

-----

# Money Tracker

## Test Specification

**Version:** <1.0>

**Date:** <09.11.2015>

**State:** <Draft>

Created by: <Andrei Martinescu>

### Created by

Name: Andrei Martinescu

Date: 09.11.2015

Signature: \_\_\_\_\_

### Approved by

Name: \_\_\_\_\_

Date: \_\_\_\_\_

Signature: \_\_\_\_\_

# 1 History

## 1.1 Document Versions

Version	Date	Author	Changes
<1.0>	09.11.2015	<Andrei Martinescu>	<Initial version>

## 2 Contents

### 2.1 Content

<b>1 History</b> .....	Error! Bookmark not defined.
1.1 Releases .....	<b>Error! Bookmark not defined.</b>
<b>2 Content</b> .....	Error! Bookmark not defined.
2.1 Table of content .....	<b>Error! Bookmark not defined.</b>
2.2 Figures .....	<b>Error! Bookmark not defined.</b>
2.3 Tables .....	<b>Error! Bookmark not defined.</b>
<b>3 Introduction</b> .....	Error! Bookmark not defined.
3.1 Purpose of this document .....	<b>Error! Bookmark not defined.</b>
3.2 Scope, area of applicability .....	<b>Error! Bookmark not defined.</b>
3.3 References .....	<b>Error! Bookmark not defined.</b>
<b>4 Test case specification</b> .....	Error! Bookmark not defined.
4.1 Test report .....	<b>Error! Bookmark not defined.</b>

### 2.2 Figures

None.

### 2.3 Tables

None

### 2.4 Open Items

None.

## **3 Introduction**

**3.1 Purpose of this document**

**3.2 Scope, area of applicability**

**3.3 References**

# Test Case Specification

Test case identifier	#TEST-VALCREATECONF-FUN01#
Title	Create wallet with no amount specified
Description	The application will create a wallet with initial amount of +00.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_NO_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of +00.00 RON.

Test case identifier	#TEST-VALCREATECONF-FUN02#
Title	Create wallet with a positive amount specified.
Description	The application will create a wallet with initial amount of +1000.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet +1000.00
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of +1000.00 RON.

Test case identifier	#TEST-VALCREATECONF-FUN03#
----------------------	----------------------------

Title	Create wallet with a positive amount specified without decimals specified, test adding necessary decimals.
Description	The application will create a wallet with initial amount of +1000.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet +1000
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of +1000.00 RON.

Test case identifier	#TEST-VALCREATECONF-FUN04#
Title	Create wallet with a positive amount specified with more than two decimals specified, test round the amount with two decimals.
Description	The application will create a wallet with initial amount of +1000.125 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet +1000.125
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of +1000.13 RON.

Test case identifier	#TEST-VALCREATECONF-FUN05#
Title	Create wallet with a positive amount specified with one decimals specified, test adding necessary decimals.
Description	The application will create a wallet with initial amount of +1000.5 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet +1000.5
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of +1000.50 RON.

Test case identifier	#TEST-VALCREATECONF-FUN06#
Title	Create wallet with a positive amount specified with leading zeroes, test removing unnecessary starting zeroes.
Description	The application will create a wallet with initial amount of +1000.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet +00001000
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of +1000.00 RON.

Test case identifier	#TEST-VALCREATECONF-FUN07#
Title	Create wallet with a negative amount specified.
Description	The application will create a wallet with initial amount of -1000.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet -1000.00
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of -1000.00 RON.

Test case identifier	#TEST-VALCREATECONF-FUN08#
Title	Create wallet with a negative amount specified without decimals specified, test adding necessary decimals.
Description	The application will create a wallet with initial amount of -1000.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet -1000
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of -1000.00 RON.



Test case identifier	#TEST-VALCREATECONF-FUN09#
Title	Create wallet with a negative amount specified with more than two decimals specified, test round the amount with two decimals.
Description	The application will create a wallet with initial amount of -1000.125 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet -1000.125
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of -1000.13 RON.

Test case identifier	#TEST-VALCREATECONF-FUN010#
Title	Create wallet with a negative amount specified with one decimals specified, test adding necessary decimals.
Description	The application will create a wallet with initial amount of -1000.5 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet -1000.5
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of -1000.50 RON.

Test case identifier	#TEST-VALCREATECONF-FUN011#
Title	Create wallet with a negative amount specified with leading zeroes, test removing unnecessary starting zeroes.
Description	The application will create a wallet with initial amount of +1000.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet -00001000
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of -1000.00 RON.

Test case identifier	#TEST-VALCREATECONF-FUN012#
Title	Create wallet with an amount specified with no sign, test adding the right sign
Description	The application will create a wallet with initial amount of 1000.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_AMOUNT_SPECIFIED</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet 1000.00
Expected Reaction	This will create a file named my.wallet relative to where the command was called, with an initial amount of +1000.00 RON.

