IN5320 - Development in Platform Ecosystems

Lecture 1: *Introduction*

20th of August 2018

Department of Informatics, University of Oslo
Magnus Li - magl@ifi.uio.no
Olav Poppe - olavpo@ifi.uio.no

Today’s lecture

1. Theoretical introduction
2. Practical introduction
3. Lectures
4. Assignments
5. Group work
6. Final Exam

Lecturers

Olav Poppe

Doctoral Research Fellow
olavpo@ifi.uio.no

Magnus Li

Doctoral Research Fellow
magl@ifi.uio.no

Group teachers

Tin Anh Nguyen

Master student
PSE research group

Magnus Boing Nordin

Master student
IS research group

Andrei Eismont

Master student
N/A research group

Ole Marius Haanæs

Master student
DESIGN research group

Kristoffer Aune

Master student
DESIGN research group

Course page

< IN5320 - Development in Platform Ecosystems

Semester page for IN5320 - Autumn 2018

Schedule >

Examination: Time and place >

Syllabus/achievement requirements >

Messages

New message

First lecture on Monday 20.08! Edit

Hi!

Note that the first lecture will be held already on Monday 20.08 at 14:15!

We're looking forward to seeing you all there!

Contact

Department of Informatics

Teachers

- Magnus Li
- Olav Poppe

Group Teachers

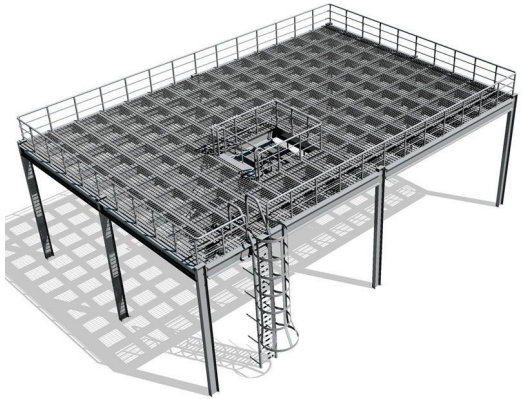
- Andrei Eismont
- Ole Marius Haanæs
- Tin Anh Nguyen
- Magnus B. Nordin
- Kristoffer Solheim

Course materials

- Course overview (draft)

Platform Ecosystems

What is a platform?



What is a platform?

VISA

coinbase

FINN

iOS

What is a platform?

“A raised level surface on which people or things can stand, usually a discrete ***structure intended for a particular activity or operation****”*

Oxford english dictionary

What is a platform?

"A raised level surface on which people or things can stand, usually a discrete ***structure intended for a particular activity or operation****"*

What is a platform?

Baldwin and Woodard (2008): Platform **architectures** refer to systems that are partitioned into:

1) A set of stable components
2) A set of complementary components that vary

Between these are interfaces that enable interaction. These are part of the platform and should be stable over time.

"The low-variety components constitute the platform. They are the long-lived elements of the system and thus imply or explicitly establish the system's interfaces, the rules governing the interactions of the different parts" (Baldwin and Woodard, 2008, p 19)

What is a platform?

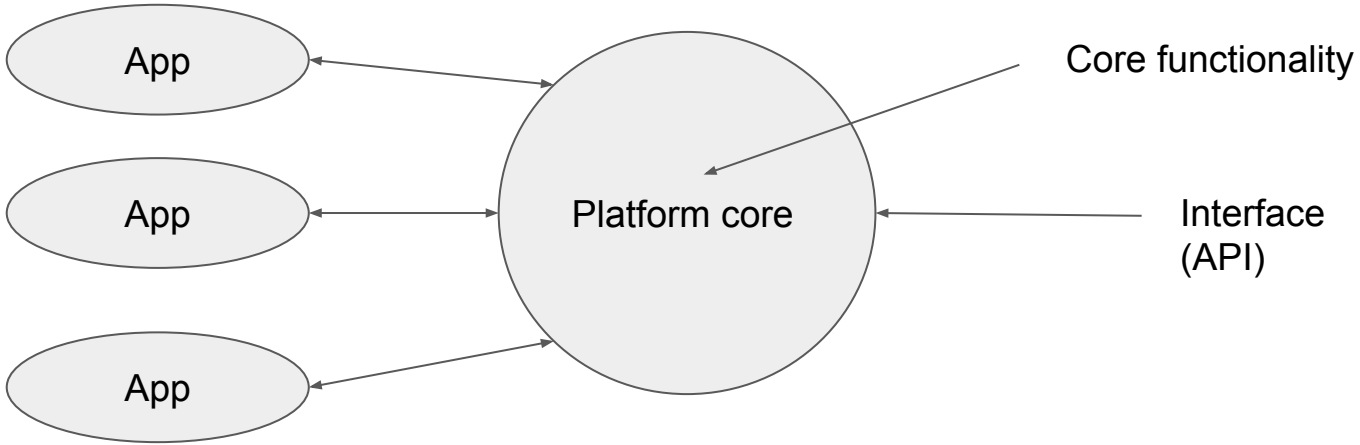
Baldwin and Woodard (2008): Platform **architectures** refer to systems that are partitioned into:

1) A set of stable components
2) A set of complementary components that vary

What is a platform?

"A software platform is a software-based product or service that serves as a foundation on which outside parties can build complementary products or services" - Tiwana 2013 p5

- Provides core functionality which is extendable
- Entails interfaces that allows third parties to develop *apps* that extend the functionality of the platform

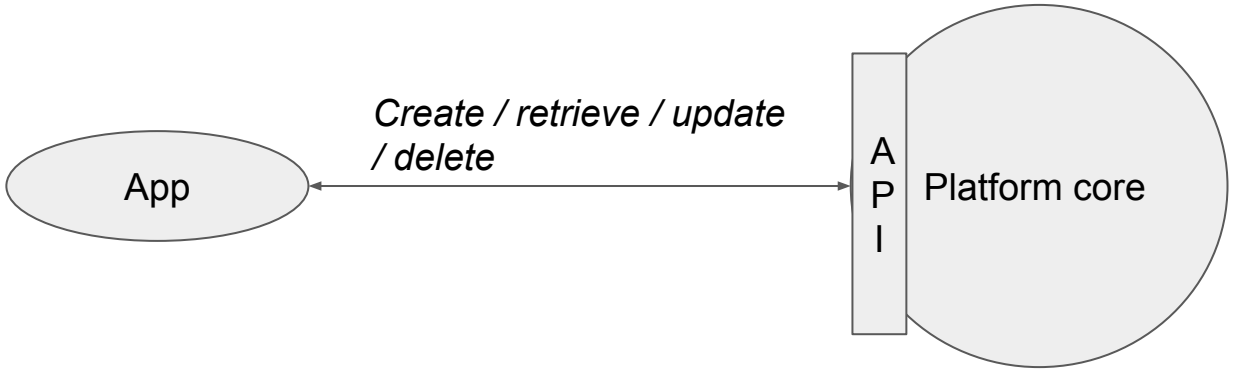


API

- Application programming interfaces (APIs) allows us to access, update, create and delete resources in the platform core.

