Lecture 6: October 25

UX Design and Architecture Design

Agenda

- Presentation 2: Technical Design
- Writing 3
    - Design System
    - UX Design
    - Architecture Design
- November Sprint Planning + Alpha Demo

Presentation 2: Technical Design

Presentations on 11/8

Presentation Requirements

- **Goal**
  - Share technical design of your project to a technically savvy audience. Convince the audience of the technical innovation & challenges.
- **Requirements**
  - 7 minutes long
  - Share system architecture
  - Be prepared to answer technical questions

Presentation Content

- Overview & motivation of project goals (keep to 1 minute)

- Architecture diagram

- Key technical components + who is developing each piece

- What is the technical innovation

- Technologies used, how pieces interact

- Goal & deliverable for alpha prototype (keep to 1 minute)

- Technical risks & challenges

Put effort into visuals, don't use a ton of text!

Design Systems

Design System

- **What**
  - Set of standards to manage design at scale by creating a shared language using reusable components and patterns
- **Why**
  - Reduces redundancy on Design and Engineering in order to build quicker
  - Creates unified language across cross-functional teams
  - Increases visual consistency across different pages and channels
  - Learning tool for new Designers
- **Atomic Design Theory** has been created to support Design System creation
 - Established by Brad Frost

Design System

What's included

- **Style guide**
  - Typography, colors, logos, etc
- **Component Library**
  - Reusable UI elements
- **Pattern Library**
  - Similar to Component but more high-level collections of components
- **Design System Team**
  - Team establishing and maintaining the Design System
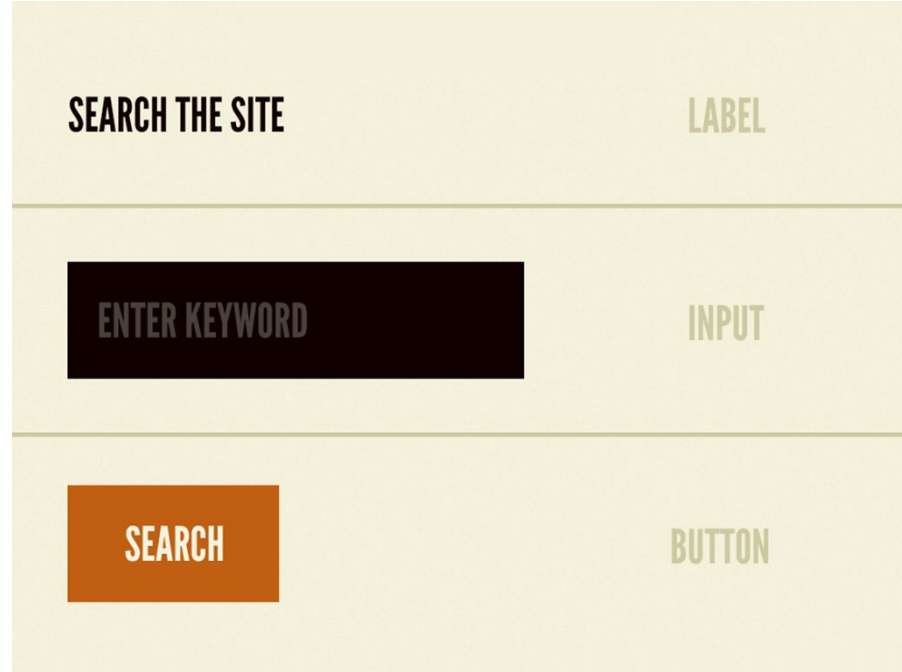 - Made of Product Designers, Visual Designers, and Engineers
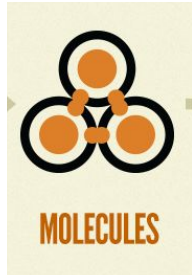
Atomic Design
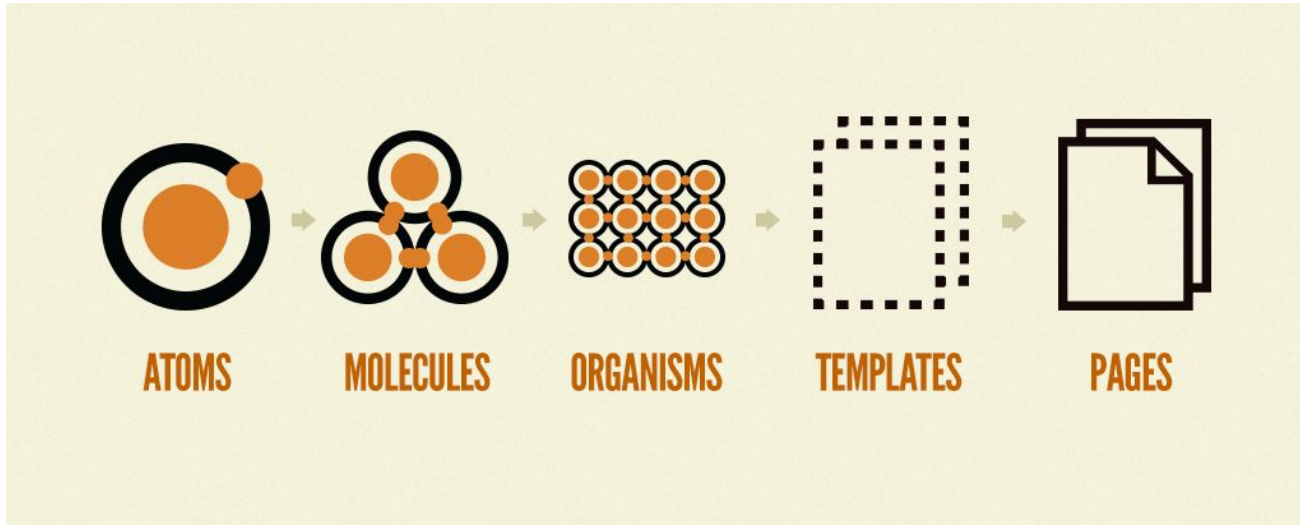
Atomic Design - Atoms
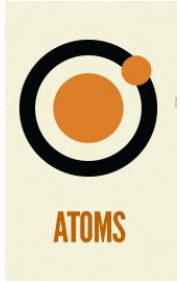
Atoms: Basic HTML Elements

