1. Tipos de requerimientos

Tipos de requerimientos

- Funcionales/No Funcionales
  - Cualidades de software
- Por formato:
  - User stories
  - Use cases
  - Requirement specifications
    - Verifiable
    - Notations
  - Context scenarios (from Personas)
- User vs system requirements
- Personas (perfiles de usuario)

FUNCIONALES/NO FUNCIONALES

Requirement types

- Functional
- Non functional

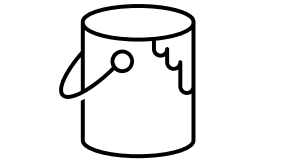
Functional

- “System behaviors or capabilities … the system is required to perform [3].”

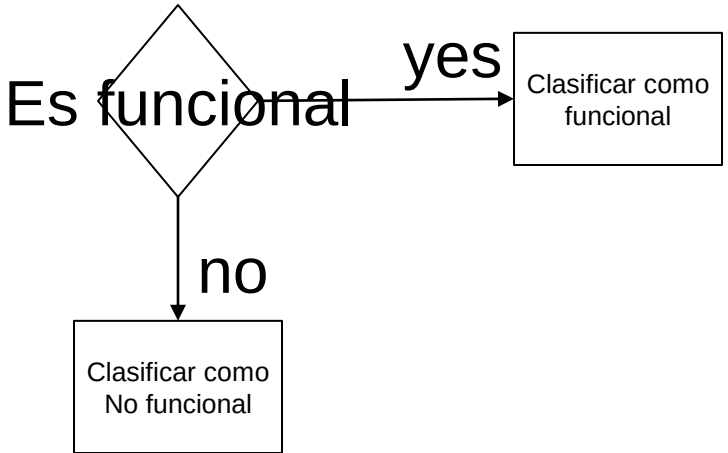
- This behavior may be expressed as: [3]
  - Services
  - Tasks or
  - Functions
- What?

Definitions by Sommerville [2]

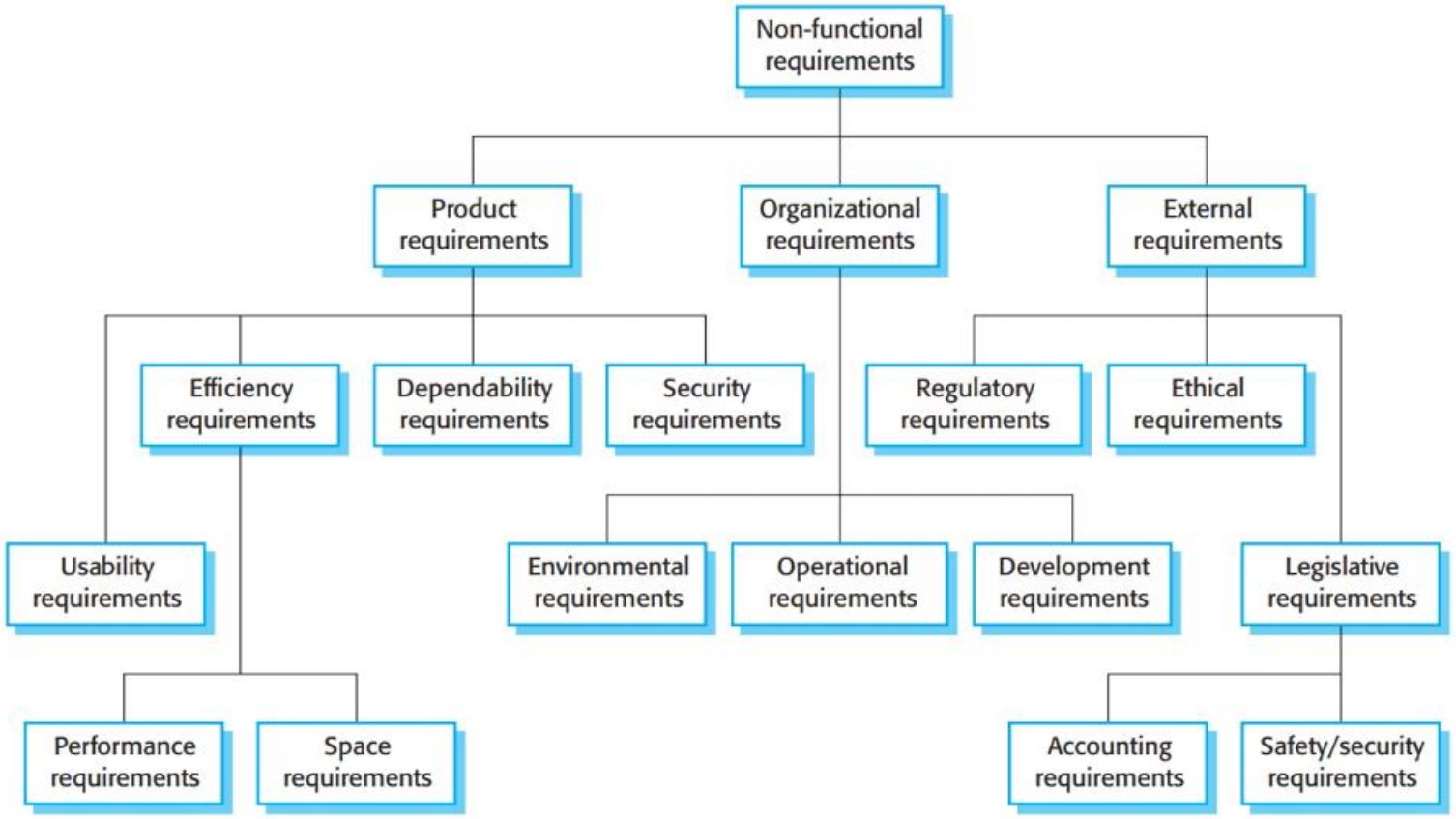
1. *Functional requirements* These are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations. In some cases, the functional requirements may also explicitly state what the system should not do.
2. *Non-functional requirements* These are constraints on the services or functions offered by the system. They include timing constraints, constraints on the development process, and constraints imposed by standards. Non-functional requirements often apply to the system as a whole rather than individual system features or services.



Funcional



Non-Functional requirements [2]



Examples [2]

PRODUCT REQUIREMENT
The Mentcare system shall be available to all clinics during normal working hours (Mon–Fri, 08:30–17:30). Downtime within normal working hours shall not exceed 5 seconds in any one day.

ORGANIZATIONAL REQUIREMENT
Users of the Mentcare system shall identify themselves using their health authority identity card.

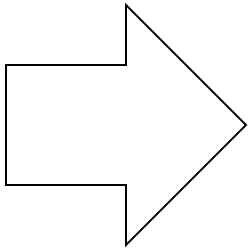
EXTERNAL REQUIREMENT
The system shall implement patient privacy provisions as set out in HStan-03-2006-priv.

Non functional [1]

- Many times they are considered as:
  - A category to group all those requirements that are _____
- Typically, they are also software qualities

Non functional [1]

- Many times they are considered as:
  - A category to group all those requirements that are not functional
- Typically, they are also software qualities + other (e.g. constraints, external interface requirements, etc.)



Qualities [3]

- Performance
- Usability
- Security
- Maintainability
- Reliability
- Efficiency
- Safety
- Portability
- Etc.

Videos

- Use Cases and Functional Requirements
  - http://www.youtube.com/watch?v=HshfGCgWaE4&feature=relmfu
- Non-Functional Requirements
  - http://www.youtube.com/watch?v=ITS8sAkwRvQ&feature=relmfu

