# **Resource Manager**

# Requirements Specification

## Introduction

Resource Manager is a system that enables professional services teams to improve project resource management by allowing staff to identify their skills. Those skills will then be assigned to the projects that require them. This system was requested by Micro Focus to avoid hiring new team members where existing members can appropriately fulfil the requirements.

## System-Wide Functional Requirements

### Auditing

All modification to system data must be audited. This includes the creation, modification and deletion of projects and skills as well as the allocation of users/skill to projects. These events are audited to ensure that all actions are accounted for by the individual user, being admin or not. Audit data includes user, action type (add [A], edit [E], delete [D]), what changes were made and the date/time that the change was made.

### Security

Only authorized users will be able to access the system and its data.

* **Authentication**: Users must enter a username and password to identify themselves to the system. This will give the user access to view and/or alter data (depending on their User Type – see next bullet point).
  + Passwords: The minimum length of a password to access the system will be 8 characters. The password must contain at least one letter and one number (alpha-numeric).
* **User types**: There will be multiple user types and each user type will have its own security rights within the system. The user types that will be supported consist of Consultant, Project Manager and System Administrator.
* **Access**: All user types will have full access to project data. By default, Project Managers will be able to access other user’s data.

### Integrity

Data should be checked for validity before being used by the system.

* **Fault trapping (I/O)** – Needs to validate user input for the front end, should prevent users from submitting incomplete or obviously incorrect data. Should prompt users for confirmation for potentially dangerous operations such as delete. Back end should validate user input again, should not allow any operations if required data is missing, corrupt or is not validated with user's authorization.
* **Bad data trapping** – The system must be fail-fast, not allowing changes to be made or data to be corrupted from a partial transaction if an error occurs.
* **Data integrity** – All data should be uniquely identifiable and were referential should not become orphaned when referred data is removed or updated.

## System Qualities

### 3.1 Usability

* Localization: Only English is required.
* The system should be easy enough to learn that once data has been entered into the system by a person, the process for future data entries are the same.
* *For other usability components please see component 4.1 User Interfaces*

### 3.2 Availability

* The systems primary hours of use are between 9 am and 5 PM AEST.
* The system should be available from within the office only.
* The system should be accessible without being connected to the company’s network.

### 3.3 Compatibility

* The system should be accessible from any desktop computer or laptop.
* Mobile devices or any other device are not supported.
* The application must be accessible from anywhere with a connection to the internet.

### 3.4 Reliability

* The system should have a 90% uptime throughout a year, including downtimes introduced due to upgrades.
* The system should be able to support up to 100 simultaneous users.
* The system should be able to have no more than one outage every 2 weeks.

### 3.5 Recovery

* The system should be able to be restored within 4 hours.
* Full backups should be done daily with hourly incremental backups.

### 3.6 Performance

* The system needs to have a maximum response time of 3 seconds when performing updates except for bulk imports of data.
* The system needs to return a full page of results in less than 10 seconds.
* The system needs to display reports within 30 seconds.

### 3.7 Capacity

* The system should be able to have up to 10 GB of data stored in its database
* The system must be able to support 100 concurrent users submitting one request every 10 seconds
* Year to year growth should be approx. 20%

### 3.8 Supportability

* The system should be built with the idea to be able to expand to support a larger user base as the system grows.
* The system should be built with knowledge that data storage requirements grow over time.

### 3.9 Security

* The system should communicate data only over secure channels such as HTTPS.

## System Interfaces

### System Interfaces

#### 4.1.1 Look & Feel

There are no specific look and feel requirements however all interfaces should be consistent across the system.

#### 4.1.2 Layout and Navigation Requirements

Navigation should be controlled through either a navigation tool on the top or a side, using the primary area of the screen for displaying data and buttons.

#### 4.1.3 Consistency

All user interfaces should follow the same design principles.

#### 4.1.4 User Personalization & Customization Requirements

Users should not be able to customize any components of the product.

### Interfaces to External Systems or Devices

There are no requirements to link to any external systems or devices, however if sample exports are provided from existing systems, they should be able to be imported into the system. For instance, the capability to ingest data from some existing systems, such a list of users or projects.

## Business Rules

### 5.1 Skill Set Management

* If a user enters a skill which contains the same word as another entered skill, the user should be asked if that skill is the same skill and the system will use that skill instead
* If a user enters a unique skill, it should be added to a list which other users can select it from.

### 5.2 Project Management

* If a project manager assigns a resource to a project that causes an overlap in that resource’s utilization, a warning should be given to the project manager.
* If a project has white space (space where a skillset is needed which is not currently filled), it should be highlighted to the project manager when looking at existing projects so that it can be easily identified when looking at projects.

### 5.3 User Management

* If a user is made inactive all their currently booked time should be removed from projects
* If a user is booked into two projects at the same time, they should be informed via a notification
* If a user tries to enter the same skill twice it should inform the user that they already have that skill and that no new skill will be added to the system

## System Documentation

There should be User Documentation that describes how to use the system for each user type.  
There should be Administrator Documentation that explains how to configure and set-up the system.