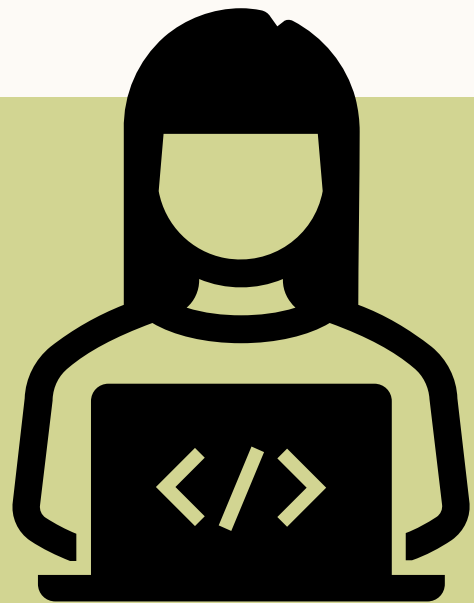


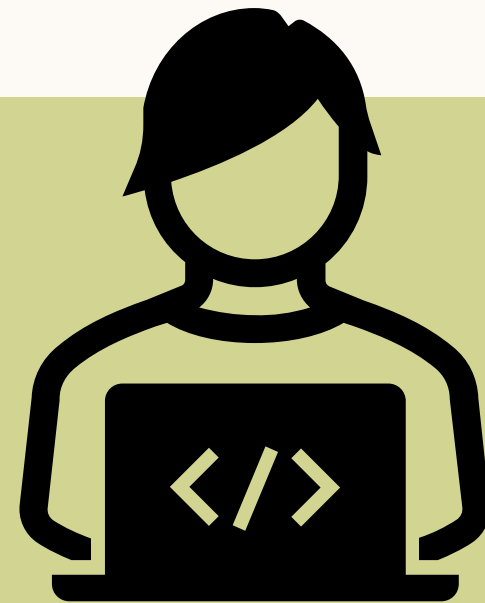
**New Checkie  
Facilities in   
Inspection Manager**