Nonfunctional Requirements

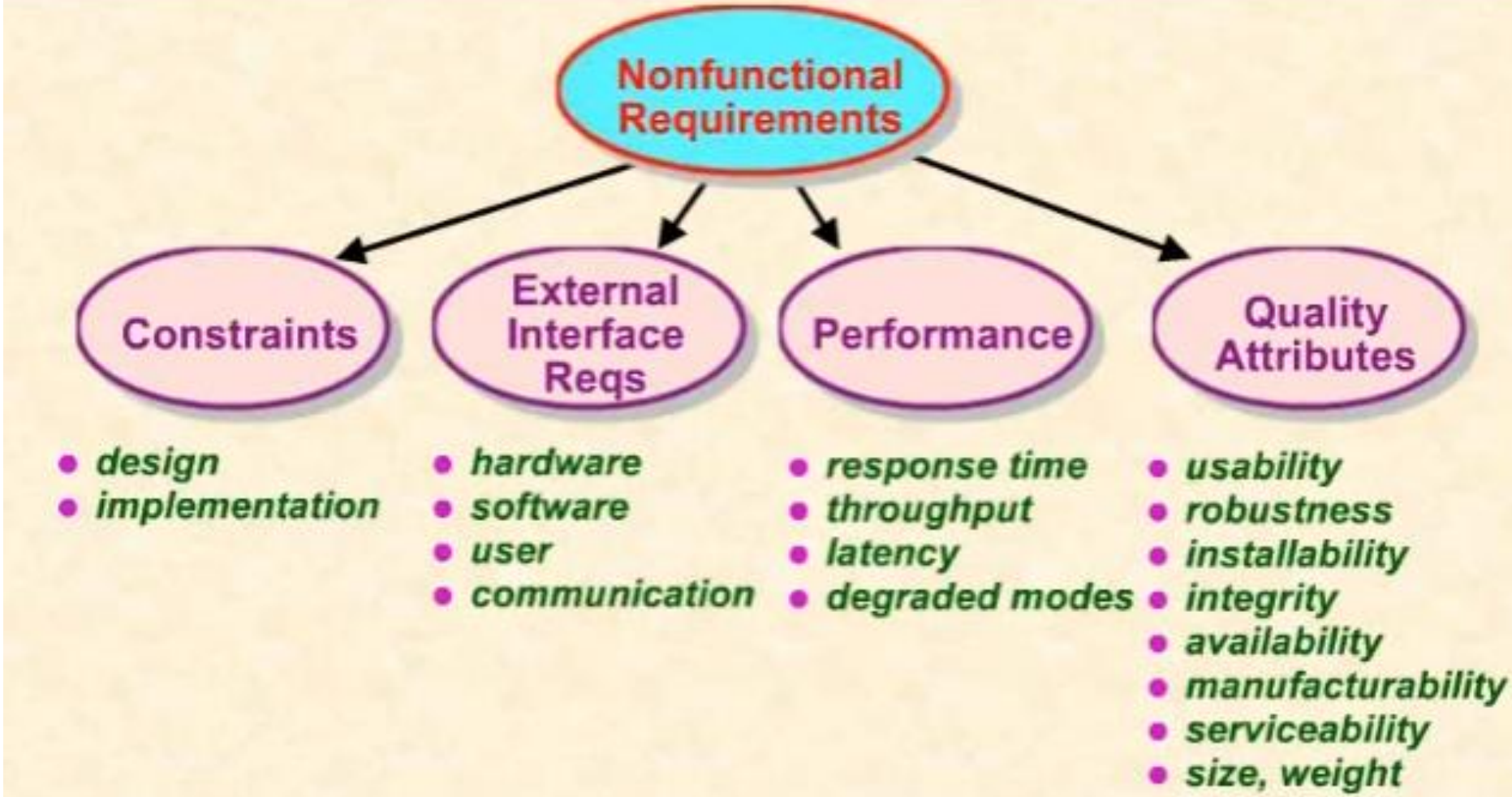
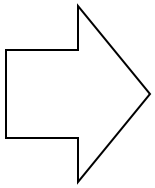
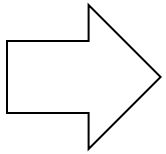


Figure adapted from [3]

Why bother?

- Functional requirements  GUI

- Non-Functional requirements  Software Architecture

Activity: E10-2 Repaso de definición de requerimientos

Activity: Write non-functional requirements

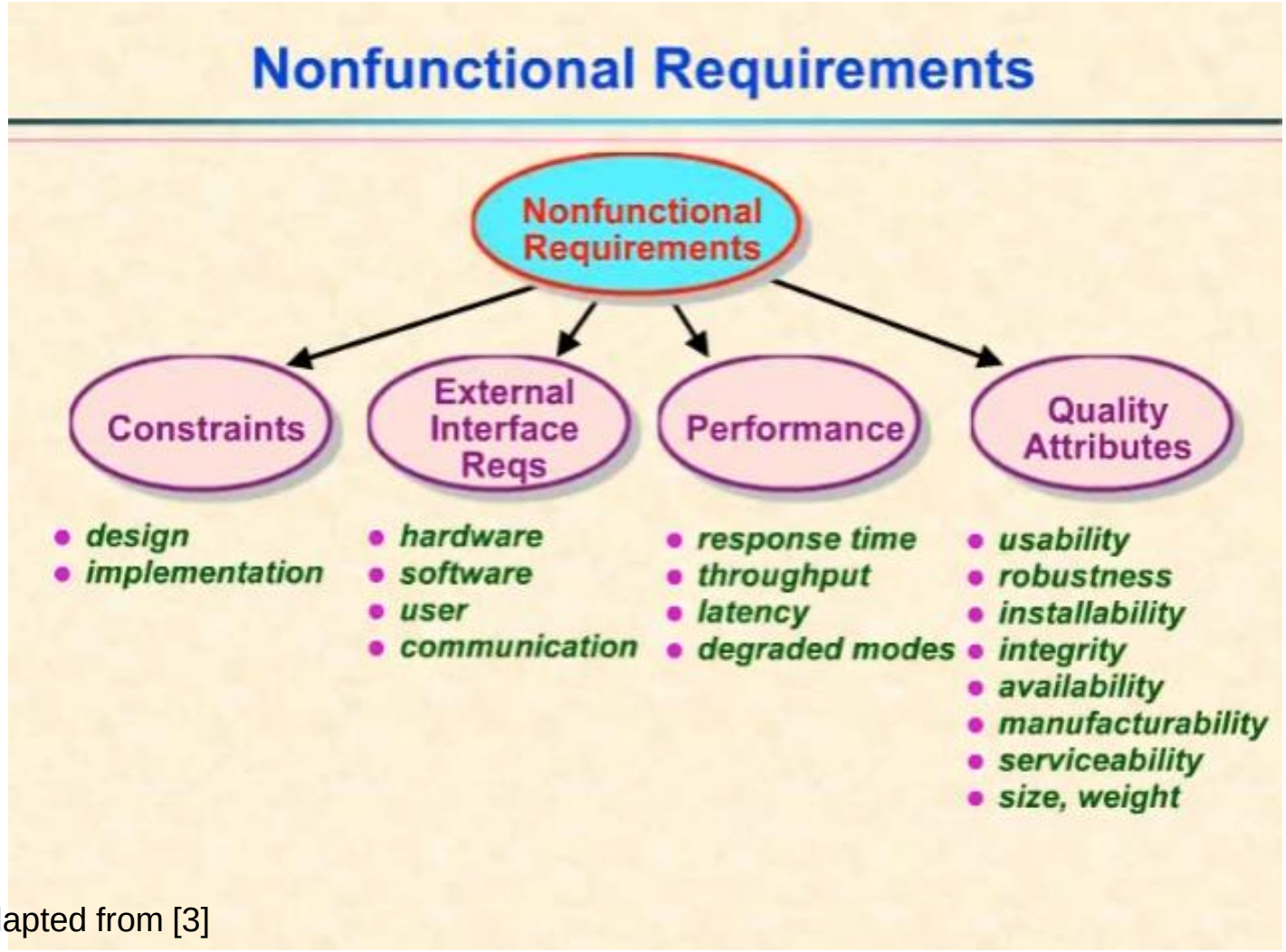


Figure adapted from [3]

References

1. IEEE Computer Society (1998). *IEEE Recommended Practice for Software Requirements Specifications*, IEEE.
2. Sommerville, I. (2011). *Ingeniería de software*, Addison Wesley, México.
3. Karl Wiegers (2011). Speaking of Requirements: Non-Functional Requirements, http://www.youtube.com/watch?v=ITS8sAkwRvQ&feature=relmfu

POR FORMATO

Por formato

- User stories
- Use cases
- Requirement specifications
- Contextual scenarios (from Personas)

Formato de Mike Cohn

USER STORIES

Acknowledgements

- This presentation is adapted from [1, 2]

User Stories

Card

As a project manager

I need to create a project schedule

So that I know when all the project tasks happen, so that I can schedule resources to do those tasks

Conversation

The 3 C's of user stories [1]

- C: Card
- C: Conversation
- C: Confirmation

- Each user story is composed of 3 aspects:
  - Written description of the story (for planning)
  - Conversations about the story, as foundational knowledge forming the story
  - Acceptance criteria which conveys and document details that can be used to determine when a story is "complete"

User Story Template

CERTSCHOOL

User Story Template (by Mike Cohn)

"As a <type of user>…
I want <some goal> …
so that <some reason>."

As an Account Manager I want to see sales per customer so that I can determine which customers are most profitable

CertSchool
Certified Agile Practitioner

INVEST in Good Stories !!!

CERTSCHOOL

I	- Dependencies lead to problems estimating and prioritizing - Can ideally select a story to work on without pulling in 18 other stories	Independent user stories
N	- Stories are not contracts - Leave or imply some flexibility	Negotiable
V	- To users or customers, not developers - Rewrite developer stories to reflect value to users or customers	Value to the end customer
E	Because plans are based on user stories, we need to be able to estimate them	Estimatable
S	- Small enough to complete in one sprint if you're about to work on it - Bigger if further off on the horizon	Small
T	Testable so that you have a easy, binary way of knowing whether a story is finished • Done or not done; no "partially finished" or "done except"	Testable

ACCEPTANCE CRITERIA [2]

Accrit Template

Acceptance Criterion (*Accrit*)

"Given <Precondition>…

When <Actor + Action> …

Then <Observable Result>."

