

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

# Software Requirements Specification

Version 1.1

for

UberSanté

Prepared by

Name	ID
Kevin Luu ( <i>Leader</i> )	40037514
Tamar Merdkhanian	40030718
Michael Tang	40028150
Tristan Vu	40028927
Wu Wen Tang	40028075
Mathew Jackson	27709315
Zhen Yee	40028478

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## Document history

Date	Version	Description	Author
10/02/2019	1.0	Add template	
13/02/2019	1.1	Add related section	

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

# Table of Contents

<b>1. Introduction</b>	<b>5</b>
1.1 Purpose	5
1.2 Scope	5
1.3 Definitions, acronyms, and abbreviations	5
1.4 References	6
1.5 Overview	6
<b>2. Overall description</b>	<b>6</b>
2.1 Product perspective	6
2.2 Product functions	7
2.3 User characteristics	7
2.4 Constraints	7
2.5 Assumptions and dependencies	8
<b>3. Specific requirements</b>	<b>8</b>
3.1 Functional requirements	8
3.1.1 Functional requirements	8
3.2 Use Cases of Critical Features	10
3.3 Non-functional requirements	13
3.3.1 Business Rules	13
3.3.2 Concurrency Properties	14
<b>4. Analysis Models</b>	<b>15</b>
4.1 Use Case Diagram	15
4.2 Domain Models	16
4.4 Sequence Diagrams	16
<b>5. Operation Contracts</b>	<b>19</b>

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## List of figures

Figure 1. A client-server architecture. 6

Figure 2. An example user interface.. 8

Figure 3. Use case model. 10

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

# 1. Introduction

## 1.1 Purpose

The content contained within this document is aimed at capturing and presenting all the requirement specifications for the UberSanté non urgent care scheduling system. The document is intended to be read by the developers of the project and other stakeholders.

## 1.2 Scope

The UberSanté non urgent care scheduling system is a web application that allows users to make, view, update and cancel non urgent medical appointments. The system will be available to patients, doctors and nurses albeit in different capacities. Patients will have the ability to make, view, modify, update and pay for their own appointments. Nurses will have the ability to make, view, modify and update appointments for any patient. Finally Doctors will be able to use the system to provide their availabilities, as well as view their upcoming appointments.

## 1.3 Definitions, acronyms, and abbreviations

Term	Definition
SRS	Software Requirements Specification (this document)
User	Any person using the system, be it Doctor, Patient or Nurse
FR	Functional Requirement
NFR	Non-Functional Requirement
UC	Use Case
Reader	A user who wishes to perform a read operation is referred to as a 'reader.' A read operation accesses in order to view (but does not modify) the

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

	contents of a shared resource.
Writer	A user who wishes to perform a write operation is referred to as a 'writer.' A write operation accesses in order to modify the contents of a shared resource.
SSD	System Sequence Diagram
EFSM	Extended Finite State Machine
UI	User Interface

## 1.4 References

- [1] IEEE Software Engineering Standards Committee, "*IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications*", October 20, 1998
- [2] International Electrotechnical Commission, "ISO/IEC 25010:2011(E), Systems and Software Quality Requirements and Evaluation (SQuaRE)", First edition, 2011-03-01.

## 1.5 Overview

The remainder of this document is broken into four sections. Section 2 provides an overall description of the system by describing the product perspective, the product functions, and user characteristics; furthermore, this section supplies any and all constraints, assumptions and dependencies that the system must adhere to. Section 3 details the specific requirements of the system by presenting the functional requirements, and the non-functional requirements of the system. Section 4 provides analysis models that interpret the system requirements such as the use case diagram, Sequence Diagrams, State Diagrams, and the overall Domain Model. Finally, Section 5 lists the Operation Contracts that correspond to the system operations seen in the SSD's.

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## 2. Overall description

### 2.1 Product perspective

UberSanté is a web application catering to Doctors, Nurses, and Patients. It's main functionalities is to allow patients to book appointments with doctors and to help patients manage their appointments. To be usable, a clinic must be in UberSanté's system. The database for UberSanté will be stored in the clouds using Amazon Web Services (AWS). Information on clinics, doctors, nurses, and patients will be stored in AWS. The application will communicate with the database via the internet.