Examples: form labels, inputs, buttons, and others that can't be broken down any further without ceasing to be functional.

ATOMS

SEARCH THE SITE — LABEL

ENTER KEYWORD — INPUT

SEARCH — BUTTON

Atomic Design - Molecules

MOLECULES

Molecules: Combination of Atoms

Examples: form label, search input, and button can join together to create a search form molecule
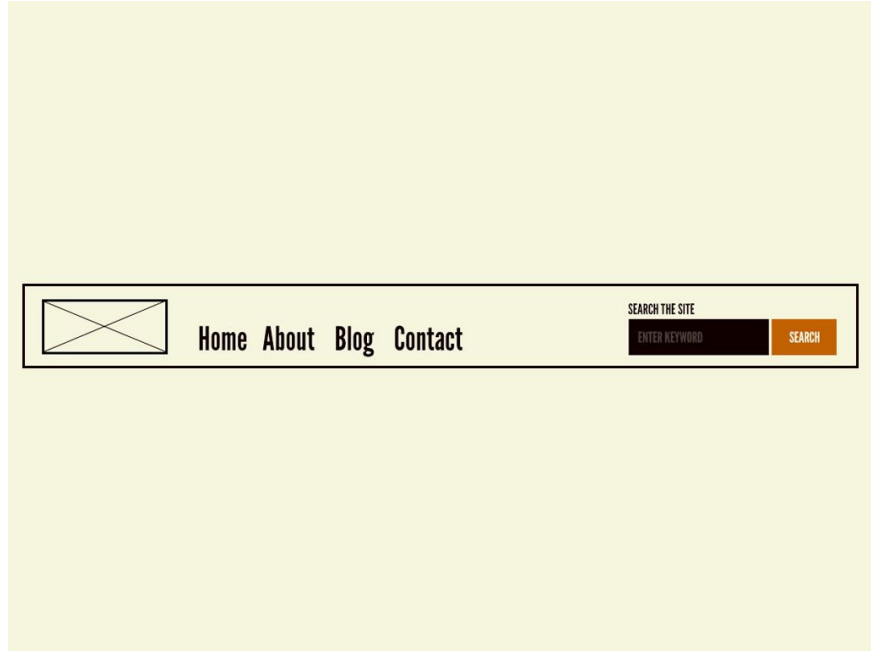

