

# Supplementary Requirements Documentation

## Home Dork

### Revision History

| Name            | Associated Letter |
|-----------------|-------------------|
| Hani Alzir      | A                 |
| Ali Habesh      | B                 |
| Stiv Abdulwahed | C                 |
| Amr Al-shaaba   | D                 |

| Date       | Version | Description                          | Author  |
|------------|---------|--------------------------------------|---------|
| 2021-09-15 | 1.0     | Modifying Supplementary Requirements | A,B,C,D |
| 2021-11-16 | 1.1     | Fixing some grammar mistakes         | A,B,C,D |
|            |         |                                      |         |
|            |         |                                      |         |

# Supplementary Requirements List

| Supplementary Requirement Name       | Priority  |
|--------------------------------------|-----------|
| S1. Usability, easy to use and learn | Essential |
| S2. Reliability-Stable               | Essential |
| S3. Performance-Responsiveness       | Essential |
| S4. User interface                   | Essential |
| S5. Design language                  | Essential |
| S6. Programming language             | Essential |

# Supplementary Requirements Descriptions

## **S1**

Usability, easy to use, and learn. The main point is to provide the user with a simple GUI that is easily understandable and doesn't require more than a few minutes to be able to start using it.

## **S2**

Reliability-Stable. The website should be reliable as insecure and also stable when it comes to being up and running whenever needed.

## **S3**

Performance-Responsiveness. The application should strive to achieve as few bugs as possible in order to not affect the user and responsiveness. The web application and mobile application will need to have a fast response time making the user not wait for more than necessary. This builds on code development and threading properly.

## **S4**

User interface. The user should have a good-looking GUI that feels smooth but yet a simple look, its important the GUI doesn't look complex, rather hide the complexity but show the simple parts.

## **S5**

The design will be a necessary tool for understanding the main concept of the application, we are going to use UML class diagrams and use case diagrams.

## **S6**

The programming language should be java. Java is easy to set up and connect to servers, doesn't require more than a few lines to also be able to communicate with API.