### 2.2 Product functions

UberSanté will allow doctors to manage their availabilities as well as to view their appointments. The system also allows patients to book appointments with doctors as well as to manage their appointments. The nurses interacting with the system can help patients book appointments or manage a doctor's appointment. Every availability that is chosen by the patient must be checked out via the cart and paid using a credit card. The database will store all information that is manipulate by the user via UberSanté.

### 2.3 User characteristics

There are three types of users: Doctors, Nurses, and Patients. Doctors should have a basic knowledge of how to interact with the system. The only knowledge a doctor needs is to input their availabilities in the system as well as to view their availabilities. For nurses, they should have a better understanding of the system because they are able to manipulate appointments for doctors and make appointments for clients. As for clients, basic knowledge of how to interact with the system will be needed to book appointments.

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## 2.4 Constraints

The web application is constrained by the server connection, the database capacity and the Internet connection. The server connection is a constraint to the application because it is connected to the database and the Internet connection of the user's device. Without it, the application cannot function entirely.

Additionally, the database capacity is constraint to the application. Since it holds the information users add into the database, queries and requests sent and accepted by the server may increase in time in order to fetch the corresponding data.

Another constraint to the application is the Internet connection. Since the application fetches data from the database over the Internet and the server, is it necessary to have an Internet connection.

## 2.5 Assumptions and dependencies

It is assumed that the user is capable of reading and understanding the English language, as the UI is written solely in English. Another assumption is that the user is running the web application on a compatible web browser. Many different web browsers are available but it is recommended to use latest versions of Google Chrome, Safari, Internet Explorer or Mozilla Firefox.

It is assumed that the user is running the web application on a compatible operating system. It is only required to be running on Mac OS X, Linux or a later version of Windows XP. It is also assumed that the user has access to an Internet connection strong enough to fetch the information from the database.



<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## 3. Specific requirements

### 3.1 Functional requirements

All functional requirements for the application <Uber Sante> are listed in section 3.1.1 and are organized by feature. Critical requirements are represented by fully dressed use cases in 3.1.2.

#### 3.1.1 Functional requirements

##### 3.1.1.1 User Management

###### **ID: FR1**

TITLE: Patient Registration

DESC: The system shall accept patients to register themselves with their health card number, birthday, gender, phone number and physical and email address.

###### **ID: FR2**

TITLE: User login

DESC: Given that a user has been registered (either a doctor, nurse or patient), they may log in to the system.

##### 3.1.1.2 Clinic System

###### **ID: FR3**

TITLE: Walk-in hours

DESC: Clinic hours are from 8AM until 8PM

###### **ID: FR4**

TITLE: Walk-in appointment

DESC: Clinic hours walk-in appointment shall be 20 minutes.

###### **ID: FR5**

TITLE: Annual checkup appointment

DESC: Clinic hours annual-checkup appointment shall be 60 minutes.

###### **ID: FR6**

TITLE: Medical rooms

DESC: Clinic has a limited number of 5 medical rooms

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

**ID: FR7**

TITLE: Number of doctors

DESC: Clinic has a limited number of 7 doctors

### 3.1.1.3 Booking, Cancelling and Viewing Appointments

**ID: FR8**

TITLE: Booking Appointment

DESC: While viewing the availabilities of the clinic, the system shall accept a patient or a nurse to make an appointment

**ID: FR9**

TITLE: Modifying Appointment

DESC: Nurse and patient can modify existing appointment in the clinic system

**ID: FR10**

TITLE: Viewing Appointment

DESC: Patient, nurse can patient are able to check an appointment

**ID: FR11**

TITLE: Cancelling Appointment

DESC: Nurses and patient can cancel an existing appointment in the clinic system

**ID: FR12**

TITLE: Modifying Appointment

DESC: Nurses and patient can modify existing appointment in the clinic system

**ID: FR13**

TITLE: Cart System

DESC: Patients shall confirm the booking by going to checkout directly via appointment cart.

**ID: FR14**

TITLE: Persisted Cart System

DESC: The system shall save the appointment cart for lately use.

**ID: FR15**

TITLE: Persisted Cart System

DESC: The system shall save the appointment cart for lately use.