Test case identifier	#TEST-VALCREATECONF-FUN013#
Title	Create wallet with an invalid amount specified, test printing error message
Description	The application will print an error message. : error: 12,24 is not a valid initial amount.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_INCORECT_AMOUNT</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet 12,24
Expected Reaction	error: 12,24 is not a valid initial amount.  Creating my.wallet aborted.

Test case identifier	#TEST-VALCREATECONF-FUN014#
Title	Create wallet with a valid amount specified, test printing confirmation message
Description	The application will print a confirmation message. : my.wallet created with the initial amount of +1000.00 RON.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet 1000
Expected Reaction	my.wallet created with the initial amount of +1000.00 RON

Test case identifier	#TEST-VALCREATECONF-FUN015#
Title	Calling an invalid create command will print an error message.
Description	The application will print an error message. : Incorrect command appealed. And the application will print a help menu.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create my.wallet 1000
Expected Reaction	Incorrect command appealed.

Test case identifier	#TEST-VALCREATECONF-FUN016#
Title	Calling an incomplete create command will print an error message.
Description	The application will print an error message. :invalid parameters for create command. The application will print a help menu.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create
Expected Reaction	Invalid parameters for create command.

Test case identifier	#TEST-VALCREATECONF-FUN017#
Title	Trying to create a wallet that already exists.
Description	The application will print an error message. : error: wallet yetother already exists!
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_WALLET_EXISTS</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer. The user already created the yetother wallet on that location.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create yetother -1021.23
Expected Reaction	error: wallet yetother already exists!

Test case identifier	#TEST-VALCREATECONF-FUN018#
Title	Trying to create a wallet that already exists specifying the path for the wallet.
Description	The application will print an error message. : error: wallet C:\learn\yetother already exists!
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_WALLET_EXISTS</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer. The user already created the yetother wallet on that path.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create C:\learn\yetother -1021.23
Expected Reaction	error: wallet C:\learn\yetother already exists!
Test case identifier	#TEST-VALCREATECONF-FUN019#
Title	Create a wallet with a specify path. Test printing confirmation message.

Description	The application will print a confirmation message. : C:\learn\wallet created with the initial amount of +100.00 RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_CREATE_WITH_PATH</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter call create command with no amount specified.  Moneytracker create C:\learn\wallet 100
Expected Reaction	C:\learn\wallet created with the initial amount of +100.00 RON

## INCOME:

Test case identifier	#TEST-VALINCOMECONF-FUN01#
Title	Create an income in the wallet. Test add a new line in wallet with the income specified.
Description	The application will print a confirmation message. Income 'salary' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).  In the wallet the user finds the new line added : time ;+;200.00;salary;RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income 200
Expected Reaction	New line added in wallet : time;+;200.00:salary;RON

Test case identifier	#TEST-VALINCOMECONF-FUN02#
Title	Create an income in the wallet. The user specify a category for amount. Test add a new line in wallet with the income specified.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income 200 -c bonus</p>
Expected Reaction	New line added in wallet : time;+;200.00:bonus;RON

Test case identifier	#TEST-VALINCOMECONF-FUN03#
Title	Create an income in the wallet. The user specify a category for amount. Test add a new line in wallet with the income specified.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income 200 - - category bonus</p>
Expected Reaction	New line added in wallet : time;+;200.00:bonus;RON

Test case identifier	#TEST-VALINCOMECONF-FUN04#
Title	Create an income in the wallet. The user specify a category for amount. Test add a new line in wallet with the income specified.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income -c bonus 200</p>
Expected Reaction	New line added in wallet : time;+;200.00:bonus;RON

Test case identifier	#TEST-VALINCOMECONF-FUN05#
Title	Create an income in the wallet. The user specify a category for amount. Test add a new line in wallet with the income specified.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income - - category bonus 200</p>
Expected Reaction	New line added in wallet : time;+;200.00:bonus;RON



Test case identifier	#TEST-VALINCOMECONF-FUN06#
Title	Create an income in the wallet. Test confirmation message.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_VALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income 200</p>
Expected Reaction	Income 'salary' in an amount of 200.00 RON was registered.

Test case identifier	#TEST-VALINCOMECONF-FUN07#
Title	Create an income in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income 200 –c bonus</p>
Expected Reaction	Income 'bonus' in an amount of 200.00 RON was registered.

Test case identifier	#TEST-VALINCOMECONF-FUN08#
Title	Create an income in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income 200 -- category bonus</p>
Expected Reaction	Income 'bonus' in an amount of 200.00 RON was registered.

Test case identifier	#TEST-VALINCOMECONF-FUN09#
Title	Create an income in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income -- category bonus 200</p>
Expected Reaction	Income 'bonus' in an amount of 200.00 RON was registered.

Test case identifier	#TEST-VALINCOMECONF-FUN10#
Title	Create an income in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Income 'bonus' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;+;200.00;bonus;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income -c bonus 200</p>
Expected Reaction	Income 'bonus' in an amount of 200.00 RON was registered.