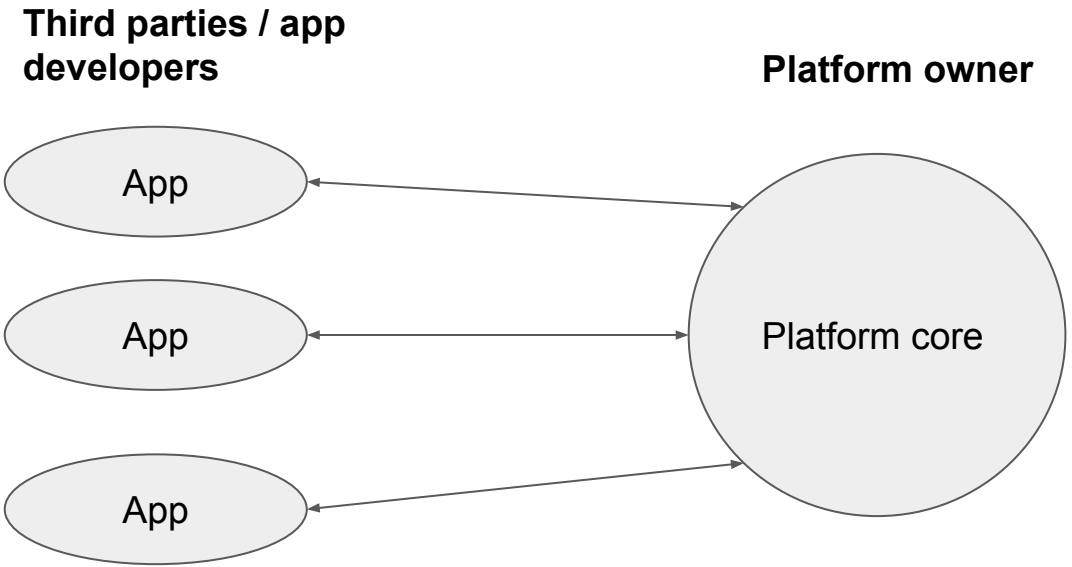
CRUD operations:

- Create
- Retrieve
- Update
- Delete



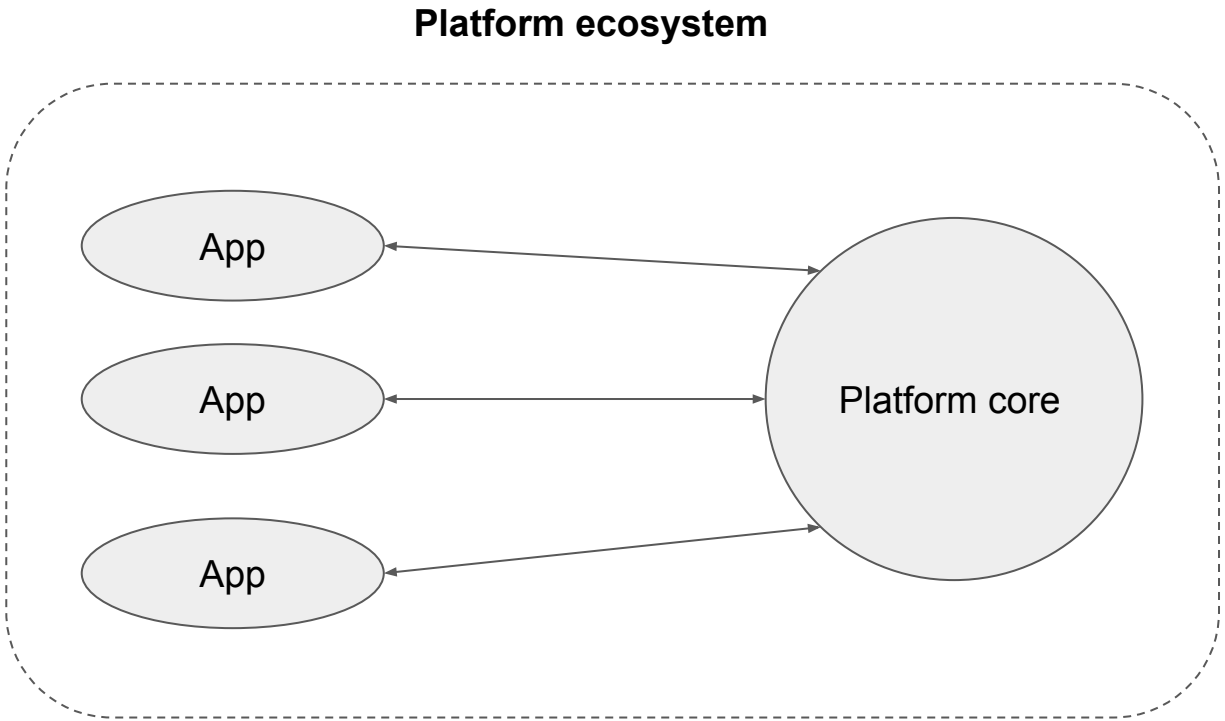
What is a platform?

- Someone controls the platform core, often referred to as the platform owner.
- This can be one or several firms, and may be proprietary for-profit, or open source.
- The apps are often developed by third parties, that is, other firms or actors.

