scientist.com

Turn that Big Challenge into Big Growth

Lea Ann Bradford (She/Her), Senior Software Engineer II

Presentation Overview

1. Breaking Down Projects

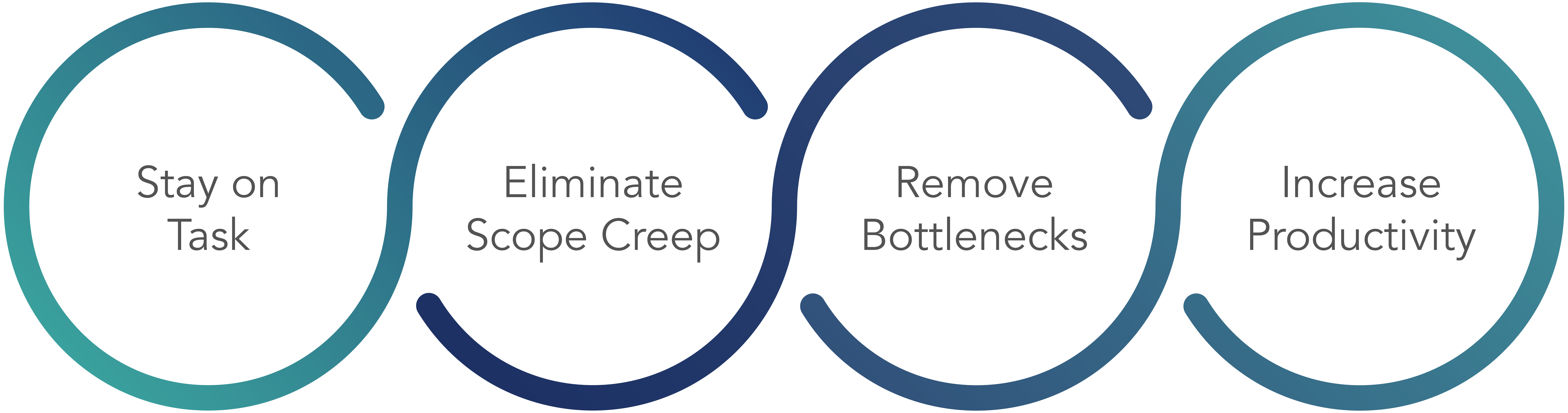
Strategies for turning ideas into actionable tasks