Note: Accrit is NOT a replacement for test cases

Accrit Example

"GIVEN the logged-in user has "Admin" privileges..

WHEN he/she attempts to enters a valid combination of Account and Routing Number to transfer funds..

THEN the funds are transferred through the payment gateway and a success message is displayed."

Note: Accrit is NOT a replacement for test cases

References

- 1. Anurag Saksena (2013). “Agile Methodology Episode 3 – User Stories”, URL: https://youtu.be/C-qBLcBWVmY
- 2. Anurag Saksena (2013). “Agile Methodology Episode 5 -- Acceptance Criteria”, URL: https://youtu.be/fwL4a0pxsmc

USE CASES

Use case formats

- Brief
  - 3 sections
  - All elaborated at the beginning
- Expanded/Fully dressed
  - Detailed description
  - A subset is elaborated in each iteration

Use case formats [1,2]

Brief

- Name
- Actors
- Description [2]

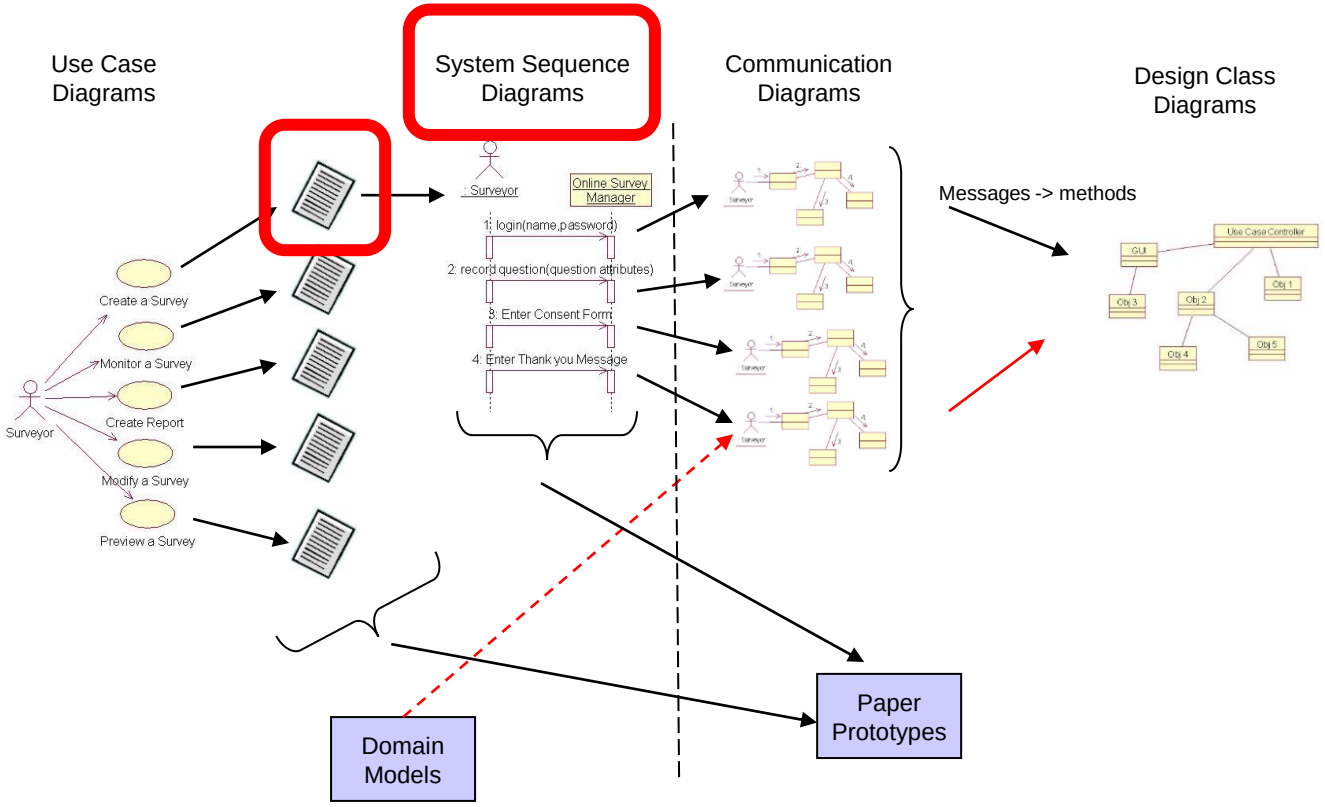
Expanded

- Name
- Actors
- Description [2]
- Stakeholders and Interests
- Preconditions
- Success Guarantee (Postconditions)
- Main Success Scenario (or Basic Flow, Happy path)
- Extensions (or Alternative Flows)
- Special Requirements
- Technology and Data Variations List
- Frequency
- Open issues

Main Success Scenario [3]

Actor Action	System Response
1. Customer arrives at POS checkout with goods and/or services to purchase.	
2. Cashier starts a new sale.	
3. Cashier enters item identifier *Cashier repeats steps 3-4 until indicates done.*	4. System records sale line item and presents item description, price, and running total. Price calculated from a set of price rules.
5. Cashier finalizes the sale	6. System presents total with taxes calculated
7. Cashier tells Customer the total and asks for payment.	
8. Customer pays and the Cashier introduces the payment	9. System handles the payment
	10. System logs completed sale and sends sale and payment information to the external Accounting system (for accounting and commissions) and Inventory system (to update inventory).
	11. System presents receipt.
12. Customer leaves with receipt and goods (if any).	

UML process [1]



References

1. Larman, C. (2004). *UML y patrones: Una introducción al análisis y diseño orientado a objetos y al proceso unificado*, Pearson Educación, 2ª ed., España.
2. Valtech (1999). Chapter 4: Creating Use Cases. Presentation
3. Larman, C. (2005). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, Pearson Educación, 3ª ed., N.J., USA, p. 674.

REQUIREMENT SPECIFICATIONS

Examples of requirement specifications

- In 60% of the cases, the program output should be produced within 20 seconds the event has started; and 30 seconds in 100% of the cases. [1]

- The TMW shall always display the identifier for the current week. [2]

VERIFIABLE REQUIREMENTS

Requirements should be verifiable [1]

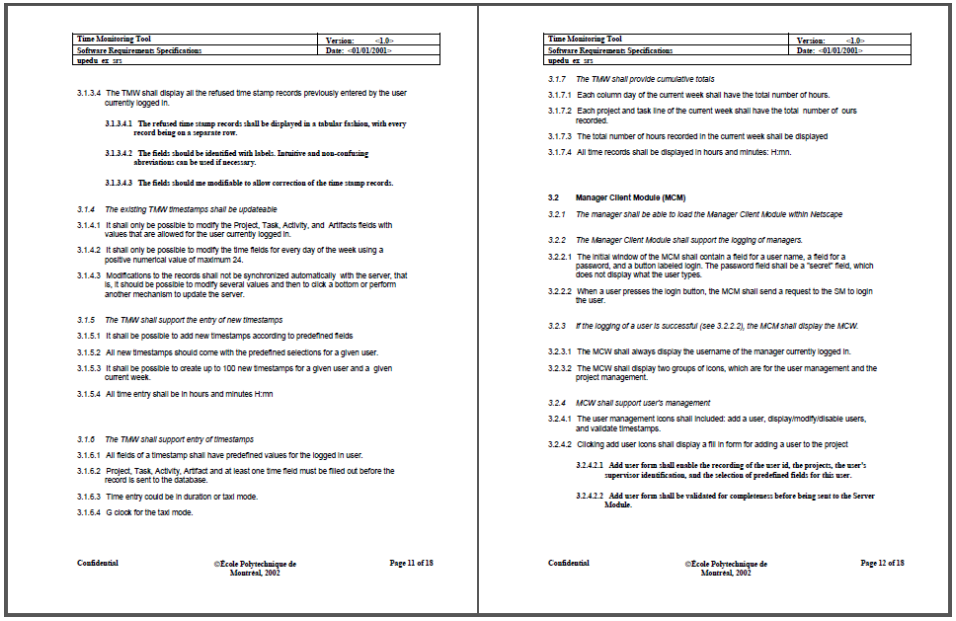
- i.e., a person or machine should be able to check if the software has satisfied the requirement.