Test case identifier	#TEST-VALINCOMECONF-FUN11#
Title	Create an income in the wallet with invalid amount. Test error message.
Description	The application will print an error. Error: income should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_INVALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter income command with an amount specified.</p> <p>Moneytracker income 0</p>
Expected Reaction	Error: income should be higher than 0.

Test case identifier	#TEST-VALINCOMECONF-FUN12#
Title	Create an income in the wallet with invalid amount. Test error message.
Description	The application will print an error. Error: income should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_INVALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -1
Expected Reaction	Error: income should be higher than 0.

Test case identifier	#TEST-VALINCOMECONF-FUN13#
Title	Create an income in the wallet with invalid amount. Test error message.
Description	The application will print an error. Error: parameter for income should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_INVALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income ten
Expected Reaction	Error: parameter for income should be a positive number.

Test case identifier	#TEST-VALINCOMECONF-FUN014#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for income should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income ten -c bonus
Expected Reaction	Error: parameter for income should be a positive number.

Test case identifier	#TEST-VALINCOMECONF-FUN15#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for income should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income ten - - category bonus
Expected Reaction	Error: parameter for income should be a positive number.

Test case identifier	#TEST-VALINCOMECONF-FUN16#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for income should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income - - category bonus ten
Expected Reaction	Error: parameter for income should be a positive number.

Test case identifier	#TEST-VALINCOMECONF-FUN17#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for income should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -c bonus ten
Expected Reaction	Error: parameter for income should be a positive number.

Test case identifier	#TEST-VALINCOMECONF-FUN018#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: income should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -1 -c bonus
Expected Reaction	Error: income should be higher than 0.

Test case identifier	#TEST-VALINCOMECONF-FUN19#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: income should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -1 - - category bonus
Expected Reaction	Error: income should be higher than 0.

Test case identifier	#TEST-VALINCOMECONF-FUN20#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: income should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -- category bonus -1
Expected Reaction	Error: income should be higher than 0.

Test case identifier	#TEST-VALINCOMECONF-FUN21#
Title	Create an income in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: income should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -c bonus -1
Expected Reaction	Error: income should be higher than 0.



Test case identifier	#TEST-VALINCOMECONF-FUN22#
Title	Create an income in the wallet. Test changes in wallet file , test if last line in wallet is an empty line.
Description	The application will add a new line in wallet and create the last line with an empty line.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_WALLET_CHANGES</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income 200
2. Action	Go to path where the wallet was created and open the file. Check the content of file.
Expected Reaction	Expect last line in wallet file is an empty line.

Test case identifier	#TEST-VALINCOMECONF-FUN23#
Title	Create an income in the wallet. Test error message if the wallet couldn't be opened.
Description	The application will print an error message. Error: could not open my.wallet to register transaction
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_WALLET_NOT_OPEN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -c bonus 100
Expected Reaction	Error: could not open my.wallet to register transaction

Test case identifier	#TEST-VALINCOMECONF-FUN24#
Title	Create an income in the wallet. Test error message if the wallet couldn't be opened.
Description	The application will print an error message. Error: could not open C:\learn\mywallet to register transaction
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_WALLET_NOT_OPEN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income -c bonus 100
Expected Reaction	. Error: could not open C:\learn\mywallet to register transaction

Test case identifier	#TEST-VALINCOMECONF-FUN25#
Title	Create an income in the wallet with no amount specified. Test error message.
Description	The application will print an error message. Error: no amount specified for 'income'.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income
Expected Reaction	The application will print an error message. Error: no amount specified for 'income'.

Test case identifier	#TEST-VALINCOMECONF-FUN25#
Title	Create an income in the wallet and add the income in default wallet. Test error message if the moneytracker.config couldn't be opened.
Description	The application will print an error message. Error: could not open configuration 'moneytracker.config'.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_INCOME_WALLET_NOT_OPEN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income 100
Expected Reaction	Error: could not open configuration 'moneytracker.config'.

## SPENDING:

Test case identifier	#TEST-VALSPENDINGCONF-FUN01#
Title	Create spend in the wallet. Test add a new line in wallet with the spending specified.
Description	<p>The application will print a confirmation message. Spend 'other' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;other;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend 200</p>
Expected Reaction	New line added in wallet : time;-;200.00:other;RON

Test case identifier	#TEST- VALSPENDINGCONF-FUN02#
Title	Create spend in the wallet. The user specify a category for amount. Test add a new line in wallet with the spending specified.
Description	<p>The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;food;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend 200 -c food</p>
Expected Reaction	New line added in wallet : time-+;200.00:food;RON

Test case identifier	#TEST- VALSPENDINGCONF-FUN03#
Title	Create spend in the wallet. The user specify a category for amount. Test add a new line in wallet with the spending specified.
Description	The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).  In the wallet the user finds the new line added : time ;-;200.00;food;RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend 200 - - category food
Expected Reaction	New line added in wallet : time;-;200.00:food;RON

