Mentimeter

IN5140

Smart Processes and Agile Methods in Software Engineering

Group Session 8

Software process modelling (BPMN)

Mentimeter

Revision

BPMN - Business Process Modelling Notation

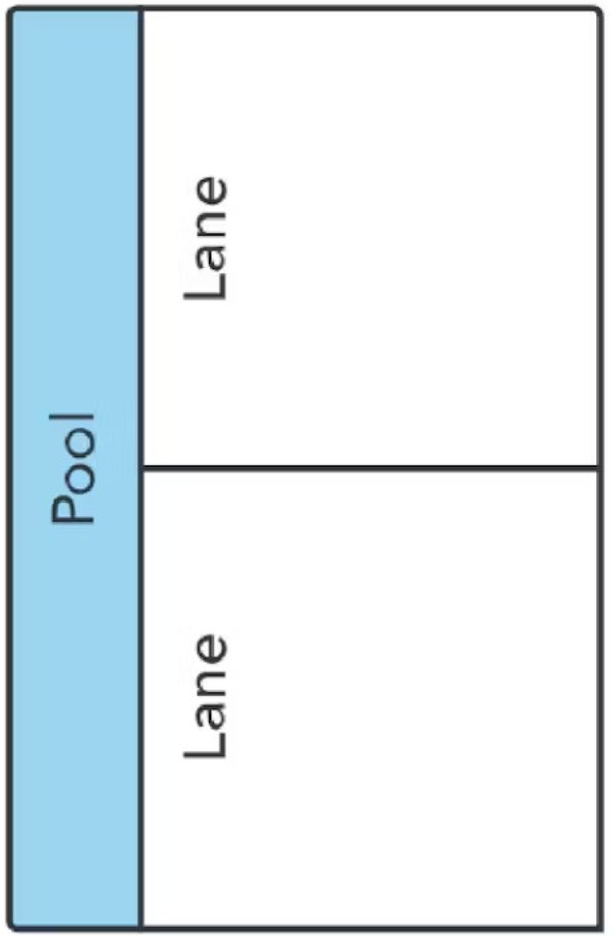
Mentimeter

BPMN - Business Process Modeling Notation

BPMN is a **flow chart method**, used for Business Process Management. It models the steps of a **planned business process** from end to end, **visually depicting** a detailed sequence of **business activities and information flows** needed to complete a process.

Mentimeter

Pools and Swimlanes



- **A pool** represents major participants in a process. A different pool may be in a different company or department but still involved in the process.
- **Swimlanes** within a pool show the activities and flow for a certain role or participant, defining who is accountable for what parts of the process.

(Taken from Luci a

Mentimeter

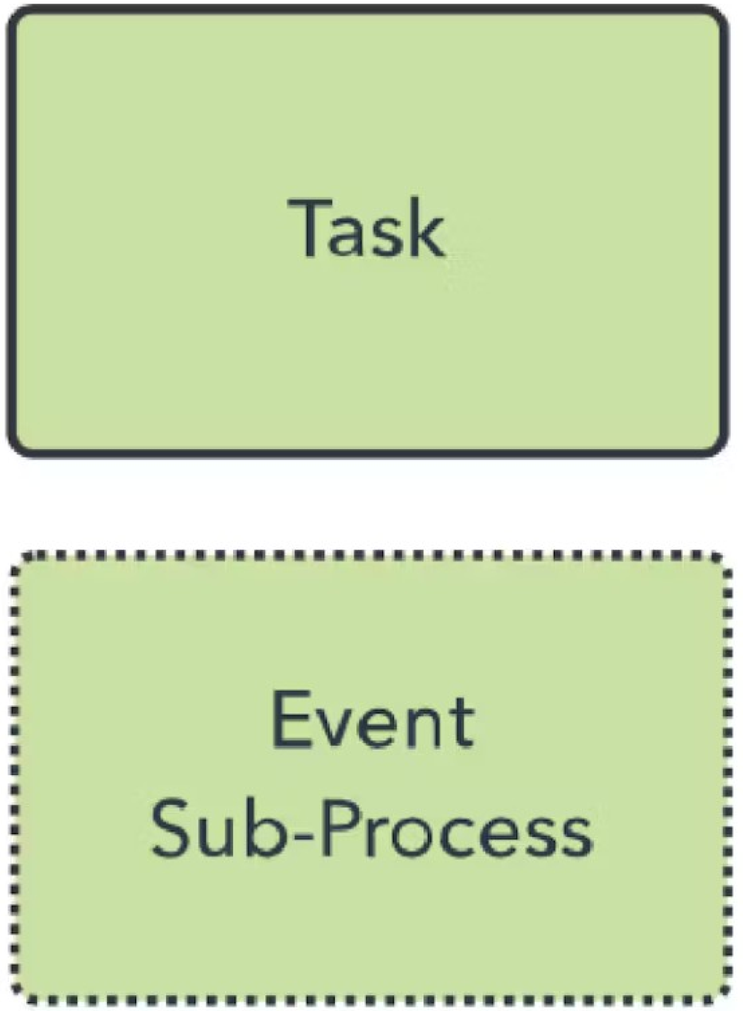
Events

- A trigger that starts, modifies or completes a process.
- They are shown by circles containing other symbols based on event type.

