

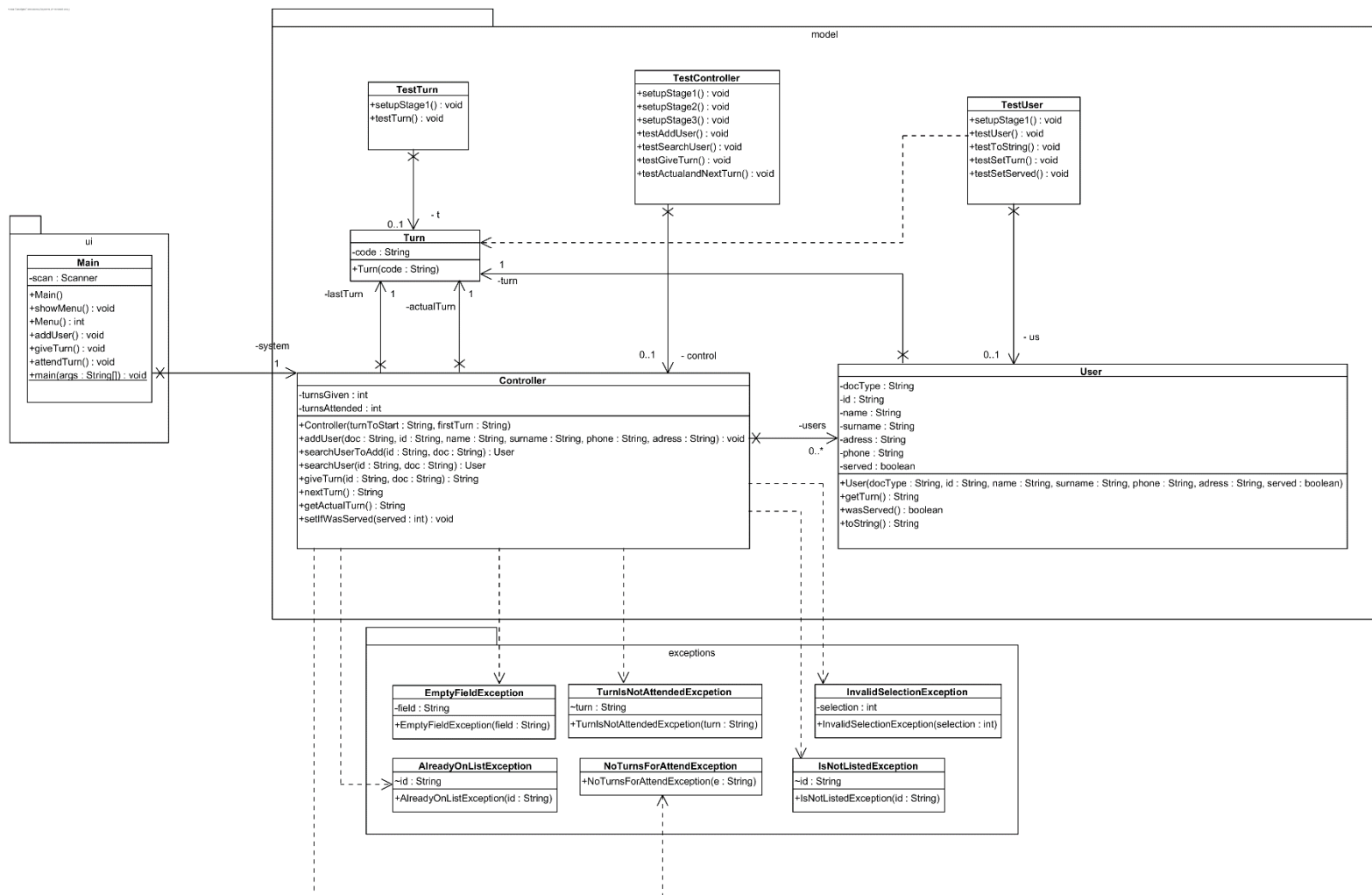
Functional requirements.

1. Search user to assign a turn. The program searches for a user with his id number; if the user was found and doesn't have an active turn, the program will assign a turn to him. If the user wasn't found, the program will ask for it to be registered. If it was found but has an active turn, the program will not assign a new turn.

2. Assign turn to a user since A00 until Z99, when the turn is Z99 the next turn is A00. The turn will be assigned only if the user has no active turns.

3. Register new user. The program asks for the user's info, if one of the fields "NAME", "DOCUMENT TYPE", "SURNAME" or "DOCUMENT NUMBER" is empty, the register will be not completed. If the user is already registered, the register will be not completed.

4. Advance turn to attend a user. Only can advance the turn if has turns to attend. The person in charge of attend has two options to choose when advance a turn, if the user was attended, the person would choose "User attended" or, if the user is not in the establishment when his turn was called, the person would choose "User absent".



Stages configuration.