Test case identifier	#TEST- VALSPENDINGCONF-FUN04#
Title	Create spend in the wallet. The user specify a category for amount. Test add a new line in wallet with the spending specified.
Description	The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).  In the wallet the user finds the new line added : time ;-;200.00;food;RON
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -c food 200
Expected Reaction	New line added in wallet : time;-;200.00:food;RON

Test case identifier	#TEST- VALSPENDINGCONF-FUN05#
Title	Create spend in the wallet. The user specify a category for amount. Test add a new line in wallet with the spending specified.
Description	<p>The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;food;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend -- category food 200</p>
Expected Reaction	New line added in wallet : time ;-;200.00:food;RON

Test case identifier	#TEST- VALSPENDINGCONF-FUN06#
Title	Create spend in the wallet. Test confirmation message.
Description	<p>The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;food;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_VALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend 200</p>
Expected Reaction	Spending 'other' in an amount of 200.00 RON was registered.

Test case identifier	#TEST- VALSPENDINGCONF-FUN07#
Title	Create spend in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;spend;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend 200 -c food</p>
Expected Reaction	Spending 'food' in an amount of 200.00 RON was registered.

Test case identifier	#TEST- VALSPENDINGCONF-FUN08#
Title	Create spend in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;food;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend 200 - - category food</p>
Expected Reaction	Spending 'food' in an amount of 200.00 RON was registered.

Test case identifier	#TEST- VALSPENDINGCONF-FUN09#
Title	Create spend in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;food;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend - - category bonus 200</p>
Expected Reaction	Spending 'food' in an amount of 200.00 RON was registered.

Test case identifier	#TEST- VALSPENDINGCONF-FUN10#
Title	Create spend in the wallet with a category specified. Test confirmation message.
Description	<p>The application will print a confirmation message. Spend 'food' in an amount of 200.00 RON was registered. (On new line will be printed the Transaction time ).</p> <p>In the wallet the user finds the new line added : time ;-;200.00;food;RON</p>
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	<p>Go to command line and enter spend command with an amount specified.</p> <p>Moneytracker spend -c bonus 200</p>
Expected Reaction	Spending 'food' in an amount of 200.00 RON was registered.



Test case identifier	#TEST- VALSPENDINGCONF-FUN11#
Title	Create spend in the wallet with invalid amount. Test error message.
Description	The application will print an error. Error: spend should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_INVALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend 0
Expected Reaction	Error: spend should be higher than 0.

Test case identifier	#TEST-VALINCOMECONF-FUN12#
Title	Create spend in the wallet with invalid amount. Test error message.
Description	The application will print an error. Error: spend should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_INVALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -1
Expected Reaction	Error: spend should be higher than 0.

Test case identifier	#TEST- VALSPENDINGCONF-FUN13#
Title	Create spend in the wallet with invalid amount. Test error message.

Description	The application will print an error. Error: parameter for spend should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_INVALID_COMMAND_MESSAGE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend ten
Expected Reaction	Error: parameter for spend should be a positive number.

Test case identifier	#TEST- VALSPENDINGCONF-FUN014#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for spend should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend ten -c food
Expected Reaction	Error: parameter for spend should be a positive number.

Test case identifier	#TEST- VALSPENDINGCONF-FUN15#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for spend should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend ten - - category food
Expected Reaction	Error: parameter for spend should be a positive number.

Test case identifier	#TEST- VALSPENDINGCONF-FUN16#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for spend should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend - - category food ten
Expected Reaction	Error: parameter for spend should be a positive number.

Test case identifier	#TEST- VALSPENDINGCONF-FUN17#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: parameter for spend should be a positive number.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -c bonus food
Expected Reaction	Error: parameter for spend should be a positive number.

Test case identifier	#TEST- VALSPENDINGCONF-FUN018#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: spend should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -1 -c food
Expected Reaction	Error: spend should be higher than 0.

Test case identifier	#TEST- VALSPENDINGCONF-FUN19#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: spend should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -1 - - category food
Expected Reaction	Error: spend should be higher than 0.

Test case identifier	#TEST- VALSPENDINGCONF-FUN20#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: spend should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend - - category food -1
Expected Reaction	Error: spend should be higher than 0.

Test case identifier	#TEST- VALSPENDINGCONF-FUN21#
Title	Create spend in the wallet with invalid amount with category specified. Test error message.
Description	The application will print an error message. Error: spend should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_SPECIFIED_CATEGORY_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -c food -1
Expected Reaction	Error: spend should be higher than 0.

Test case identifier	#TEST- VALSPENDINGCONF-FUN22#
Title	Create spend in the wallet. Test changes in wallet file, test if last line in wallet is an empty line.
Description	The application will add a new line in wallet and create the last line with an empty line.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_WALLET_CHANGES</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend 200
2. Action	Go to path where the wallet was created and open the file. Check the content of file.
Expected Reaction	Expect last line in wallet file is an empty line.

