Universidad de Guadalajara

Centro Universitario de los Valles

Software Requirements Specification (Baseline)

System for the Analysis of Nanoparticles Micrographs

Ing. Iker Ismak Toscano Santos
Ing. Diego Alejandro Morales Bravo

May 19, 2023 Ameca, Jalisco

Change control

Table 3.1: Change control

Revisión	Description	Actor	Date	Version
0.1	Preliminary version	Developer team	24/01/2023	SRS-SANM-V0.1
1	First version	Developer team	19/05/2023	SRS-SANM-V1.0

3.1.1 Introduction

Introducing a groundbreaking web-based system designed to revolutionize micrograph analysis. By harnessing artificial intelligence techniques, this platform simplifies the measurement and counting of nanoparticles. Researchers can extract valuable information from digital microscope images swiftly and accurately, eliminating the need for laborious manual analysis. With its advanced algorithms and machine learning capabilities, this system automates nanoparticle detection and offers customization options for diverse research needs.

This intelligent solution not only saves time and effort but also enhances scientific progress. Researchers can now focus on data interpretation, analysis, and critical decision-making, rather than being burdened by manual image processing. The web-based nature of the system ensures accessibility and flexibility, allowing researchers to conveniently upload and process micrographs from any location. Streamline your analysis, improve accuracy, and unlock new possibilities in materials science, nanotechnology, and biomedical research with this cutting-edge micrograph analysis system.

3.1.1.1 Purpose

This document serves to outline the specifications of functional and non-functional requirements for the development and implementation of a website dedicated to micrograph analysis. The website aims to provide a comprehensive platform for analyzing and interpreting micrographs, catering to the needs of researchers and scientists in various domains.

3.1.1.2 Scope

The scope of this project is to develop a publicly available system that supports the analysis of nanoparticles in micrographs. The system will provide tools and functionalities to facilitate the identification, measurement, and counting of nanoparticles within digital microscope images. The aim is to create a user-friendly web-based platform that can be accessed by researchers and scientists for efficient and accurate analysis of micrographs.

3.1.1.3 Definitions and abbreviations

Table 3.2: Definitions and abbreviations

Term	Definition
SAM	System for the Analysis of Nanoparticles Micrographs
SRS	Software requirements specification

3.1.1.4 References

[1] IEEE Std 830™-1998, IEEE Recommended Practice for Software Requirements Specifications.

3.1.1.5 Document overview

In order to orient the reader, this document is organized into the following sections:

The introduction provides background and important factors considered in the current process of monitoring and evaluating the academic performance of master's students.

The second section provides an overview of the system, describing the general factors included in the product and its requirements. The stakeholders addressed in this section of the document are the users involved and the system development team. The users will be able to identify the functionalities of the system, and the developers will be able to understand the software and communication constraints under which the development will proceed.

Finally, the third section describes the specific requirements of the system and the follow-up they will have during the development of the project. The elements are grouped into functional and non-functional requirements, in such a detail that facilitates the work of the development team using a natural and simple language, in order to integrate all project stakeholders into the process of building the software product.

3.1.2 General description

This section specifies the factors of interest and the functional requirements of the SAM system. To achieve this, it will be necessary to make a detailed description of the environment where the system will be implemented, the factors involved in the application space. In this way, this collected information will provide the guidelines for the development and implementation of this software.

3.1.2.1 Product perspective

The System for Analysis of Nanoparticles in Micrographs (SAM) web application is intended to be a tool to improve the current manual analysis process for the collection of micrograph data.

3.1.2.2 Product characteristics

The SAM system intends to make use of technological tools to achieve the objective set by the client. This requires: analysis, system design, as well as the database and user interface.

The product will allow the following functionalities:

1. Create user profiles
2. User login
3. Password recovery
4. Upload digital microscope images for analysis.
5. Generate reports of analyzed images
6. Consult previous reports

3.1.2.3 Roles characteristics

This section describes the types of users that will make up the system. Elements such as technical expertise and frequency of access to the system are considered for each user.

Table 3.3: Users of the system description, privileges and technical experience.

