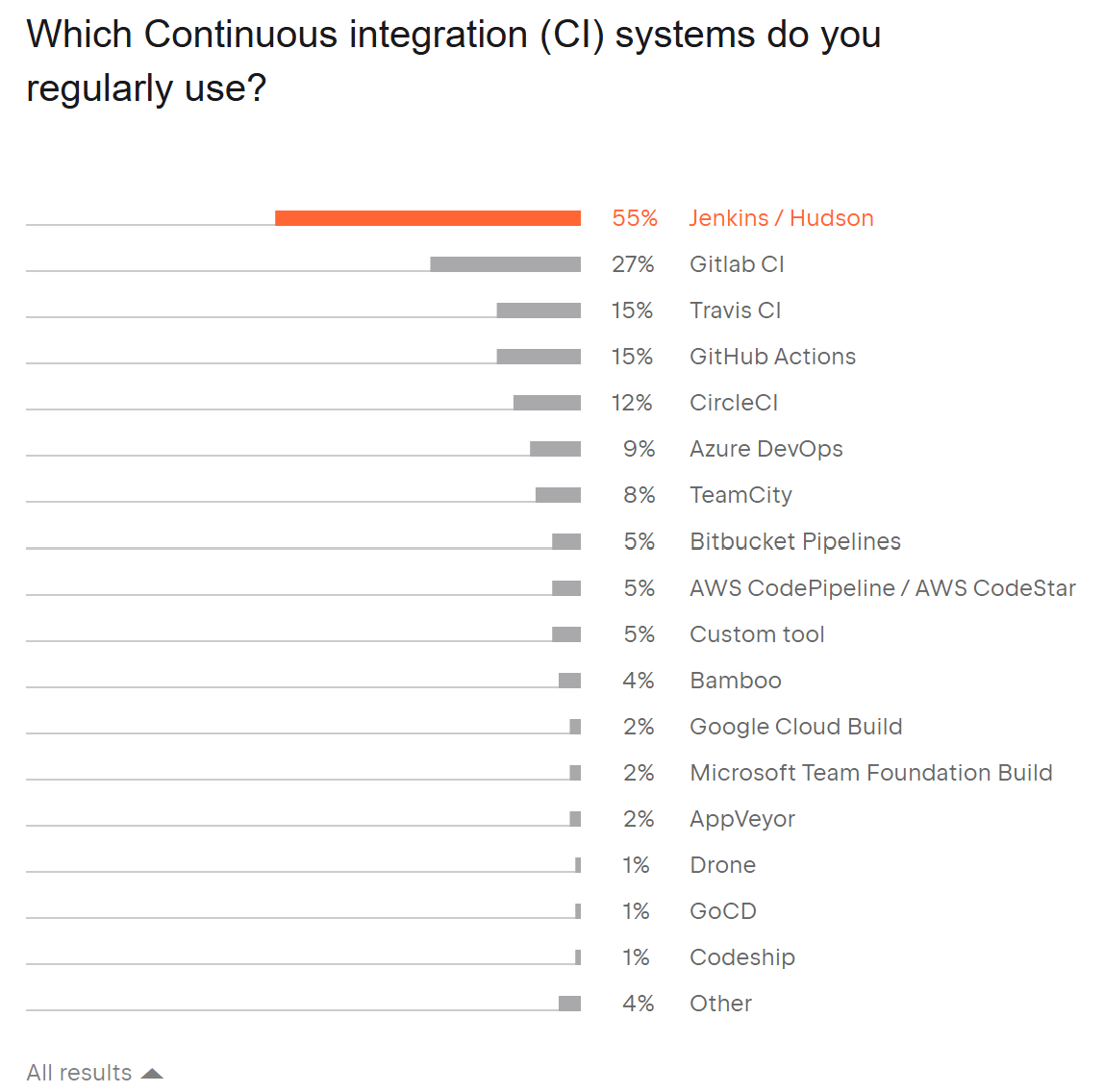
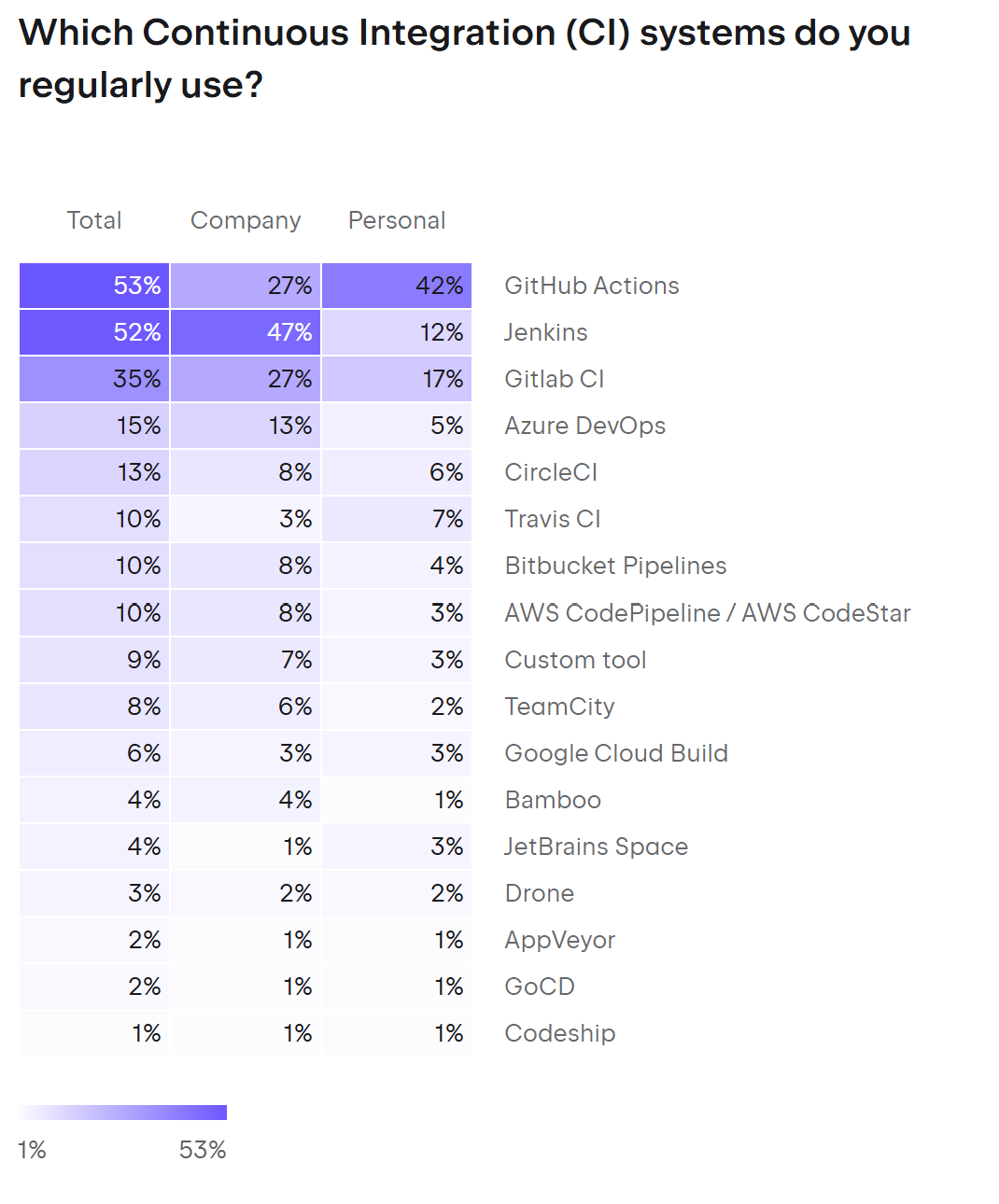
