# 1. Operational Concept Description (OCD)

## 1.1. Project name

*State here the name of the project, name of the team, and team’s members.*

De naam van het project is The light swarm. Ons team heet The swarm en bestaat uit 5 teamleden: Stefanie Corijn, Lieke Steenbakkers, Niels Baljon, Marijn Sopers en Rick Snijders.

## 1.2. Scope

*This section shall be divided into the following paragraphs.*

### 1.3. System overview

*[This paragraph shall briefly state the purpose of the system to which this document applies. It shall describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.]*

⇒ *Describe here the goal of the swarm and the idea behind it (detsils later on)*

De chariots (robots) gaan door een ruimte rijden. In deze ruimte is een lightbron aanwezig. De bedoeling is dat de chariots deze lightbron vinden. Wanneer de lightbron gevonden is komen alle chariots naar de lightbron en maken ze een vorm. De lightbron vinden valt onder search and rescue. Dat alle chariots naar de lightbron komen en een vorm maken is formation.

**1.4. Referenced documents**

*This section shall list the number, title, revision, and date of all documents referenced in this document.*

## 1.5. Concept for a new or modified system

*This section shall be divided into the following paragraphs to describe a new or modified system.*

### 1.5.1. Background, objectives, and scope

*This paragraph shall describe the background, mission or objectives, and scope of the new or modified system.*

### 1.5.2. Operational policies and constraints

*This paragraph shall describe any operational policies and constraints that apply to the new or modified system. N/A if not applicable*

### 1.5.3. Description of the new or modified system.

*This paragraph shall provide a description of the new or modified system, identifying differences associated with different states or modes of operation (for example, regular, maintenance, training, degraded, emergency, alternative-site, wartime, peacetime). The distinction between states and modes is arbitrary. A system may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If the system operates without states or modes, this paragraph shall so state, without the need to create artificial distinctions. The description shall include, as applicable:*

*a. The operational environment and its characteristics*

*b. Major system components and the interconnections among these components*

*c. Interfaces to external systems or procedures  
d. Capabilities/functions of the new or modified system*

*e. Charts and accompanying descriptions depicting inputs, outputs, data flow, and manual and automated processes sufficient to understand the new or modified system or situation from the user's point of view*

*f. Performance characteristics, such as speed, throughput, volume, frequency*

*g. Quality attributes, such as reliability, maintainability, availability, flexibility, portability, usability, efficiency*

*h. Provisions for safety, security, privacy, and continuity of operations in emergencies*

### 1.5.4. Users/affected personnel.

*This paragraph shall describe the types of users of the new or modified system, including, as applicable, organizational structures, training/skills, responsibilities, and interactions with one another.*

## 1.6. Operational scenarios.

*This section shall describe one or more operational scenarios that illustrate the role of the new or modified system, its interaction with users, its interface to other systems, and all states or modes identified for the system. The scenarios shall include events, actions, stimuli, information, interactions,*

*etc., as applicable. Reference may be made to other media, such as videos, to provide part or all of this information.*

## 1.7. Analysis of the proposed system. 1.7.1. Summary of advantages.

*This summary shall include new capabilities, enhanced capabilities, and improved performance, as applicable, and their relationship to deficiencies identified in*

### 1.7.2. Summary of disadvantages/limitations.

*This paragraph shall provide a qualitative and quantitative summary of disadvantages or limitations of the new or modified system. These disadvantages and limitations shall include, as applicable, degraded or missing capabilities, degraded or less-than-desired performance, greater- than-desired use of computer hardware resources, undesirable operational impacts, conflicts with user assumptions, and other constraints.*

### 1.7.3. Alternatives and trade-offs considered.

*This paragraph shall identify and describe major alternatives considered to the system or its characteristics, the trade-offs among them, and rationale for the decisions reached.*

# 2. SSS: System Subsystem Specification

## 2.1. System wide requirements

*This paragraph shall state the system wide requirements (any further specification at sub-system level or interface lever shall be stated in SRS and/or IRS).*

# 3. SRS: System Requirements Specification

## 3.1. Requirements.

*This section shall be divided into the following paragraphs to specify the system requirements, that is, those characteristics of the system that are conditions for its acceptance. Each requirement shall be assigned a project- unique identifier to support testing and traceability and shall be stated in such a way that an objective test can be defined for it. Each requirement shall be annotated with associated qualification method(s) (see section 4) and, for subsystems, traceability to system requirements (see section 5.a), if not provided in those sections. The degree of detail to be provided shall be guided by the following rule: Include those characteristics of the system that are conditions for system acceptance; defer to design descriptions those characteristics that the acquirer is willing to leave up to the developer. If there are no requirements in a given paragraph, the paragraph shall so state. If a given requirement fits into more than one paragraph, it may be stated once and referenced from the other paragraphs.*

