Session 3:
Requirements for Apps

Karim Lekadir, PhD

Universitat de Barcelona
Dpt. Matemàtiques & Informàtica

Calendar

Week		Theory	Labs	Deliverables
1	16 February	Introduction to PIS	Introduction to Android	
2	23 February	Design pattern 1	Session 1	
3	2 March	Requirements	Session 2	
4	9 March	Examples requirements	Session 3	
5	16 March	Feedback deliverable 1	Feedback & support	
6	23 March	Design pattern 2	Feedback & support	1. Requirements + UI
7	30 March	Software testing	Feedback & support	
8	6 April	Week partials (not for PIS)	Feedback & support	
9	13 April	*Semana Santa*	Feedback & support	
10	20 April	Feedback deliverable 2	Feedback & support	
11	27 April	*Matefest-Infofest (no lectiu)*	Feedback & support	2. Design + demo
12	4 May	*Fira d'Empreses*	Feedback & support	
13	11 May	Feedback deliverable 3	Feedback & support	
14	18 May	Trial exam	Feedback & support	
15	25 May	Trial exam	Presentations	3. Final project

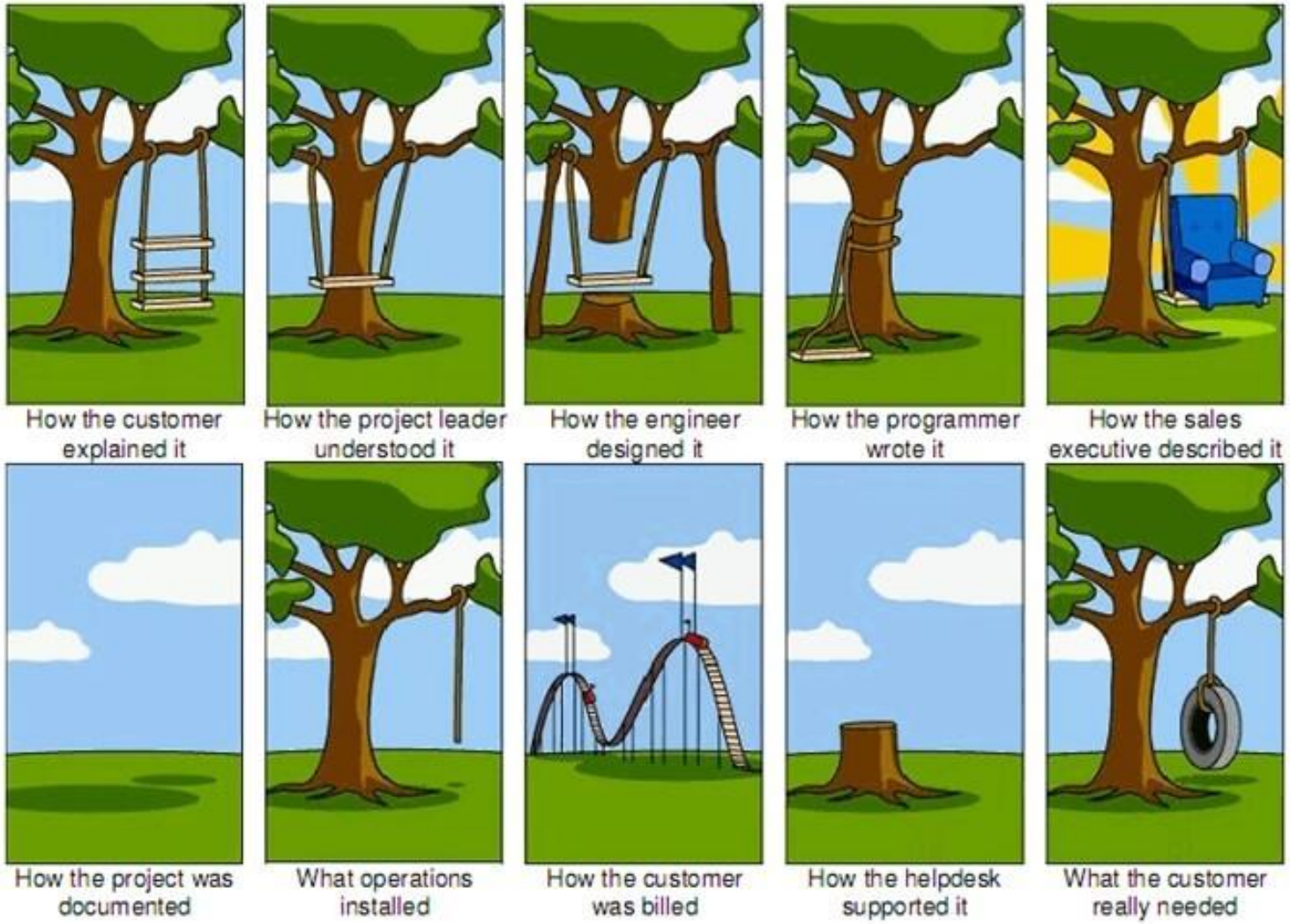
Requirements Analysis

- Description of the software/app that is, as much as possible:
  - ✓ Detailed
  - ✓ Precise
  - ✓ Comprehensive/complete
  - ✓ Clear and user-friendly
  - ✓ Multi-actor (user, developer, programmer, evaluator, etc)

Types

- User requirements
  - ✓ Functional
  - ✓ Non-functional
- System capabilities
- External conditions, e.g. ethical & legal requirements

Justification



Objectives

1. Better understand expectations
2. Remove assumptions, approximations and grey areas
3. Improve quality and usability of the software
4. Improve communication between users and developers
5. Improve communication within development team

Requirements

Example Software Requirements Specification Document for ReqView

Libor Buš, Eccam s.r.o.

