

Personal Finance Tracker

Project description

The personal finance tracker is an application you can use to manage your bank accounts. Users are able to log in, create multiple bank accounts and even share it with others, but the owner has the most authority over the bank account. The application allows users to make transactions to others. A person can set up budget goals so that application can give advice: the tracker will tell the person how much money he still needs in order to reach his goal, or how much he is allowed to spend. The user also has the ability to look through all of his expenses/income and sort them by certain amounts, names or dates. Additionally, there are bank administrators that can manage bank accounts and delete users to keep the application safe.

User stories

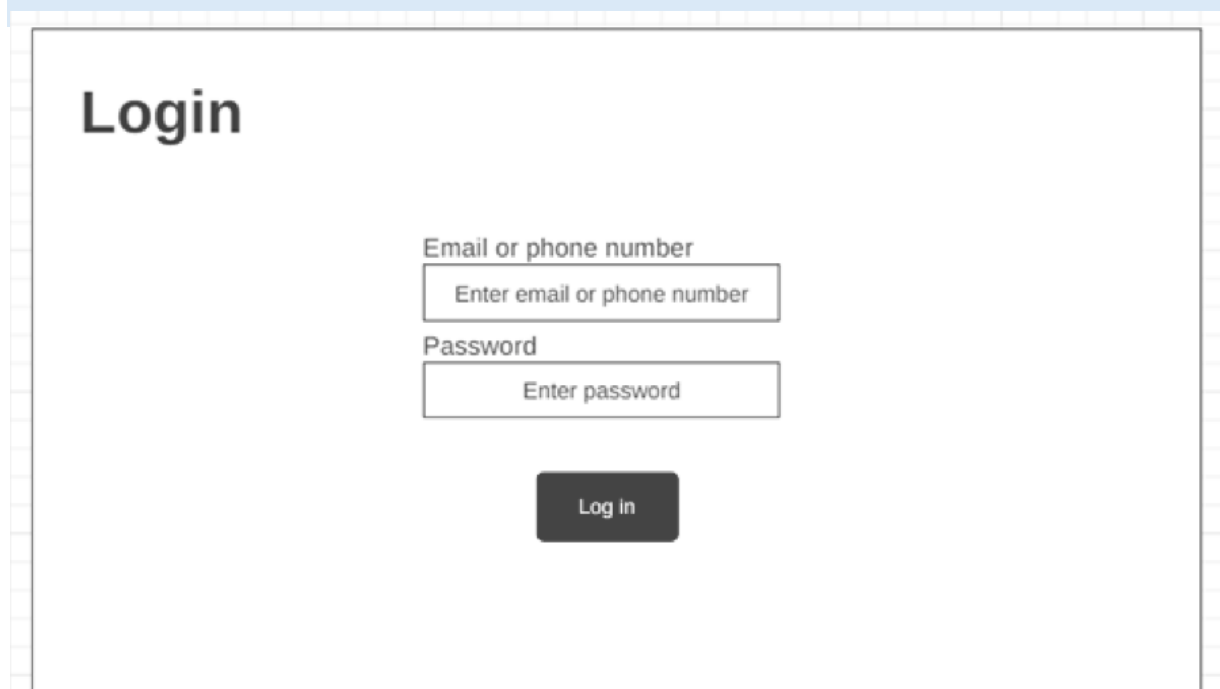
STORY – LOGIN PAGE

As a user,

I'm able to log into my account,

So that I can access my bank accounts.

WIREFRAME



The wireframe shows a login page layout. At the top left, the word "Login" is displayed in a large, bold font. Below it, centered, are two input fields. The first field is labeled "Email or phone number" and contains the placeholder text "Enter email or phone number". The second field is labeled "Password" and contains the placeholder text "Enter password". Below these fields is a dark rectangular button with the text "Log in" in white. The entire form is set against a light gray grid background.