- They should be specific enough that we can determine what evidence is needed to verify them

Image by Gerd Altmann from Pixabay

For verifying requirements

- The QA team (testing) can check if the requirements in the SRS are satisfied



Time Monitoring Tool	Version: -1.0-
Software Requirements Specifications:	Date: -01/01/2001-
upedu_ex_srs	

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

3.1.5 The TMW shall support the entry of new timestamps

3.1.5.1 It shall be possible to add new timestamps according to predefined fields

3.1.5.2 All new timestamps should come with the predefined selections for a given user.

3.1.5.3 It shall be possible to create up to 100 new timestamps for a given user and a given current week.

3.1.5.4 All time entry shall be in hours and minutes H:mn

3.1.6 The TMW shall support entry of timestamps

3.1.6.1 All fields of a timestamp shall have predefined values for the logged in user.

3.1.6.2 Project, Task, Activity, Artifact and at least one time field must be filled out before the record is sent to the database.

3.1.6.3 Time entry could be in duration or taxi mode.

3.1.6.4 G clock for the taxi mode.

Time Monitoring Tool Version: -1.0-
Software Requirements Specifications: Date: -01/01/2001-
upedu_ex_srs

3.1.7 The TMW shall provide cumulative totals

3.1.7.1 Each column day of the current week shall have the total number of hours.

3.1.7.2 Each project and task line of the current week shall have the total number of ours recorded.

3.1.7.3 The total number of hours recorded in the current week shall be displayed

3.1.7.4 All time records shall be displayed in hours and minutes: H:mn.

3.2 Manager Client Module (MCM)

3.2.1 The manager shall be able to load the Manager Client Module within Netscape

3.2.2 The Manager Client Module shall support the logging of managers.

3.2.2.1 The initial window of the MCM shall contain a field for a user name, a field for a password, and a button labeled login. The password field shall be a "secret" field, which does not display what the user types.

3.2.2.2 When a user presses the login button, the MCM shall send a request to the SM to login the user.

3.2.3 If the logging of a user is successful (see 3.2.2.2), the MCM shall display the MCW.

3.2.3.1 The MCW shall always display the username of the manager currently logged in.

3.2.3.2 The MCW shall display two groups of icons, which are for the user management and the project management.

3.2.4 MCW shall support user's management

3.2.4.1 The user management icons shall included: add a user, display/modify/disable users, and validate timestamps.

3.2.4.2 Clicking add user icons shall display a fill in form for adding a user to the project

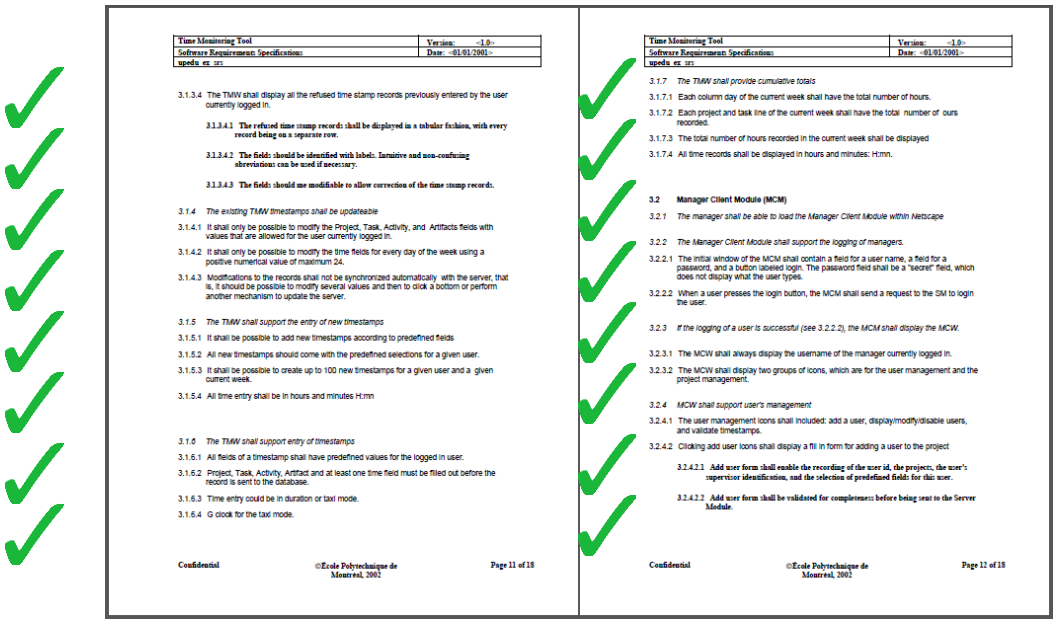
3.2.4.2.1 Add user form shall enable the recording of the user id, the projects, the user's supervisor identification, and the selection of predefined fields for this user.

3.2.4.2.2 Add user form shall be validated for completeness before being sent to the Server Module.

Software Requirements Specification (SRS)

For verifying requirements

- The QA team (testing) can check if the requirements in the SRS are satisfied



Time Monitoring Tool | Version: -1.0-
Software Requirements Specifications: | Date: -01/01/2001-
upedu_ex_srs

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

3.1.5 The TMW shall support the entry of new timestamps

3.1.5.1 It shall be possible to add new timestamps according to predefined fields

3.1.5.2 All new timestamps should come with the predefined selections for a given user.

3.1.5.3 It shall be possible to create up to 100 new timestamps for a given user and a given current week.

3.1.5.4 All time entry shall be in hours and minutes H:mn

3.1.6 The TMW shall support entry of timestamps

3.1.6.1 All fields of a timestamp shall have predefined values for the logged in user.

3.1.6.2 Project, Task, Activity, Artifact and at least one time field must be filled out before the record is sent to the database.

3.1.6.3 Time entry could be in duration or taxi mode.

3.1.6.4 G clock for the taxi mode.

Confidential ©Ecole Polytechnique de Montréal, 2002 Page 11 of 18

Time Monitoring Tool | Version: -1.0-
Software Requirements Specifications: | Date: -01/01/2001-
upedu_ex_srs

3.1.7 The TMW shall provide cumulative totals

3.1.7.1 Each column day of the current week shall have the total number of hours.

3.1.7.2 Each project and task line of the current week shall have the total number of ours recorded.

3.1.7.3 The total number of hours recorded in the current week shall be displayed

3.1.7.4 All time records shall be displayed in hours and minutes: H:mn.

3.2 Manager Client Module (MCM)

3.2.1 The manager shall be able to load the Manager Client Module within Netscape

3.2.2 The Manager Client Module shall support the logging of managers.

3.2.2.1 The initial window of the MCM shall contain a field for a user name, a field for a password, and a button labeled login. The password field shall be a "secret" field, which does not display what the user types.

3.2.2.2 When a user presses the login button, the MCM shall send a request to the SM to login the user.

3.2.3 If the logging of a user is successful (see 3.2.2.2), the MCM shall display the MCW.

3.2.3.1 The MCW shall always display the username of the manager currently logged in.

3.2.3.2 The MCW shall display two groups of icons, which are for the user management and the project management.

3.2.4 MCW shall support user's management

3.2.4.1 The user management icons shall included: add a user, display/modify/disable users, and validate timestamps.

3.2.4.2 Clicking add user icons shall display a fill in form for adding a user to the project

3.2.4.2.1 Add user form shall enable the recording of the user id, the projects, the user's supervisor identification, and the selection of predefined fields for this user.

3.2.4.2.2 Add user form shall be validated for completeness before being sent to the Server Module.

Confidential ©Ecole Polytechnique de Montréal, 2002 Page 12 of 18

Software Requirements Specification (SRS)

What do you observe?

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 **The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.**

3.1.3.4.2 **The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.**

3.1.3.4.3 **The fields should me modifiable to allow correction of the time stamp records.**

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

Examples adapted from [3]

Requirement specification have a code

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

sub reqs.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

Examples adapted from [3]

Is it satisfied?

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

Examples adapted from [3]

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

Examples adapted from [3]

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

Examples adapted from [3]

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

Examples adapted from [3]

3.1.3.4 The TMW shall display all the refused time stamp records previously entered by the user currently logged in.

3.1.3.4.1 The refused time stamp records shall be displayed in a tabular fashion, with every record being on a separate row.

3.1.3.4.2 The fields should be identified with labels. Intuitive and non-confusing abreviations can be used if necessary.

3.1.3.4.3 The fields should me modifiable to allow correction of the time stamp records.

3.1.4 The existing TMW timestamps shall be updateable

3.1.4.1 It shall only be possible to modify the Project, Task, Activity, and Artifacts fields with values that are allowed for the user currently logged in.

3.1.4.2 It shall only be possible to modify the time fields for every day of the week using a positive numerical value of maximum 24.

3.1.4.3 Modifications to the records shall not be synchronized automatically with the server, that is, it should be possible to modify several values and then to click a bottom or perform another mechanism to update the server.

