

Any questions from last week?

0 questions

0 upvotes

SIT773 - SOFTWARE REQUIREMENTS ANALYSIS AND MODELLING

- Lecture 9: Finalisation of Analysis
- Dr Hourieh Khalajzadeh (Hannah)
- School of Information Technology
- hkhalajzadeh@deakin.edu.au



Outline

- Revisiting the SDLC lifecycle
- Discuss each phase
- Customer sign off

```

mirror_mod = modifier_ob.
    # Set mirror object to mirror
    mirror_mod.mirror_object =
    # operation == "MIRROR_X":
    mirror_mod.use_x = True
    mirror_mod.use_y = False
    mirror_mod.use_z = False
    # operation == "MIRROR_Y":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
    # operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

```

```

# selection at the end -add
mirror_ob.select= 1
modifier_ob.select=1
context.scene.objects.active
    ("Selected" + str(modifier_ob.
mirror_ob.select = 0
    = bpy.context.selected_object
data.objects[one.name].select
print("please select exactly

```

--- OPERATOR CLASSES ---

```

types.Operator):
    # X mirror to the selected
    object.mirror_mirror_x"
    mirror X"

```

```

context):
    context.active_object is not





```


The background of the slide features a 3D rendering of interlocking puzzle pieces. Most pieces are light gray, but one piece in the upper right quadrant is a vibrant red, standing out prominently. The lighting creates soft shadows, giving the pieces a three-dimensional appearance.

Revisiting the SDLC lifecycle

Software Development Life Cycle (SDLC)

Week 1

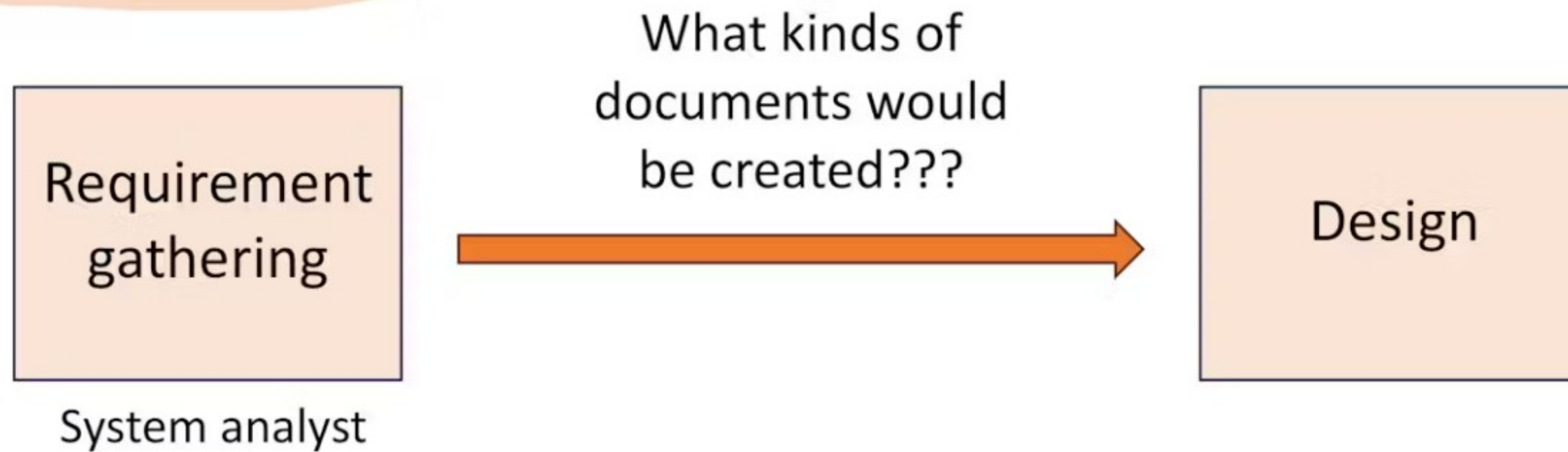
	Planning	find scope, cost, team, risks etc related to the project
	Analysis	Requirement analysts understand the user needs and the context of use
	Design	Explore potential solutions, and finalise one
	Build	development begins across the team
	Testing	tests the s/w for errors: automated or user testing
	Deploy	s/w is released to users in real-world setting
	Maintenance	vendor provides support for upgrades and bug fixing.