# Team Members



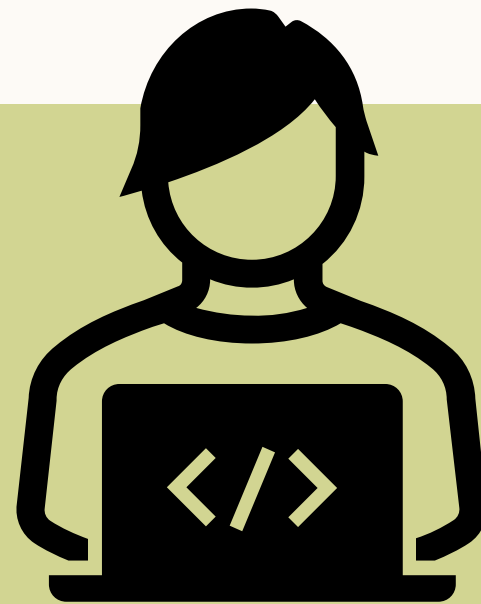
**Joy**

Team Leader



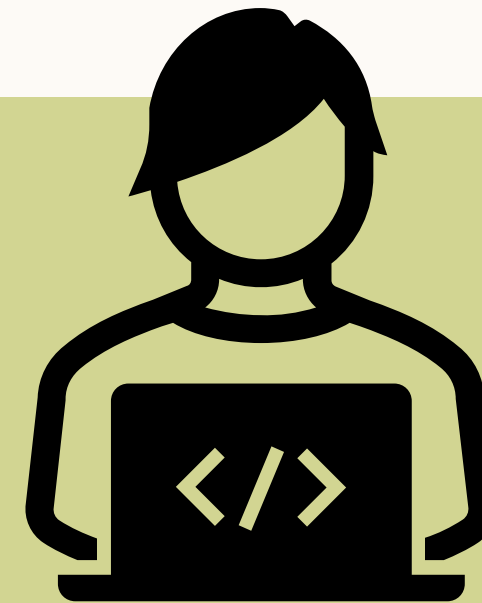
**Usmonkul**

Front-end Dev