12.6.2019

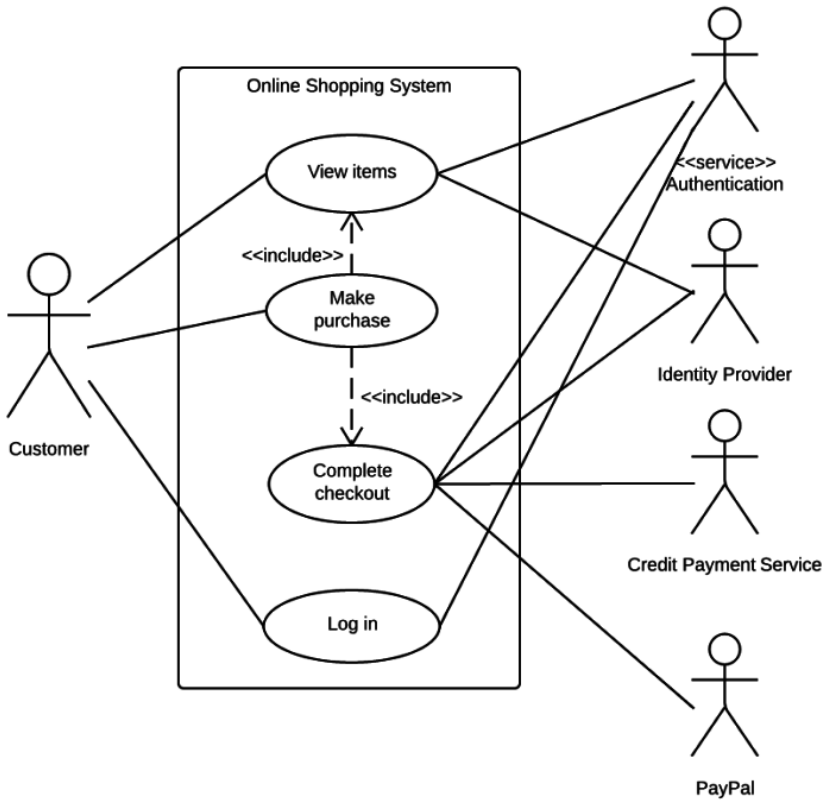
ID	Description	Traces	Component
LLR-14	If the PressurizationEvent and the DelayDOEVActuationTimeoutEvent are raised, the Door Opening EV Actuation Command shall be set to Open and the waitForDoorsOpen method shall be activated.	HLR-2 HLR-3 HLR-6 HLR-7	SequenceController
LLR-43	If the General EV Actuation Command is set to Open, the waitForHydraulicPressure method shall be activated.	HLR-6	SequenceController.
LLR-44	If the waitForHydraulicPressure method is active and the overall value of the Hydraulic Circuit Pressure monitorable variable is less than 30,000 kPa, the waitForHydraulicPressure method shall remain active until the PressurizationTimeoutEvent is raised.	HLR-6 HLR-12	SequenceController
LLR-45	If the waitForHydraulicPressure method is active and the overall value of the Hydraulic Circuit Pressure is greater than or equal to 30,000 kPa and less than 35,000 kPa, the waitForHydraulicPressure method shall end and the PressurizationEvent shall be raised.	HLR-6	SequenceController
LLR-46	If the General EV Actuation Command was set to Open, the DelayDOEVActuation method shall be activated.	HLR-7	SequenceController
LLR-56	If the waitForHydraulicPressure method is active and 2 seconds have elapsed since the General EV Actuation Command was set to Open, the PressurizationTimeoutEvent shall be raised.	HLR-12	SequenceController
LLR-57	If the waitForHydraulicPressure method is active, the PressurizationTimeoutEvent is raised and the overall value of the Hydraulic Circuit Pressure monitorable variables is less than 30,000 kPa, the FailureEvent shall be raised.	HLR-12	SequenceController
LLR-35	If the waitForHydraulicPressure method is active and the RevertEvent is raised, all the actions that were previously executed shall be reverted.	HLR-4	SequenceController

