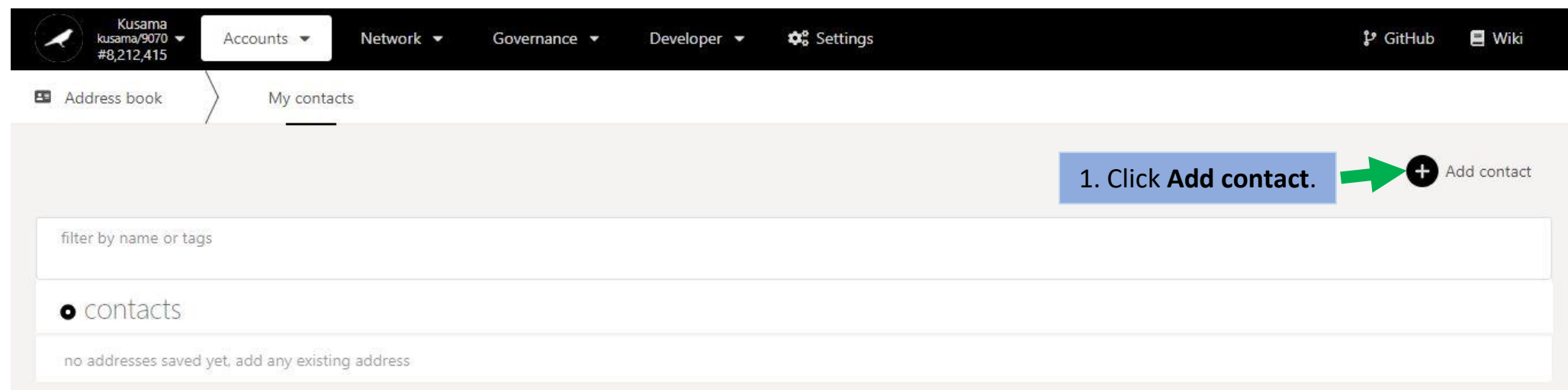
3. Click on Start generation to complete the procedure.

2. Address book: Bookmark your favourite KSM addresses as contacts.



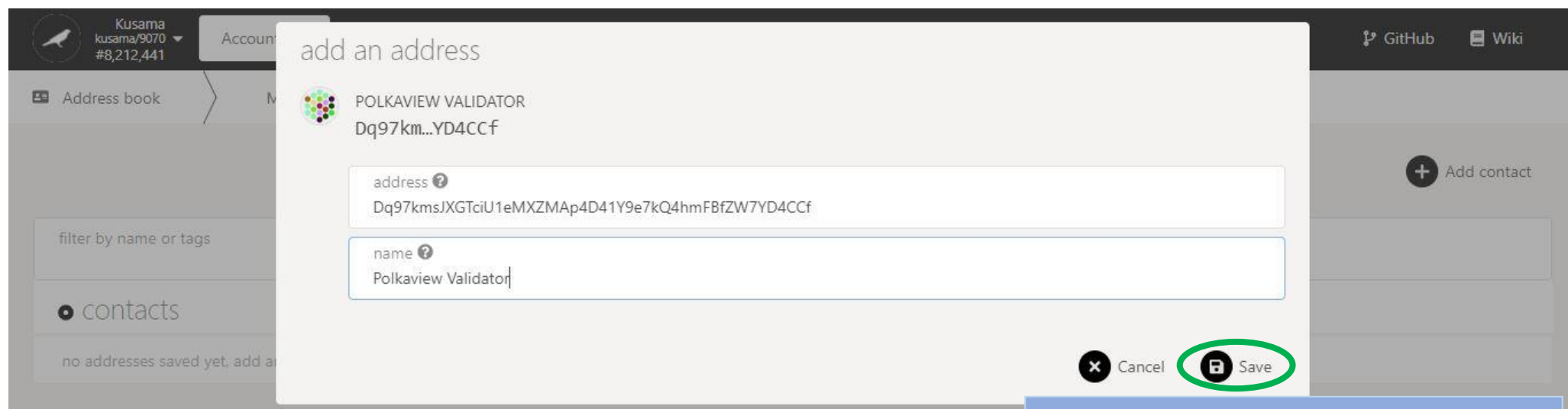
GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1