Examples adapted from [3]

Example: A verifiable requirement

- In 60% of the cases, the program output should be produced within 20 seconds the event has started; and 30 seconds in 100% of the cases [1]

- Characteristics
  - Specific
  - Determine the evidence
  - Determine acceptance criteria and thresholds

Pitfall: Compound requirements

- Requirements that are actually 2 or more, but combined

> “It shall be possible to add new timestamps according to predefined fields and should come with the predefined selections for a given user. [3]”

Split the compound requirements

"It shall be possible to add new timestamps according to predefined fields and should come with the predefined selections for a given user. [3]"

"It shall be possible to add new timestamps according to predefined fields. [3]"

"All new timestamps should come with the predefined selections for a given user. [3]"

What is the problem

If I ask you: “Has the requirement been satisfied?” Your answer?

“It shall be possible to add new timestamps according to predefined fields and should come with the predefined selections for a given user. [3]”

“It shall be possible to add new timestamps according to predefined fields. [3]”

“All new timestamps should come with the predefined selections for a given user. [3]”

USER AND SYSTEM REQUIREMENTS

User requirements [2]

The user requirements for a system should describe the functional and nonfunctional requirements so that they are understandable by system users who don't have detailed technical knowledge. Ideally, they should specify only the external behavior of the system. The requirements document should not include details of the system architecture or design. Consequently, if you are writing user requirements, you should not use software jargon, structured notations, or formal notations. You should write user requirements in natural language, with simple tables, forms, and intuitive diagrams.

System requirements

System requirements are expanded versions of the user requirements that software engineers use as the starting point for the system design. They add detail and explain how the system should provide the user requirements. They may be used as part of the contract for the implementation of the system and should therefore be a complete and detailed specification of the whole system.

How to write specifications?

You may start with general statements first (user requirements)

- Don’t discard them
  - Just keep refining them

- Example
  - “It should have a good user interface.”

General

↓ Make it more specific

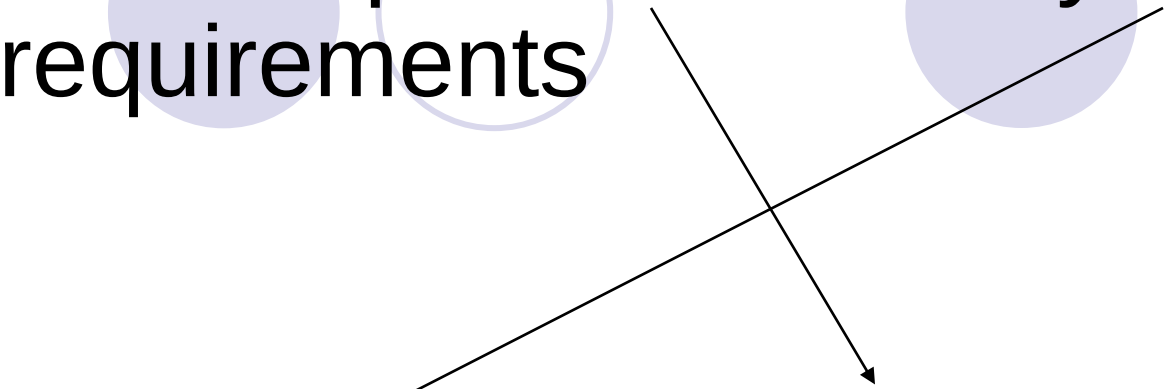
  - Answer: what do we mean by “good”

↓ Break it down

3.2 It should have a good user interface

- 3.2.1 The system shall provide the undo capability in predetermined actions: action 1, 2, 3, 4, 5, 6,…
- 3.2.2 The system should present only the necessary information on the screen, which is: data 1, data 2, data 3, data 4.
- 3.2.3 The system should show confirmation messages when deleting any element in a collection. These include: element 1, 2, 3, etc.
- …

User requirement and system requirements



3.1.7 The TMW shall provide cumulative totals

3.1.7.1 Each column day of the current week shall have the total number of hours.

3.1.7.2 Each project and task line of the current week shall have the total number of ours recorded.

3.1.7.3 The total number of hours recorded in the current week shall be displayed

3.1.7.4 All time records shall be displayed in hours and minutes: H:mn.

Expand it, give details in sub-requirements

NOTATIONS

Notations for writing system requirements [2]

Notation	Description
Natural language sentences	The requirements are written using numbered sentences in natural language. Each sentence should express one requirement.
Structured natural language	The requirements are written in natural language on a standard form or template. Each field provides information about an aspect of the requirement.
Graphical notations	Graphical models, supplemented by text annotations, are used to define the functional requirements for the system. UML (unified modeling language) use case and sequence diagrams are commonly used.
Mathematical specifications	These notations are based on mathematical concepts such as finite-state machines or sets. Although these unambiguous specifications can reduce the ambiguity in a requirements document, most customers don't understand a formal specification. They cannot check that it represents what they want, and they are reluctant to accept it as a system contract. (I discuss this approach, in Chapter 10, which covers system dependability.)

Used in the “Personas” technique

CONTEXT SCENARIOS

Example: Vivien Strong [1]

- A persona for a Personal Digital System (PDA):
  - Vivien Strong, a “real-estate agent in Indianapolis.
  - Vivien’s goals are to balance work and home life, cinch the deal, and make each client feel like he is her only client. [1]”

See the attached full example

An example context scenario

A persona for a PDA/phone convergence device and service: vivien Strong, a real-estate agent in Indianapolis. Vivien's goals are to balance work and home life, cinch the deal, and make each client feel like he is her *only* client.

Vivien's context scenario might be as follows:

1. Getting ready in the morning, Vivien uses her phone to check e-mail. It has a large enough screen and quick connection time so that it's more convenient than booting up a computer as she rushes to make her daughter, Alice, a sandwich for school.
2. Vivien sees an e-mail Vivien sees an email from her newest client, Frank, who wants to see a house this afternoon. Vivien entered his contact information a few days ago, so now she can call him with a simple action right from the email screen.
3. While on the phone with Frank Beatty and switches to speakerphone so she can look at the screen while talking. She looks at her appointment to see when she's free. When she creates a new appointment, the phone automatically makes it an appointment with Frank,