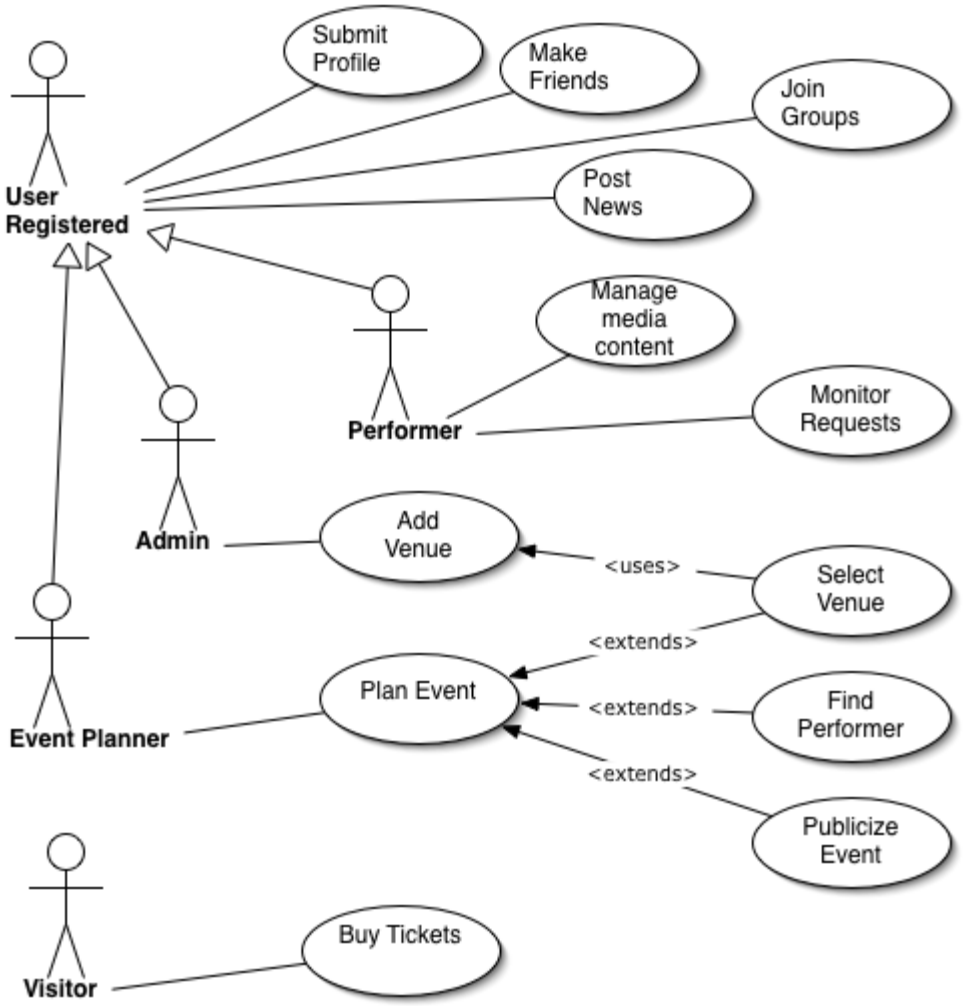
Methodology

Steps
1. Use case diagrams
2. Mock-ups of UIs
3. Co-creation activities
4. Requirements document

Approach
1. Collaborative
2. Structured (Block-by-block)
3. Detailed and precise
4. Iterative

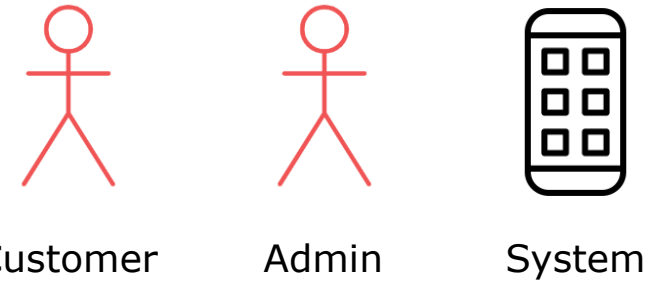
1. Use Cases


Online Shopping System
View items
<<include>>
Make purchase
<<include>>
Complete checkout
Log in
Customer
<<service>> Authentication
Identity Provider
Credit Payment Service
PayPal

