Project Abstract（2025/06/10）

市民閲覧性を最大化する地方公共団体 Web リニューアル基盤

1. サービス概要

セクション	内容
1.1 ミッション	すべての自治体サイトを「迷わず辿り着ける・読める・操作できる」体験に統一し、災害・行政手続き情報を誰一人取り残さず届ける。
1.2 対象ステークホルダー	住民（高齢者・障害者を含む）、自治体職員、NPO／ボランティア開発者、公共 DX 推進省庁、研究者。
1.3 主要価値	①アクセシビリティ AA 準拠、②自治体横断の検索性、③多言語・読み上げ対応、④災害時の秒単位更新。
1.4 提供機能	- 自治体 URL 登録だけで自動クロールと UI 最適化- PDF／スキャン画像をテキスト化し全文検索に統合- デジタル庁デザインシステム (DA-DS) 準拠の React UI で即時 SSR/ISR 配信- API 連携：他サービスが自治体データを JSON 取得可能
1.5 非機能要件	パフォーマンス：P95 < 1 s（静的キャッシュ時）、稼働率 99.9 %／月；セキュリティ：OWASP Top10 準拠；法令遵守：著作権法 30 条の4 情報解析目的に即したクロール、個人情報自動マスキング。
1.6 ガバナンス	公開前にアクセシビリティ & 法務 CI、GitOps（PR レビュー必須）。運用ドキュメントは Creative Commons 4.0 で公開し外部貢献を許容。

2. エンド-ツー-エンド処理フロー

2.0 オーケストレーション

- **イベント駆動ワークフロー** — AWS Step Functions / GCP Workflows
- **優先度別キュー** — 災害情報は High-Priority キューで即時処理
- **冪等性** — 「入力ハッシュ＝出力キー」で再送を吸収

2.1 クロール & ローデータ永続化

- **処理**: Playwright → HTML, 添付を S3 保存
- **警戒**: CMS テーブルレイアウト崩れ、100 MB 超ファイルでタイムアウト

- **対策**: サイズ/MIME フィルタ、署名 URL 直アップロード

2.2 非テキスト資源抽出 → リッチテキスト統合

- **処理**: pdfminer.six + PaddleOCR（縦書き判定）
- **リスク**: OCR 精度低下、CPU コスト増
- **対策**: 座標ソート、Cloud Run Job Spot

2.3 一次 JSON 化（LLM 抽出チェーン）

- **処理**: LangChain Structured Output + GPT-4o
- **品質**: Zod strict parse、個人情報マスク

2.4 BestUXUI スキーマ変換

- **実装**: TypeScript + Zod、バージョニング、AJV fallback

2.5 DA-DS コンポーネントマッピング

- **処理**: ルックアップテーブル、ラッパーパッケージ化
- **ライセンス**: CC-BY 4.0 クレジット表示

2.6 Next.js (SSR/ISR) & デプロイ

- **設定**: ISR revalidate ≥ 600 s、緊急ページ SSR no-store
- **デプロイ**: GitHub Actions → Vercel / CloudFront

2.7 CI/CD & ガバナンス

- **テスト**: Storybook VRT、axe-core、focus-order
- **レビュー**: 法務 & A11y merge gate
- **監視**: Datadog RUM、Cost Explorer Slack 報告

3. データフロー図

```
[URL List] → Crawler → Raw Store → OCR → LLM(JSON) → BestUXUI → React+DA-DS → Next.js → CDN
```

4. 導入直後の To-Do

1. BestUXUI スキーマ v1.0 凍結

2. クロール許諾ポリシー公開・オプトアウト窓口
3. 災害優先キュー SLA 定義
4. LLM コスト上限アラート設定
5. Storybook 未実装一覧を OSS タスク化