SEARCH THE SITE

ENTER KEYWORD

SEARCH

Atomic Design - Organisms

Organisms: complex UI components composed of groups of molecules and/or atoms and/or other organisms

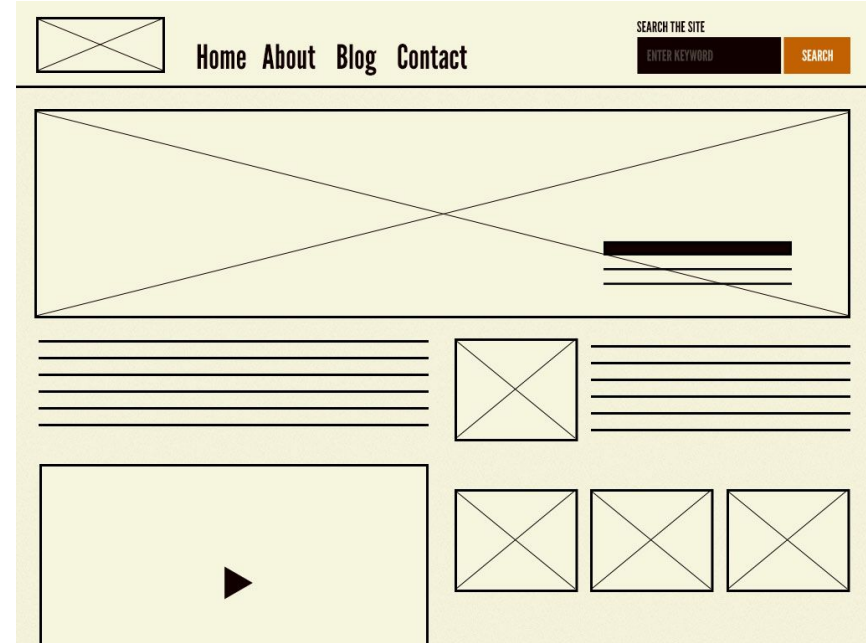**Examples**: header navigation, footer navigation, menu panels

Atomic Design - Templates

Templates: page–level objects that place components into a layout and articulate the design's underlying content structure

Examples: Header combined with visual content with descriptions and action buttons – all laid out in one repeatable space
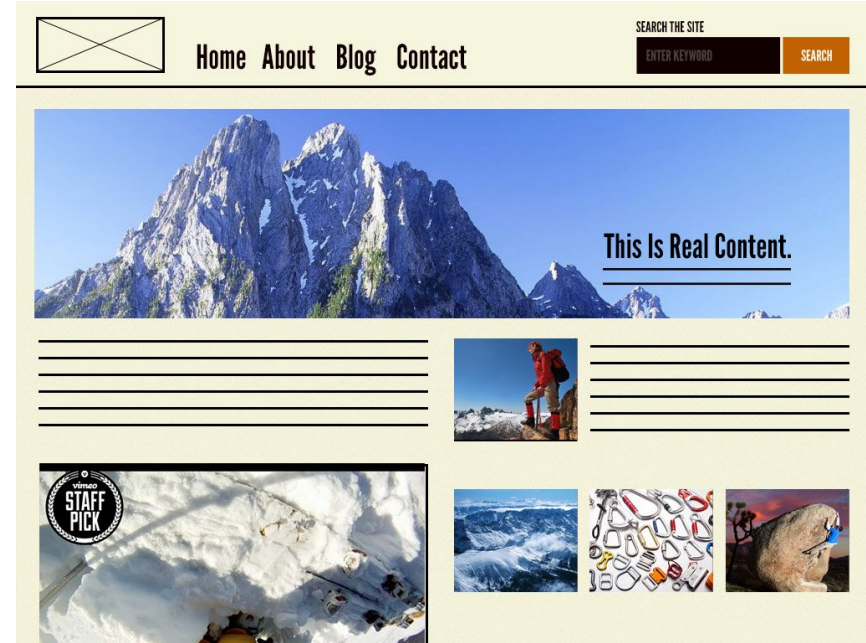
TEMPLATES

