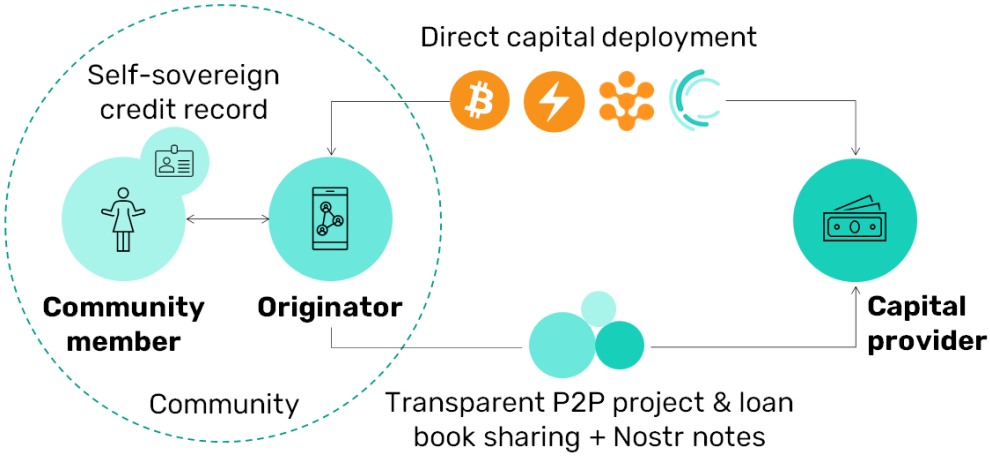
OVERVIEW

Overview

Growr protocol is an open-source peer-to-peer protocol that enables access to fair financing. The protocol helps micro-entrepreneurs from local communities to receive instant productive loans based on their self-sovereign credit record without the need for over-collateralization. In contrast to traditional microfinance, the protocol creates an open and global lending marketplace that connects borrowers to both traditional and decentralized capital providers, enforcing fair competition for pricing and fully transparent deployment of capital down to each micro-loan.

The following diagram provides a high-level overview of the Growr protocol.



Originators manage the “last-mile” processes through their Growr node. They help community members to build their self-sovereign credit record and facilitate access to the protocol services & project funding for them.

Capital providers fund projects either by accessing the Growr node of an originator directly; or by running their own node. The P2P data exchange enables real-time reporting of the project portfolio for full transparency.

Currently, we have designed 4 main groups of protocol processes:

- Setup
- Lending
- Funding
- Crowdfunding

Participants

Borrowers

Borrowers, represented by self-employed, micro-businesses, and smallholder farmers, apply for productive loans from the marketplace, most often with the help of a local originator, and then repay the loan plus its price.

Originators

Originators facilitate access to the protocol by grouping several borrowers with similar needs and presenting project applications to the marketplace on their behalf. They publish offers to the loan marketplace by creating *projects* on behalf of the borrowers. The originators provide junior (first-loss) capital to finance the whole project or part of it. They can be:

- *Local cooperatives, guilds or other community organizations* formed by borrowers to gain better access to loans and to standardize their relationship with the rest of the participants in the ecosystem.
- *Bitcoin circular economies* using solutions such as Federated Chaumian mints, enabling access to micro-financing to their users.
- *Telcos, retailers and gig-economy platforms* that onboard and vet the users into their services and then facilitate their access to the protocol as an embedded financial service.
- *Digital wallets and fintech providers* that already offer financial services and that can expand to unsecured decentralized lending.

Trusted parties

Trusted parties assert facts about the borrowers in the form of verifiable credentials. They can be:

- The *originator*, issuing credentials for its members or users.
- *Merchants, buyers, unions, chambers or other local organizations* that serve the community or have knowledge of their members.
- *Independent third-party data providers* that can issue credentials related to the activity of the borrower and relevant to the risk assessment process, such as KYC/AML, account data, and on-chain activity.
- *Financial health providers* that publish educational materials and tools to help borrowers develop good financial habits and issue credentials that assert knowledge, skills, and accomplishments.

Capital providers

Capital providers allocate senior capital to the loan marketplace and delegate the actual lending activity to the originators in the form of senior tranches for the financed projects. They can be:

- Individual Bitcoin investors.
- Large institutional investors or digital asset managers.

- Decentralized finance protocols.

Growr components

The following section contains a list of the main Growr components, participating in the lending and funding processes of the protocol. For more details, see Architecture components.

Front-end applications:

- Node operator portal
- Lending portal
- Investor portal
- Borrowing applications

Identity services:

- Organization service
- User service
- Identity service (SSCR)
- Credential issuing service
- Credential verification service

Financing services:

- Project service
- Loan service
- Project investor service
- Funding service

Payment services:

- Payment service
- Bitcoin service
- Lightning service
- Rootstock service

Publishing services:

- Nostr service

SETUP OVERVIEW

Overview

Below is a diagram that summarizes the setup processes:

In progress...

The next sections describe the following processes:

- Growr node deployment
- User setup
- Project setup

Growr node deployment

There are 2 main options for Growr node deployment:

- Self-hosted by originators or capital providers
- Managed by the Growr team

Self-hosting

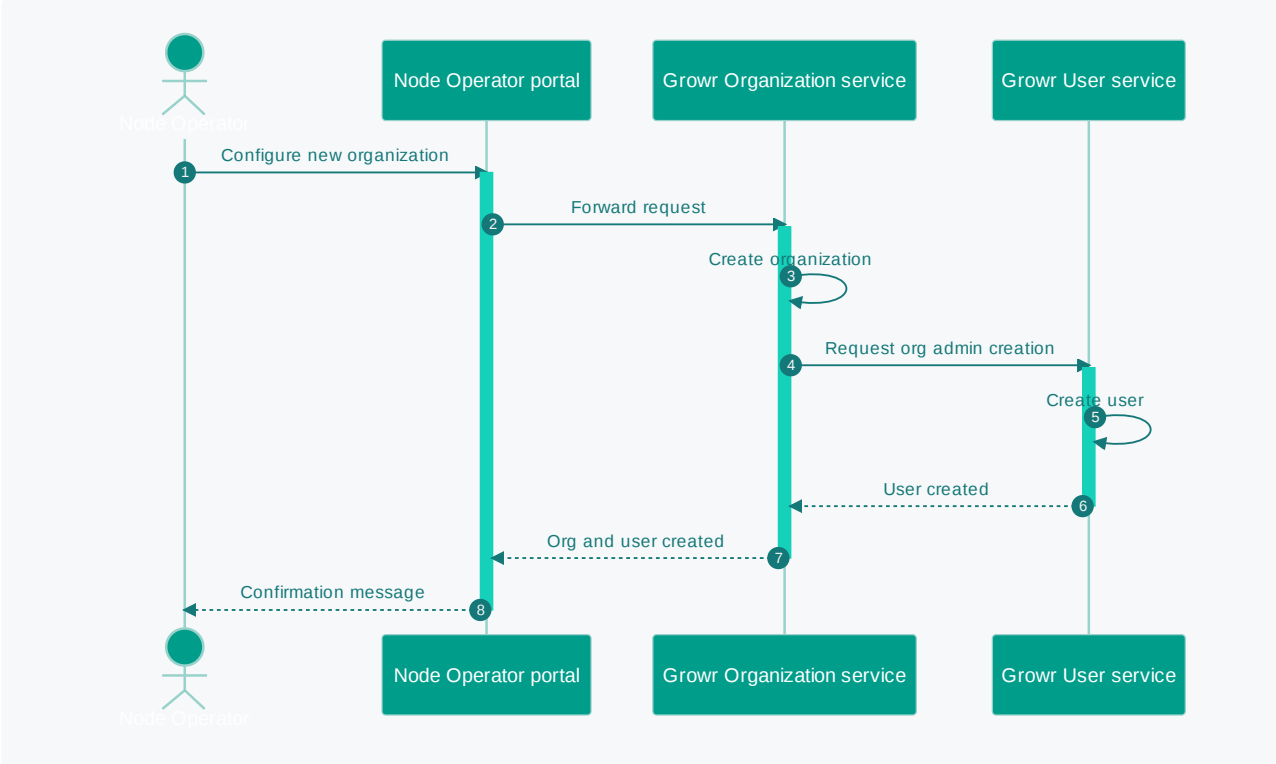
In progress

Managed service

In progress

Organization setup

Once the Growr node is deployed, the first thing to do is to create an organization with one initial user.



Process steps:

1. Growr node operator setups a new organization through the Node Operator portal.
2. The portal routes the request to the Growr Organization service.
3. Growr Organization service creates a new record in the Organizations collection.
4. Growr Organization service sends a request to the Growr User service to create an initial admin user for the new organization.
5. Growr User service creates a new record in the Users collection.

User setup

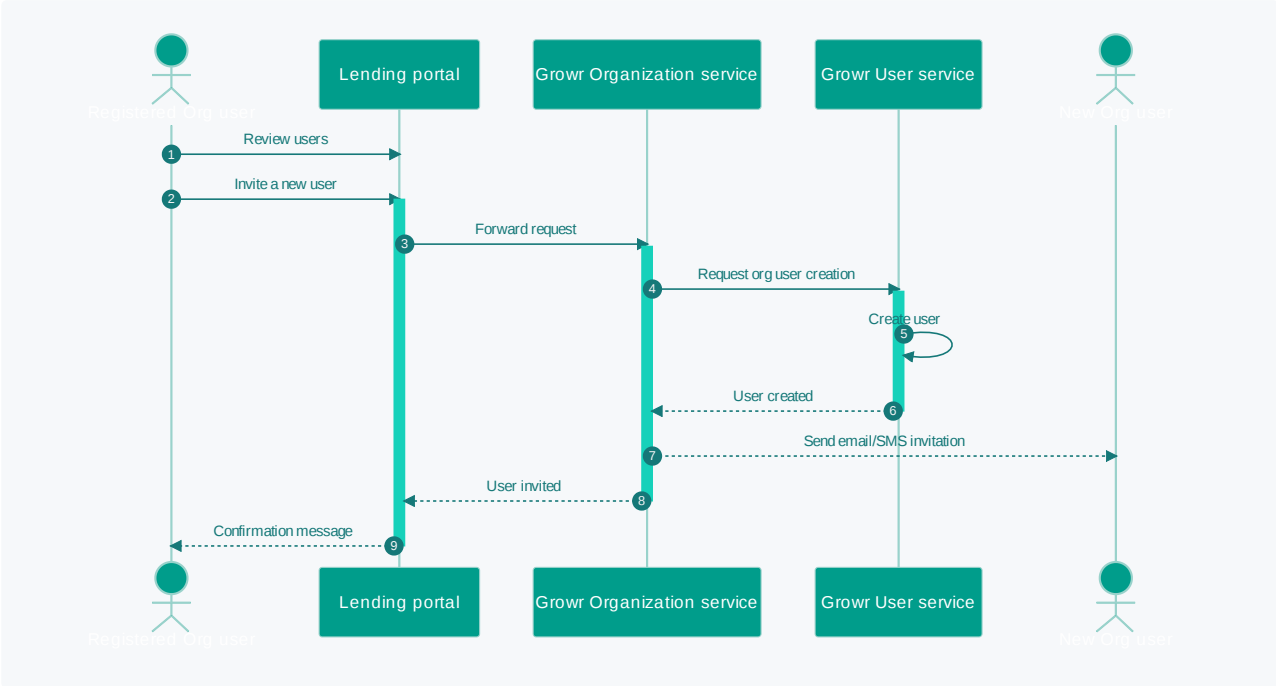
Currently, the Growr User service supports 5 types of users:

- *Node operator (super-admin):* manages the Growr node configuration.
- *Originator admin:* manages projects, lending and funding parameters, and monitors the performance.
- *Originator agent:* manages operations with the borrowers.
- *Investor:* discovers and invests in projects.
- *Borrower:* onboards and subscribes to lending services.

User invitation

In order for a user to be registered, one of the following prerequisites must be met:

- The user is created by the Node operator during organization setup.
- The user is invited by an originator admin.



Process steps:

1. A registered user reviews the list of users in the Lending portal.
2. The user configures a new user.
3. The portal routes the request to the Growr Organization service.
4. Growr Organization service sends a request to the Growr User service to create a new user related to the organization.
5. Growr User service creates a new record in the Users collection.
6. Response message.
7. Growr Organization service sends an invitation email with a link to the Lending portal and (optional) OTP code.

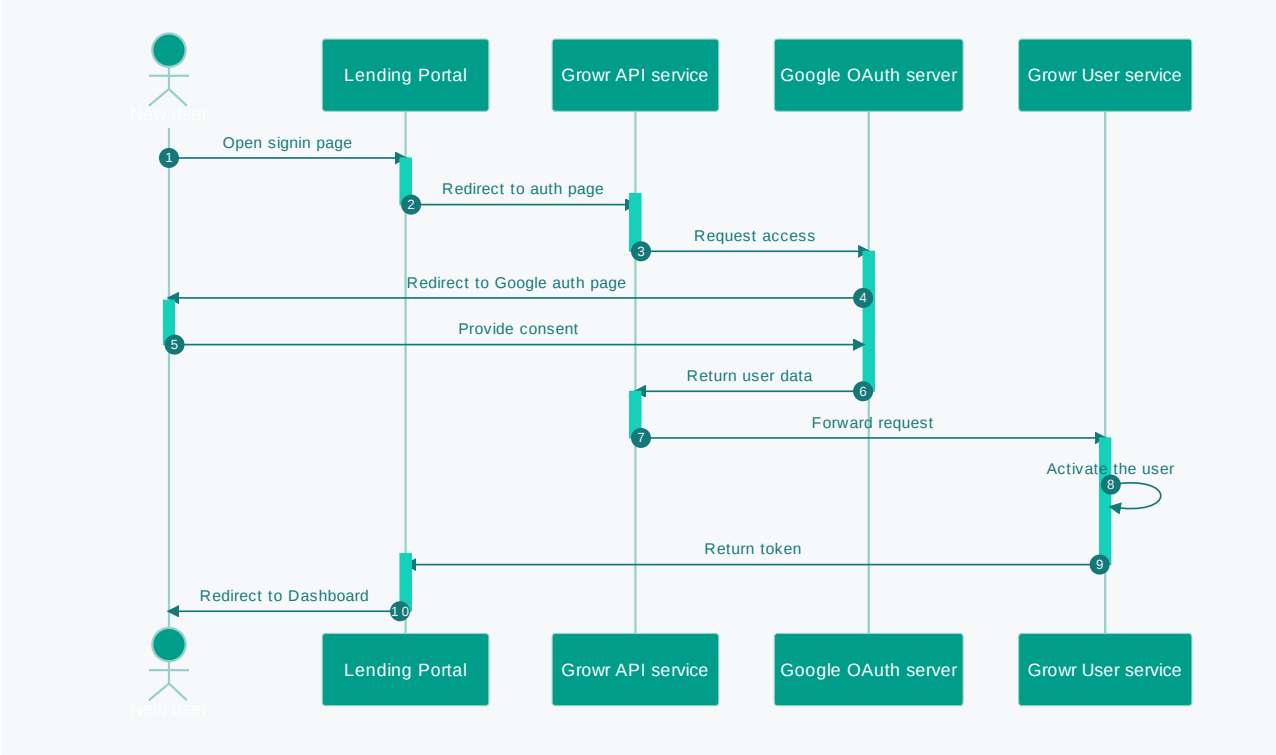
User registration

Currently, the Growr User service supports 3 types of registration:

- Social account (Google, Facebook): applicable to users with the type *Originator admin* and *Investor*
- Email + password: applicable to users with the type *Originator admin* and *Investor*
- Phone (USSD): applicable to users with the type *Originator agent* and *Borrower*

User registration with a social account

Prerequisite: The user has received an invitation email.