Test case identifier	#TEST-VALSPENDINGCONF-FUN23#
Title	Create spend in the wallet. Test error message if the wallet couldn't be opened.
Description	The application will print an error message. Error: could not open my.wallet to register transaction
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_WALLET_NOT_OPEN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -c food 100
Expected Reaction	Error: could not open my.wallet to register transaction

Test case identifier	#TEST-VALSPENDINGCONF-FUN24#
Title	Create spend in the wallet. Test error message if the wallet couldn't be opened.
Description	The application will print an error message. Error: could not open C:\learn\mywallet to register transaction
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_WALLET_NOT_OPEN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend -c food 100
Expected Reaction	Error: could not open C:\learn\mywallet to register transaction

Test case identifier	#TEST- VALSPENDINGCONF-FUN25#
Title	Create spend in the wallet with no amount specified. Test error message.
Description	The application will print an error message. Error: no amount specified for 'spend'.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend
Expected Reaction	The application will print an error message. Error: no amount specified for 'spend'.

Test case identifier	#TEST- VALSPENDINGCONF-FUN25#
Title	Create spend in the wallet and add the spend in default wallet. Test error message if the moneytracker.config couldn't be opened.
Description	The application will print an error message. Error: could not open configuration 'moneytracker.config'.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_SPENDING_WALLET_NOT_OPEN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend 100
Expected Reaction	Error: could not open configuration 'moneytracker.config'.

**Balance:**



Test case identifier	#TEST- VALBALANCECONF-FUN01#
Title	Create balance. Test the balance from wallet.
Description	The application will print a confirmation message. The balance will be printed on the command line.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_BALANCE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter balance command.  Moneytracker balance
Expected Reaction	Balance for my.wallet is +90.00 RON

Test case identifier	#TEST- VALBALANCECONF-FUN02#
Title	Create balance. Test the confirmation message.
Description	The application will print a confirmation message. The balance will be printed on the command line.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_BALANCE_VALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter balance command.  Moneytracker balance
Expected Reaction	Balance for my.wallet is +90.00 RON

Test case identifier	#TEST- VALBALANCECONF-FUN03#
Title	Create invalid balance command. Test the error message.
Description	The application will print an error message. Error: invalid parameters.
Test technique	Functional
Trace from	<i>#REQ_FUN_SRS_BALANCE_INVALID_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter balance command.  Moneytracker balance
Expected Reaction	Error: invalid parameters.

Test case identifier	#TEST- VALSAFETYCONF-FUN01#
Title	Check safety of personal data from computer in case of application crash.
Description	In case of application crash the personal data from the user's computer will be safety. The application will not affect the data.
Test technique	Functional
Trace from	<i>#REQ_SEC_CRS_PERSONAL_SAFETY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter huge data commands and try to crash the application
Expected Reaction	The data from the computer and the files from the application are not affected

Test case identifier	#TEST- VALCREATEDATACONF-FUN01#
Title	Check the correct create command.
Description	In case of entering a correct create command the application will execute the command
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_STRUCT</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter a correct create command
Expected Reaction	The application will execute the command.

Test case identifier	#TEST- VALCREATEDATACONF-FUN02#
Title	Create wallet with valid parameters. Test if last line in wallet is an empty line
Description	In case of entering a correct create command the application will execute the command
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_WALLET_FILE_STRUCT</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter a correct create command. Enter wallet.
Expected Reaction	The last line from wallet is an empty line.

Test case identifier	#TEST- VALCREATEDATACONF-FUN03#
Title	Create wallet with valid parameters. Test if application created the wallet with the name specify by user.
Description	In case of entering a correct create command the application will execute the command and create a wallet with name specify by user
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_NAME_OF_WALLET</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter a correct create command. Check name of wallet.
Expected Reaction	The name of wallet is the same with the name specified by user in the create command.

Test case identifier	#TEST- VALCREATEDATACONF-FUN04#
Title	Create wallet with valid parameters. Test if application created the wallet relative to where the command was called from command line.
Description	In case of entering a correct create command the application will execute the command and create a wallet with name specify by user
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_LOCATION_NO_PATH</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter a correct create command. Check the path of wallet.
Expected Reaction	The path of wallet is the same with the path of where the command was called from the command line.

Test case identifier	#TEST- VALCREATEDATACONF-FUN05#
Title	Create wallet with valid parameters and a path specified. Test if application created the wallet in the path specified by user.
Description	In case of entering a correct create command the application will execute the command and create a wallet with name specify by user
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_LOCATION_PATH_SPECIFIED_LOCATION</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter a correct create command. Check the path of wallet.
Expected Reaction	The path of wallet is the same with the path entered by user in command line.