Atomic Design - Pages

Pages: specific instances of templates that show what a UI looks like with real representative content in place

Examples: Home page, Contact Us page, etc

Example Design Systems

- https://developer.apple.com/design/human-interface-guidelines
- https://m3.material.io/foundations
- https://www.ibm.com/design/language/

Design Tools

- Figma
- InVision

UX Design and Technical Specs

Writing Assignments

Section 1: Proposal ...

Executive Summary - Writing 1 ✅

Section 2: Specs...

Gantt Chart ✅

Technical Summary - Writing 2 🟡

Product Specifications - Writing 3 (due 11/19)

Technical Specifications - Writing 3 (due 11/19)

"Putting it all together" ... Technical Design Document - Writing 4

Writing 3 Requirements

Product / Design Specs

- User Stories
- User Flows
- Wireframes/Mockups/API Docs

Technical Specs

- Architecture Diagrams
- External APIs, Internal APIs, and Frameworks
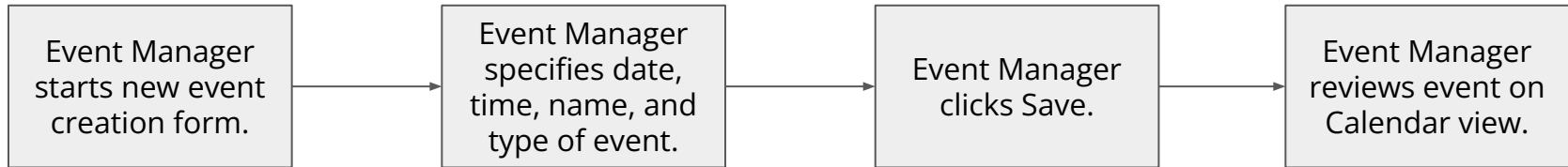- Main Algorithms of System

UX Design and User Flows

- Built off of User Stories / Use Cases

- Creates high-level app flow

UX Design

- Built off of User Stories / Use Cases

As a Event Manager, I would like to schedule events so that people can see which events are available to register for.

| Event Manager starts new event creation form. | → | Event Manager specifies date, time, name, and type of event. | → | Event Manager clicks Save. | → | Event Manager reviews event on Calendar view. |

UX Design

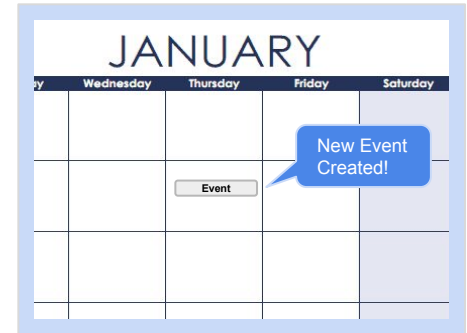- Built off of User Stories / Use Cases

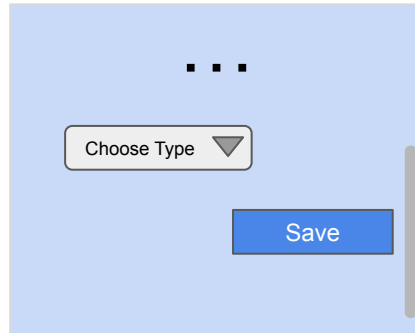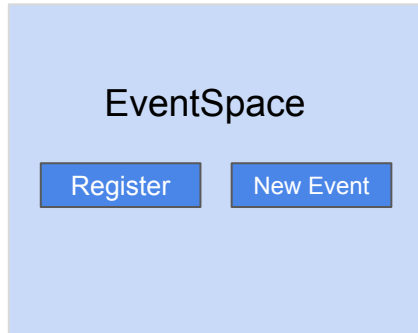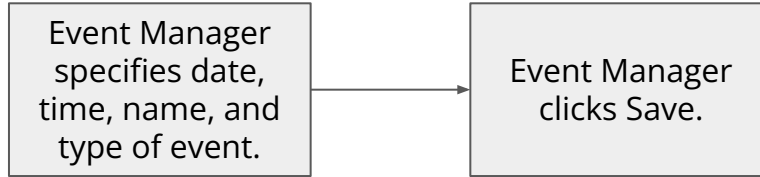*As a Event Manager, I would like to schedule events so that people can see which events are available to register for.*

