Lab4: design

Goal of this lab is to practice the transition from requirements to design.

Consider the LaTazza application (see requirements at end of this document)

Propose a design capable of satisfying the requirements. Feel free to work alone or in team.

The design should be formalized in a document, the template of the document is in file TemplateDesignDocument.md

For diagrams (class diagram, package diagram, ecc) use tools (StarUML, Astah, or else) or plantUml

Update Official Requirements Document.md
Maurizio Morisio authored 1 year ago

aaa8f8b3

Official Requirements Document.md 11.3 KB

Official Requirements Document

Authors: Maurizio Morisio, Luca Ardito, Riccardo Coppola

Date: 29/05/2019

Version: 5

Change history

Version	Changes
2	Fixed defect in scenario 2: precondition was wrong
	Fixed defect in scenario format: added post conditions
	Fixed defect in use case 2: variants text canceled
3	Fixed defect in [use case 3]: recharge only positive
	Added Non functional requirement NFR5
4	Fixed defect in [use case 3]: post condition is on Colleague account, and LaTazza
	Fixed defect in [use case 1]: post condition is on Colleague account, not LaTazza
5	Fized defect in [use case 4]: precondition modified

Contents

Abstract

Some colleagues share a coffee machine in a common space for breaks. Capsules are left aside the machine. Whoever uses a capsule writes this down in a notebook left aside the coffee machine. One of the colleagues (called □□*administrator*□□) copies consumptions in an excel sheet and collects money from colleagues to (re)order capsules.

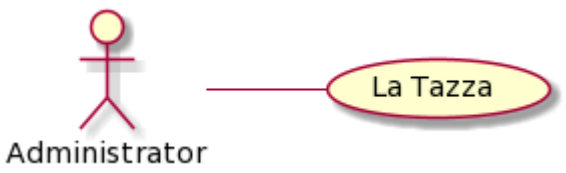
One of the colleagues, the □□*hacker*□, volunteers to develop a simple application to support the administrator. To keep things simple the application is standalone, and is meant to be used only by the administrator. Possibly the administrator role is taken by different colleagues over time. Instead of the notebook, the hacker sets up a WhatsApp group. Whenever a colleague uses a capsule, he sends a message to the group.

Stakeholders

Stakeholder name	**Description**
Administrator	Uses the application to manage purchase and sharing of capsules in a small office (5-10 people)
Colleagues	Do not use the application directly. They are interested in a correct accounting of capsule consumption and payments

Context Diagram and interfaces

Context Diagram



Interfaces

Actor	**Logical Interface**	**Physical Interface**
Administrator	GUI	Screen, keyboard

Stories and personas

John works in the office and has an inclination for order and discipline. So he volunteers to keep track of what happens on the coffee machine. When a capsule of a certain type is close to sold out John reorders. Capsules can be purchased only in multiples of a minimum quantity (ex 40). On the money side, John computes how much each colleague should pay for what she has consumed. In practice every colleague has a virtual account, initially charged with a reasonable amount of cash given to John (ex 10 euros). John maintains this account and asks to recharge it when close to zero. Physically the money for capsules is in cash, and John manages it as if it where his personal money.

All colleagues trust each other, so negative accounts are allowed. John uses his personal money if needed to reorder capsules.

John is happy to do this work, but would like to hand over to someone else after a certain amount of time. In any case when he is on vacation another colleague takes over temporarily.

John would like to have a simple way to report (ex every one or two weeks) to each colleague all expenses and consumption for LaTazza. Without this report nobody has a clear idea of how much was the consumption.

Mary works in the office and likes to share time with her colleagues in front of the coffee machine. On the side of the coffee machine is a tray with capsules. When she wants a coffee (or else) she takes a capsule from the tray and tells to John. To do so she uses a whatsapp group (LaTazza friends).

Mr. Guest is a visitor in the office. As such he does not have the privilege of having an account. In case he can directly pay the capsule, cash, to John.

