

Product Development All Hands

September 2022

Presented by Sarah Randolph

One Team, many roles

- The Product Development org
- Working in squads
- Segmented by product domains allowing for specialization and expertise
- Allowing us to scale faster and more efficiently at reasonable working hours



How we win

In our work we need to be

- Collaborative
- Curious
- Celebratory

In our products we need to

- Simplify
- Scale
- Ship value



Our Strategy

- Product
- Technology
- Design

The 2022 VTFM

1

Grow and deepen our integrations and depth into the development process

- Integrates with LMS, SSO & Risk Management systems
- Integrates with Developer tools

2

Grow capabilities and offer new Pricing & Packaging offerings

- 1H address a new customer segment
- Launch Product Offering Packages for Different Customer Segments
- Launch offering for Companies to benchmark themselves against peers
- Launch a revamped trial in U.S. Public Sector

3

Enhance ways of working

- Establish best in class Employee Value Proposition
- Improving X-functional collaboration
- Become a data driven organisation

4

Address pain points and improve product experiences

- Customers can onboard without CS
- Improve the Platform experience (Quality, Performance, Usability, Accessibility)

Product Strategy Focus Areas

In 2022, our roadmap balances increased renewal risk, increased competition, strategic bets for the future and the need to pay down technical debt to enable a better UX and to increase efficiency

Types of Work

BAU/Fixes/Debt

Enhancements

New Capabilities

Outcomes

Internal Value

Tablestakes

Differentiated Value

Phases of Work

Discovery

Delivery

Optimization

Q2/3 Intended Customer-Facing Outcomes

Grow Stronger



Shipped

- More content

Coming in Q3

- New Product Trial
- GitHub & Jira integrations enhancements

Next

- Company Admin Onboarding
- Automated Course Creation

Increase Renewals

Shipped

- More inclusive language in content

Coming in Q3

- Accessibility and design uplifts
- Automated Translation, Challenge Descriptions

Next

- Incremental value on the above
- Improved experiences

Crush the Competition



Shipped

- New Pricing & Packaging offering GA
- NEW! Extra Challenges

Coming in Q3

- Live Lessons GA
- Storyline Content

Next

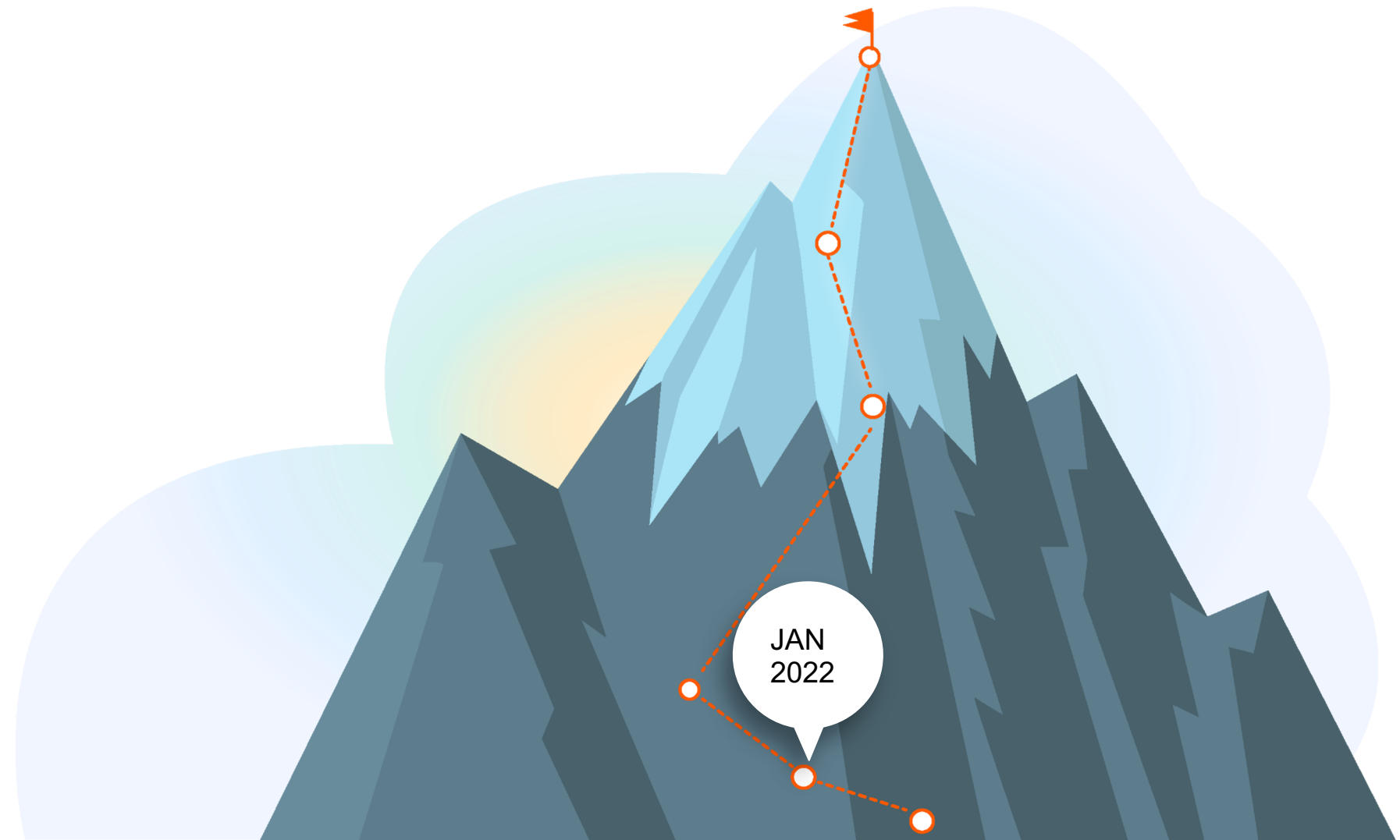
- Extra missions and walkthroughs GA

Design & UX Strategy

2022 and beyond

Where we started (2022)

- One designer for the entire organization
- Limited resources / availability / abilities
- Limited tooling
- Limited deliverables
- Limited ability to measure data
- Manual usability testing



Where we're at in our journey - 6 levels of design maturity



Where we're at now

- UX Maturity - Still at the beginning of our journey
- Building out team, hiring for strategic domains
- Creating Processes & improving ways of working
- Improved tooling
 - CS for vector graphics
 - Usertesting
- Ways we're already collaborating



Where we're at in our journey - levels of design maturity



Our Growing Team

- Creating New Structure & Processes
- Ways of working together
- Our Roles
- NEW! EMEA Team
- NEW! APAC Team
- NEW! USA Team
- 4 Pillars of Design

Now hiring...