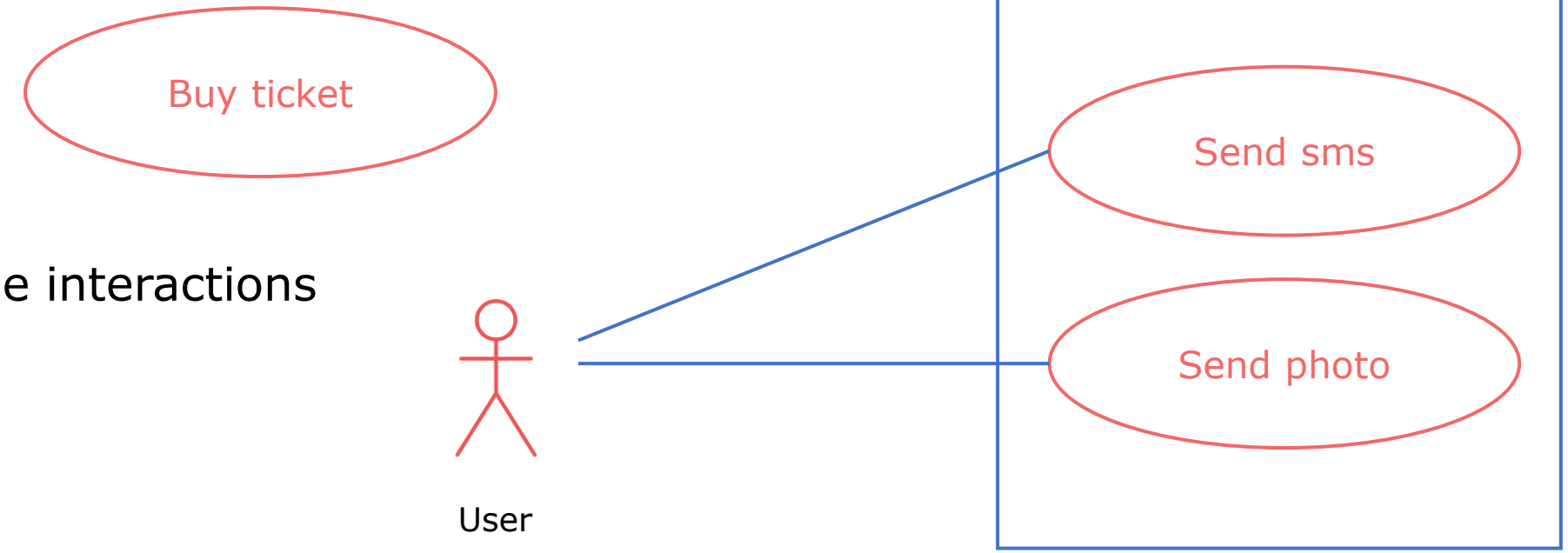

Submit Profile
Make Friends
Join Groups
Post News
User Registered
Manage media content
Performer
Monitor Requests
Admin
Add Venue
<uses>
Select Venue
<extends>
Plan Event
<extends>
Find Performer
<extends>
Publicize Event
Event Planner
Buy Tickets
Visitor