**ID: FR16**

TITLE: Update doctor schedule availability

DESC: The system shall accept registered doctors to update their schedule availability.

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

**ID: FR17**

TITLE: Schedule Availability for doctors

DESC: The system shall accept registered doctors to complete their schedule availability.

**ID: FR18**

TITLE: Schedule Walk-in appointment for doctors

DESC: The system shall accept registered doctors to enter 20 minute walk-in appointments.

**ID: FR19**

TITLE: Schedule Annual appoint for doctors

DESC: The system shall accept registered doctors to enter 60 minute annual checkup appointments.

**ID: FR20**

TITLE: Schedule Availability for doctors

DESC: The system shall check the availability of doctors and rooms when booking or canceling appointments.

## 3.2 Use Cases of Critical Features

Use Case UC1	Book an Appointment
Related to	FR8
Primary Actor	Patient, Nurse
Stakeholder and Interests	Patient and nurse can book an annual or a walk-in appointment
Preconditions	The patient does not have more than 1 annual appointment to book an annual appointment.
Postconditions	The appointment is booked through a doctor's availability
Basic flow (main success)	<ol style="list-style-type: none"> <li>1) Patient clicks on "Book an appointment"</li> <li>2) Patient chooses "Walk-in" or "Annual"</li> <li>3) Patient clicks on an availability</li> <li>4) Patient view summarized appointment and confirms</li> </ol>

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

	booking 5) Patient clicks on "Cart" 6) Patient presses checkout
Alternative flow	1) Nurse chooses a health card number for a patient 2) Nurse clicks on "Book an appointment" 3) Nurse chooses "Walk-in" or "Annual" 4) Nurse clicks on an "Book" 5) Nurse view summarized appointment and confirms booking
Priority	High

Use Case UC2	Patient Update Appointment
Related to	FR12
Primary Actor	Nurse, Patient
Stakeholder and Interests	Nurse and Patient can update their appointment to a new time
Preconditions	Patient or Nurse needs to be logged in and have an appointment currently booked
Postconditions	The appointment has a new booking time. The past appointment is now canceled.
Basic flow (main success)	1) Patient clicks on the appointment he wants to update 2) Patient navigates through the calendar 3) Patient clicks a date for their new appointment 4) Patient confirms their new appointment  7) Nurse picks a patient 8) Nurse clicks on the appointment she wants to update 9) Nurse navigates through the calendar 10) Nurse clicks a date for their new appointment 11) Nurse confirms their new appointment
Alternative flow	1) Patient clicks on the appointment he wants to update

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

	2) Patient navigates through the calendar 3) Patient does not find any other date for his appointment 4) Patient keeps his current appointment 5) Patient cancels his current appointment
Priority	High

Use Case UC3	Patient Cancel Appointment
Related to	FR11
Primary Actor	Nurse, Patient
Stakeholder and Interests	Nurse and Patient can cancel current appointments.
Preconditions	Nurse or Patient is logged in.
Postconditions	The appointment is canceled. The availability at that time is now available.
Basic flow (main success)	1) Patient clicks on the appointment he wants to cancel. 2) Patient confirms the cancel.  1) Nurse picks a patient 2) Nurse clicks on the appointment she wants to cancel 3) Nurse confirms the cancel
Alternative flow	
Priority	High

Use Case UC4	Doctor Updates Availability
Related to	FR13, FR14, FR15, FR16
Primary Actor	Doctor
Stakeholder and Interests	Doctor: is able to update his availabilities in the system.

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

Preconditions	Doctor needs to be logged into the system.
Postconditions	Doctors availabilities are updated.
Basic flow (main success)	<ol style="list-style-type: none"> <li>1) Doctor navigates to “My Availabilities”.</li> <li>2) Presses the “+” (add) button and selects walk-in or annual checkup.</li> <li>3) System prompts Doctor to input starting day and time.</li> <li>4) Doctor presses “Apply”</li> <li>5) System registers new availability and displays result in Doctor’s calendar.</li> <li>6) Doctor presses on a specific availability on the calendar if he/she wishes to update it or delete it.</li> <li>7) System prompts Doctor to change starting day and time or to remove the event.</li> <li>8) System updates availability based on selected option.</li> </ol>
Alternative flow	<p>4.a) System rejects availability since it overlaps with another one.</p> <p>4. b) system rejects availability since it’s outside clinic hours</p>
Priority	High

### 3.3 Non-functional requirements

#### 3.3.1 Business Rules

##### **ID: NFR1**

Title: Patient minimum age

DESC: Registered patients should be over 18 years of age.