## 3.2. Required states and modes.

*If the system is required to operate in more than one state or mode having requirements distinct from other states or modes, this paragraph shall identify and define each state and mode. Examples of states and modes include: idle, ready, active, post-use analysis, training, degraded, emergency, backup, wartime, peacetime. The distinction between states and modes is arbitrary. A system may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If no states or modes are required, this paragraph shall so state, without the need to create artificial distinctions. If states and/or modes are required, each requirement or group of requirements in this specification shall be correlated to the states and modes. The correlation may be indicated by a table or other method in this paragraph, in an appendix referenced from this paragraph, or by annotation of the requirements in the paragraphs where they appear.*

**3.3. 3.2 System capability requirements.**

*This paragraph shall be divided into subparagraphs to itemize the requirements associated with each capability of the system. A "capability" is defined as a group of related requirements. The word "capability" may be replaced with "function," "subject," "object," or other term useful for presenting the requirements.*

**3.3.1. 3.2.x (System capability).**

*This paragraph shall identify a required system capability and shall itemize the requirements associated with the capability. If the capability can be more clearly specified by dividing it into constituent capabilities, the constituent capabilities shall be specified in subparagraphs. The requirements shall specify required behavior of the system and shall include applicable parameters, such as response times, throughput times, other timing constraints, sequencing, accuracy, capacities (how much/how many), priorities, continuous operation requirements, and allowable deviations based on operating conditions. The requirements shall include, as applicable, required behavior under unexpected, unallowed, or "out of bounds" conditions, requirements for error handling, and any provisions to be incorporated into the system to provide continuity of operations in the event of emergencies.*

## 3.4. System external interface requirements.

*This paragraph shall be divided into subparagraphs to specify the requirements, if any, for the system's external interfaces. This paragraph may reference one or more Interface Requirements Specifications (IRSs) or other documents containing these requirements.*

## 3.5. System internal interface requirements.

*This paragraph shall specify the requirements, if any, imposed on interfaces internal to the system. If all internal interfaces are left to the design or to requirement specifications for system components, this fact shall be so stated. If such requirements are to be imposed, paragraph 3.3 of this DID provides a list of topics to be considered.*

## 3.6. System internal data requirements.

*This paragraph shall specify the requirements, if any, imposed on data internal to the system. Included shall be requirements, if any, on databases and data files to be included in the system. If all decisions about internal data are left to the design or to requirements specifications for system components, this fact shall be so stated. If such requirements are to be imposed, paragraphs 3.3.x.c and 3.3.x.d of this DID provide a list of topics to be considered.*

## 3.7. Adaptation requirements.

*This paragraph shall specify the requirements, if any, concerning installation- dependent data that the system is required to provide (such as site-*

*dependent latitude and longitude or site-dependent state tax codes) and operational parameters that the system is required to use that may vary according to operational needs (such as parameters indicating operation- dependent targeting constants or data recording).*

## 3.8. Safety requirements.

*This paragraph shall specify the system requirements, if any, concerned with preventing or minimizing unintended hazards to personnel, property, and the physical environment. Examples include restricting the use of dangerous materials; classifying explosives for purposes of shipping, handling, and storing; abort/escape provisions from enclosures; gas detection and warning devices; grounding of electrical systems; decontamination; and explosion proofing. This paragraph shall include the system requirements, if any, for nuclear components, including, as applicable, requirements for component design, prevention of inadvertent detonation, and compliance with nuclear safety rules.*

3.9. Security and privacy requirements**.**

*This paragraph shall specify the system requirements, if any, concerned with maintaining security and privacy. The requirements shall include, as applicable, the security/privacy environment in which the system must operate, the type and degree of security or privacy to be provided, the security/privacy risks the system must withstand, required safeguards to reduce those risks, the security/privacy policy that must be met, the security/privacy accountability the system must provide, and the criteria that must be met for security/privacy certification/accreditation.*