ACCEPTANCE CRITERIA

- User can log in with his credentials.
- Login validation: username (email address/phone number) and password have to be correct, if one of them is wrong send a warning.
- After a successful login the user is brought to the main page.

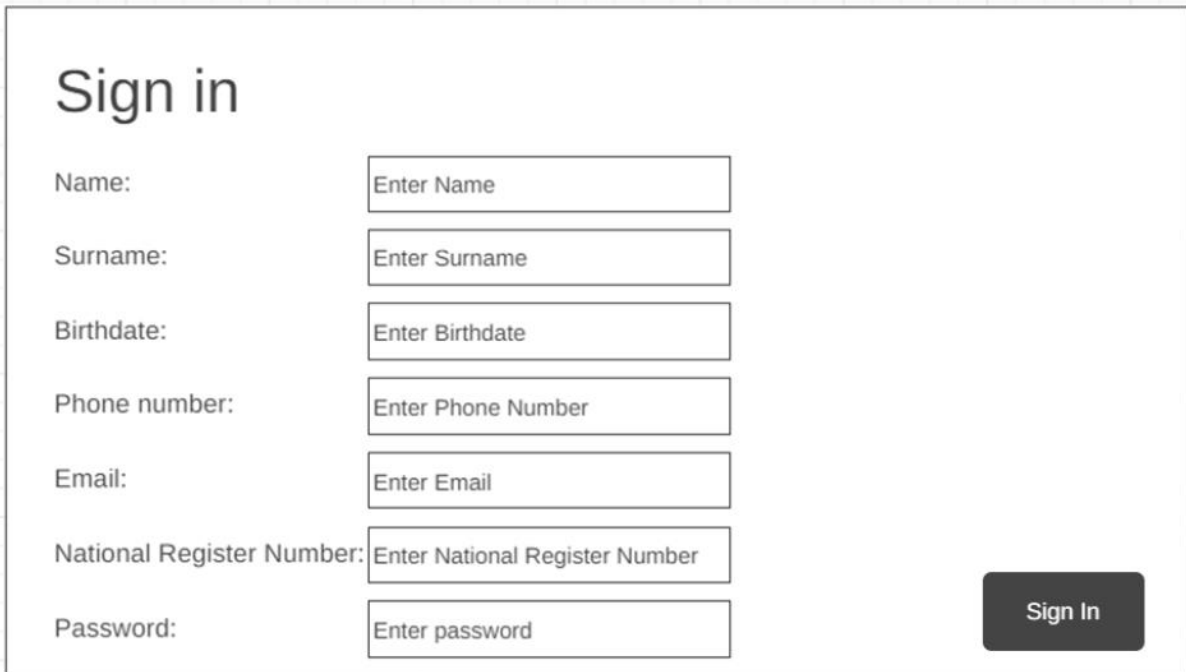
STORY – CREATE ACCOUNT

As a new client,

I'm able to create an account,

So that I can create and manage my back accounts.

WIREFRAME



The wireframe shows a 'Sign in' form with the following fields and a button:

- Sign in** (Title)
- Name:**
- Surname:**
- Birthdate:**
- Phone number:**
- Email:**
- National Register Number:**
- Password:**
- Sign In** (Button)

ACCEPTANCE CRITERIA

- User has to enter his name, surname, birthdate, phone number, email address and national register number.
- Only one account can be created with phone number/email address/national register number.
- User can choose his own password.
- Validation on user information and password.
- User can immediately log in with his email address/phone number and password after creating an account.
- Password needs to be at least 8 characters long, with a number, an uppercase letter, a lowercase letter and a special character.

STORY – UPDATE ACCOUNT

As a user,

I'm able to update my account information,

So that user information is correct.

WIREFRAME

The wireframe shows a 'Settings' page with the title 'Settings' in a large, bold font. Below the title is the subtitle 'Update account settings'. There are four input fields labeled 'Name:', 'Surname:', 'Phone number:', and 'Email:'. Each label is followed by a rectangular input box. Below these fields is a dark gray button with the text 'Confirm' in white.