##### **ID: NFR2**

Title: Payment method

DESC: Credit card is the only acceptable payment method.

##### **ID: NFR3**

Title: Annual checkup limit

DESC: The system accepts only one annual checkup appointment per patient.

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

**ID: NFR4**

Title: Walk-in limit

DESC: The system accepts an unlimited number of walk-in appointments per patient.

### 3.3.2 Concurrency Properties

**ID: NFR5**

TITLE: Shared Resource Safety 1

DESC: A shared resource must not be simultaneously accessed by a writer and reader.

**ID: NFR6**

TITLE: Shared Resource Safety 2

DESC: A shared resource must not be simultaneously accessed by more than one writer.

**ID: NFR7**

TITLE: Shared Resource Liveliness

DESC: Users who wish to obtain access to a shared resource will eventually be allowed access.

**ID: NFR8**

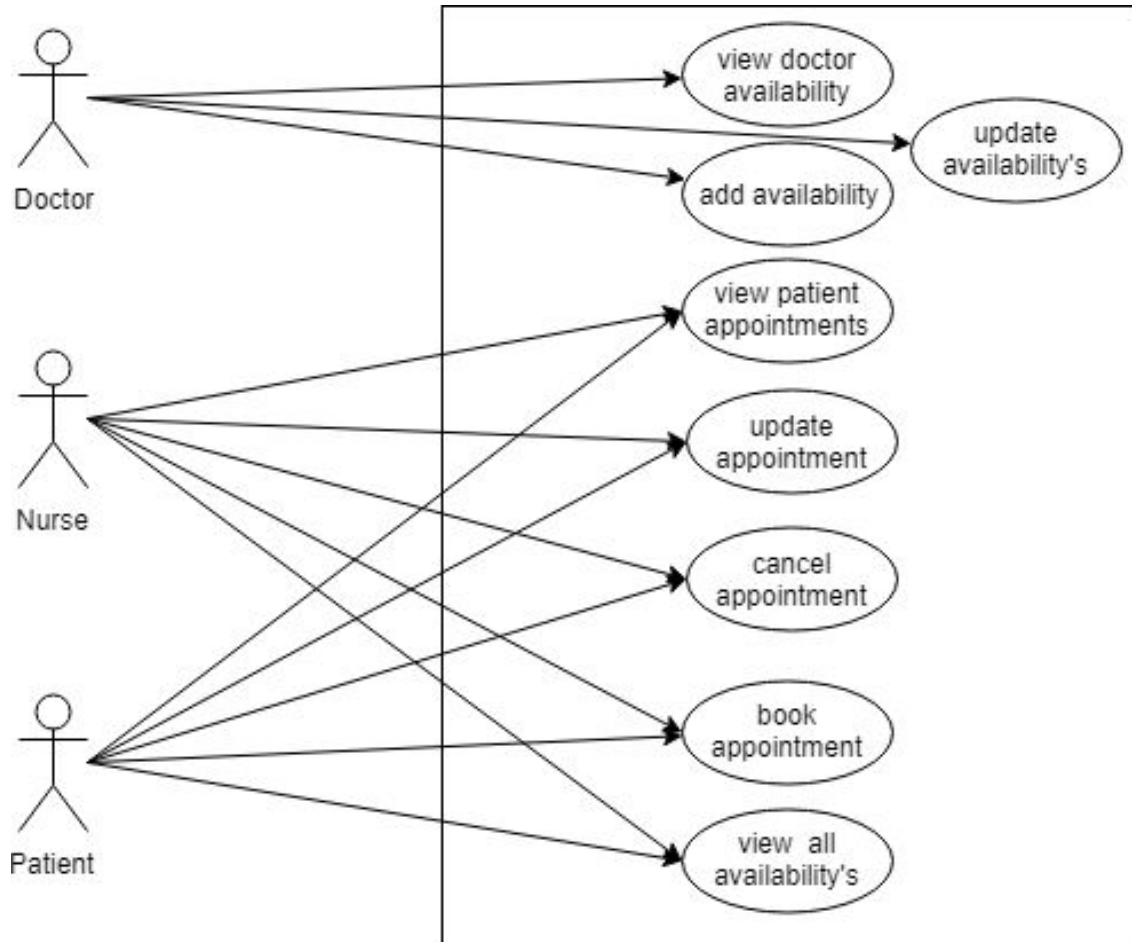
TITLE: Shared Resource Fairness

DESC: Requests for access must succeed infinitely often. No user will wait forever to be serviced. Furthermore, in this system we want writers to have priority over readers.