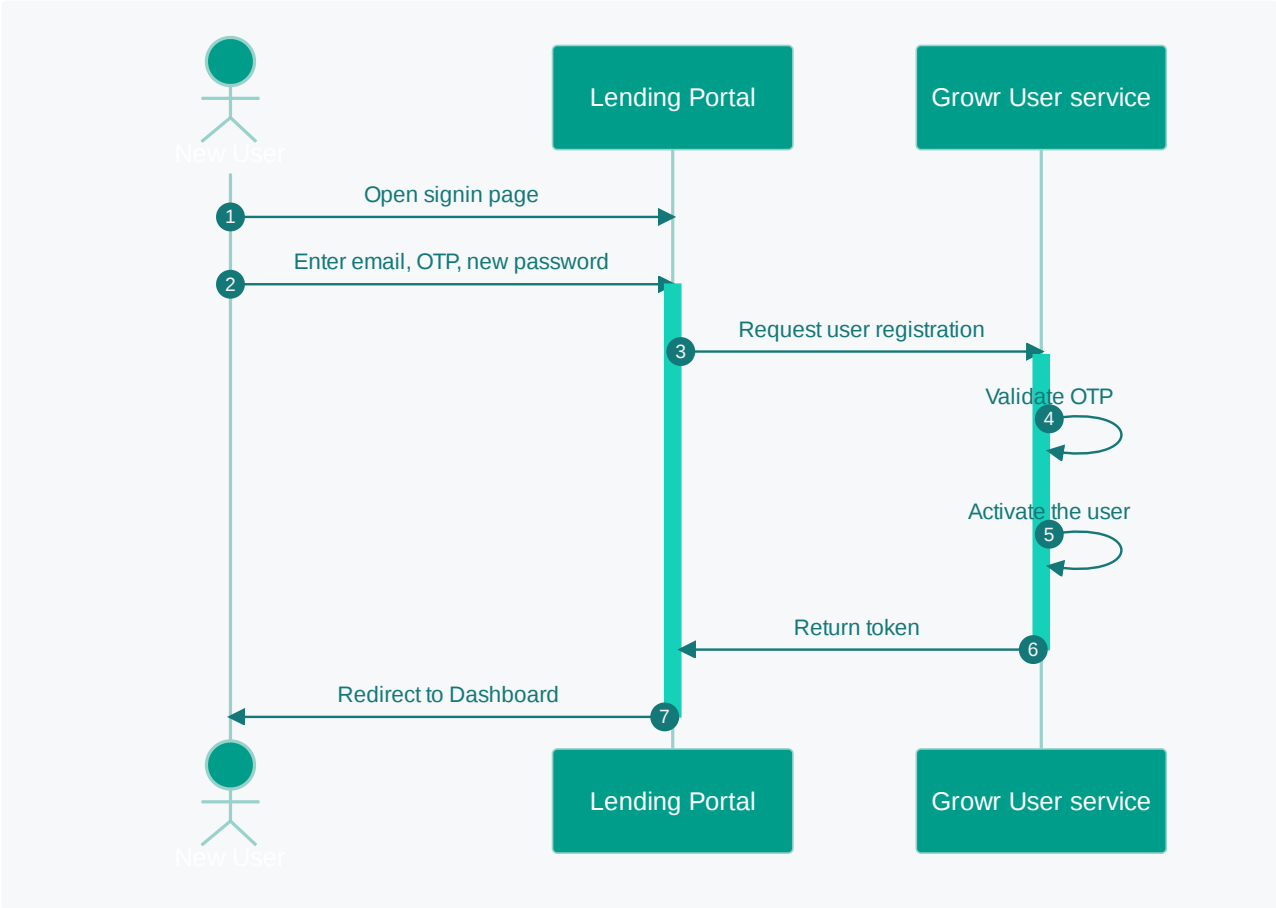


Process steps:

1. An invited user opens the Lending portal and is automatically redirected to the /signup page where the user initiates the signup process.
2. The Lending portal redirects the user to the authentication route of the Growr API service.
3. Growr API service calls the Google authentication API.
4. Google authentication server redirects the user to the authentication page.
5. The user selects one of his available Google accounts and provides the requested consent.
6. Google authentication server confirms the authentication and returns the user data to the Growr API service.
7. Growr API service redirects the request to the Growr User service.
8. Growr User service updates the record (changes the status to ACTIVE) in the Users collection.
9. Growr User service returns a token to the Lending Portal. The token will be used for all subsequent calls to Growr services.
10. The Lending portal lets the user in and redirects him/her to the main page (Dashboard).

User registration with email and password

Prerequisite: The user has received an invitation email with an OTP code.

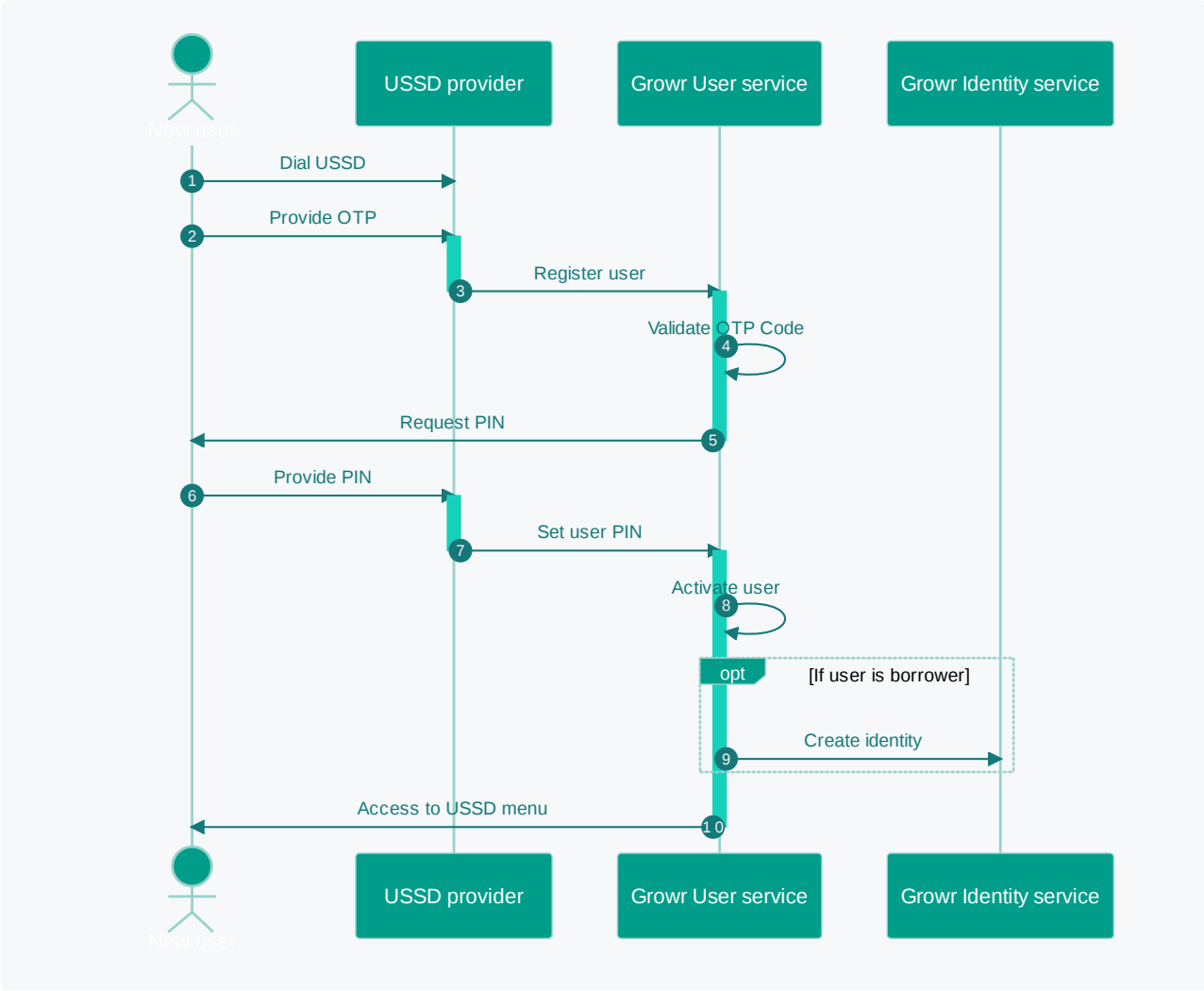


Process steps:

1. An invited user opens the Lending portal and is automatically redirected to the /signup page where the user initiates the signup process.
2. The user enters his email (where an invitation was received), the received OTP code and configures a new password.
3. The Lending portal sends a request for registration of the new user with his/her email and password.
4. Growr User service validates the provided OTP code.
5. Growr User service updates the record (changes the status to ACTIVE) in the Users collection.
6. Growr User service returns a token to the Lending Portal. The token will be used for all subsequent calls to Growr services.
7. The Lending portal lets the user in and redirects him/her to the main page (Dashboard).

User registration with phone

Prerequisite: The user has received an invitation SMS with an OTP code.

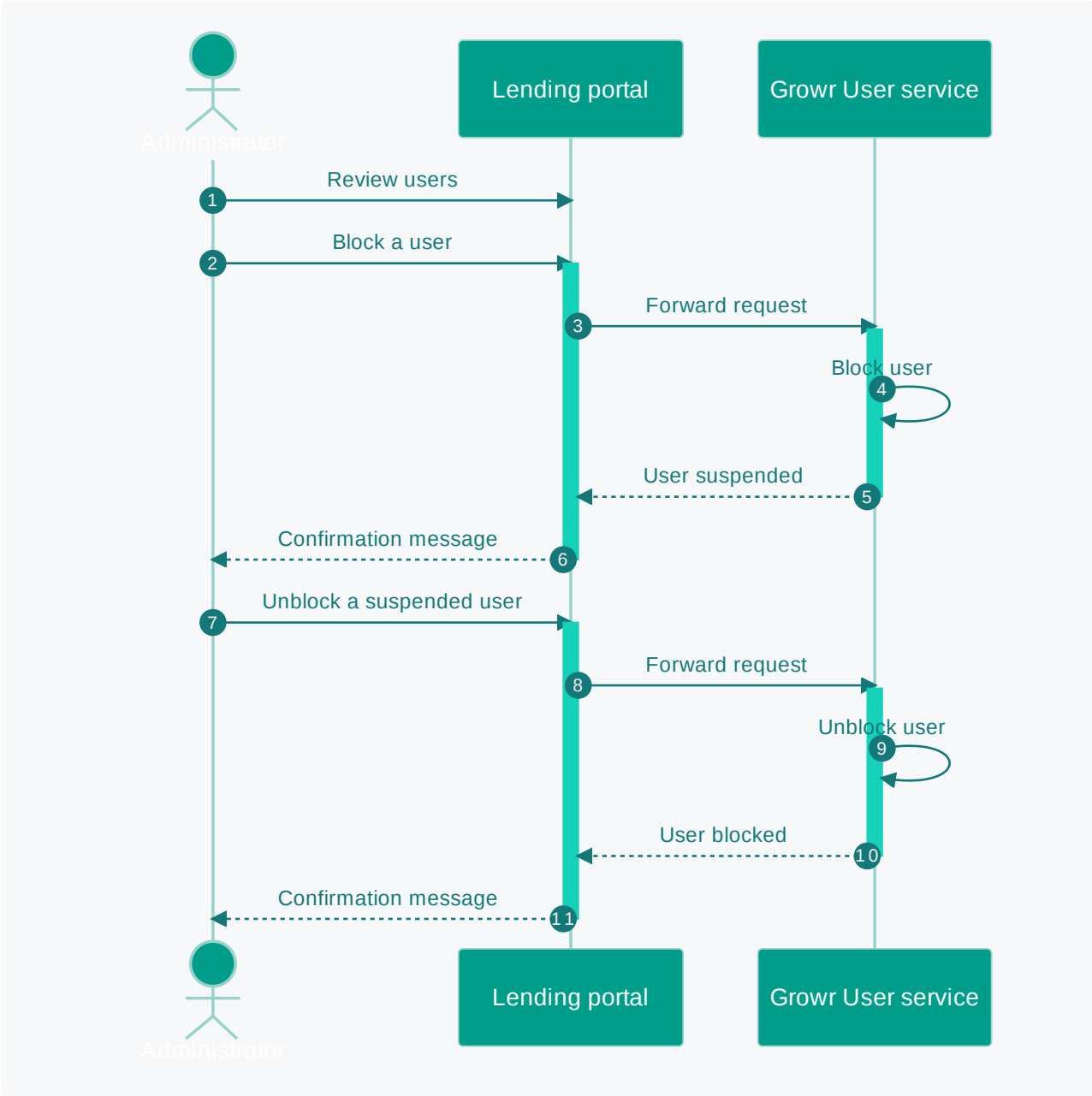


Process steps:

1. An invited user dials a number to receive a USSD menu.
2. The user provides the OTP code that was received as an SMS.
3. The USSD provider sends a request for registration of the new user with a phone number.
4. Growr User service validates the provided OTP code.
5. Growr User service requests a PIN to be provided by the user.
6. The user enters and confirms a PIN using the USSD menu.
7. The USSD provider sends a request for PIN setting to Growr User service.
8. Growr User service registers the PIN and updates the record (changes the status to ACTIVE) in the Users collection.
9. Growr User service requests from Growr Identity service to create an SSCR identifier (DID) and a private key.
10. The user is authorized to access the full USSD menu options.

Suspending users

Administrators (admins and/or super-admins) can temporarily block other users to prevent them from using the node services.



Process steps:

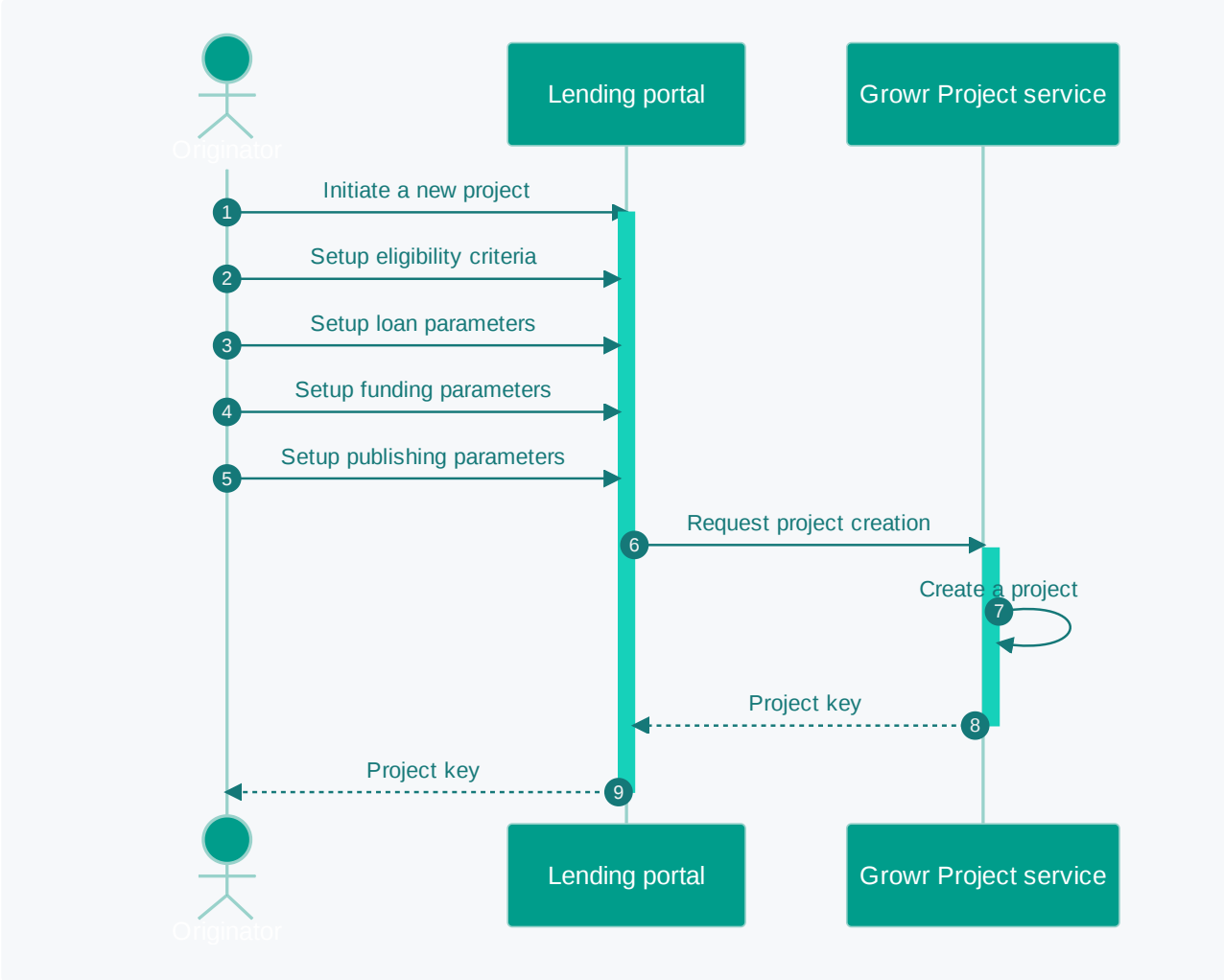
1. An administrator reviews the list of users registered in the node.
2. The user selects a given user from the list and requests its suspension.
3. The Lending portal sends the suspension request.
4. Growr User service updates the record (changes the status to BLOCKED) in the Users collection.
5. Response message.
6. Response message.

