Link Innovation — Services
Full Page Mockups & Copy

Included: AI & Automation, Cybersecurity, Cloud Services, Web App Development, Mobile App Development, Automation Testing

Generated mockup PDF with content for designer/developer handoff.

Exported by Link Innovation Design System

AI & Automation — Page Content (Link Innovation

Hero CTAs:

Problems We Solve

- Manual processes slow down operations and increase error rates. - Teams struggle to surface timely insights from growing data. - LLMs and agents are hard to integrate and monitor in production. - Model drift and performance gaps create business risk.

Our Capabilities

Intro: We package capabilities into fast, measurable outcomes — from discovery to production.

How We Deliver

4-step delivery model: 1. Discover — Data assessment, business KPI mapping, PoC scope. 2. Design — Solution architecture, LLM prompt engineering, agent design. 3. Build — Model training, pipeline, RPA bots, integration with systems. 4. Operate — MLOps, monitoring, periodic retraining, business dashboards. Typical timeline: PoC (3–6 weeks) → MVP (6–10 weeks) → Production (ongoing ops).

Outcomes & KPIs

- Time saved on manual tasks: up to 60% (typical) - Forecast accuracy improvement: 10–30% - Request automation: 50–80% automation rate - Faster detection of model drift with monitoring

Architecture & Tech Stack

Recommended stack: - Model & LLM: OpenAI / Azure OpenAI, Anthropic, Hugging Face - Prompt orchestration & agents: LangChain, LlamaIndex, custom agent frameworks - MLOps: MLflow, TFX, Kubeflow, Sagemaker/Vertex pipelines - Processing & infra: Python (FastAPI), Docker, Kubernetes (EKS/GKE/AKS) - Data & streaming: PostgreSQL, S3, Kafka / PubSub, Spark (if needed) - RPA / Orchestration: UiPath, Airflow, Temporal - Monitoring: Prometheus, Grafana, ELK, Sentry - Security: Vault, WAF, IAM, encryption at rest & transit

Case Study

Title: Autonomous Claims Processing for InsureCo Problem: Manual claim intake averaged 36 hours per claim and high error rates. Solution: Link Innovation built an agentic workflow combining GenAI summarization, OCR, and RPA, integrated with the insurer’s claims system and dashboards. Outcome: Claim intake time reduced from 36 hours → 6 hours (83% faster). Error rate reduced 45%. ROI realized in 5 months.

Pricing Models

- Discovery / Readiness Review — fixed fee (1–2 weeks) — deliverable: AI readiness report & PoC plan. - PoC / MVP — T&M or fixed-scope (4–8 weeks) — working prototype. - Productization & Ops — monthly retainer / managed service (SLA-based). - License / Component — for reusable platforms (white-labelled agents).

FAQ

Q: How do you protect sensitive data in LLMs? A: We use data isolation, encryption, private hosted models, redaction, and strict access policies.

CTAs, Forms

Primary CTA: Request an AI Readiness Review Form fields: Company, Name, Email, Phone (optional), Service interest, Short description, Budget (optional), Upload sample data (optional). Include UTM fields. Confirmation microcopy: “Thanks — we’ll review your request and send available times for a 30-min intro.”

Wireframe / Layout

1. Hero (full width) with left illustration, H1, subhead, CTAs 2. Problems (bullets) 3. Capabilities grid (cards) 4. Delivery timeline (visual)

Cybersecurity — Page Content (Link Innovation)

Hero CTAs:

Problems We Solve

- Undetected threats and slow incident response. - Insecure application code and CI/CD pipelines. - Cloud misconfigurations exposing sensitive data. - Gaps in identity management and privileged access controls.

Our Capabilities

Intro: Practical security services designed for enterprises with modern cloud and development practices.

How We Deliver

Security lifecycle model: 1. Assess — risk assessment, asset mapping, baseline controls. 2. Protect — hardening, secure SDLC, IAM implementation. 3. Detect — monitoring, logging, advanced analytics. 4. Respond & Recover — playbooks, incident response, remediation. Typical engagement: Assessment (1–2 weeks) → Remediation (2–6 weeks) → Managed services (ongoing).