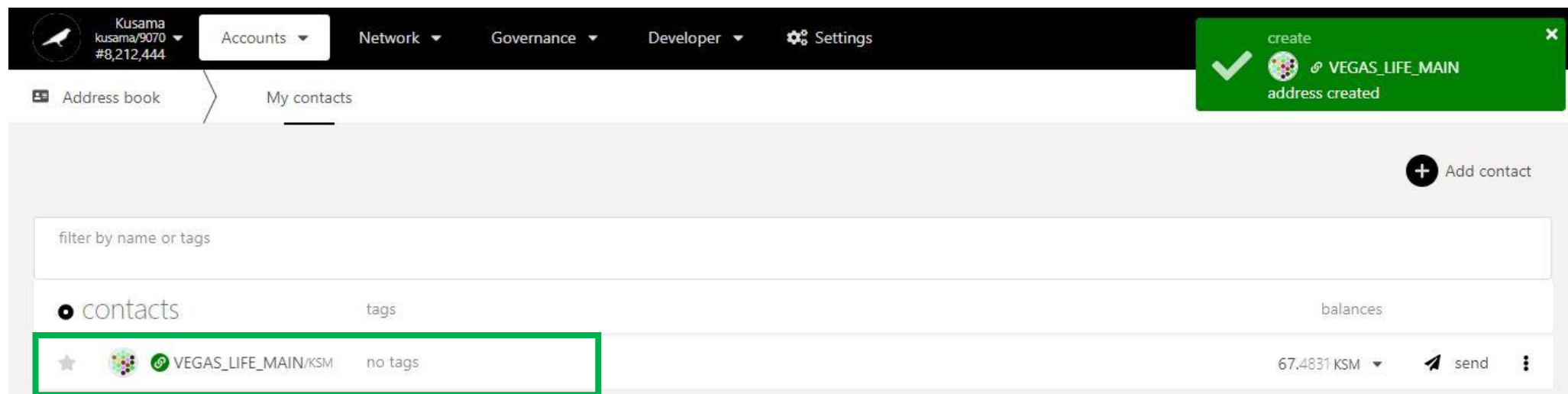


GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1



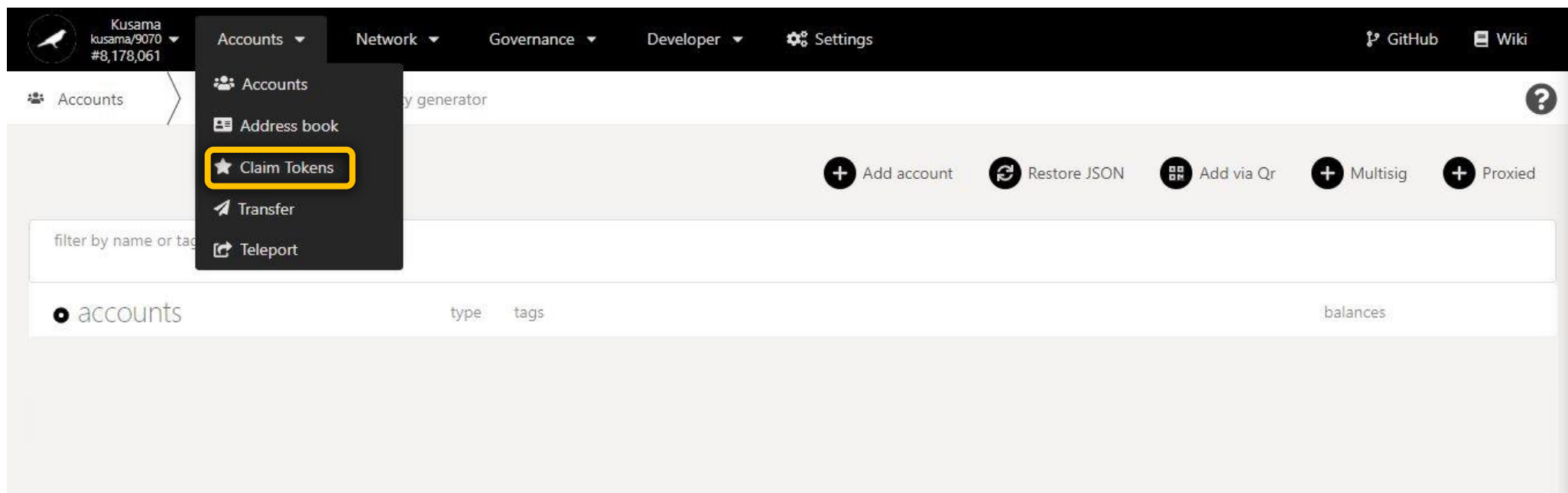
3. Click on **Save** to complete the procedure.