Product Design Manager

Product Designer EMEA

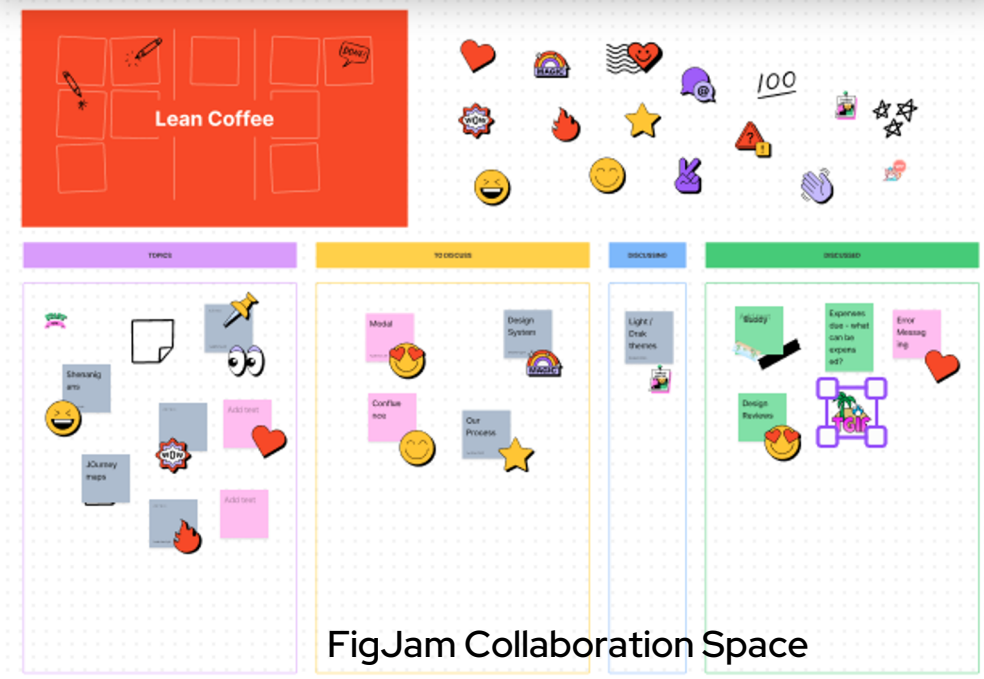
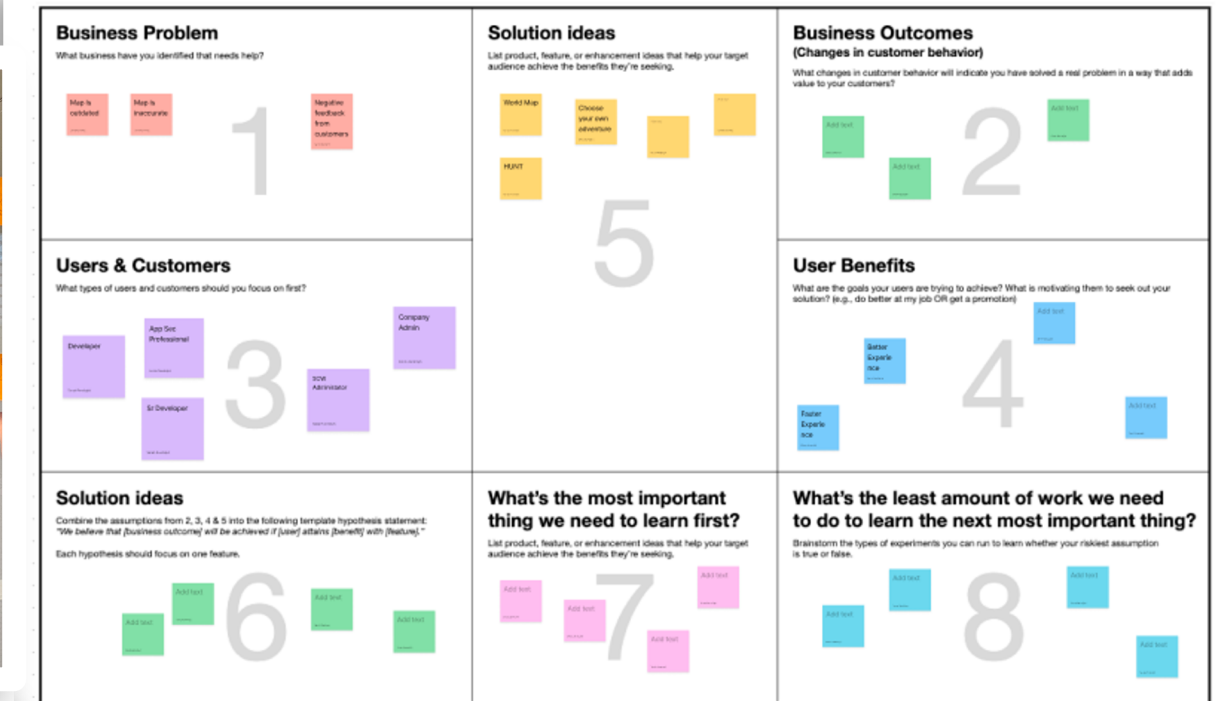
Ways we're already collaborating

- Workshops
- Jamboard
- FigJam
- Miro / Mural
- Squads & Teams
- LUMA

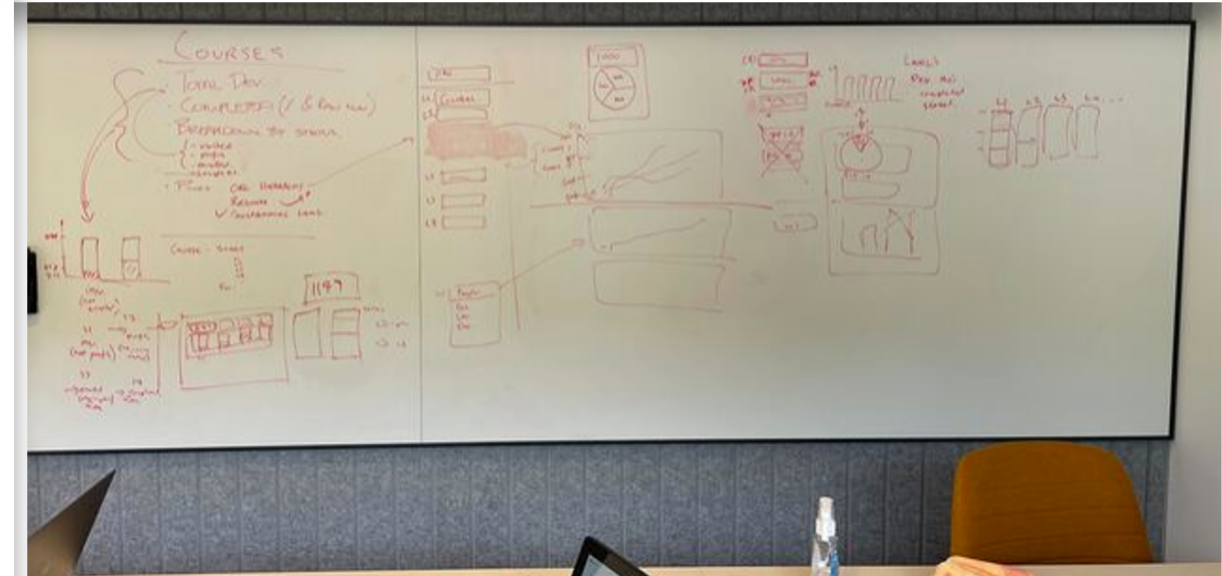
Bruges Office / LCL Workshop



FigJam UX Lean Canvas



FigJam Collaboration Space



Reporting / Discovery Dashboard Workshop

LUMA

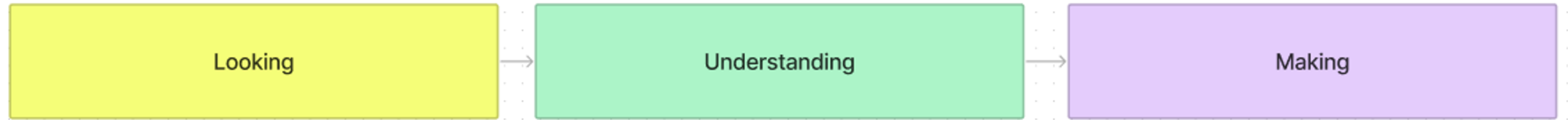
Looking
Understanding
Making

Human Centered
Design framework
for innovation and
collaboration

LUMA Workplace™
Methods and recipes

Use template now

Your digital whiteboard account



Ethnographic	Participatory Research	Evaluative Research	People & Systems	Patterns & Priorities	Problem Framing	Concept Ideation	Modeling & Prototyping	Design Rationale
Interviewing	What's on your radar?	Think out loud testing	Stakeholder Mapping	Affinity Clustering	Problem Tree Analysis	Thumbnail Sketching	Storyboarding	Concept Poster
Fly on the Wall Observation	Buy a feature	Heuristic Review	Personas	Bulls Eye Diagramming	Statement Starters	Creative Matrix	Schematic Diagramming	Video Scenario
Contextual Inquiry	build your own	Critique	Experience Diagramming	Importance / Difficulty Matrix	Abstract Laddering	Round Robin	Rough & Ready Prototyping	Cover Story Mock-up
Walk a mile emersion	Journaling	SUS / SEQ Scale	Concept Mapping	Visualize the Vote	Rose / thorn / bud	Alternative Worlds	Appearance Modeling	Quick reference guide

Improved Tooling - UserTesting.com

Scenario

You'll be reviewing an early concept for a product that offers a learning experience for secure coding. We are not testing your coding skills or abilities. This is not intended to be in its final state, but feel free to give design feedback along the way. We want to learn about your expectations for the product, whether it works as expected, whether you encountered any errors, whether you can complete the tasks, and what we should name it. Please talk out loud and share your thoughts.

Start

3 Success task

Did you complete the task successfully?

- Yes, I'm sure I completed the entire task successfully. (9)
- No, The task took too long that I gave up. (2)

Watch task 3 2:34

4 Difficulty question

Overall, this task was:

Difficulty Level	Number of Contributions
1 Very Difficult	0
2	2
3	2
4	5
5 Very Easy	2

Watch task 4 5:18

[View usability tests](#)

← Back UserTesting video 4202992F - LCL Lessons A1, A3, A4 and C#...

Dashboard / LCL Lessons A1, A3, A4 and C# audience / Session 3

3 of 11

07:28

Steps

- 1 Simple authorization: AccountController
- 2 Simple authorization: AuthController
- 3 Role-based au...

Nicely done! Our next job is to fix the AuthController. As you can see, [AllowAnonymous] is set on controller level, which gives anyone access to these endpoints. Ideally, we'd want:

- The login GET and POST methods to be accessible to any user, authenticated or not.
- Make sure the 1 to authenticated

Positive

Good to know that allow them on and on the most exploratory or any other authorization attribute to get when you using this.