## 3.10. System environment requirements.

*This paragraph shall specify the requirements, if any, regarding the environment in which the system must operate. Examples for a software system are the computer hardware and operating system on which the software must run. (Additional requirements concerning computer resources are given in the next paragraph). Examples for a hardware-software system include the environmental conditions that the system must withstand during transportation, storage, and operation, such as conditions in the natural environment (wind, rain, temperature, geographic location), the induced environment (motion, shock, noise, electromagnetic radiation), and environments due to enemy action (explosions, radiation).*

## 3.11. System quality factors.

*This paragraph shall specify the requirements, if any, pertaining to system quality factors. Examples include quantitative requirements concerning system functionality (the ability to perform all required functions), reliability (the ability to perform with correct, consistent results -- such as mean time between failure for equipment), maintainability (the ability to be easily serviced, repaired, or corrected), availability (the ability to be accessed and operated when needed), flexibility (the ability to be easily adapted to changing requirements), portability of software (the ability to be easily modified for a new environment), reusability (the ability to be used in multiple applications), testability (the ability to be easily and thoroughly tested), usability (the ability to be easily learned and used), and other attributes.*

## 3.12. Design and construction constraints.

*This paragraph shall specify the requirements, if any, that constrain the design and construction of the system. For hardware-software systems, this paragraph shall include the physical requirements imposed on the system. These requirements may be specified by reference to appropriate commercial or military standards and specifications. Examples include requirements concerning:*

1. *Use of a particular system architecture or requirements on the architecture, such as required subsystems; use of standard, military, or existing components; or use of Government/acquirer-furnished property (equipment, information, or software)*
2. *Use of particular design or construction standards; use of particular data standards; use of a particular programming language; workmanship requirements and production techniques*
3. *Physical characteristics of the system (such as weight limits, dimensional limits, color, protective coatings); interchangeability of parts; ability to be transported from one location to another; ability to be carried or set up by one, or a given number of, persons.*
4. *Materials that can and cannot be used; requirements on the handling of toxic materials; limits on the electromagnetic radiation that the system is permitted to generate.*
5. *Use of nameplates, part marking, serial and lot number marking, and other identifying markings*
6. *Flexibility and expandability that must be provided to support anticipated areas of growth or changes in technology, threat, or mission.*

## 3.13. Other requirements.

*This paragraph shall specify additional system requirements, if any, not covered in the previous paragraphs. Examples include requirements for system documentation, such as specifications, drawings, technical manuals, test plans and procedures, and installation instruction data, if not covered in other contractual documents.*

# 4. SSDD: System Subsystem Design Description

## 4.1. System-wide design decisions.

*This section shall be divided into paragraphs as needed to present system- wide design decisions, that is, decisions about the system's behavioral design (how it will behave, from a user's point of view, in meeting its requirements, ignoring internal implementation) and other decisions affecting the selection and design of system components. If all such decisions are explicit in the requirements or are deferred to the design of the system components, this section shall so state. Design decisions that respond to requirements designated critical, such as those for safety, security, or privacy, shall be placed in separate subparagraphs. If a design decision depends upon system states or modes, this dependency shall be indicated. Design conventions needed to understand the design shall be presented or referenced. Examples of system-wide design decisions are the following:*

1. *Design decisions regarding inputs the system will accept and outputs it will produce, including interfaces with other systems, configuration items, and users (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is given in Interface Design Descriptions (IDDs), they may be referenced.*
2. *Design decisions on system behavior in response to each input or condition, including actions the system will perform, response times and other performance characteristics, description of physical systems modeled, selected equations/algorithms/ rules, and handling of unallowed inputs or conditions.*
3. *Design decisions on how system databases/data files will appear to the user (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is given in Database Design Descriptions (DBDDs), they may be referenced.*
4. *Selected approach to meeting safety, security, and privacy requirements.*
5. *Design and construction choices for hardware or hardware-software systems, such as physical size, color, shape, weight, materials, and markings.*
6. *Other system-wide design decisions made in response to requirements, such as selected approach to providing required flexibility, availability, and maintainability.*

## 4.2. System architectural design.

*This section shall be divided into the following paragraphs to describe the system architectural design. If part or all of the design depends upon system states or modes, this dependency shall be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design shall be presented or referenced.*

*Note: For brevity, this section is written in terms of organizing a system directly into Hardware Configuration Items (HWCIs), Computer Software Configuration Items (CSCIs), and manual operations, but should be interpreted to cover organizing a system into subsystems, organizing a subsystem into HWCIs, CSCIs, and manual operations, or other variations as appropriate.*