Test case identifier	#TEST- VALCREATEDATACONF-FUN06#
Title	Create wallet with valid parameters and a path specified. Test if the default currency from the first line is RON.
Description	In case of entering a correct create command the application will execute the command and create a wallet with name specify by user
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_DEFAULT_CURRENCY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter a correct create command. Open the wallet. Check the default currency.
Expected Reaction	The default currency specified in the wallet is RON

Test case identifier	#TEST- VALCREATEDATACONF-FUN06#
Title	Create wallet. Test adding necessary decimals.
Description	The application will create a wallet with initial amount of +100.00 RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_AMOUNT_LIMITATION</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line .  Moneytracker create mywallet 100  Open the wallet. Check the amount.
Expected Reaction	The amount printed in mywallet has two decimals. +100.00

Test case identifier	#TEST- VALCREATEDATACONF-FUN07#
Title	Create wallet. Test adding necessary decimals.
Description	The application will create a wallet with initial amount of +100.50 RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_AMOUNT_LIMITATION</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line .  Moneytracker create mywallet 100.5  Open the wallet. Check the amount.
Expected Reaction	The amount printed in mywallet has two decimals. +100.50

Test case identifier	#TEST- VALCREATEDATACONF-FUN08#
Title	Create wallet. Test adding necessary decimals.
Description	The application will create a wallet with initial amount of +100.55 RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_AMOUNT_LIMITATION</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line .  Moneytracker create mywallet 100.55  Open the wallet. Check the amount.
Expected Reaction	The amount printed in mywallet has two decimals. +100.55

Test case identifier	#TEST- VALCREATEDATACONF-FUN09#
Title	Create wallet. Test adding necessary decimals.
Description	The application will create a wallet with initial amount of +100.55 RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_AMOUNT_LIMITATION</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line .  Moneytracker create mywallet 100.549  Open the wallet. Check the amount.
Expected Reaction	The amount printed in mywallet has two decimals. +100.55

Test case identifier	#TEST- VALCREATEDATACONF-FUN09#
Title	Create wallet. Test correct amount format.
Description	The application will print an error message. Error: 100,549 is not a valid initial amount.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_CREATE_AMOUNT_LIMITATION_DOT</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line.  Moneytracker create mywallet 100,549
Expected Reaction	Error: 100,549 is not a valid initial amount.

Test case identifier	#TEST- VALINCOMEDATACONF-FUN01#
Title	Create income. Test correct command format.
Description	The application will print confirmation message. Test that on a correct command income the application will execute the command.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_INCOME_VALID_COMMAND_STRUCTURE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line.  Moneytracker income 200
Expected Reaction	The application execute the valid income command.



Test case identifier	#TEST- VALINCOMEDATACONF-FUN02#
Title	Create an income in the wallet with invalid amount. Test error message.
Description	The application will print an error. Error: income should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_ DAT_ SRS_ INCOME_ AMOUNT_ VALID</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified.  Moneytracker income 0
Expected Reaction	Error: income should be higher than 0.

Test case identifier	#TEST- VALINCOMEDATACONF-FUN03#
Title	Create an income in the wallet with valid amount with two decimals. Test confirmation message.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_ DAT_ SRS_ INCOME_ AMOUNT_ VALID_ DECIMALS</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified with two decimals.  Moneytracker income 100.25
Expected Reaction	Expect the command is a valid command. Expect the income in wallet is 100.25

Test case identifier	#TEST- VALINCOMEDATACONF-FUN04#
Title	Create an income in the wallet with valid amount. Test the default currency added for the income in the wallet.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_INCOME_DEFAULT_CURRENCY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified Moneytracker income 100.25
Expected Reaction	Expect the default currency for income is RON.  Time;+;100.25;salary;RON

Test case identifier	#TEST- VALINCOMEDATACONF-FUN05#
Title	Create an income in the wallet with valid amount. Test the time added for the income in the wallet.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_INCOME_TIME</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified Moneytracker income 100.25
Expected Reaction	Expect the time added in wallet is the right time for transaction.  Time;+;100.25;salary;RON

Test case identifier	#TEST- VALINCOMEDATACONF-FUN06#
Title	Create an income in the wallet with valid amount. Test the sign for income. The correct sign for income is +.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_INCOME_DEFAULT_SIGN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified Moneytracker income 100.25
Expected Reaction	Expect the sign added for income is +.  Time;+;100.25;salary;RON

Test case identifier	#TEST- VALINCOMEDATACONF-FUN07#
Title	Create an income in the wallet with valid amount. Test the default category added in the wallet.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_INCOME_DEFAULT_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified Moneytracker income 100.25
Expected Reaction	Expect the default category for income is salary.  Time;+;100.25;salary;RON

Test case identifier	#TEST- VALINCOMEDATACONF-FUN08#
Title	Create an income in the wallet with valid amount and a specified category. Test the category added in the wallet.
Description	The application will print a confirmation message. Time;+;100.25;bonus;RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_INCOME_DEFAULT_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command with an amount specified Moneytracker income 100.25 –c bonus
Expected Reaction	Expect the category added in the wallet is same with category entered by user in the command line.  Time;+;100.25;bonus;RON

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN01#
Title	Create spend. Test correct command format.
Description	The application will print confirmation message. Test that on a correct command spending the application will execute the command.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_SPENDING_VALID_COMMAND_STRUCTURE</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line.  Moneytracker spend 200
Expected Reaction	The application execute the valid spend command.

