

Startup Studio

AI-Accelerated Building and Validation

01.29.26 C3W2

Agenda

- Class size and waitlist
- Class structure recap
- Homework 1 Discussion
- Exercise
 - Fermi Estimation

Instructor Intro

Who am I?

Ken St. Clair

- Early-stage start-up Product Leader with 20+ years experience
 - Software engineering, sales engineering, product design, product leadership
 - Two robotics products from 0→1
 - Experience with 0→1, growth, acquisition, and depreciation
 - B2B, B2B2C, B2C
- BS Computer Science, Georgia Tech
- Masters of Human-Computer Interaction, Carnegie Mellon



aescape



 MISO
ROBOTICS

facebook

Carnegie
Mellon
University

 Georgia Institute
of Technology

Whom have I Worked with?

Select Partners and Clients

Gyms + Spas

Print Media

Broadcast

Streaming/Sports

Food Service

E Q U I N O X

The New York Times

TURNER

MLG
MAJOR LEAGUE GAMING

LIFE TIME
HEALTHY WAY OF LIFE

Bloomberg

Disney

twitch

OUTBACK
STEAKHOUSE



LOTTE NEW YORK PALACE

REMEDY PL

**THE
WALL STREET
JOURNAL.**

The Atlantic

ESPN

ROKU

White Castle



THE WELL

**THE
HUFFINGTON
POST**

VICE

NBC

VIACOM

CBS

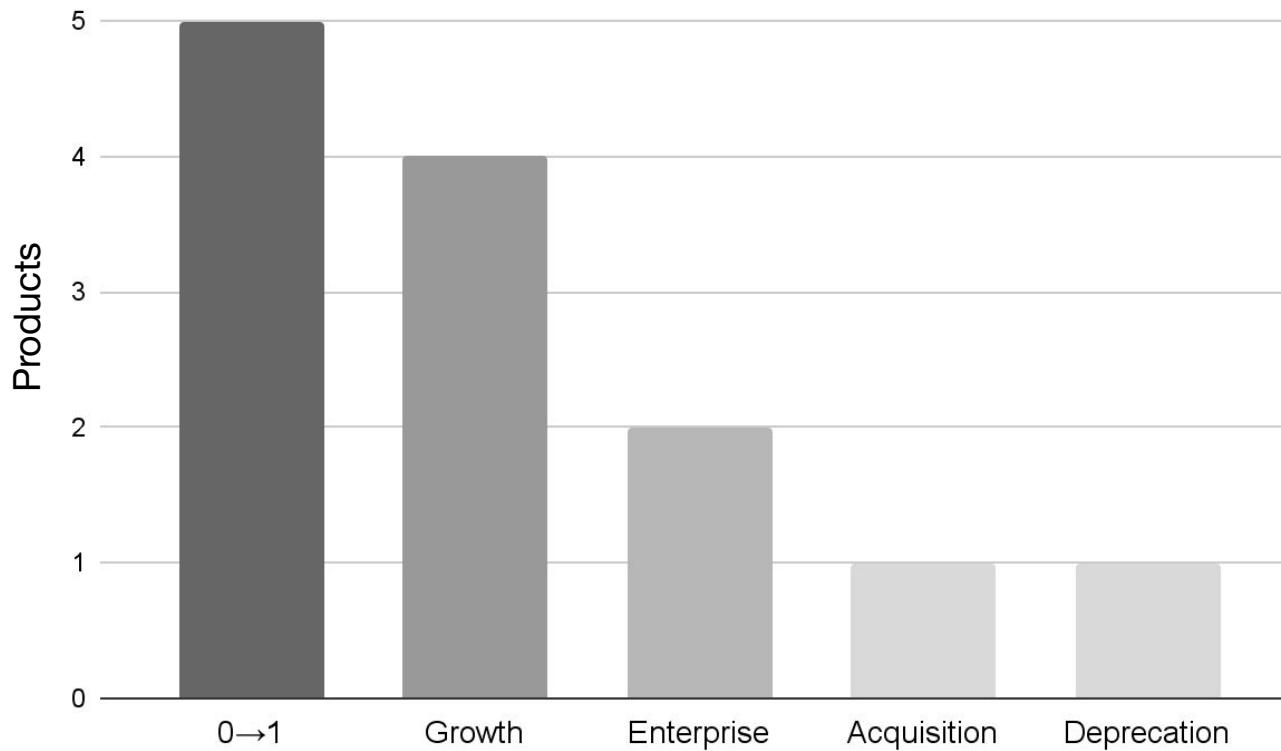
NFL

tinder

Nationwide

More...

