

VISHNU D1.0 - General specifications



COLLABORATORS

	<i>TITLE :</i> VISHNU D1.0 - General specifications	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY	Benjamin Isnard, Daouda Traoré, and Eugène Pamba Capo-Chichi	December 14, 2010

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME
01	07/12/2010	Formatting example	B.Isnard
02	13/12/2010	Pre-delivrable	B.Isnard

Contents

1	Document presentation	1
1.1	Document objectives	1
1.2	Document structure	1
1.3	References	1
1.4	Glossary	1
2	Use cases for User Management System (UMS)	2
2.1	Use case descriptions	2
2.1.1	U1 - Session with manual closure	2
2.1.2	U1.1 - Open session	2
2.1.3	U1.2 - Close session	3
2.1.4	U1.3 - Execute synchronous user request	3
2.1.5	U1.3.1 - Configure Option	4
2.1.6	U1.3.2 - Display options	4
2.1.7	U1.3.3 - Change password	4
2.1.8	U1.3.4 - Display session command history	5
2.1.9	U1.3.5 - Display sessions log	5
2.1.10	U1.4 - Execute asynchronous user request	5
2.1.11	U1.5 - Reconnect to session	6
2.1.12	U2 - Session with automatic closure on timeout	6
2.1.13	U2.1 - Close session auto	7
2.1.14	U3 - Session with automatic closure on disconnect	7
2.1.15	U4 - Create new local user config	8
2.1.16	U4.1 - Update local user config	8
2.1.17	U4.2 - Delete local user config	9
2.1.18	U4.3 - Display local user configs	9
2.1.19	UA1 - Create new user account	9
2.1.20	UA1.1 - Update user account	10
2.1.21	UA1.2 - Delete user account	10
2.1.22	UA2 - Reset user password	10

2.1.23	UA3 - Save configuration	11
2.1.24	UA4 - Restore configuration	11
2.1.25	UA5.1 - Display sessions	12
2.1.26	UA5.2 - Display users	12
2.1.27	UA5.3 - Display local users configs	12
2.2	Use case diagrams	13
2.2.1	UC UMS Admin	13
2.2.2	UC UMS User Auto	14
2.2.3	UC UMS User Manual	15
2.2.4	UC UMS User account	15
2.3	Data dictionary	16
3	Use cases for Tasks Management System (TMS)	17
3.1	Use case descriptions	17
3.1.1	T1 - AsyncCommandOnMachine	17
3.1.2	T1.1 - SubmitJob	17
3.1.3	T2 - SyncCommandOnMachine	18
3.1.4	T2.1 - GetJob	19
3.1.5	T2.2 - CancelJob	19
3.1.6	T2.3 - ListQueue	20
3.1.7	T2.4 - ListJobs	20
3.1.8	TA1 - SetMachineRefreshPeriod	20
3.1.9	TA2 - SetMachineEnv	21
3.1.10	TA3 - LaunchTmsServer	21
3.2	Use case diagrams	22
3.2.1	UC TMS Overview	22
3.3	Data dictionary	22

List of Figures

2.1	UC UMS Admin	13
2.2	UC UMS User Auto	14
2.3	UC UMS User Manual	15
2.4	UC UMS User account	15
3.1	UC TMS Overview	22

Chapter 1

Document presentation

1.1 Document objectives

This document presents the external specifications of the Vishnu system at a general level. At this level, we describe the interaction of a user with the system without providing implementation details. The different steps that constitute the scenario are detailed as well as the content of the messages exchanged. The main objective is to describe the system from the user point of view.

These general specifications are a prerequisite for the detailed specifications step in the software development process.

1.2 Document structure

The document is divided into 4 parts corresponding to the 4 modules that compose the Vishnu system:

- UMS: Users Management System
- TMS: Tasks Management System
- FMS: Files Management System
- IMS: Information Management System

Each module corresponds to a chapter in the document, and each chapter contains two sections:

- A first section containing "Use case descriptions" that follow the standard UML description of a use case
- A second section containing the "Use case diagrams" that describe the organization of the different use cases. These diagrams follow the UML2.0 standard.

