# 1. Introduction

## 1.1 Purpose of the system

## 1.2 Design goals

## 1.3 Definitions, acronyms, and abbreviations

## 1.4 Overview

# 2. Proposed software architecture

## 2.1 Overview

## 2.2 Subsystem decomposition

## 2.3 Hardware/software mapping

## 2.4 Persistent data management

## 2.5 Access control and security

Every SUN user has a username and password pair for identification. When the user wants to use the mobile application, the correct username and password pair must be provided. A token-based authentication system will be utilized for authentication and authorization purposes.

As the first step of the authentication, API client of the mobile application will send the credentials of the user to the login end point of the API. Login system will perform the checks and generate a token for the user if the entered information is correct. Any further requests to the API will contain the authentication token in the HTTP header. Backend of the application will match the authentication token and respond the request.

While responding the request, the backend will honor the authorization information of the data. Private information about users or groups will not be included in the responses given to unauthorized users.

There will be benefits of using a token-based authentication over Basic HTTP Authentication or a similar strategy that sends username and password with each request. These benefits are, shrinking the window of opportunity of capturing the password, eliminating the need of saving the credentials of the user to the mobile device, easy revocation of authentication tokens.

In order to implement an auto login feature on the mobile application, authentication information must be kept on the mobile device. In case of the username and password based system, the authentication information will contain the personal password of the users and using a token-based system will prevent this.

Furthermore, if an attacker somehow captures the authentication token, generating a new token will revoke the captured one and the captured token will be useless.

The passwords of the users will be kept on the database by salting and hashing. Salts of the users will be unique and be generated using a secure random number generator. Then salt will be prepended to the user’s password and hashed with SHA256 hashing algorithm.

Finally, all data communication between mobile application and the backend will use HTTPS protocol for safety reasons.

## 2.6 Global software control

## 2.7 Boundary conditions

# 3. Subsystem services

# 4. Glossary

# 5. References