Mentimeter

Activities

- A particular activity or task performed by a person or system. It's shown by a rectangle with rounded corners.
- They can become more detailed with for example sub-processes.



Mentimeter

Gateways

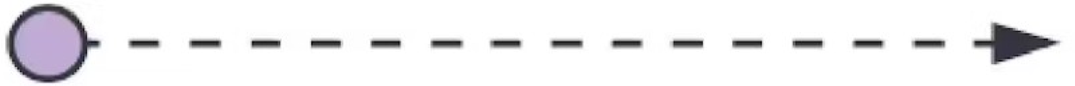
Exclusive | Event based | Parallel | Inclusive | Exclusive event based | Complex | Parallel event based

- Decision point that can adjust the path based on conditions or events. They are shown as diamonds.
- Gateways can also be an empty diamond

Mentimeter

Flow

- **Sequence flow:** shows the order of activities to be performed



- **Message flow:** depicts messages that flow across “pools,” or organization boundaries such as departments. It shouldn’t connect events or activities within a pool.

Mentimeter

BPMN Tutorial Video

Mentimeter

BPMN - tools

- Diagrams.net *(previously Draw.io)*
  - https://www.diagrams.net/
- **Lucidchart**
  - https://www.lucidchart.com/
- yEd Graph Editor
  - https://www.yworks.com/downloads#yEd
- Yaoqiang BPMN Editor - Open Source
  - https://sourceforge.net/projects/bpmn/
- Signavio
  - https://www.signavio.com/ (free 30-day trial)
- Pen and paper

Mentimeter

How to make a BPMN:

- **Skim through** the textual description
  - Identify **pools**: which companies or **organizations** are involved.
  - Identify **swimlanes**: which **roles** are present in each pool.
    - Identify the **tasks** occurring in each swimlane
    - What **dependencies** can you identify
- Model the tasks with metadata
- Check that you are still answering the exercise and accounted for all major occurrences of the process.

Mentimeter

You do not need to make the BPMN diagram too extensive or complicated. Most importantly, it should be readable and understandable!

Key elements to check:

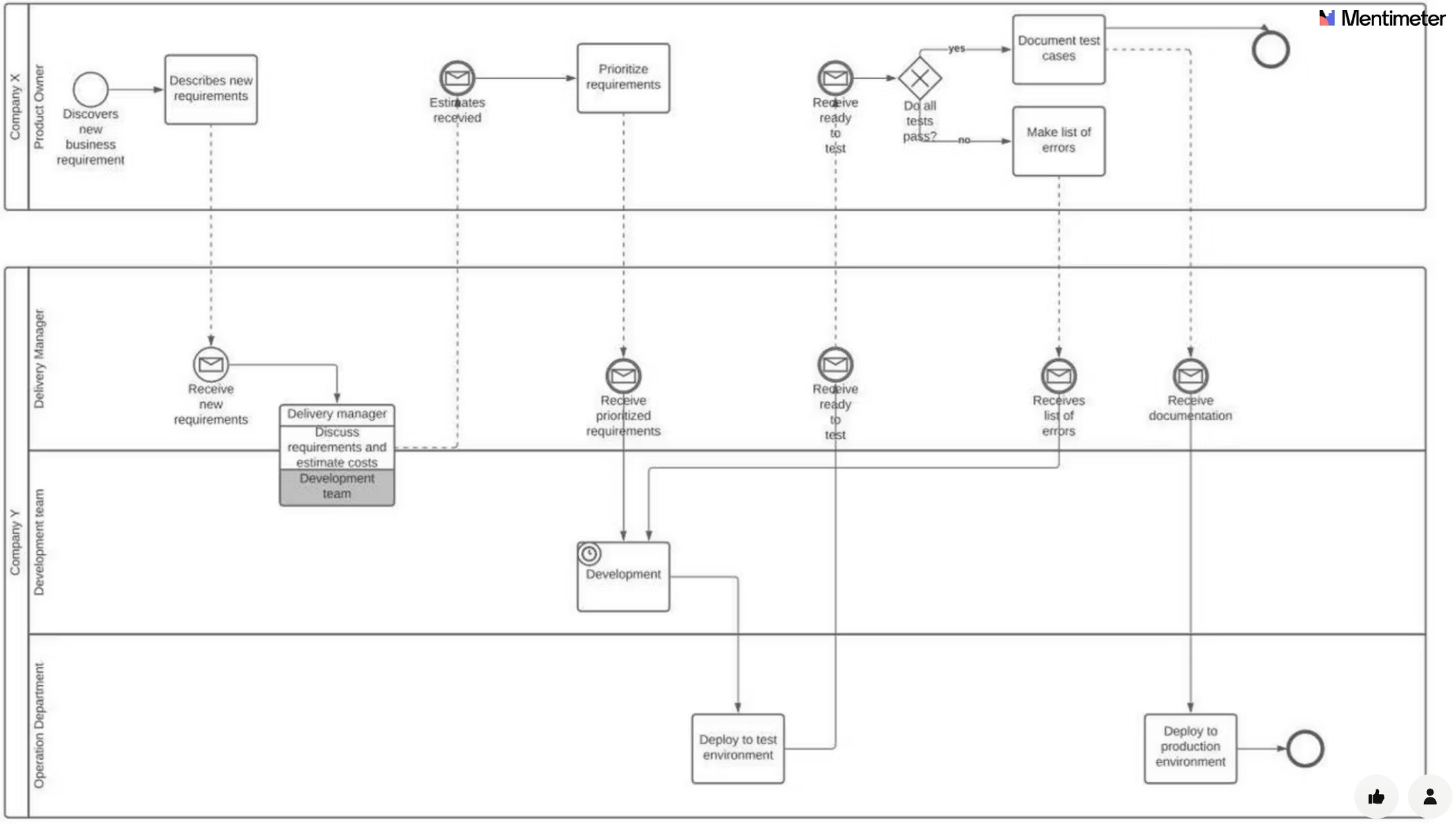
- Correct lanes and pools
- Activities and gateways
- Start and stop event
- Sequence flow
- Collaborations

Mentimeter

Case Description 1 - From Lecture

Background:
Company X uses a web application to support its business. The web application was developed and is currently maintained by Company Y. The two companies work together on improvements and further development; X identifies needs for new functionality and orders this from Y, and Y regularly delivers new versions of the web application with the requested functionality to X.

