1 Context

An IT company needs to develop a project management system in order to improve its internal software development process (SDP) management, covering both the people and activities involved in developing software projects. The company uses scrum for project management.

Taking into account the number of projects and the human resources involved, and the continuous changes that occur in this context, there is clearly the need for a centralized and flexible information system. The company needs a solution that adapts to each employee's profile, providing the necessary features for each profile, maximizing the productivity and competitiveness of the organization.

The objective of this project is to build a project management solution for this company, including its resources and activities, in functionalities that should cover the following functional groups:

- Administration: manage user accounts and associate the profiles available to solution administrators, manage resources and general configuration;
- Project management: project creation and management, including the creation of activities, association of resources to projects and record of progress in activities;
- Reporting: Generate reports aligned with user profile's needs.

2 Functional Requirements

2.1 USER PROFILES

The system should allow the configuration of profiles associated with users, in order to restrict access to features. Each profile will have permissions associated with available actions and functionalities. A user account can have one or more profiles.

When registration is made, it is automatically associated with the visitor/guest profile, and then the administrator can add other available profiles.

The Administrator profile allows associating profiles with the remaining users of the system, and therefore an account with this profile should be automatically created at the time the system is installed.

The Director is a profile that allows the user to have access to all projects.

The Project Manager, Product Owner and Scrum Master profiles are special profiles, since they are not fixed profiles assigned to particular users, but the result of the roles the users have in a particular project over a certain period. For example, when creating a project on the system, a user is defined as the manager of that project, making that user project manager, remaining a "regular" user in the other projects to which is assigned.

The User profile corresponds to the resources that perform tasks/effort in the activities of the projects.

2.2 ACCESS AND AUTHENTICATION ON THE SYSTEM

2.2.1 Sign in to the system

To get access to the system a user must enter credentials in the form of email and password.

After successful authentication, the system should create a session that will allow the user to be identified in subsequent requests.

2.2.2 Logout

A mechanism should be available that allows the user to close the session on the system.

2.3 USER ADMINISTRATION

2.3.1 Register with the system

The system should make it possible to register users through a form accessible via a link in the login zone. Each user's registration must request at least the following information:

- Name;
- Function;
- E-mail;
- Password;
- Password confirmation;
- Photo (optional).

Creating accounts in the system must ensure that the email address used is unique.

When creating a new user account, the user is automatically associated with the "Visitor" profile. At the same time, the administrator is notified that an appropriate profile must be assigned to the created account.

As long as the account is not associated with any more profiles, as a visitor the user will only be able to send a request to the administrator to be given a profile.

2.3.2 Password reset

The login form should have a link that allows users to *reset* their passwords. In this form, the account email and a button must be entered. A message will be sent to the mailbox with a link allowing the user to securely exchange e-mail.

2.3.3 User Management

The system should have an administration area, which allows the system administrator to perform the following activities:

- List all system users;
- Search for users with availability of at least the following fields:
  - E-mail;
  - Profile.
- Associate user accounts with existing profiles;
- Activate and inactivate user accounts;
- Edit other user account information.

2.4 PROJECT MANAGEMENT

2.4.1 Project registration

The system should allow the registration of projects, the base entity that will serve for the association of activities and resources (users). Each project must have at least the following information:

- Code (unique alphanumerical identifier for each project);
- Project manager (user who will have the project manager profile for that record);
- Name;
- Description;
- Start Date;
- Sprint duration (initial estimate);
- Number of planned sprints;
- End Date (date it was closed);
- Customer;
- Business Sector;
- Typology (Fixed Cost / Time and materials);
- Product Owner (may change over time);
- Scrum Master (may change over time);
- Project team (may change over time);
- Project Status (Planned / Inception / Elaboration / Construction / Transition / Warranty / Closed);
- Budget (monetary amount available for resource spending).

The process of creating information relating to a project should be available exclusively for the Director profile. However, the Project Manager should be able to edit part of the information.