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## 4. Analysis Models

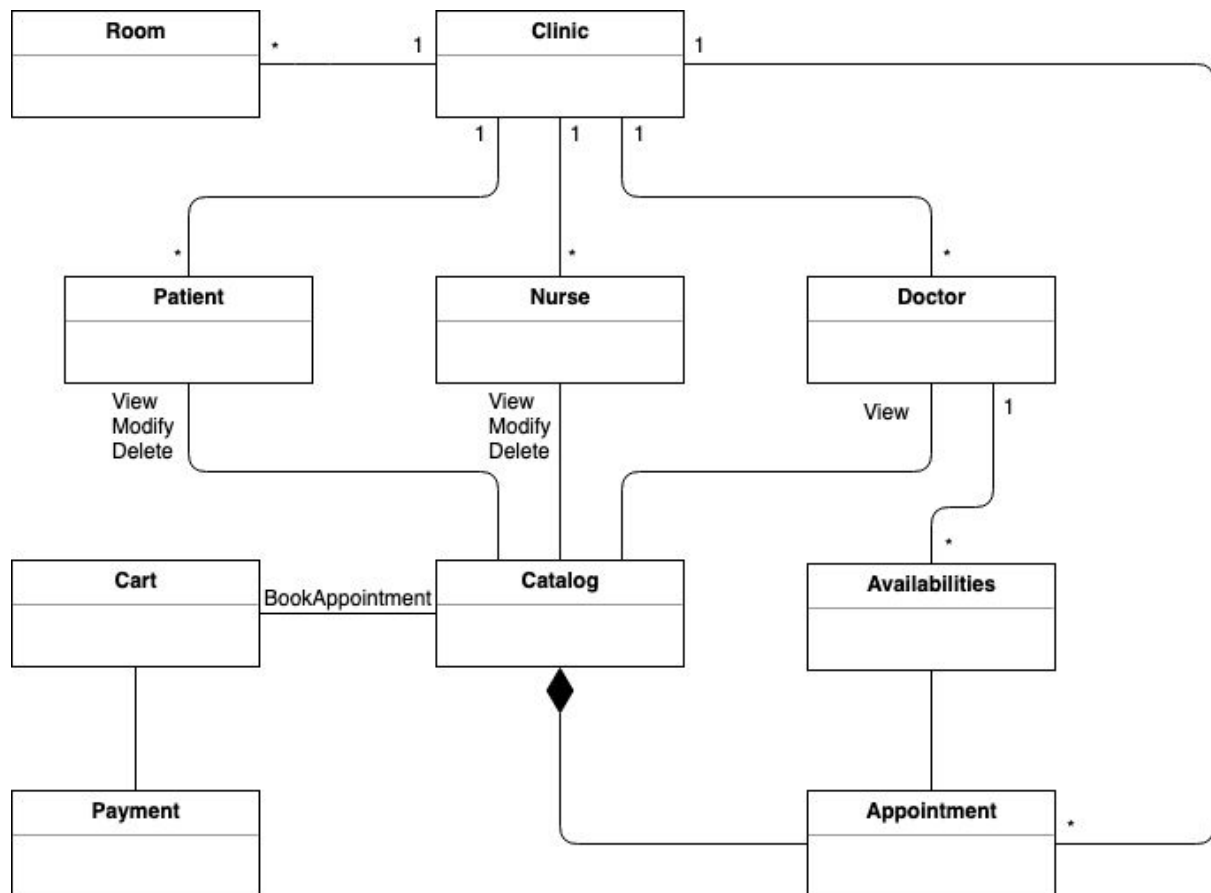
### 4.1 Use Case Diagram





<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## 4.2 Domain Models

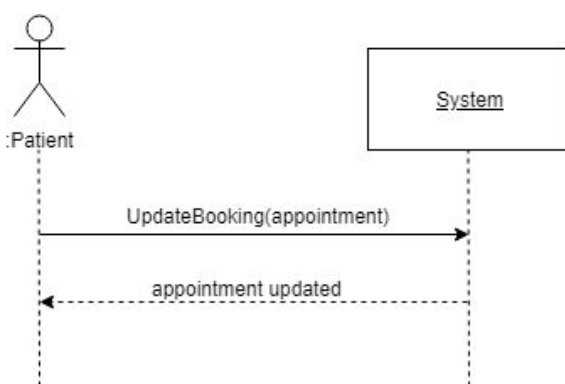


## 4.4 Sequence Diagrams

<b>SSD1</b>	
Cross References	UC1

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

Diagram	<pre>sequenceDiagram     actor Patient as :Patient     participant System     Patient-&gt;&gt;System: onSubmit(info)     System--&gt;&gt;Patient: confirmation</pre>		
System Operations	<table><tr><td>System</td></tr><tr><td>onSubmit(info)</td></tr></table>	System	onSubmit(info)
System			
onSubmit(info)			

SSD2	Patient Update Appointment		
Cross References	UC2		
Diagram	 <pre>sequenceDiagram     actor Patient as :Patient     participant System     Patient-&gt;&gt;System: UpdateBooking(appointment)     System--&gt;&gt;Patient: appointment updated</pre>		
System Operations	<table border="1" data-bbox="820 1666 1163 1845"><tr><td>System</td></tr><tr><td>updateBooking(appointment)</td></tr></table>	System	updateBooking(appointment)
System			
updateBooking(appointment)			

<b>SSD3</b>	<b>Patient Cancel Appointment</b>
-------------	-----------------------------------

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

Cross References	UC3
Diagram	<pre> sequenceDiagram     actor Patient as :Patient     participant System     Patient-&gt;&gt;System: cancelBooking(appointment)     System--&gt;&gt;Patient: appointment cancelled </pre>
System Operations	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px; text-align: center;">System</div> <div style="padding-top: 5px; text-align: center;">cancelBooking(appointment)</div> </div>

<b>SSD4</b>	<b>Doctor Update Availabilities</b>
Cross References	UC6