2. Additional Considerations

Timelines, Milestones, Dependencies

Breaking down a project

Breaking Down Projects is Important



Stay on Task

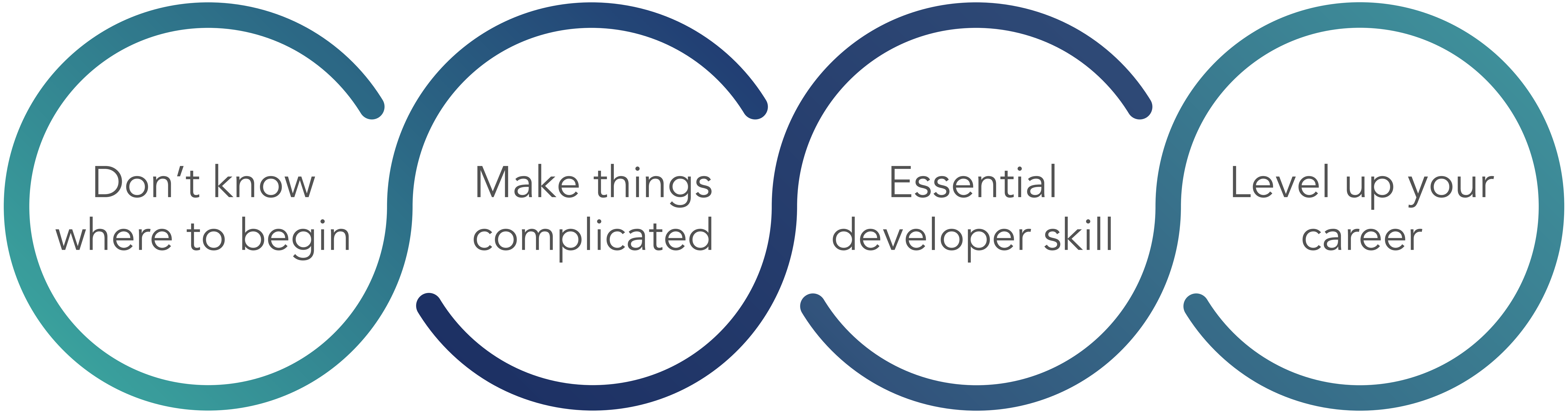
Eliminate Scope Creep

Remove Bottlenecks

Increase Productivity

It is essential for finishing projects in a timely manner

Breaking Down Tasks is Important



It is arguable the most essential skill for software engineers

Strategies

Top Down Approach

Start big and whittle items down

Categorize It

Look for patterns as you break items down

Top-Down Approach

Step 1

Identify overarching pieces and list them, perhaps on a whiteboard

Step 2

Narrow down pieces and keep going until it wouldn’t be reasonable to continue

Step 3

Tasks should begin with a single sentence that will become the title of the issues

Step 4

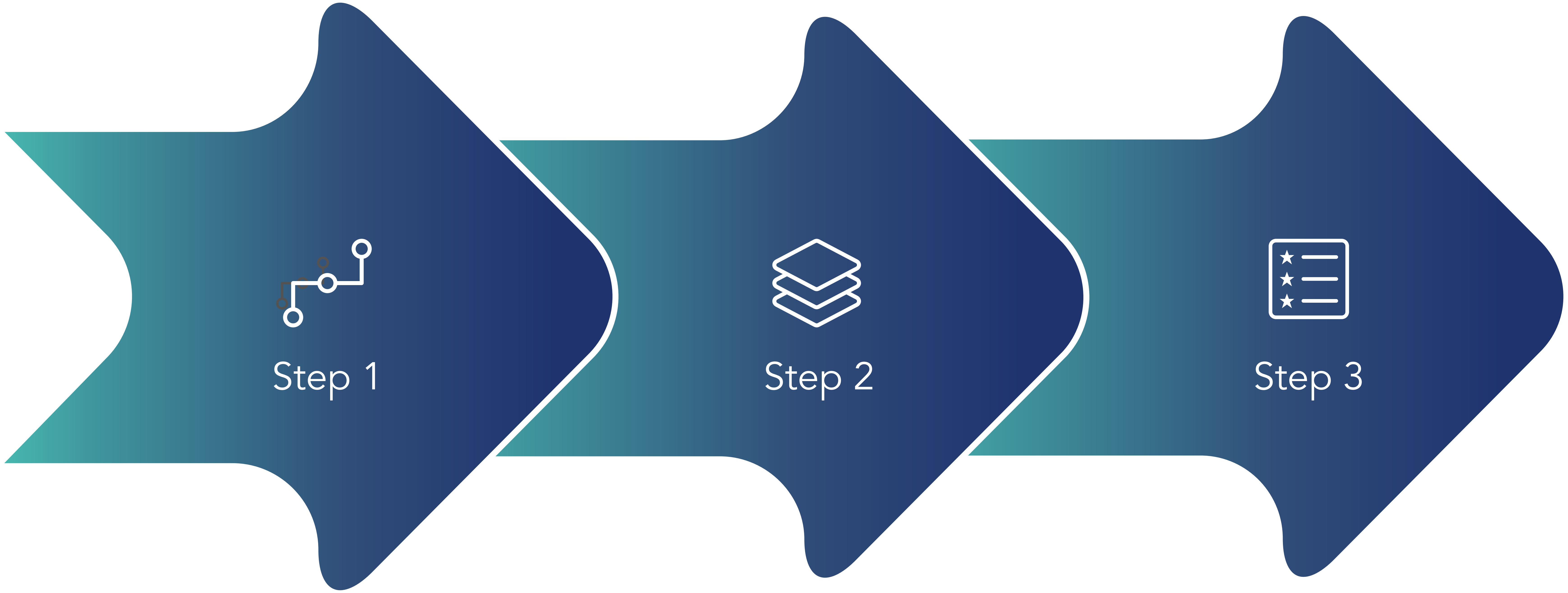
Break down the task - Spike tickets and deciding how it should be done

Step 5

Break down the task - decision documents, architecture decision records, team expertise

We are not thinking about solving the problems, only about what the tasks are

Categorizing



Step 1

Look for patterns
to emerge

Step 2

Group clusters of
work together

Step 3

Create single sentence
summary of tasks

Which steps need to be performed in order for the project to be finished?

Completing Issues

Making sure the issues have the right information

How to do it?

Does the approach rely on or affect other tickets?

