

# IOB22

## Digital Product Development

# Hello, World!

[dpd-io@tudelft.nl](mailto:dpd-io@tudelft.nl)

# House Keeping - Hybrid



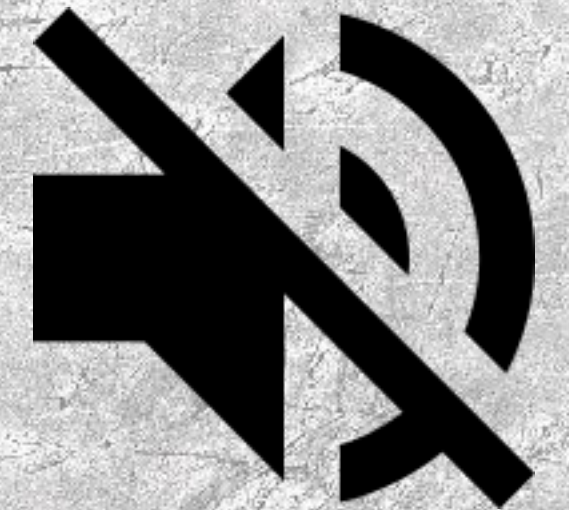
MS Teams events are recorded and made available



All interactions via MS Teams Cam & Mic



When not interacting



In the room speaker off at all time

# What's the plan for this session?

- You & Us
- What to learn & how?
- Break
- Interactive session: DPD Canvas

# A Warm Welcome from the Whole Team!



# All on Discourse!

- Everyone is on Discourse
- You can prompt the attention of anyone in the team with personal tags such as @jacky



**IOB22 Community**



**Course Feedback**



**Programming Assignments**



**Exercises**

# Answer on Discourse!

IOB22 > Getting Started



## About you

How do you rate the importance of digital knowledge and skills for designers?

What is your programming experience?

How do you feel about this course?