Is this correct?

```
[Route("Auth")]
public class AuthController : Controller
{
    private readonly SignInManager<User> _signInManager;
    private readonly UserManager<User> _userManager;

    public AuthController(UserManager<User> userManager, SignInManager<User> signInManager)
    {
        _userManager = userManager;
        _signInManager = signInManager;
    }

    [AllowAnonymous]
    [HttpGet]
    [Route("Login")]
    public async Task<ActionResult> Login()
    {
        if (await HttpContext.RequestServices.GetService<UserManager<User>>() != null)
        {
            return RedirectToAction("Index", "Account");
        }

        return View();
    }

    [Route("Login")]
    [HttpPost]
}
```

Ln 22, Col 21 Spaces: 4 UTF-8 LF C#

7:28 / 21:50

Clip

Report a problem

Design Organization Strategies

- Improve outcomes with better research

- Research Organization
- Prototyping
- User Experience Program
- Personas / User Journeys
- Innovation Sessions

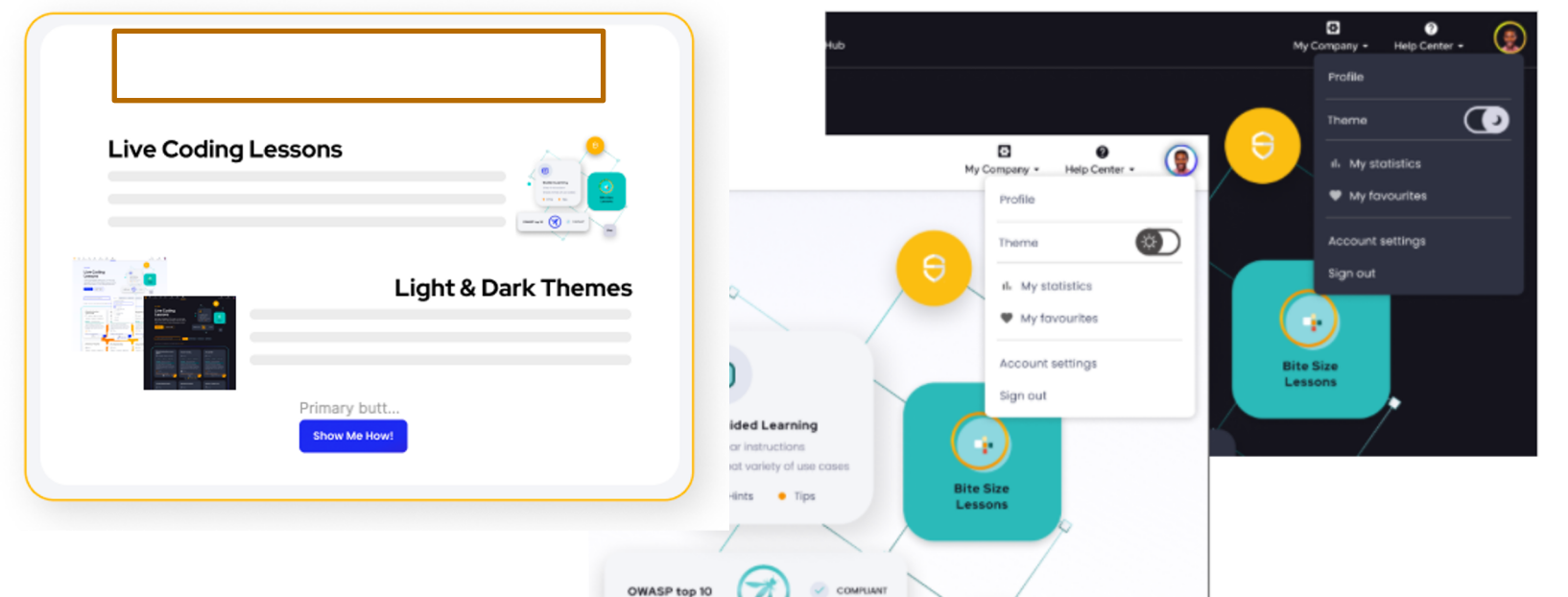
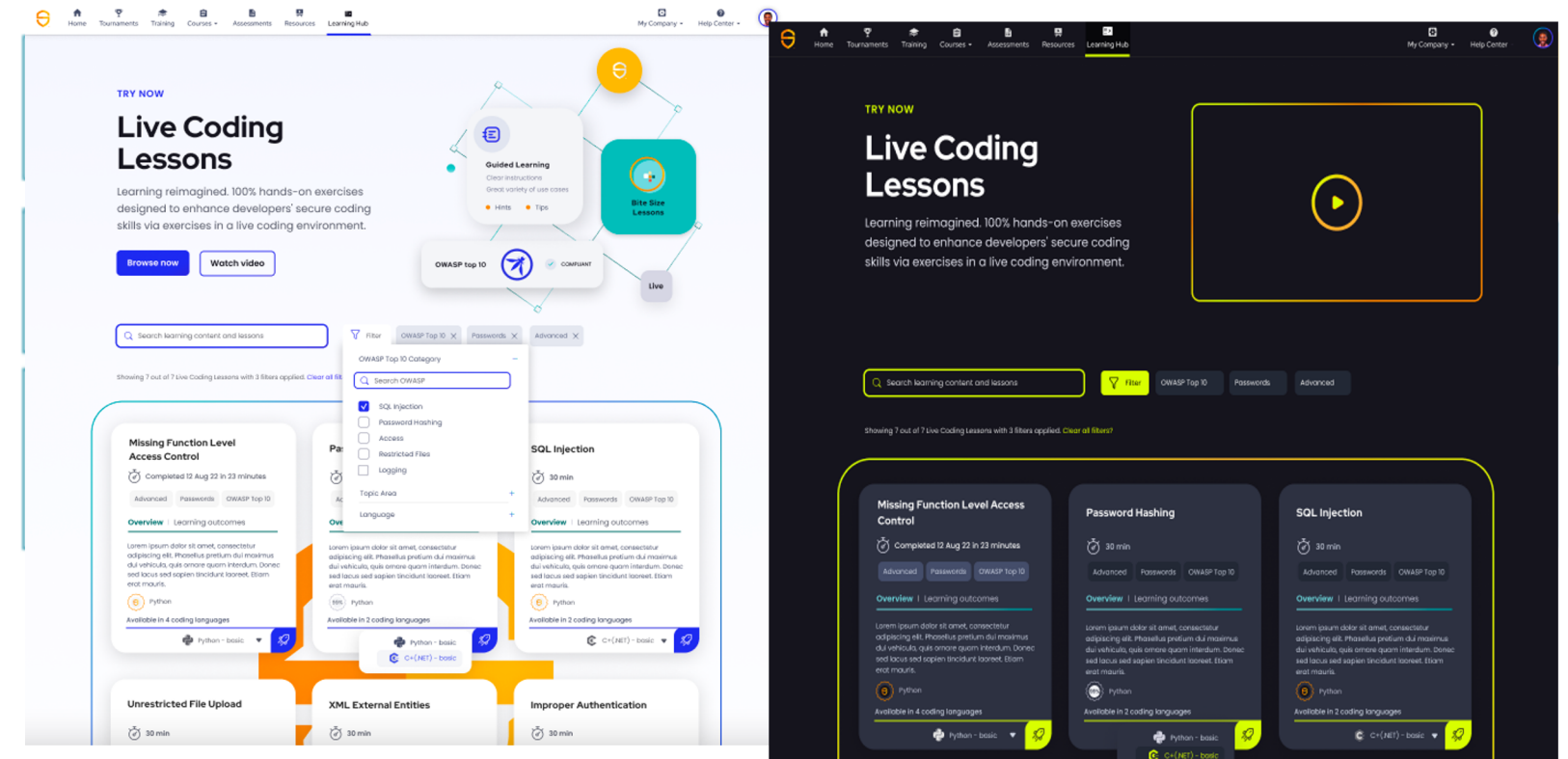
- Simplify UX and enable scale

- Design System
 - Remove friction
 - Accessibility
 - New Light / Dark / HC Themes

- Squad-driven storytelling and design work

- Live Lessons
- Project X
- Reporting

Our goal is to conform to the Web Content Accessibility Guidelines ("WCAG") 2.1 Level AA.