1.3 References

1.4 Glossary

Chapter 2

Use cases for User Management System (UMS)

2.1 Use case descriptions

2.1.1 U1 - Session with manual closure

Title	U1 - Session with manual closure
Summary	The user opens a new session and closes it manually
Actors	User
Precondition	<ul style="list-style-type: none"> - The user is authenticated - VISHNU is installed and running on the client system
Postcondition	<ul style="list-style-type: none"> - The session state is closed - A session log has been created - All user requests submitted within the session are completed
Base sequence	<ol style="list-style-type: none"> 1. Include::U1.1 Open session 2. System is ready to process user commands 3. Include::U1.2 Close session (before the maximum inactivity delay if option CLOSE_POLICY is equal to CLOSE_ON_TIMEOUT)
Branch sequence	<ol style="list-style-type: none"> 2a. U1.3 Execute synchronous user request 2b. U1.4 Execute asynchronous user request 2c. U1.5 Reconnect to session
Exception sequence	<ol style="list-style-type: none"> 1a. Include::U1.1 exceptions 3a. If session cannot be closed due to running commands, user must wait until all commands are completed before trying step 3 again
Extensions	<ul style="list-style-type: none"> U1.3 - Execute synchronous user request U1.5 - Reconnect to session U1.4 - Execute asynchronous user request

2.1.2 U1.1 - Open session

Title	U1.1 - Open session
Summary	The user opens a session
Actors	User
Precondition	<ul style="list-style-type: none"> - The user is connected on a client host in which VISHNU is installed and that can be connected to the VISHNU infrastructure

Postcondition	<ul style="list-style-type: none"> - A session is active - The user's environment contains a session certificate
Base sequence	<ol style="list-style-type: none"> 1. User provides login, password and optionnally the way of closing the session manually or automatically (on disconnect or on timeout) to the "connect" command (when the default option is not set the closing mode is manual) 2. System validates login, password (User is authenticated) and optionnally, the name of the closing mode (CLOSE_ON_DISCONNECT or CLOSE ON TIMEOUT) if the SESSION_CLOSE_POLICY is set. 3. System creates the session and activates it 4. System provides the session certificate to the user
Branch sequence	<ol style="list-style-type: none"> 2a. If the password is a temporary password (after reset by the administrator) the System asks the user to enter a new password, then asks for a confirmation, and registers the new password if both steps are ok. If non-interactive request then this is an exception (a change password request is required).
Exception sequence	<ol style="list-style-type: none"> 2a. The user login is unknown 2a1. The System returns an error message 2b. The user password is invalid 2b1. The System returns an error message 2c. The SESSION_CLOSE_POLICY option is unknown 2c1. The System returns an error message 2d. VISHNU infrastructure is unreachable or unavailable 2d1. The System returns an error message 2e. The user password is temporary and request is non-interactive: the System returns an error message

2.1.3 U1.2 - Close session

Title	U1.2 - Close session
Summary	The user closes the session manually
Actors	User
Precondition	<ul style="list-style-type: none"> - The user is connected on the client system - The user has an open session on the client system
Postcondition	<ul style="list-style-type: none"> - The session is closed - A session log has been created - All user requests submitted during the session are completed
Base sequence	<ol style="list-style-type: none"> 1. The System checks that there are no running commands within the session 2. The System closes the session 3. The System informs the user that the session has been closed
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 1a. If there are running commands within the session, the System informs the user that the session can not be closed

2.1.4 U1.3 - Execute synchronous user request

Title	U1.3 - Execute synchronous user request
Summary	The user submits a synchronous request to the System
Actors	User