Product Stages I've Worked On



Class Size and Waitlist

Waitlist

- Very impressive amount of demand for this course (>150 registrations)
- Course is capped at 50. This is the final decision from the administration.
- We are limited by the studio nature of the course and the final, which will include industry leaders in a Demo Day
- Final enrollment list published after class Thursday

Class Intro

What's a “Studio”

A course focused on making, iterating, and critique.

Students learn by building real artifacts and receiving continuous feedback, rather than by absorbing theory first.



What's “Startup Studio”

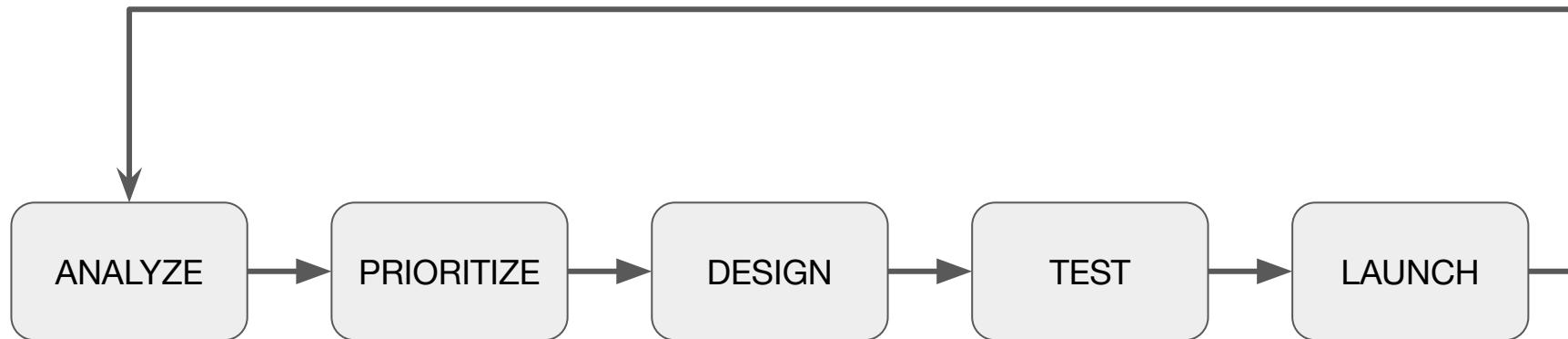
A course focused on making, iterating, critique, **experimentation and launching**.

Students learn by building real artifacts and receiving continuous feedback, rather than by absorbing theory first.