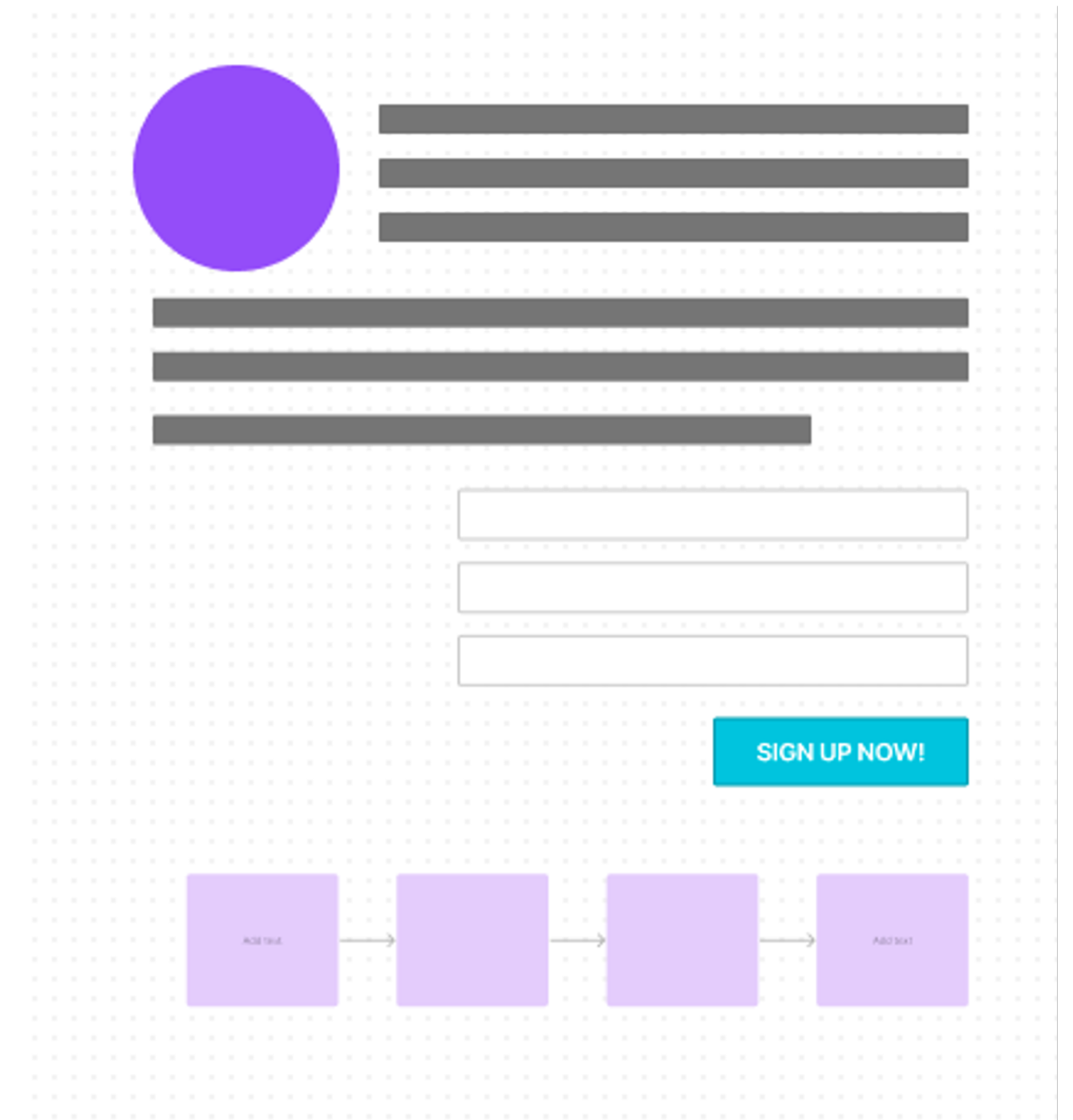


User Experience Program

What is it? Why do we need one?

Automating the workflow / process of recruitment, coordination, interviewing and analysing usability testing, as well as compensating participants.

- Have a database of users to test with at anytime
- Find users willing to give us feedback
- Interview users to better understand how they're using our product
- Identify pain points in the current experience of using our product
- Get regular feedback from engaged users
- Reduce the amount of time we spend in usability testing sessions
- Reduce the amount of time we spend recruiting test participants
- Streamline and automate the above processes as much as possible!

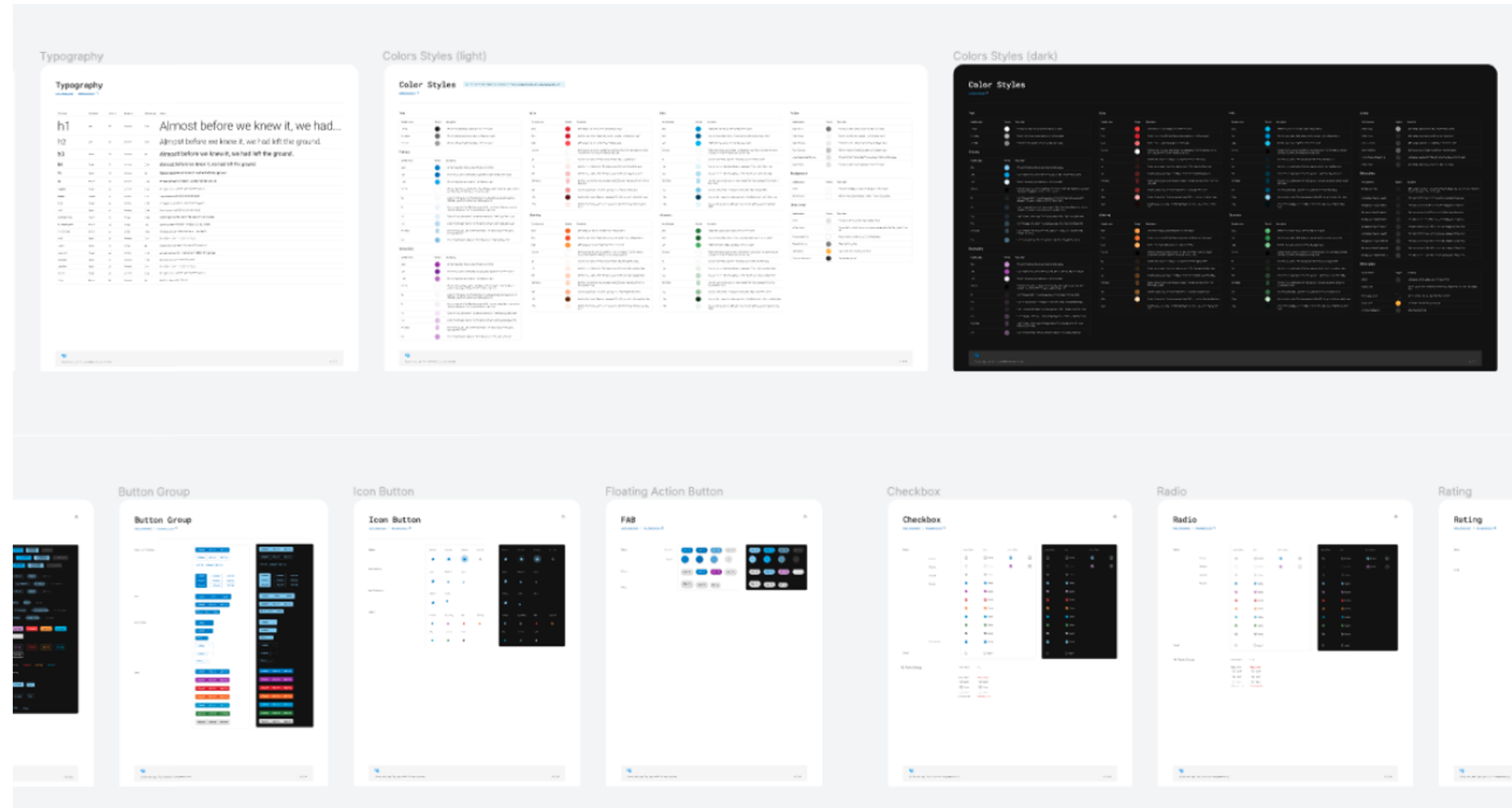


Design System - Figma

Stage One:
Build out design
system in Figma

Stage two:
Storybook UI

Stage three:
Build a public facing
design system.



Accessibility & Inclusivity

Putting our users first is at the heart of everything we do. We strive to create and provide an inclusive and accessible experience to everyone.

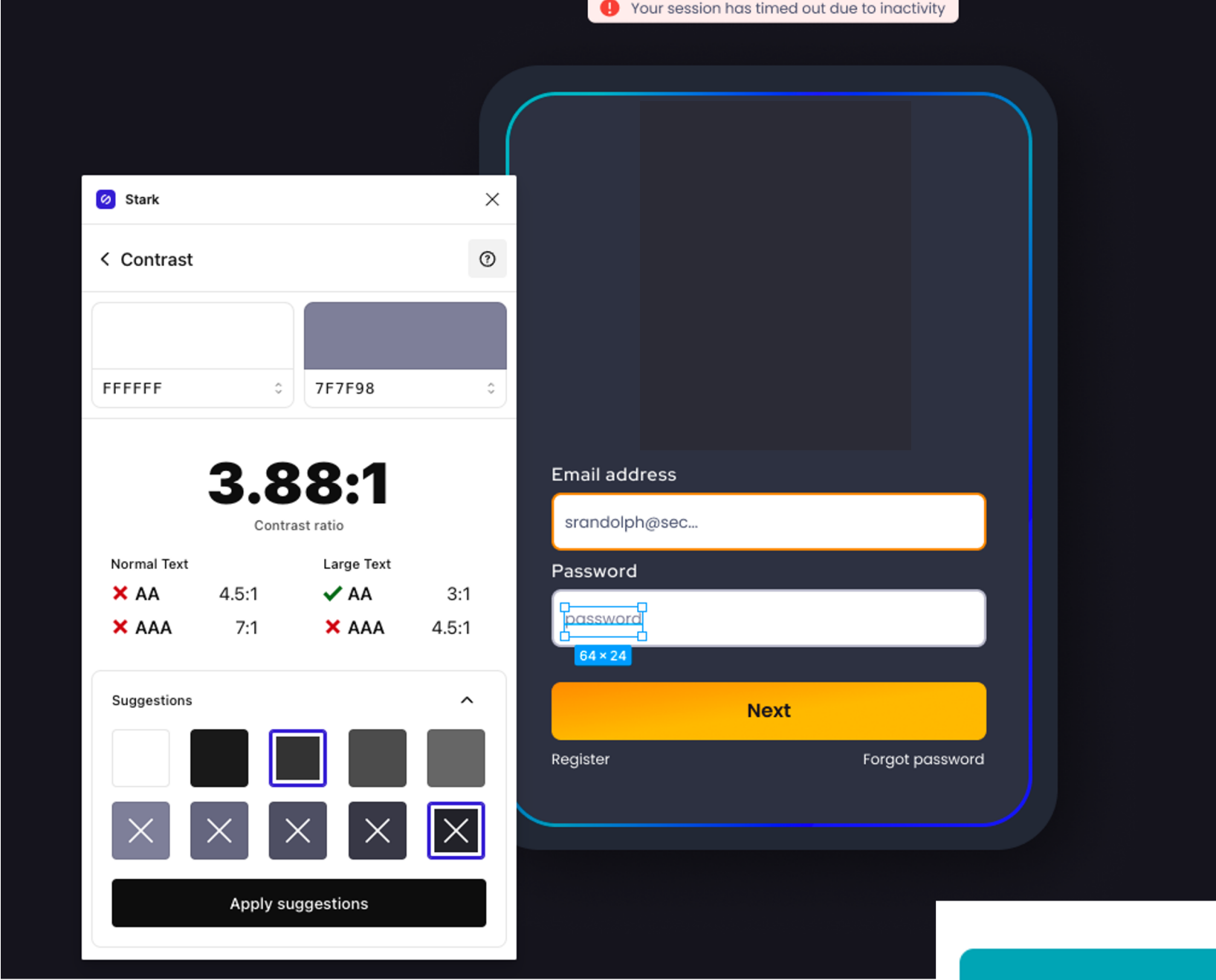
Our goal is to conform to the Web Content Accessibility Guidelines (“WCAG”) 2.1 Level AA.