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN02#
Title	Create spend in the wallet with invalid amount. Test error message.
Description	The application will print an error. Error: spend should be higher than 0.
Test technique	Functional
Trace from	<i>#REQ_ DAT_ SRS_ SPENDING_ AMOUNT_ VALID</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified.  Moneytracker spend 0
Expected Reaction	Error: spend should be higher than 0.

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN03#
Title	Create spend in the wallet with valid amount with two decimals. Test confirmation message.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_ DAT_ SRS_ SPENDING_ AMOUNT_ VALID_ DECIMALS</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified with two decimals.  Moneytracker spend 100.25
Expected Reaction	Expect the command is a valid command. Expect the spend in wallet is 100.25

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN04#
Title	Create spend in the wallet with valid amount. Test the default currency added for spend in the wallet.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_SPENDING_DEFAULT_CURRENCY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter SPEND command with an amount specified Moneytracker SPEND 100.25
Expected Reaction	Expect the default currency for SPEND is RON.  Time;-;100.25;other;RON

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN05#
Title	Create spend in the wallet with valid amount. Test the time added for spend in the wallet.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_SPENDING_TIME</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified Moneytracker spend 100.25
Expected Reaction	Expect the time added in wallet is the right time for transaction.  Time;-;100.25;other;RON

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN06#
Title	Create spend in the wallet with valid amount. Test the sign for spend. The correct sign for income is -.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_SPENDING_DEFAULT_SIGN</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified  Moneytracker spend 100.25
Expected Reaction	Expect the sign added for income is -.  Time;-;100.25;other;RON

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN07#
Title	Create spend in the wallet with valid amount. Test the default category added in the wallet.
Description	The application will print a confirmation message.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_SPENDING_DEFAULT_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified  Moneytracker spend 100.25
Expected Reaction	Expect the default category for income is other.  Time;-;100.25;other;RON

Test case identifier	#TEST- VALISPENDINGDATACONF-FUN08#
Title	Create spend in the wallet with valid amount and a specified category. Test the category added in the wallet.
Description	The application will print a confirmation message. Time;-;100.25;other;RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_SPENDING_DEFAULT_CATEGORY</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter spend command with an amount specified  Moneytracker spend 100.25 –c food
Expected Reaction	Expect the category added in the wallet is same with category entered by user in the command line.  Time;-;100.25;food;RON

Test case identifier	#TEST- VALIBALANCEDATACONF-FUN01#
Title	Create balance. Test the response of application on a correct balance command.
Description	The application will print a confirmation message. Balance for my.wallet is +900.00 RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_BALANCE_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter balance command  Moneytracker balance
Expected Reaction	Expect the application response correct on a valid balance command



Test case identifier	#TEST- VALIBALANCEDATACONF-FUN02#
Title	Create balance. Test the response of application on a correct balance command.
Description	The application will print a confirmation message. Balance for my.wallet is +900.00 RON
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_BALANCE_COMMAND</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter balance command  Moneytracker balance  Open config , read default wallet , open wallet and calculate the balance
Expected Reaction	Expect the application will read the config file and calculate the balance for the default wallet from the config.

Test case identifier	#TEST- VALDECIMALSDATACONF-FUN01#
Title	Create income. Test the adding decimals
Description	The application will print a confirmation message. Income 'salary' in an amount of 200.00 RON was registered.
Test technique	Functional
Trace from	<i>REQ_DAT_SRS_ADD_DECIMALS</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command  Moneytracker income 200
Expected Reaction	Income 'salary' in an amount of 200.00 RON was registered.

Test case identifier	#TEST- VALREMOVEZEROESDATACONF-FUN01#
Title	Create income. Test the remove zeroes
Description	The application will print a confirmation message. Income 'salary' in an amount of 200.00 RON was registered.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_REMOVE_ZEROES</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line and enter income command  Moneytracker income 00000000000000000200
Expected Reaction	Income 'salary' in an amount of 200.00 RON was registered.

Test case identifier	#TEST- VALONECONFIGDATACONF-FUN01#
Title	Test if application has more than one config file in the application directory.
Description	The application must have only one config file in application directory
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_ONE_CONFIG</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to application directory an find the config file
Expected Reaction	Only one config file could be found in application directory

Test case identifier	#TEST- VALWALLETCONF-FUN01#
Title	Test if application has created the wallet on computer in the path specified or in the path relative to where the command line was opened.
Description	The application must create the wallet
Test technique	Functional
Trace from	<i>#REQ_DAT_CRS_WALLET</i>
Initial State	There is a command line interpreter opened, with the same path configured as the application exe file stored on the computer.
1. Action	Go to command line. Add create command  Moneytracker create my.wallet 2000
Expected Reaction	Check if the wallet was created and has the correct format