Functional and non functional requirements

Functional Requirements

ID	**Description**
FR1	Record that a colleague has used n capsules of a certain type, update his account
FR2	Record that a visitor has used n capsules of a certain type

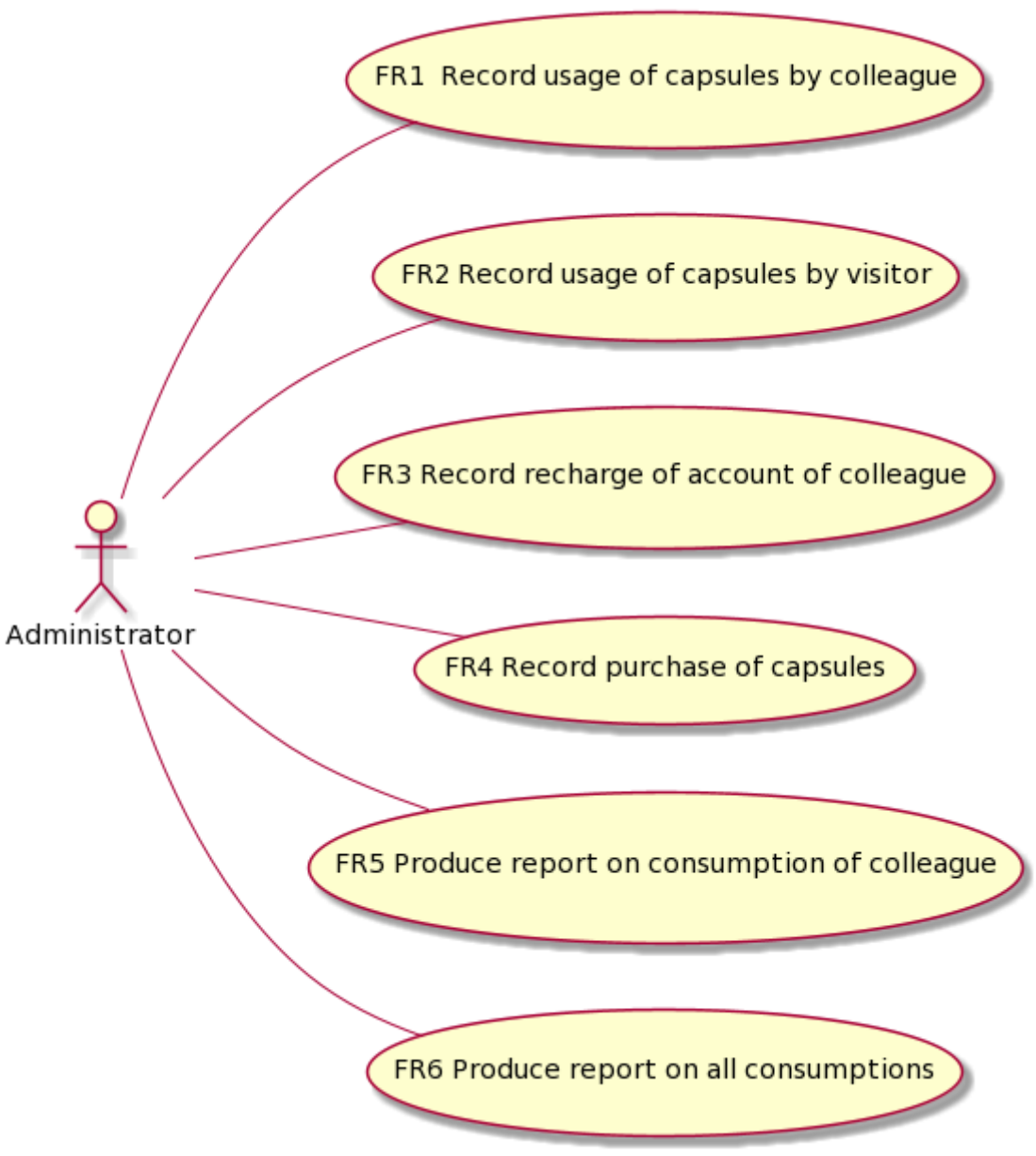
ID	Description
FR3	Record that a colleague has recharged x euros on her account
FR4	Record that n capsules of a certain type have been received, and paid for
FR5	Produce a report about consumption and recharges of a colleague over a certain period of time
FR6	Produce a report about all consumption and recharges over a certain period of time
FR7	Manage types of capsules and prices
FR8	Manage colleagues and accounts