ACCEPTANCE CRITERIA

- A user can change account information.
- Validation: phone number and email address are all unique, when updating information, these values remain unique.
- User cannot change his name, birth date (age) and national register number.
- After updating information, application automatically changes user information from all accounts that have the user in their contacts.

STORY - DELETE ACCOUNT

As a user,

I'm able to delete my account,

So that everything is deleted.

WIREFRAME

The wireframe shows a settings page titled 'Settings'. Below the title is the section 'Delete account'. Under this section, there is a label 'Enter password to confirm deletion' followed by a text input field containing the placeholder text 'Enter Password'. Below the input field is a dark button labeled 'Confirm'.

ACCEPTANCE CRITERIA

- A user can delete his account, when deleting it the user has to enter his password.
- An account can only be deleted if it only has closed bank accounts or no bank accounts.
- After deleting the account, the application will delete information about the user in all contacts that have the user in their contacts.

STORY – DELETE ACCOUNT V2

As a bank administrator,

I'm able to delete accounts,

So that fraudulent accounts are deleted.

WIREFRAME

The wireframe shows a 'Settings' page with a title 'Settings' in bold. Below the title, there is a label 'Delete user account' followed by a text input field with the placeholder text 'Enter the user email address'. Below this, there is a label '(Optional) Tell the user why you are deleting his account' followed by a large text area with the placeholder text 'Note'. At the bottom left of the form, there is a dark grey button labeled 'Confirm'.

ACCEPTANCE CRITERIA

- A bank administrator can delete accounts, using the email address of the user.
- Validation on email address, if there isn't an account with the email send an error.
- An account can only be deleted if it only has closed bank accounts or no bank accounts.
- After deleting the account, the application will delete information about the user in all contacts that have the user in their contacts.
- An administrator can add a note to tell the user why his account has been deleted.

STORY – CREATE BANK ACCOUNT

As a user,

I'm able to create a new bank account,

So that I can manage multiple accounts.

WIREFRAME

The wireframe shows a form titled "Create new bank account". It contains the following elements:

- A label "Bank account type" followed by a dropdown menu with the text "Type" and a downward arrow.
- A label "Shared bank account" followed by two radio buttons labeled "Yes" and "No".
- A label "Share with" followed by a dropdown menu with the text "Contact" and a downward arrow.
- A label "Name" followed by a text input field with the placeholder text "Name".
- A dark grey button labeled "Confirm" positioned below the "Name" field.

ACCEPTANCE CRITERIA

- A user can create a new bank account and choose what type of bank account it is, select if it is shared or not
- If the bank account is shared, user can choose who he shares it with among his contacts
- When a new bank account is created, status is open
- When a new bank account is created, balance is zero
- Bank account is automatically added when account is created.
- The user can add a nickname for the bank account.
- Number of bank accounts cannot exceed 8.

STORY – VIEW BANK ACCOUNT

As a user,

I'm able to see a certain bank account,

So that I can see my bank account in more detail.

WIREFRAME

The wireframe shows a user interface for viewing a bank account. It features a title 'Bank account name' at the top left. Below the title is a table with six rows of transaction data. To the right of the table, there is a section for account details, including 'Status: Open' and 'Shared contacts: user1, user2, ...'. The table has a small black square icon next to the first row.

Transaction made by user1
+900
-500
+100
-200
-50
-200

Status: Open
Shared contacts: user1, user2, ...

ACCEPTANCE CRITERIA

- User can see all his expenses and income on the bank account.
- The user can see who he shared his accounts with and can see the status of the bank account (open, blocked, frozen and closed).
- If there were transactions made on a shared bank account, the name of the person who made the transaction is visible.
- The user can click on a transaction to see more detailed information.