# Learning Objectives

Digital Product Development  
IOB22

## What am I Going to Learn?

 Explain

 Specify

 Develop

 Analyse

# What's the Plan?

*8 Weeks, 8 modules*

technical requirements

**1** intro

Digital  
Product

Development

Canvas

**2**

infrastructure

**3**

network

**4**

data

**5**

software

**6**

business

**7**

development

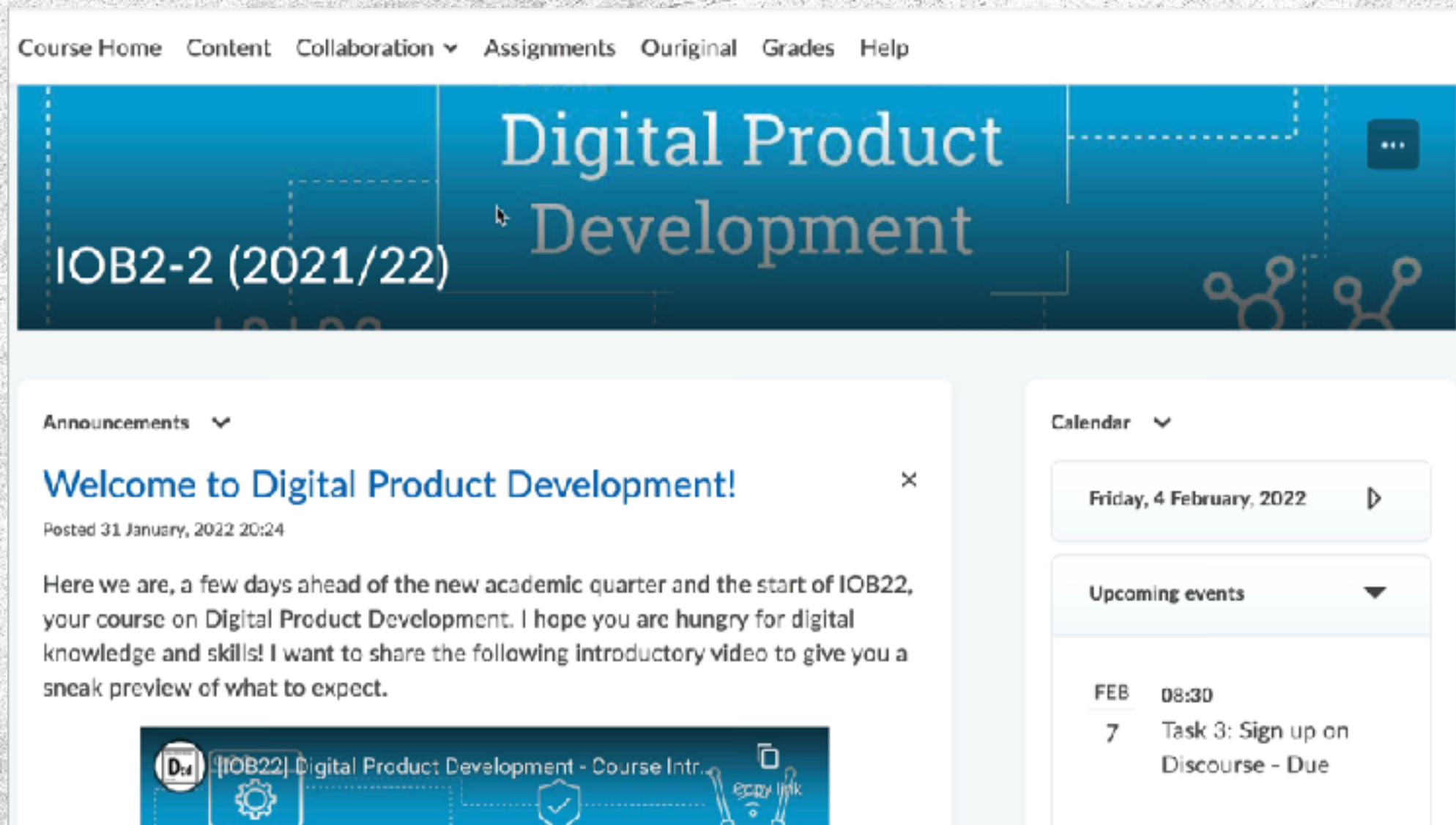
**8**

responsibility

organisation, process and role



# Weekly Rhythm



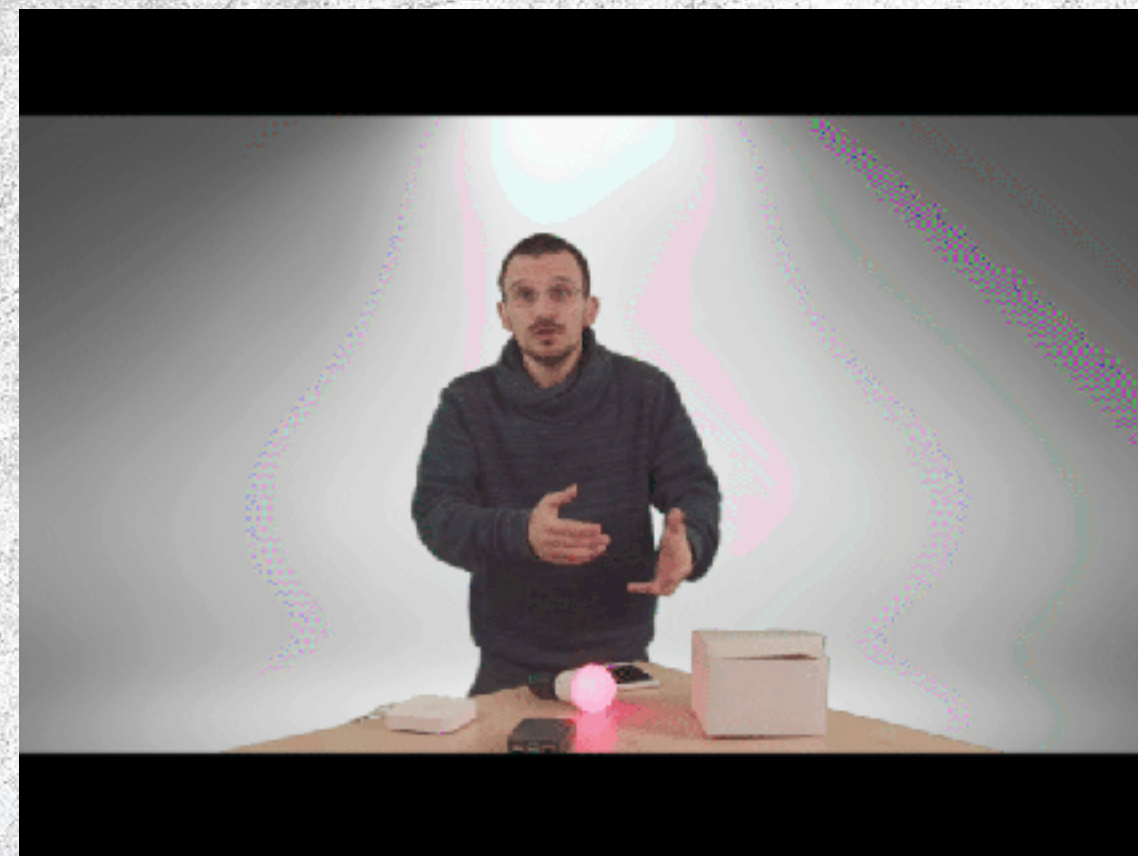
**Brightspace > Activities**  
**Weekly suggested list of task**