Outcomes & KPIs

- Mean time to detect (MTTD) improvement: from days to minutes. - Vulnerability remediation rate: 90% within SLA. - Cloud misconfiguration reduction: 70% after remediation. - Compliance readiness: ISO/SOC audit pass rates improved.

Architecture & Tech Stack

Recommended stack: - SIEM & MDR: Splunk, Elastic, Azure Sentinel, Datadog Security - Endpoint & EDR: CrowdStrike, SentinelOne - AppSec tools: Burp Suite, Snyk, Veracode, OWASP ZAP - Cloud security: Prisma Cloud, AWS Security Hub, Azure Defender - IAM & PAM: Okta, Azure AD, CyberArk - DevSecOps: SCA/SAST/DAST CI integrations, IaC scanning (Checkov/Terraform) - Incident response: Phantom, Cortex XSOAR, custom SOAR playbooks

Case Study

Title: Securing FinTech Platform X Problem: FinTech client suffered frequent false positives and long alert triage times. Solution: Implemented MDR with tuned detection rules, integrated EDR, and improved runbooks. Outcome: Reduced triage time by 70%, eliminated 40% of false positives, and passed SOC 2 audit. Assets: detection metrics, runbook examples, remediation timeline.

Pricing Models

- Security Assessment — fixed fee (1–2 weeks) - Remediation Project — fixed or T&M (2–8 weeks) - Managed Security (MDR/SOC) — monthly retainer (SLAs) - Ad-hoc testing — penetration testing, red team exercises

FAQ

Q: Do you offer 24×7 SOC support? A: Yes — MDR is available as a managed service with optional 24×7 coverage and escalation. Q: How do you handle sensitive customer data? A: We use secure vaults, encryption, strict RBAC and attestations; we can work on-site or under NDA. Q: Can you integrate into our CI/CD pipeline? A: Yes — we integrate SAST/SCA/DAST into CI and provide remediation guidance.

CTAs, Forms

Primary CTA: Start a Free Security Assessment Form fields: Company, Name, Email, Critical systems, Preferred SLA, Short description, Budget. Confirmation microcopy: “Thanks — our security team will review and follow up with available audit slots.”

Wireframe / Layout

1. Hero with security illustration, H1, CTA 2. Problems & threats (visual) 3. Capabilities grid (cards) 4. Security lifecycle (Assess→Protect→Detect→Respond)

Cloud Services — Page Content (Link Innovation)

Hero CTAs:

Problems We Solve

- Risky lift-and-shift migrations and long rollback windows. - High cloud spend due to poor resource sizing. - Lack of observability and frequent outages. - Slow, manual infrastructure provisioning.

Our Capabilities

Intro: End-to-end cloud services covering migration, automation, operations, and optimization.

How We Deliver

Cloud delivery lifecycle: 1. Assess — cloud readiness and TCO analysis. 2. Architect — migration plan and reference architectures. 3. Migrate & Modernize — execution and cutover planning. 4. Operate & Optimize — FinOps and SRE-led operations. Timeline examples: Readiness review (1–2 weeks), Migration (4–12 weeks), Ongoing managed ops (monthly).

Outcomes & KPIs

- Deployment frequency: 3× improvement with CI/CD and IaC. - Cost savings: 20–40% through FinOps measures. - Uptime & reliability: 99.9%+ with SRE practices. - Mean time to recovery improved substantially.

Architecture & Tech Stack

Recommended stack: - Cloud platforms: AWS, Azure, GCP - IaC: Terraform, Pulumi, AWS CDK - Containers & Orchestration: Docker, Kubernetes (EKS/AKS/GKE) - CI/CD: GitHub Actions, Jenkins, GitLab CI, ArgoCD - Monitoring: Prometheus, Grafana, Datadog, CloudWatch - Storage & DB: RDS, Aurora, S3, Redis - FinOps: CloudHealth, Spot.io, Cost Explorer

Case Study

Title: Migration & Cost Optimization for SaaSCo Problem: SaaSCo faced high cloud spend and slow release cadence. Solution: Performed migration to managed Kubernetes, refactored key services, implemented IaC and FinOps program. Outcome: Reduced monthly cloud costs by 38% and improved deployment velocity by 4×.