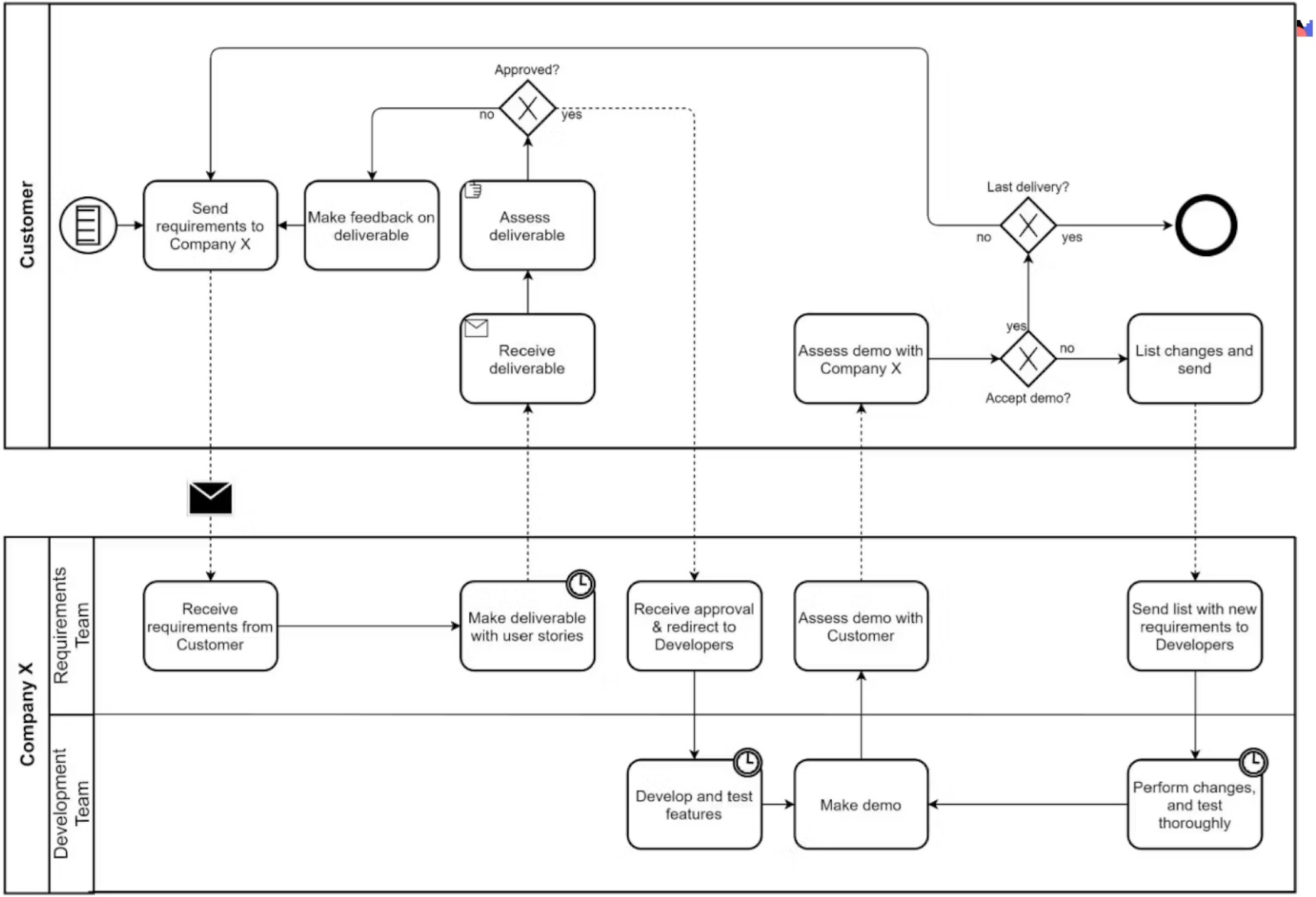
The process of developing new functionality:
Company X has a Product owner for the web application who is responsible for identifying new business opportunities, for describing new requirements and for testing. The Product owner regularly hands over a requirement specification to the Delivery manager at Company Y. The Delivery manager is responsible for the successful delivery of new versions of the application. The Delivery manager cooperates with a team of developers in company Y. They discuss the requirements from company X and estimate the tasks. Then the Delivery manager informs the Product owner about the estimates and costs. Based on this information, the Product owner decides which requirements should be prioritized for the new version of the system and communicates this back to the Delivery manager. Then the developers start development. When the new features have been developed, they are deployed to the test environment by the Operation department at Company Y. The Delivery manager requests the Product owner to test that implemented features correspond to the specified requirements. If some of the features do not pass the test, the Product owner makes a list of errors and informs the Delivery manager, who in turn informs the developers of the necessary changes and error corrections. When all tests are passed, the test cases are documented and sent to the Delivery manager, who requests the Operation department to deploy the new version to the production environment. The Operation department deploys the new version.


Company X
Product Owner
Discovers new business requirement
Describes new requirements
Estimates recevied
Prioritize requirements
Receive ready to test
Do all tests pass?
yes
no
Document test cases
Make list of errors
Company Y
Delivery Manager
Receive new requirements
Delivery manager
Discuss requirements and estimate costs
Development team
Receive prioritized requirements
Receive ready to test
Receives list of errors
Receive documentation
Development team
Development
Operation Department
Deploy to test environment
Deploy to production environment

Mentimeter

Case Description 2 - Exam 2017

A software company in Norway (Company X) is developing an administrative system for the Norwegian public sector (the customer). The current development process is as follows: Company X receives a fixed list of requirements from the customer. Company X then first rewrites the requirements into user stories and then groups them into a set of deliverables. For example, requirements that deal with profile administration are grouped into a deliverable entitled “User Profile”. After defining a deliverable, Company X sends it to the customer representatives for approval. Once the customer approves the deliverable, Company X sends the deliverable to a development team in Ireland that is owned by company X. The Irish team codes the requested functionality and sends back a demo to Company X. Company X and the customer assess in collaboration the work of the Irish development team. If the deliverable (demo) is accepted by the customer, Company X starts to plan a new deliverable. However, if the deliverable (demo) is rejected, Company X must ensure the necessary changes, including fixing bugs, performed by the Irish development team. After implementation, they return a new demo for approval. This cycle is repeated until the customer finally approves the deliverable.