Precondition	- A session (for the current user and client host) is active
Postcondition	- The request is completed - A request log is created
Base sequence	1. The user sends the request to the System 2. The System returns the results to the user
Branch sequence	
Exception sequence	1.a Invalid session (bad session certificate or unavailable session) 1.b Invalid request 1.c Permission denied (admin request issued by normal user) 1.d Ressource not available 1.e VISHNU System crashed
Extension of	U1 - Session with manual closure U3 - Session with automatic closure on disconnect U2 - Session with automatic closure on timeout

2.1.5 U1.3.1 - Configure Option

Title	U1.3.1 - Configure Option
Summary	The user wants to modify the value of an option attached to his/her VISHNU account
Actors	User
Precondition	
Postcondition	
Base sequence	1. The user sends a request for modifying the value of an option to the System 2. System registers the new value for the option 3. System returns an acknowledgment to the user
Branch sequence	
Exception sequence	2a. Invalid option name 2b. Invalid option value

2.1.6 U1.3.2 - Display options

Title	U1.3.2 - Display options
Summary	The user displays all options concerning his/her VISHNU account
Actors	User
Precondition	
Postcondition	
Base sequence	1. The user sends a request to list all his/her options 2. The System returns all options of the user
Branch sequence	
Exception sequence	

2.1.7 U1.3.3 - Change password

Title	U1.3.3 - Change password
Summary	The user wants to change his/her password
Actors	User
Precondition	
Postcondition	- The password is changed

Base sequence	<ul style="list-style-type: none"> - The user sends a request containing a new password - The System registers the new user's password - The System returns an acknowledgment to the user
Branch sequence	
Exception sequence	

2.1.8 U1.3.4 - Display session command history

Title	U1.3.4 - Display session command history
Summary	The user displays all the commands sent during one session
Actors	User
Precondition	
Postcondition	
Base sequence	<ol style="list-style-type: none"> 1 - The user sends a request containing the session id 2 - The System returns the list of all commands issued by the user during the session which id corresponds to the provided session id. Each command has exactly the same format and parameters as the original submission and can be resubmitted as-is to the System.
Branch sequence	
Exception sequence	2a - Invalid session id (unknown / belonging to another user, if the current user is not an administrator)

2.1.9 U1.3.5 - Display sessions log

Title	U1.3.5 - Display sessions log
Summary	The user displays his/her sessions (active or closed)
Actors	User
Precondition	
Postcondition	
Base sequence	<ul style="list-style-type: none"> - The user sends a request to list all his/her sessions (active and/or closed) that have an open timestamp within an interval provided by the user (start and finish date) - The System returns all (or only active, or only closed) sessions of the user matching the search criteria with the following information for each session: id, date of opening, client host name, closure policy (timeout or disconnect), time before automatic closure (if applicable)
Branch sequence	
Exception sequence	

2.1.10 U1.4 - Execute asynchronous user request

Title	U1.4 - Execute asynchronous user request
Summary	The user submits an asynchronous request to the system
Actors	User
Precondition	- A session (for the current user and client host) is active
Postcondition	<ul style="list-style-type: none"> - The request is completed - A request log is created

Base sequence	<ol style="list-style-type: none"> 1. The user sends the request to the system 2. The System returns an acknowledgment to the user 3. The System runs the request in the background 4. When the request is completed the system updates the status of the request
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 1.a Invalid session (bad session certificate or session unavailable) 1.b Invalid request 1.c Permission denied 1.d Ressource not available 1.e VISHNU System crashed
Extension of	U1 - Session with manual closure U2 - Session with automatic closure on timeout U3 - Session with automatic closure on disconnect