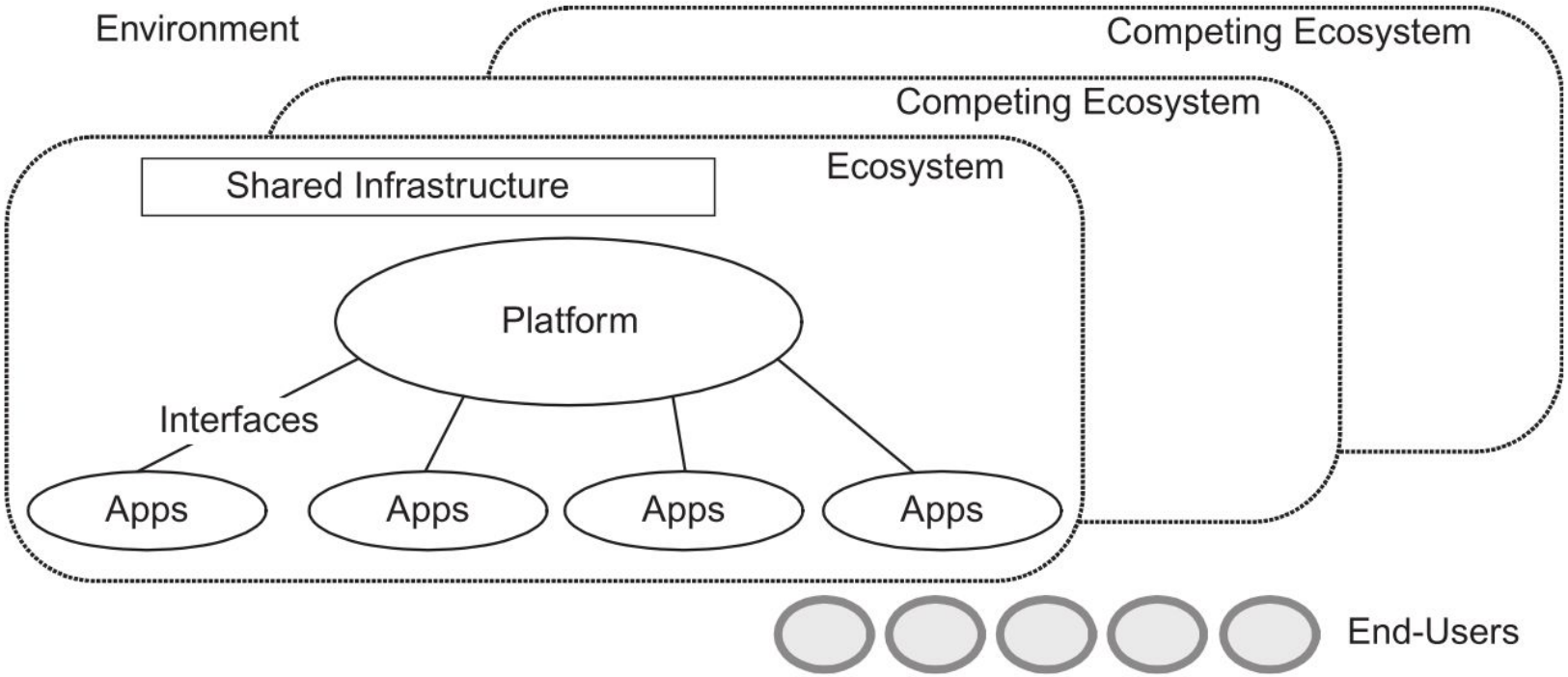


What is a platform ecosystem?

- A platform ecosystem refer to the platform core and the apps that interoperate with it.



Platform ecosystems



Tiwana 2013 p6

What is a platform ecosystem?

Development in Platform Ecosystems

Development in Platform Ecosystems

- Software *development* involves a variety of activities. In this course we will focus on three aspects related to development in platform ecosystems:

- Innovation
- Design
- Programming

Development in Platform Ecosystems - Innovation

- When we develop new applications, this often involves *innovation*.
- Innovation with digital components and artifacts, called *digital innovation* are easier and more available than traditional innovation.
- “Everyone” can now create novel solutions, and distribute them over the internet.
- Different system architectures might however have an enabling or constraining effect on innovation.

- Platform architectures and governance models have some characteristics that may promote innovation.

Development in Platform Ecosystems - Design

- When we develop new applications, this often involves *design*.
- Design can mean a lot of things, and also involves several activities such as
  - Requirements gathering
  - User participation / engagement
  - Graphic design
  - Usability testing
  - User evaluations
  - And so forth.
- Designers often want interfaces and functionality to be based on the user's needs, and be locally relevant to them.
- A problem however emerges when large-scale systems are used by a wide audience, maybe throughout several user groups, domains, departments, organizations, and geographical locations.

Development in Platform Ecosystems - Design

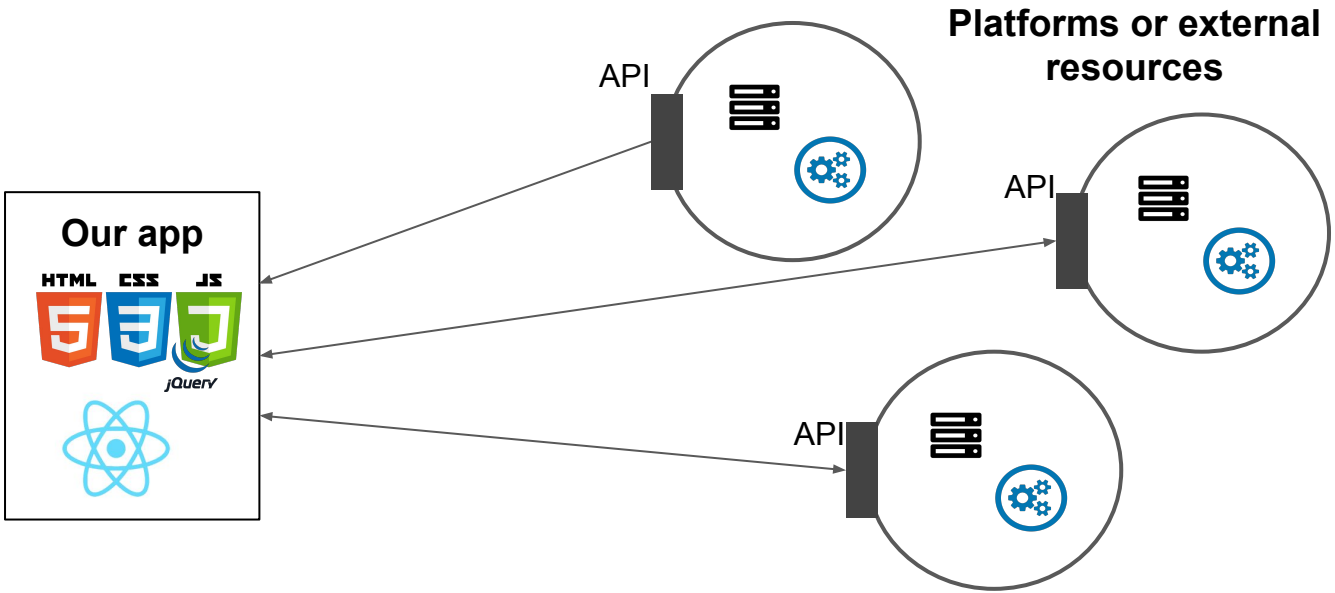
- By providing flexibility for variation, platforms may address issues related to heterogeneity and scale.

- We will look at how platforms may handle diversity both on the local and global level.