Role type	Description	Privileges	Technique experience
User	User with privileges to generate reports.	Access to all system functionalities	None

3.1.2.4 Restrictions

- The SAM system will be designed for web environments.
- The browser where the system will be tested will be Chromium v96, which includes support for many popular browsers.
- Users can access the application through an HTML5-compliant web browser on a desktop computer such as a Windows, Mac, Chromebook or Linux computer. HTML5 compliant web browsers that can be used include Google Chrome, Opera, Mozilla Firefox, Safari and Microsoft Edge.
- The speed of response is determined by the internet connectivity of the user accessing and performing an action in the system.

3.1.2.5 Assumptions and dependencies

- System requirements will change during the development of the application.
- Availability of work of the development team.
- Availability of time of the project stakeholders.

3.1.3 Specific requirements

This section outlines the specific requirements necessary to meet the client's expectations and ensure the acceptance of the product as a quality offering. The functional requirements encompass the core functionalities, including image upload, nanoparticle detection, measurement, counting, customizable parameters, and data visualization. On the other hand, the non-functional requirements focus on aspects such as usability, performance, security, compatibility, reliability, and scalability. By addressing these specific requirements, the system aims to provide a comprehensive and user-friendly platform for efficient and accurate analysis of nanoparticles in micrographs.

3.1.4 External interface

3.1.4.1 User interfaces

The user interface will have a set of windows with buttons, lists and text fields. The user interface will be displayed from the specified web browser. To access the interface you must have an internet connection.

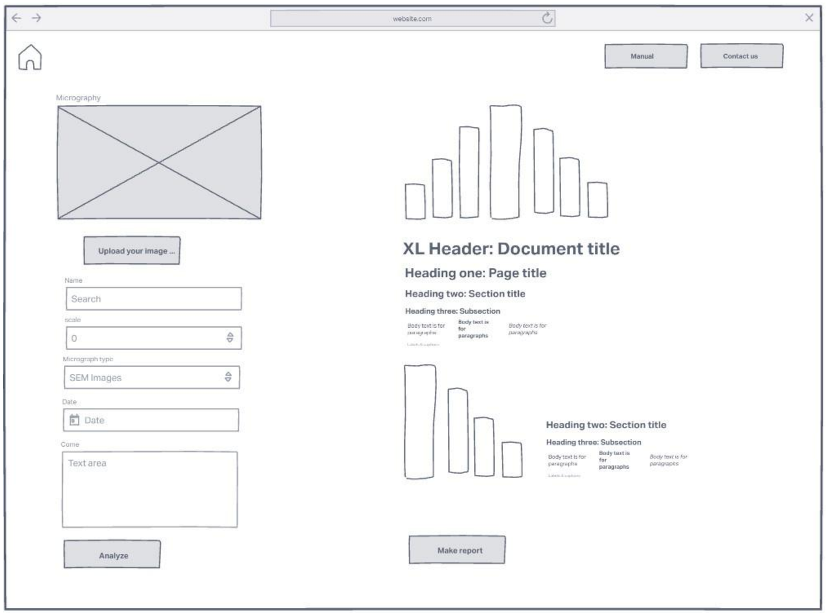
website.com
Manual
Contact us
Micrography
Upload your image ...
Name
Search
scale
0
Micrograph type
SEM Images
Date
Date
Come
Text area
Analyze
XL Header: Document title
Heading one: Page title
Heading two: Section title
Heading three: Subsection
Body text is for paragraphs
Body text is for paragraphs
Body text is for paragraphs
Heading two: Section title
Heading three: Subsection
Body text is for paragraphs
Body text is for paragraphs
Body text is for paragraphs
Make report

Figure 3.1 Análisis module, The interface features an intuitive and user-friendly loading screen for users to upload micrographs and provide accompanying descriptions for analysis and inclusion in a report.