STORY – UPDATE BANK ACCOUNT

As a user,

I'm able update a bank account,

So that I can change certain aspects of the bank account

WIREFRAME

Bank Account Name Settings

Change contacts:

User1	-
User2	-
User3	+

Freeze bank account: ☐ Freeze

Close bank account: ☐ Close

Enter password

Confirm

ACCEPTANCE CRITERIA

- User can change the status of the bank account to frozen, but cannot unfreeze it.
- The owner of the bank account can remove or add people to the bank account.
- The owner can close the bank account, but the balance has to be zero.
- The application requests user to enter his password before closing the bank account.

STORY – UPDATE BANK ACCOUNT STATUS

As an administrator,

I'm able to update the status of a bank account,

So that I can help the owner when he isn't able to do it himself.

WIREFRAME

The wireframe shows a settings page titled "Bank Account Name Settings Of User0". It contains three rows of controls:

- "Freeze bank account:" with a radio button for "Unfreeze".
- "Close bank account:" with a radio button for "Open".
- "Block bank account:" with radio buttons for "Block" and "Unblock".

Below these controls is a text input field with the placeholder text "Add a note". At the bottom left is a dark "Confirm" button.

ACCEPTANCE CRITERIA

- The administrator can (un)block the bank account
- The administrator can unfreeze the bank account, if the user froze it.
- The administrator can open the bank account, if the user had closed it.

STORY – DELETE BANK ACCOUNT

As an owner,

I'm able to delete a bank account,

So that I don't have to manage it.

WIREFRAME

The wireframe shows a form titled "Bank Account Name Settings". It includes a "Change contacts:" section with a table listing "User1", "User2", and "User3" with minus, minus, and plus icons respectively. Below this are two radio button options: "Freeze bank account:" with "Freeze" selected, and "Close bank account:" with "Close" selected. A text input field labeled "Enter password" is positioned next to the "Close" option. A "Confirm" button is located below the radio buttons. At the bottom, there is a label "Enter password to confirm deletion:" followed by a "Password" input field and a red "Delete Bank account" button.

Bank Account Name Settings							
Change contacts:	<table border="1"><tr><td>User1</td><td>-</td></tr><tr><td>User2</td><td>-</td></tr><tr><td>User3</td><td>+</td></tr></table>	User1	-	User2	-	User3	+
User1	-						
User2	-						
User3	+						
Freeze bank account:	<input type="radio"/> Freeze						
Close bank account:	<input type="radio"/> Close						
	<input type="text" value="Enter password"/>						
<input type="button" value="Confirm"/>							
Enter password to confirm deletion: <input type="text" value="Password"/>							
<input type="button" value="Delete Bank account"/>							

ACCEPTANCE CRITERIA

- The owner can delete a bank account, if the status is closed. When deleting it, the owner has to enter his password.
- After deleting the bank account, the application will remove it from every user that used this bank account.

STORY – DELETE BANK ACCOUNT V2

As an administrator,

I'm able to delete a bank account,

So that fraudulent bank accounts are deleted.

WIREFRAME

The wireframe shows a form titled "Bank Account Name Settings Of User0". It contains three rows of settings, each with a label and two radio buttons: "Freeze bank account:" with "Unfreeze", "Close bank account:" with "Open", and "Block bank account:" with "Block" and "Unblock". Below these is a text input field labeled "Add a note". At the bottom left is a dark grey "Confirm" button, and at the bottom right is a red "Delete Bank account" button.

Bank Account Name Settings Of User0

Freeze bank account: ☐ Unfreeze

Close bank account: ☐ Open

Block bank account: ☐ Block ☐ Unblock

Add a note

Confirm

Delete Bank account

ACCEPTANCE CRITERIA

- The administrator can delete a bank account even if it still has a balance.
- The administrator can add a note to tell the user why it has been deleted.
- When the administrator deletes the bank account, the application will remove it from every user that used this bank account.

STORY – MAIN PAGE

As a user,
I'm able to see all bank accounts,
So that I can manage them on the main page.