Development in Platform Ecosystems - Programming

- When we develop new applications, this involves *programming*.
- Different architectures and governance models will impact our way of implementation.
- Throughout the course, we will practice front-end web development of platform applications.
    - Mandatory individual assignments (basic web programming + APIs)
    - Group project (App development for the DHIS2 platform)

Development in Platform Ecosystems - Programming



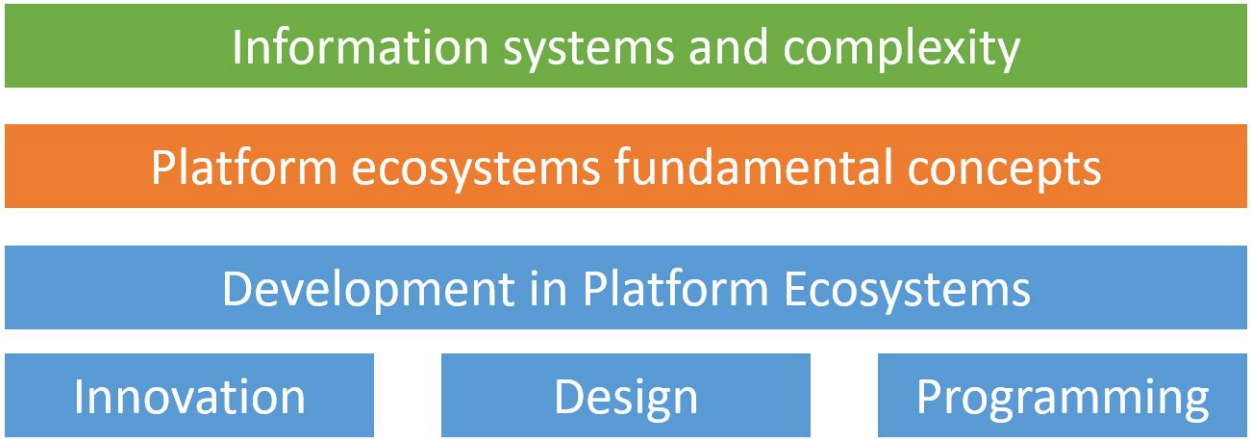
Development in Platform Ecosystems - Programming

Development in Platform Ecosystems - Programming

- JavaScript frameworks are used to make web development faster and easier.
- They provide built-in functionality for common aspects of development so that we do not have to built everything from scratch.
- The most commonly used frameworks are React, Angular and Vue.

Course structure

Development in Platform Ecosystems



Development in Platform Ecosystems

Week		Topic
34		Introduction
35		HTML, CSS, JavaScript
36		Ajax and APIs
37		Web services: history and REST principles
38		React.js
39		Information systems and socio-technical complexity
40		Platforms and ecosystems - fundamental concepts
41		Design, requirements and user participation within platforms
42		Design, requirements and user participation within platforms
43		Developing web applications for DHIS2
44		Developing web applications for DHIS2
45		Platforms and innovation
46		Open source history, philosophy, methodology
47		Cases from HISP and other projects
48		Course summary

Syllabus

Syllabus

Syllabus/achievement requirements

- Course topic overview

Information systems and complexity

- I. Sommerville et al., (2012) - *Large-scale complex IT systems*
- K. Rolland, E. Monteiro (2002) - *Balancing the Local and the Global in Infrastructural Information Systems*

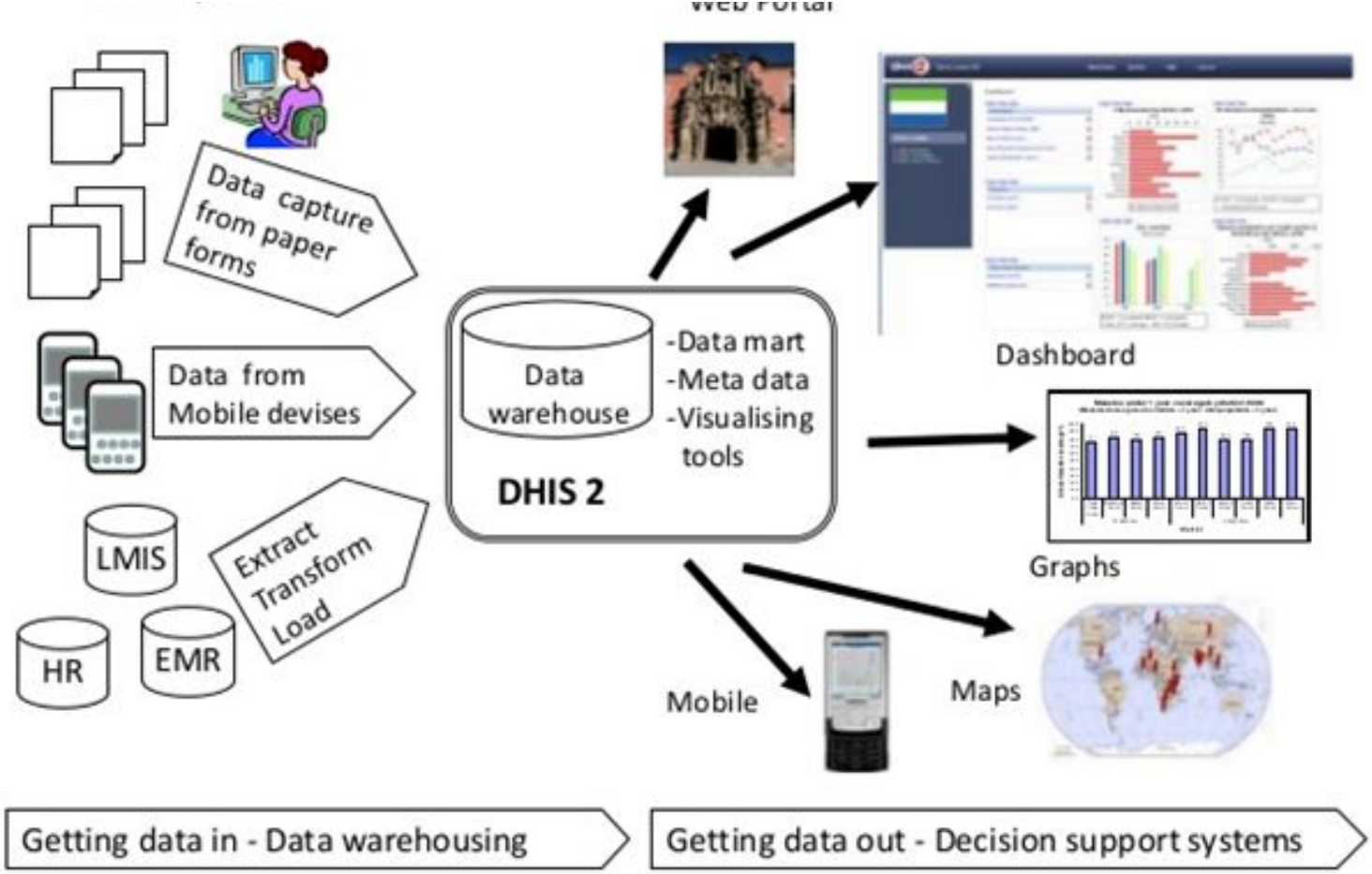
Platform ecosystems fundamental concepts