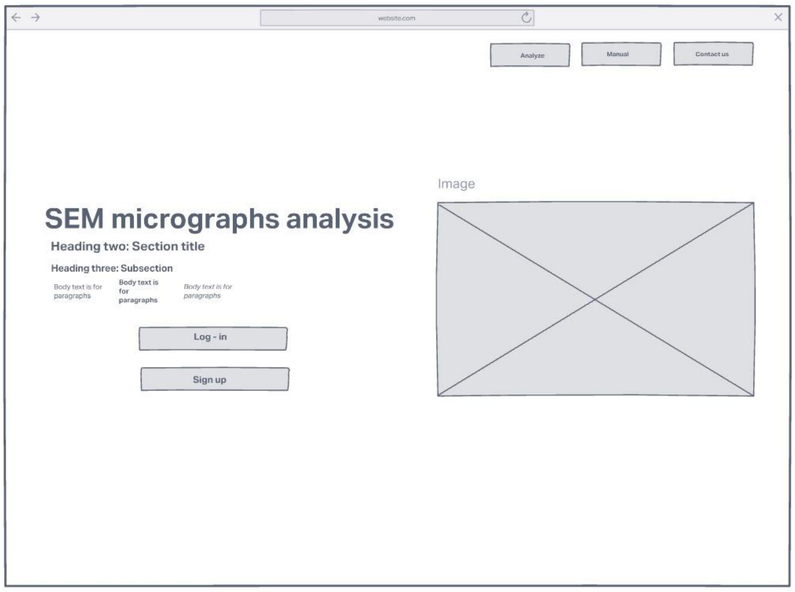
website.com
Analyze
Manual
Contact us
SEM micrographs analysis
Heading two: Section title
Heading three: Subsection
Body text is for paragraphs
Body text is for paragraphs
Body text is for paragraphs
Log - in
Sign up
Image

Figure 3.2 Login module, This view showcases the user login functionality, enabling users to securely log in and initiate their sessions within the application.

3.1.4.2 Software interfaces

Web browser: Google Chrome, Opera, Mozilla Firefox, Safari, Microsoft Edge.

3.1.4.3 Functional requirements:

- The system will be a landing page where you will find general information about the system by scrolling down.
- The system will have a section where you can log in with your username and password.
- The system will have a registration option for new users.
- The system will have a user manual section where information on how to use the system will be displayed.
- The system will have the option to enter the description of the images to be segmented.
- The system will allow you to select the type of micrographs to be analyzed from a list of options.
- The system returns the uploaded images together with the statistics resulting from the analysis.
- The system will return a histogram containing the nanoparticle count information by size.
- The system will return a downloadable report in PDF with the information of the analysis performed with the description information, date and time and user information such as name and email.
- The system will have a contact section where you will find the information of the developer, as well as the researchers involved in the project.

3.1.4.4 No-functional requirements:

- The system will be developed on the front-end using CSS for the design.
- The system will be developed with the use of HTML for the sections and information.
- The system will be developed on the back-end with python and the use of the Django framework for URL management.
- The system will have a Mongo type NoSQL database.
- The system will make use of PyTorch, Pillow and Matplot for image processing and deep learning techniques.
- The system will be in English to be more globalized.

3.1.5 Functions

This section will describe the system requirements in detail as shown in the following tables:

1. Sign up form
2. Sign up form: Email input
3. Sign up form: Name input
4. Sign up form: Occupation input
5. Sign up form: Password input
6. Log in form
7. Log in form: Email input
8. Log in form: Password input
9. Password recovery
10. Report form
11. Report form: Description
12. Report form: Upload image
13. Report generation
14. History of reports

3.1.6 Functional requirements tables

Functional Requirement 01 - Sign up form

The "Sign up form" functional requirement pertains to the implementation of a registration form within a platform. This form enables users to register by providing their name, email address, occupation, and password. Upon submitting the form, email confirmation is requested from the user. In case the email is already registered in the database, an error message is displayed.

<table>
<tr><th colspan="4">Functional requirements specification</th></tr>
<tr><td>Code</td><td colspan="2">Name</td><td>Priority level</td></tr>
<tr><td>FR-01</td><td colspan="2">Sign up form</td><td>High</td></tr>
<tr><td>Description</td><td colspan="3">A user can register by filling out and submitting the form.</td></tr>
<tr><td>Inputs</td><td>Source</td><td>Output</td><td>Restriction</td></tr>
<tr><td>+Email
+Password
+Name
+Occupation</td><td>User</td><td>> Confirmation message</td><td>> Only one user per email</td></tr>
<tr><td>Process</td><td colspan="3">By filling in the requested fields: name, email, occupation and password and submitting the form, a user is registered on the platform. Confirmation is requested by email from the user. If the email is already registered in the database, an error message will be displayed.</td></tr>
</table>

Functional Requirement 02 - Sign up form: Email input

The "Sign up form: Email input" functional requirement focuses on validating user-entered email addresses during the registration process. The system will use HTML and Javascript to verify that the input conforms to a valid email address format. If the input is invalid, an error message will be displayed to the user. This requirement ensures data integrity and enhances the user registration experience.