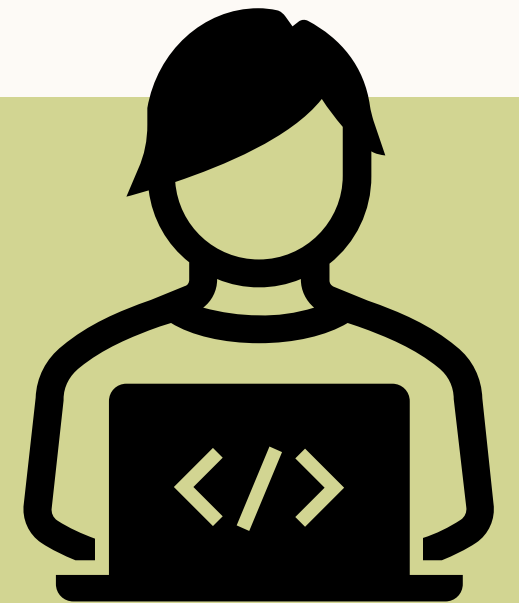
**Saurav**

Front-end Dev



**Ernist**

Back-End

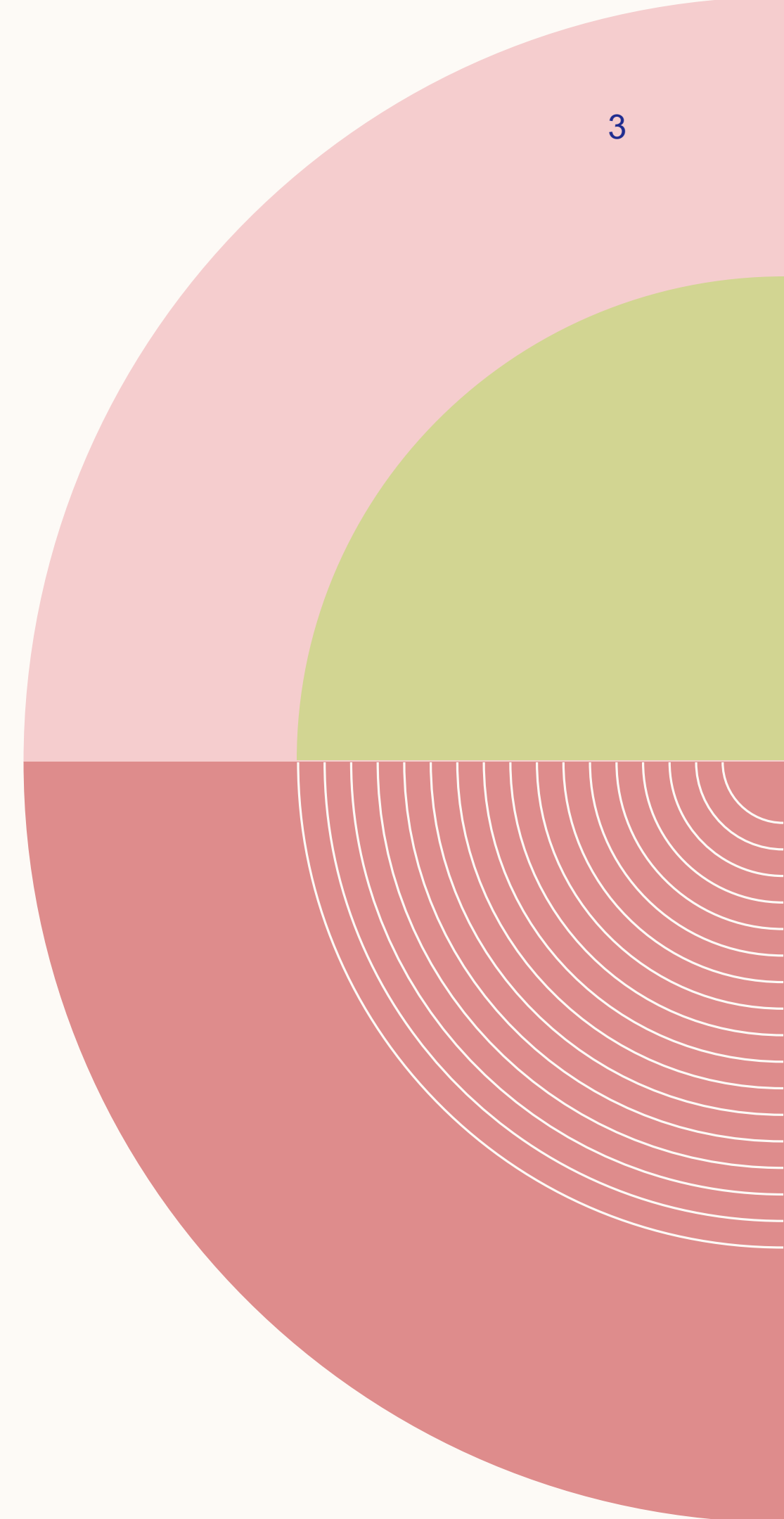
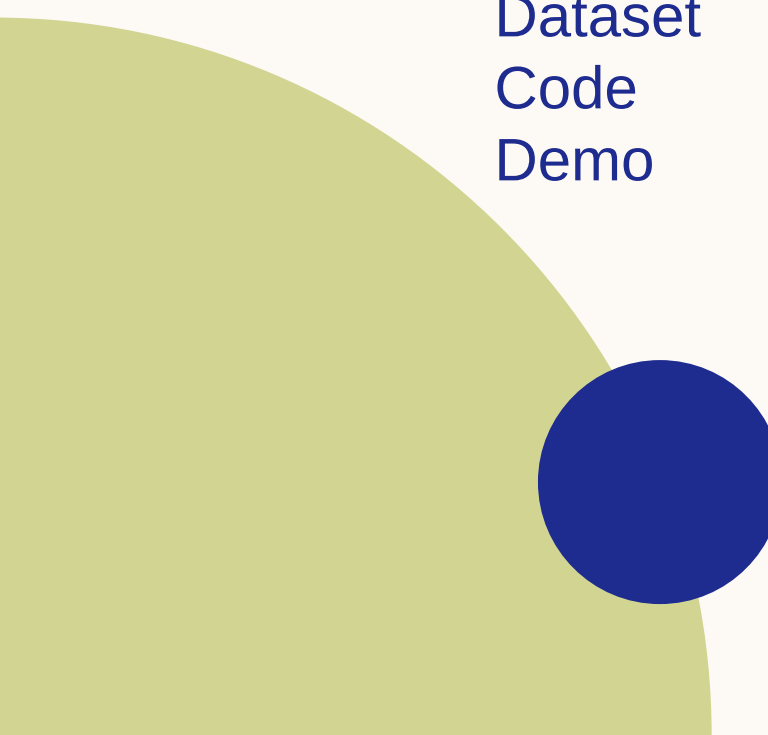