WIREFRAME

Main Page

Bank account 1 200 Open	Bank account 3 10000 Open
Bank account 2 500 Frozen	Bank account 4 Balance: 1900 Open

ACCEPTANCE CRITERIA

- The user can see all bank accounts he has or that are shared with him.
- The name, status and balance of the bank accounts are visible on the main page.
- The user can click on the bank accounts that redirect them to a more detailed page of the bank account.

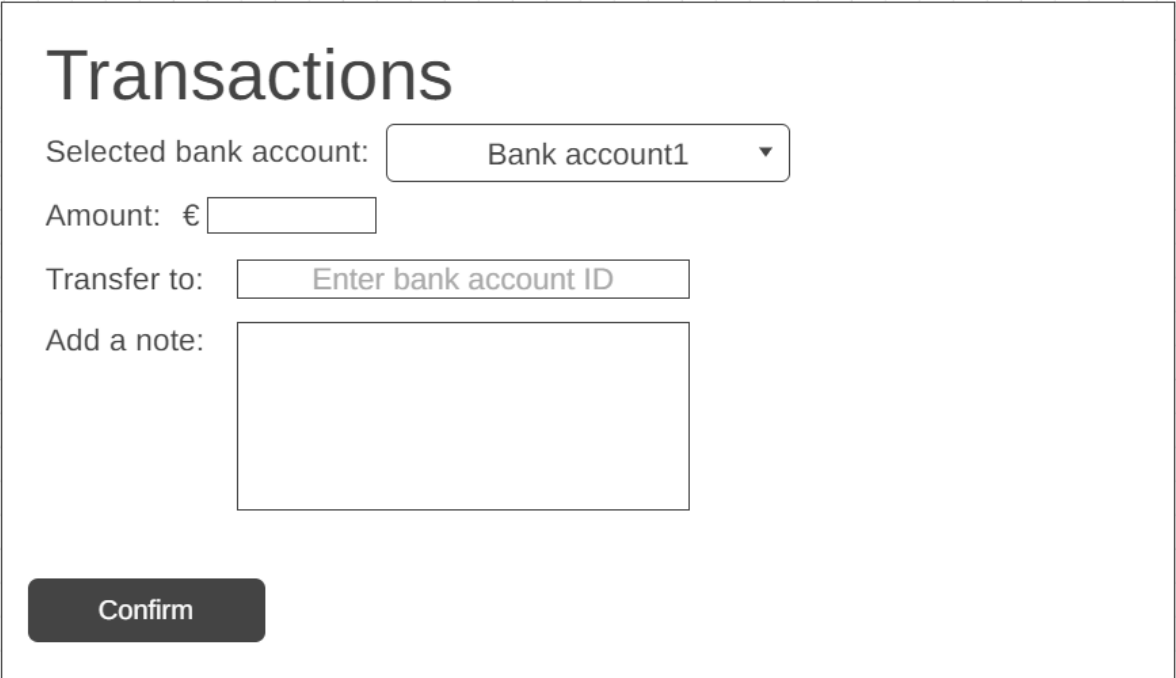
STORY – TRANSACTION PAGE

As a user,

I'm able to access the transaction page,

So that I can make transactions to other accounts

WIREFRAME



The wireframe shows a transaction form titled "Transactions". It includes a dropdown menu for "Selected bank account" with "Bank account1" selected. Below this is a text input for "Amount: €". The "Transfer to:" field is a text input with the placeholder "Enter bank account ID". There is a larger text area for "Add a note:". At the bottom left is a dark "Confirm" button.

ACCEPTANCE CRITERIA

- When the user clicks on the transaction page, he has to select an bank account from which he wishes to make a transaction from.
- The user can enter a number in, but the number cannot be higher than the balance of this bank account or negative.
- The user can transfer money to another account that he owns or to someone else's account. The money is immediately taken from the account and transferred to the other bank account.
- The user can add a note to the transaction.
- The user cannot select a bank account that is frozen, blocked or closed.