2.1.11 U1.5 - Reconnect to session

Title	U1.5 - Reconnect to session
Summary	The user wants to use a session in which he/she was disconnected previously without closing it
Actors	User
Precondition	- The user is connected on a client host in which VISHNU is installed and that can be connected to the VISHNU infrastructure
Postcondition	- A session is active - The user's environment contains a session certificate
Base sequence	<ol style="list-style-type: none"> 1. User provides its login, password and the short identifier of the session (in fact, for each session, a short identifier is assigned) to the System 2. The System validates the user's login, password and the identifier of the session 3. The System provides the chosen session certificate to the user
Branch sequence	1a. If the user is already within an active session then go to step 3 directly. The current session will be automatically closed according to the UC U2 or U3 depending on the close policy for that session.
Exception sequence	cf U1.1 (Open session) 2.f The identifier of the session is nonexistent 2f1. The System returns an error message 2.g The identifier relates to a session belonging to another user 2g1. The System returns an error message 2.h The identifier is for a session closed 2h1. The System returns an error message
Extension of	U1 - Session with manual closure U2 - Session with automatic closure on timeout U3 - Session with automatic closure on disconnect

2.1.12 U2 - Session with automatic closure on timeout

Title	U2 - Session with automatic closure on timeout
Summary	The user opens a new session that is closed by the System after the inactivity delay has expired

Actors	User
Precondition	<ul style="list-style-type: none"> - VISHNU is installed and running on the client system - The client system can be connected to VISHNU - The option SESSION_CLOSE_POLICY is CLOSE_ON_TIMEOUT (either user option is defined or this is the default policy)
Postcondition	<ul style="list-style-type: none"> - A session log has been created - The session state is closed - All user requests submitted during the session are complete
Base sequence	<ol style="list-style-type: none"> 1. U1.1 Open session 2. The System is ready to process user commands 3. After inactivity delay has expired: U1.3 Close session auto
Branch sequence	<ol style="list-style-type: none"> 2a. U1.3 Execute synchronous user request 2b. U1.4 Execute asynchronous user request 2c. U1.5 Reconnect to session 2d. If the user disconnects from the client terminal or the client system crashes or is shutdown, the session remains open and all asynchronous commands that were not completed are kept running
Exception sequence	see U1
Extensions	U1.5 - Reconnect to session U1.4 - Execute asynchronous user request U1.3 - Execute synchronous user request

2.1.13 U2.1 - Close session auto

Title	U2.1 - Close session auto
Summary	The session is closed by the system
Actors	
Precondition	<ul style="list-style-type: none"> - The user is connected on the client system - The user has an open session on the client system either the inactivity timeout for the session has expired or the user disconnected from its shell session
Postcondition	<ul style="list-style-type: none"> - The session is closed - The session close event is stored in the system's log
Base sequence	<ol style="list-style-type: none"> 1. The system checks if the user has got no running commands (file transfers or tasks) 2. The system registers session closure
Branch sequence	<ol style="list-style-type: none"> 1a. If the user has got some running commands, the system does not close the session and reset the inactivity timeout. After this delay is expired, back to step 1.
Exception sequence	

2.1.14 U3 - Session with automatic closure on disconnect

Title	U3 - Session with automatic closure on disconnect
Summary	The user opens a new session that will be closed by the system after the user disconnects from the client terminal
Actors	User

Precondition	<ul style="list-style-type: none"> - VISHNU is installed and running on the client system - The client system can be connected to VISHNU - The option SESSION_CLOSE_POLICY is CLOSE_ON_DISCONNECT (either user option is defined or this is the default policy)
Postcondition	<ul style="list-style-type: none"> - A session log has been created - The session state is closed - All user requests submitted during the session are complete
Base sequence	<ol style="list-style-type: none"> 1. U1.1 Open session 2. System is ready to process user commands 3. The user disconnects from the terminal (either by typing an exit command, by closing the window, or by remote disconnection) 4. U1.3 Close session auto
Branch sequence	<ol style="list-style-type: none"> 2a. U1.4 Execute synchronous user request 2b. U1.5 Execute asynchronous user request 3a. if the client system crashes or is shutdown, go to step 4
Exception sequence	see U1
Extensions	U1.3 - Execute synchronous user request U1.5 - Reconnect to session U1.4 - Execute asynchronous user request