<table>
<tr><th colspan="4">Functional requirements specification</th></tr>
<tr><td>Code</td><td colspan="2">Name</td><td>Priority level</td></tr>
<tr><td>FR-02</td><td colspan="2">Sign up form: Email input</td><td>High</td></tr>
<tr><td>Description</td><td colspan="3">The user must fill the email field to register. The system must validate that the entry is a valid email address.</td></tr>
<tr><td>Inputs</td><td>Source</td><td>Output</td><td>Restriction</td></tr>
<tr><td>+Email</td><td>User</td><td>> Error message</td><td>*Valid emails only
*^[\w-\.]+@([\w-]+\.)+[\w-]{2,4}$
*Must be filled</td></tr>
<tr><td>Process</td><td colspan="3">When filling the email field, the system must perform a validation with HTML and Javascript that the entry corresponds to a valid email address. If this is not the case, an error message will be displayed.</td></tr>
</table>

Functional Requirement 03 - Sign up form: Name input

The Sign up form: Name input requires users to fill in their name to complete the registration process. The system validates that the input is a valid name using HTML and JavaScript. If the name is invalid, an error message is displayed. This ensures accurate and reliable user data during registration.

Functional requirements specification			
Code	Name		Priority level
FR-03	Sign up form: Name input		High
Description	The user must fill the name field to register. The system must validate that the entry is a valid name.		
Inputs	Source	Output	Restriction
+Name	User	> Error message	*Valid names only *[a-zA-ZÀ] *Must be filled
Process	When filling the name field, the system must perform a validation with HTML and Javascript that the entry corresponds to a valid name. If this is not the case, an error message will be displayed.		

Functional Requirement 04 - Sign up form: Occupation input

This feature ensures that the user provides essential information about their occupation, allowing for better categorization and understanding of their background within the sign-up process. In this functionality, the user is presented with a list of occupation options in a drop-down menu. They are required to choose the occupation that best represents their current profession or work field. If the user fails to make a selection from the available options and submits the form, an error message will be displayed, prompting them to select an occupation before proceeding.

Functional requirements specification			
Code	Name		Priority level
FR-04	Sign up form: Occupation input		High
Description	The user must select from the drop-down list the occupation that best describes him/herself.		
Inputs	Source	Output	Restriction
+Occupation	User	> Error message	*Must be filled
Process	The user selects an option from the drop-down list. If this is not the case, an error message will be displayed.		

Functional Requirement 05 - The Sign up form

Password input is a crucial component of the registration process. In order to successfully register, users are required to provide a password. The system ensures that the password entered by the user is valid and meets the necessary criteria.

Functional requirements specification			
Code	Name		Priority level
FR-05	Sign up form: Password input		High
Description	The user must fill in the password field to register. The system must validate that the entry is a valid password.		
Inputs	Source	Output	Restriction
+Password	User	> Error message	*Must be filled
Process	When filling the name field, the system must perform a validation with HTML and Javascript that the entry corresponds to a valid name. If this is not the case, an error message will be displayed.		

Functional Requirement 06 - Log in form

By entering the required information, such as their email address and password, users can successfully log in to the system. Upon successful login, the module gains the ability to analyze micrographs. However, if the provided email address is not registered in the system or the password is incorrect, a warning message will be displayed to the user.

Functional requirements specification			
Code	Name		Priority level
FR-06	Log in form		High
Description	A user can log in by filling out the form with their username and password.		
Inputs	Source	Output	Restriction
+Email +Password	User	> Confirmation message	
Process	By filling in the requested fields: e-mail and password, the user logs in, successful login enables the module to analyze micrographs. In case the email address is not registered, or the password is wrong, a warning message will be displayed.		

Functional Requirement 07 - Log in form: Email input

This requirement specifies that in order to log in, the user must provide their email address. The system needs to verify that the entered value is a valid email address. To achieve this, the system will employ HTML and JavaScript for email validation during the input process. The validation will ensure that the entered email address is both syntactically correct and matches an email registered in the database. If the entered value fails to meet these criteria, an error message will be shown to the user.