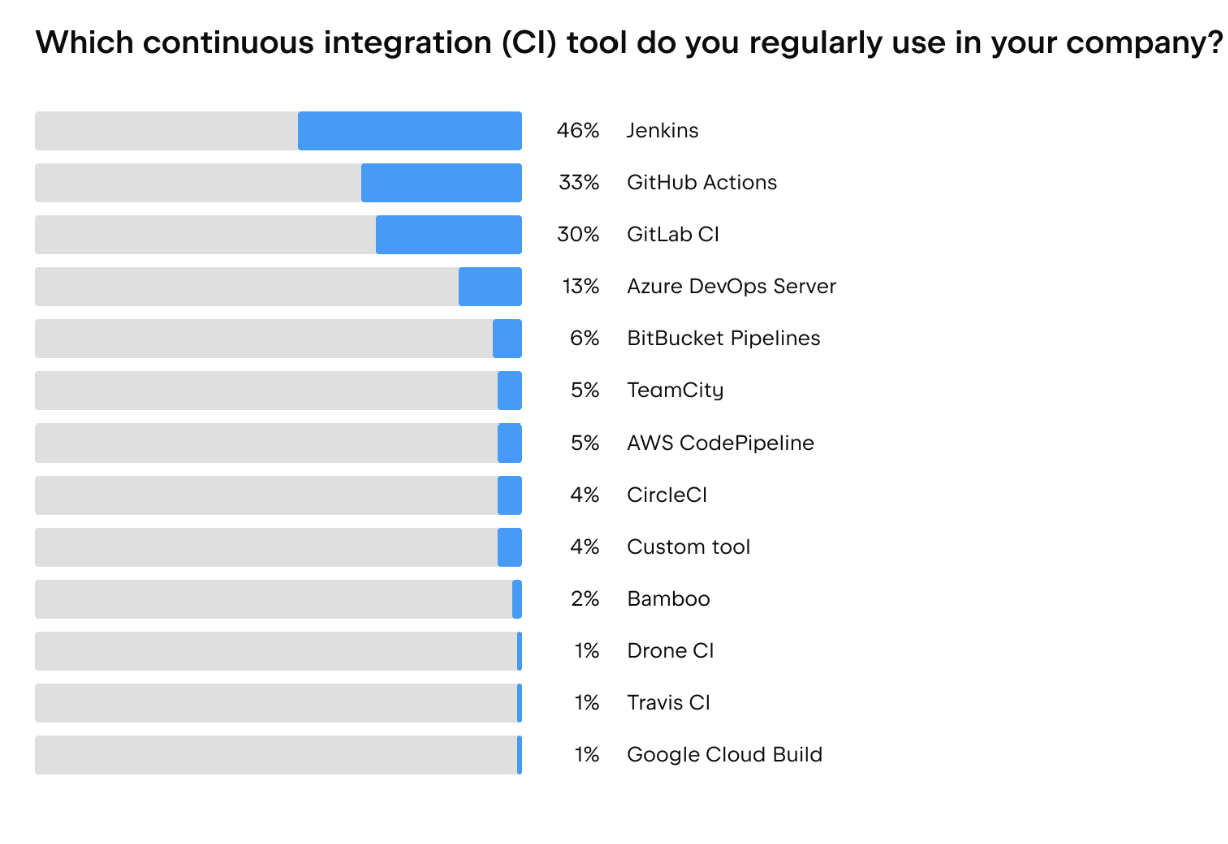
**DevPilot 기획서**