The steps for unblocking the user (i.e. removing his/her suspension) are the same.

Project setup

A key component in the Growr protocol is the lending *project*. The project represents a loan offer with predefined conditions and eligibility criteria. Every loan in the protocol is approved and disbursed through a project. In addition, the project determines the funding needs and the potential investment channels. Projects are usually initiated by originators on behalf of their borrowers.

Project creation



Process steps:

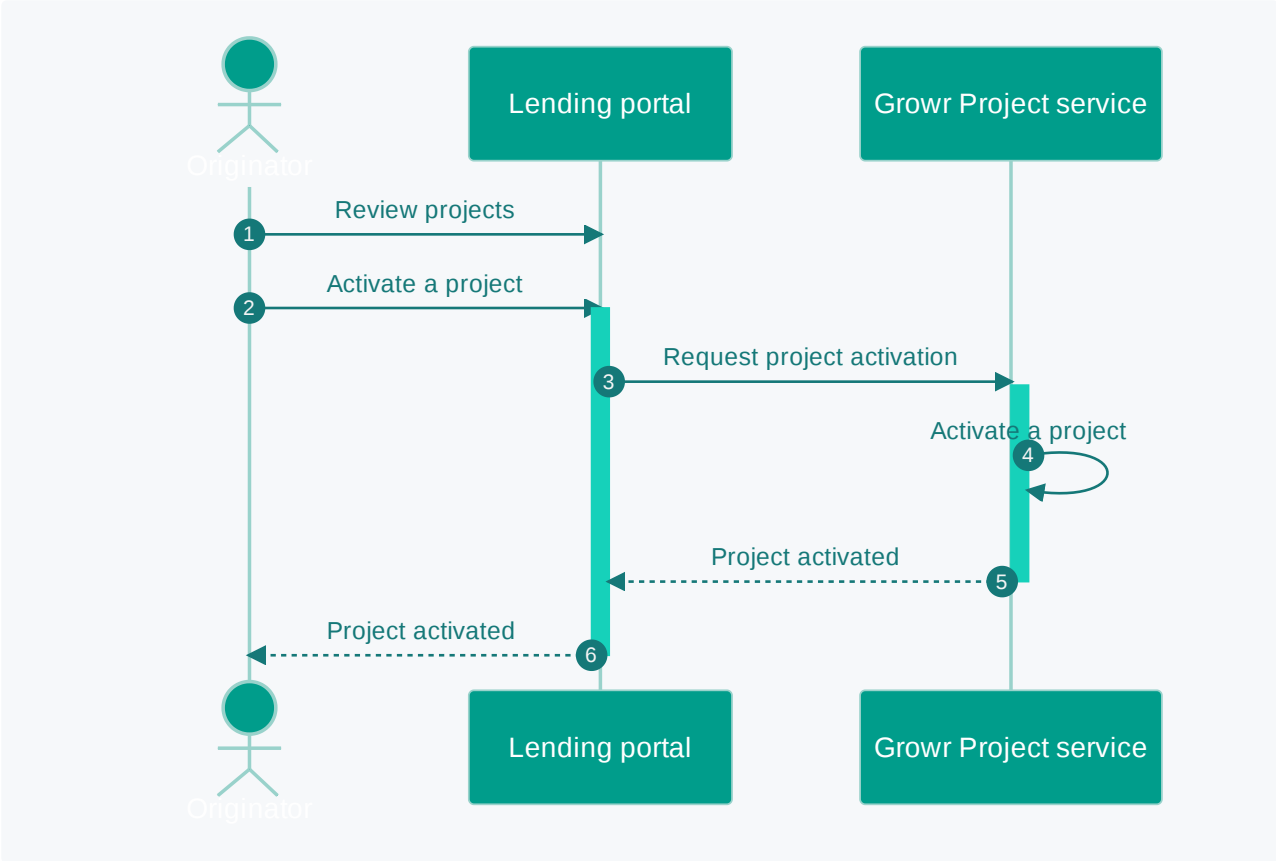
1. An originator admin creates a new project for lending service in the Lending portal.
2. The user configures the eligibility criteria for approval of loans from the project.
3. The user configures the parameters of the loan.
4. The user configures one or more funding sources for the project.
5. The user configures a description and media parameters of the project.
6. The request is transferred from the Lending portal to the Growr Project service.
7. Growr Project service creates a new project in the Projects book.
8. Response message.
9. Response message.

Project activation

Once a project is created, it must be activated to become operational. Once a project is activated, the following processes are unlocked:

- The lending service from the project becomes active i.e. new loan applications are accepted.
- The project becomes discoverable (in case the project is public).
- The project funding wallet is created and an on-chain listener is registered on the network (in case the project has one or more configured on-chain funding sources).

After a project is activated, it can be temporarily deactivated. This status pauses the lending service i.e. new loan applications will not be accepted until the project is activated again.



Process steps:

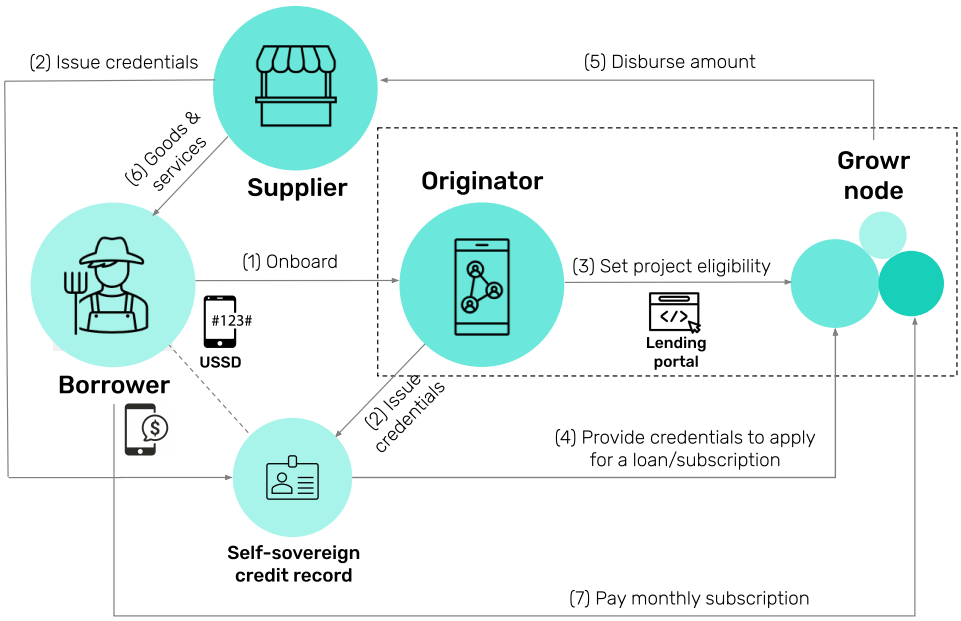
1. An originator admin reviews the projects in the Lending Portal.
2. The user selects a project in status DRAFT and activates it.
3. The request is transferred from the Lending portal to the Growr Project service.
4. Growr Project service activates the project.
5. Response message.
6. Response message.

The steps for project deactivation are the same.

LENDING OVERVIEW

Overview

Below is a diagram that summarizes the lending processes:



1. The *borrower* is onboarded on a platform, provided by an *originator* (such as a cooperative, guild, wholesale buyer, digital wallet, or another provider), where the borrower defines their funding needs.
2. One or more *trusted parties*, such as merchants of suppliers, provide credentials to the *borrower* to start building their self-sovereign credit record.
3. The *originator* creates a project with details about the local activities and the financing needs of the *borrowers*.
4. *Borrowers* go through a simple application process to receive a loan after asserting their eligibility with their verifiable credentials.
5. The disbursed amount is received by the *borrowers* in the borrowing application or directly paid to a *third party*, such as a merchant or a supplier.
6. The *third party* provides the necessary goods and services to the *borrower* to achieve their goals.
7. Depending on their business activity and the agreed lending conditions, the *borrowers* repay their loans regularly or as a bullet payment at the end of the season.

The next sections describe the following processes:

- Borrower onboarding
- Loan application
- Loan utilization
- Loan repayment

Borrower onboarding

The Growr protocol does not enforce any specific requirements upon onboarding implementation, so developers may design features and functionalities that correspond to the desired user experience.

In a custodial model, the borrowers would onboard using a mobile or web application, or even a USSD interface, provided by an originator. The originator is responsible for onboarding the users into their application and facilitating their access to the protocol. The private keys of the user DID and wallets are managed by the originator's app.

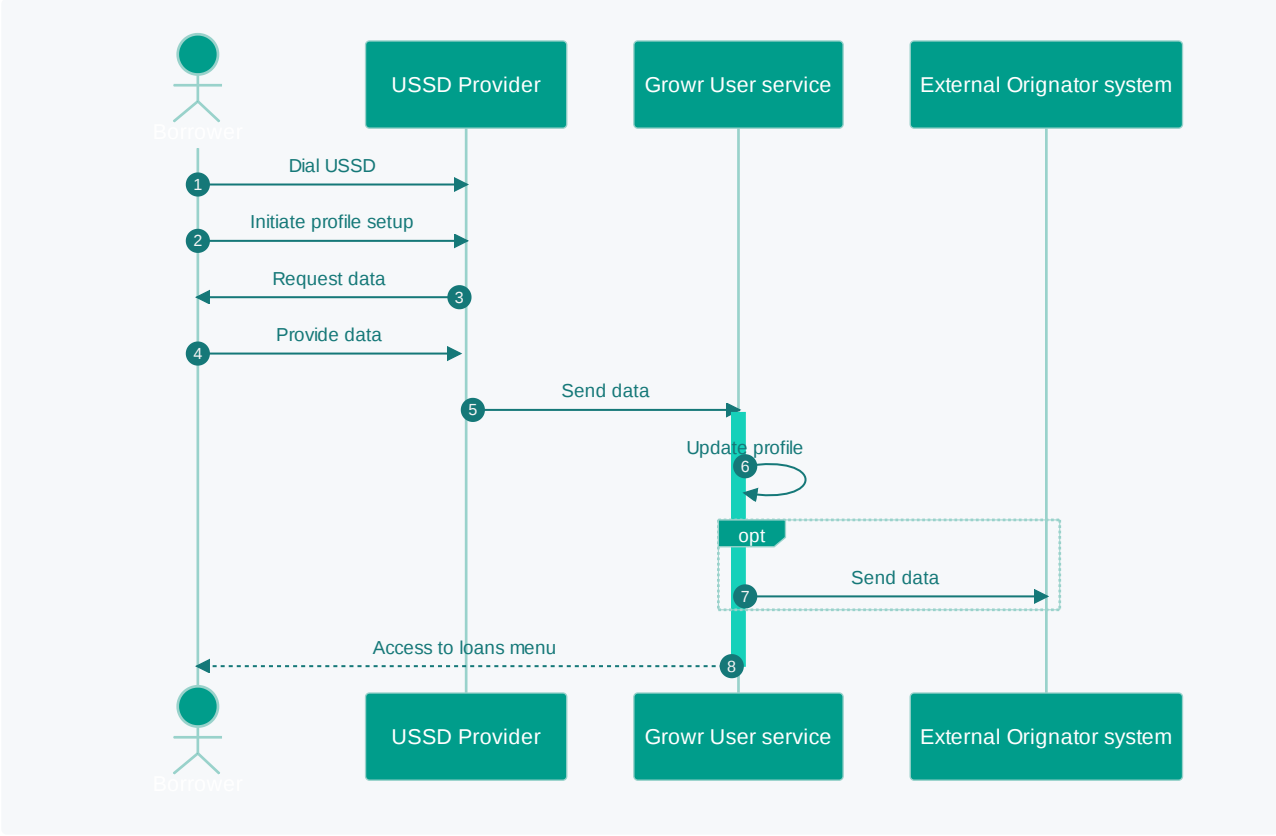
In a non-custodial model, the borrowers would use a borrowing dApp to create and manage their self-custody wallet, then claim credentials and apply for a loan from the protocol without involving a third party.

The processes in the following sections illustrate a scenario with a custodial application with a USSD interface.