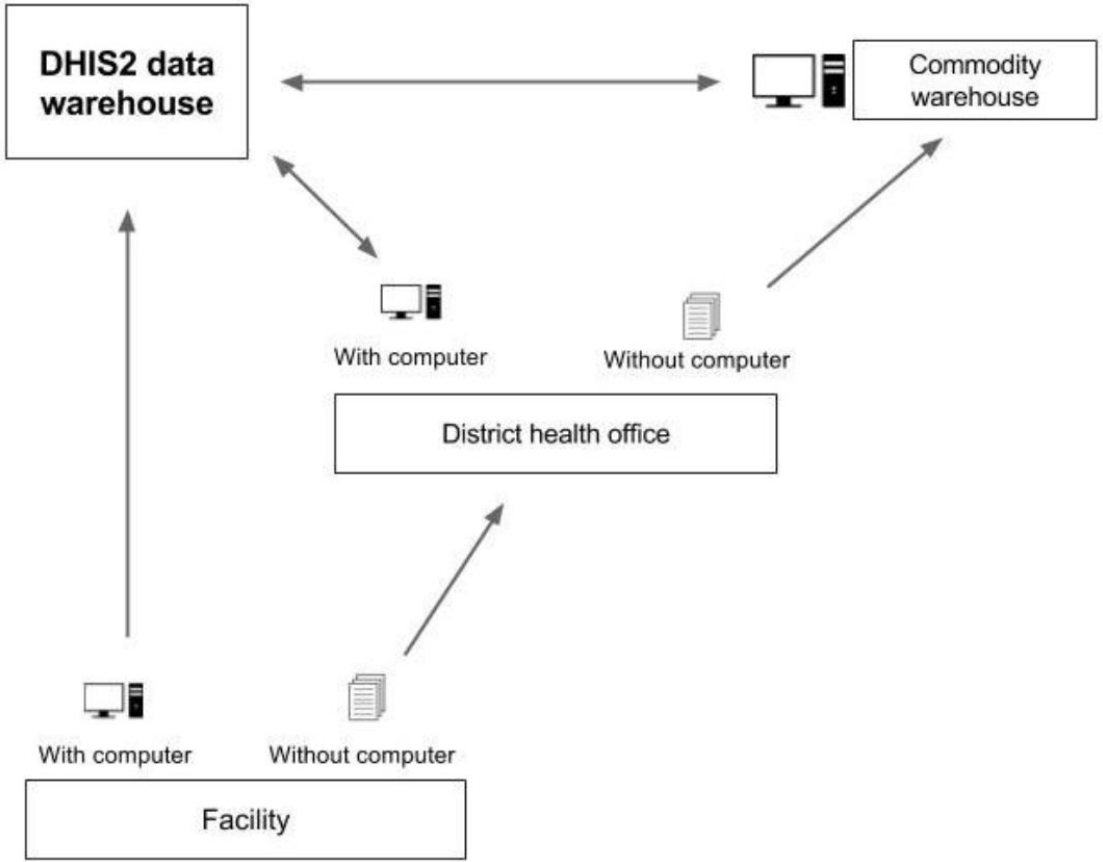
- C. Baldwin & C. Woodard (2008) - *The architecture of platforms: A unified view*
- A. Tiwana (2013) - *Platform Ecosystems*, chapter 1 *The Rise of Platform Ecosystems*
- A. Tiwana (2013) - *Platform Ecosystems*, chapter 2 *Core Concepts and Principles*
- A. Tiwana (2013) - *Platform Ecosystems*, chapter 5 *Platform*