Test case identifier	#TEST- VALCONFIGCONF-FUN01#
Title	Test if the config file from moneytracker application has the correct format
Description	The application must have only one config file.
Test technique	Functional
Trace from	<i>#REQ_DAT_SRS_MONEYTRACKER.CONFIG</i>
Initial State	There is a folder of application. Check for config file.
1. Action	Check for config file. Open the config file
Expected Reaction	<p>The config file contains:</p> <pre> default_wallet = default_currency = default_income_category = default_spending_category = currencies = rate_EUR RON = rate RON EUR = rate_USD RON = rate EUR USD = </pre>

	<p>One or more fields from config file can be configured with correct values</p> <p>Example:</p> <pre> default_wallet = my.wallet default_currency = RON default_income_category = salary default_spending_category = other currencies = RON, EUR, USD rate_EUR_RON = 4.42 rate_RON_EUR = 0.23 rate_USD_RON = 3.92 rate_EUR_USD = 1.13 </pre>
--	---

Test case identifier	#TEST- VALPERFORMANCECONF-FUN01#
Title	Test if the application was installed correctly on the computer.
Description	The application must create the wallet
Test technique	Functional
Trace from	<i>#REQ_PER_SRS_SYSTEM_DEPENDABILITY</i>
Initial State	There is a user PC for installing the application. Install the application
1. Action	Install the application. Run the application
Expected Reaction	The application installed successfully.

Test case identifier	#TEST- VALPERFORMACECONF-FUN02#
Title	Test the application space required for installing on the hard drive.
Description	The application must create the wallet
Test technique	Functional
Trace from	<i>#REQ_PER_SRS_HARD_DRIVE_SPACE</i>
Initial State	There is a user PC for installing the application. Install the application. Check the space required by the application.
1. Action	Install the application. Check the space required for the application to store on the hard drive
Expected Reaction	The application space required to store on the hard drive is less than 10 MB

Test case identifier	#TEST- VALPERFORMACECONF-FUN03#
Title	Test the reliability of the application
Description	The application not crash on huge commands data entered.
Test technique	Functional
Trace from	<i>#REQ_PER_CRS_RELIABILITY</i>
Initial State	There is a user PC for installing the application. Install the application. Enter huge data commands
1. Action	Enter 50 valid or invalid commands
Expected Reaction	The application has a reliability of 100%

Test case identifier	#TEST- VALPERFORMACECONF-FUN04#
Title	Test the availability of the application
Description	The application is available every time the computer is opened and the user runs the application
Test technique	Functional
Trace from	<i>#REQ_PER_SRS_AVAILABILITY</i>
Initial State	Enter in the application file.
1. Action	Open the computer. Open the application
Expected Reaction	The application has a availability of 100%

Test case identifier	#TEST- VALPERFORMACECONF-FUN05#
Title	Test the response time of the application
Description	The response time of the application is less than 2 seconds for every command.
Test technique	Functional
Trace from	<i>#REQ_PER_SRS_RESPONSE_TIME</i>
Initial State	Enter in the application file.
1. Action	Open the computer. Open the application. Enter command
Expected Reaction	The application response on the command in less than 2 seconds.

Test case identifier	#TEST- VALAOTHERCONF-FUN01#
Title	Test the portability of the application
Description	Install the application on more than one computer
Test technique	Functional
Trace from	<i>#REQ_OTH_SRS_PORTABILITY</i>
Initial State	Open the computers
1. Action	Install the application on the computers.
Expected Reaction	The application runs correctly on computers

Test case identifier	#TEST- VALAOTHERCONF-FUN02#
Title	Test the usability of the application
Description	Install the application on the computer
Test technique	Functional
Trace from	<i>#REQ_OTH_SRS_USABILITY</i>
Initial State	Open the computers. Open the application
1. Action	Install the application on the computers. Read application commands
Expected Reaction	The user should be able to learn the application in less than 12 hours.

Test case identifier	#TEST- VALAOTHERCONF-FUN03#
Title	Test the extendibility of the application
Description	Install the application on the computer.
Test technique	Functional
Trace from	<i>#REQ_OTH_SRS_EXTENDIBILITY</i>
Initial State	Open the computers. Open the application.
1. Action	Install the application on the computers. Read application commands.
Expected Reaction	The application should be easy to extend. The code should be written in a way that favors implementation of new functions.

Test case identifier	#TEST- VALAOTHERCONF-FUN04#
Title	Test the testability of the application
Description	Install the application on the computer.
Test technique	Functional
Trace from	<i>#REQ_OTH_SRS_TESTABILITY</i>
Initial State	Open the computers. Open the application.
1. Action	Install the application on the computers. Read application commands. Make manual test.
Expected Reaction	The application should support testing. The code should be written in a way that allows testing, and has a high testability coverage.



## 4.1 Test Report

Test Case ID	Date Tested	Tester	Pass /Fail	Summary of Defect	Closed prior to Production Release?	Comments