Borrower registration

See User registration

Borrower profile setup with a phone

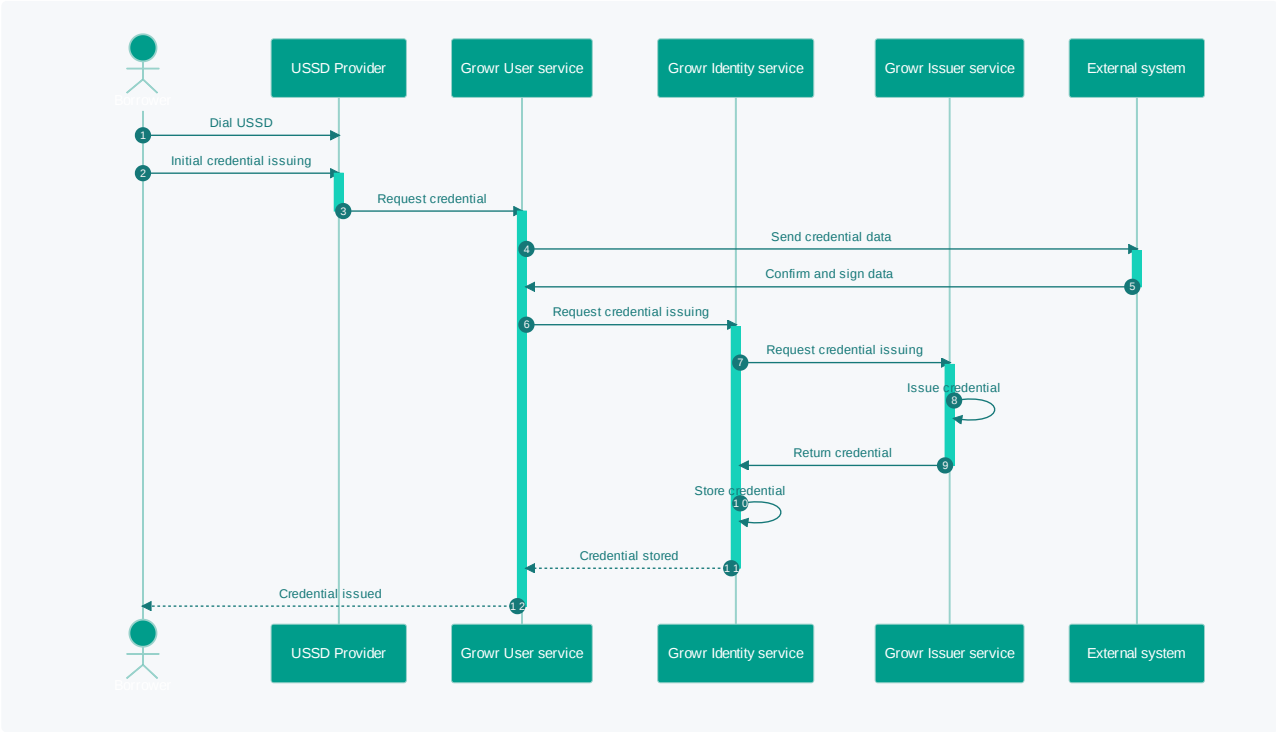


Process steps:

1. A borrower dials a number to receive a USSD menu.
2. The user selects the "profile setup" option from the menu.

3. The USSD shows a series of screens, prompting the borrower to provide information about him/her and his/her activities.
4. The borrower provides the requested data.
5. The USSD provider sends the data to the Growr User service.
6. Growr User service updates the record in the Users collection by adding the collected information.
7. Optionally, the Growr User service sends the data to an external Originator system (eg. a scoring system).
8. The borrower is now authorized to access the "loans" option from the menu.

Credential issuing with a phone



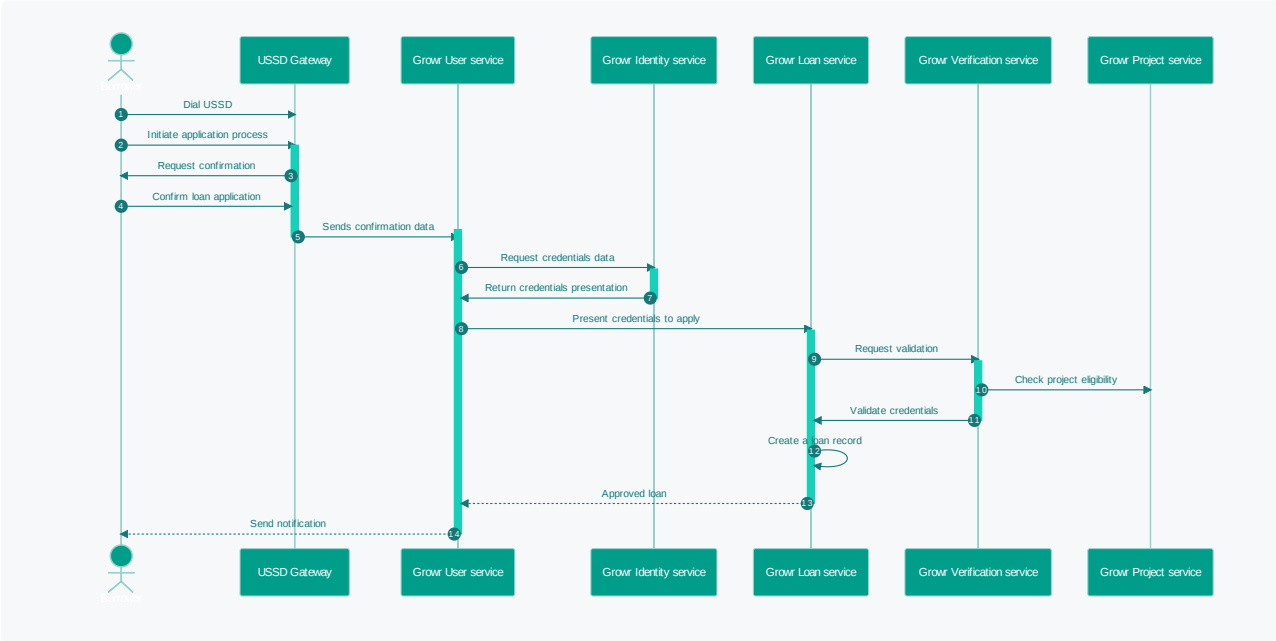
Process steps:

1. A borrower dials a number to receive a USSD menu.
2. The user selects the "credentials" option from the menu and selects data attributes with a proven source.
3. The USSD provider sends the request to Growr User service.
4. Growr User service sends the credential data to a trusted external system.
5. The external system verifies the data, signs it and returns the signed data back to Growr User service.
6. Growr User service initiates credential issuing.
7. Growr Identity service requests credential issuing from the Growr Issuing service.
8. Growr Issuing service issues the verifiable credential.
9. Growr Issuing service returns the credential to the Growr Identity service.
10. Growr Identity service stores the issued verifiable credential in the borrower's SSCR.
11. Growr Identity service confirms that the credential is issued and stored securely.

12. Growr User service sends an SMS notification to the borrower.

Loan application

Loan application with a phone



Process steps:

1. A borrower dials a number to receive a USSD menu.
2. The user selects the "loan application" option from the menu.
3. The USSD shows the terms and conditions of the loan and prompts the borrower to provide confirmation.
4. The borrower confirms the loan application.
5. The USSD provider sends the data to the Growr User service.
6. Growr User service requests verifiable credentials stored in the borrower’s Growr SSCR agent.
7. Growr Identity service creates a credential presentation and returns it to the Growr User service.
8. Growr User service presents the credentials in front of the Growr Loan service to apply for a loan.
9. Growr Loan service requests credentials validation from the Growr Verification service.
10. Growr Verification service reads the project eligibility criteria from the Growr Project service.
11. Growr Verification service validates the credentials and verifies the eligibility of the farmer in front of the Growr loan service.
12. Growr Loan service creates a new record in the protocol’s Loans book.
13. Growr Loan service confirms that the loan is approved and created.
14. Growr User service sends an SMS notification to the borrower.

The key step of the process is the eligibility assessment performed by the Verification service. It includes the following checks:

- Verify that the borrower has presented all the necessary verifiable credentials required for the project

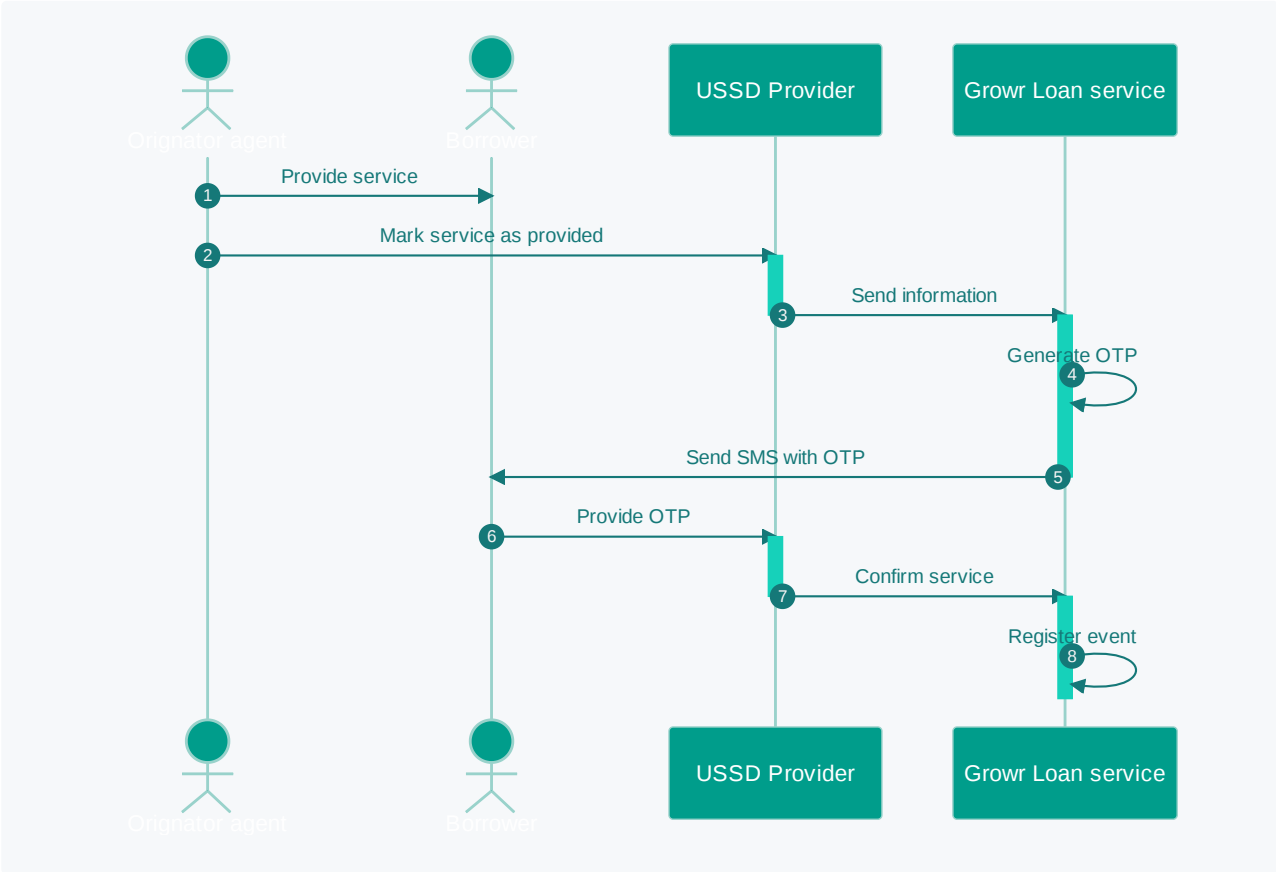
- Verify the validity of each credential:
- Verify that the credential presentation is signed with the borrower's identity
- Verify that the credential is signed by a trusted credential issuer
- Verify that the credential is not expired
- Verify that the credential is not revoked

Loan utilization

Loan disbursement

In progress

Service tracking with phone

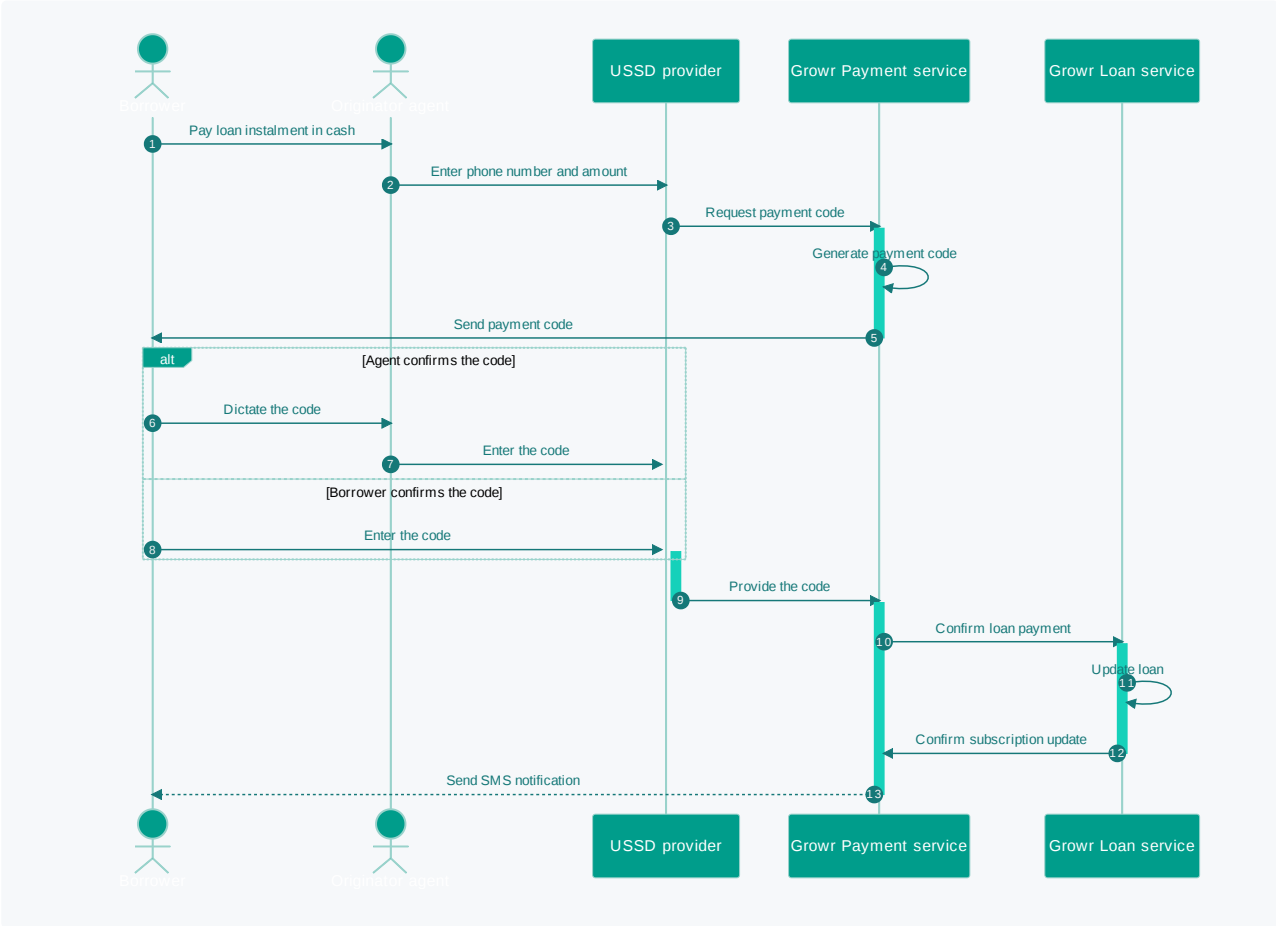


Process steps:

1. An agent from the originator (or from a local service provider) provides the service/good, for which the borrower applied.
2. The agent marks the service as successfully provided using a USSD interface.
3. The USSD provider sends the information to the Growr loan service.
4. Growr Loan service generates an OTP code.
5. Growr Loan service sends an SMS with the code to the borrower.
6. The borrower browses the USSD menu and enters the received OTP code.
7. The USSD provider sends the confirmation to the Growr Loan service.
8. Growr Loan service registers the event in the internal Loan Event book.