Functional requirements specification			
Code	Name		Priority level
FR-07	Log in form: Email input		High
Description	The user must fill the email field to log in. The system must validate that the entry is a valid email address.		
Inputs	Source	Output	Restriction
+Email	User	> Error message	*Valid emails only *^[\w-\.]+@([\w-]+\.)+[\w-]{2,4}$ *Must be filled
Process	When filling in the email field, the system must perform a validation with HTML and Javascript that the entry corresponds to a valid email address, and it must be registered in the database. If not, an error message will be displayed.		

Functional Requirement - Log in form: Password input 08

In order to log in, users are required to provide their email address in the designated field. The system ensures that the entered value is a valid email address by conducting validation using HTML and JavaScript. Additionally, the system checks if the provided email address is registered in the database. If the email address is invalid or not found in the database, an error message is displayed to the user.

Functional requirements specification			
Code	Name		Priority level
FR-08	Log in form: Password input		High
Description	The user must fill in the password field to log in.		
Inputs	Source	Output	Restriction
+Password	User	> Error message	*Must be filled
Process	The system must validate that the password matches the user's password.		

Functional Requirement 09 - Password recovery

Users are provided with a form where they can enter their email address. The system performs a validation process to check if the entered email address is registered in the database. If the email address exists, a password recovery link is sent to the user via email. However, if the email address is not registered, no email will be sent for password recovery.

Functional requirements specification			
Code	Name		Priority level
FR-9	Password recovery		High
Description	A user can recover his password if he forgets it.		
Inputs	Source	Output	Restriction
+Email	User	> Confirmation message	
Process	The user fills out a form where the e-mail address is entered. The system validates that the email address is registered in the database. If it exists, a link for password recovery is sent, otherwise the email will not be sent.		

Functional Requirement 10 - Report form

The Report Form is a feature that allows users to generate reports by providing specific information, including a description, image, and type of analysis. By filling in these required fields and submitting the form, users can create comprehensive reports. Once all the necessary fields are completed and the form is submitted, an alert message will be displayed to confirm the successful submission. This alert ensures that users are aware that their report has been successfully generated and can proceed accordingly.

Functional requirements specification			
Code	Name		Priority level
FR-10	Report form		High
Description	A user can generate reports by filling in the entries for: description, image and type of analysis.		
Inputs	Source	Output	Restriction
+Description +Image +Type of analysis	User	> Alert message	
Process	In case of filling in all requested fields and submitting the form, an alert message is displayed.		

Functional Requirement 11 - Description Input

The "Report Form: Description Input" functional requirement focuses on ensuring that users provide a description when filling out a form. The system is responsible for validating that the description field is not left empty. To achieve this, the system performs a validation using HTML and Javascript to ensure that the entry in the description field is filled. This requirement aims to enhance the accuracy and completeness of data captured through the form.

Functional requirements specification			
Code	Name		Priority level
FR-11	Report form: description input		Mid
Description	The user must fill the description field. The system must validate that entry is filled.		
Inputs	Source	Output	Restriction
+Description	User	> Error message	*Must be filled
Process	In the description field, the system must perform a validation with HTML and Javascript that the entry is filled.		

Functional Requirement 12 - Image Input

The "Report Form: Image Input" functional requirement focuses on requiring users to upload an image when filling out a form. The system is responsible for validating that the uploaded entry is a valid image. To accomplish this, the user is prompted to select a file with image extensions such as jpg, png, and tif. If the selected file does not have a valid extension, a warning message will be displayed. This requirement ensures that only appropriate image files are accepted, enhancing the overall quality and usability of the form.

Functional requirements specification			
Code	Name		Priority level
FR-12	Report form: image input		High
Description	The user must upload an image. The system must validate that entry is a valid image.		
Inputs	Source	Output	Restriction
+Image file	User	> Error message	*An image must be uploaded * Valid extensions (jpg, png and tif)
Process	The user selects a file with valid image extensions: jpg, png and tif. If the file is not of a valid extension a warning message will appear.		