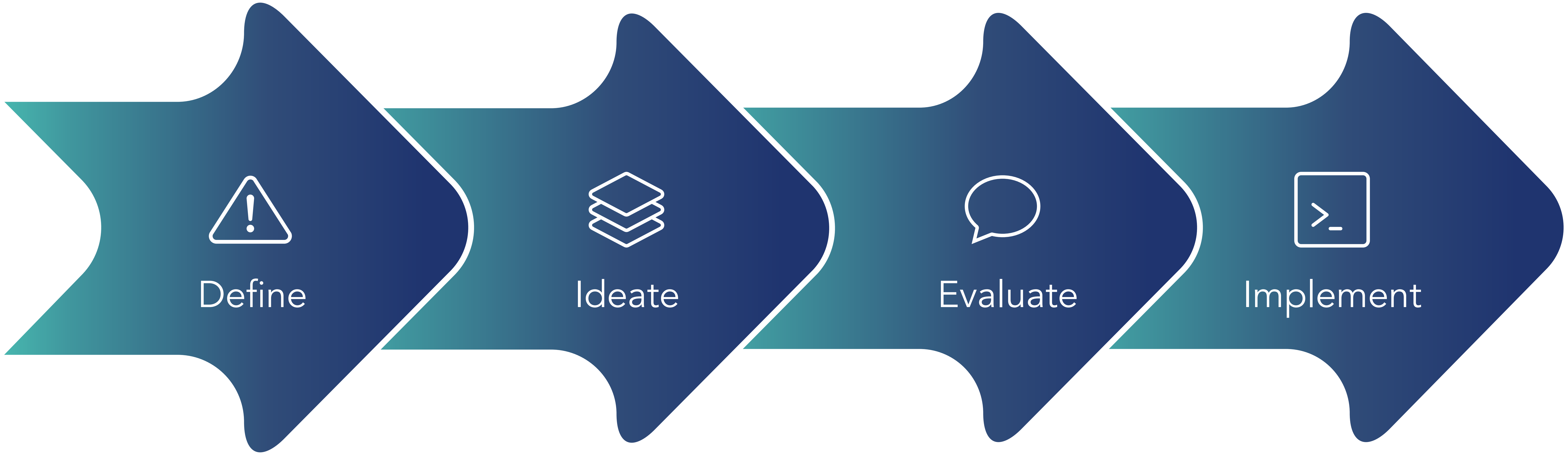
Uncertainty?