Name	Class	Stage
setupStage1	Controller	An object of the class Controller with actualTurn="A00", lastTurn="A0/" and an object of the class User with name="Alejandro", surname="Garcia", docType="CC", id="1193151954", phone="3114209888", address="Cra 61 #18-16" and give turn to this user.
setupStage2	Controller	An object of the class Controller with actualTurn="A00" and lastTurn="D99".
setupStage3	Controller	An object of the class Controller with actualTurn="A00" and lastTurn="Z99". Add a user with name="alejandro", surname="garcia", docType="CC", id="1193151954", phone="3114209888" and address="Cra 61 #18-16".
setupStage1	User	An object of the class User with name="Alejandro", surname="Garcia", docType="CC", id="1193151954", phone="3114209888", address="cra 61 #18-16" and served=false.
setupStage1	Turn	An object of the class Turn with code="A00".

Design of test cases.

Objective of the test: Verify the correct running of the method addUser of the class Controller, adding correctly a user with the info passed by parameters.				
Clase	Método	Escenario	Valores de Entrada	Resultado
Controller	addUser	setupStage1	docType="", id="1234", name="Maria", surname="Garcia", phone="3146131522", address="cra 61 #18-16"	Throws EmptyFieldException
Controller	addUser	setupStage1	docType="CC", id="", name="Maria", surname="Garcia", phone="3146131522", address="cra 61 #18-16"	Throws EmptyFieldException

Controller	addUser	setupStage1	docType="CC", id="1234", name="", surname="Garcia", phone="3146131522", address="cra 61 #18-16"	Throws EmptyFieldException
Controller	addUser	setupStage1	docType="CC", id="1234", name="Maria", surname="", phone="3146131522", address="cra 61 #18-16"	Throws EmptyFieldException
Controller	addUser	setupStage1	docType="CC", id="119315994", name="Angel", surname="Garcia", phone="3105358790", address="Cra 61 #18-16"	The user will be added. Can be checked with the method searchUser(id, doc)
Controller	addUser	setupStage1	docType="CC", id="119315954", name="Alejandro", surname="Garcia", phone="3114209888", address="Cra 61 #18-16"	Throws AlreadyOnListException
Controller	addUser	setupStage2	docType="CC", id="119315954", name="Alejandro", surname="Garcia", phone="3114209888", address="Cra 61 #18-16"	The user will be added. Can be checked with the method searchUser(id, doc)

Objective of the test: Verify the correct running of the method searchUser of the class Controller, finding correctly a user with the info passed by parameters.

Clase	Método	Escenario	Valores de Entrada	Resultado
Controller	searchUser	setupStage2	docType="CC", id="1193151954"	Throws IsNotListedException
Controller	searchUser	setupStage1	docType="CC", id="1193151954"	Found and return the user with the info passed by

				parameter. Can be checked comparing the toString methods
--	--	--	--	--

Objective of the test: Verify the correct running of the method giveTurn of the class Controller, assigning correctly a turn to user with the info passed by parameters.				
Clase	Método	Escenario	Valores de Entrada	Resultado
Controller	giveTurn	setupStage1	docType="CC", id="1193151954"	Throws TurnIsNotAttendedException
Controller	giveTurn	setupStage1	setTurn(null) docType="CC", id="1193151954"	Now the turn of the user has been updated to "A01"
Controller	giveTurn	setupStage2	addUser(doctype="CC", id="1193151954", name, surname, phone, adress) id docType	The turn gave to the user is "E00"
Controller	giveTurn	setupStage3	docType="CC", id="1193151954"	The turn gave to the user is "A00"

Objective of the test: Verify the correct running of the methods getActualTurn and nextTurn of the class Controller, advancing correctly turns and to return.				
Clase	Método	Escenario	Valores de Entrada	Resultado
Controller	actualTurn	setupStage1		The method returns "Actual turn is: A00"
Controller	nextTurn	setupStage1		The method returns "The next turn is: A01"
Controller	nextTurn	setupStage1		Throws NoTurnsForAttendException

Objective of the test: Verify the correct running of the Constructor of the class User.

Clase	Método	Escenario	Valores de Entrada	Resultado
User	User	setupStage1		With getters, check that the info was saved on the class

Objective of the test: Verify the correct running of the method toString of the class User.

Clase	Método	Escenario	Valores de Entrada	Resultado
User	toString	setupStage1		The method returns the info of the class by the toString

Objective of the test: Verify the correct running of the method setTurn of the class User.

Clase	Método	Escenario	Valores de Entrada	Resultado
User	setTurn	setupStage1	new Turn("A00")	Set new turn to the user

Objective of the test: Verify the correct running of the method setServed of the class User.

Clase	Método	Escenario	Valores de Entrada	Resultado
User	setServed	setupStage1	true	Update the attribute "Served" of the user

Objective of the test: Verify the correct running of the Constructor of the class Turn.

Clase	Método	Escenario	Valores de Entrada	Resultado
Turn	Turn	setupStage1		With getters, check that the info was saved on the class