2.1.15 U4 - Create new local user config

Title	U4 - Create new local user config
Summary	The user creates a new local user config for a given user on a given machine
Actors	User
Precondition	<ul style="list-style-type: none"> - The user has an account on VISHNU - The user has no local user config defined for the machine
Postcondition	<ul style="list-style-type: none"> - Local user config is registered - An email is sent to the user with a message containing a SSH public key
Base sequence	<ol style="list-style-type: none"> 1. The user provides local user config information for a given machine: user login (unix), home directory 2. The System generates a ssh private/public key pair 3. The System sends an email to the user containing the public key and asking the user to add this key to the authorized_keys on the machine 4. The user updates his/her authorized_keys file on the machine by adding the public key
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 1a. Invalid login 3a. Invalid email address

2.1.16 U4.1 - Update local user config

Title	U4.1 - Update local user config
Summary	The user updates his/her local user config for a given machine
Actors	User
Precondition	<ul style="list-style-type: none"> - The user has an account on VISHNU - The user has a local user config defined for the machine

Postcondition	- local user config is updated
Base sequence	1. The user provides the identifier of his/her local config and information to be updated 2. The System updates the local user account information 3. The System returns an acknowledgment to the user
Branch sequence	
Exception sequence	1a. Invalid account (unknown or inactive)

2.1.17 U4.2 - Delete local user config

Title	U4.2 - Delete local user config
Summary	The user deletes his/her local user config on a given machine
Actors	User
Precondition	- The local user config exists for the given machine - There is no job or file transfer running on the given machine
Postcondition	- The local user config for the given machine is deleted
Base sequence	1. The user provides the identifier of the local user config 2. The System deletes the local user config identified 3. The System returns an acknowledgment to the user
Branch sequence	
Exception sequence	1a. Invalid login (unknown or inactive)

2.1.18 U4.3 - Display local user configs

Title	U4.3 - Display local user configs
Summary	The user displays all of his/her local configs
Actors	User
Precondition	
Postcondition	
Base sequence	1. The user sends a request to list all his/her local configs 2. The System returns all local configs
Branch sequence	
Exception sequence	

2.1.19 UA1 - Create new user account

Title	UA1 - Create new user account
Summary	The administrator creates a new user account in the System (database)
Actors	Admin
Precondition	- The user has not an account on VISHNU
Postcondition	- The user account is created and in an active state - The account's password must be changed at the first connection

Base sequence	<ol style="list-style-type: none"> 1. The administrator provides the new user's last name, first name, email address and specifies whether the user has standard or admin rights 2. The System creates the new user account and initializes the password with a random-generated string (temporary password) 3. The System sends an email to the user containing the temporary password 4. The System returns an acknowledgment to the Admin
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 2a. Login already used by another active user 3a. Invalid email address

2.1.20 UA1.1 - Update user account

Title	UA1.1 - Update user account
Summary	The administrator updates the user account (database)
Actors	Admin
Precondition	- The user has an account on VISHNU
Postcondition	- The user account is updated
Base sequence	<ol style="list-style-type: none"> 1. The administrator provides the user's information changes 2. The System updates user account (database) 3. The System returns an acknowledgment to the administrator
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 1.a Invalid login or login unknown 1.b The user information set are invalid 1.c The user information set are not consistent with the System (inconsistency between user id and login)

2.1.21 UA1.2 - Delete user account

Title	UA1.2 - Delete user account
Summary	The administrator deletes a user account
Actors	Admin
Precondition	<ul style="list-style-type: none"> - The user has an account on VISHNU - There is no job or file transfer running for the user
Postcondition	<ul style="list-style-type: none"> - The user account is no longer in the System - System does not contain any information related to the user
Base sequence	<ol style="list-style-type: none"> 1. The administrator provides a user's login 2. The System deletes the user account together with all the information (configuration, history) related to it. 3. The System returns an acknowledgment to the administrator
Branch sequence	
Exception sequence	1a. Invalid login (unknown or inactive)