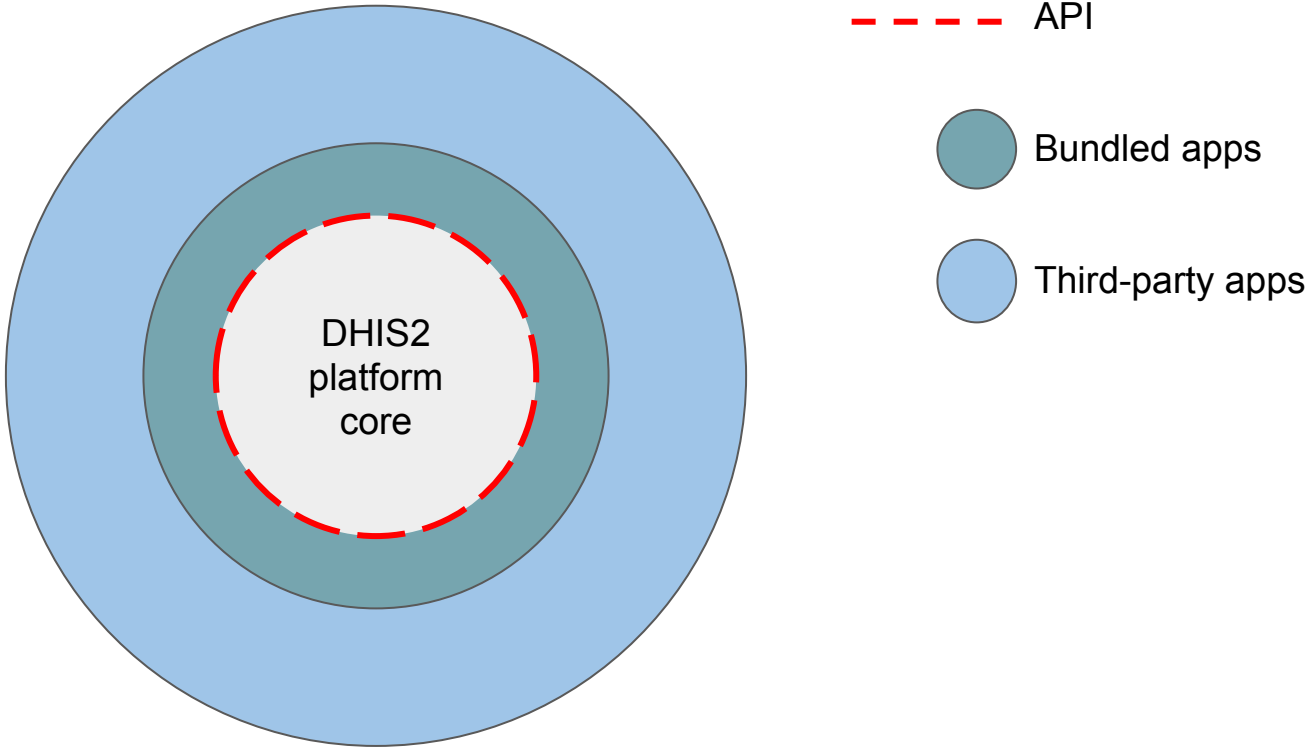
Platform Ecosystems: DHIS2

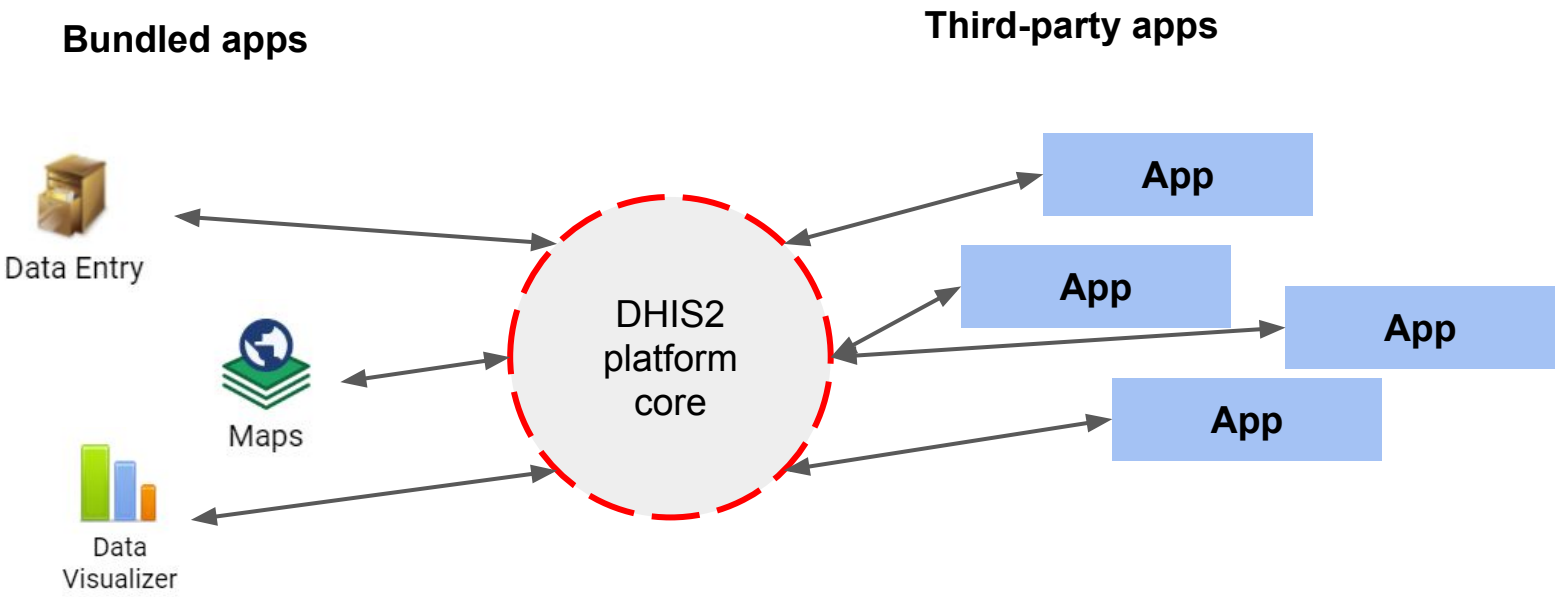
DHIS2 as Platform

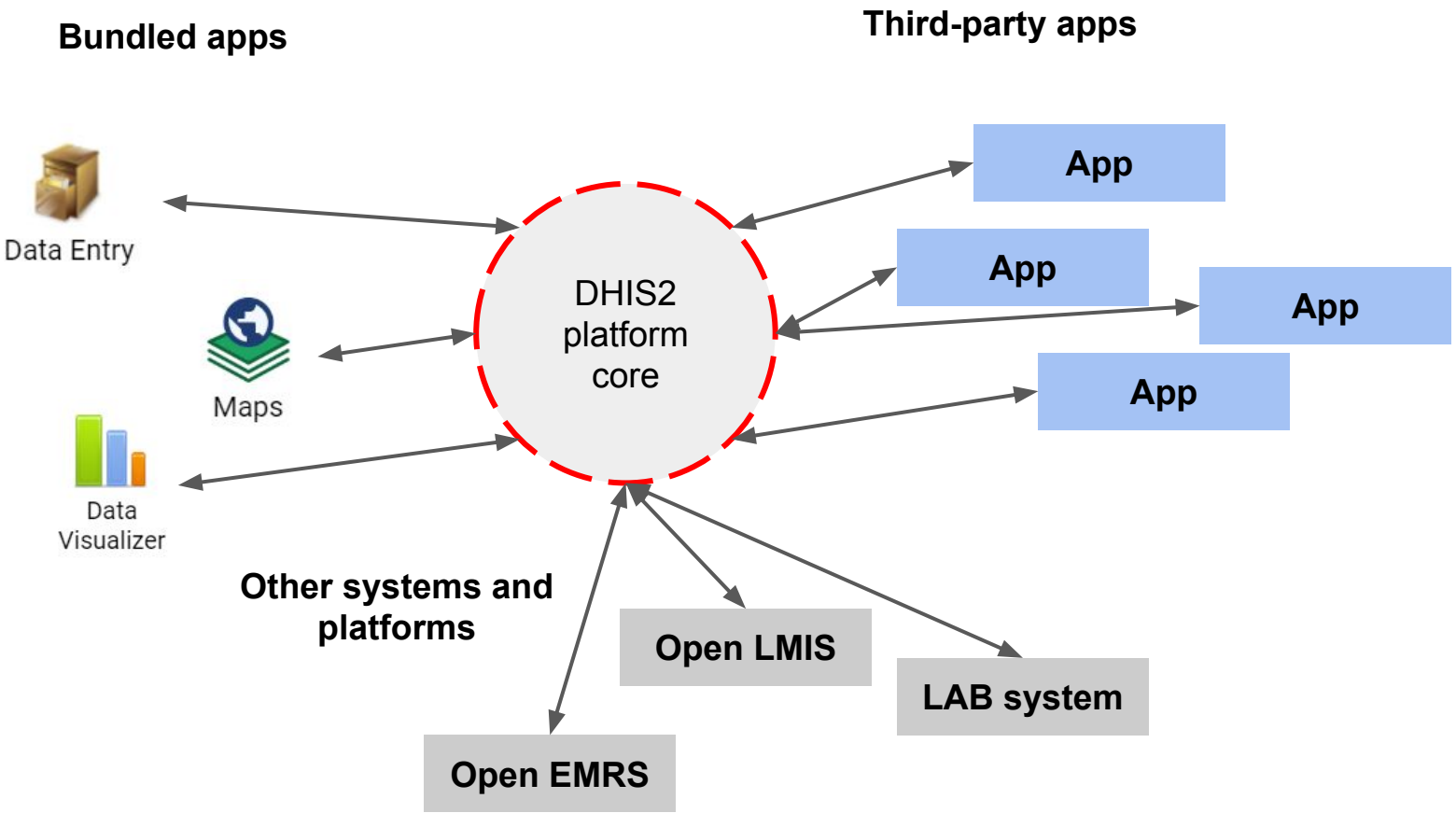
- District Health Information Software 2 (DHIS2) will be used as an example of a platform ecosystem in this course.
- The software (platform core) is developed at the Department of Informatics (Information systems research group)
- Supports collection, storage, analysis and presentation of health-related information.
- Used in over 60 countries world-wide.

