

Requirements Document

Wallot

Version 1.0

Introduction

This document is about the Wallot app specifications, providing the developer team informations to the project and implementation, just like to do tests and system homologation.

This introduction is gonna provide the needed informations to make a good use of this document, showing the objectives and conventions that was adopted, beside it has a list of reference to others related documents. This sections is gonna provide the Wallot's app specifications:

1º Chapter is a general describe, a overview of the app and basically is gonna provide you general vision of the app. Describing the users and showing the scope of the project.

2º Chapter describes the functional requirements and specifies all functional requirements of the project, showing the inputs and outputs of each case, priorities, flow event, etc.

3º Chapter describes the non-functional requirements, like security, confiability, usability, hardware and software requirements, performance, etc.

And finally, the **4º Chapter** that shows the user interface.

Chapter 1 - Overview

Wallot is a financial management software that has as target public people that want manage their finances in a more organize and easy access way. Wallot will organize your expenses in a short and long term, showing to where go and comes the user money. It will create forms, stats and will follow any finance planning define by the user, that can follow, edit and create others simultaneous planning. The amazing differential of the app is that many of the user expenses will be imported with the cost, category and time, without the need manual insert this informations.

For decades, Brazilians suffers with the unstable economy and short investment of the govern in finance education. The result of this is more than 63 millions people buried in debts. Is more than the population of Italy. Besides the low rankings in research about knowledge in financial concepts.

Knowing this scenario in that, currently, not all people have time, knowledge or disposition to organize their finances, Wallot primordial purpose is help the user achieve their finance health. In the same scenario that the expenses are frequent and diverse, Wallot aims to inform this expenses and aware the user in a way that he knows where are going his money. Another purpose is an adapt financial planning system, that can attend the planning for the month, year, travels, property and many others, in order that the user could know what he will need to achieve his financial goals.

Chapter 2 - Functionals Requirements

[RF001] User Registration

Author: User

Priority: Essential

Interface: Login and User Registration

Input and pre-conditions: To activate the registration the user need to inform the name, email, password and confirm the password. Or he can login with an Facebook, Google or Twitter Account.

Output and post-conditions: The app will inform if the operation was succeeded or not. If succeeded, the app will return to login screen. If not, the app with stay in the registration screen.

Main Flow Event: When the user tap in “Registre-se” in login screen, the user will go to the registration screen. There the user is gonna provide all necessary information to do the registration and after tap in “Cadastrar”, the Wallot app will create his account.

Secondary Flows: “User already created” - The app will inform that already exists the user and will return the registration screen.

“email or password invalid” - The app will return to the registration screen.

[RF002] Login

Author: User

Priority: Essential

Interface: Login and Main Screen

Input and pre-conditions: To do the login, is necessary inform the user email and password.

Output and post-conditions: performed the login, the user are gonna taken to the main screen.

Main Flow Event: In login screen, if inform an email and a password that is bound to an user, the user will be authenticated and taken to the main screen.

Secondary Flows: if in login screen, the email and/or password informed doesn't exists, the user will receive a message that says “Email and/or password invalid.”

[RF003] Category

Author: Wallot

Priority: Essential

Interface: Navigation Bar, Configuration Screen and Category Screen

Input and pre-conditions: To access the category screen, the user need to access the navigation bar, tap in “Configuração” and tap in “Categorias”.

Output and post-conditions: accessed the category screen, the user can access the categories.

Main Flow Event: The user can access the categories

Secondary Flows: //

[RF004] Subcategory

Author: User

Priority: Desirable

Interface: Navigation Bar, Configuration Screen, Category Screen and “Outros” screen.

Input and pre-conditions: To access the subcategory screen, the user need to access the navigation bar, tap in “Configuração”, tap in “Categorias” and tap in “Outros”.

Output and post-conditions: The user is gonna get access to the subcategories and can create a new one or delete old ones.

Main Flow Event: The user can access the subcategories and create or delete a subcategory.

Secondary Flows: //

[RF005] Change Account Information

Author: User

Priority: Essential

Interface: Navigation Bar, Configuration Screen, “Contas” screen and “Editar Usuário” screen

Input and pre-conditions: To access the change account information screen, the user need to access the navigation bar, tap in “Configuração”, tap in “Contas”, choose the account and tap in it.

Output and post-conditions: The user is gonna access the change account information screen.

Main Flow Event: The user can change his name, email and/or password.

Secondary Flows: If the user change his name, email and/or password to invalid ones, his is gonna get a error message.

[RF006] Disable Account

Author: User

Priority: Essential

Interface: Navigation Bar, Configuration Screen, “Contas” screen

Input and pre-conditions: To access the disable account screen, the user need to access the navigation bar, tap in “Configuração” and tap in “Contas”.

Output and post-conditions: The user is gonna access the disable account screen and his can disable.

Main Flow Event: The user account is gonna be disabled.

Secondary Flows: //

[RF007] Create Budget

Author: User

Priority: Essential

Interface: Navigation Bar, Budget Screen

Input and pre-conditions: To access the create budget screen, the user need to access the navigation bar and tap in “Orçamento”.

Output and post-conditions: //

Main Flow Event: The user is gonna created a Budget.

Secondary Flows: //

[RF008] List Budget

Author: Wallot

Priority: Essential

Interface: ?

Input and pre-conditions: ?

Output and post-conditions: ?

Main Flow Event: ?

Secondary Flows: ?

[RF009] Analyze Budget

Author: Wallot

Priority: Essential

Interface: ?

Input and pre-conditions: ?

Output and post-conditions: ?

Main Flow Event: ?

Secondary Flows: ?

[RF010] Create Transaction

Author: User

Priority: Essential

Interface: Main Screen, Floating Action Button and Transaction Screen

Input and pre-conditions: The user needs to fill all the form. With the description, value, type, category, subcategory, account, etc.

Output and post-conditions: Ifs succeeded, transaction will be maid, if not, a message error will pop up.

Main Flow Event: A transaction will be maid.

Secondary Flows: If some of the blanks is filled in an invalid way, a message error will pop up.