Modules	Weeks
Introduction	Feb 7
Infrastructure	Feb 14
Network	Feb 21
Data	Mar 28
Software	Mar 7
Business	Mar 14
Development	Mar 21
Responsibility	Mar 28

x8  
Q3.1-8

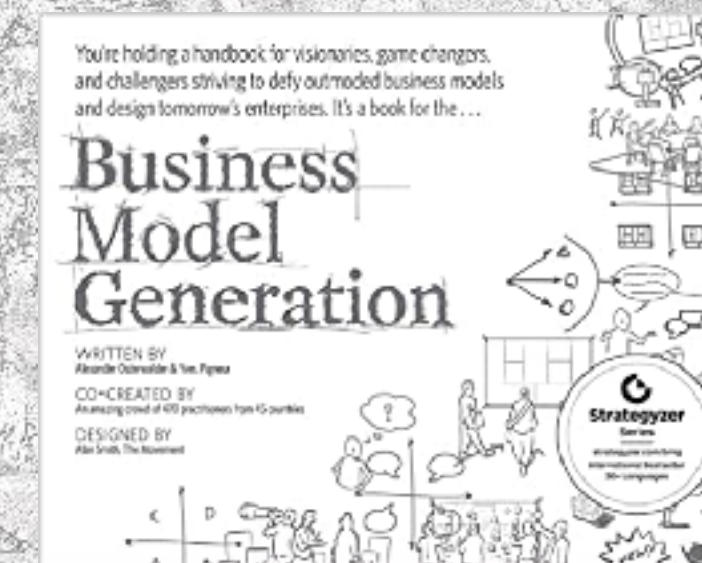
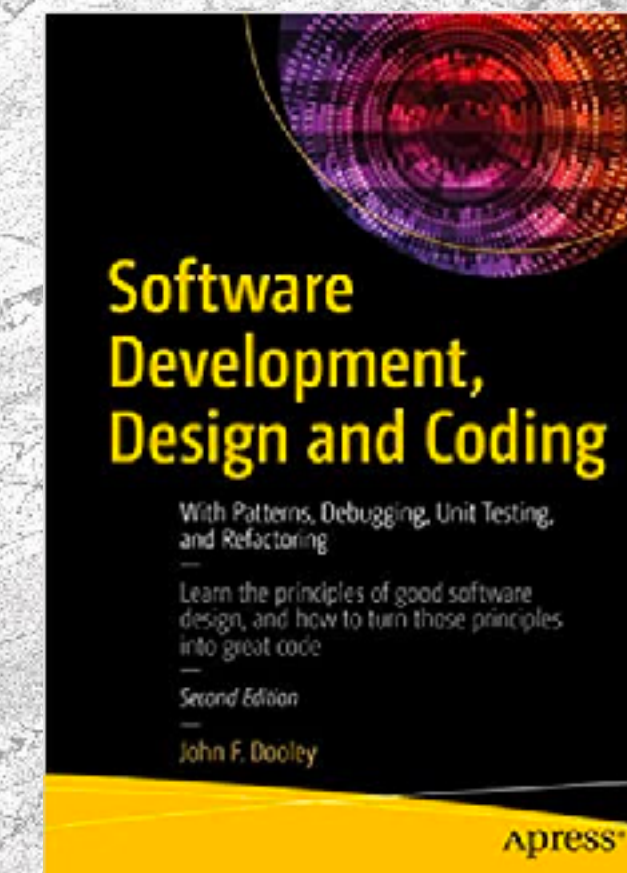
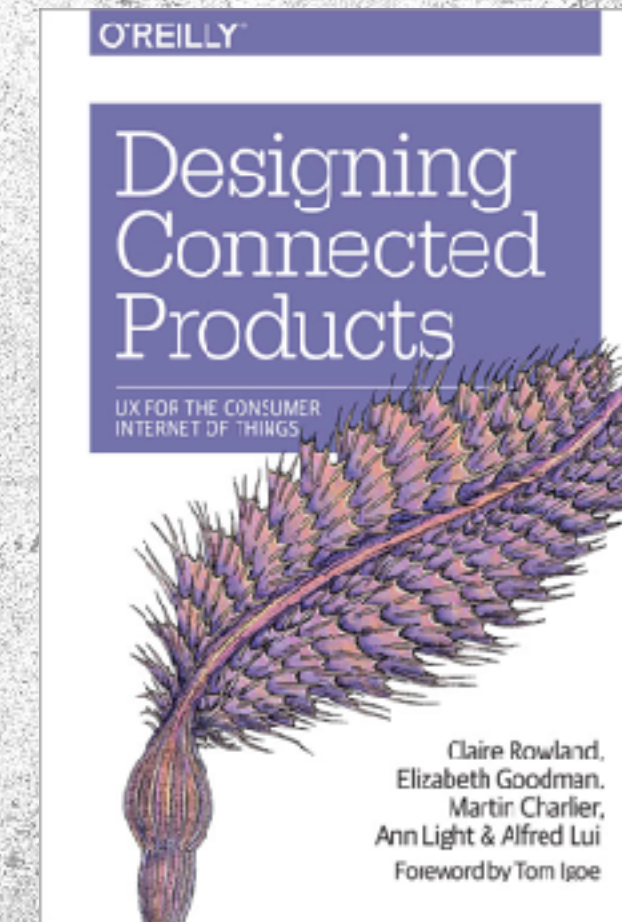