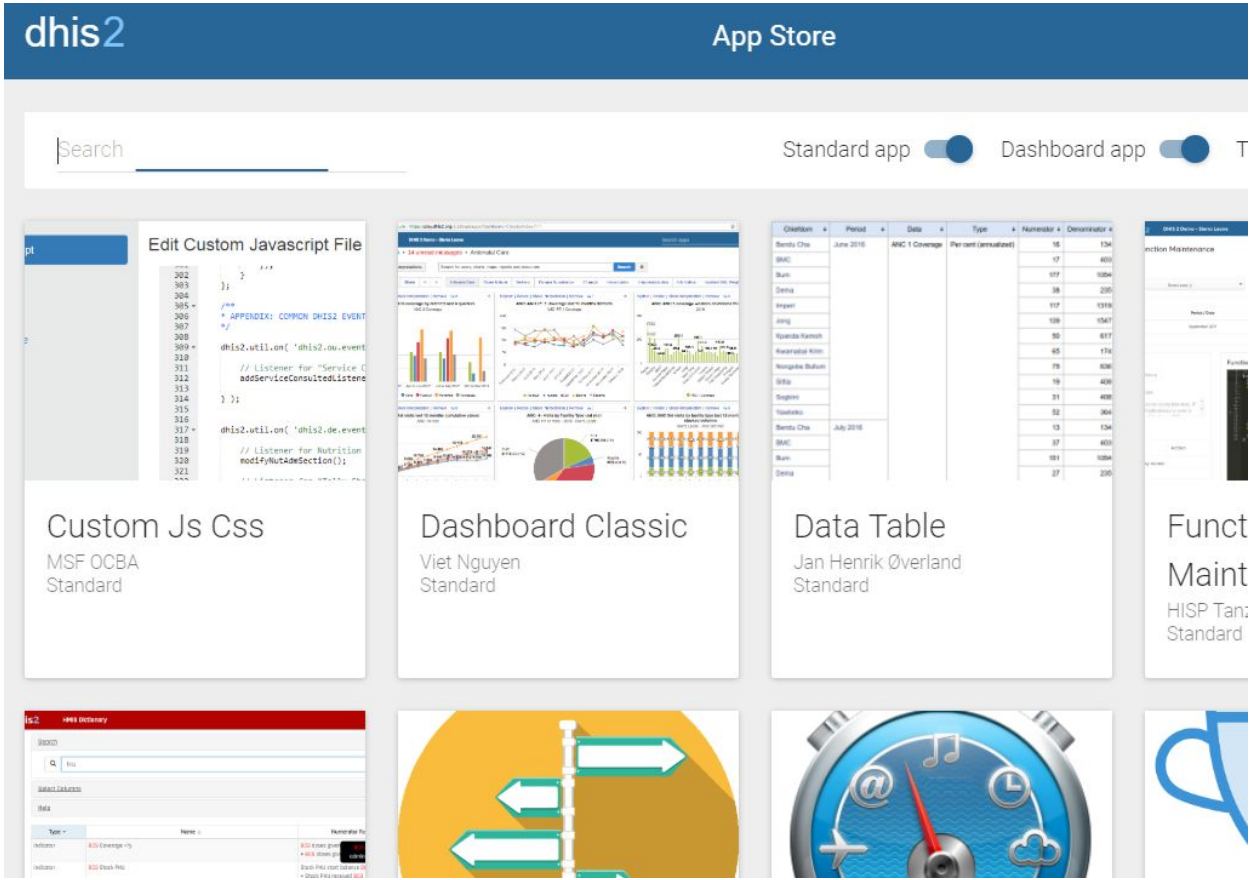

Data capture from paper forms
Data from Mobile devises
LMIS
HR
EMR
Extract Transform Load
Data warehouse
-Data mart
-Meta data
-Visualising tools
DHIS 2
Dashboard
Graphs
Mobile
Maps
Getting data in - Data warehousing
Getting data out - Decision support systems


DHIS2 data warehouse
Commodity warehouse
With computer
Without computer
District health office
With computer
Without computer
Facility


DHIS2
platform
core
API
Bundled apps
Third-party apps


Bundled apps
Third-party apps
Data Entry
Maps
Data Visualizer
DHIS2 platform core
App
App
App
App


Bundled apps
Third-party apps
Data Entry
Maps
Data Visualizer
DHIS2 platform core
App
App
App
App
Other systems and platforms
Open LMIS
LAB system
Open EMRS


dhis2
App Store
Search
Standard app
Dashboard app
Edit Custom Javascript File
Custom Js Css
MSF OCBA
Standard
Dashboard Classic
Viet Nguyen
Standard
Data Table
Jan Henrik Øverland
Standard
Funct
Maint
HISP Tanz
Standard

DHIS2 as Platform - API

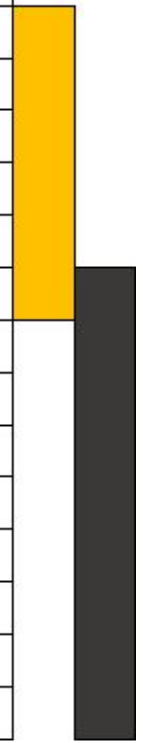
- The DHIS2 API allows internal and third-party developers to communicate with the core resources.

Example: https://play.dhis2.org/2.29/api/resources

Group project

Development in Platform Ecosystems

Week		Topic
34		Introduction
35		HTML, CSS, JavaScript
36		Ajax and APIs
37		Web services: history and REST principles
38		React.js
39		Information systems and socio-technical complexity
40		Platforms and ecosystems - fundamental concepts
41		Design, requirements and user participation within platforms
42		Design, requirements and user participation within platforms
43		Developing web applications for DHIS2
44		Developing web applications for DHIS2
45		Platforms and innovation
46		Open source history, philosophy, methodology
47		Cases from HISP and other projects
48		Course summary



Group project

- Develop web-app for the DHIS2 platform.
- Teams of 4 students.
- 2 - 4 different cases will be provided, and the teams are free to select one case to “solve”.

- The case will provide a background with information about the users, use, and context, and a problem to be solved.
- Based on the selected case, your group shall define requirements for your solution.
- Your web-app shall attempt to provide a solution to the problem, while clearly taking into account the context and users of your case.

