INFOSAFE

Team Contributions

InfoSafe

Seed Analytics

———

Team Name: FrAgile
Team Members: Christof Steyn

                  Chris Mittendorf

                  Karel Smit

                  Yané van der Westhuizen

                  Alistair Ross

Team Contact: **fragile.cos301@gmail.com**

Table Of Contents

Demo 2 Contributions

Christof Steyn

- Helped setting up the backend services for the database:
    - Asset Controllers and services.
    - Datascope Controllers and Services.
    - User Services.
- Helped link our backend to our frontend:
    - Asset Creation Linking.
    - Asset Viewing Linking.
    - User Creation Linking.
    - User Viewing Linking.
    - Datascope Creation Linking.
    - Datascope Viewing Linking.

Chris Mittendorf

- Worked on documentation for the project
    - System Architecture
    - Coding Standards
    - Database design
    - User Manual
    - SRS V2
- Worked on Backend with:
    - Password encryption
    - Initial random password generation
- Worked on frontend with:
    - User creation integration with password generation and encryption

- Organized GItHub including:
    - README updates
    - GitHub badges
    - Wiki page for organization and standards

Yané van der Westhuizen

- Completed designs for several pages and overlays including adapting initial designs.
- Implemented frontend for:
    - User Viewing
    - User Editing
    - Data Scope Viewing
    - Asset/Device Editing
- Refactored the home page implementation.
- Helped resolve, if any, errors or bugs occurred when merging frontend implementations.
- Assisted with creating the presentation for the demo.

Karel Smit

- Created designs for multiple subsystems and overlays.
- Frontend:
    - Create User overlay
    - Edit User overlay
    - Create Data Scope overlay
    - Edit Data Scope overlay
    - Create Device overlay
    - Edit Device overlay
    - Fixing of various issues related to the frontend
- Assisted with integration of the front and back-end.
- Unit testing for frontend.

Alistair Ross

- Setting up JWT:
    - Authentication scheme for logged in Users.
    - Setting up jwt tokens and sessions
    - Make the backend accept requests from logged in users.
    - Helped with setting up backend services and controllers.
    - Linking backend to frontend
- Testing:
    - Created all our Cypress Integration Tests.
    - Automated the Cypress Tests.
    - Wrote Spring Tests.

Demo 3 Contributions

Christof Steyn

- Backend:
    - Created/Integrated Edit endpoints for Users
    - Created/Integrated Edit endpoints for DataScopes
    - Created/Integrated Edit endpoints for Risks
    - Created/Integrated Edit endpoints for Assets
    - Created/Integrated Edit endpoints for all the Requests
    - Created/Integrated Edit endpoints for Tasks
    - Helped with integrating the User backend
    - Helped with integrating the Datascope backend
    - Helped with integrating the Asset backend
    - Helped with integrating the Request backend
    - Helped with integrating the Task backend
    - Updated all the DB models to match frontend data
    - Helped with linking the different roles to the correct subsystems

Chris Mittendorf

- Database:
    - Setup AWS RDS for the system
    - Redesigned the entities
- Hosting:
    - Setup initial AWS hosting
    - Began work on automatic deployment from GitHub to AWS
- Backend:
    - Created Models, Controllers, Repositories, Requests and Services for:
        - Access Requests
        - Asset Requests
        - Assets Types
        - Assigned Assets
        - Assigned Risks
        - Assigned Support Requests
        - Assigned Risks
        - Hashed Passwords

- - - Support Requests
- Documentation
    - Updated:
        - Database Designs
        - User Manual
        - SRS Doc
        - System Architecture
        - Coding Standards
    - Created:
        - Installation Manual

Yané van der Westhuizen

- Created/Updated designs for:
    - Data Scope View
    - Data Scope Edit
    - Data Scope Creation
    - Access Requests View
    - Access Requests Edit
    - Access Requests Creation
    - Requests Tab
    - Asset Requests Tab
    - Device View
    - Asset Requests Review
    - Device Creation
    - Device Edit
- Implemented/Improved front-end for:
    - Login
    - Home
    - ISO View
    - DISO View
    - Data Custodian View
    - Device Creation
    - Device Edit
    - Access Request View
    - Access Request Edit

- - Access Request Approval
    - Support Request View
    - Support Request Edit
    - Asset Request View
    - Asset Request Review
- Helped create the presentation slides for Demo 3.
- Fixed several front-end related issues.