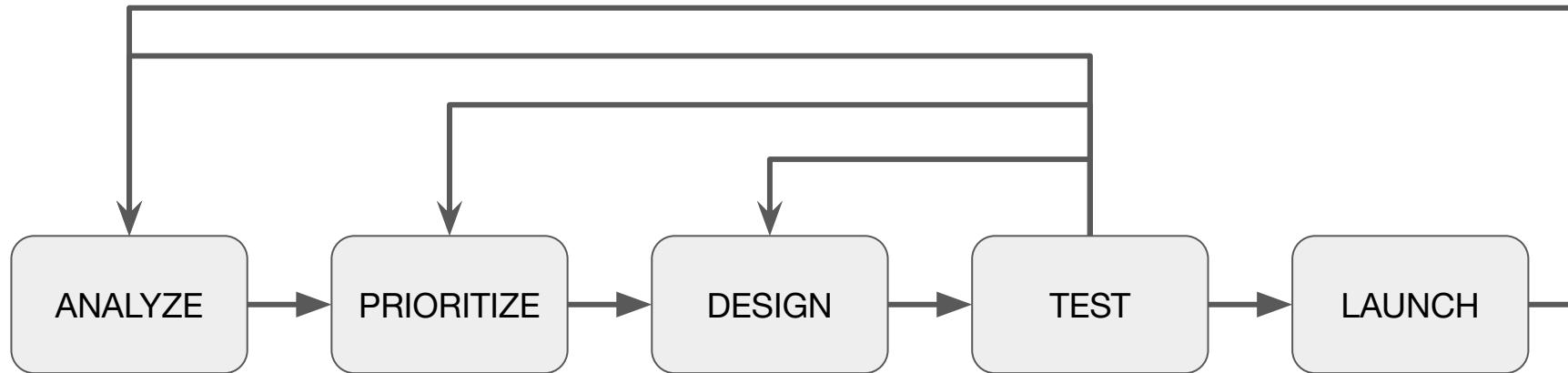
Startup OS

The operating system for this class



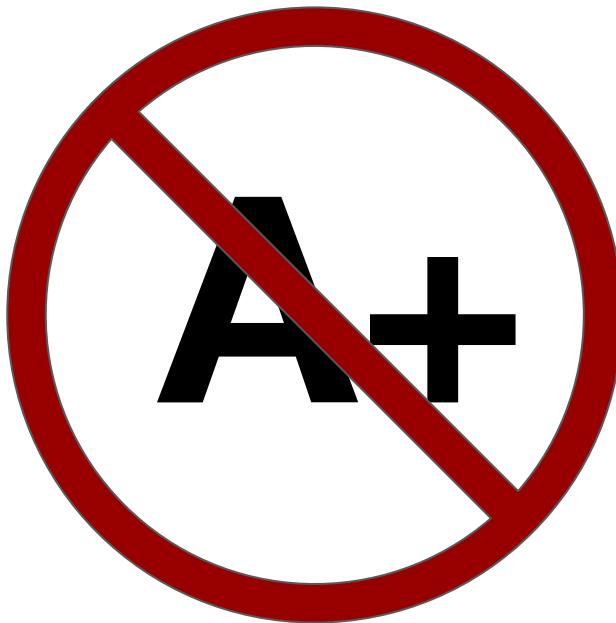
Startup OS

The operating system for this class



What We're NOT Focusing On

- Hardware
- B2B
- Research/Publishing
- Theory/Deep Foundational Learning



This is not an A+ course

Class Position on AI

Use AI for everything.

The AI you use will eventually come with a monetary cost.

Waitlist

- The course is highly oversubscribed. If you are on the fence in any way, kindly make way for those that will find the most value from the course
- Be sure to let your administrators know you'd like to see the course offered again.

Homework 1 Discussion

What did you Build? krs2212

- What did you create?
- Which AI tools did you use?
- What was your process?
- How long did it take?

Gmail chat me your link and I'll show it (krs2212)

What did AI do well?

What surprised you about what AI could do? Where did it exceed your expectations?

- Letting AI decide color was fairly clean
- Original UI looks strong
- Quick for tightly scoped
- Does well at making stuff up for quick pitches/prototypes
- API integration much easier
- Suggesting thing you don't know
- Feature by feature gave strong results

What did AI Struggle with?

Where did you have to step in? What did AI get wrong repeatedly? What took longer than expected?

- Feature 3 breaks feature 1
- A lot of bloat
 - Garbage code - really big functions that aren't needed
 - Workflow Errors create new files
- Wanted to use a specific library/mathematical process, but wasn't able to bring it into the tool that was being used
- Made up user ratings and other marketing data that was not true
- More polished UX, especially around semantics, was lacking
- Dead buttons/links not leading anywhere
- Unrealistic expectations of humans in the UX
- Outdated versions of SDKs
- Large large codebases without a lot of direction
- CONTEXT IS LOST EVERY CHAT
- Get stuck on ideas, goes down rabbit holes
- Complex deployment

The Reality of AI-Assisted Building

AI is a productivity multiplier, not a replacement

You still need:

- Clear vision & requirements
- Technical judgment
- Debugging skills
- Architecture understanding

How do Humans do this?