Diagram	<pre> sequenceDiagram     actor Doctor as : Doctor     participant System     Doctor-&gt;&gt;System: addAvailability(Availability)     System--&gt;&gt;Doctor: new availability is saved     System--&gt;&gt;Doctor: new availability rejected; error message sent </pre>
System Operations	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px; text-align: center;"><b>System</b></div> <div style="padding-top: 5px; text-align: center;">addAvailability(Availability)</div> </div>

<UberSanté>	Version: 1.1.0
Software Requirements Specification	Date: 2018-03-11

## 5. Operation Contracts

### Contract C1: bookAppointment

**Operation:** bookAppointment(user, cart, payment)

**Cross References:** UC1, SSD1

**Preconditions:** Patient or Nurse needs to be logged in and the patient must not have more than 1 annual appointment to book an annual appointment.

**Postconditions:** The appointment has been booked. Availability is now occupied.

### Contract C4: updateAppointment

**Operation:** updateAppointment(user, time)

**Cross References:** UC4, SSD4

**Preconditions:** Patient or Nurse needs to be logged in and have an appointment currently booked

**Postconditions:** The appointment has a new booking time. The past appointment is now canceled.

### Contract C5: cancelAppointment

**Operation:** cancelAppointment(user, time)

**Cross References:** UC5, SSD5

**Preconditions:** Nurse or Patient is logged in and have an appointment currently booked

**Postconditions:** The appointment is canceled. The availability at that time is now available for other users.

### Contract C6: addAvailability

**Operation:** addAvailability(Availability)

**Cross References:** UC6, SSD6

**Preconditions:** Availability is within clinic operating hours and availability doesn't conflict with pre-existing availabilities

**Postconditions:** New availability is persisted