2.1.22 UA2 - Reset user password

Title	UA2 - Reset user password
Summary	The administrator resets a user password

Actors	Admin
Precondition	
Postcondition	- The password of the user is temporary and must be changed at the first connection by the user
Base sequence	<ol style="list-style-type: none"> 1. The administrator provides a user's login 2. The System resets the user's password using a random-generated string 3. The System sends an email to the user containing the new temporary password 4. The System returns an acknowledgment to the administrator
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 1a. Invalid login (unknown or inactive) 3a. Invalid email address

2.1.23 UA3 - Save configuration

Title	UA3 - Save configuration
Summary	The administrator saves the configuration of the system
Actors	Admin
Precondition	
Postcondition	<ul style="list-style-type: none"> - The configuration is saved on a file - A log information is created
Base sequence	<ol style="list-style-type: none"> 1. The administrator requests to save the configuration in a file 2. The System creates a configuration file containing: the list of users, the list of local users configs and the list of machines according to the local users configs 3. The Systems saves the file on the client host
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 2a. File creation problems 2b. VISHNU System crashed

2.1.24 UA4 - Restore configuration

Title	UA4 - Restore configuration
Summary	The administrator restores a saved configuration
Actors	Admin
Precondition	<ul style="list-style-type: none"> - All users are disconnected from VISHNU - The configuration file was saved using the "save configuration" feature.
Postcondition	- The System is operational on all the machines that are both properly configured in the saved configuration and where the VISHNU processes are running.
Base sequence	<ol style="list-style-type: none"> 1. The administrator opens a session as the Root user 2. The administrator checks that there is no other user/admin connected to VISHNU 3. The administrator loads the configuration file 4. The System replaces the current configuration with the loaded configuration.
Branch sequence	

Exception sequence	<p>1a. If the Root user already has an open session, the System cannot open a second session with the Root user</p> <p>3a. If the configuration file cannot be loaded, the System provides an error message. The System configuration is reset to a basic configuration with only the Root user configured.</p>
--------------------	---

2.1.25 UA5.1 - Display sessions

Title	UA5.1 - Display sessions
Summary	The administrator displays all past and present sessions stored in the database.
Actors	Admin
Precondition	
Postcondition	
Base sequence	<ul style="list-style-type: none"> - The administrator sends a request to list all sessions (active and/or closed) that have an open timestamp within an interval provided by the user (start and finish date) - The System returns the list of sessions that match the search criteria and provides detailed information about these sessions (user id, open and close timestamp, client machine id)
Branch sequence	
Exception sequence	

2.1.26 UA5.2 - Display users

Title	UA5.2 - Display users
Summary	The administrator displays the description of all users registered in the database
Actors	Admin
Precondition	
Postcondition	
Base sequence	<ol style="list-style-type: none"> 1. The administrator sends a request to list all users 2. The System returns all users with the following information for each user: id, firstname, lasname, login, status, email and password state.
Branch sequence	
Exception sequence	

2.1.27 UA5.3 - Display local users configs

Title	UA5.3 - Display local users configs
Summary	The administrator displays the local user configs for all users registered in the database
Actors	Admin
Precondition	
Postcondition	
Base sequence	<ul style="list-style-type: none"> - The administrator sends a request to list all local users configs - The System returns all the local users configs for all users
Branch sequence	
Exception sequence	