Event Manager starts new event creation form.	→	Event Manager specifies date, time, name, and type of event.	→	Event Manager clicks Save.	→	Event Manager reviews event on Calendar view.

Atomic Design Components

Event Manager specifies date, time, name, and type of event. → Event Manager clicks Save.

New Event

Name

Date   Time

Choose Type

Save

Date

YYYY-MM-DD

< December 2021 >

M	T	W	T	F	S	S
28	29	30	31	1	2	3
4	5	6	7	**8**	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Time

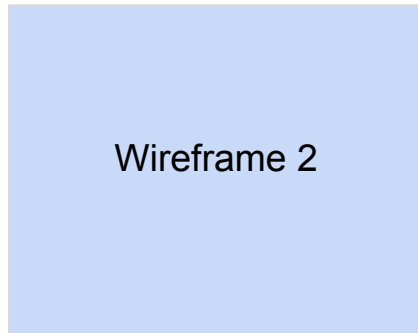| 4	▲	
5	30	
6	45	
7	00	AM
8	15	PM
9	30	
10	45	
11	00	
12	▼	

✓   ✕

Choose Type
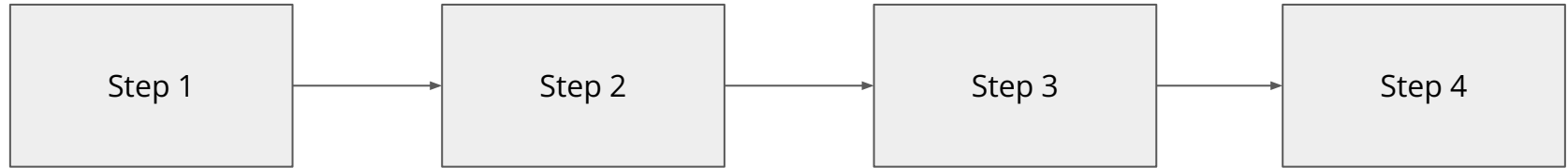
Consult

Check-in

Planning

Product and Design Specifications

As a [persona], I would like to [action] so that [benefit].