High-level Accessibility strategy and plan DONE	Conducted interviews with developers with visual impairments DONE	Design System defining clear standard and principles for our products IN PROGRESS
Inclusive language on the platform DONE	Completed an automated audit on the platform DONE	Accessibility consultant and manual audit DONE
Initial themes and slices ready for collaboration with engineering DONE	Incorporated some customer led accessibility audits and feedback DONE	27 hours of accessibility training for full engineering team DONE

H1-22	<ul style="list-style-type: none">✓ Manual audit✓ Accessibility training for engineers
H2-22	<ul style="list-style-type: none">⦿ Color contrast issues⦿ Light / Dark / High Contrast themes⦿ Creation of a design system⦿ Challenge player● Fixing keyboard focus and navigation● Accessible forms
H1-23	<ul style="list-style-type: none">● Visual testing processes● Accessible comms● Event and element sequencing

WCAG are published by the World Wide Web Consortium (“W3C”) as a part of Web Accessibility Initiative (“WAI”), available at [w3.org/TR/WCAG/](https://www.w3.org/TR/WCAG/).

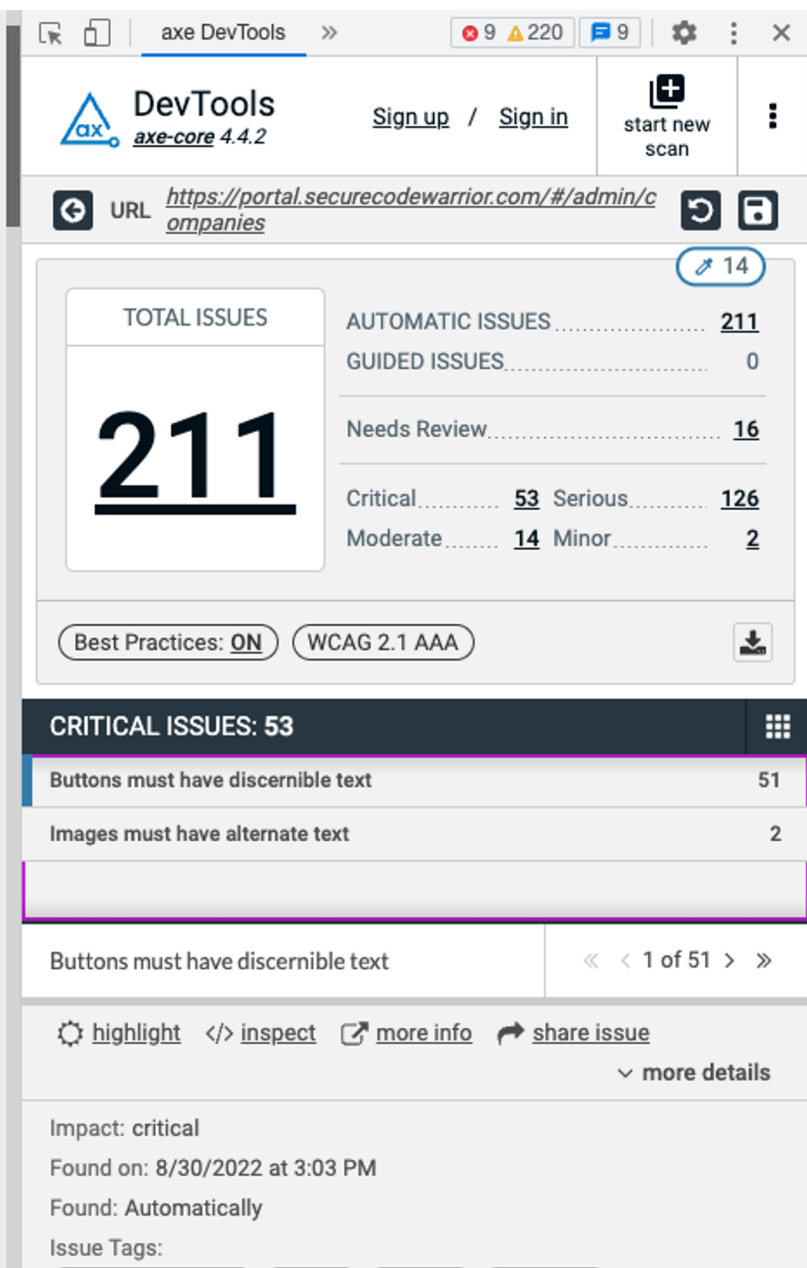
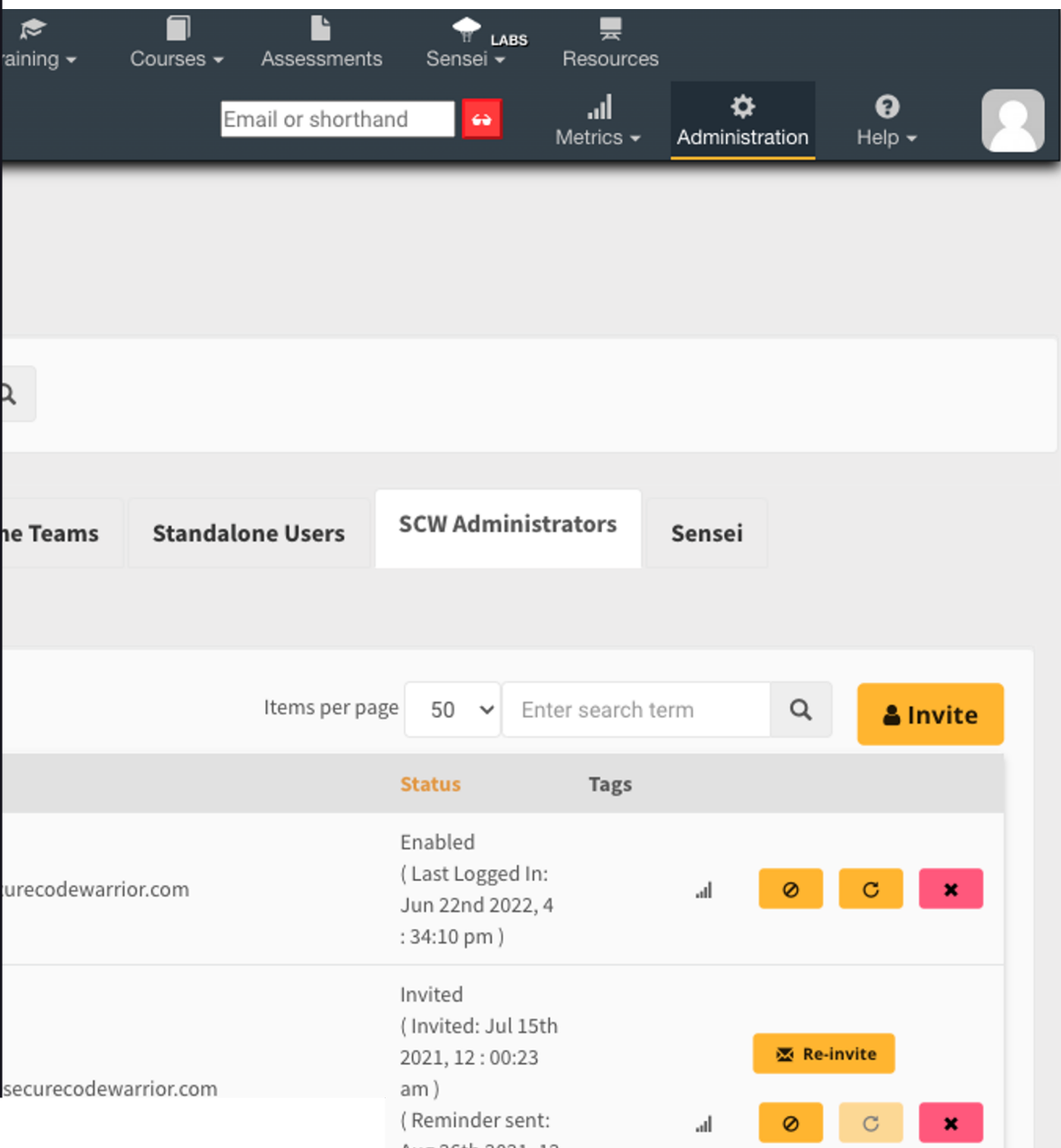
How we design for accessibility



Stark Pro Figma Plugin



[Button buddy](#)



Axe Dev Tools


New Customer Controlled Themes

Customer will be able to switch from their profile drop down in the nav.