Non Functional Requirements

ID	Type (efficiency, reliability, .. see iso 9126)	Description	Refers to
NFR1	Usability	Application should be used with no training by any colleague in the office	All FR
NFR2	Performance	All functions should complete in < 0.5 sec	All FR
NFR3	Portability	The application runs on MS Windows (7 and more recent)	All FR
NFR4	Portability	The application (functions and data) should be portable from a PC to another PC in less than 5 minutes	All FR
NFR5	Localisation	Decimal numbers use . (dot) as decimal separator	

Use case diagram and use cases

Use case diagram



Use Cases

Use case 1, UC1 - FR1 Record usage of capsules by colleague

Actors Involved	**Administrator**
Precondition	Capsule T exists, colleague C exists
Post condition	T.quantity_post < T.quantity_pre
	C.PersonalAccount.balance_post < C.PersonalAccount.balance_pre
Nominal Scenario	Administrator selects capsule type T, selects colleague C, Deduce quantity for capsule T, deduce price of T by account of colleague C
Variants	Account of colleague C has not enough money, issue warning

Use case 2, UC2 - FR2 Record usage of capsules by visitor

Actors Involved	**Administrator**
Precondition	Capsule T exists, visitor has no account
Post condition	T.quantity_post < T.quantity_pre
	LaTazzaAccount.amount_post > LaTazzaAccount.amount_pre
Nominal Scenario	Administrator selects capsule type T, Deduce quantity for capsule T, add price of T on LaTazzaAccount.amount
Variants	

Use case 3, UC3 - FR3 Record recharge of account of colleague

Actors Involved	**Administrator**
Precondition	Personal Account PA exists , quantity >0
Post condition	PA.balance_post = PA.balance_pre + quantity
	LaTazzaAccount.balance_post = LaTazzaAccount.balance_pre + quantity
Nominal Scenario	Administrator selects account PA of colleague C, increase account of quantity, increase LaTazza account of quantity
Variants	

Use case 4, UC4 - FR4 Record purchase of capsules

Actors Involved	**Administrator**
Precondition	Capsule type CT exists, LaTazzaAccount.balance has enough money for the purchase
Post condition	CT.quantity_post > CT.quantity _pre
	LaTazzaAccount.balance_post < LaTazzaAccount.balance_pre
Nominal Scenario	At time of order Administrator records money spent for order. At time of reception administrator selects capsule type CT, increases its quantity by a given number
Variants	

Use case 5, FR5 Produce report on consumption of colleague

Actors Involved	**Administrator**
Precondition	Colleague C exists
Post condition	

Actors Involved	**Administrator**
Nominal Scenario	Administrator selects colleague C, defines a time range, application collects all transactions for C (recharges and capsules taken) in the time range and presents them
Variants	

Use case 6, FR6 Produce report on all consumptions

Actors Involved	**Administrator**
Precondition	
Post condition	
Nominal Scenario	Administrator defines a time range, application collects all transactions (recharges, purchases, and capsules taken) in the time range and presents them
Variants	