The background of the slide features a large puzzle piece in the center, with several other puzzle pieces visible around it. The puzzle pieces are white and grey, and the central piece is red. The background is dark grey.

Requirement analysis

Managing Documentation

Managing Documentation



What types of documents are generated by the system analyst for handover to the team?

use cases

Requirements Specification
Document
System Design
Document
Use Case
Diagram
test Cases and
Scenarios

use case
diagrams
ERD
Wireframe
project plan
requirement
analysis

User stories

Requirements Specification
Document
System Design
Document
Use Case
Diagram
test Cases and
Scenarios

SDLC plan

SRS Document
Project
completion report

mockups

What types of documents are generated by the system analyst for handover to the team?

Requirements
Documentation

performance, security,
UAT

What qualities should the requirement list possess for a successful handover?


clarity, completeness,,
consistency,

software system's vision
statement matched

Correct, clear,
prioritised

feasibility, traceability,
testability etc

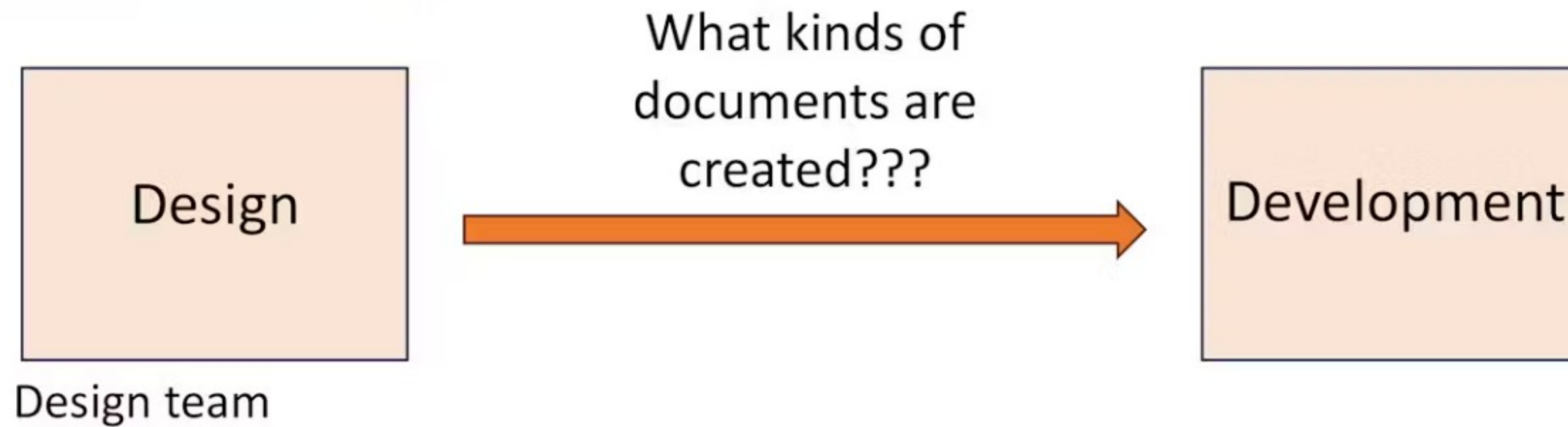
SMART

The background of the slide is a 3D rendering of interlocking puzzle pieces. Most pieces are light gray, but one piece in the upper right quadrant is a vibrant red. The lighting creates soft shadows, giving the pieces a sense of depth and volume.

Design phase

Managing Documentation

Managing Documentation



What types of documents are generated in the design phase?