Pricing Models

- Cloud Readiness Review — fixed fee - Migration Project — fixed/t&m (4–12 weeks) - Managed Cloud Ops (SRE) — monthly retainer - FinOps Program — consulting + ongoing automation

FAQ

Q: Which cloud platform should we choose? A: We recommend assessing business needs; we are multi-cloud and can recommend AWS/Azure/GCP based on workload.

CTAs, Forms

Primary CTA: Schedule a Cloud Readiness Review Form fields: Company, Name, Email, Cloud provider preference, Approx monthly spend, Short description. Confirmation microcopy: “Thanks — our cloud team will review and follow up with a readiness plan.”

Wireframe / Layout

1. Hero with cloud illustration, H1, CTAs 2. Problems & goals 3. Capabilities grid 4. Migration plan & timeline

Web App Development — Page Content (Link Inno

Hero CTAs:

Problems We Solve

- Slow, unmaintainable legacy web platforms. - Poor user experience and accessibility gaps. - Monolithic architectures that hinder scaling. - Fragmented data and poor integrations.

Our Capabilities

Intro: Full-stack web development services across frontend, backend, and integrations.

How We Deliver

Web delivery process: 1. Discover & Design — UX, flows, prototypes. 2. Build — component-driven frontend, API-first backend. 3. Test — automated testing, performance tuning. 4. Deploy & Monitor — CI/CD, observability, feature flags. Typical timeline: MVP (4–8 weeks), Enterprise portal (8–16 weeks).

Outcomes & KPIs

- Page load improvements (LCP/FID) and better Core Web Vitals. - Faster time-to-market for features via component reuse. - Accessibility compliance improvements. - Reduced maintenance costs long-term.

Architecture & Tech Stack

Recommended stack: - Frontend: React, Next.js, TypeScript, Tailwind CSS - Backend: Node.js, Express, NestJS, Next API routes - DB: PostgreSQL, MongoDB, Redis - APIs: REST, GraphQL, tRPC - DevOps: Vercel, Netlify, Docker, GitHub Actions - Monitoring: Sentry, Datadog, Lighthouse

Case Study

Title: Enterprise Dashboard for RetailCo Problem: RetailCo needed consolidated operational dashboards across stores. Solution: Built Next.js portal with aggregated data pipelines and custom visualizations. Outcome: Reduced reporting time from days to real-time dashboards; improved decision making.

Pricing Models

- MVP Quote — fixed scope (4–8 weeks) - Product Development — T&M or fixed priced engagements - Support & Maintenance — monthly retainer

FAQ

Q: Do you support accessibility standards? A: Yes — we follow WCAG guidelines and perform audits for compliance. Q: Can you integrate existing APIs? A: Yes — we design secure adapters and integration layers.

CTAs, Forms

Primary CTA: Get an MVP Quote Form fields: Company, Name, Email, Project description, Platform, Budget, Timeline. Confirmation microcopy: “Thanks — we’ll review your project and share an estimated timeline.”

Wireframe / Layout

1. Hero with dashboard imagery, H1, CTAs 2. Problems & benefits 3. Capabilities grid 4. Architecture & stack

Mobile App Development — Page Content (Link In

Hero CTAs:

Hero Web Mobile

Problems We Solve

- Slow time-to-market for native apps. - Fragmented user experiences across platforms. - Offline and sync challenges in low-connectivity regions. - Complex app store submissions and versioning.

Our Capabilities

Intro: Mobile engineering services covering cross-platform, native enhancements, offline sync, and analytics.

How We Deliver

Mobile delivery model: 1. Discovery & Design — UX, wireframes, device flows. 2. Build — cross-platform/core native modules. 3. Test — device lab, automation, crash monitoring. 4. Release & Monitor — store release, analytics, iteration. Typical timeline: MVP (6–10 weeks), Production (ongoing improvements).