STORY – VIEW BUDGET GOAL

As a user,

I'm able to view my budget goal,

So that I have a more deeper understanding.

WIREFRAME

Budget goal

Advice: save money

Goal: € 40 000	Current: € 30 000	8 months left	Current: Save for car
Goal: € 5 000	Achieved	Vacation	
Goal: € 5 000	Not Achieved	Goal 3	

ACCEPTANCE CRITERIA

- The user can see all the budget goals he made in the past.
- Advice is given to the user for his current goal. The application tells the user how much he is allowed to spend/how much he has to save.
- If the set date has passed, the status of budget goal automatically updates.

STORY – CREATE BUDGET GOAL

As a user,

I'm able to create a budget goal,

So that I can easily get advice.

WIREFRAME

The wireframe shows a form titled "New budget goal" on a grid background. The form contains the following elements:

- Title:** "New budget goal" in a large, bold font.
- Budget goal:** A label "Budget goal: €" followed by a text input field containing the placeholder "amount".
- Deadline:** A label "Deadline:" followed by a text input field containing the placeholder "dd/mm/yyyy".
- Include bank accounts:** A label followed by a table with three rows:

Bank account 1	✓
Bank account 2	
Bank account 3	✓
- Budget goal name:** A label followed by a text input field containing the placeholder "Name".
- Budget goal description:** A label followed by a larger text input field containing the placeholder "Description".
- Confirm:** A dark grey button with the text "Confirm".

ACCEPTANCE CRITERIA

- The user can only have one active budget goal at a time.
- Validation on budget goal and date, budget goal cannot be a negative number and date has to be in the future.
- The user can choose which bank accounts he wants to include in his budget goal. The user has to select at least one bank account.
- The user can set a name and description for his budget goal.

STORY – SORTING INCOME PAGE

As a user,

I'm able to access a sorting page of a certain bank account,

So that I can check for more detailed information.

WIREFRAME

Income of Bank account1

Filter by

From user3	+ 2400	11/01/2011
From user2	+ 2400	11/01/2011
From user3	+ 2400	10/10/2010
From user5	+ 2400	01/01/2010

ACCEPTANCE CRITERIA

- The user can sort by certain dates, amounts and ID of bank accounts.
- Validation on date: cannot be in the future.
- Validation on amount: cannot be negative.
- Validation on ID: if no ID has been found return an error.
- The user can see an income overview
 - o Total income overview
 - o Yearly income overview
 - o Monthly income overview

STORY – SORTING EXPENSE PAGE

As a user,

I'm able to access a sorting page of a certain bank account,

So that I can check for more detailed information.

WIREFRAME

Expenses of Bank account1		
Filter by	user name ▼	user3
From user3	- 100	11/01/2011
From user3	- 50	11/01/2011
From user3	- 1000	10/10/2010
From user3	- 500	01/01/2010

ACCEPTANCE CRITERIA

- The user can sort by certain dates, notes, amounts and ID of bank accounts.
- Validation on date: cannot be in the future.
- Validation on amount: cannot be negative.
- Validation on ID: if no ID has been found return an error.
- The user can see an expense overview
 - o Total expense overview
 - o Yearly expense overview
 - o Monthly expense overview

STORY – OVERVIEW CONTACTS

As a user,

I'm able to see all my contacts,

So that I can see who is in my contacts.

WIREFRAME

Contacts

user 2	Edit
user 3	Edit
user 4	Edit
user 5	Edit
+ add new contact	

ACCEPTANCE CRITERIA

- The user can see all his contacts.
- On the contacts page, the user can see the name, surname and a note of his contact.
- When clicking on a contact, a more detailed description is given of the contact.

STORY – ADD CONTACTS

As a user,

I'm able to add contacts,

So that I can easily send them transactions.