WireframesArchitecture
Design

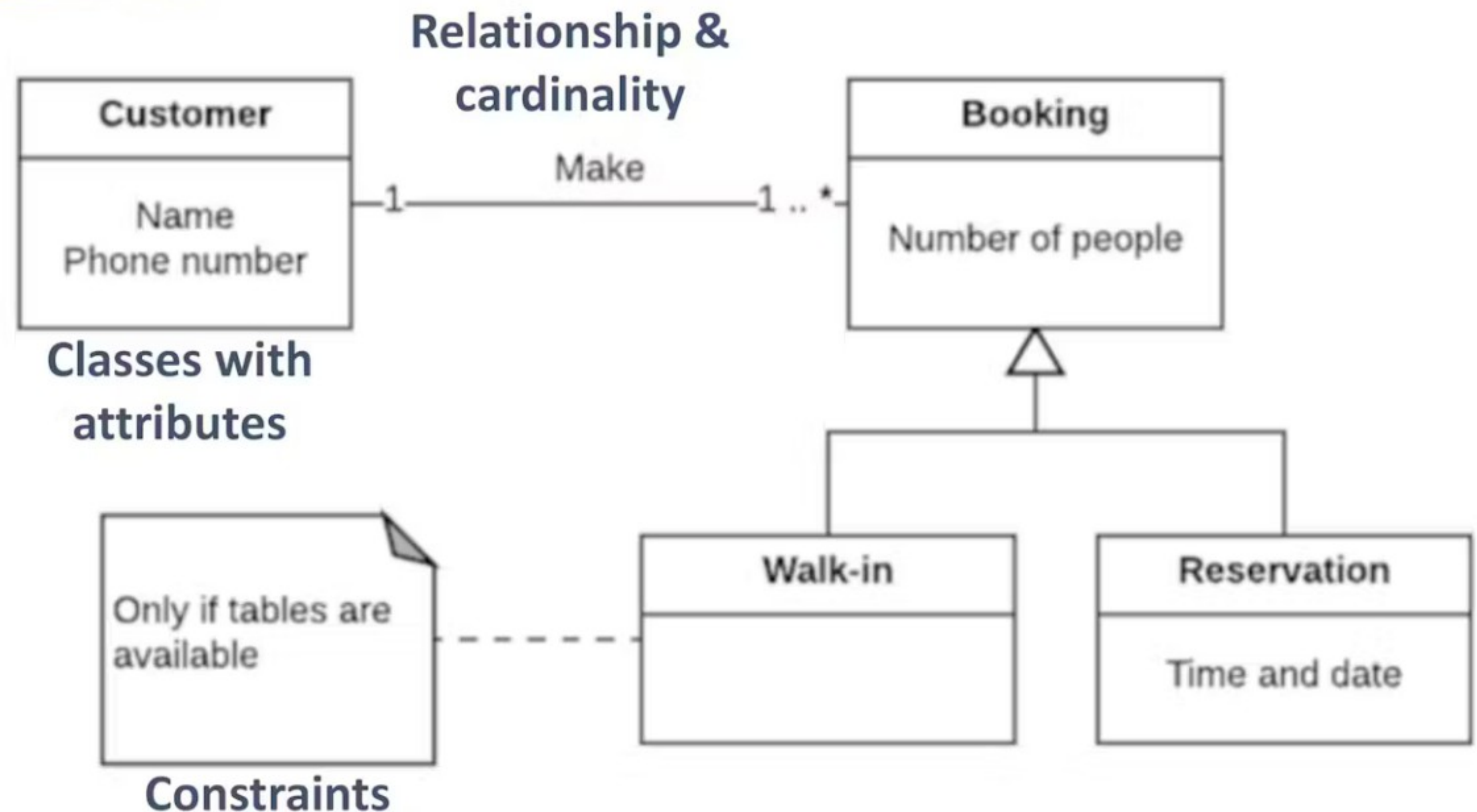
Domain model
diagrams

Implementation
Document

Test Cases or test
design

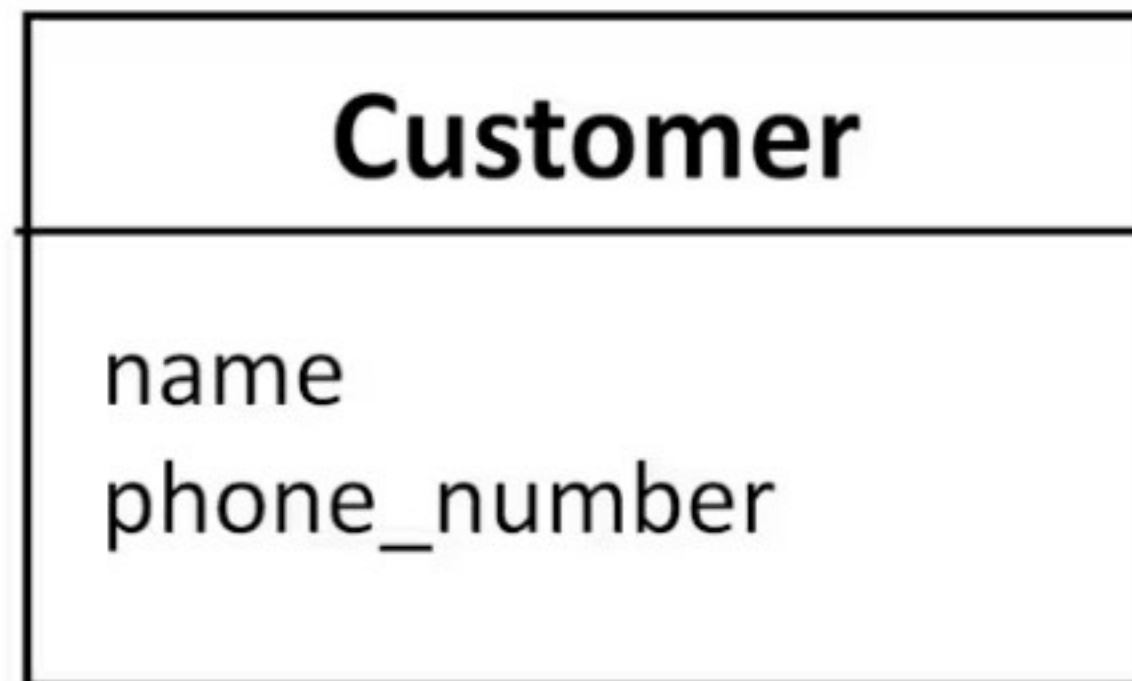
Domain Model (Week 2)

- Domain model captures all entities in the domain and how they relate to each other.
- These diagrams are used to communicate the problem to the project team.
- It's an iterative process to develop domain models.
- Use Lucidchart to create these models

