

Vibe Coding i byggeriet

Fra Mandag d. 23. november 2026 til Torsdag d. 7. januar 2027

Ballerup, WTC – World Trade Center Ballerup

Dag 1: Mandag d. 23. november 2026

08:30 – 09:00 **Registrering – og morgenmad**

09:00 – 09:30 **Introduktion af underviser og dagens program**
v/ Mathias Sønderskov Schaltz

Hvad er vibe coding og hvorfor bruger vi tid på det?

Eksempler "fra det virkelige liv" i LINK Arkitektur.

For at sætte stemningen løber vi igennem udvalgte projekter:

- Revit Checker – check rumprogrammer
- Rhino Aioli – Klimastudier på volumenstudier.
- Rhino/Revit Daylight – Dagslyssimuleringer
- Rhino/Revit/Webapp – Kørekurver
- Hverdagens små "scripts" direkte fra AI chat.

Takeaway: AI gør det muligt hurtigt at gå fra idé til prototype, og helt til færdigt produkt.

09:30 – 10:00 **Introduktion til AI kodning med LLM'er**

Vi gennemgår de nye AI-kodeagenter og de vigtigste begreber.

Claude Code / Codex: AI-agenter der kan arbejde direkte med vores filer, terminal og kode, enten lokalt eller via skyen.

Hvordan virker AI kodning?

LLM: Er en avanceret "gæt det næste ord"-algoritme.

Agent: LLM + tools + data(filer, git etc)

Github: Versionshistorik, samarbejde, automatiske actions

Vi gennemgår de vigtigste principper for at arbejde effektivt med AI-agenter:

- Context – hvordan giver vi AI'en den rigtige information?
- Feedback loops – hvordan får vi AI'en til at opdage og rette sine egne fejl?
- Skills / CLAUDE.md – hvordan giver vi AI'en permanente instruktioner?
- Guardrails og prompt injections – hvad skal man være opmærksom på, når AI får adgang til vores computer og kode?

10:00 – 10:15 **Pause**

10:15 – 10:45 **Opsætning**

Vi får jer sat op med:

ClaudeCode (hvis I har Codex kan I også bruge det, men underviser kører i skrivende stund Claude)

Github – for versionsstyring af jeres kode samt test i skyen og build + lave en konto.

Visual Studio – for at køre og teste lokalt.

Vi får sat alle deltagernes computere op med:

Github konto + et repo (en slags hotel til dit kode, med versionshistorik)

Codex/ClaudeCode CLI

Codex/ClaudeCode adgang til Github

10:45 - 11:00

Bygge vores første plugin til revit lokalt

Vi ser hvor langt vi kan komme ved hjælp af ClaudeCode lokalt. Vi får den til at holde os i hånden undervejs.

Vi gennemgår et feedback loop:

"Idea → prompt → code → compile → error → give error to Claude → fix → test → refine"

Opgaven er en modeltjekker der finder alle rum som:

- er større end xx m²
- har længere end yy meter til flugtvej

Der skal udvikles et UI omkring det, og så når vi sikkert også at lægge flere ting på.

11:00 - 12:00

Self-work på opgaven

Nu bygger I videre på jeres egen modeltjekker.

Underviser hjælper med prompting, debugging og Revit API-udfordringer.

Vi lærer AI'en at kende og samarbejder med den.

Hvis man er færdig med grundopgaven, kan man f.eks. bygge:

- flere modelchecks
- bedre UI
- rapport / eksport
- visualisering af resultater
- egne idéer

12:00 - 12:45

Frokost

12:45 - 14:00

Arbejde videre på opgave i teams

14:00 - 14:30

WebApps - Hvad er de og hvorfor?

Kan anvendes i både Rhino og Revit

Hurtigere iterationer med LLM i udvikling/test

Case: LINK KøreKurver

14:30 - 16:00

Arbejde videre på opgave i teams

Inkl. pause med kage

16:00 - 16:30

Fælles opsamling og spørgsmål

16:30

Tak for i dag - og på gensyn i morgen

Dag 2: Tirsdag d. 24. november 2026

09:00 - 09:15

Opsamling fra i går

Gennemgang af dagens program.

09:15 - 10:00

Fra eksperiment til projekt:

v/ Mathias Sønderskov Scholtz

(Github for versionsstyring, parallelle tracks + (evt)Virtual Machine for testing)

Nu tager vi vores lille eksperiment og gør det til et rigtigt softwareprojekt.

- Git og commits
- Branches og parallelle tracks
- Hvordan AI-agenter kan arbejde direkte på et GitHub repository
- README og automatisk dokumentation
- GitHub Actions til build og test
- Hvordan vi kan teste kode automatisk, selvom Revit ikke nødvendigvis kan køre direkte i skyen

Vi laver en simpel static webapp på GitHub, som fungerer som dokumentation og en lille "viewer" til vores projekt.

Hvis tiden tillader det, ser vi også på hvordan en Windows/Virtual Machine kan bruges til automatiseret Revit-test.

10:00 - 11:40

Gruppearbejde

11:40 - 12:00

Iterativt loop omkring en dagslysberegner

v/ Mathias Sønderskov Scholtz

En gennemgang af hvordan LINK Arkitektur har skrevet deres egen beregningskerne, og hvordan man kan sætte tests op, som LLM'en kan bevæge sig imod.

Udviklet henover en sommerferie. Primært på Claude på telefonen og Claude Remote.

12:00 - 12:45

Frokost

12:45 - 14:00

Distribution

v/ Mathias Sønderskov Scholtz

Vi får lavet en msi installer.

Case: Hvordan LINK har Github -> Actions -> Azure, samt Rhino Yak Packagemanager.

14:00 - 14:15

Pause med kage

14:15 - 15:30

Fælles præsentationer af projekter

15:30 - 16:00

Hvad vi ikke fik rørt i dag

v/ Mathias Sønderskov Scholtz

Rhino/Revit MCP – få din LLM til at snakke direkte i din model

Små hacky workflows: Python scripts (pyRevit)

OpenSource projekter udefra (og faldgrupper med agent instructions).

16:00 - 16:30

Afslutning

Hvad gør I fremadrettet? Andre AI udbydere? Åbne modeller? What is the 'moat'?

Hvor finder man mere info?

16:30

Kurset slutter - tak for nu

På gensyn på webinarret

Opfølgingswebinar: Torsdag d. 7. januar 2027

13:00 - 14:00

Online opsamling på kurset

v/ Mathias Sønderskov Scholtz