Step 1	→	Step 2	→	Step 3	→	Step 4

Wireframe 1	Wireframe 2	Wireframe 3	Wireframe 4

User Flow Specifications
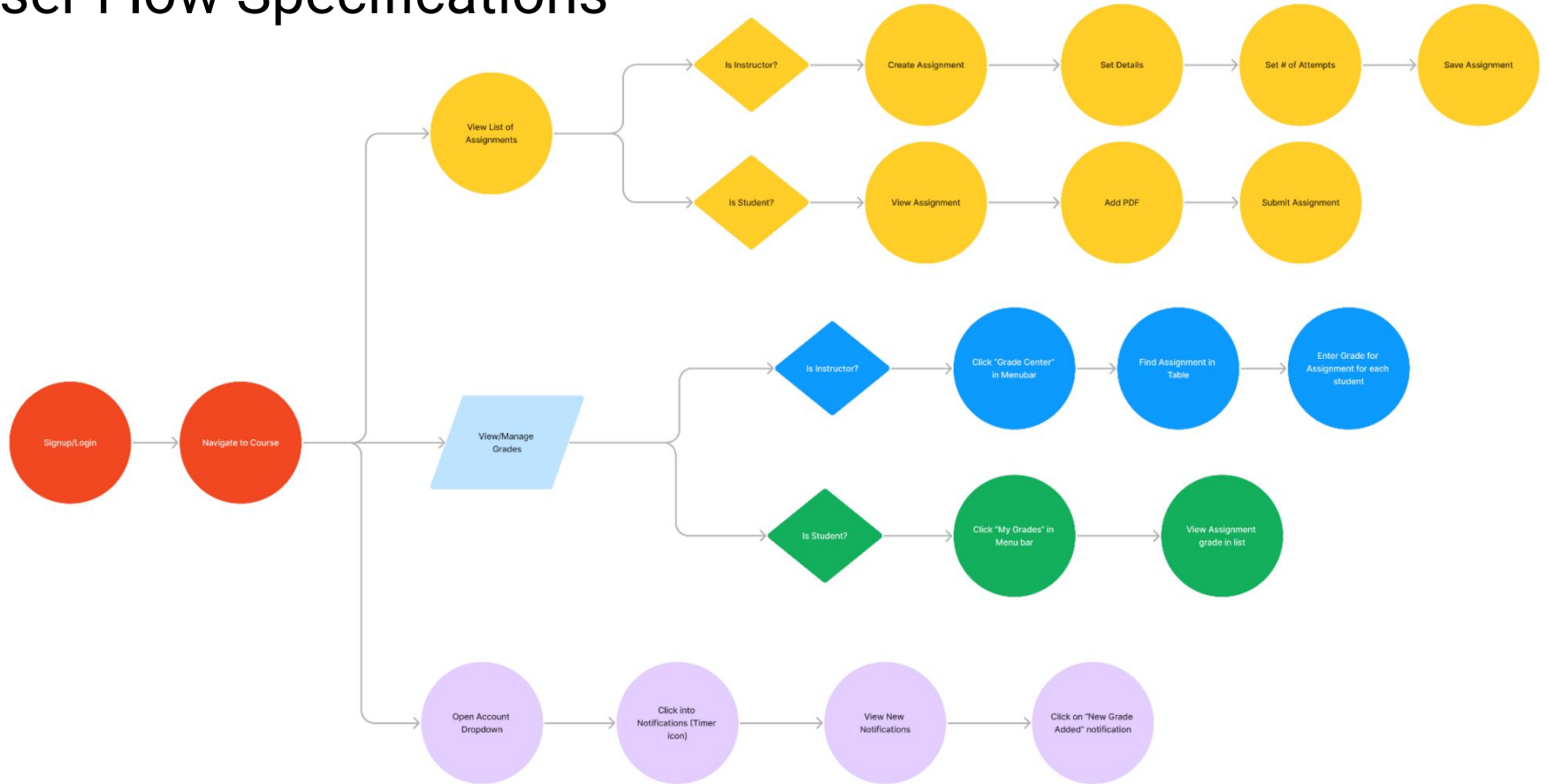
UX Design for APIs

As a Event Manager, I would like to schedule events so that people can see which events are available to register for.
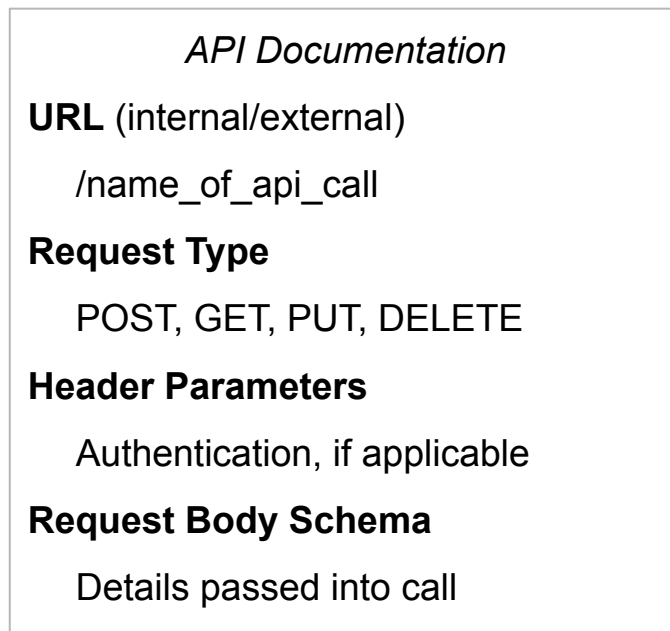

