システム方式設計書

1. プロジェクト名

- 社内ドキュメント・資産・人材管理ができる社内インフラシステム

2. アーキテクチャ概要

- ウェブベースのアプリケーション

- クライアント、サーバー、データベースの3階層アーキテクチャ構造

- プレゼンテーション階層 : React.jsを使用してユーザーインターフェースを構成

- ビジネスロジック階層 : Springを使用して主要ロジックを処理

- データ階層 : MySQLを使用して権限、ユーザー、メニュー、会社情報を保存·管理

3. 技術スタック

- 開発言語 : Java1.8, html, css, javascript

- フレームワーク : React.js(フロントエンド), Spring(バックエンド)

- データベース : MySQL

- サーバー : Apache-Tomcat9.0

- 配布環境 : AWS EC2

- CI/CD : GitHub Actions

4. データ設計

4.1. データベース ERD

- ユーザーテーブル : user\_id, name, email, role\_id, created\_at, updated\_at

- 権限テーブル : role\_id, role\_name, description, created\_at

- メニュー·テーブル : menu\_id, menu\_name, url, role\_id

- 会社情報テーブル : company\_id, company\_name, address, representative, email

5. インターフェース設計

5.1. 外部システムとの連動

- 電子メール システムと連動してユーザー情報が修正されたときに自動電子メールを 送信

5.2. 内部インタフェース設計

- POST /api/users/create : ユーザー情報追加

- PUT /api/roles/update : ユーザー情報と権限情報の修正

- GET /api/menu/list : メニューリストの一覧と権限確認

- PUT /api/company/update : 会社情報アップデート

6. セキュリティ設計

- 重要なデータは、スプリング セキュリティを使用して暗号化し、保存

- すべてのユーザーはシステムアクセス時に認証が必要であり、JWTトークンを使用し てセッションの有効性を検査

7. メンテナンス計画

- モニタリング : log4jでエラー状況を記録し、今後の問題についてデバッグ

- バックアップおよび復旧 : 障害発生時にバックアップデータを使って迅速に復旧

できるように手順を用意