**1. 프로젝트 개요**

* **프로젝트 소개**: CI/CD 배포 원스톱 자동화
* **배경**
  + 초기 스타트업 · 주니어 1인 개발자는 EC2 위에 애플리케이션을 배포할 때 Docker 설치, Jenkins 구성·파이프라인 작성 등에 대한 러닝 커브와 소요 시간이 큼.
  + 이러한 까닭으로 JETBRAINS의 Developer Ecosystem Report에 따르면, 2020 년 이후 꾸준히 Jenkins의 Company 사용률이 감소하고 있음 (55% -> 47% -> 46%)

* + Github action 이나 Gitlab CI/CD가 Jenkins의 대체제로 활용될 수 있으나, 이 또한 배경 지식이 부족할 경우 시행착오를 겪어야 함.
  + 또한 AI의 발전에 따라 배경 지식이 부족한 상태로 프로그램을 제작하는 사람들의 증가할 것으로 예측됨.
  + 초기 스타트업, 1인 개발을 하고 싶은 사람 모두 MVP를 빠르게 배포해볼 수 있는 툴의 필요성 증대.
* **목표**
  + 사용자가 UI를 통해 쉽고 간편하게 Jenkins와의 연동, 인증, 파이프라인 구성 등을 설정하여, 설정 완료된 Jenkins 및 컨테이너를 EC2 서버에 대신 배포해주는 오픈소스 플랫폼을 지원.
  + MVP 단계에서는 Spring Boot, React, MySQL 스택만 지원하고, 추후 확장 예정

**2. 주요 기능**

|  |  |  |
| --- | --- | --- |
| **구분** | **기능명** | **설명** |
| CI/CD 연동 | GitLab/GitHub 키관리 | 컨테이너 내 Jenkins가 해당 호스트의 Webhook을 수신하도록 자동 설정 |
| 인프라 자동화 | EC2 연결 및 준비 | SSH 스크립트로 Docker/Jenkins 설치 |
| 아키텍처 UI | 드래그 앤 드랍 UI | 사용자 정의 블록 조합으로 환경 구성 가능한 시각화 에디터 |
| 파이프라인 템플릿 | 스택별 파이프라인 | Spring, React, MySQL 빌드·테스트·배포 스크립트 제작 및 제공 |
| 모니터링 | 실행 로그·상태 조회 | Jenkins 콘솔·Docker 컨테이너 상태 실시간 확인 |
| 챗봇 | 배포 지식 전달을 위한 챗봇 | 배포 지식의 부재로 어려움을 겪는 사용자를 위한 LLM 기반 챗봇 서비스 |

**3. 기술 스택**

* **백엔드**: Spring Boot (Java), FAST API(Python)
* **프론트엔드**: React (+TypeScript), 드래그 앤 드랍 라이브러리 (예: React DnD)
* **DB**: MySQL
* **원격 제어**: SSH 스크립트
* **인증·보안**: JWT, SSH 키 관리
* **배포 환경**: AWS EC2 (Ubuntu 22.04 LTS 권장)
* **챗봇 서비스:** Python

**4. 기대 효과**

* **백엔드 지식 없이도 배포 자동화 구현 가능**
  + Jenkins, GitHub Actions 등 기존 CI/CD 도구에 필요한 복잡한 설정이나 스크립트 작성 없이 EC2 서버에 자동 빌드 및 배포 체계를 구축하여, 인프라 지식이 부족한 프론트엔드 중심 개발자도 단독으로 서비스 운영이 가능.
* **운영 부담 및 보안 리스크 최소화**
  + 복잡한 설정을 자동화하여, 초기 세팅 실수나 SSH 키 유출과 같은 보안 사고를 줄이고, 인프라 운영에 대한 부담을 낮춤
* **빠른 MVP 출시로 시장 검증 속도 향상**
  + 배포에 익숙하지 않은 개발자도 러닝 커브 없이 빠르게 배포를 완료할 수 있어, 서비스 아이디어를 신속히 검증하고 린(Lean)하게 개발 가능