1. Use Cases

1. Define actors and users



2. Define functionalities

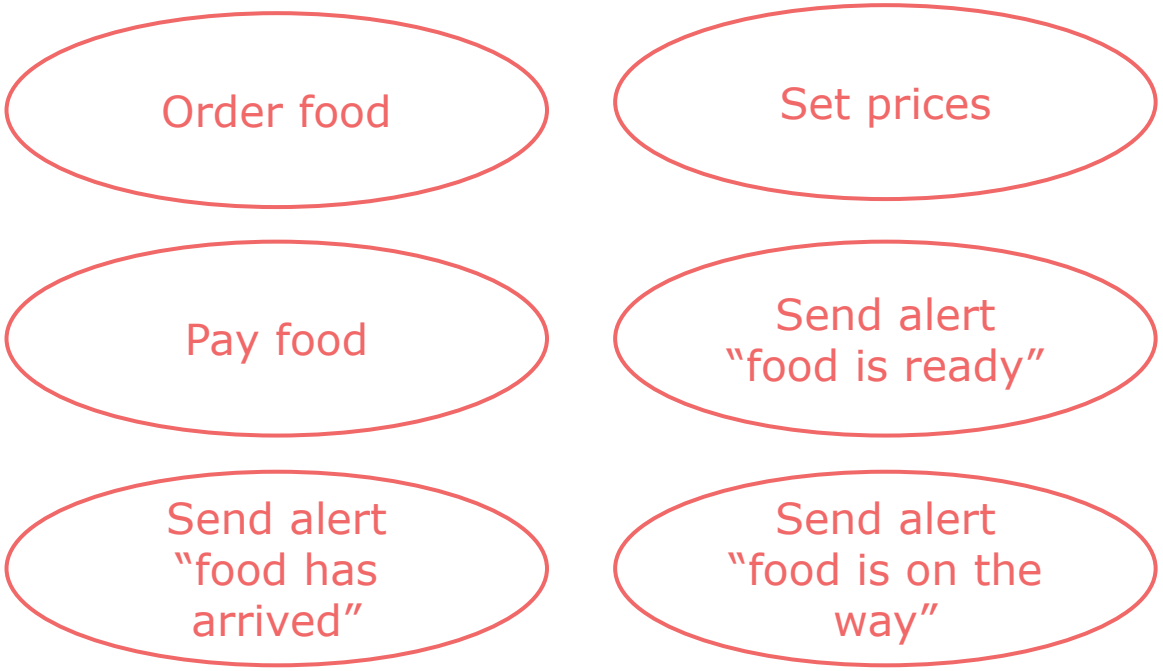


3. Define interactions

1. Use Cases

1. Define actors and users:

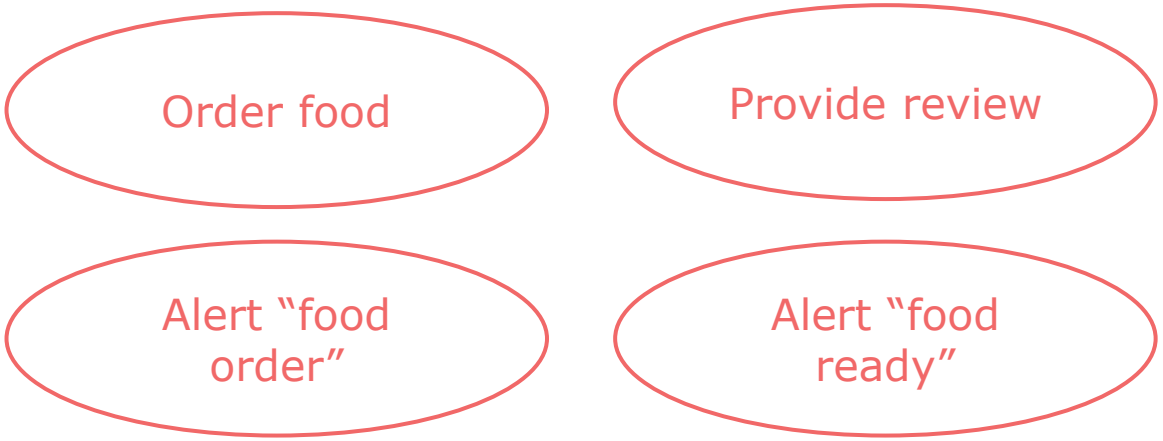
2. Define functionalities:



1. Use Cases

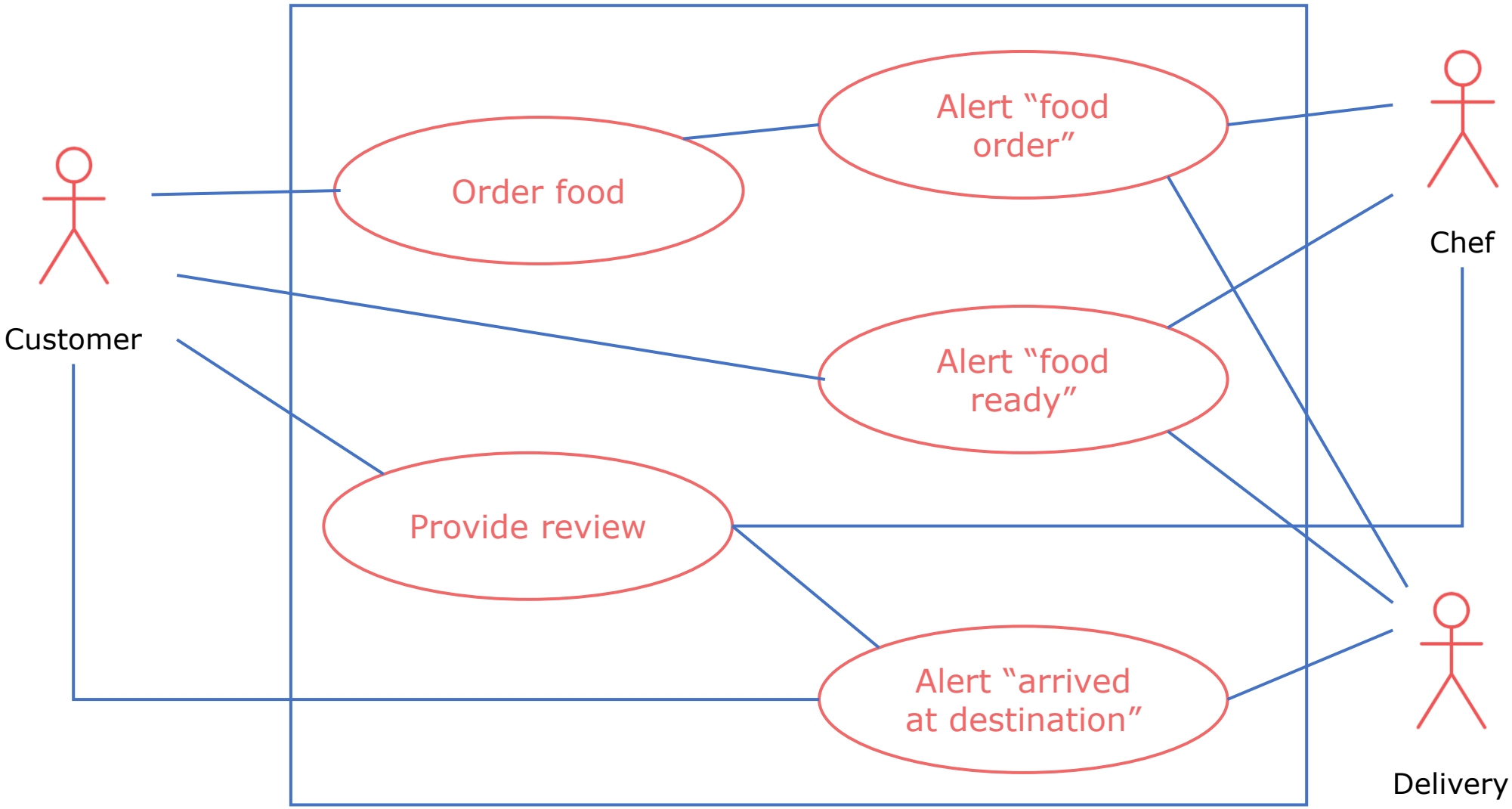
1. Define actors and users:

2. Define functionalities:



3. Define interactions:

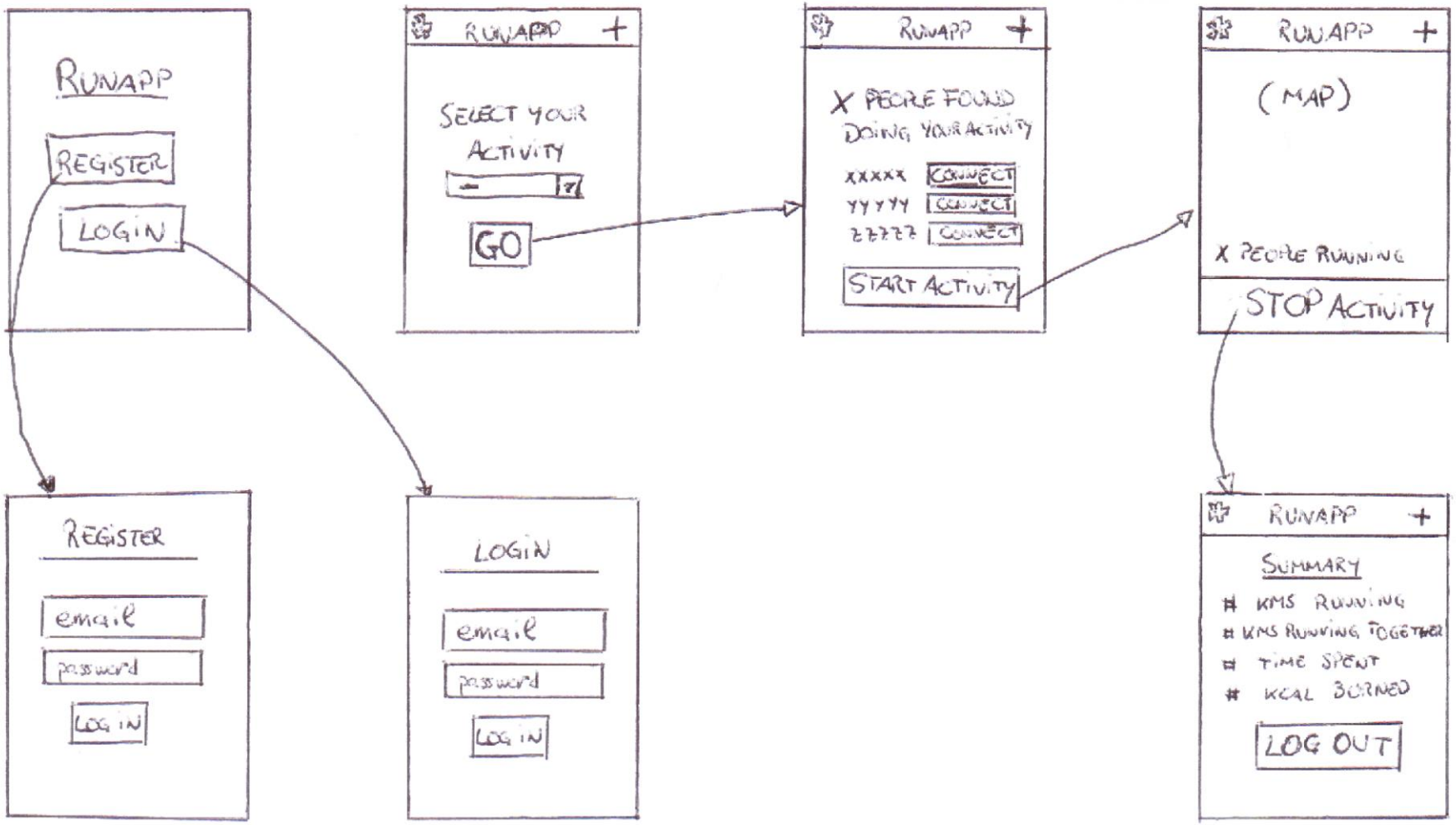
1. Use Cases


Order food
Alert "food order"
Alert "food ready"
Provide review
Alert "arrived at destination"
Customer
Chef
Delivery