4. Your contact has been added in the *My contacts* list!


Note: If your contact has an identity, it will automatically replace the name you provided.

3. Claim tokens: Claim your KSM from your ETH address.



GUIDE TO POLKADOT-JS APPS – PART I: Accounts (Basic setups)

Version 1.1




Kusama
kusama/9070
#8,135,295


Accounts ▾


Network ▾

Governance ▾

Developer ▾

 Settings

 GitHub


 Wiki

★ Claim Tokens

Claim tokens

claim your KSM tokens

1. Select your Kusama account.


 claim to account ?

✓ ANAELLE LTD@KSM

HjcErRijmpoiBiKEHT3edPXM3NFycJogVwDP...

2. Enter the ETH address from the sale.

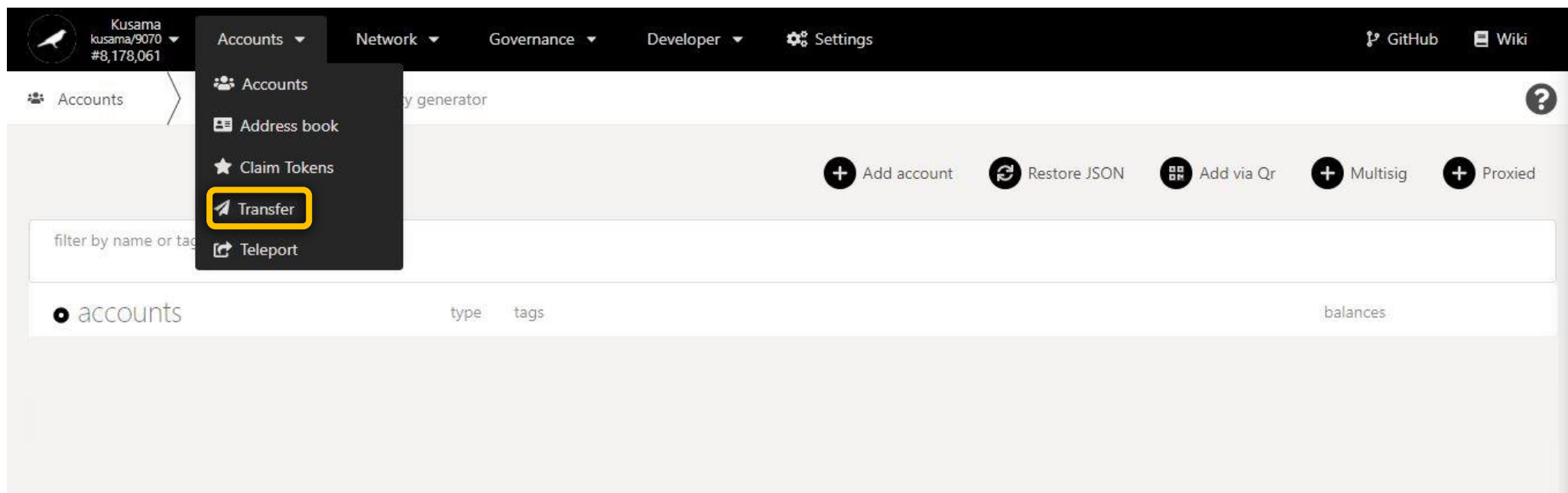
Pre-sale ethereum address ?

 Continue

1. Enter the information required.

2. Click on **Continue** to view and complete subsequent steps in the procedure.

4. Transfer: Send KSM to another KSM address.



The receiving account must be an account from **your My Accounts or My contacts list**.

The screenshot shows the 'send funds' modal in the Polkadot.js app. It includes two options for sending: 'send from account' (selected) and 'send to address'. The 'send from account' option shows a balance of 1.0043 KSM. The 'send to address' option shows a balance of 0.3427 KSM. The amount field is set to 0, and the existential deposit is 33.3333 micro. A toggle for 'Transfer with account keep-alive checks' is currently turned on. The 'Make Transfer' button is circled in green. A green arrow points from the 'send from account' option to the 'ANAELLE LTD@KSM' account. Another green arrow points from the 'Transfer with account keep-alive checks' toggle to the explanatory text box.