For those that have worked in building software, what's the human process?

- Product manager analyzes business goals
 - Negotiate with other business stakeholders
- PM works with Engineering Manager
 - Divides work into sprints/chunks
- Storyboard for workflow from UX designer
 - Breaking down into stories/features for consumability
- Test
 - Unit/Regression
 - QA
 - Beta
 - User Testing
- Users
- Marketers
- Designer

What works for humans typically also works for AI

How to Make AI Development Work

Clear requirements documents

→ helps AI understand context

Breaking work into specialized roles

→ matches AI's strength in focused tasks

Living documentation

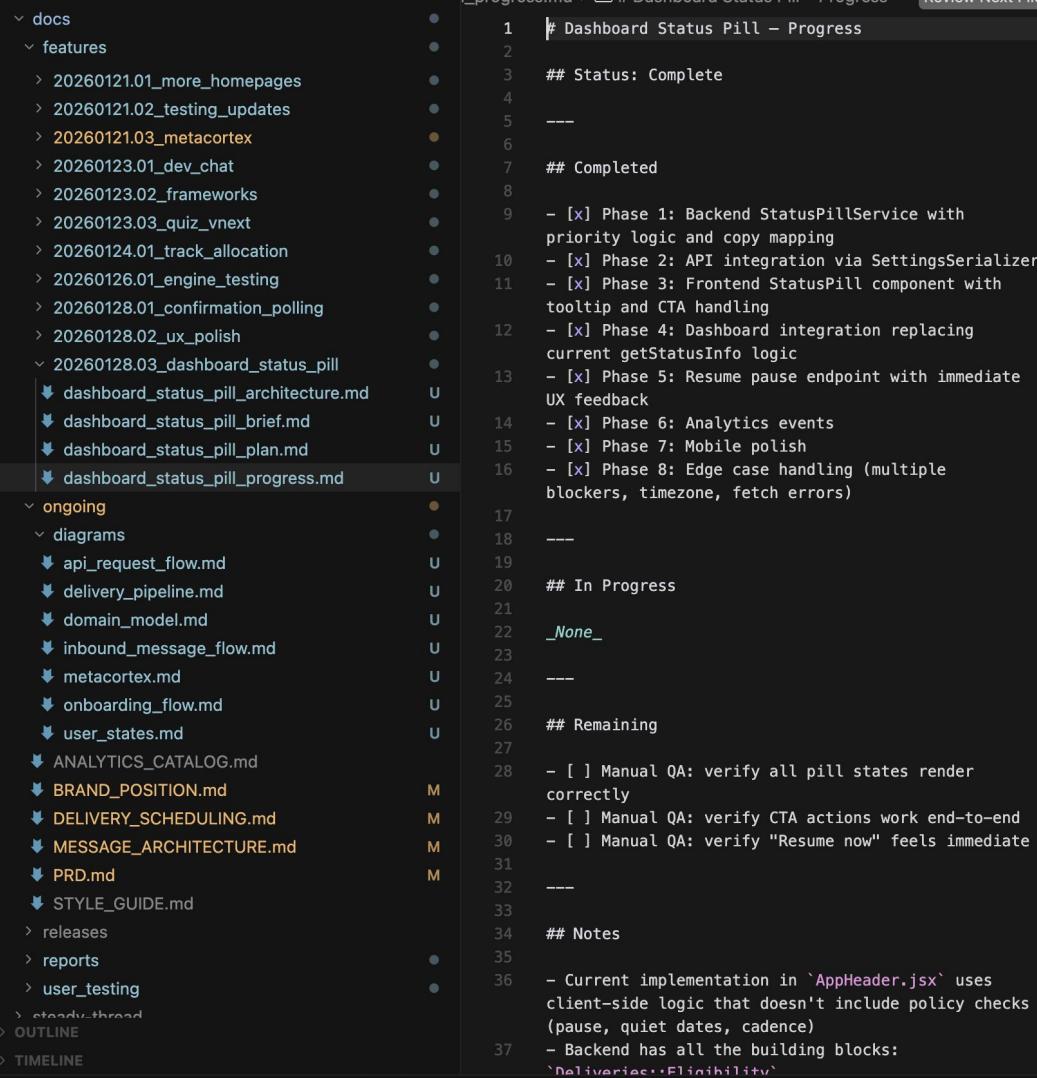
→ gives AI persistent memory

Human oversight at key checkpoints

→ catches AI's blind spots

A Quick Glance at Structure

```
 1  # Dashboard Status Pill - Progress
 2
 3  ## Status: Complete
 4
 5  ---
 6
 7  ## Completed
 8
 9  - [x] Phase 1: Backend StatusPillService with
10 priority logic and copy mapping
11 - [x] Phase 2: API integration via SettingsSerializer
12 - [x] Phase 3: Frontend StatusPill component with
tooltip and CTA handling
13 - [x] Phase 4: Dashboard integration replacing
current getStatusInfo logic
14 - [x] Phase 5: Resume pause endpoint with immediate
UX feedback
15 - [x] Phase 6: Analytics events
16 - [x] Phase 7: Mobile polish
17 - [x] Phase 8: Edge case handling (multiple
blockers, timezone, fetch errors)
18
19  ---
20  ## In Progress
21
22  _None_
23
24  ---
25
26  ## Remaining
27
28  - [ ] Manual QA: verify all pill states render
correctly
29  - [ ] Manual QA: verify CTA actions work end-to-end
30  - [ ] Manual QA: verify "Resume now" feels immediate
31
32  ---
33
34  ## Notes
35
36  - Current implementation in 'AppHeader.jsx' uses
client-side logic that doesn't include policy checks
(pause, quiet dates, cadence)
37  - Backend has all the building blocks:
`Deliveries` + `Eligibility`
```



The image shows a file browser interface with a dark theme. On the left, a tree view displays a project structure with the following files and folders:

- docs
- features
 - 20260121.01_more_homepages
 - 20260121.02_testing_updates
 - 20260121.03_metacortex (highlighted in orange)
 - 20260123.01_dev_chat
 - 20260123.02_frameworks
 - 20260123.03_quiz_vnext
 - 20260124.01_track_allocation
 - 20260126.01_engine_testing
 - 20260128.01_confirmation_polling
 - 20260128.02_ux_polish
- 20260128.03_dashboard_status_pill
 - dashboard_status_pill_architecture.md
 - dashboard_status_pill_brief.md
 - dashboard_status_pill_plan.md
 - dashboard_status_pill_progress.md (highlighted in orange)
- ongoing
 - diagrams
 - api_request_flow.md
 - delivery_pipeline.md
 - domain_model.md
 - inbound_message_flow.md
 - metacortex.md
 - onboarding_flow.md
 - user_states.md
 - ANALYTICS_CATALOG.md
 - BRAND_POSITION.md
 - DELIVERY_SCHEDULING.md
 - MESSAGE_ARCHITECTURE.md
 - PRD.md
 - STYLE_GUIDE.md
- releases
- reports
- user_testing
- steady-thread
- OUTLINE
- TIMELINE

Exercise

Fermi Estimation

What is Fermi Estimation?

Breaking down complex estimates into smaller, manageable assumptions to calculate approximate audience sizes.

Why It Matters

Helps you prioritize which personas to target. Larger audiences = more potential impact and validation.

Fermi Estimation

Example Framework

1. Start with a large population (e.g., US adults = 260M)
2. Apply filters based on your persona (e.g., $\times 40\%$ live in cities = 104M)
3. Narrow by behavior/need (e.g., $\times 15\%$ order takeout weekly = 15.6M)
4. Apply additional constraints until you reach your target audience

Fermi Estimation

AGM Trick (Approximate Geometric Mean)

Time per day a 15-yr old watches TV?

2min - 400min

$2 \times 10^0 - 4 \times 10^2$

Average coefficients and powers

3×10^1

End of Class 3