**Jinhong**

Back-end

# AGENDA

- Introduction
- Primary goals
- Project plan
- Development environment
- Functional requirement
- SW Architecture Diagram
- High level design
- Dataset
- Code
- Demo





CHEKIE

# Introduction

CHEKIE Web manager is a website to manage an inspection in the company. The manager will be able to manage registering an inspection target, item, and inspectors.

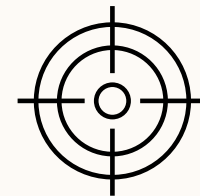
# Goal



1



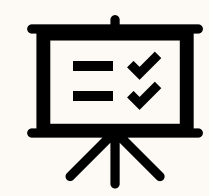
2



3



4



5

Guaranteed inspection and real-time confirmation of inspection omissions and issues

Prompt response

Provide a work environment free from time and place restrictions

Improve convenience and productivity

Eliminates the need for paper-based manual inspection

# TIMELINE

**Month 3**

**Month 3 - 4**

**Month 4**

**Month 5**

**Month 6**

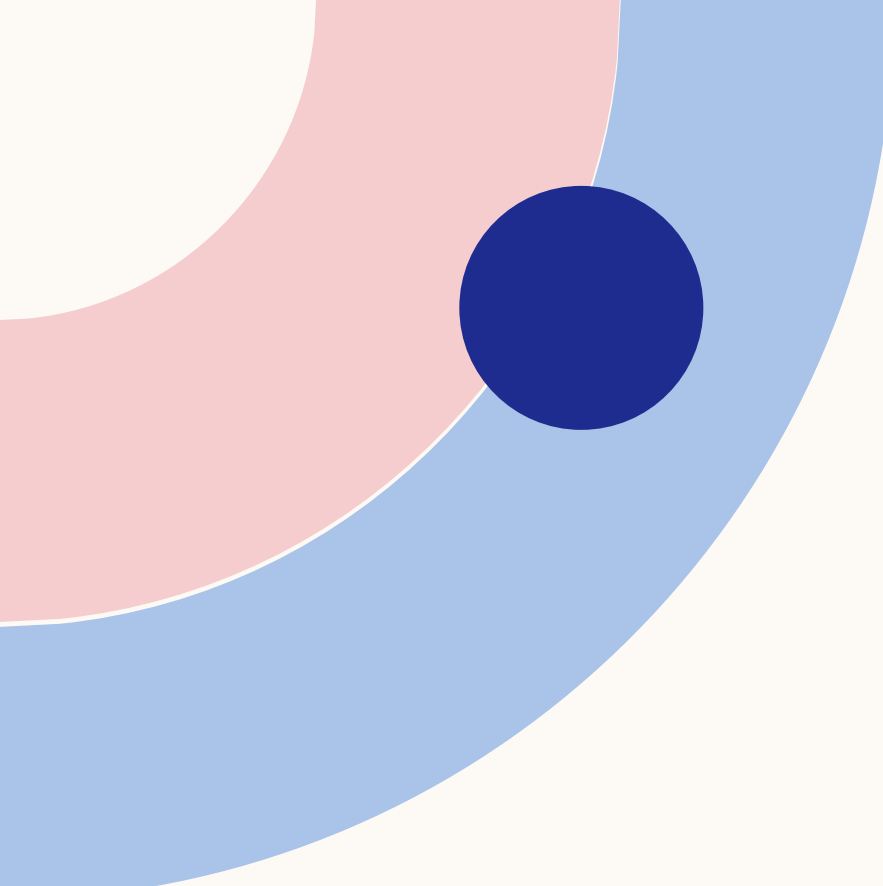
Research and  
planning.

Writing functional  
requirement and  
specifications.

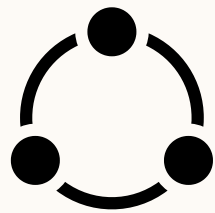
UI Design

UI Development

Testing & Launching

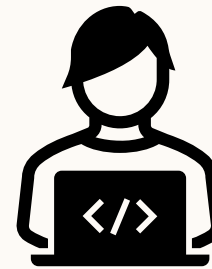


# Development Environment



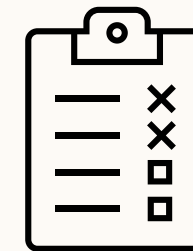
## tools

- Git
- GitHub
- Visual Studio Code
- Figma



## Language & Framework

- JavaScript
- TypeScript
- Next Js | React Js
- Tailwind CSS
- HTML(JSX)