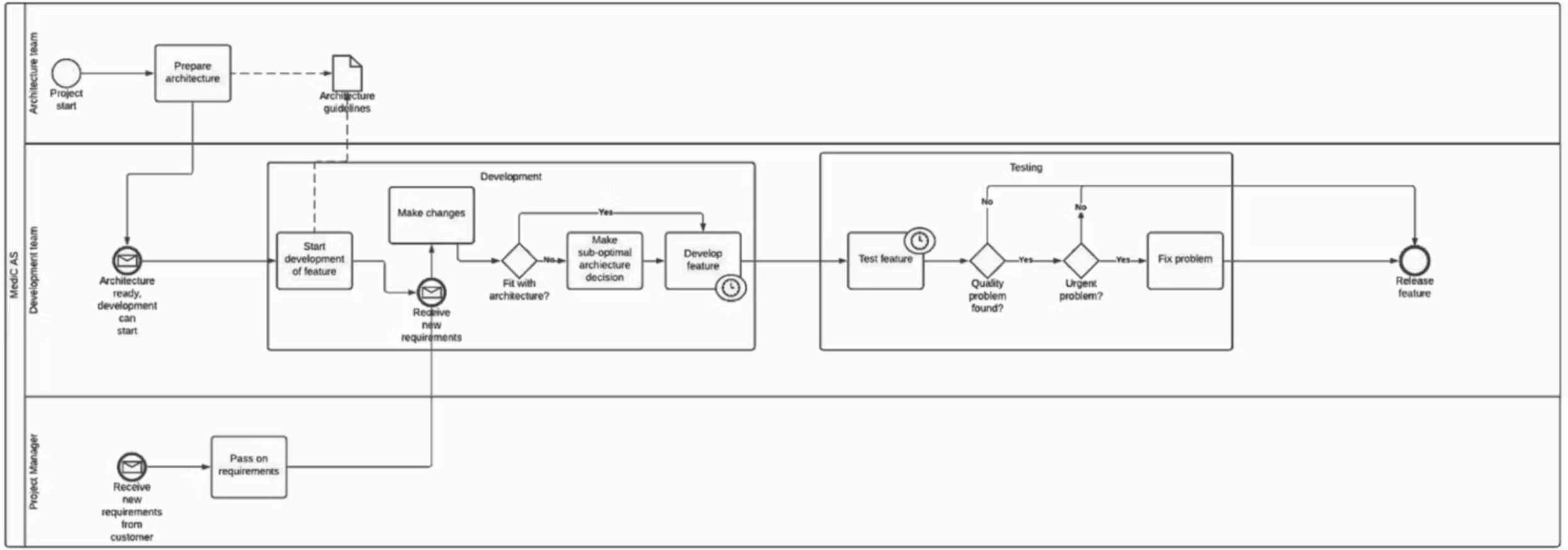

Customer
Approved?
no
yes
Send requirements to Company X
Make feedback on deliverable
Assess deliverable
Receive deliverable
Last delivery?
no
yes
Assess demo with Company X
yes
no
Accept demo?
List changes and send
Company X
Requirements Team
Development Team
Receive requirements from Customer
Make deliverable with user stories
Receive approval & redirect to Developers
Assess demo with Customer
Send list with new requirements to Developers
Develop and test features
Make demo
Perform changes, and test thoroughly

Mentimeter

Case Description 3 - Exam 2020

MediC experiences some challenges when scaling up. Architects are part of a special architecture team and prepare the whole architecture upfront without adjusting the architecture later. During development, developers find issues that were not considered by the architects, and they need to implement features in a different way from what is described in the initial architecture models. The architects become upset about developers not following the architecture, and the developers do not trust the architects' guidelines. When the teams need to take care of the new information found during implementation, they take sub-optimal architecture decisions, because they do not involve the architects. The project manager is in dialogue with the customer and passes new requirements to the development teams continuously. The development teams implement features, and the code is reviewed and tested just before release. However, it is difficult and time consuming to fix all the quality problems that are found, only the very urgent ones are fixed. The release is delivered anyway. During the next iterations, there is no budget to fix previously found quality problems, so they stay in the system. The project manager expects the teams to proceed to deliver new features fast, but now the teams seem to get stuck fixing bugs and changing all parts of the software for any small new changes that are requested.

Mentimeter


MediC AS
Architecture team
Development team
Project Manager
Project start
Prepare architecture
Architecture guidelines
Architecture ready, development can start
Development
Start development of feature
Make changes
Receive new requirements
Fit with architecture?
No
Yes
Make sub-optimal archiecture decision
Develop feature
Testing
Test feature
Quality problem found?
No
Yes
Urgent problem?
No
Yes
Fix problem
Release feature
Receive new requirements from customer
Pass on requirements

Mentimeter

Next Week

Managing Technical Debt

Guest lecture from Visma

Mentimeter

Feedback

Mentimeter

Feedback & Requests