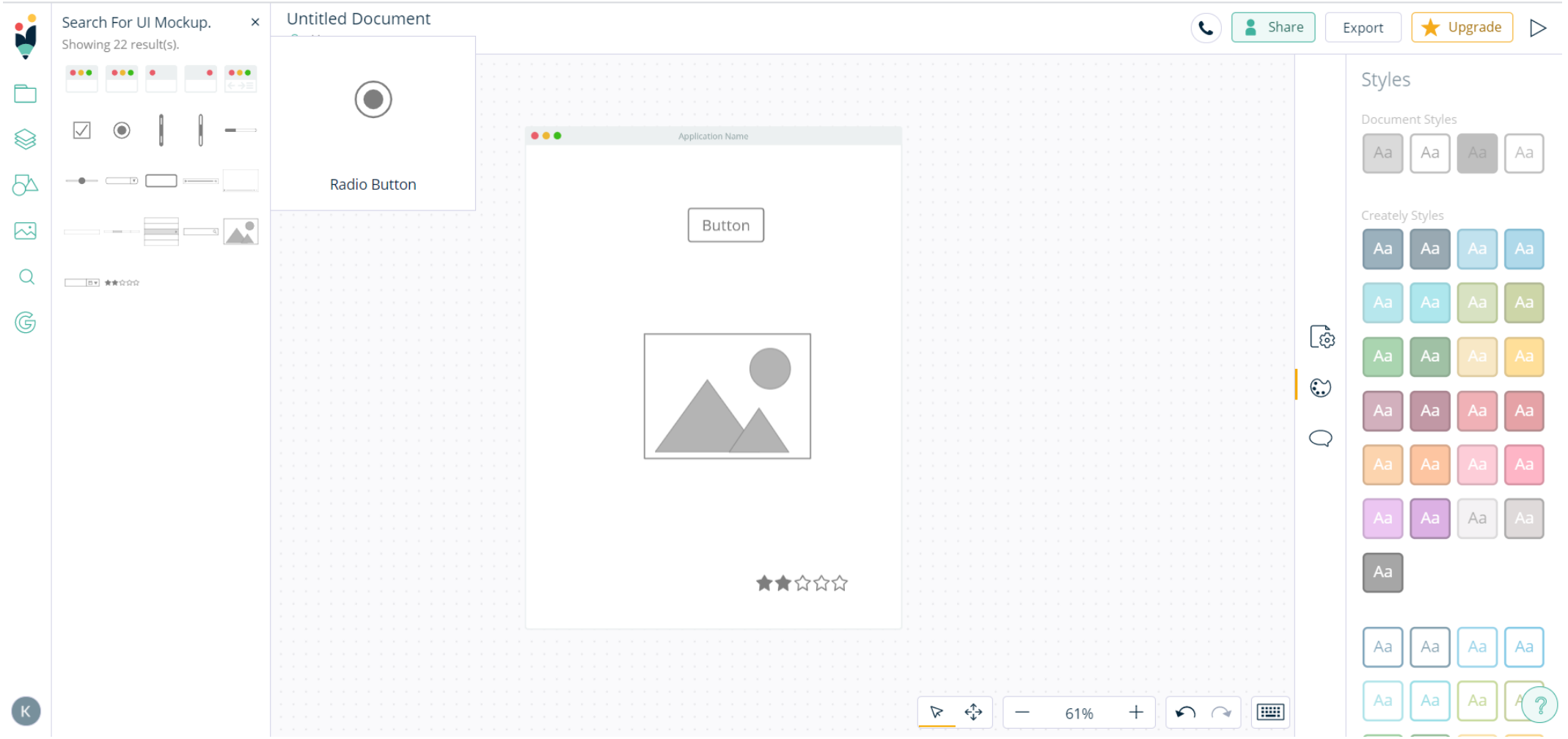
2. Mock-Ups

Es mostrarà una gràfica amb les calories ingerides cada dia dels últims 30 dies

2. Mock-Ups


RUNAPP
REGISTER
LOGIN
RUNAPP
SELECT YOUR ACTIVITY
GO
RUNAPP
X PEOPLE FOUND DOING YOUR ACTIVITY
XXXXX CONNECT
YYYYY CONNECT
ZZZZZ CONNECT
START ACTIVITY
RUNAPP
(MAP)
X PEOPLE RUNNING
STOP ACTIVITY
REGISTER
email
password
LOG IN
LOGIN
email
password
LOG IN
RUNAPP
SUMMARY
KMS RUNNING
KMS RUNNING TOGETHER
TIME SPENT
KCAL BURNED
LOG OUT

2. Mock-Ups


Search For UI Mockup.
Showing 22 result(s).
Untitled Document
Share
Export
Upgrade
Radio Button
Application Name
Button
Styles
Document Styles
Creately Styles
61%

3. Co-Creation Activities

- Activities with the users, developers and other actors to:
  - ✓ Engage users in the software development process
  - ✓ Collect feedback & suggestions
  - ✓ Identify potential concerns & obstacles

Agenda	
09h-10h:	Project presentation
10h-11h:	Use cases & mock-ups
11h-12h:	Focus groups
12h-13h:	Brainstorming
13h-14h:	Summary & next steps