## 4.3. System components.

*This paragraph shall:*

1. *Identify the components of the system (HW and SW). Each component shall be assigned a name.*
2. *Show the static relationship(s) of the components in a diagram.*
3. *State the purpose of each component and identify the system-wide*

*design decisions allocated to it.*

# 5. SDD: Software Design Description

*Fill in only if further design of subsystems is required (think of class diagram, sequence diagram etc of a subsystem)*

# 6. IDD: Interface Design Description

## 6.1. Interface design.

*This paragraph shall be divided into the following subparagraphs to describe the interface characteristics of the system components. It shall include both interfaces among the components and their interfaces with external entities such as other systems, configuration items, and users. One or more interface diagrams shall be provided, as appropriate, to depict the interfaces.*

### 6.1.1. (Project unique identifier of interface).

*This paragraph shall identify an interface by project unique identifier, shall briefly identify the interfacing entities, and shall be divided into subparagraphs as needed to describe the interface characteristics of one or both of the interfacing entities. The design description shall include the following, if applicable, presented in any order suited to the information to be provided, and shall note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):*

1. *Type of interface (such as real-time data transfer, storage-and- retrieval of data, etc.) to be implemented*
2. *Sources (setting/sending entities) and recipients (using/receiving entities)*
3. *Characteristics of communication methods that the interfacing entity(ies) will use for the interface, such as:* 
   1. *Project-unique identifier(s)*
   2. *Communication links/bands/frequencies/media and their*

*characteristics*

* 1. *Message formatting*
  2. *Flow control (such as sequence numbering and buffer allocation)*
  3. *Data transfer rate, whether periodic/aperiodic, and interval*

*between transfers*

* 1. *Routing, addressing, and naming conventions*
  2. *Transmission services, including priority and grade*
  3. *Safety/security/privacy considerations, such as encryption, user*

*authentication, compartmentalization, and auditing*

1. *Characteristics of protocols that the interfacing entity(ies) will use for the interface, such as:*

*1. Project-unique identifier(s)*

1. *Priority/layer of the protocol*
2. *Packeting, including fragmentation and reassembly, routing, and*

*addressing*

1. *Legality checks, error control, and recovery procedures*
2. *Synchronization, including connection establishment,*

*maintenance, termination*

1. *Status, identification, and any other reporting features*

*e. Other characteristics, such as physical compatibility of the interfacing entity(ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)*

## 6.2. Interface identification and diagrams.

*For each interface identified, this paragraph shall state the project-unique identifier assigned to the interface and shall identify the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification shall state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams shall be provided, as appropriate, to depict the interfaces.*

### 6.2.1. (Project unique identifier of interface).

*This paragraph (beginning with 3.2) shall identify an interface by project unique identifier, shall briefly identify the interfacing entities, and shall be divided into subparagraphs as needed to describe the interface characteristics of one or both of the interfacing entities. If a given interfacing entity is not covered by this IDD (for example, an external system) but its interface characteristics need to be mentioned to describe interfacing entities that are, these characteristics shall be stated as assumptions or as "When [the entity not covered] does this, [the entity that is covered] will ...." This paragraph may reference other documents (such as data dictionaries, standards for protocols, and standards for user interfaces) in place of stating the information here. The design description shall include the following, as applicable, presented in any order suited to the information to be provided, and shall note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):*

*a. Priority assigned to the interface by the interfacing entity(ies)*

1. *Type of interface (such as real-time data transfer, storage-and- retrieval of data, etc.) to be implemented*
2. *Characteristics of individual data elements that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:* 
   1. *Names/identifiers* 
      1. *Project-unique identifier*
      2. *Non-technical (natural-language) name*
      3. *DoD standard data element name*
      4. *Technical name (e.g., variable or field name in code or*

*database)*

* + 1. *Abbreviation or synonymous names*
  1. *Data type (alphanumeric, integer, etc.)*
  2. *Size and format (such as length and punctuation of a character*

*string)*

* 1. *Units of measurement (such as meters, dollars, nanoseconds)*
  2. *Range or enumeration of possible values (such as 0-99)*
  3. *Accuracy (how correct) and precision (number of significant*

*digits)*

* 1. *Priority, timing, frequency, volume, sequencing, and other*

*constraints, such as whether the data element may be updated*

*and whether business rules apply*

* 1. *Security and privacy constraints*
  2. *Sources (setting/sending entities) and recipients (using/receiving*