# Self-Study Material



Short Videos

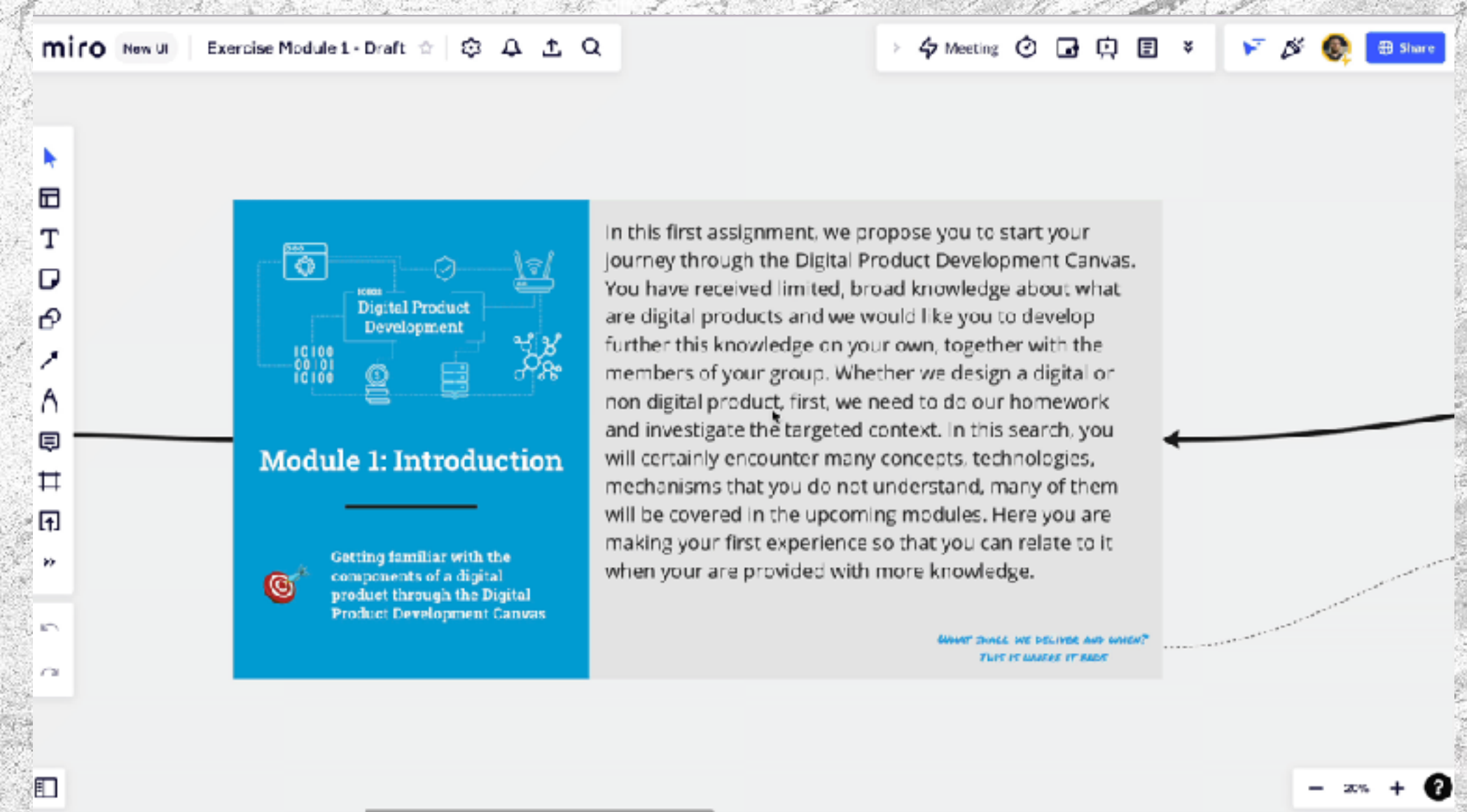
**Brightspace > Material**  
**All material per module**