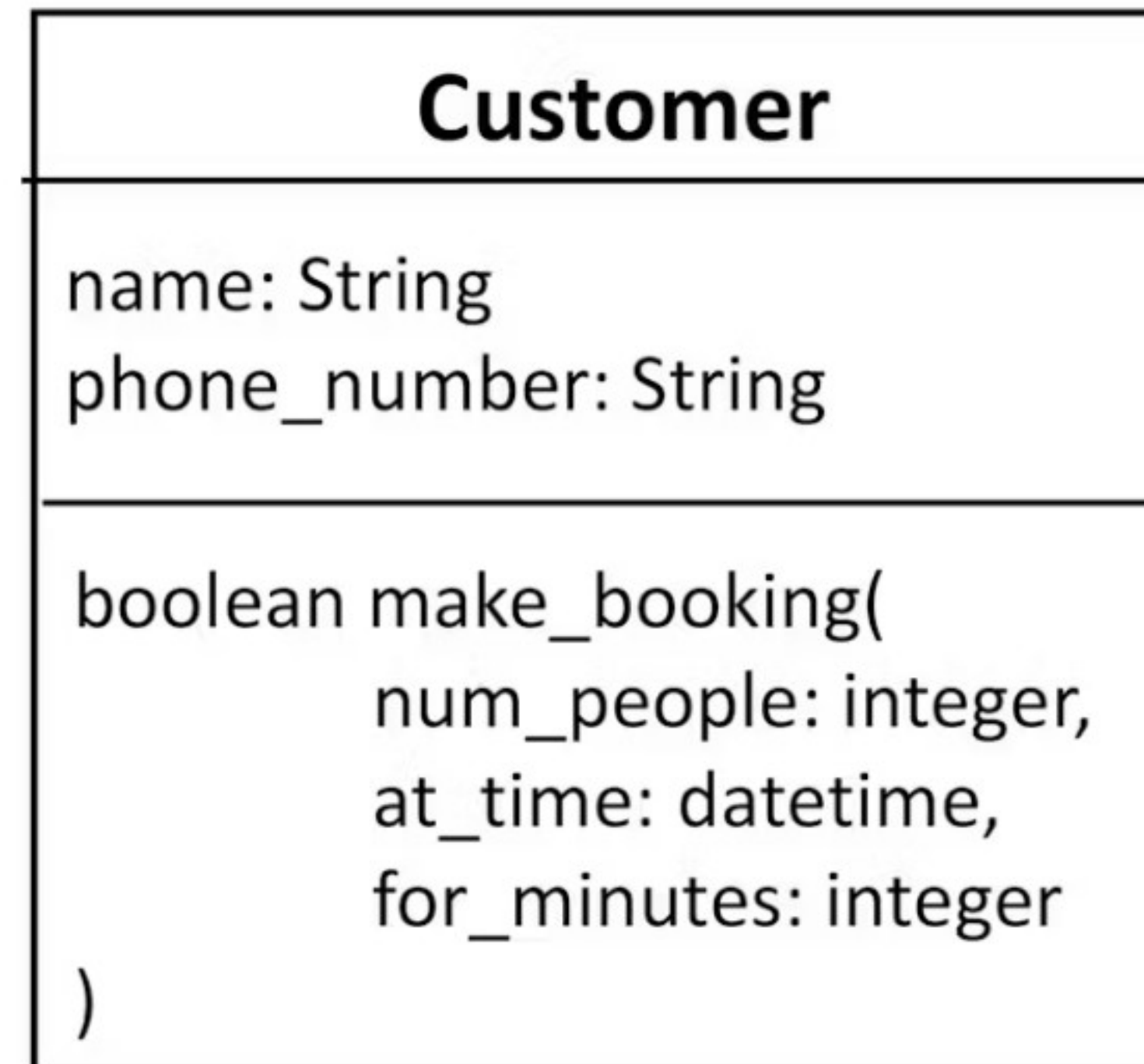


Design Class Diagram

- specify data types within each class
- add behaviours to classes
- add coordinating classes



Design Class
Diagram



Data types

Class
Behaviours

Design Class Diagram

Customer
name: String phone_number: String
boolean make_booking(num_people: integer, at_time: datetime, for_minutes: integer)

Data types

**Class
Behaviours**

Restaurant
tables: [Table] trading_hours: Schedule
Schedule get_trading_hours() Table get_tables()

Data types

**Class
Behaviours**

What are the issues with documentation?

Inconsistencies

Version control

Not having a concrete set of requirements from the beginning (specially if using Waterfall method)

hard to manage alot of documents

Not documenting changes

Incomplete or unclear requirements
Missed stakeholders
Lack of agreement

What strategies should be used to properly manage the documentation?

Version control, access management

central system to manage all documents
Version control

Chain of custody of the document should also be considered

A solid orange horizontal bar.

Development

Testing

What kind of testing is performed during the development process?