## Other

- Jest
- Webpack
- Vercel

# Functional Requirement



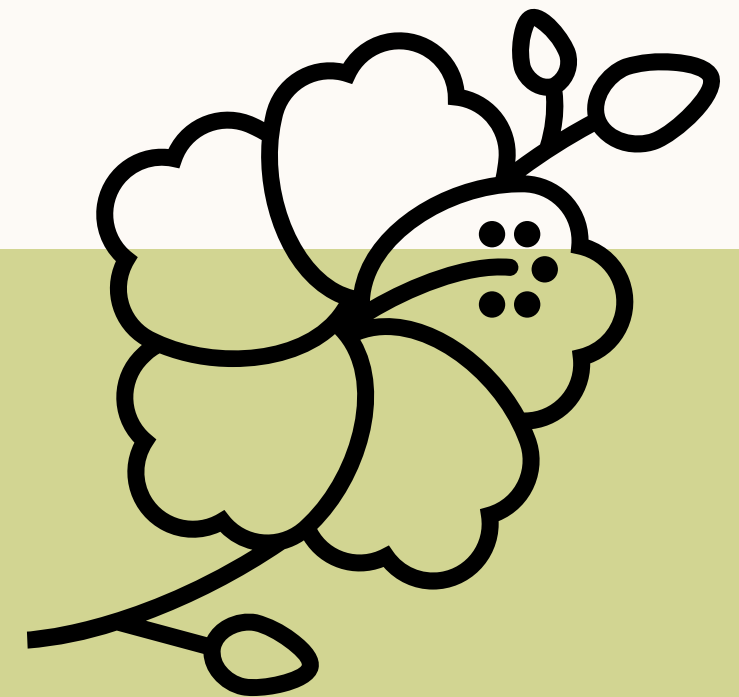
**Register target**



**Inspection item**



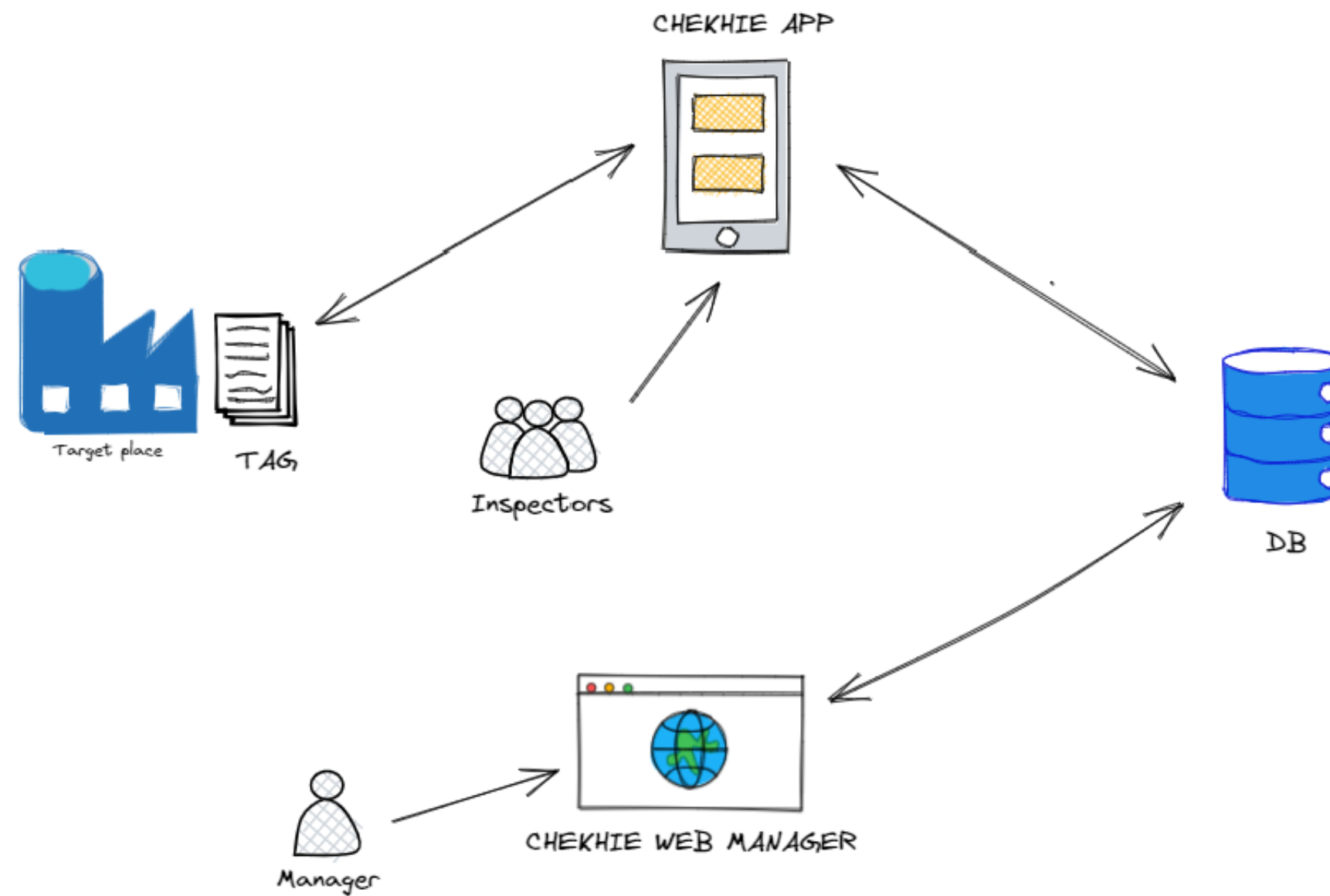
**Manage Inspector**



**Manage Category**



# HLD



# Figma Design

<https://www.figma.com/file/bWGMYhyQv5QNr8b46qdMws/CHEKIE?node-id=0%3A1&t=CdbxniNSAFzZrQOm-1>



# Dataset

We got the dataset from the backend developer. We worked jointly and in the last, we got the dataset from the backend developer and developed the page



```
1  import Head from "next/head";
2  import Auth from "../auth";
3  import { useUserStore } from "@context/auth.context";
4  import Intro from "@components/intro/intro";
5
6  export default function Home() {
7    const isUserExist = useUserStore((state) => state.isUserExist);
8    return (
9      <>
10     <div className=" overflow-hidden">
11       <Head>
12         <title>Checkie - Home</title>
13       </Head>
14       {!isUserExist ? <Auth /> : <Intro />}
15     </div>
16   </>
17 );
18 }
```