66

Full context scenario

- Check out the attached document

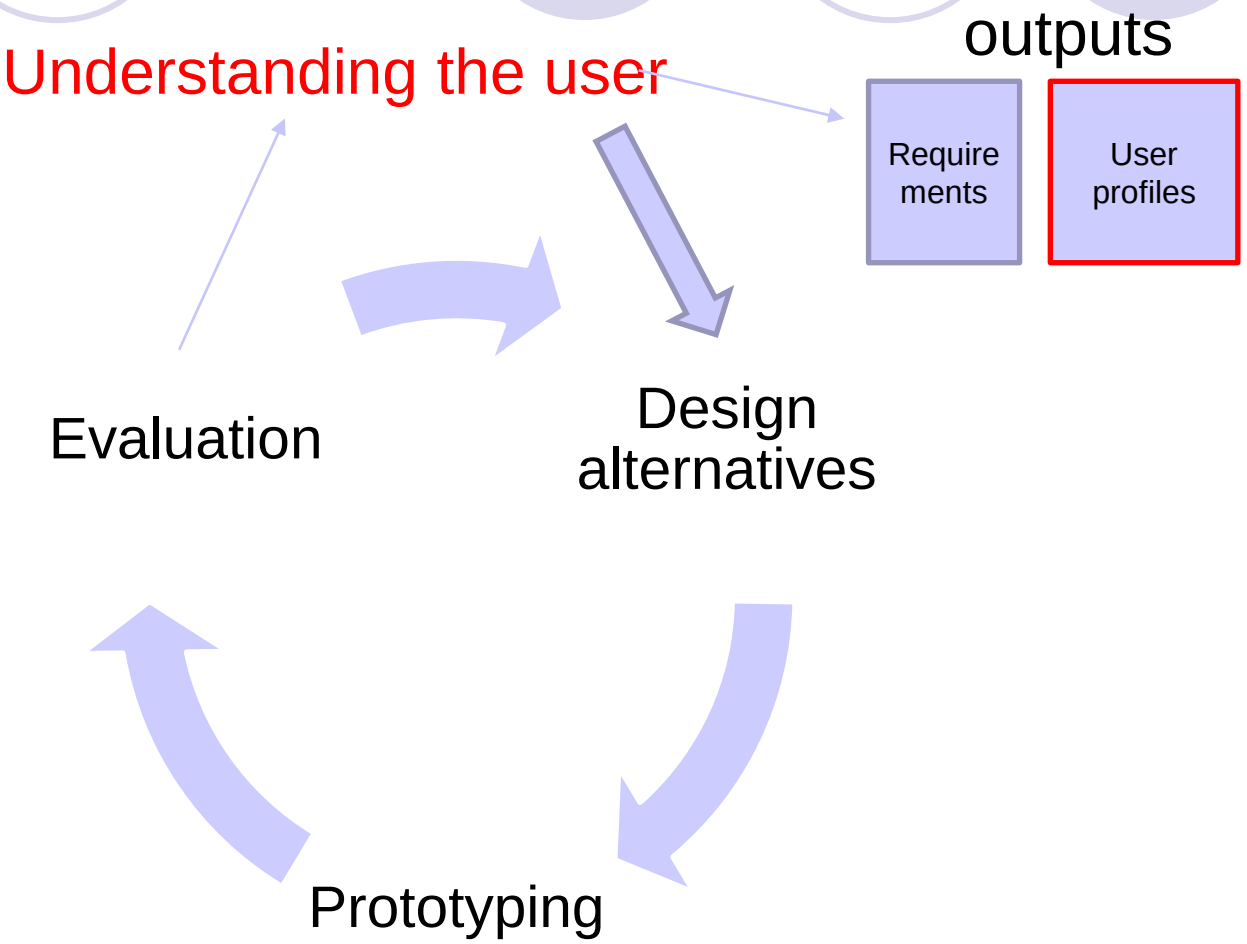
67

Tipos de requerimientos

- Funcionales/No Funcionales
  - Cualidades de software
- Por formato:
  - User stories
  - Use cases
  - Requirement specifications
    - Verifiable
    - Notations
  - Context scenarios (from Personas)
- User vs system requirements
- Personas (perfiles de usuario)

PERSONAS

Iterative design cycle



Based on [1]

70

The “Personas” Technique

- Most common technique to model users

71

CAPLIN

Ginnie

BACKGROUND

- 15, Female
- Ongoing Private Education
- Ambitious
- Comfortable using technology to communicate

MOTIVATIONS

- Keeping in touch with her network
- Fashion/street cred
- Keeping up with peers.

FRUSTRATIONS

- Sad people trying to be 'friends' on Facebook
- Having to be in bed @ 11pm
- Being swamped in friends updates
- Missing important status updates

Receives private tutoring in Maths and English as these are not her strong subjects. Enjoys playing for the school's 2nd teams for netball and Lacrosse and is good at art.

She loves recording her favourite shows: ER and Sun Valley High on Sky+ and spends some of her time on her Laptop that Daddy bought her watching videos on YouTube, downloading music, keeping up to date with her friends on Facebook and chatting via MS IM to her cousin who is at University in Leeds.

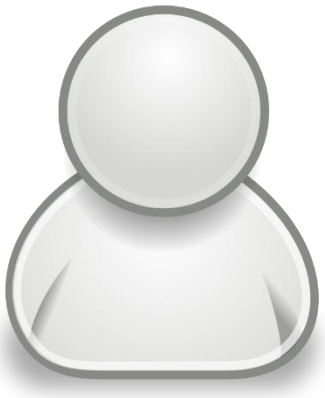
She loves Ugg boots and Abercrombie & Fitch and uses the Internet to shop and find the cheapest prices.

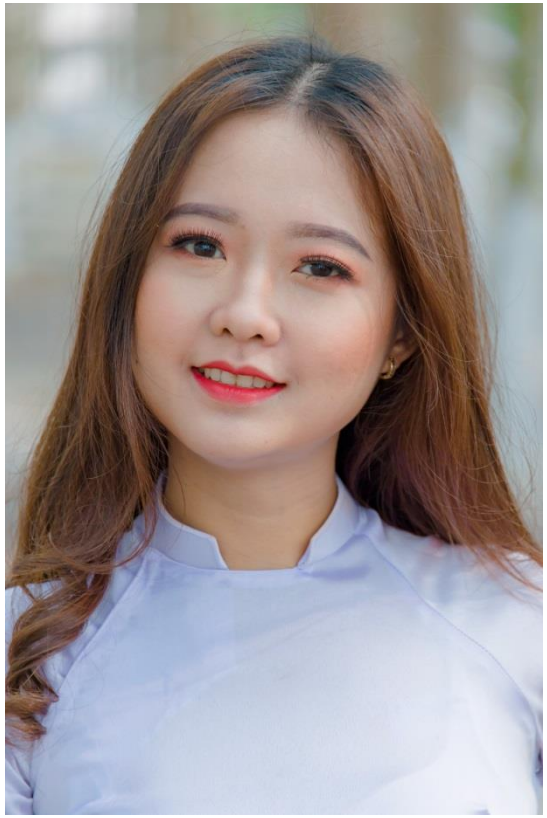
"I want to easily hook up with my friends whilst watching TV"

Personas are based on research

- Interview users
- Information about users supplied by stakeholders and subject matter experts
- Market research data such as from focus groups and surveys
- Data gathered from literature reviews and previous studies

73





74

Personas [4]

- It is a descriptive model of the user [1]
- Capture a set of user characteristics (user profile) [2]
- Not real people, but synthesised from real users [2]
- Bring them to life with a name, characteristics, goals, personal background [2] and a picture [1]
- Develop a small set of personas with one primary [2]

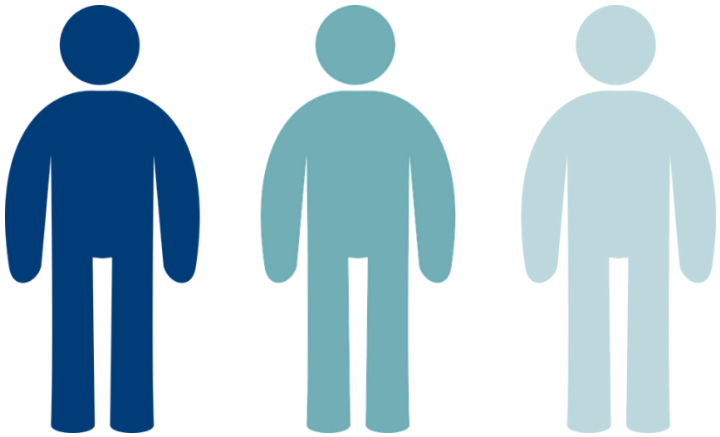
75

Advantages of Persona

- Better products
  - Designing for specific types of individuals with specific needs
- Better communication between analysts and engineers

76

Users can be better satisfied



David's goals

- To store instruments
- For work

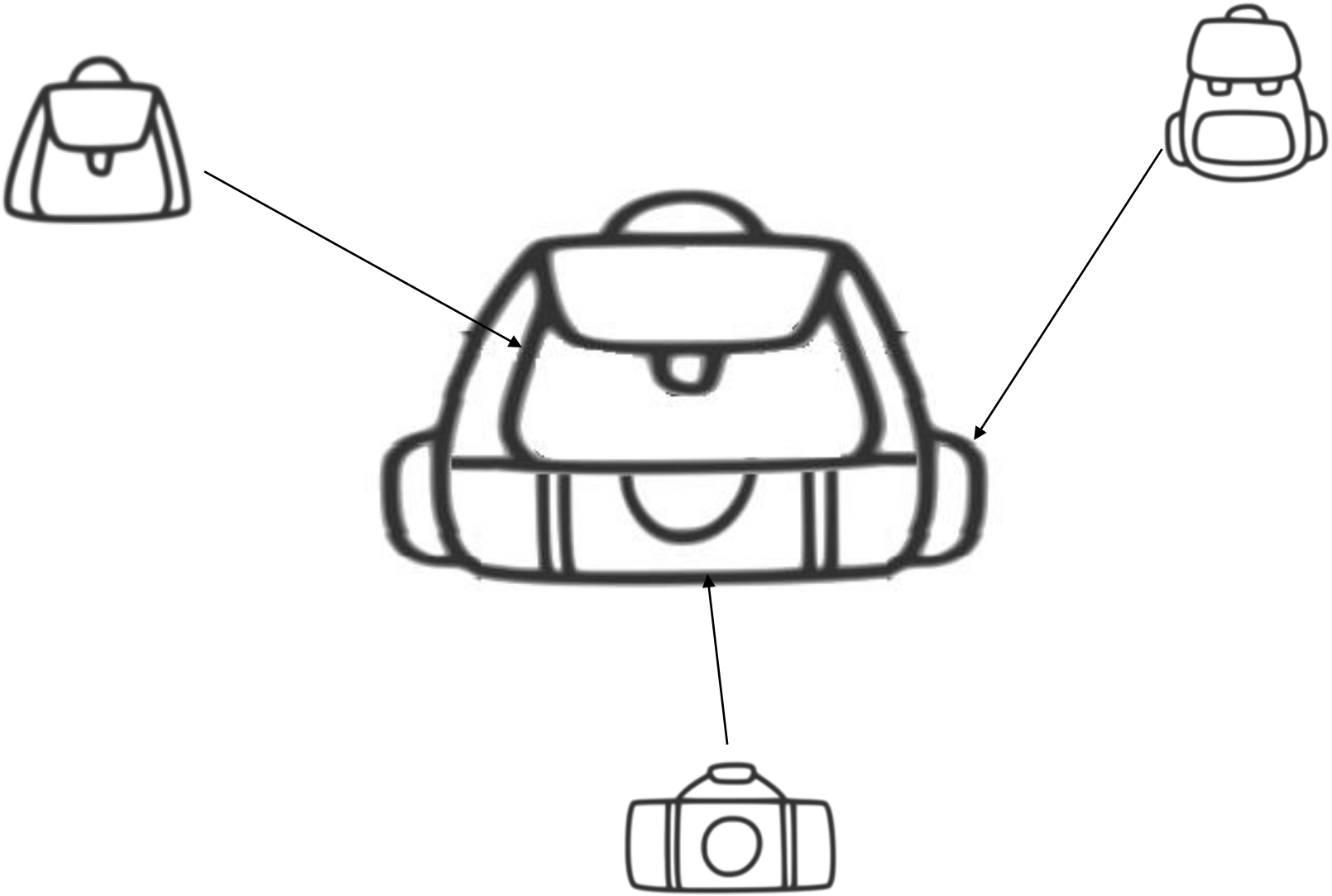
Carol's goals

- Personal items
- Casual

Steve's goals

- To hike
- That lasts

Figure inspired from [1]