2.4.2 Project search

The system should be equipped with a form that allows the search of projects. Searchable fields should be at least:

- Code;
- Name;
- Date;
- Customer;
- Project status.

Project search should be accessible to all users, albeit the users should only get information of the projects they were/are involved. The Director is the only role that has access to all projects.

2.4.3 Record of activities of a project

In scrum, the requirements of the project are expressed as user stories and the set of all user stories is the “project backlog”. Almost all activities in the project are related to these user stories. User stories are created by the Product Owner (PO) and added to the project backlog.

The project timeline is divided into sprints (multiple of weeks), typically of the predefined duration, and the project team, PO and SM don't change during the sprint.

Each sprint has a "sprint backlog", i.e. the set of user stories that should be addressed during the sprint. A user story may include several tasks that have to be done for the user story to be completed. During the sprint, project team members register work done in these tasks.

During project execution, a user story that is too broad in scope may be further decomposed into several more detailed user stories. This can only be done by the PO (user stories in the project backlog - product backlog grooming) or by the team members (user stories in the sprint backlog). The later needs to be accepted by the PO at the sprint review in order to be added to the product backlog.

At the end of the sprint, the unfinished user stories in the sprint backlog and those whose implementation may have been rejected by the PO return to the product backlog.

A task must have at least the following information:

- Name of the task;
- Description of the task;
- User story;
- Start Date;
- End Date;
- Hours spent (0 at the time it is created);
- Effort estimate (start by having the initial estimate, but can be updated several times throughout the project; uses Fibonacci series for duration in hours.);
- Percentage of execution (0% if it has not yet started, up to 100% when it is completed; value calculated automatically based on the relationship between the hours spent and the estimated effort);
- Precedence (optional, list of tasks that must be completed before the start of the task);
- Type of task (Meeting, Documentation, Design, Implementation, Testing, Deployment, etc.);
- Task status (Planned / Running / Finished / Blocked);
- Responsible (human resource responsible for the execution of the task).

There may be other technical (e.g. deployment, etc.) or general/independent tasks (e.g. meetings, etc.) which may happen once or periodically. For example, there may be user stories and non-functional requirements related to system deployment, but the actual deployment takes place multiple times along the project and there may be an effort related to it. As such, there may be a task in each sprint related to deployment (e.g. changes in configuration due to implemented user stories, etc.). Meetings also have an effort and many have no relation to user stories (e.g. sprint review, sprint retrospective, etc.).

The creation of tasks is done by the team members. The users available for the activities are the team members, the PO and the SM.

2.4.4 Task update

Users associated with a project should be able to update their tasks in each project to which they are associated.

A particular scenario of updating tasks is to update the effort. As each resource spends time with a task, you must update it with the time recording used. Each update should originate a record with a comment associated with the task, where attachment files can be added.

2.4.5 Pooling resources to projects

The director should have the ability to associate resources with projects. This association can be made at the time of project creation or before each sprint. It is a scrum best practice that no resource should be allocated to more than a project at a given time, so that developers are focused on the tasks in hand. But this may not be feasible for specific technical and/or short duration tasks.

When associating a resource with a project, the following information should be specified:

- Start date;
- End date;
- Cost per hour (monetary value that the resource costs for each hour used in the project);
- Percentage of allocation (100% for full-time, or the percentage corresponding to part-time);

The system should ensure that at no time (i.e. week) is a resource associated with an allocation of more than 100% in different projects. You should also ensure that a resource is not associated with project activities that go out of the allocation time they have set.

2.4.6 View the status of activities in the project

The project activity list should have two distinct viewing modes, which can be used interchangeably or simultaneously:

- Table-shaped view;
- View in Gantt format (by sprint).

2.4.7 Allocations report

The allocations report should display information about the projects in which each resource is assigned at each time. For a specified start and end date, clear and easy-to-query information about the distribution of time by the different projects is displayed.