Unit testing

integration testing
unit testing

Functional testing

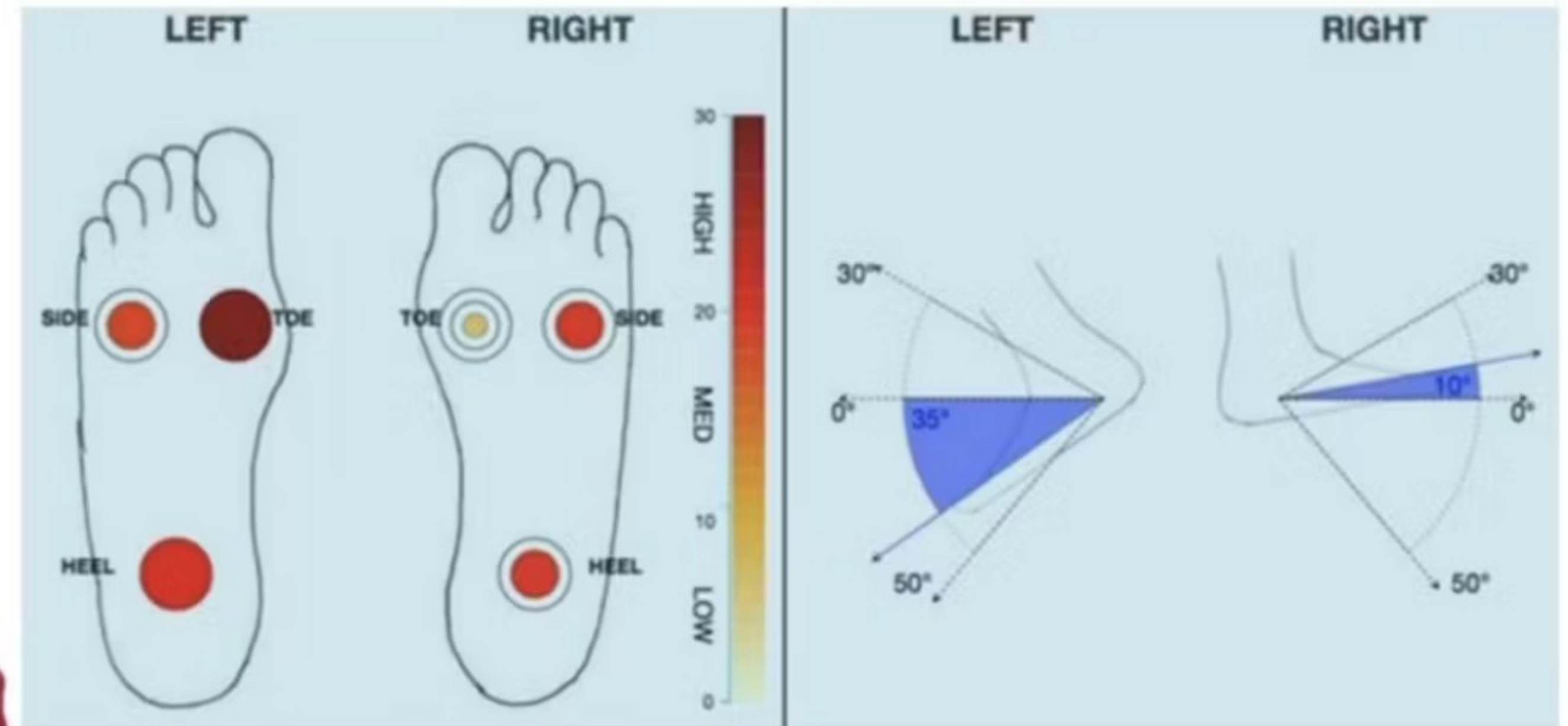
Regression testing?

Stress testing

performance testing

security testing,
compatibility

Can you guess what types of testing were performed during the SoPhy development?



Weight Distribution,
Foot Orientation

Range of Movement

The background of the slide is a 3D rendering of interlocking puzzle pieces. Most pieces are light gray, but one piece on the right side is a vibrant red. The lighting creates soft shadows, giving the pieces a sense of depth.