A product for all

- But often products are designed for various individuals

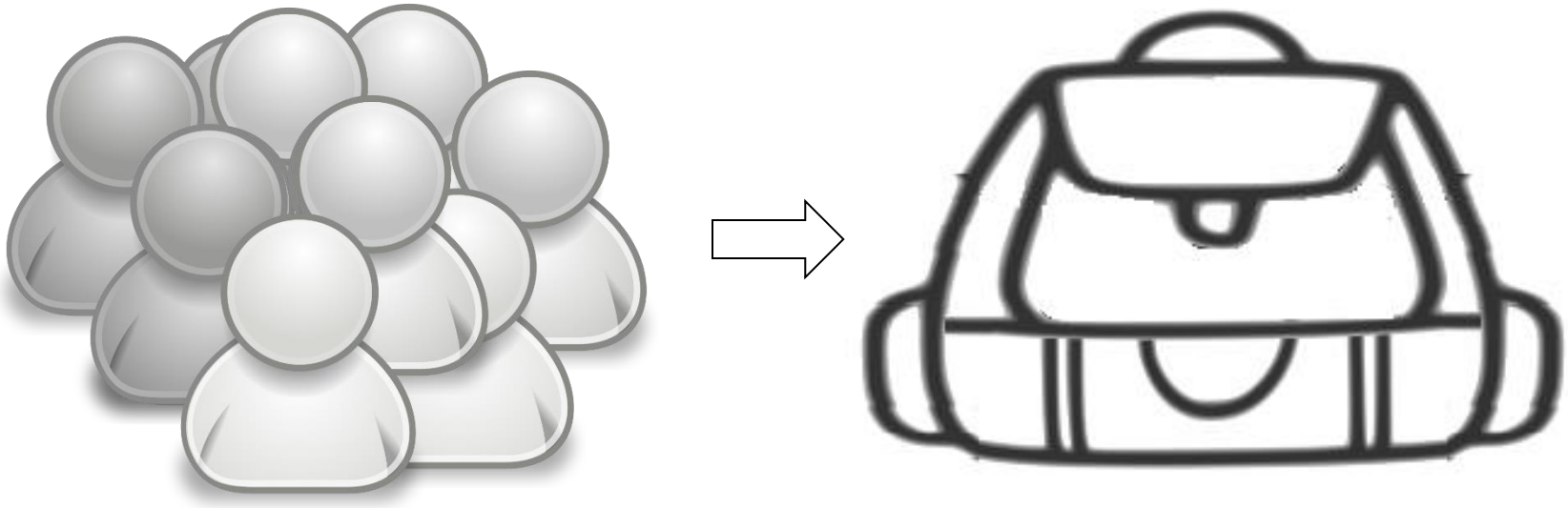
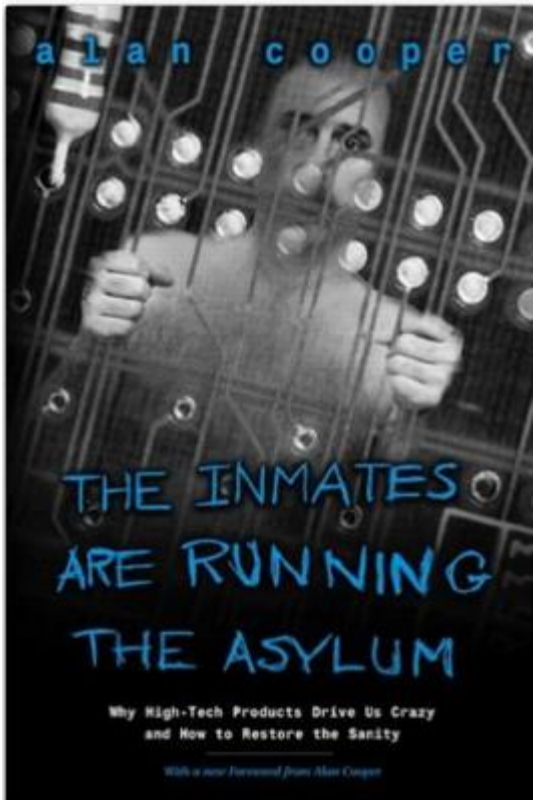


Figure inspired from [1]

Personas by Alan Cooper

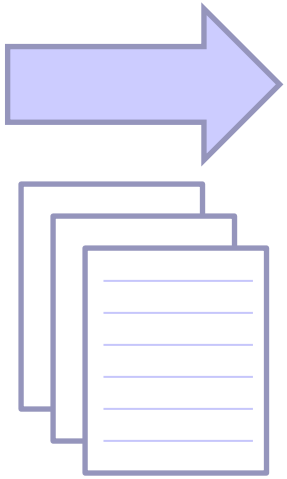
- Book:
  - The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity
- By Alan Cooper