Event Manager starts new event creation form.	**POST** http://eventSpace.api.com/new_event
	
	{
	    "name": "Smith Engagement Party Planning",
	    "type": "Planning",
	    "start":{
	        "date": "2015-05-28",
	        "time": "09:00:00-07:00"
	    }
	    "end":{
	        "date": "2015-05-28",
	        "time": "17:00:00-07:00"
	    }
	}

Event Manager specifies date, time, name, and type of event.

Event Manager clicks Save.

Event Manager reviews event on Calendar view.

GET http://eventSpace.api.com/calendar_events

UX Design for APIs

As a [persona], I would like to [action] so that [benefit].

```
Step 1

    ↓

Step 2

    ↓

Step 3
```

API Documentation

URL (internal/external)

/name_of_api_call

Request Type

POST, GET, PUT, DELETE

Header Parameters

Authentication, if applicable

Request Body Schema
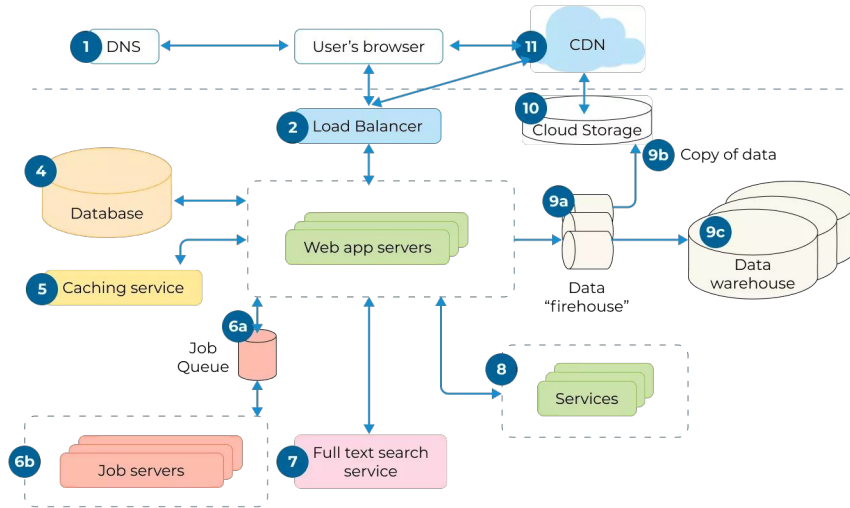
Details passed into call
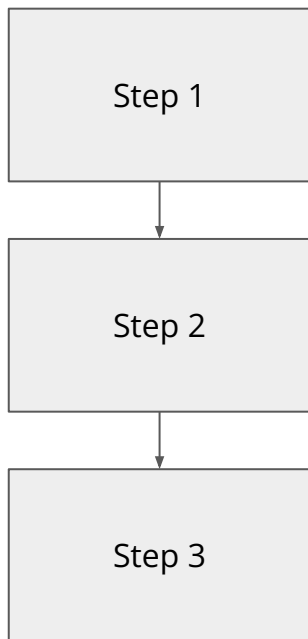
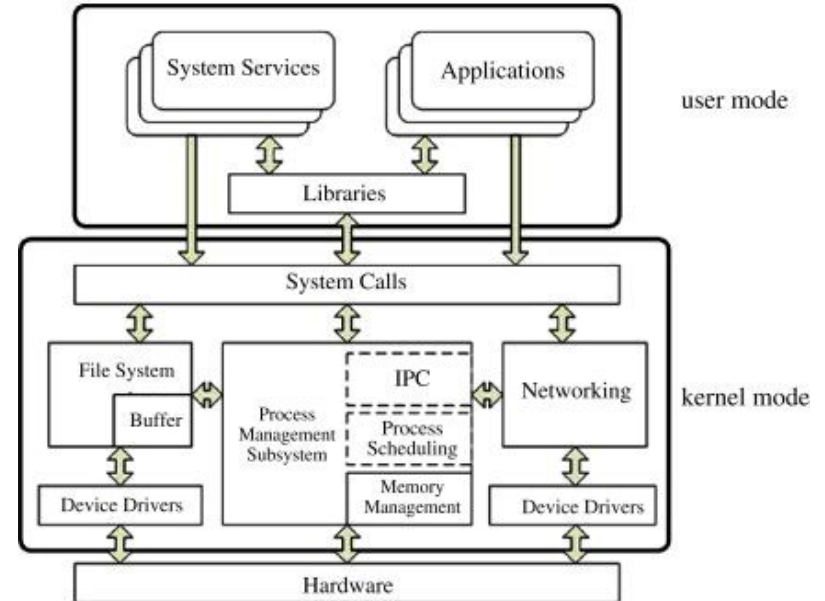# Architecture/System Design