WIREFRAME

The wireframe shows a form titled 'Add contact' on a grid background. The form contains the following elements:

- Name:** A text input field with the placeholder text 'Nickname'.
- Email address:** A text input field with the placeholder text 'Email address'.
- Phone number:** A text input field with the placeholder text 'Phone number'.
- User ID:** A text input field with the placeholder text 'ID'.
- Relationship with contact:** A dropdown menu with 'Family' selected and a downward arrow.
- Add contact:** A dark grey button with white text.

ACCEPTANCE CRITERIA

- The user is able to add a contact to his contacts list
- The application asks for name (nickname), ID, email, phone number and relation of the contact with the user.
- Validation on email, phone number and ID: if no user with the same values has been found return an error.
- Validation on email, phone number and ID: if no value has been given, return error. There has to be a value in the correct form.

STORY – UPDATE CONTACTS

As a user,
I'm able to update contacts,
So that I can change information about them.

WIREFRAME

The wireframe shows a form titled "Edit contact information" on a grid background. The form contains four input fields: "Nickname" for Name, "Email address" for Email address, "Phone number" for Phone number, and a dropdown menu for "Relationship with contact" currently showing "Family". A "Confirm" button is located at the bottom left of the form area.

Edit contact information

Name:

Email address:

Phone number:

Relationship with contact:

Confirm

ACCEPTANCE CRITERIA

- The user can edit contact information of users in his contact list.
- Validation on email, phone number and relation.
- ID cannot be edited.
- After editing the contact information is updated immediately.

STORY – DELETE CONTACTS

As a user,
I'm able to delete contacts,
So that I don't have them in my contacts.

WIREFRAME

The wireframe shows a form titled "Edit contact information" on a grid background. The form contains four input fields: "Nickname" for Name, "Email address" for Email address, "Phone number" for Phone number, and a dropdown menu for "Relationship with contact" currently showing "Family". At the bottom left is a dark grey "Confirm" button, and at the bottom right is a red "Delete Contact" button.

Edit contact information

Name:

Email address:

Phone number:

Relationship with contact:

ACCEPTANCE CRITERIA

- A user can delete a contact.
- After deleting a contact, contact information is removed from his contact list.

STORY – VIEW LOANS ON BANK ACCOUNT

As a user,

I'm able to view my loans on a bank account,

So that I am informed about the loans.

WIREFRAME

Loans on bank account name

Next fee: € 500 11/11/1111	Loan1: € 10 000 interest rate: 2,50%
Paid off	Loan 2: € 10 000 interest rate: 2,50%

ACCEPTANCE CRITERIA

- User can view all loans on bank account.
- The user can view the amount that he lend from the bank, interest rate and status.
- If the loan is still to be paid off, the user sees the next payment day and amount.

STORY – CREATE LOAN

As a user,

I'm able to request a new loan,

So that I can borrow money

WIREFRAME

The wireframe shows a rectangular box representing a form. At the top left of the box is the title 'Request new loan' in a large, bold, sans-serif font. Below the title, there are two input fields. The first is labeled 'Amount: €' and contains the placeholder text 'enter amount'. The second is labeled 'Loan type:' and is a dropdown menu with 'Loan type' and a downward arrow. Below these two fields is a dark gray button with the text 'Request Loan' in white. The entire form is set against a light gray background with a grid of small squares.

ACCEPTANCE CRITERIA

- The user can request money from the bank, the user can choose the amount and type.
- Validation on amount: cannot be negative value.
- Loan request is sent to administrator, he decides whether or not to accept the request and suggest an interest rate and end date.

STORY – ACCEPT LOAN REQUEST

As an administrator,

I'm able to deny/accept loan request,

So that we make good deals.

WIREFRAME

Accept loan

User1	€ 10 000	2%	Accept	Deny
User2	€ 10 000	5%	Accept	Deny

ACCEPTANCE CRITERIA

- Administrator can see a list of requested loans, with the user who requested it, the amount and the loan type.
- The administrator can deny or accept the request, after suggesting an interest percentage.
- After the loan has been accepted, the loan is viewable in the user's loan list and the money is deposited in the user's bank account.