Group project - presentations

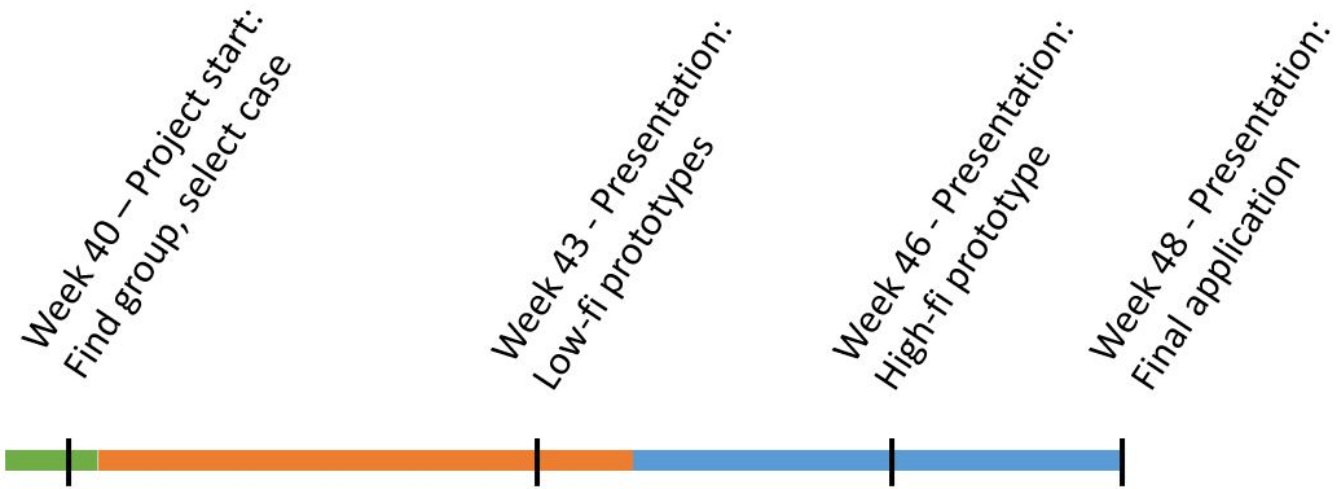
Throughout the project, your team will give three mandatory presentations of your work:

1. Requirements and low-fi prototypes (sketches, wireframes, etc.).
2. First attempt at high-fidelity prototype (web-app).
3. Final project presentation (Graded A-F. Counts 40% of your course grade)

Presentation 1 and 2 will be held in the group sessions. Fellow students, group teachers, and course lecturers will be present. Each presentation (est. 5 min), will be followed by a discussion.

Presentation 3 will be held in front of a panel of DHIS2 core developers and the course lecturers, which together decide the project grade.

Group project



Individual assignments

Mandatory individual assignments

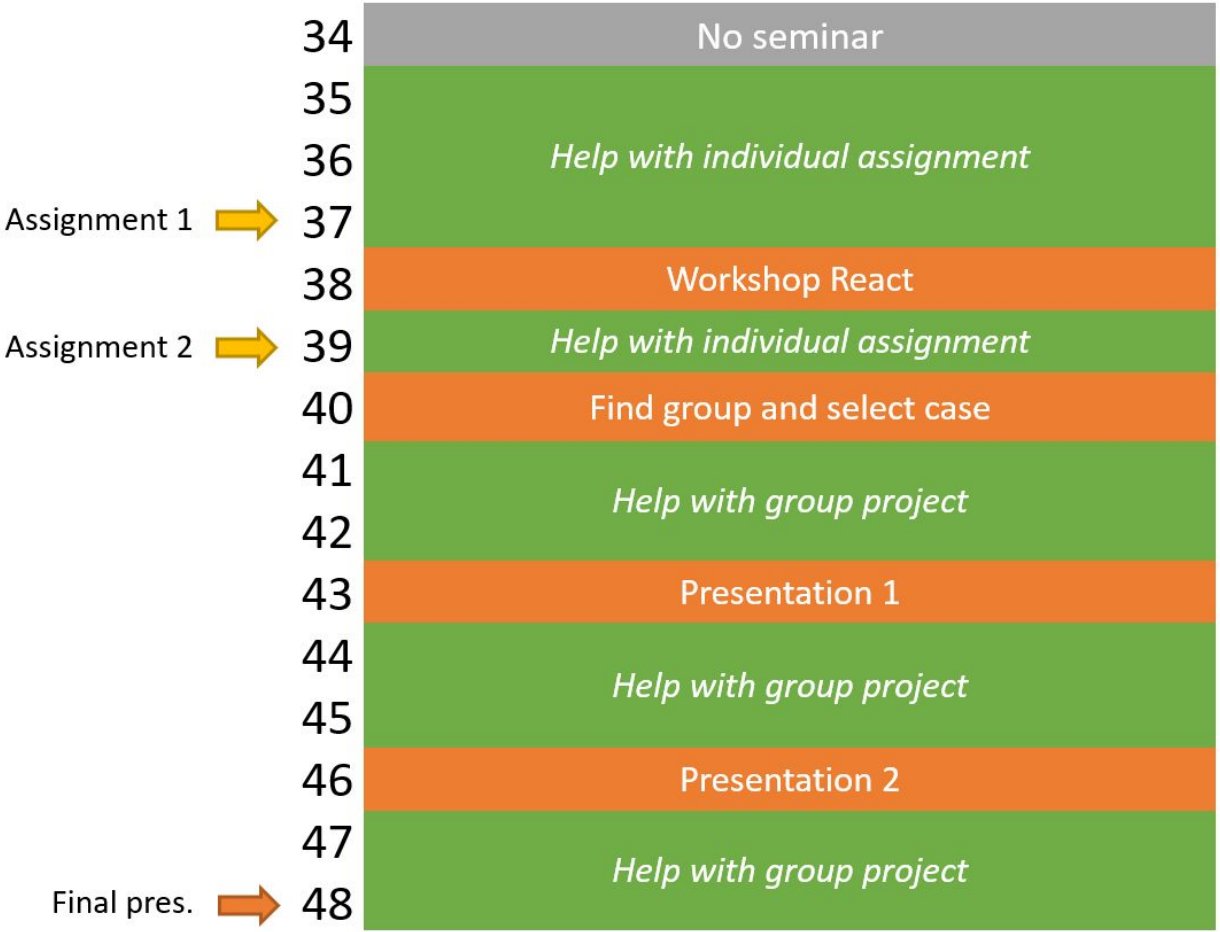
- To get hands-on experience with development in platform ecosystems.
- Prepare you for the group assignment.
- Focus on front-end basics such as HTML, CSS, JavaScript, and AJAX.
- Also, a lecture and workshop on React as a JavaScript framework will be held.

Oblig 1: A collection of minor assignments.

Oblig 2: Mini-project

Seminar groups

Seminar groups



Seminar groups

Group sessions

- Gruppe 1 - Fri 10:15-12:00
- Gruppe 2 - Tue 12:15-14:00
- Gruppe 3 - Fri 12:15-14:00
- Gruppe 4 - Wed 10:15-12:00
- ~~Gruppe 5 - Thu 08:15-10:00~~

Final exam

Final exam

- December 4th of December
- Individual
- Four hours
- No help resources
- Digital (Silurveien)
- Collection of questions requiring short and long answers.
- Mostly theoretical, some practical.
- Theoretical weekly assignments provide an indicator.
    - Some of these may be used in the final exam.

Exam from 2017 (When the course name was Open Source Development)