Testing

User Acceptance Testing

User acceptance testing

users should be
'outsiders'

Lab testing with
target users

Alpha and beta
release to
specific set of
users

Methods: screen
recording,
interviews,
observations
etc.

What issues can be uncovered through the user acceptance testing?

Functionalities that don't align with initial user requirements

requirement gaps

Requirement process not aligned with requirements

usability issues, performance issues, unexpected behaviours

What are the challenges of conducting user acceptance testing?

Accessibility

limited users

Not having access for
the users

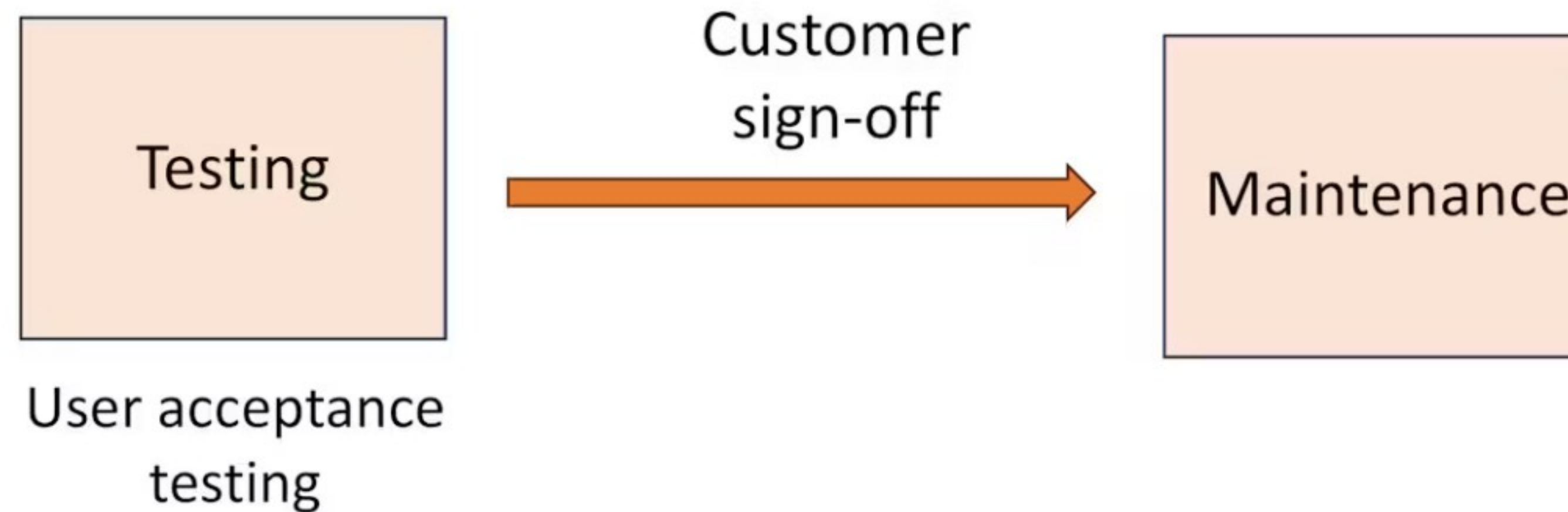
managing feedback



Customer sign-off

- The customer acknowledges that the work was delivered by the organisation and that the organisation should be paid.

Customer Sign-off



What could go wrong with sign-off?

Clear
CommunicationThorough
Review ProcessInvolve All
Key StakeholdersCreate a
Change Management Plan

agree on acceptance
criteria, regular updates

What are the strategies to manage the potential sign-off issues?

Clear contract on deliverables

Create a Change Management Plan

Good communication with the customer up to the sign off, including checking on expectations and timeframes.

Prioritize Requirements
Involve All Key Stakeholders
Clear Communication
Thorough Review Process

Any questions?

0 questions
0 upvotes



Let us look at the tasks now.

Thank you!

I will see you all next week.