- Diagrams
- Internal/External APIs and Frameworks
- Main Algorithms of System

Architecture/System Design

● Diagrams

Web App Architecture

Linux Architecture

Diagram tools

- LucidChart
- Excalidraw
- Draw.io

Architecture/System Design

- Internal/External APIs and Framework details

Twitter API

Goal:

Get all tweets to
generate sentiment
analysis

Endpoints Used:

GET
/tweet/sentiment/{happy}

Internal Backend API

Goal:

Register new user for
service

Endpoints Used:

POST
/register/

```
{
    first_name: XXX
    last_name: YYY
    …
}
```

AWS

Goal:

Serves as a hosting
platform for the
application to run

Details to note:

- Instance
- Credentials
- Etc.

Architecture/System Design

- Main Algorithms of System

Sentiment Classification Algorithm (ML)

Goal:
Train system on the 10,000 tweets generated by the samplings pulled from Twitter's API to classify sentiment.

Description:
First starting with traditional nlp techniques + xgboost for classification. Measure performance against precision, recall, accuracy, latency.

[Name] Algorithm

Goal:
[Purpose of this algorithm for the system]

Description:
Detailed description of how this algorithm works within the project

Writing 3 Requirements

Product / Design Specs

For Frontend UI projects: Product and Design Specs Slide
For Developer projects: UX Design for APIs

- User Stories
- User Flows
- Wireframes/Mockups/API Docs

Technical Specs (eg. System Design Slide)

- Architecture Diagrams
- External APIs, Internal APIs, and Frameworks
- Main Algorithms of System

Writing 3 - Product and Design Specs

For Frontend UI projects: [Product and Design Specs Slide](#)
For Developer projects: [UX Design for APIs](#)

- User Story 1 [*Bulleted list format*]

    User Flows for User Story 1 [*Flow Chart format*]

    Wireframes/Mockups/API docs for User Story 1 [*Image or API docs format*]

… repeat for most critical user stories in your project

Writing 3 - Technical Specs

Much of this can be reused from the Technical Summaries from Writing 2

- Architecture Diagrams [*Image format*]

  Diagram Examples

- List of integrated APIs and Frameworks, with sub-bullets detailing the Goals and Endpoints used [*Bulleted list format*]

  APIs and Frameworks

- Main Algorithms of System detailed with the description of how it was used and built within your project [*Paragraph format*]

  Algorithmic Details

Reminders

Mentor Meeting

- November Sprint Planning

Deliverables & Due Dates

- Writing 2: 10/29 (this Sunday!)
- Presentation 2: 11/8
- Writing 3: 11/19

November Sprint Planning + Alpha Demo

November Sprint Planning: **Code complete for Alpha Demo**

- Retro
  - Take time to discuss how the past sprint went and changes you want to make going into the next sprint
- Card Commitment
  - Decide on which cards each team member will commit to in the next sprint. Be sure to scope tickets appropriately & add subtasks if needed.
- Alpha Demo Plan
  - Scope your Alpha Demo and create the necessary cards to reach the demo by the beginning of December