For the Director profile, information about the allocation of all users registered in the system should be listed.

For the Project Manager, PO and SM profiles, only the information corresponding to projects to which the user has access as a manager should be listed.

For the User profile, the information relating to himself must be displayed.

2.4.8 KPI report

The KPI report, made available to the director, the project manager and PO, must display information relating to the performance of the project, including at least the following EVM indicators (Earned Value Management):

- SPI (Schedule Performance Index);
- CPI (Cost Performance Index).

Calculations involving budget should take into account the budget of the project and the hourly cost of human resources (users) who have carried out activities therein.

The reports may be generated on screen or exported to an XML file.

2.5 SCRUM RECORDKEEPING

The minutes of some scrum ceremonies are to be kept on the system, i.e. sprint planning, sprint review and sprint retrospective.

2.6 LEGACY DATA INPUT

The system will allow for the import of projects created in a legacy platform. Projects violating current business rules should not be imported.

3 Non-functional Requirements

3.1 SESSION TIMEOUT

If there is a period of inactivity of more than 30 minutes, the session should be automatically closed.

3.2 LOGGING

All user operations, as well as system messages, either warning or error, must be saved in *log files on the server,* identified with at least the following levels: *debug, info, warn, error.*

3.3 SAVED PASSWORDS

Passwords should not be openly saved to the database allowing them to be read or decrypted by a user with access to the database.

3.4 LANGUAGE AND LOCATION

All text, as well as date formats and decimal formats, must be configured so that you can have more than one language in the application and changed in *runtime.*

3.5 RESPONSIVE LAYOUT

The development of the user interface should be responsive, that is, adapt the design of the forms to the type of device that is used by the user. For different devices, with different screen sizes, such as a computer and a mobile phone, the layout should adapt and display the information in a different layout to maximize the usability and user experience of the application.

3.6 REGISTRATION AND AUTHENTICATION

3.6.1 Security in the registration process

Security mechanisms should be implemented in the process of registering users in the system.

3.6.2 Email registration activation

The user registration process may become more secure if it is supplemented by activating the account via email. When you create a new record, it is created as inactive. At the same time an e-mail message is sent to the user that will allow them to activate the account, through a code or an automatic activation link.

4 User stories

US001 - As Non-Registered User, I want to register as an application user.

US002 - As Non-Registered User, I want to activate a just registered user account.

US003 - As Visitor, I want to send a request to the administrator to assign him/her a given profile.

US004 - As Administrator, I want to search for users

US005 - As Director, I want to register/create a new project.

US006 - As Administrator, I want to update profiles assigned to a user account.

US007 - As Director, I want to associate a human resource (user) to a project.

US008 - As Project Manager, I want to edit some project information.

US009 - As Product Owner, I want to create a user story and add it to the Product Backlog.

US010 - As Authenticated User, I want to update its own data (e.g., photo, function)

US011 - As Authenticated User, I want to change his/her password.

US012 - As Director, I want to create a new project typology

US013 - As Administrator, I want to create user profiles

US014 - As Director, I want to define the SM of a project

US015 - As Director, I want to get a list of all projects

US016 - As Director/PM/PO/SM, I want to view status of activities in a project

US017 - As Authenticated User, I want to get a list of all projects I'm currently allocated to

US018 - As PO/SM/TeamMember, I want to consult the product backlog, i.e. to get the list of user stories sorted by priority.

US019 - As TeamMember, I want to estimate the effort of a user story

US020 - As Product Owner, I want to refine a broad user story of the ProductBacklog into more focused user stories

US021 - As Product Owner, I want to change the priority of a user story in the product backlog.

US022 - As Project Manager, I want to create a sprint

US023 - As Team Member, I want to add a user story in the product backlog to the sprint backlog

Version	Description
0	Base version to be used for database modelling
1	Base version to be used in the project
2	Updated for sprint 2 – 10 new user stories