Book chapters

# Group Exercise

- **Monday Mornings 10:45 (2hrs)**
  - Get together with your group in studio 1-14 or group MS Team channel
  - Connect to Miro and create a new board with the exercise template
  - Complete the exercise
  - Post your conclusion, reflections and question on Discourse



**We give primarily feedback on your conclusion, reflection and questions posted on Discourse.**

# Python Programming Assignments

Self-study, 4hrs a week

5 assignments in total

Support on Discourse

Wednesdays 13:45

Weekly programming sandpit



Code @ IDE
Search Code @ IDE

---

- Home
- Computational Thinking ^
- Environment
- 01 Calculator v
- 02 Vending Machine v
- 03 eReader v
- 04 Generative Art v
- 05 COVID Dashboard v
- Data-Centric Design v
- Practice v
- Prototyping v
- Troubleshooting
- Key Concepts

## From Design to Computational Thinking with Python

TABLE OF CONTENTS

- 1 [Introduction](#)
- 2 [Python Programming Assignments](#)
- 3 [What's next?](#)

---

### Introduction

Welcome to this practical introduction to Computational Thinking for designers. As Industrial Design Engineers, you continuously train yourself to master Design Thinking. Through this process, you **empathise, define, ideate, prototype and test**. As digital technology becomes ubiquitous, it impacts your design solutions and design process.

Your products embed or rely on computers to realise some of their functionalities. Your prototype involves computers to test and analyse the feasibility of your solutions. Your data combines qualitative and quantitative material to understand the challenges to address appropriately. For each of these tasks, you need a proper understanding of how computers manipulate information and how you can teach computers what you want them to do.

Along with your Design Thinking, this series of Python programming assignments aims to get you acquainted with another complementary approach: Computational Thinking. Computational Thinking relies on four steps. First, we **decompose the problem** into smaller parts, breaking it down to identify precisely each component of the problem to solve. Then, we **look for patterns**, similarities that we can tackle the same way. Third, it leads us to **elaborate components** for our Design that we can reuse for a whole category of problems (generalisation) that we can reuse without looking inside (abstraction). Finally, we **design an algorithm**, instructions that tell the computer what to do.

By the end of this series of Python programming assignments, you should have the confidence to use the Computational Thinking approach to teach computers simple tasks to perform. In addition, you should be able to break down simple problems into plain English instructions. Finally, you should be able to autonomously search the Internet for the Python syntax that is not yet in your toolbox.

**Disclaimer!** While we believe that industrial designers should get acquainted with Computational Thinking in a way that fits their discipline, we do not pretend we have the solution yet. We welcome your comments and suggestion for improvement. Reach out to Jacky (J.Bourgeois@tudelft.nl)

This site uses [Just the Docs](#), a documentation theme for Jekyll.

# Feedback Session

Interactive panel discussions

Reaction to your work on  
Discourse

Tips, Tops, Q&A, Polls

Examples of exam questions

*Hybrid: 65 students are randomly selected  
each week to attend in Joost van der Grinten*



# How do I get Graded?

- Online (Möbius), open-book exam on **April 13 (Wednesday, week 10)**
- Multiple-choice questions
- Fill in the blanks questions
- Programming questions

# No Other Grade?

- No other grade.
- All activities are formatives.
- The **teaching team** provide the structure
- **You** are in charge of your learning
- Ask questions and try out
- The more you engage in exercises and discussions, the more you gain feedback

# Attendance check

- Only today for closing the student list
- Sign the student list in the studios
- Add an 'online' note for your group member attending via the MS Teams group channel.
- Group entirely online: give me a nudge on MS Teams so that I can drop-in



# Contact

- Discourse for all content related matter
- For personal matter: [dpd-io@tudelft.nl](mailto:dpd-io@tudelft.nl)

# Discourse Moderators / Python Support

@iantiemann



@Floris\_de\_Groot



@SepehrTA

*Pssst ... they took this course last year,  
Ask them for tips!*

# What's next?

@sjoerdvandommelen

## Interactive Session

## Discovering the Digital Product Development Canvas



@jacky

# Break

Get some fresh  
air!

10 minute break