1. Follow on-screen instructions carefully.

The transferred balance will be subtracted (along with fees) from the sender account.

The beneficiary will have access to the transferred fees when the transaction is included in a block.

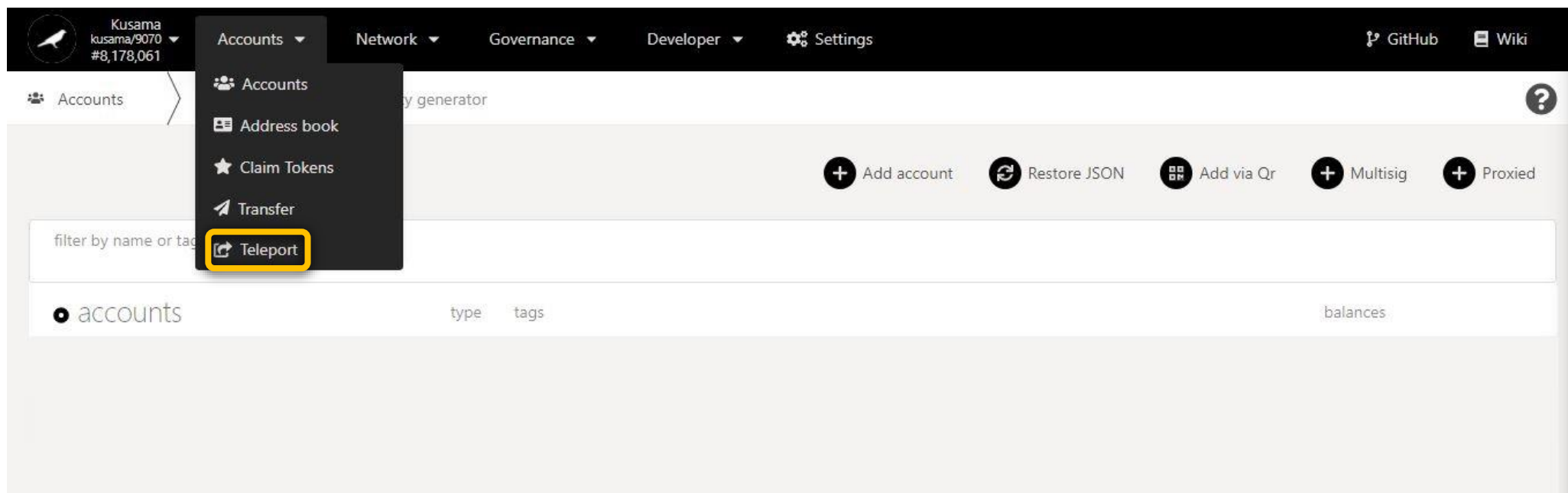
If the recipient account is new, the balance needs to be more than the existential deposit. Likewise if the sending account balance drops below the same value, the account will be removed from the state.

Transfer with account keep-alive checks ☒ With the keep-alive option set, the account is protected against removal due to low balances.

2. Click on **Make Transfer** to complete the procedure.

If you want to **empty the account completely**, switch the keep-alive checks **OFF**. The existential deposit will be transferred, and the account will be deactivated.

5. Teleport: Send your KSM to a parachain address.



Always double-check the **parachain** information.

teleport assets

send from account: ANAELLE LTD@KSM (transferrable 1.0043 KSM)

destination chain: Statemine

send to address: ALTD@KSM (EXTENSION)

amount: 1 KSM

destination transfer fee: 7.9999 micro

destination existential deposit: 3.3333 micro

1. Follow on-screen instructions carefully.

The transferred balance will be subtracted (along with fees) from the sender account.

The destination chain for this asset teleport. The transferred value will appear on this chain.

The beneficiary will have access to the transferred amount when the transaction is included in a block.

If the recipient account is new, the balance needs to be more than the existential deposit on the recipient chain.

The amount deposited to the recipient will be net the calculated cross-chain fee.

Cancel Teleport

2. Click on **Teleport** to complete the procedure.

Always double-check the **existential deposit** information.