# FURPS+

## Functionality

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Comment |
| FR-001 | Provide mobile access to platform | Web Application,  Tomcat |
| FR-002 | Provide facilities to store and access data | MySQL JDBC driver,  Java Persistence API using Spring Data and Hibernate |
| FR-003 | Provide services to protect access to certain resources or information. | Spring Security |
| FR-004 | Provide facilities to convert a case to PDF format | Researching best approach |
| FR-005 | Provide services that allow applications to send mails. | Java Mail and Spring JavaMailSender |
| FR-006 | Provide facilities for account management |  |
| FR-007 | Provide incidence report facilities | I.e. ‘Indbrud’, ‘112-skader’, ‘Samarit-skader’, ’Småskader’ |
| FR-008 |  |  |
| FR-009 |  |  |
| FR-010 |  |  |

## Usability

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Comment |
| NFR/U-001 |  |  |
| NFR/U-002 |  |  |
| NFR/U-003 |  |  |

## Reliability

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Comment |
| NFR/R-001 | Data skal gemmes to steder, så der er backup |  |
| NFR/R-002 |  |  |
| NFR/R-003 |  |  |

## Performance

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Comment |
| NFR/P-001 |  |  |
| NFR/P-002 |  |  |
| NFR/P-003 |  |  |

## Supportability

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Comment |
| NFR/S-001 | Project can be built for multiple platforms | Java |
| NFR/S-002 | Project source must be easily maintained, obtained through low-coupling. | Implements Spring MVC |
| NFR/S-003 | Web Application can run on multiple platforms | Using Tomcat |
| NFR/S-004 | Provide easy project configuration and dependency management | Using Maven |
| NFR/S-005 | Data can be persisted to any SQL store | Java Persistence API using Spring Data and Hibernate |
| NFR/S-006 | Frontend is up to standards and easily maintained | Thymeleaf templating engine |
| NFR/S-007 |  |  |
| NFR/S-008 |  |  |

## Design Constraints

A design constraint, as the name implies, limits the design — for example, requiring a relational database stipulates the approach that we take in developing the system.

## Implementation Constraints

An implementation constraint puts limits on coding or construction – standards, platform, or implementation language.

## Interface Constraints

An interface constraint is a requirement to interact with an external item. When you develop within an enterprise, quite often you must interact with external systems

## Physical Constraints

Physical constraints affect the hardware used to house the system – for example, shape, size, and weight.