Placeholder or Spike tickets

Early involvement

Engineers should be involved from the start

General Task Breakdown



Define

Ideate

Evaluate

Implement

Define the problem

Come up with multiple solutions

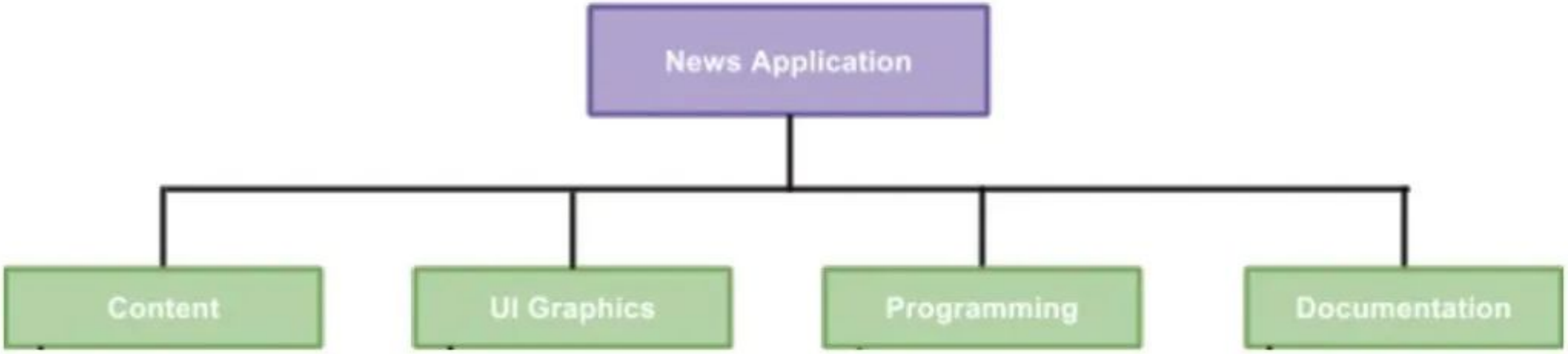
Evaluate solutions and pick one

Implement the solutions

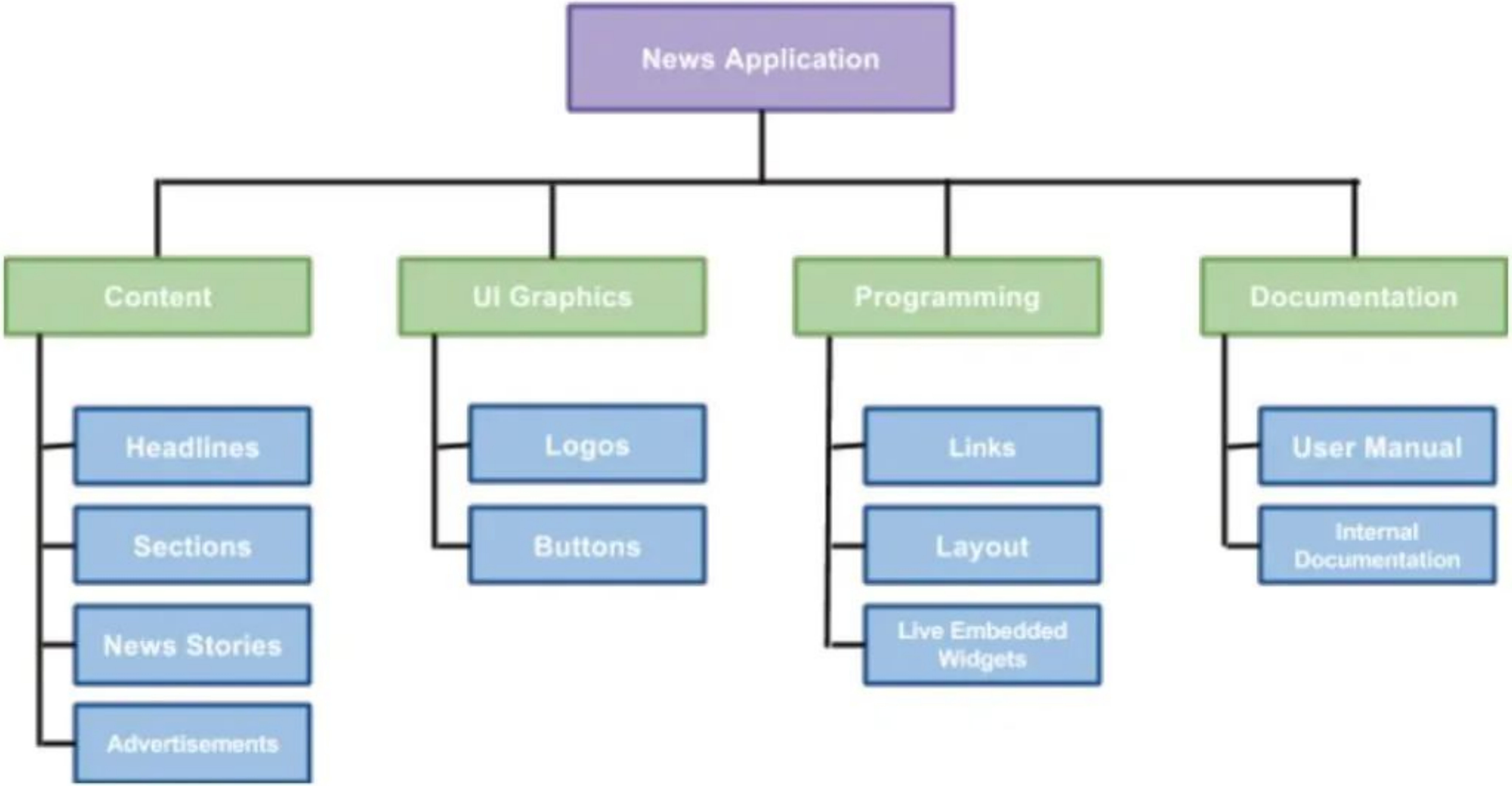
Basic steps to solve problems

Let’s See an Example

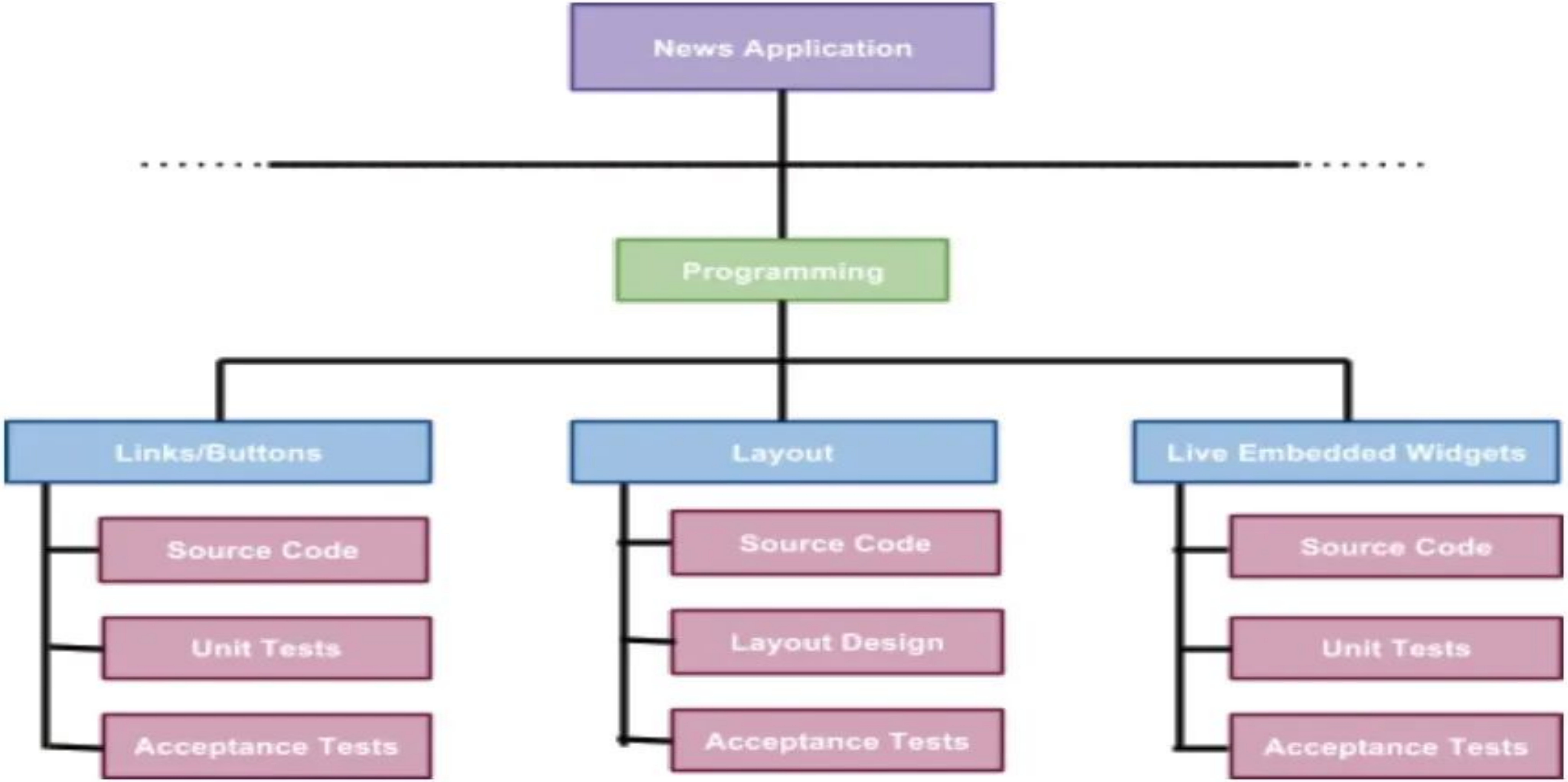
Large Sweeping Categories



Filling in More Details



Breaking Down a Section



News Application

Programming

Links/Buttons

- Source Code
- Unit Tests
- Acceptance Tests

Layout

- Source Code
- Layout Design
- Acceptance Tests

Live Embedded Widgets

- Source Code
- Unit Tests
- Acceptance Tests

Timelines
Milestones
Dependencies

Dependencies

Which tasks are blocking

Highest priority tasks unblock other tickets

Batch together clusters of work

Work has a general order in which it needs to be completed

Parallelize tasks

You can work simultaneously on pieces that are independent of each other

Milestones

How Long Will it Take?

Create your milestones based on length of the project

Monthly ◇ Weekly ◇ Sprint

Integrate milestones with your team's cadence

Phases ◇ Categories ◇ Parts

What should your milestones be?

Timelines

How Long Will it Take?

The number and size of your tickets

Can Work Be Parallelized?

Group tasks together that can be distributed

Allow For Changes

Feedback, quality assurance and bug fixes

Final Thoughts

Thank you!