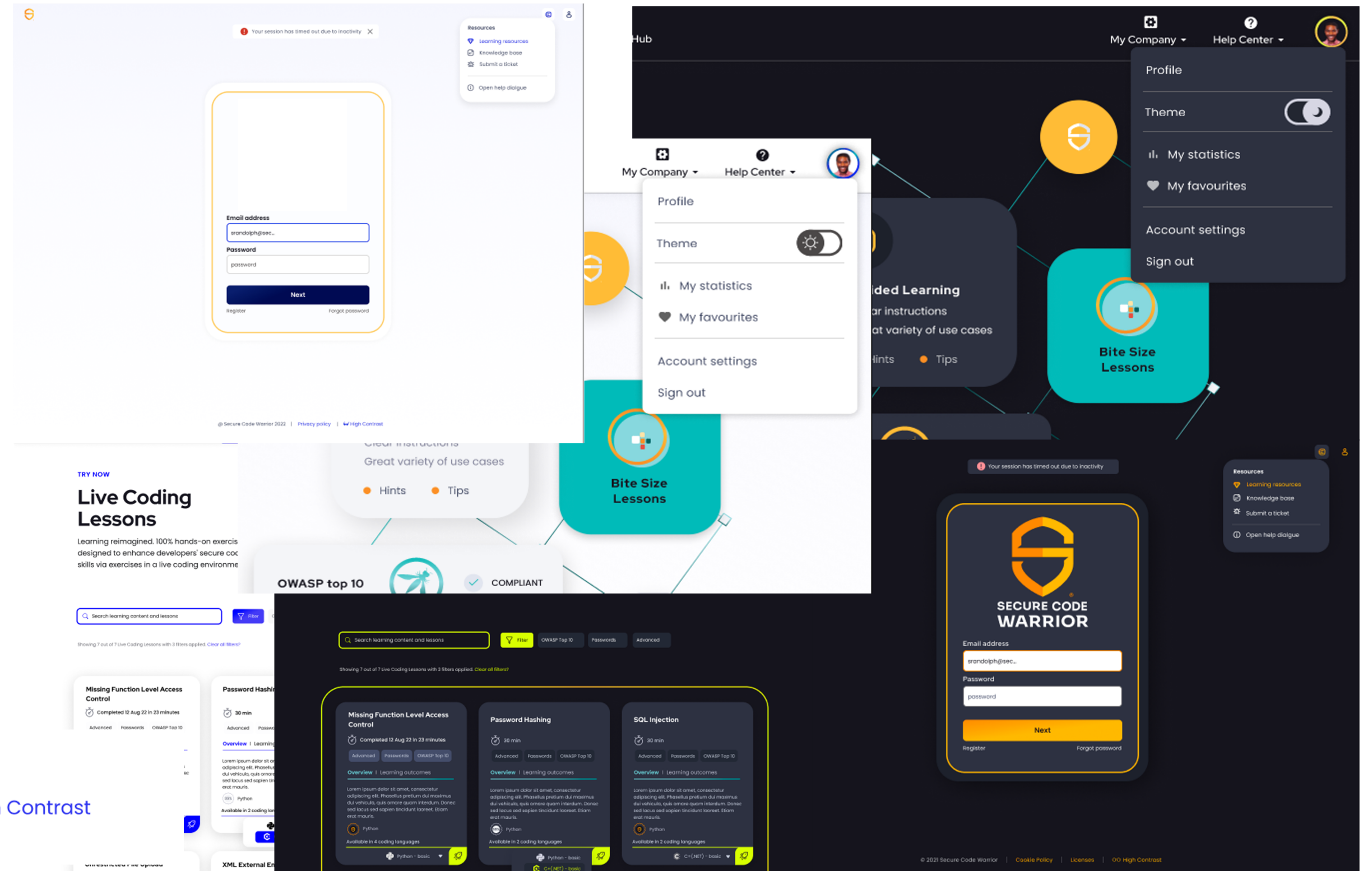
Like what you see?

On date, we will be launching new our new Light & Dark Themes

How it works



Show Me How!



Resources

- Learning resources
- Knowledge base
- Submit a ticket
- Open help dialogue

My Company Help Center

Profile

Theme

- My statistics
- My favourites

Account settings

Sign out

Secure Code Warrior 2022 | Privacy policy | High Contrast

TRY NOW

Great variety of use cases

- Hints
- Tips

Bite Size Lessons

OWASP top 10 COMPLIANT

Search learning content and lessons

Showing 7 out of 7 Live Coding Lessons with 3 filters applied. Clear all filters

Missing Function Level Access Control

Completed 12 Aug 22 in 23 minutes

Advanced Passwords OWASP Top 10

Overview | Learning outcomes

Python

Available in 2 coding languages

Password Hashing

30 min

Advanced Passwords OWASP Top 10

Overview | Learning outcomes

Python

Available in 2 coding languages

SQL Injection

30 min

Advanced Passwords OWASP Top 10

Overview | Learning outcomes

Python

Available in 2 coding languages

Secure Code Warrior

Email address

randolph@sec...

Password

password

Next

Register Forget password

Your session has timed out due to inactivity

Resources

- Learning resources
- Knowledge base
- Submit a ticket
- Open help dialogue

My Company Help Center

Profile

Theme

- My statistics
- My favourites

Account settings

Sign out

© 2021 Secure Code Warrior | Cookie Policy | Licenses | High Contrast

Live Coding Lessons

Dashboard / LCL Lessons A1, A3, A4 and C# audience / Session 3

07:28

Steps

- Simple authorization: AccountController
- Simple authorization: AuthController

Nicely done! Our next job is to fix the AuthController. As you can see, [AllowAnonymous] is set on controller level, which gives anyone access to these endpoints. Ideally, we'd want:

- The Login GET and POST methods to be accessible to any user, authenticated or not.
- Make sure the Login POST method is accessible to any user, authenticated or not.

Positive

Good to know that allow them on and on the most exploratory or any other authorization attribute to get when you using this.

Role-based au

Is this correct?

```

1  [AllowAnonymous]
2  public class AuthController : Controller
3  {
4      private readonly SignInManager<User> _signInManager;
5      private readonly UserManager<User> _userManager;
6
7      public AuthController(UserManager<User> userManager, SignInManager<User> signInManager)
8      {
9          _userManager = userManager;
10         _signInManager = signInManager;
11     }
12
13     [AllowAnonymous]
14     [HttpGet]
15     [Route("Login")]
16     public async Task<ActionResult> Login()
17     {
18         if (await HttpContext.RequestServices.GetService<IUserManager<User>>().IsEmailConfirmedAsync(await HttpContext.RequestServices.GetService<IUserManager<User>>().FindByEmailAsync(HttpContext.Request.Query["email"])))
19         {
20             return RedirectToAction("Index", "Account");
21         }
22         return View();
23     }
24
25     [Route("Login")]
26     [HttpPost]
    
```

Report a problem

Lessons > A7-Improper authentication

INTRODUCTION

- ### Step 1: Implement a password check

The AuthController.cs, and let's implement the first layer of the authentication process (**Something you know**): a password check.

Task 1:
 .NET provides us with the **Identity** API to do the heavy lifting for us. How neat! In particular, its `SignInManager<User>`, which is injected in the constructor, will help us authenticate.

Task 2:
 Scroll to the `Login` POST method, and above the return statement:

Call the `PasswordSignInAsync` on the `_signInManager` instance.

 - Pass in 4 arguments:
 - the user's submitted email address,
 - the user's submitted password,
 - set `isPersistent` to false to prevent the login attempt to persist across multiple requests,
 - set `lockoutOnFailure` to true to lock out the user after too many incorrect password attempts.
 - As this is an `async` call, make sure to `await` it.

Reveal solution

Submit
- ### Step 2: Generate 2FA token
- ### Step 6: Execute the query

Outline

Timeline

master

Home Tournaments Training Courses Assessments Resources Learning Hub

My Company Help Center

TRY NOW

Live Coding Lessons

Learning reimagined. 100% hands-on exercises designed to enhance developers' secure coding skills via exercises in a live coding environment.