Loan repayment

Cash payment

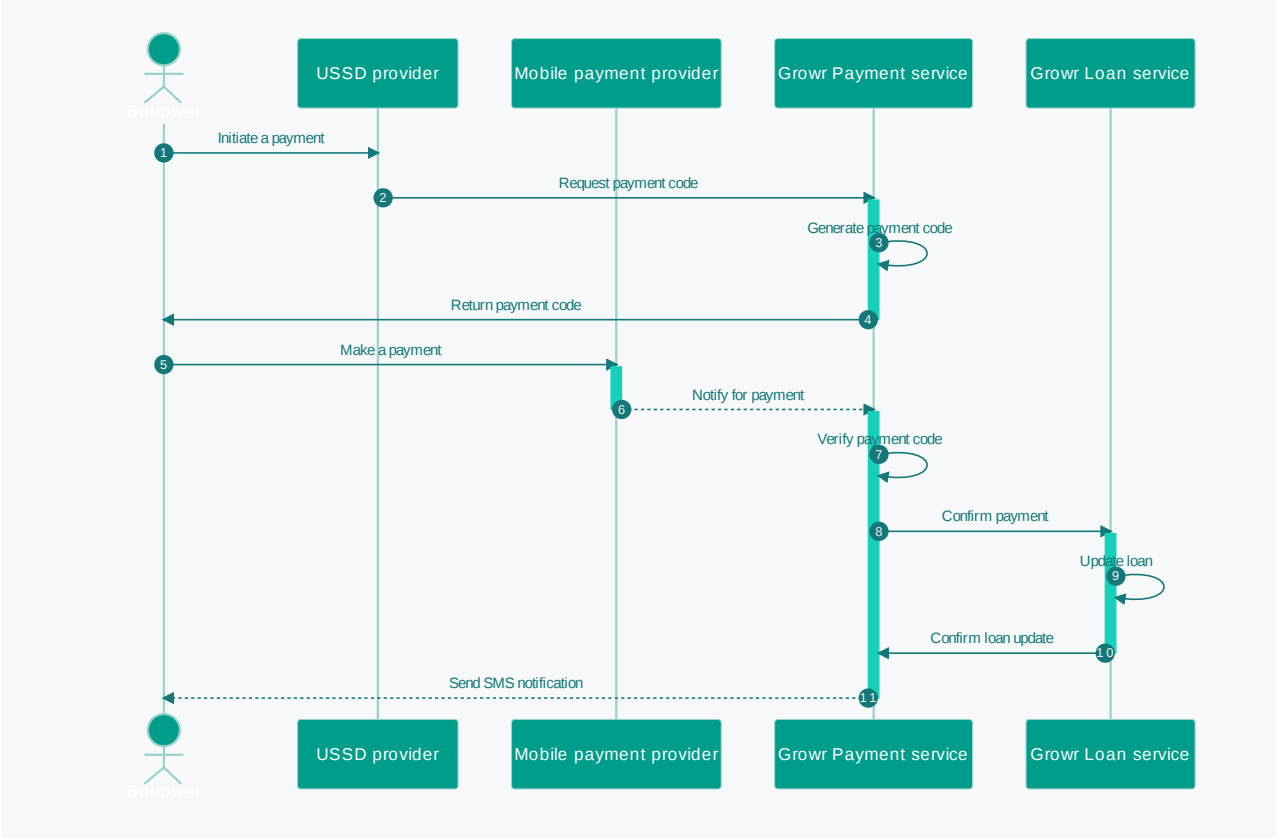


Process steps:

1. An originator agent visits a borrower and reminds him/her about an upcoming loan payment. The borrower pays the amount in cash to the agent.
2. The agent registers the payment by entering the borrower's phone number and the paid amount through a specific USSD menu.
3. The USSD provider sends the request to the Growr Payment service.
4. Growr Payment service generates a unique OTP code for the respective loan payment.
5. Growr Payment service sends an SMS with the OTP code to the borrower.
6. The borrower dictates the code to the agent.
7. The agent enters the code to verify the transaction.
8. Alternatively, the borrower enters the code by himself.
9. The USSD provider sends the confirmed code to the Growr Payment service.
10. Growr Payment service verifies the code and asserts the payment in front of the Growr Loan service.
11. Growr Loan service registers the payment event and updates the loan - respectively increasing the paid amount and decreasing the due amount.
12. Growr Loan service returns a confirmation to the Growr Payment service.

13. Growr Payment service sends an SMS notification to the borrower.

Mobile payment



Process steps:

1. According to the loan repayment schedule, a borrower initiates a payment request from the respective USSD menu.
2. The USSD provider sends the request to the Growr Payment service.
3. Growr Payment service generates a unique code (with a predefined prefix) for the respective loan payment.
4. Growr Payment service sends an SMS with the payment code to the borrower.
5. Using the provided code, the borrower makes a payment using his Mobile Money wallet.
6. The Mobile payment provider detects a payment with the given prefix and notifies Growr Payment service about the transaction.
7. Growr Payment service verifies the payment code from the transaction.
8. Growr Payment service asserts the payment in front of the Growr Loan service.
9. Growr Loan service registers the payment event and updates the loan.
10. Growr Loan service returns a confirmation to the Growr Payment service.
11. Growr Payment service sends an SMS notification to the borrower.

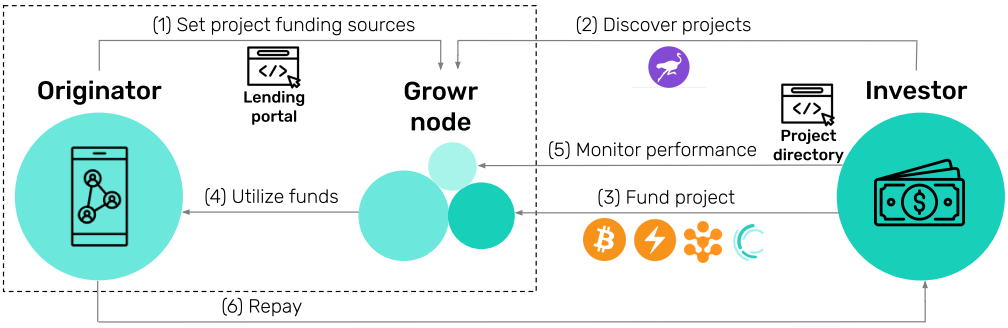
Onchain payment

In progress

FUNDING OVERVIEW

Overview

Below is a diagram that summarizes the funding processes:



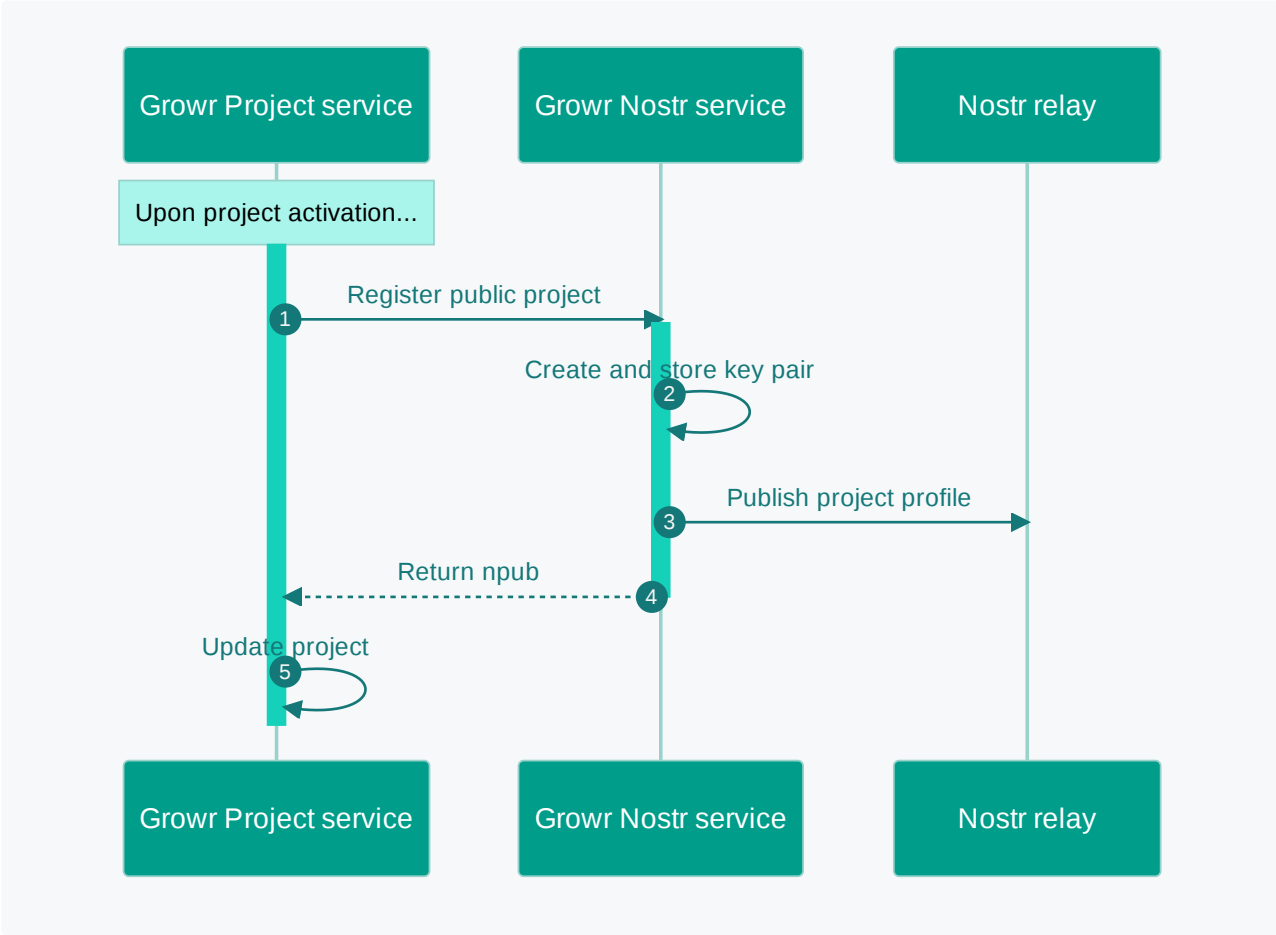
1. While creating the loan offer, the *originator* defines the project funding terms and available payment rails.
2. A *capital provider* reviews the projects with their predefined eligibility criteria and funding parameters, assesses the risk and confirms his investment intent.
3. The *capital provider* deploys funding to the project as a senior tranche via any of the possible payment rails.
4. The *originator* utilizes the provided capital and performs the actual lending activity to the *borrowers*.
5. The *capital provider* monitors the performance of their investment through Growr protocol that provides reporting of real-time events and full transparency into how the capital is deployed into borrower subscriptions.
6. According to the agreed funding terms, the *originator* repays back the senior tranche plus the generated yield.

The next sections describe the following processes:

- Project discovery
- Investor onboarding
- Project funding
- Project funds utilization
- Project repayment

Project discovery

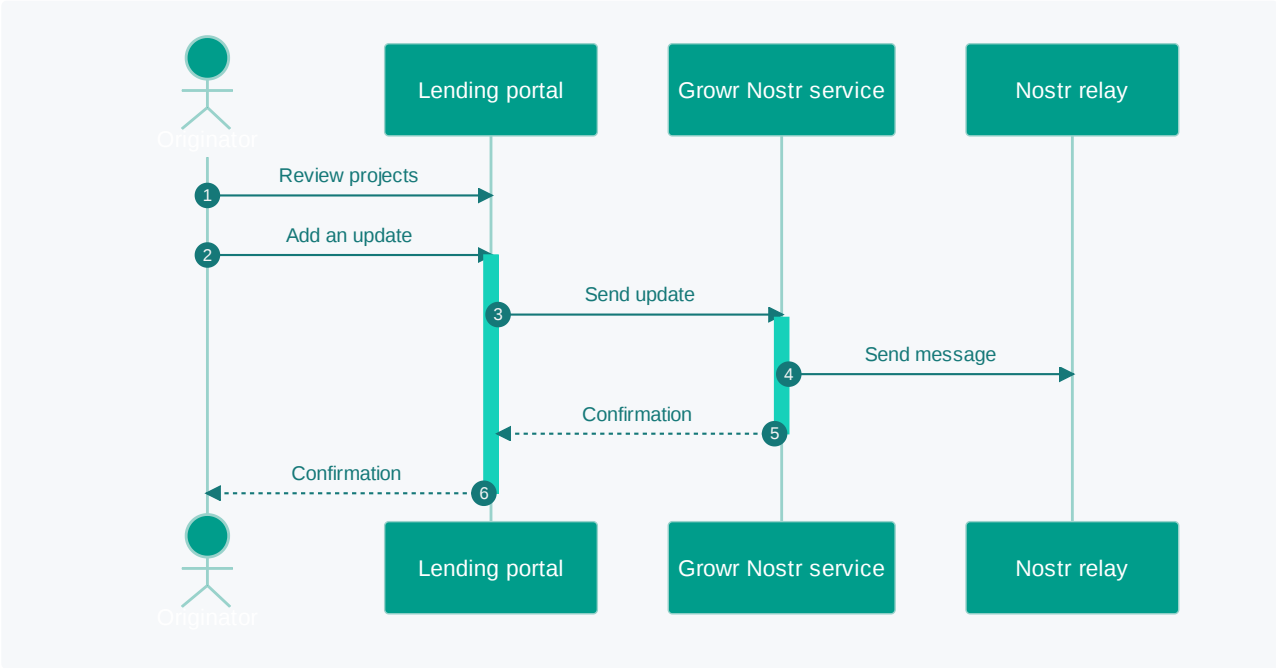
Project publishing



Process steps:

1. Growr Project service sends information to Growr Nostr service about a new public project.
2. Growr Nostr service generates a new pair of public-private keys and stores them securely.
3. Growr Nostr service publishes a Nostr profile for the project through a Nostr relay.
4. Growr Nostr service returns the npub of the Nostr profile.
5. Growr Project service stores the npub address and updates the record of the respective project in the Project book.

Project feed



Process steps:

1. An originator reviews the projects in the Lending portal and selects one (that is public).
2. The originator enters a new message with a description of the project status or achievements.
3. The Lending portal sends the information to Growr Nostr service.
4. Growr Nostr service posts a message to the project's Nostr profile through a Nostr relay.
5. Response message.
6. Response message.

Project discovery

Investors can discover lending projects in 3 ways:

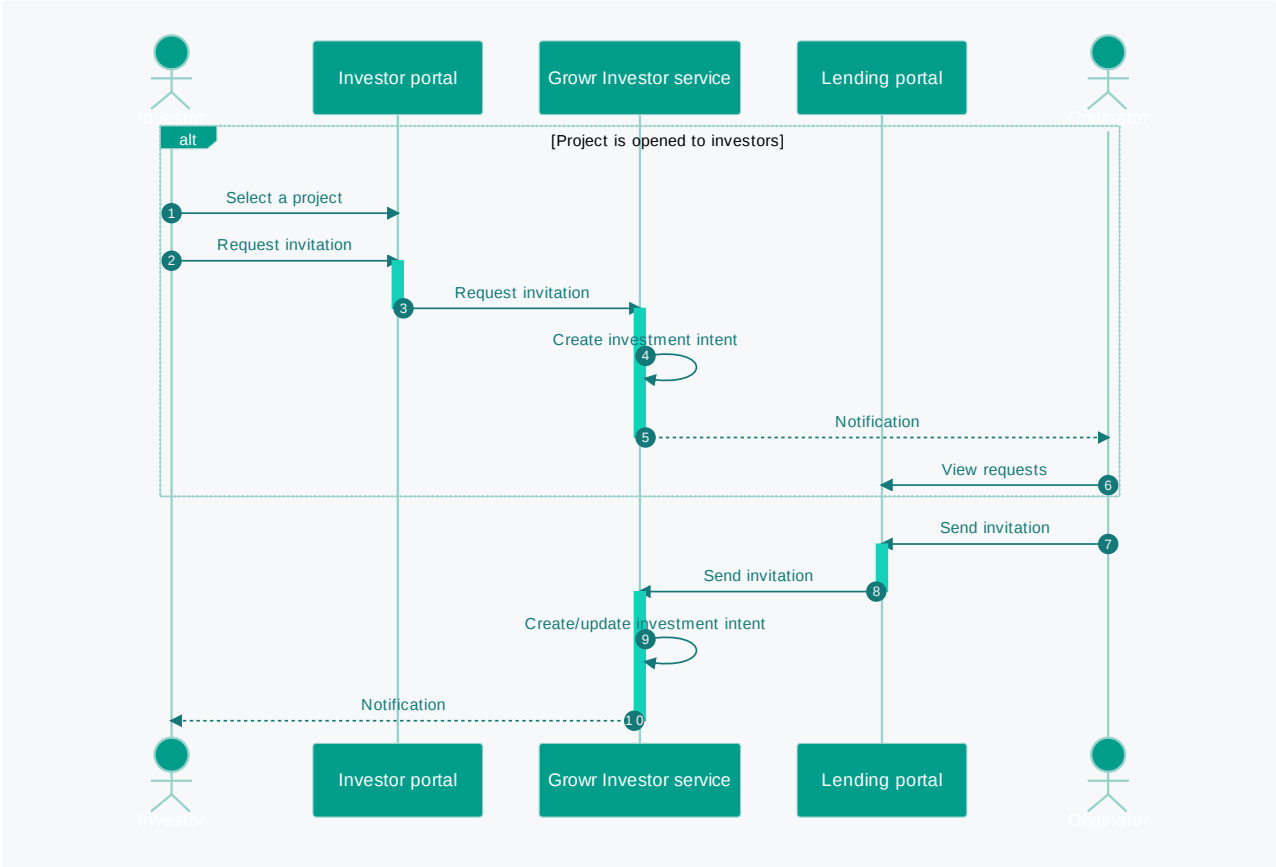
- Review the Nostr profile of a public project.
- Review a public project in the project directory in the Investor portal.
- Review a non-public project in the Investor portal after he/she has been invited as an investor to this project.