Karel Smit

- Created/Updated the UI/UX designs for the following subsystems:
    - Create Risk
    - Edit Risk
    - Review Risk
    - Task Approval
    - ISO Risk Tab Design
    - DISO Risk Tab Design
- Implemented the front-end for the following:
    - Task Creation
    - Task View
    - Task Update
    - Task Approval
    - Risk Creation
    - Risk View
    - Risk Update
    - Risk Review
    - Employee view
    - Asset Manager view
- Various UI fixes and changes

Alistair Ross

- Backend Implementation
    - Continued implementation of service, controllers, repositories and models.
    - Implemented multiple "GET" and "POST" methods to improve frontend functionality.

- ○ Improved JWT authentication, as well as implementing CORS configuration to allow for communication with the frontend.
- Frontend and backend integration
  - ○ Implemented all viewing
  - ○ Implemented all creating
  - ○ Setup error checking to ensure correct data is saved

Demo 4 Contributions

Christof Steyn

- Hosting:
    - Setting up our aws ec2 server for hosting.
- Testing:
    - Created and updated all our frontend unit tests.
- DevOps:
    - Setting up the correct deployment workflows on our project so that our frontend and backend can deploy and build successfully on our server.
    - Creating and setting up the correct workflows for our frontend unit tests.
    - Setting up the Codecov pipeline to create and post our coverage reports.
- Integration:
    - Integrating all the view pages for the following subsystems:
        - DataScopes
        - Tasks
        - Risks
        - Access Requests
        - Asset Requests
        - Support Request
        - Devices
    - Integrating all the edit pages for the following subsystems:
        - DataScopes
        - Tasks
        - Risks
        - Access Requests
        - Asset Requests
        - Support Request
        - Devices
    - Helped make all the pages on the project asynchronous.

Chris Mittendorf

- Hosting and Deployment
    - Helped setup and upgrade and maintain the AWS EC2 instance
    - Upgraded and maintained the AWS RDS database

- Helped create the automatic deployment workflow
        - Enabled HTTPS to ensure the security of the website
        - Setup and configured all AWS secrets
    - Backend
        - Implement all functions and configurations for the AWS secrets manager
        - Implemented functions for communication to the database
    - Frontend
        - Helped with mino styling issues
        - Created the content of the Help and About pages
    - Integration
        - Helped across the different subsystems with integration issues
    - Testing
        - Setup all non-functional tests and surveys for the user testing
    - Documentation
        - Created the testing policy document
        - Updated and maintained all documents for the project

Yané van der Westhuizen

- Created/Updated designs for:
    - Dashboard/Home
    - Role Creation
    - General Tab View
    - General Pop-Up View
- Implemented new designs for:
    - Home/Dashboard
    - Role Creation
    - Users
    - Data Scopes
    - Tasks
    - Devices
    - Risks
    - Requests
    - Access Requests
    - Asset Requests
    - Support Requests
    - About
    - Navbar

- Added design and front-end implementations for:
    - Help Pop-ups
    - Searching subsystems
    - Several subsystem pop-ups (Create/Delete/Edit/Review)
    - Role Creation: Multiple views to view current system roles and create new roles.
- Fixed:
    - Display and styling for several pages and pop-ups.

Karel Smit

- Redesign of front-end systems for dynamic roles:
    - Users
    - Data Scopes
    - Devices
    - Tasks
    - Access Requests
    - Support Requests
    - Asset Requests
    - Risks
    - Requests
    - Navbar
- Created new front-end systems:
    - Forgot password
    - Change password
    - Delete pop-up
    - Help pop-up
- Updated front-end systems:
    - Support request (multiple views based on permissions)
    - Various subsystem pop-ups
    - Navbar
    - User settings
    - Home
- Fixed:
    - Styling for various pages and pop-ups
    - Display issues for various pop-ups
    - Integration for delete device
    - Integration for delete data scope

- ○ Integration for view roles

Alistair Ross

- Implementation of Backend
  - ○ Implemented majority of backend structure including associating all models and handling different database calls needed for frontend
- Role creation
  - ○ Implemented backend role creation
  - ○ Integrated with users to give users authorities
- Increased Security
  - ○ Made tokens stateless
  - ○ Implemented authority checking
- Implemented email and notification
  - ○ Implemented automatic email and notifications on the backend to send to users when something significant occurs.
- Change Password and OTP
  - ○ Implemented change password functionality
  - ○ Implemented an OTP system for forgotten password
- Multiple Database implementation
  - ○ Setup configuration for multiple database
  - ○ Setup all delete functionality
- Integration
  - ○ Laid out integration structure as well as integrated a large amount of the frontend with the backend
- Testing
  - ○ Implemented cypress tests to workflow
- Hosting
  - ○ Assisted in setting up AWS ec2 instance as well as assisting with final build procedures