Functional Requirement 13 - Micrographs Type Input

The "Report Form: Micrographs Type Input" functional requirement focuses on requiring users to select a micrographs type from a drop-down list when filling out a form. The system is responsible for validating that a selection is made from the available options. If the user fails to select an option from the drop-down list, an error message will be displayed. This requirement ensures that the form captures accurate and specific information regarding the micrographs type, enhancing the effectiveness and clarity of the reported data.

Functional requirements specification			
Code	Name		Priority level
FR-13	Report form: micrographs type input		High
Description	The user must select from the drop-down list the micrographs type.		
Inputs	Source	Output	Restriction
+Micrographs type	User	> Error message	*Must be filled
Process	The user selects an option from the drop-down list. If this is not the case, an error message will be displayed.		

Functional Requirement 14 - Report Generation

The "Report Generation" functional requirement enables users to generate reports within the system. This functionality allows the system to analyze requested images and present a comprehensive report containing specific information. The generated report includes the description of the image, the original image itself, the segmented image, and a graph depicting the frequency of nanoparticles relative to their area. This requirement facilitates effective data analysis and provides users with detailed insights and visual representations of the analyzed images.

Functional requirements specification			
Code	Name		Priority level
FR-14	Report generation		High
Description	A user can generate reports.		
Inputs	Source	Output	Restriction
	User	> Report	
Process	After the system analyzes the requested image, a report with the following information is displayed: Description, the original image, the segmented image, a graph of the frequency of nanoparticles versus their area.		

Functional Requirement 15 - History of Reports

The "History of Reports" functional requirement allows users to access and view all the reports they have generated within the system. The reports are organized and sorted based on the date they were generated. By accessing the report history module, users can conveniently review and retrieve their previously generated reports. This requirement enhances user experience by providing a centralized location to track and access their historical reports, facilitating easy reference and analysis of past data.

<table>
<tr><th colspan="4">Functional requirements specification</th></tr>
<tr><td>Code</td><td colspan="2">Name</td><td>Priority level</td></tr>
<tr><td>FR-15</td><td colspan="2">History of reports</td><td>High</td></tr>
<tr><td>Description</td><td colspan="3">A user can view all the reports he has generated, sorted by date.</td></tr>
<tr><td>Inputs</td><td>Source</td><td>Output</td><td>Restriction</td></tr>
<tr><td></td><td>User</td><td></td><td>* Only reports that you have made can be displayed.</td></tr>
<tr><td>Process</td><td colspan="3">In the report history module you can view all your reports by date.</td></tr>
</table>

3.1.7 Performance requirements

- The system must have the capacity to support a minimum of 30 concurrent users.
- the system must have a database manager.
- the system must have a fast response time that is not noticeable to the user and must be able to alert the user and the administrator if the response time is not met.

3.1.8 Safety requirements

- The system must keep user data private and encrypted in the database.
- The system has to be able to be flexible and adapt to different state-of-the-art browsers.

3.1.9 Reliability Requirements

- The system must be reliable by keeping all passwords and personal keys out of the browser programming and only storing them in environment variables.
- The system must validate the type of data entered in the fields is correct.

3.1.10 Availability Requirements

- The system must be available 90% of the time as there are no platforms such as Azure, Google cloud or AWS to guarantee online availability and relies on the school network.
- The database manager must be available 80% of the time.
- The database manager must be used locally (only by the system).

3.1.11 Maintainability Requirements

- The system shall have parameterisable features to allow for future maintenance (source code).
- The system shall be created in such a way that modules can be added in the future.

3.1.12 Portability Requirements

- The system must be able to run on any operating system that supports the chromium web browser version 96.
- The system will be developed in different frameworks in the back-end and front-end.

3.1.13 Design constraints

- The system should be intuitive using UX/UI conventions.
- The system must have a logo that identifies it and gives it an identity that refers to its user.

3.1.14 System attributes

- The system shall be able to send emails.
- The system shall be able to generate a PDF report.
- The system shall be able to read images in jpeg , jpg , png

3.1.15 Other requirements

- The system does not interact with another external system.
- The system shall have internal documentation to explain the operation to users.

3.1.16 Error control

- The system must have in its external programming system to avoid crashes due to different factors.
- The system shall have error messages to alert the user of a malfunction such as image loading or format error.

3.1.17 Design

In this section, the design of the software is specified using UML diagrams to represent the whole system. First we will have some general diagrams to identify the components and the scenarios that are going to be addressed by the software, after that, we will explore in detail the modules of the system.