Investor onboarding

Investor registration

See User registration

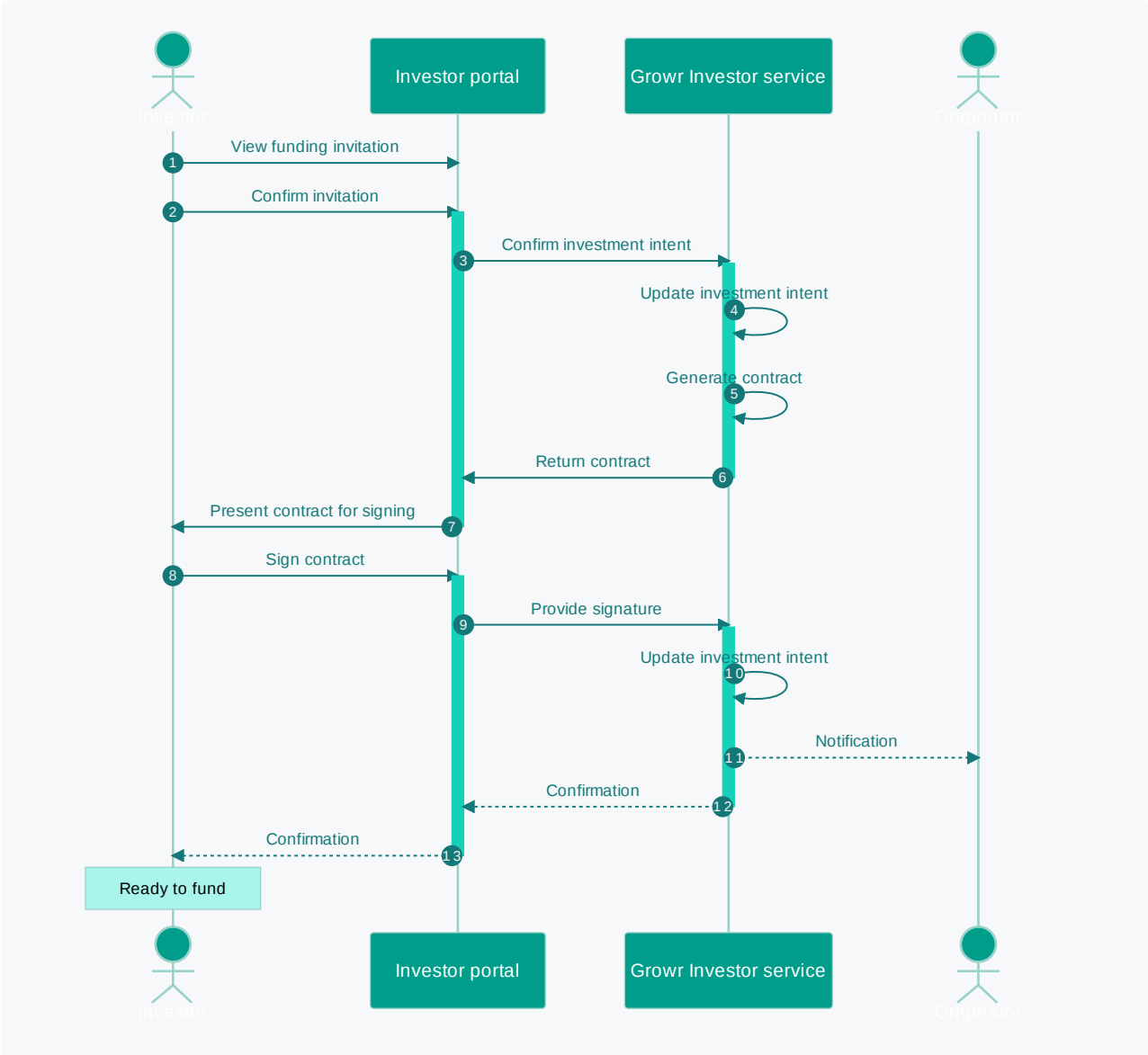
Project funding invitation



Process steps:

1. An investor reviews the projects through the Investor portal and selects one that is open for investment.
2. The investor requests a funding invitation for the selected project.
3. The Investor portal sends the request to Growr Investor service.
4. Growr Project investor service registers the investment funding intent with the status REQUESTED.
5. Growr Project investor service sends an email notification to the originator.
6. An originator admin reviews the funding invitation requests in the Lending portal.
7. The originator sends an invitation for project funding to a given investor (with or without an existing invitation request).
8. The Lending portal sends the request to Growr Investor service.
9. Growr Project investor service registers (or update if existing) the investment funding intent with the status INVITED.
10. Growr Project investor service sends an email notification to the investor.

Project funding commitment



Process steps:

1. An investor is invited to fund a project.
2. The investor confirms the funding invitation through the Investor portal, indicating the investment amount, funding source and wallet address.
3. The Investor portal sends the confirmation to Growr Investor service.
4. Growr Project investor service updates the investment intent with the status CONFIRMED.
5. Growr Project investor service generates an investment contract between the investor and the originator.
6. Growr Project investor service sends the contract back to the Investor portal.
7. The investor downloads the contract from the Investor portal and reviews it.
8. The investor signs the contract.
9. The Investor portal sends the signed contract together with the investor's signature to the Growr Project investor service.
10. Growr Project investor service updates the investment funding intent with the status SIGNED.
11. Growr Project investor service sends an email notification to the originator.

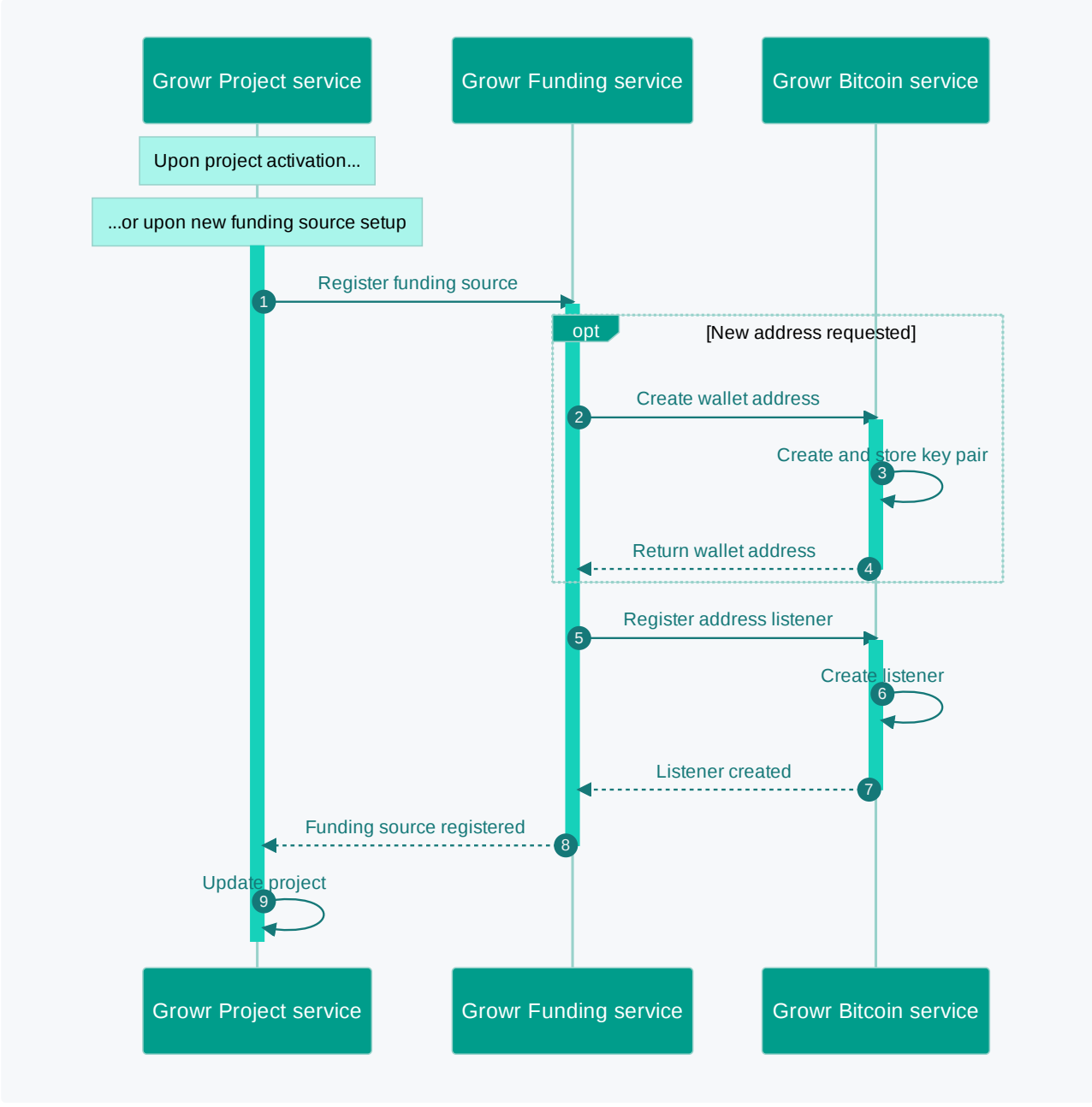
12. Response message.
13. Response message.

Project funding

Currently, the Growr protocol supports 4 types of funding sources:

- *Traditional (bank account):* Growr protocol does not monitor or manage any payments; they are executed “off-chain” and registered in the system by the originator.
- *Digital (on-chain wallet):* Growr protocol monitors the project wallet and automatically tracks all investments and repayments.
- *Digital (multi-sig wallet):* Growr protocol creates a multi-sig wallet, in which the investor transfers digital assets. To claim the money from the wallet, the originator signs a loan agreement.
- *Crowdfunding:* Growr protocol supports bitcoin crowdfunding leveraging Lightning Network as a payment rail and Nostr as a message exchange. Crowdfunded projects have a Nostr profile and multiple investors can fund them via zap messages in Nostr.

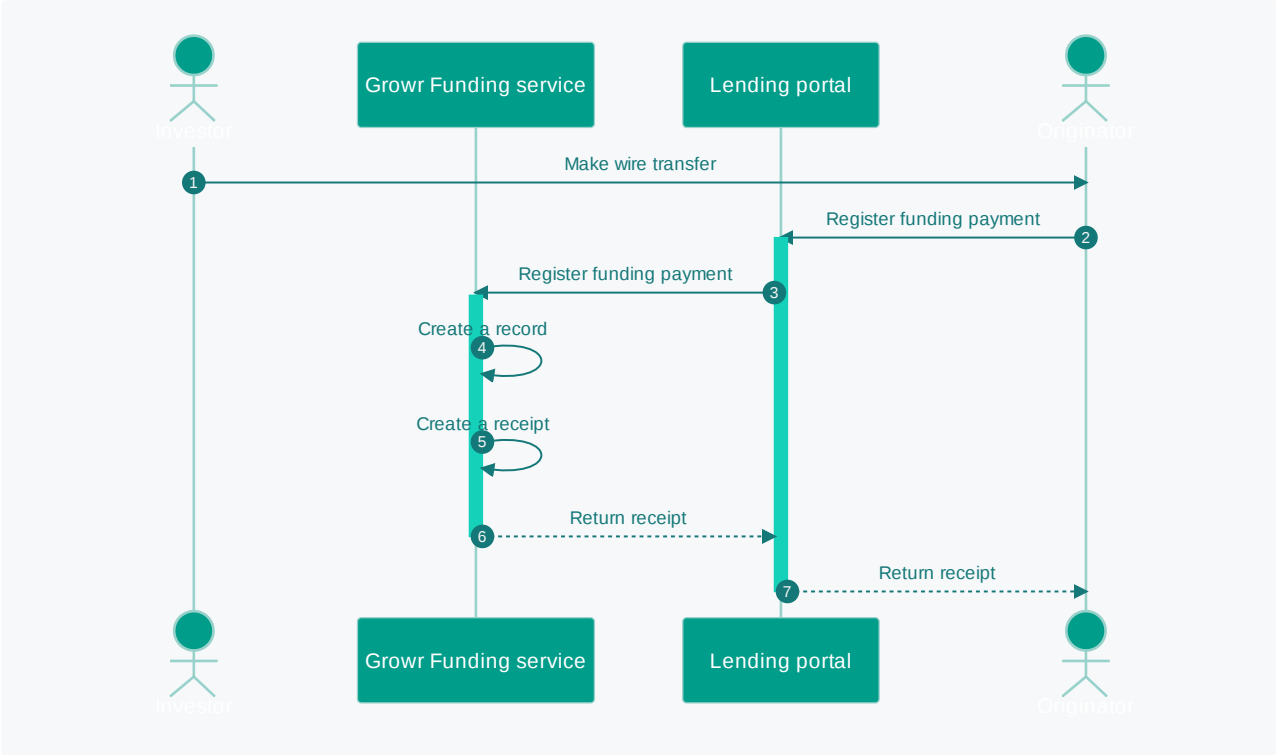
Funding setup



Process steps:

1. Growr Project service sends information to Growr Funding service about a new funding source.
2. Growr Funding service requests the creation of a new Bitcoin address for the project.
3. Growr Bitcoin service generates a new pair of public-private keys and stores them securely.
4. Growr Bitcoin service returns the new wallet address to the Growr Funding service.
5. Growr Funding service requests the creation of a new on-chain listener.
6. Growr Bitcoin service registers a new listener that will detect all transactions to the project wallet address.
7. Growr Bitcoin service returns confirmation.
8. Growr Funding service returns confirmation and information about the newly created wallet address.
9. Growr Project service stores the new wallet address and updates the record of the respective project in the Project book.