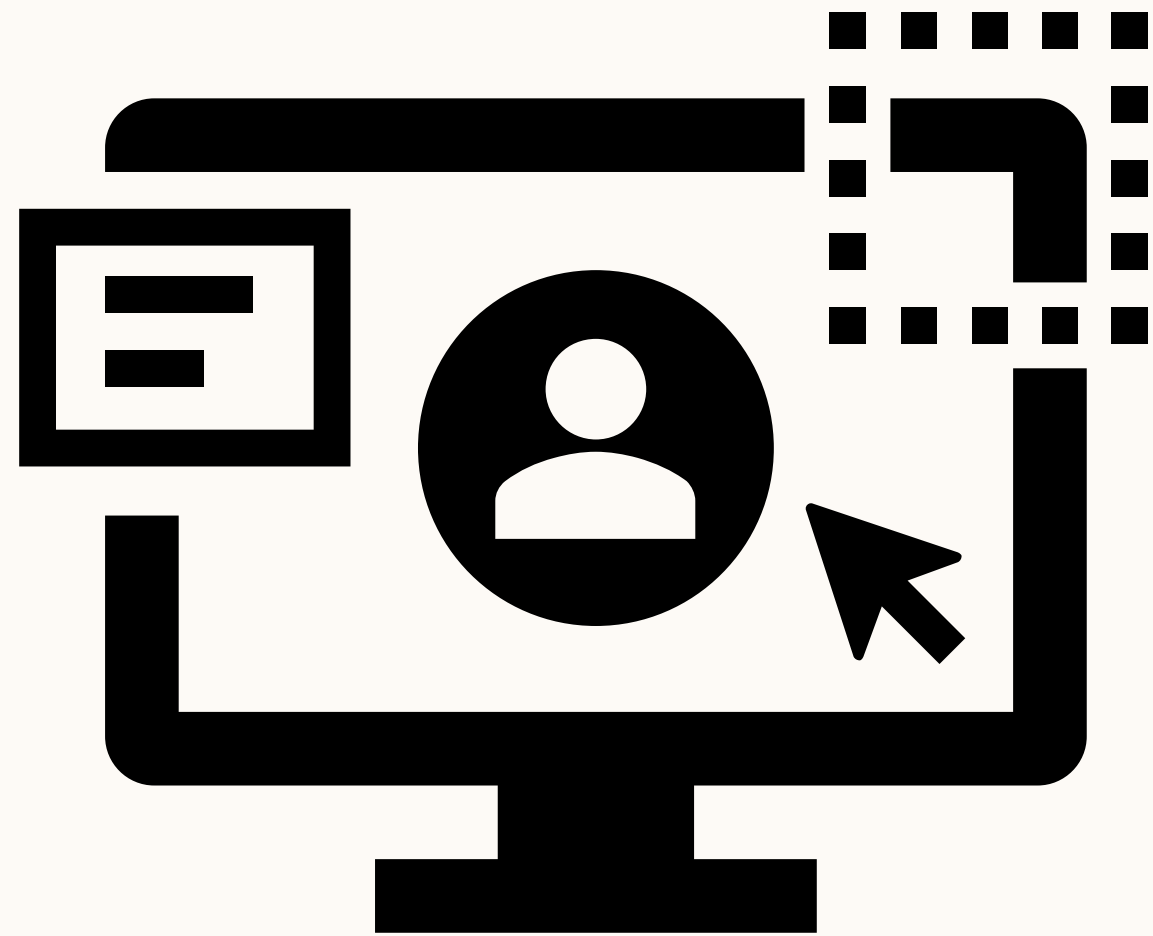
# Code

# Code

```
1  /** @type {import('next').NextConfig} */
2  const nextConfig = {
3    • reactStrictMode: true,
4    images: {
5      domains: ["placeholder.co"],
6    },
```

# CODE

- <https://github.com/Usmonkul/checkie-manager>



# Demo

CHECKIE

HOME SIGN OUT

Are you still using  
paper based inspection  
on your targets?

Use **Checkie mobile app** and an **Inspection manager web app** to do your inspection fast, real-time, and without paper.

Contact Us

Download App



# THANK YOU