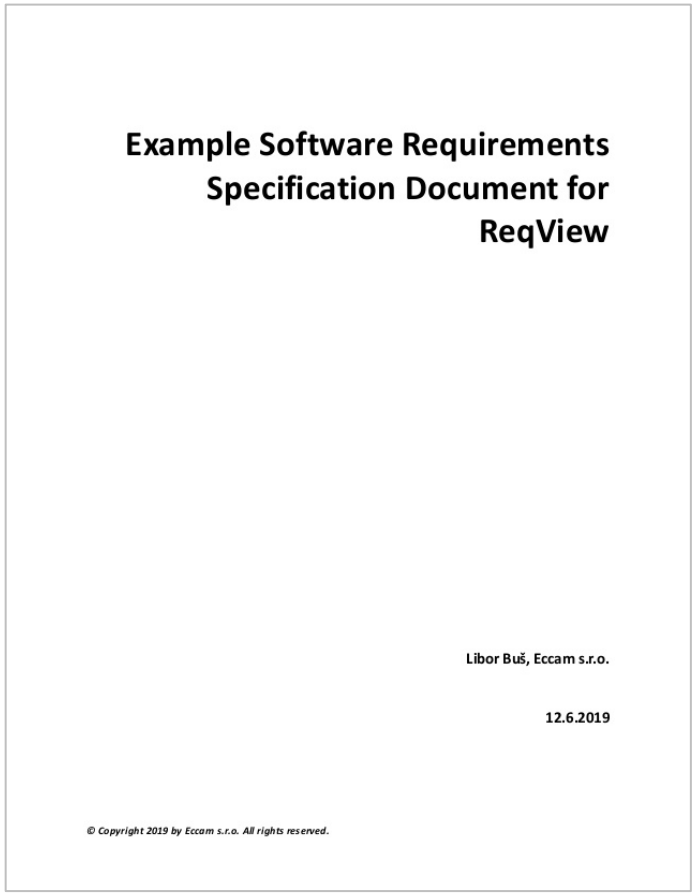
3. Co-Creation Activities

	Advantage	Limitation
Survey	Structured	Limited dialogue
Interviews	Improved participation	Lack of consensus
Meeting / Workshop	Improved consensus	Bias

- Co-creation activities need to be prepared in advance (e.g. focus questions)
- Use a reasonable number of participants/users (e.g. less than 20)

4. Requirements

- Describe at least the minimum set of requirements
- Structured: e.g. definitions, functional requirements, system requirements, etc…
- Use tables and graphs as much as possible
- Use simple natural language for users, more technical for developers
- There are standards for software specifications (e.g. IEEE)



Example Software Requirements Specification Document for ReqView

Libor Buš, Eccam s.r.o.

12.6.2019

4. Requirements

- Students will be able to download classes and upload deliverables
  - ✓ Students will be able to download classes
  - ✓ Students will be able to upload deliverables
- Students will be able to log in using their password, first name, family name and other student information
  - ✓ Using their password, first name, family name and student ID
- Each page of the system should be able to load in short time
  - ✓ Each page of the system should be able to load in less then 5 seconds

Project 2021

Note-taking App integrating multiple options:

- Text
- Photo
- Sound
- Link (e.g. Facebook event)
- Drawing
- Calendar
- File
- Etc

Project 2021

Note-taking App integrating multiple functionalities:

- Add
- Edit
- Remove
- Share
- See statistics
- Add to calendar
- Mark as important
- Etc

Project 2021

Can be domain-specific:

- Personal
- Group (e.g. family, friends)
- Work
- Sports
- Cultural events
- Political
- News
- Etc…

First Deliverable

- Short presentation of the app (One paragraph)
- Use Case(s)
- Mock-ups of UIs (+ one sentence per interface)
- Table of requirements (Detailed):
  - ✓ 5 functional requirements
  - ✓ 2 non-functional requirements

Next Steps

- Your next week:
  - ✓ Discuss your idea with your team-mates
  - ✓ Define users
  - ✓ Define functionalities
  - ✓ Draft a first version of the use cases & mock-ups
- Next class: Example of Deliverable 1 (use case, mock-up, requirements, etc)

Question 1

Of these requirements, which ones are most important?

A. User requirements.

B. Legal requirements.

C. System requirements.

D. They are equally important in a proper software development process.

Question 2

Among these steps, what comes first in the requirement gathering process?

A. First draft of the requirements.

B. First draft of the mock-ups.

C. User identification for requirement gathering.

D. Discussion with experts.

Question 3

Use cases are graphical models that are used to represent:

A. Detailed requirements of the app.

B. High-level view of the app’s components.

C. A summary of the main users.

D. A summary of the main user interfaces.

Question 4

For a software app that aims to predict heart attacks, which users should be interviewed for the requirement analysis?

A. Patients as they are the end-users.

B. Cardiologists only as they are the clinical experts.

C. Cardiologists and patients.

D. Cardiologists, patients and general practitioners.

Question 5

For a software app that aims to track the location of Covid-19 patients, which users should be interviewed for the requirement analysis?

A. Citizens as they are the end-users.

B. Medical doctors.

C. Covid-19 specialists.

D. Citizens and lawyers (to address data privacy).

Question 6

Which language should you use for requirement documentation?

A. English to make sure all customers understand

B. Simple language that can be understood by all

C. Technical language of the application domain

D. Unambiguous, detailed and specific descriptions

Question 7

For an app estimating heart attack risk, how many users should be involved in the requirement gathering process?

A. At least one male and one female user

B. N=50

C. N=20

D. N=5

Question 8

How much time should requirement gathering last?

A. 3 to 6 months

B. Till the beginning of the system design

C. Until testing is complete successfully

D. Until requirement documentation is finalised

Question 9

What comes after the co-creation workshops?

A. Writing of requirements document

B. Another iteration of use case definition

C. Software design

D. All three are possible

Question 10

A new iteration of requirement analysis may be needed depending on:

A. Feedback from developers

B. Feedback from users that tested the first prototype

C. Feedback from system design

D. Quality of the mock-ups