3.1.17.1 Modules hierarchy

Nanoparticles are microscopic particles that have numerous applications in industries such as healthcare, electronics, and manufacturing. The software platform is designed to help researchers and scientists analyze microscope images to detect and characterize nanoparticles. It includes several modules, such as user registration, login, image upload, image analysis, report generation, report history, user database, and report database. The platform uses artificial intelligence algorithms to analyze microscope images and provide detailed reports on the presence and location of nanoparticles. The configuration items, such as the user and report databases, ensure that the platform can store and retrieve relevant information for each user and their analysis reports. The platform is a valuable tool for researchers and scientists working with nanoparticles, as it provides accurate and reliable analysis results in a user-friendly interface.

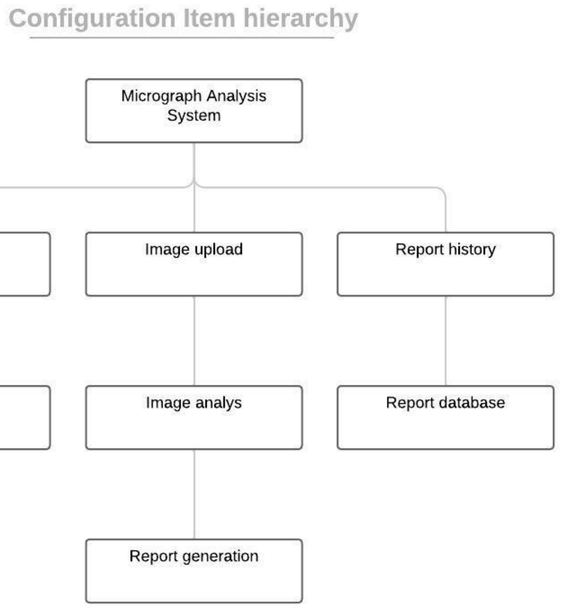

Figure 4.1 Module hierarchy Moving down through the levels, the blocks representing the modules become more specialized and specific, carrying out detailed functions with a reduced level of generality.

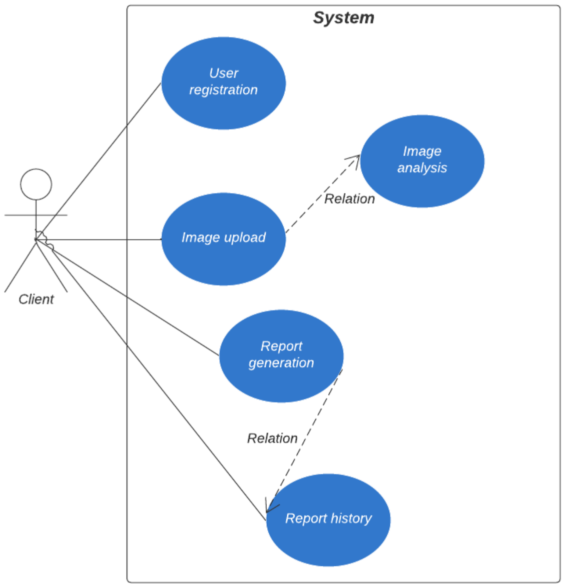

Figure 4.2 Use case users with the projects module.

3.1.18 Modules

This is a list of modules and configuration items for a software platform designed for nanoparticle analysis and detection using microscope images. The modules include user registration, login, image upload, image analysis, report generation, and report history. The configuration items include the user and report databases, which store user and report information for later access.

1. User registration module: This module would allow users to create an account on the platform by providing their personal and contact information.
2. Login module: Once registered, users could log into the platform using their email address and password.
3. Image upload module: This module would allow users to upload microscope images for nanoparticle analysis and detection.
4. Image analysis module: This module would use artificial intelligence to analyze the uploaded microscope images and detect nanoparticles.
5. Report generation module: This module would allow users to generate detailed reports on the results of the image analysis, including data on the presence of nanoparticles and their location in the image.
6. Report history module: This module would allow users to view reports generated by themselves in the past and access the analysis details for each image.
7. User database: This configuration item would allow the platform to store information about registered users.
8. Report database: This configuration item would allow the platform to store the reports generated by users for later access.