Outcomes & KPIs

- Time-to-market reduction: ~40% with cross-platform. - Crash rate: reduced via CI/CD and monitoring. - Engagement: improved retention with performance optimization.

Architecture & Tech Stack

Recommended stack: - Frameworks: React Native, Flutter, Swift, Kotlin - Backends: Firebase, Supabase, AWS Amplify, GraphQL APIs - CI/CD: Fastlane, Bitrise, App Center, GitHub Actions - Analytics: Firebase Analytics, Mixpanel, PostHog - Testing: Appium, Detox, BrowserStack

Case Study

Title: Field Ops Mobile App for FieldCo Problem: FieldCo needed offline data capture and sync for remote teams. Solution: Built an offline-first React Native app with background sync and conflict resolution. Outcome: Field data availability increased from 60% to 98% and user adoption rose.

Pricing Models

- Discovery & Prototype — fixed scope - App Build — fixed/T&M engagement - Support & Maintenance — monthly retainer

FAQ

Q: Do you support Android TV or tablets? A: Yes — we design responsive experiences for multiple device classes. Q: How do you handle offline conflicts? A: We design sync strategies with conflict resolution, versioning, and user prompts.

CTAs, Forms

Primary CTA: Start Mobile Discovery Form fields: Company, Name, Email, Platform (iOS/Android), Device targets, Timeline. Confirmation microcopy: “Thanks — our mobile team will reach out to schedule a discovery call.”

Wireframe / Layout

1. Hero with device mockups, H1, CTAs 2. Problems & user needs 3. Capabilities grid 4. Architecture & offline strategy

Automation Testing — Page Content (Link Innovat

Hero CTAs:

Problems We Solve

- Manual regression cycles delaying releases. - Unreliable test coverage and flaky tests. - Performance bottlenecks undetected until production. - Mobile fragmentation and device coverage challenges.

Our Capabilities

Intro: Testing services and frameworks for modern development teams.

How We Deliver

Testing engagement model: 1. Audit — test coverage & flakiness analysis. 2. Framework build — implement automation frameworks and pipelines. 3. Integrate — CI/CD embedding and reporting dashboards. 4. Maintain — test upkeep and expansion as product evolves. Timeline: Audit (1–2 weeks), Framework (2–6 weeks), Ongoing maintenance.

Outcomes & KPIs

- Regression cycle time reduced by up to 80%. - Test coverage improvement and fewer production bugs. - Performance baselining and fewer outages under load.

Architecture & Tech Stack

Recommended stack: - Web: Cypress, Playwright, Selenium - Mobile: Appium, Detox, BrowserStack - API: Postman, Newman, RestAssured - Performance: JMeter, k6, Gatling - CI/CD: GitHub Actions, Jenkins, Azure DevOps - Reporting: Allure, TestRail, custom dashboards

Case Study

Title: Automated Regression for SaaS Platform Y Problem: Long regression cycles delayed releases and caused rollbacks. Solution: Implemented Cypress E2E suites, integrated into CI with parallel runs and flaky test handling. Outcome: Regression time reduced from 48 hours to under 3 hours and rollback rate decreased significantly.

Pricing Models

- Testing Audit — fixed fee - Framework Build — fixed/t&M - Continuous Testing Support — monthly retainer

FAQ

Q: How do you handle flaky tests? A: We implement test isolation, retries with backoff, improved selectors, and parallelization strategies. Q: Do you provide device labs? A: Yes — we integrate with BrowserStack, Firebase Test Lab, and cloud device providers.

CTAs, Forms

Primary CTA: Request a Test Automation Assessment Form fields: Company, Name, Email, Tech stack, Areas of concern, Priority. Confirmation microcopy: “Thanks — our testing practice will review and get back to schedule an audit.”

Wireframe / Layout

1. Hero with pipeline imagery, H1, CTAs 2. Problems & testing gaps 3. Capabilities grid 4. Audit & framework steps