Browse now Watch video

Guided Learning
 Clear instructions
 Great variety of use cases
 Hints Tips

Bit Size Lessons

OWASP top 10 COMPLIANT

Live

Search learning content and lessons

Filter OWASP Top 10 Passwords Advanced

Showing 7 out of 7 Live Coding Lessons with 3 filters applied. Clear all filters

OWASP Top 10 Category

Search OWASP

- SQL Injection
- Password Hashing
- Access
- Restricted Files
- Logging

Topic Area

Language

Missing Function Level Access Control

Completed 12 Aug 22 in 23 minutes

Advanced Passwords OWASP Top 10

Overview Learning outcomes

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus pretium dui maximus dui vehicula, quis ornare quam interdum. Donec sed lacus sed sapien tincidunt laoreet. Etiam erat mauris.

Python

Available in 4 coding languages

Python - basic

SQL Injection

30 min

Advanced Passwords OWASP Top 10

Overview Learning outcomes

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus pretium dui maximus dui vehicula, quis ornare quam interdum. Donec sed lacus sed sapien tincidunt laoreet. Etiam erat mauris.

Python

Available in 2 coding languages

Python - basic

Unrestricted File Upload

30 min

Advanced Passwords OWASP Top 10

Overview Learning outcomes

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus pretium dui maximus dui vehicula, quis ornare quam interdum. Donec sed lacus sed sapien tincidunt laoreet. Etiam erat mauris.

Python

Available in 2 coding languages

Python - basic

XML External Entities

30 min

Advanced Passwords OWASP Top 10

Overview Learning outcomes

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus pretium dui maximus dui vehicula, quis ornare quam interdum. Donec sed lacus sed sapien tincidunt laoreet. Etiam erat mauris.

Python

Available in 2 coding languages

Python - basic

Improper Authentication

30 min

Advanced Passwords OWASP Top 10

Overview Learning outcomes

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus pretium dui maximus dui vehicula, quis ornare quam interdum. Donec sed lacus sed sapien tincidunt laoreet. Etiam erat mauris.

Python

Available in 2 coding languages

C#(.NET) - basic

Developer Challenges

Public facing challenges for developers in new dark theme

Accessible colors & fonts

Fixed: Alt tags

Fixed: Accessible forms

Fixed: keyboard focus and navigation

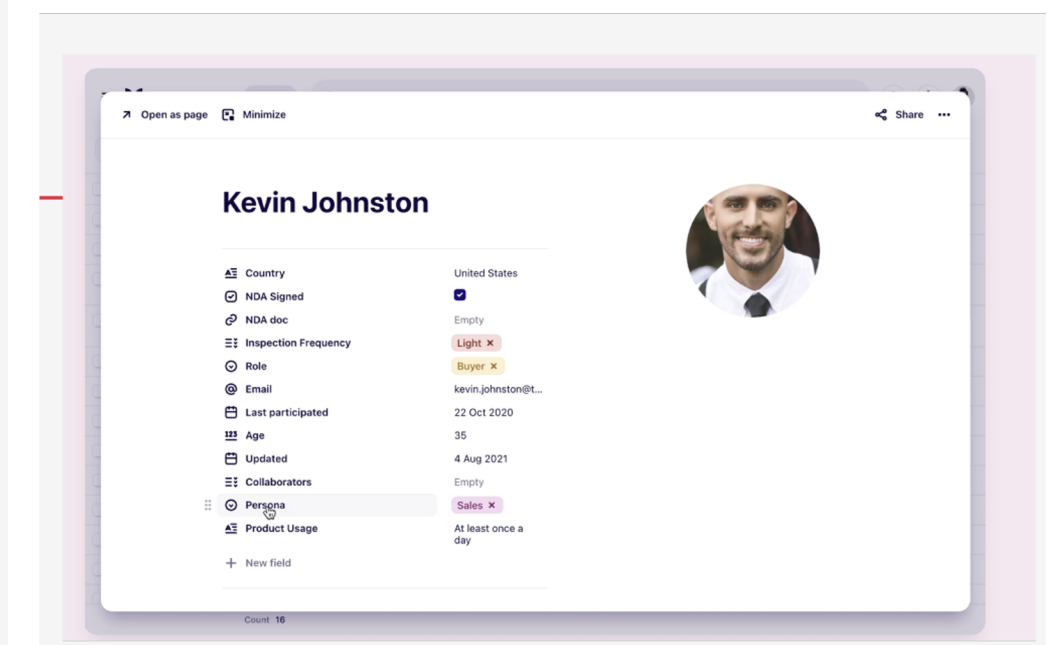
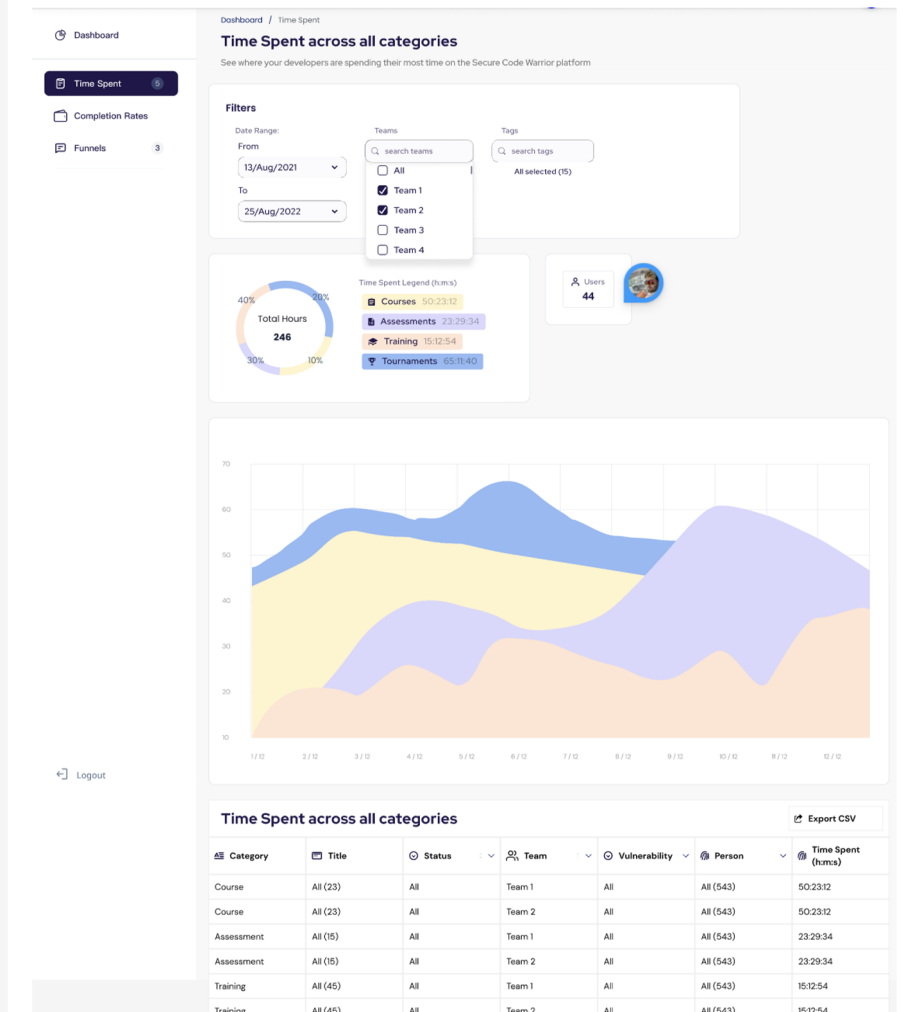
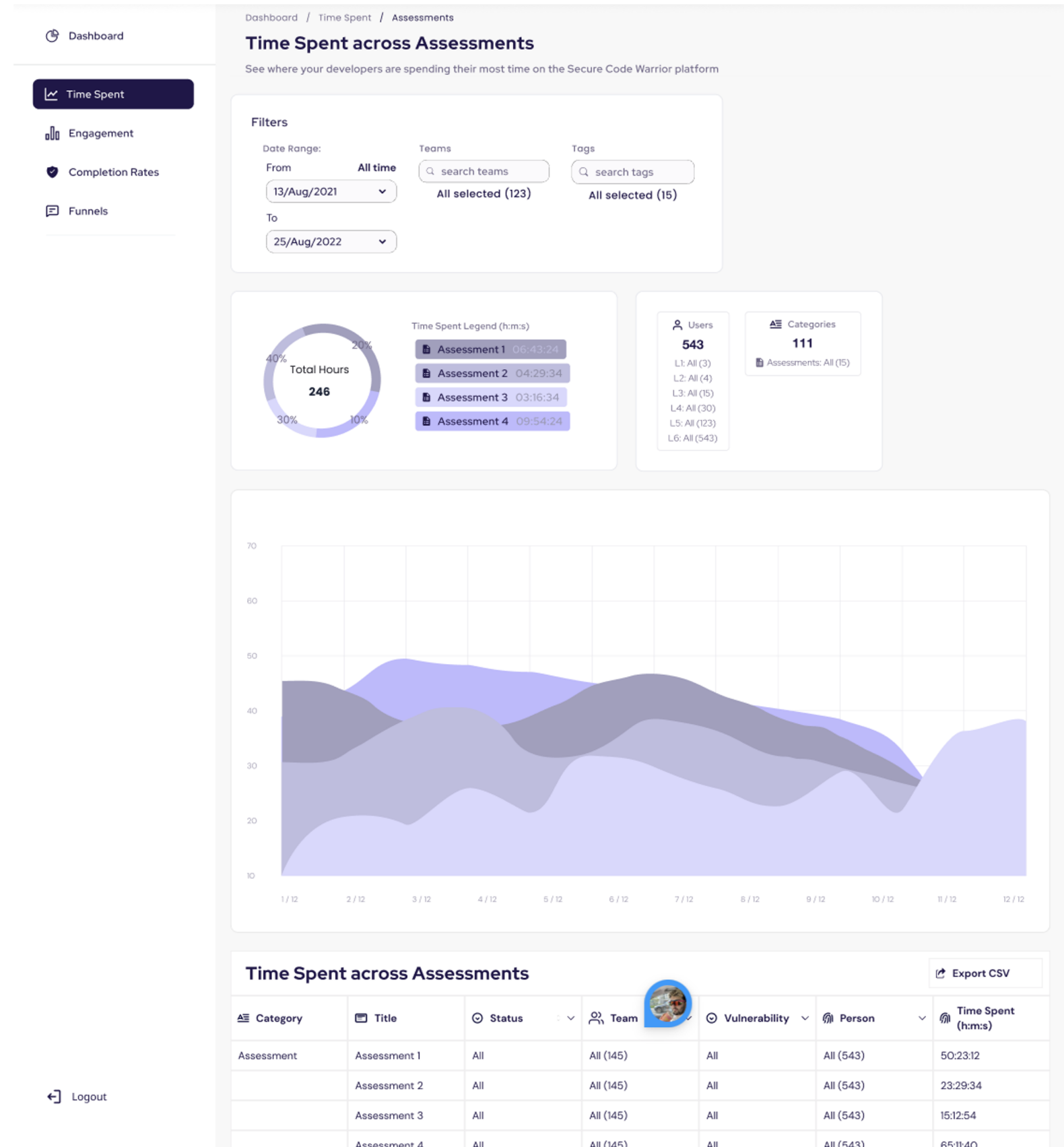
The screenshot shows the 'SCW Devlympics 2022 Tournament' page. It features a dark theme with a light blue trophy graphic. The main text reads: '2 Main Arenas to Play in. 34 Prizes to be Won Across 11 Competitions. Only 1 will be Proclaimed as the Ultimate Warrior.' Below this are four feature cards: 'Play in as many tournaments as you want', 'The tournament is open for 12 hours to cover global timezones', 'Devlympics will be streaming live on Twitch', and '1.5-2.5 hrs playing time'. At the bottom, there are two call-to-action boxes for 'Devlympics 2022 Ultimate Warrior' and 'Secure Coding Champion'.