*entities)*

1. *Characteristics of data element assemblies (records, messages, files,*

*arrays, displays, reports, etc.) that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:*

* 1. *Names/identifiers* 
     1. *Project-unique identifier*
     2. *Non-technical (natural language) name*
     3. *Technical name (e.g., record or data structure name in*

*code or database)*

* + 1. *Abbreviations or synonymous names*
  1. *Data elements in the assembly and their structure (number, order, grouping)*
  2. *Medium (such as disk) and structure of data elements/assemblies on the medium*
  3. *Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)*
  4. *Relationships among assemblies, such as sorting/access characteristics*
  5. *Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply*
  6. *Security and privacy constraints*

*8. Sources (setting/sending entities) and recipients (using/receiving entities)*

1. *Characteristics of communication methods that the interfacing entity(ies) will use for the interface, such as:* 
   1. *Project-unique identifier(s)*
   2. *Communication links/bands/frequencies/media and their*

*characteristics*

* 1. *Message formatting*
  2. *Flow control (such as sequence numbering and buffer allocation)*
  3. *Data transfer rate, whether periodic/aperiodic, and interval*

*between transfers*

* 1. *Routing, addressing, and naming conventions*
  2. *Transmission services, including priority and grade*
  3. *Safety/security/privacy considerations, such as encryption, user*

*authentication, compartmentalization, and auditing*

1. *Characteristics of protocols the interfacing entity(ies) will use for the*

*interface, such as:*

* 1. *Project-unique identifier(s)*
  2. *Priority/layer of the protocol*
  3. *Packeting, including fragmentation and reassembly, routing, and*

*addressing*

* 1. *Legality checks, error control, and recovery procedures*
  2. *Synchronization, including connection establishment,*

*maintenance, termination*

* 1. *Status, identification, and any other reporting features*

1. *Other characteristics, such as physical compatibility of the interfacing entity(ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)*

# 7. STD: System Test Description

## 7.1. Test preparation

*This section shall be divided into the following paragraphs.*

### 7.1.1. (Project unique identifier of test case).

*This paragraph shall identify a test by project-unique identifier, shall provide a brief description, and shall be divided into the following subparagraphs. When the information required duplicates information previously specified for another test, that information may be referenced rather than repeated.*

**7.x.y.1. Hardware preparation**

*This paragraph shall describe the procedures necessary to prepare the hardware for the test. Reference may be made to published operating manuals for these procedures. The following shall be provided, as applicable:  
a. The specific hardware to be used, identified by name and, if applicable, number b. Any switch settings and cabling necessary to connect the hardware*

*c. One or more diagrams to show hardware, interconnecting control, and data paths d. Step-by-step instructions for placing the hardware in a state of readiness*

**7.x.y.2. Software preparation**

*This paragraph shall describe the procedures necessary to prepare the item(s) under test and any related software, including data, for the test. Reference may be made to published software manuals for these procedures. The following information shall be provided, as applicable:*

*a. The specific software to be used in the test  
b. The storage medium of the item(s) under test (e.g., magnetic tape, diskette)*

*c. The storage medium of any related software (e.g., simulators, test drivers, databases)*

*d. Instructions for loading the software, including required sequence  
e. Instructions for software initialization common to more than one test case*

## 7.2. Test descriptions

### 7.2.1. (Project unique identifier of test case).

*This section shall indicate:*

*1. Requirements addressed*

1. *Prerequisite conditions*
2. *Test inputs*
3. *Expected test result*
4. *Criteria for evaluating results*
5. *Test procedure*

# 8. STR: System Test Report

## 8.1. Overview of test results.

*For each paragraph 7.2.x.y test results will be reported here.*

***8.1.x.y (Project-unique identifier of a test).***

## 8.2. Test log.

*This section shall present, possibly in a figure or appendix, a chronological record of the test events covered by this report. This test log shall include:*

1. *The date(s), time(s), and location(s) of the tests performed*
2. *The hardware and software configurations used for each test including, as applicable,*

*part/model/serial number, manufacturer, revision level, and calibration date of all hardware,*

*and version number and name for the software components used*

1. *The date and time of each test related activity, the identity of the individual(s) who performed the*

*activity, and the identities of witnesses, as applicable*

# 9. Notes.

*This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.*

# 10. Appendixes.

*Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix shall be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes shall be lettered alphabetically (A, B, etc.).*