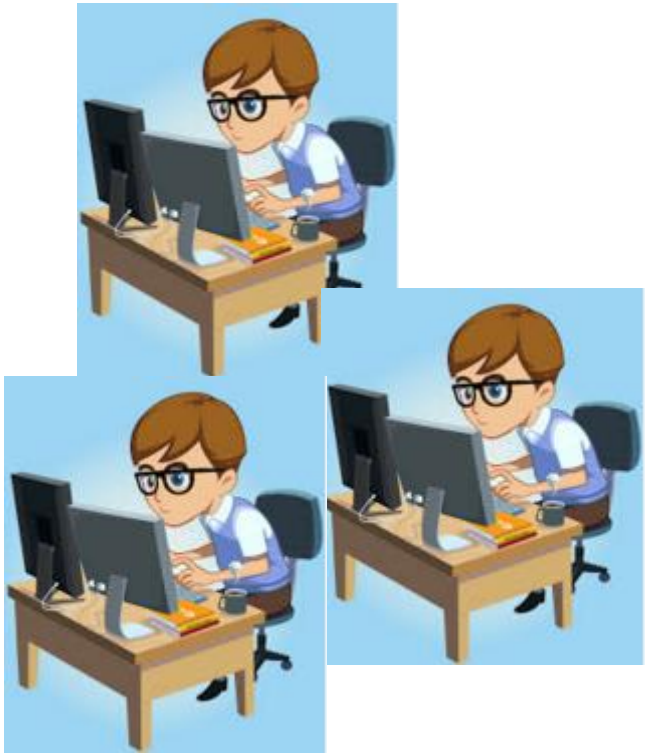
80

Origin of Personas

- Details of users were lost going from designers to developers



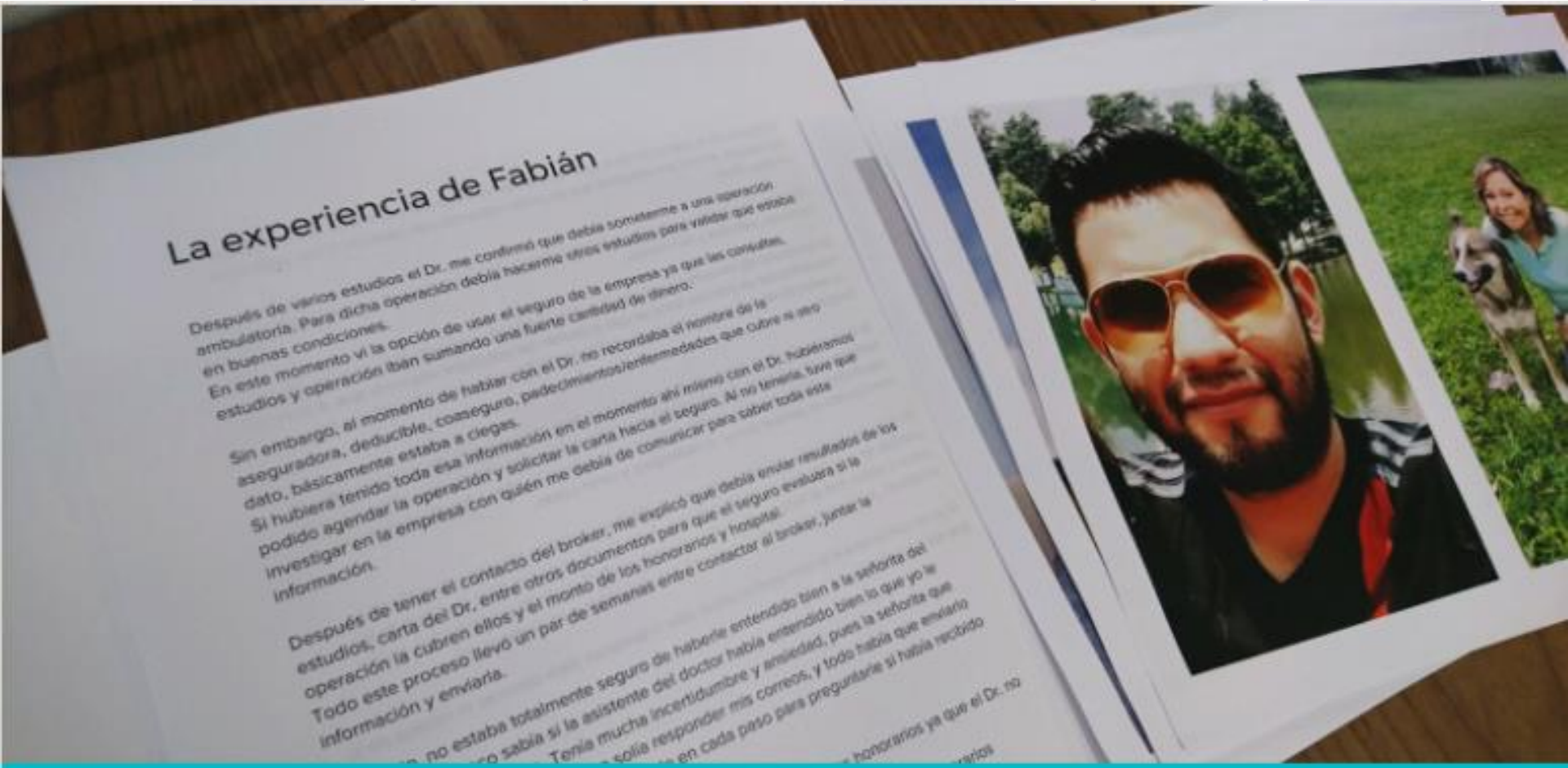
User profiles



81

- Identify their behavior and reasons behind it

Adopted from [12]



Generate empathy. Discover their motivations and aspirations.

Adopted from [12]


Melanie Swift
" I want my friends and family to see my son as I do."
6
1
2
2-2
2-1
3
4
5
GOOD JOB!
2-2

Adapted from [3]

ELEMENTS OF A PERSONA

86

How does a Persona look like? [1]

Persona narrative

+

Context scenarios

An example context story

A persona for a PDA (Personal Digital Assistant)/phone

Vivien Strong, a "real-estate agent in Indianapolis. Vivien's goals are to balance work and home life, clinch the deal, and make each client feel like he is her only client. [1]".

Vivien's context scenario might be as follows:

1. "Getting ready in the morning, Vivien uses her phone to check e-mail. It has a large enough screen and quick connection time so that it's more convenient than booting up a computer as she rushes to make her daughter, Alice, a sandwich for school.
2. Vivien sees an e-mail from her newest client, Frank, who wants to see a house this afternoon. Vivien entered his contact information a few days ago, so now she can call him with a simple action right from the email screen.
3. While on the phone with Frank, she switches to speakerphone so she can look at the screen while talking. She looks at her appointments to see when she's free. When she creates a new appointment, the phone automatically makes it an appointment with Frank, because it knows with whom she is talking. She quickly keys the address of the property into the appointment as she finishes her conversation.
4. After sending Alice off to school, Vivien heads into the real estate office to gather the papers she needs for the plumber working on in another property. Her phone has already updated her Outlook appointments, so the rest of the office knows where she will be in the afternoon.
5. The day goes by quickly, and she's running a bit late. As she heads towards the property she'll be showing Frank, the phone alerts her that her appointment is in 15 minutes. When she flips open the phone, it shows not only the appointment, but a list of all documents related to Frank, including emails, memos, phone messages, call logs to Frank's number, and even thumbnail pictures of the property that Vivien sent as e-mail attachments. Vivien presses the call button, and the phone automatically connects to Frank because it knows her appointment with him is soon. She lets him know she will be there in 20 minutes.
6. Vivien knows the address of the property, but is a bit unsure exactly where it is. She pulls over and taps the address she put into the appointment. The phone downloads directions a long with a thumbnail map showing her location relative to the destination.
7. Vivien gets to the property on time and starts showing it to Frank. She hears the phone ring from her purse. Normally, while she is in an appointment, the phone will automatically transfer directly to voicemail, but Alice has a code she can press to get through. The phone knows it's Alice calling, and uses a distinctive ring tone.
8. Vivien takes the call—Alice missed the bus in needs a pick up. Vivien calls her husband to see if he can do it. She gets his voicemail; he must be out of service range. She tells him she's with a client, and asks if he can get Alice. Five minutes later the phone makes a brief

Part 1

Part 2

87

A Persona narrative

Persona:	USDA Senior Manager Gatekeeper
Photo:	
Fictional name:	Matthew Johnson
Job title/ major responsibilities:	Program Staff Director, USDA

A Persona narrative (Cont.)

Demographics:	• 51 years old • Married • Father of three children • Grandfather of one child • Has a Ph.D. in Agricultural Economics.
Goals and tasks:	He is focused, goal-oriented within a strong leadership role. One of his concerns is maintaining quality across all output of programs. Spends his work time: • Requesting and reviewing research reports, • preparing memos and briefs for agency heads, and • supervising staff efforts in food safety and inspection.

A Persona narrative (Cont.)

Environment:	He is comfortable using a computer and refers to himself as an intermediate Internet user. He is connected via a T1 connection at work and dial-up at home. He uses email extensively and uses the web about 1.5 hours during his work day.
Quote:	"Can you get me that staff analysis by Tuesday?"

Persona developed by the U.S. Department of Agriculture's (USDA) Economic Research Service (ERS).

Adopted from [10]

90

Actividad: E10-3 Repaso de la técnica de Pesonas

References

- 1. Alan Cooper and Robert Reimann (2003). About Face 2.0: The Essentials of Interaction Design, Wiley
- 2. https://books.google.com.mx/books?id=ccGqdRkc5mkC&pg=PT235&lpg=PT235&dq=workflow+directed+graphs&source=bl&ots=Q1foLx_hfk&sig=gfbJUZgTGZ3-2UNGFHcZr2ESan4&hl=en&sa=X&redir_esc=y#v=onepage&q&f=false
- 3. http://www.smartdraw.com/workflow-diagram/examples/
- 4. http://hciresearch4.hcii.cs.cmu.edu/M-HCI/2009/Eaton/research/?page=cis
- 5. https://en.wikipedia.org/wiki/Floor_plan
- 6. http://ux.stackexchange.com/questions/81138/explain-behavioral-variables-in-interaction-design
- 7. http://asinthecity.com/2011/05/13/explaining-personas-used-in-ux-design-%E2%80%93-part-2/
- 8. http://www.cooper.com/journal/2002/11/getting_from_research_to_perso
- 9. http://ux.stackexchange.com/questions/81138/explain-behavioral-variables-in-interaction-design
- 10. http://www.usability.gov/how-to-and-tools/methods/personas.html
- 11. http://uxmag.com/articles/using-proto-personas-for-executive-alignment
- 12. Clara Balderas (2018). Tecnologías Adictivas, a talk.

92

APPENDIX

References

- 1. Vinod Narayan (2016). Prioritization Models in Agile, URL: https://www.youtube.com/watch?v=Dt_5KQg06_Q&feature=youtu.be, as of 3/9/18.
- 2. Agile Business Consortium (2018). What is DSDM?, URL: https://www.agilebusiness.org/what-is-dsdm, as of 3/9/18.
- 3. Frank Turley (2014). PRINCE2 2017- The MoSCoW Prioritisation Technique, URL: https://youtu.be/btZxWPu72Zw, as of 2/15/19.
- 4. Anurag Saksena (2013). “Agile Methodology Episode 1 -- Scrum Framework”, URL: https://youtu.be/mJRsSbWR3Mc