The screenshot shows the 'Ultimate Warrior Competition - Web Leaderboard' page. It features a dark theme with a light blue trophy graphic. The page displays a podium for the top three winners: Anna Novak (2nd, 16500 pts), Jonathan Smith (1st, 20500 pts), and Samantha Singh (3rd, 16200 pts). Below the podium is a list of participants with their names and scores. The user 'YOU' is highlighted at rank 12 with 15900 pts and a company rank of #10. A tooltip indicates that the company rank allows the user to see where they are placed against other company players.

Rank	Name	Score
1st	Jonathan Smith	20500 pts
2nd	Anna Novak	16500 pts
3rd	Samantha Singh	16200 pts
4	Jaylon Septimus	16150 pts
5	Skylar Vaccaro	16150 pts
6	Mira Kenter	16150 pts
7	Lindsey Schleiher	16150 pts
8	Cheyenne Rosser	16150 pts
9	Leo George	16150 pts
10	Cheyenne Siphron	16150 pts
11	Lindsey Schleiher	16150 pts
12	YOU	15900 pts
13	Leo George	15850 pts
14	Cheyenne Siphron	15800 pts
15	Leo George	15800 pts

Reporting

Early reporting work in progress



Where we're headed next...



6 levels of design maturity

The Road Ahead

- Regions of Excellence
- Innovation with Triads
- Welcome new(er) team members

Strategic domains

APAC + AMER

Intelligence

Providing data, insights and reporting to empower customers & devs

Integrations

Connecting to other products

Growth

Optimizing pre-purchase, buying, onboarding and growth experiences

EMEA

Learning

Best-in-class secure coding educational content and learning experiences as well as their associated configuration

Global

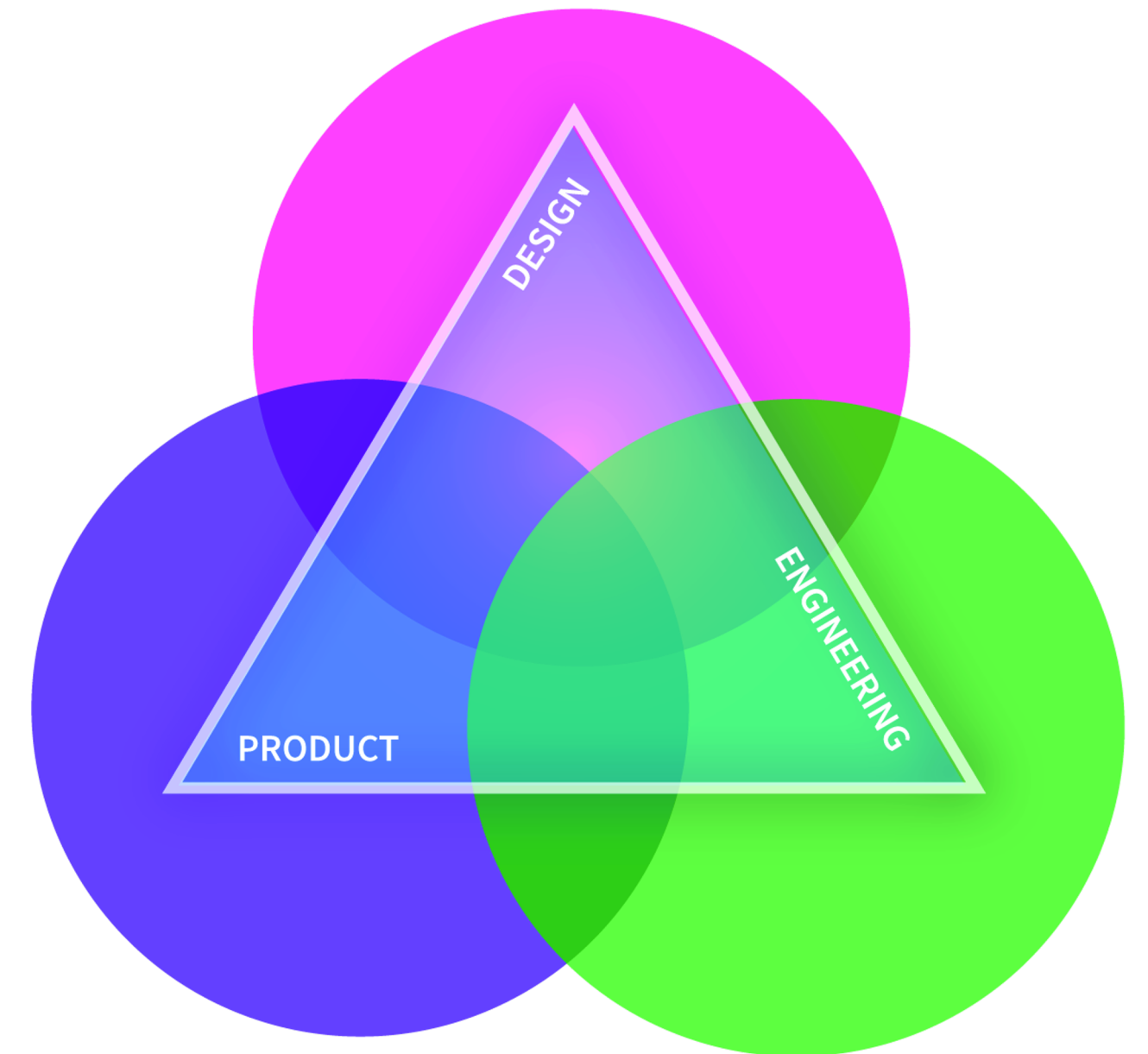
Foundations

Provides centralized value that benefits the whole

i.e. security, privacy & compliance, accessibility, product data, API experience, internal dev enablement

Innovation with Triads

- Archetypical product triad:
 - Product Manager + Designer + Tech Lead
- Shared ownership leads to shared (and deeper) understanding
- Less handoffs = less miscommunication and better solutions
- Other triad combinations and even “quads” are possible, e.g. including content, data science etc.



Welcome New Product & Design Hires!



Questions? Just Ask

- VP of Product
- VP of Engineering
- Head of Design