Project funding with a bank account

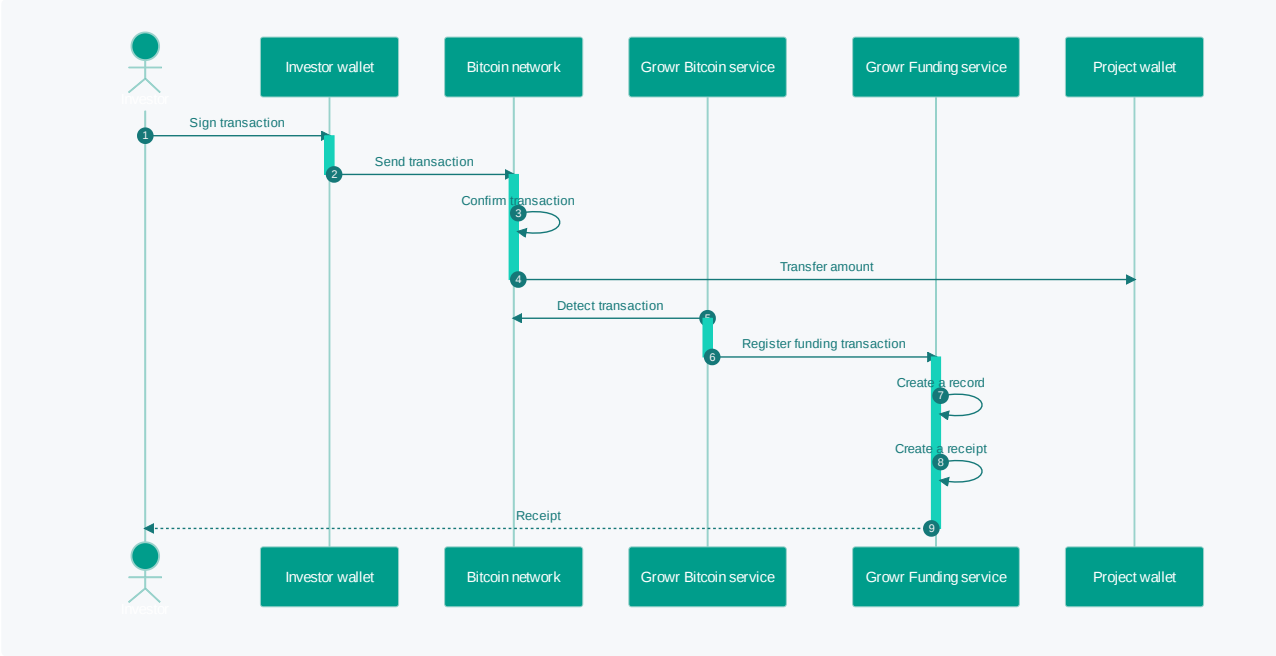


Process steps:

1. An already onboarded investor makes a wire transfer to a bank account, created for the project.
2. Upon receiving the transfer, the originator of the respective project registers the received investment through the Lending portal.
3. The Lending portal sends the information to the Growr Funding service.
4. Growr Funding service creates a new record in the Funding book.
5. Growr Funding service creates a receipt for the investment with a signature on behalf of the originator.
6. Growr Funding service returns the receipt to the Lending portal.

7. The Lending portal presents the receipt to the investor.

Project funding with an on-chain wallet

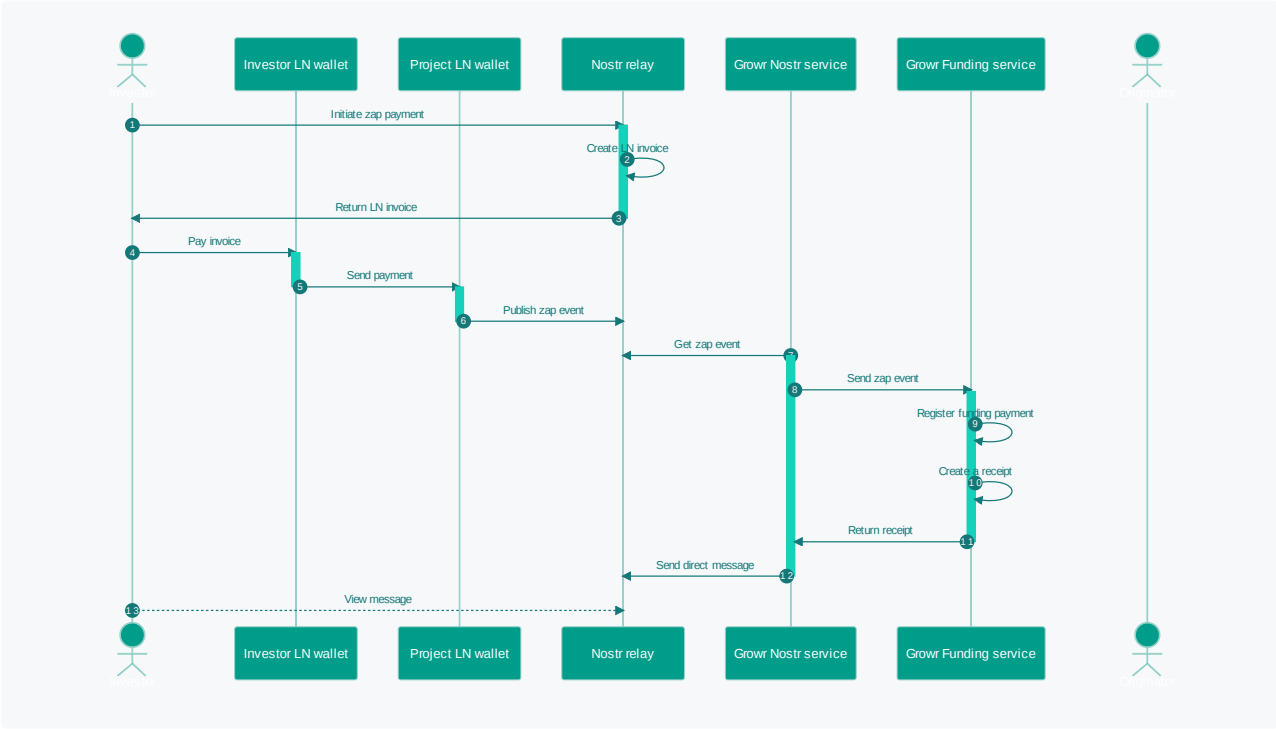


Process steps:

1. An already onboarded investor signs an on-chain transaction using his Bitcoin wallet.
2. The investor's wallet sends the transaction to the Bitcoin network.
3. The transaction is confirmed by the network in the next block.
4. The amount is transferred to the Bitcoin wallet address of the project.
5. Growr Bitcoin service detects an incoming transaction to the project wallet.
6. Growr Bitcoin service sends the information to the Growr Funding service.
7. Growr Funding service creates a new record in the Funding book.
8. Growr Funding service creates a receipt for the investment with a signature on behalf of the originator.
9. Growr Funding service sends an email with an attached receipt to the investor.

The process is the same for projects with funding in Rootstock or Liquid networks.

Project crowdfunding



Process steps:

1. An investor initiates a zap payment (event kind = 9734) to the project Nostr profile using his Nostr client.
2. The Nostr relay, which includes also an LNURL server, creates a Lightning invoice.
3. The Lightning invoice is presented to the investor.
4. The investor makes a payment to the invoice using his Nostr or Lightning client.
5. Upon paying the invoice, the funds arrive at the Lightning address of the project.
6. A Zap note is created (event kind = 9735) and sent to the Nostr relay.
7. Growr Nostr service listens for new messages on the Project Nostr profile and detects the new Zap event.
8. Growr Nostr service sends the information to the Growr Funding service.
9. Growr Funding service creates a new record in the Funding book.
10. Growr Funding service creates a receipt for the investment with a signature on behalf of the originator.
11. Growr Funding service returns the receipt to Growr Nostr service.
12. Growr Nostr service creates a new direct message (event kind = 4) with an attached receipt.
13. The investor receives the message using his Nostr client.

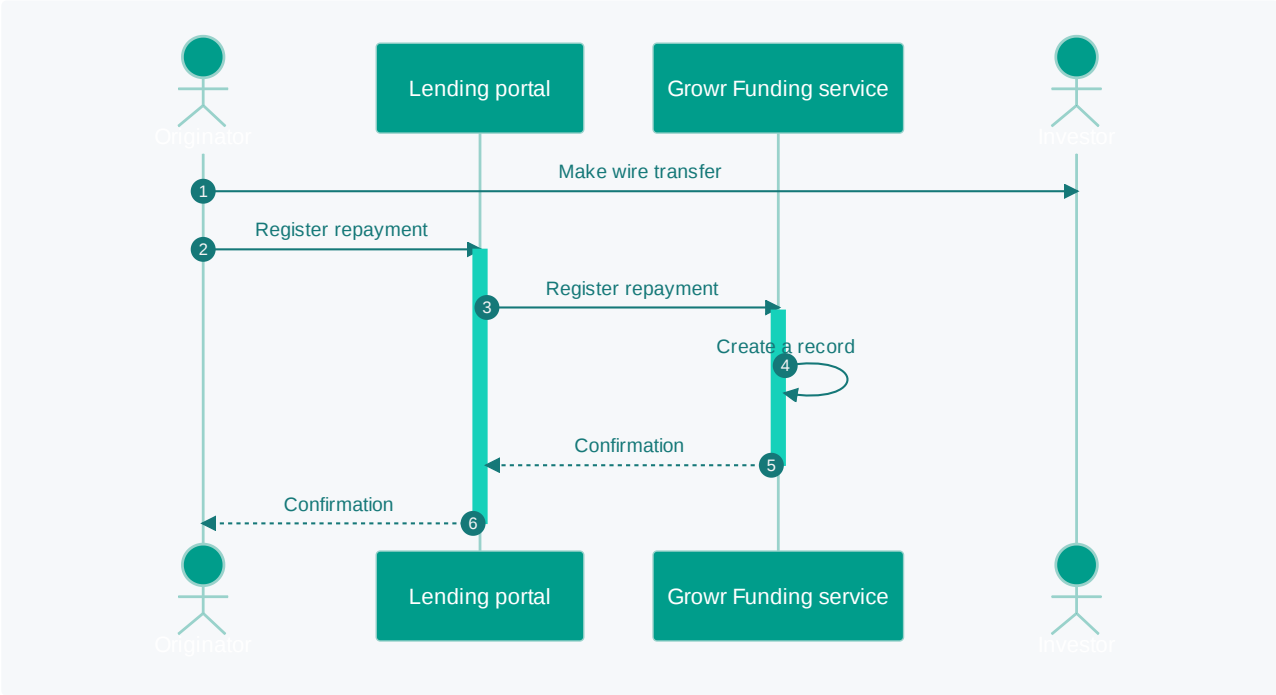
Project funds utilization

The originator needs project funds to provide the necessary lending operations. Therefore, the collected project funds are utilized through the utilization of the approved loans.

See Loan utilization

Project repayment

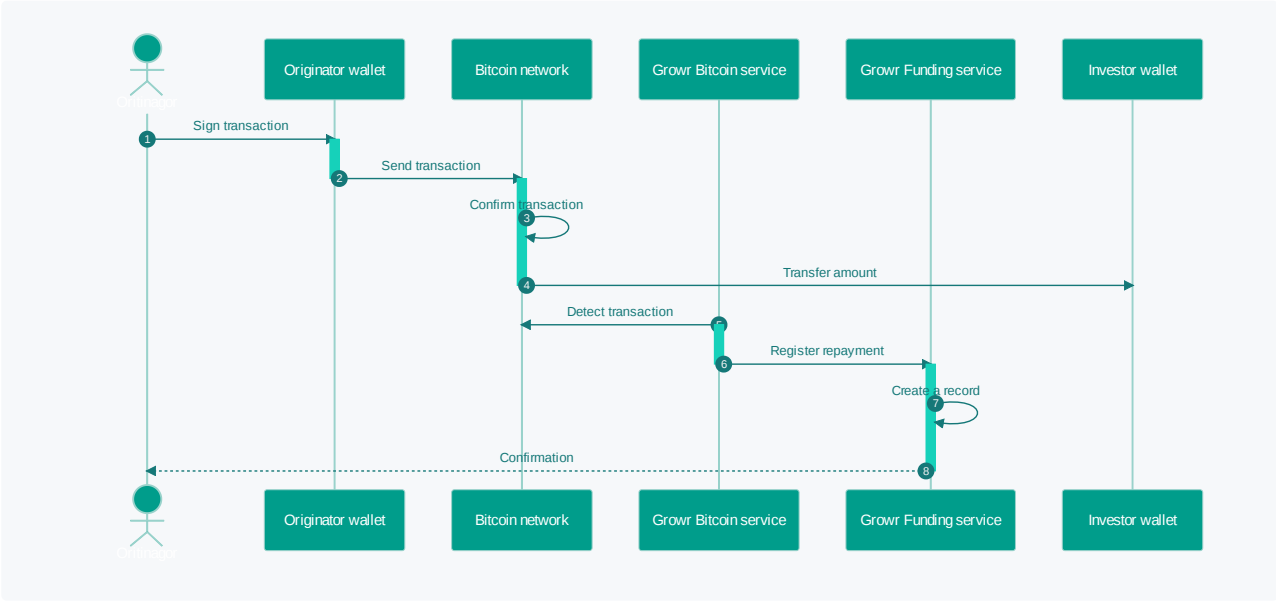
Project repayment with a bank account



Process steps:

1. According to the agreed repayment plan, an originator makes a wire transfer to the bank account of the investor.
2. The originator also registers the repayment through the Lending portal.
3. The Lending portal sends the information to the Growr Funding service.
4. Growr Funding service creates a new record in the Funding book.
5. Response message.
6. Response message.

Project repayment with an on-chain wallet



Process steps:

1. An originator signs an on-chain transaction using his Bitcoin wallet.
2. The originator's wallet sends the transaction to the Bitcoin network.
3. The transaction is confirmed by the network in the next block.
4. The amount is transferred to the Bitcoin wallet address of the investor.
5. Growr Bitcoin service detects the outgoing transaction.
6. Growr Bitcoin service sends the information to the Growr Funding service.
7. Growr Funding service creates a new record in the Funding book.
8. Response message.

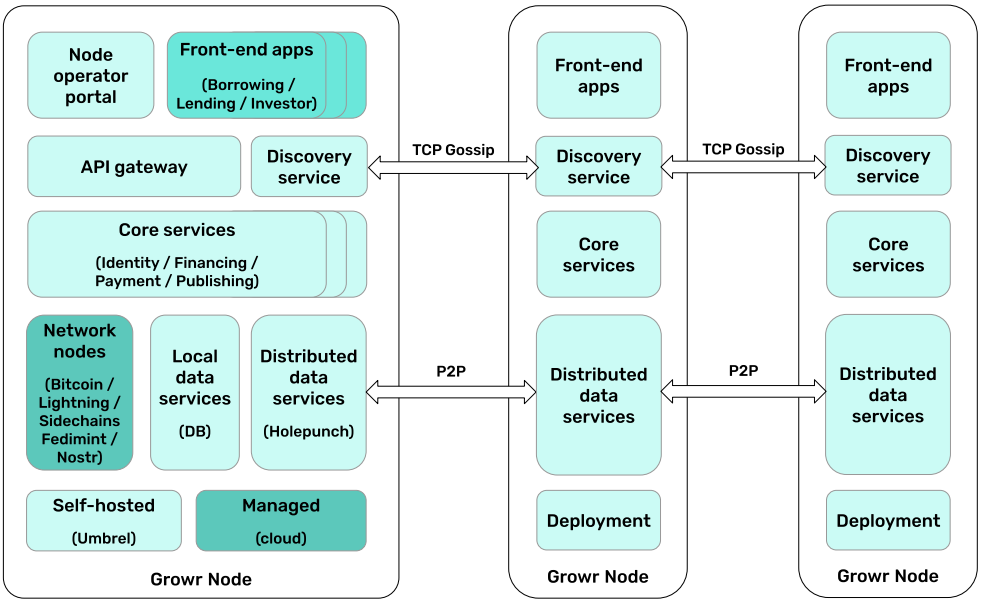
ARCHITECTURE OVERVIEW

Overview

The design of Growr architecture includes the following considerations:

- The *identity system* should allow users to prove their identity without exposing their personal data. It should also allow the users to decide whether to control their data by themselves or to use a custodial service.
- The *storage system* should enable secure and reliable storage of the data gathered from all lending and funding activities. To provide transparency in the protocol, tamper-proof cryptographically verified data should be easily and openly distributed.
- The *communication system* should facilitate the outreach and engagement of borrowers, originators and investors. It should enable a secure and verifiable exchange of lending documents and signatures.