Relevant scenarios

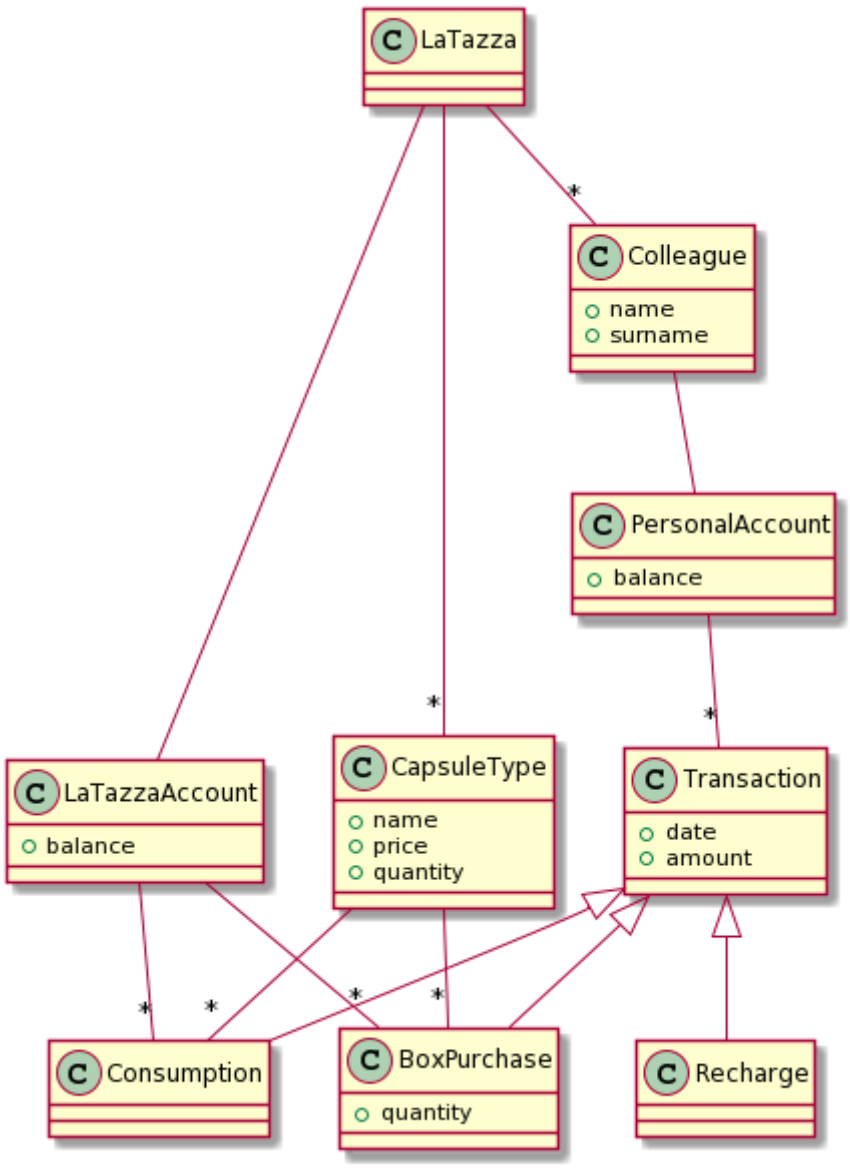
Scenario 1

Scenario ID: SC1	**Corresponds to UC1**
Description	Colleague uses one capsule of type T
Precondition	account of C has enough money to buy capsule T
Postcondition	account of C updated, count of T updated
Step#	Step description
1	Administrator selects capsule type T
2	Administrator selects colleague C
3	Deduce one for quantity of capsule T
4	Deduce price of T from account of C

Scenario 2

Scenario ID: SC2	**Corresponds to UC1**
Description	Colleague uses one capsule of type T, account negative
Precondition	account of C has not enough money to buy capsule T
Postcondition	account of C updated, count of T updated
Step#	Step description
1	Administrator selects capsule type T
2	Administrator selects colleague C
3	Deduce one for quantity of capsule T
4	Deduce price of T from account of C
5	Account of C is negative, issue warning

Glossary



System Design

Not really meaningful in this case. Only one personal computer is needed.

Deployment Diagram

As a stand-alone application only one node is needed.