The following diagram provides a high-level overview of the Growr functional architecture.



Growr protocol implements a global decentralized micro-lending marketplace as a P2P network of Growr nodes. Each *Growr node* represents a package of open-source components and a set of lending projects. It enables originators to publish projects with predefined conditions and eligibility criteria, and borrowers to apply and get financing using credentials from their self-sovereign credit record. It also enables capital providers to fund projects either by accessing the Growr node of an originator directly; or by running their own node.

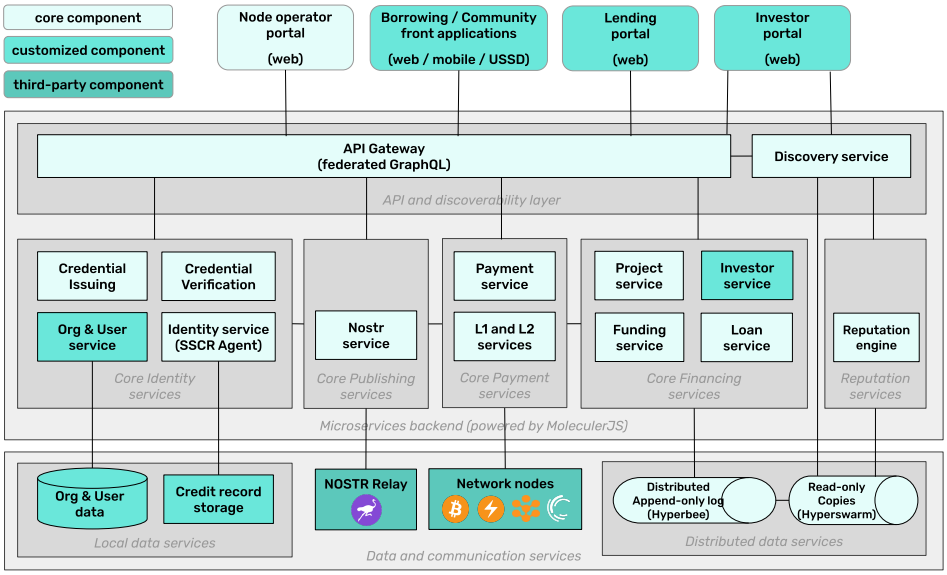
Growr architecture provides the opportunity to pack the applications and services in a few options, based on the deployment requirements and specifics. There are several deployment blueprints for the Growr node depending on the use cases; for example, a self-hosted originator node, a cloud-hosted originator node or a cloud-hosted investor node. The backend services can be packed as a

single docker image and can be included in different configurations to run all or a few services per deployment target.

The Growr nodes communicate with each other in 2 ways. The first option is through a Discovery service, using the TCP Gossip transport protocol. The second option is through replication of the distributed data across the network, using Holepunch's Hyperbee and Hyperswarm protocols.

Architecture components

The following component diagram presents the applications and services, available in the Growr node.



At the top of the diagram are the *front-facing applications*. These apps and services are on top of FOSS protocol and services. They can be developed by Growr or other integrators per deployment. The node can be extended with different applications fitting the purpose.

The *backend of the Growr node* consists of a group of services that are maintained as open-source software under an open license but can be extended following the community and product guidelines. All the services in these groups are deployed with proof that the running service has a well-known identity or a public key, and its code is identical to the source code in the Growr repositories.

Growr node relies on 2 types of *data services*:

- *Local data services:* Database collection that stores editable and manageable data locally in the node.
- *Distributed data services:* Append-only B-tree logs with read-only copies that are shared and can be discovered by external services by their public key.

Front-end applications

Node operator portal

The *Node operator portal* is used by the deployment and administration team to set up and manage the Growr node.

Borrowing and Community front apps

Those applications are used by the borrowers to:

- Register to the protocol
- Operate their self-sovereign credit record
- Apply for loans from different projects
- Repay their loans

Depending on the local environment and user base, those applications could be developed with web, mobile or USSD interface.

Lending portal

The *Lending portal* is used by originators to:

- Register to the protocol
- Setup projects
- Publish projects
- Manage projects
- Monitor lending activities
- Monitor funding activities

Investor portal

The *Investor portal* is used by investors to:

- Register to the protocol
- Discover projects
- Invest into projects
- Monitor project performance

Impact portal

The *Impact portal* is an open-source standalone web application that uses aggregated data from different nodes from the protocol. It displays a dashboard presenting the global impact of the network.

Back-end services

Overview

The backend of Growr has microservices architecture based on *MoleculerJS*. Besides the traditional features, such as load balancing, service discovery, and fault tolerance, MoleculerJS has the ability to distribute messages across multiple nodes using the TCP gossip protocol and allows easy switching of the transport technology depending on the use case.

Below is a table containing a summarized list of the back-end services:

Group	Service	Data storage
Access	API gateway	-
Access	Discovery	-
Identity	Organization	local
Identity	User	local
Identity	Identity	Local / External
Identity	Credential issuing	-
Identity	Credential verification	-
Financing	Project	distributed
Financing	Loan	distributed
Financing	Investor	local
Financing	Funding	distributed
Payment	Payment	local
Payment	Bitcoin	external (L1)
Payment	Lightning	external (L2)
Payment	Rootstock	external (L1)
Publishing	Nostr	external (Nostr)

API gateway

The *API gateway* controls access to the back-end services. It exposes APIs using GraphQL, providing a flexible and efficient way to retrieve data from the services.

Discovery service

The *Discovery service* ensures the discoverability of Growr nodes and the possibility to use other node services (for example, Verification or Issuing) using the TCP Gossip transport protocol.

Identity services

The main *Identity service* (also called *SSCR Agent*) is based on Self Sovereign Identity (SSI) standard. SSI is designed for managing identity data in a decentralized, secure way. It supports the creation, management, and sharing of verifiable credentials that can be used to prove identity attributes without revealing sensitive personal information.

In the current protocol implementation, the *Identity service* is implemented as a custodial service that operates the self-sovereign credit record (SSCR) of the user with his permission. It controls the DID of the borrower and his/her verifiable credentials.

The *Credential issuing service* is responsible for issuing verifiable credentials for given borrowers based on data received or verified by the originator or other third parties. The *Credential verification service* is responsible for the validation of presentations of verifiable credentials and assessing the eligibility of a given user to apply for a loan. To be trusted by all network participants, those two services have well-known DIDs and deployment addresses.

The *Organization service* and the *User service* are responsible for the registration and management of the users of the frontend applications in the node. They read and write data to database collections having the same name.

Financing services

The *Project service:* is responsible for creating and managing the lending projects. It reads and writes data to the *Project book*.

The *Loan service:* controls the approval, creation, utilization and repayment of loans. It reads and writes data to the *Loan book*.

The *Funding service:* is responsible for project funding and project repayment operations. It reads and writes data to the *Funding book*.

The *Project investor service:* provides onboarding and contract management services for project investors. It reads and writes data to a database collection having the same name.

Payment services

Growr protocol implements a set of services with payment management functions.

The main *Payment service* is responsible to track all payments related to activities in the protocol. It reads and writes data to the *Payment book*.

The *Bitcoin service*, *Lightning service* and *Rootstock service* provide integration with the respective payment networks. They are responsible for wallet management, transaction execution and transaction listening.

Publishing services

Growr protocol implements publishing through *NOSTR*. It is an innovative protocol designed to create censorship-resistant global social networks. Its tamper-proof design is based on cryptographic

keys and signatures, relying on a network of distributed relays to ensure messages are sent in a robust and censorship-resistant manner.

Growr *Nostr service* is responsible for creating a public Nostr profile and posting various types of messages on behalf of this profile.

Supporting services

The backend of the Growr protocol contains also a set of supporting services that are not directly exposed via APIs but are used internally by the other services. They include:

- Database management, MongoDB
- Append-only log, Hyperbee and Hyperswarm
- Private key management, NDS
- Authentication, Passport
- JSON rule engine
- Email & SMS

Data services

Credit record storage

Credit record storage provides decentralized storage of the self-sovereign credit record (SSCR) of the users. Each record represents a unique global decentralized identity and contains general-purpose and protocol-specific verifiable credentials. The credentials data is encrypted and accessible only by the identity owner.

The SSCR contains both hard information (facts such as credit score and history, debt-to-income ratio, bank account verification, and business financial indicators) and soft information (such as endorsement, community membership, and self-declared business plans) that are used in the credit risk assessment.

Potential types of verifiable credentials to be stored in the SSCR include:

Credential category	Type	Proof	Description
KYC	hard	identity	proves a successfully passed KYC process (including AML/CFT risk check)
Financial data	hard	creditworthiness	shows the financial habits of the user, as well as the trends in his cash flow (eg. transaction history, products)
Savings history	hard	creditworthiness	proves the financial discipline of the user (eg. regular micro-payments)
Credit history	hard	creditworthiness & trustworthiness	proves the financial discipline of the user (eg. past loans from the protocol)
Business activity	hard	creditworthiness	proves past and future revenue (income statement, cash flow, balance sheet)
Ownership	hard	creditworthiness	proves the ownership of assets required for achieving the goals (eg. land, machines, raw materials)
Community membership	soft	reputation	asserts the trust of the organization
Social endorsement	soft	reputation	asserts the trust of other protocol participants having a certain reputation level
Financial health	soft	trustworthiness	proves successfully passed financial education or mentoring

Local data services

Some of the Growr services require database collections storing editable and manageable data locally in the node. This data usually does not contain lending or funding events and is not required to be distributed in the network (eg. user registration and onboarding). The collections can be implemented in MongoDB or other NoSQL databases.

The Growr protocol's local collections are:

- *Organizations*
- *Users*
- *Project investors*

Distributed data services

For storing and distributing project lending and funding data, Growr is using Holepunch's Hyperbee, an append-only B-tree based on Hypercore. It provides a key-value data store that is both fast and efficient, making it good for storing large amounts of data in a decentralized network. Additionally, it supports peer-to-peer synchronization, allowing the data to be replicated across the network in a highly efficient and fault-tolerant manner.

In addition, read-only copies are created and contain aggregated data plus audit logs, sanitized from any personal-revealing data. This data is publicly exposed to ensure transparency in the network and to monitor its global impact. Those read-only copies are implemented as a Hyperswarm with published well-known Public Key and Topic.

The Growr protocol's distributed data logs are:

- *Project book:* enables the creation of new projects with strictly defined eligibility criteria and potential funding sources.
- *Loan book:* enables the creation of loans after an eligibility check of the borrower against the requirements of a given project.
- *Funding book:* stores information about all funding operations
- *Payment book:* stores information about loan utilization, repayment and other payment events.

Reference

Data structures

Structure	Attribute(s)	Description
Project book	key	Unique identifier
Project book	status	DRAFT | ACTIVATED | DEACTIVATED | CLOSED
Project book	organizationId userId	Information about who created the project
Project book	name description	Descriptive attributes of the project
Project book	isPublic	Whether the project is publicly discoverable or not
Project book	profileUrl bannerUrl npub	Publishing attributes of the project
Project book	startDate	Initial date of lending operations
Project book	endDate	No new loans are accepted after this date
Project book	amount currency	Total target amount and currency of the loans from the project
Project book	eligibilityCriteria	List of verifiable credentials and conditions required for loan approval
Project book	loanType loanMinDuration loanMaxDuration loanMinAmount loanMaxAmount	Restrictive parameters for the loans from the project
Project book	isInviteOnly	Whether the project requires an invitation to become an investor
Project book > funding [ ]	type	BANK | ONCHAIN
Project book >	status	ACTIVE | INACTIVE

funding [ ]		
Project book > funding [ ]	networkName tokenName fundingAddress repaymentAddress	Details of the payment rail and currency
Project book > funding [ ]	terms interestRate leverageFactor	Descriptive attributes of the funding source
Project book > funding [ ]	fundingMinAmount fundingMaxAmount	Restrictive parameters for the project investments
Funding book	projectKey	Unique identifier of the project
Funding book	fundingType fundingNetwork fundingToken	Details of the payment rail and currency
Funding book	paymentType	INCOMING | OUTGOING
Funding book	amount	Amount of the payment
Funding book	fundingAddress	Project wallet address that received the investment (for incoming events)
Funding book	repaymentAddress	Project wallet address that initiated the repayment (for outgoing events)
Funding book	investorAddress	Wallet address of the investor
Funding book	proof	Proof of the payment (trx id, invoice hash, etc.)
Loan book	key	Unique identifier of the loan
Loan book	projectKey	Unique identifier of the project
Loan book	organizationId	Unique identifier of the originator
Loan book	userDid	DID of the borrower
Loan book	verifierDid	DID of the credential verifier service
Loan book	status	DRAFT | APPROVED | DISBURSED | REPAID | DEFAULTED
Loan book	type	Type of the loan product
Loan book	startDate	Loan duration parameters

	endDate duration	
Loan book	interestRate disbursementFee cashBackRate gracePeriod	Loan financial parameters
Loan book	currency	Currency of the loan
Loan book	contractAmount	Amount to be disbursed to the borrower
Loan book	interestAmount	Total interest amount
Loan book	commitmentAmount	Total amount to be repaid (contractAmount + interestAmount)
Loan book	repaidAmount	Total repaid amount

Services

Service	Action(s)	Description
Identity (SSCR)	findByDid	Get mapping between user id and DID
Identity (SSCR)	createWallet	Create a new SSCR with DID
Identity (SSCR)	getVC	Claim a new verifiable credential
Identity (SSCR)	createVCPresentation	Create a presentation of a verifiable credential
Credential issuing	issueVC	Issue a verifiable credential
Credential verification	validateVC	Validate a verifiable credential
Organization	findById findByUser	Get organizations by different filters
Organization	createOrganization	Create a new organization
Organization	addMember	Add user to an organization
User	findById findByOrganization	Get users by different filters
User	addUser registerBorrower	Add a new user

	registerOriginator registerInvestor	
User	updateUser	Update details of a given user
User	getOTP validateOTP setPIN validatePIN loginByEmail loginBySocial	Actions required for the different types of user authentication
Project	findByKey findByOrganization findByAddress findPublic	Get projects by different filters
Project	createProject	Create a new project
Project	updateProject modifyProjectParams addProjectFunding	Update different project attributes
Project	activateProject deactivateProject	Enable or pause the lending activities within a project
Loan	findByKey findByProject findByOrganization findByUser	Get loans by different filters
Loan	applyForLoan	Request loan approval
Loan	signLoan	Sign loan application
Loan	updateLoan	Update loan details upon loan disbursement or repayment
Project investor	findByProject findByInvestor	Get project

Project investor	requestInvitation acceptInvitation rejectInvitation	Actions executed by the investor during the project investment setup
Project investor	inviteInvestor approveInvestor rejectInvestor	Actions executed by the originator during the project investment setup
Funding	findByKey findByProject	Get funding events by different filters
Funding	fund	Create a new funding event
Funding	refund revert	Create a new repayment event
Funding	detect	Process detected payment and call a method for the new event
Payment	findByLoan findByAddress	Get loan payments by different filters
Payment	addDisbursement confirmDisbursement addRepayment confirmRepayment	Create and confirm payment events
Network (Bitcoin, Lightning, Rootstock)	createWallet	Create a new private-public key pair
Network	createListener	Create a new on-chain listener for a given network and address
Network	getPayments	Get all on-chain payments detected by a given listener
Nostr	createProfile	Create Nostr profile
Nostr	createPost createDM	Post different kinds of messages
Nostr	getFeed	Get a list of different events associated with a

	getZapRequests getZaps	given profile