2.2 Use case diagrams

2.2.1 UC UMS Admin

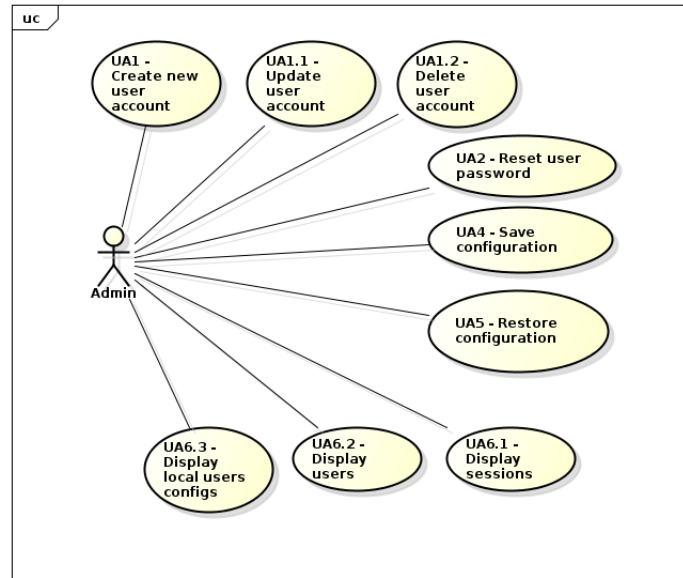


Figure 2.1: UC UMS Admin

2.2.2 UC UMS User Auto

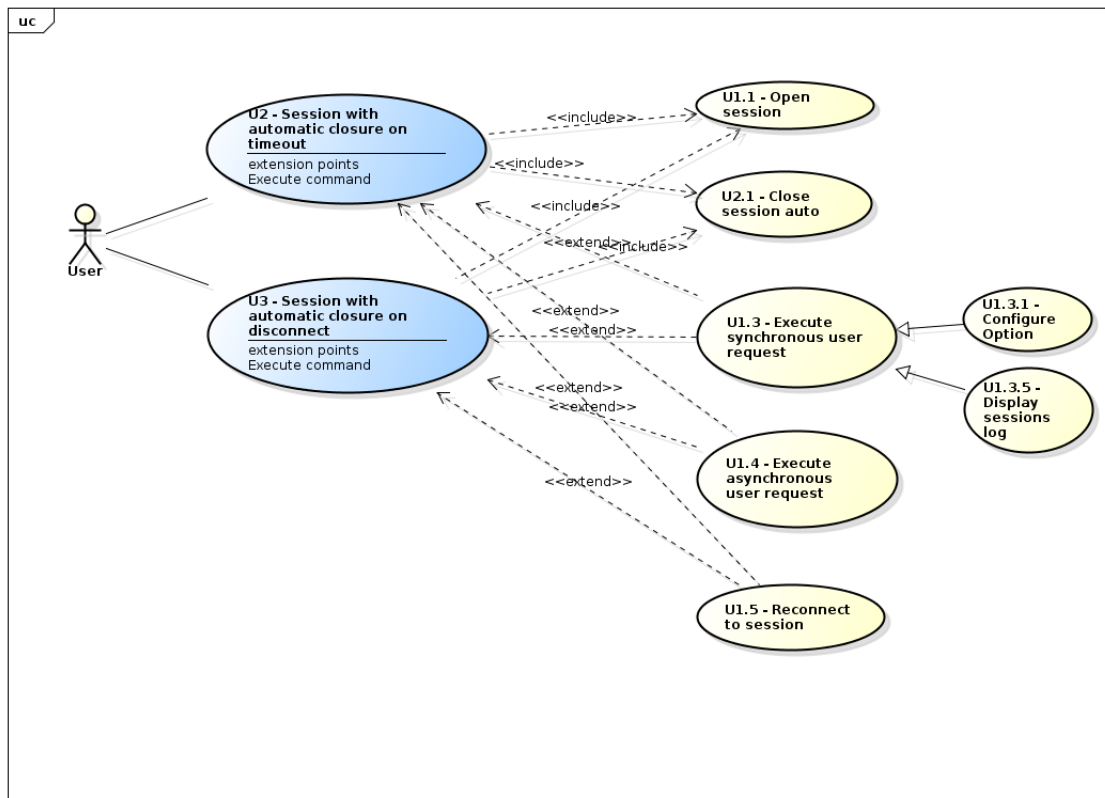


Figure 2.2: UC UMS User Auto

2.2.3 UC UMS User Manual

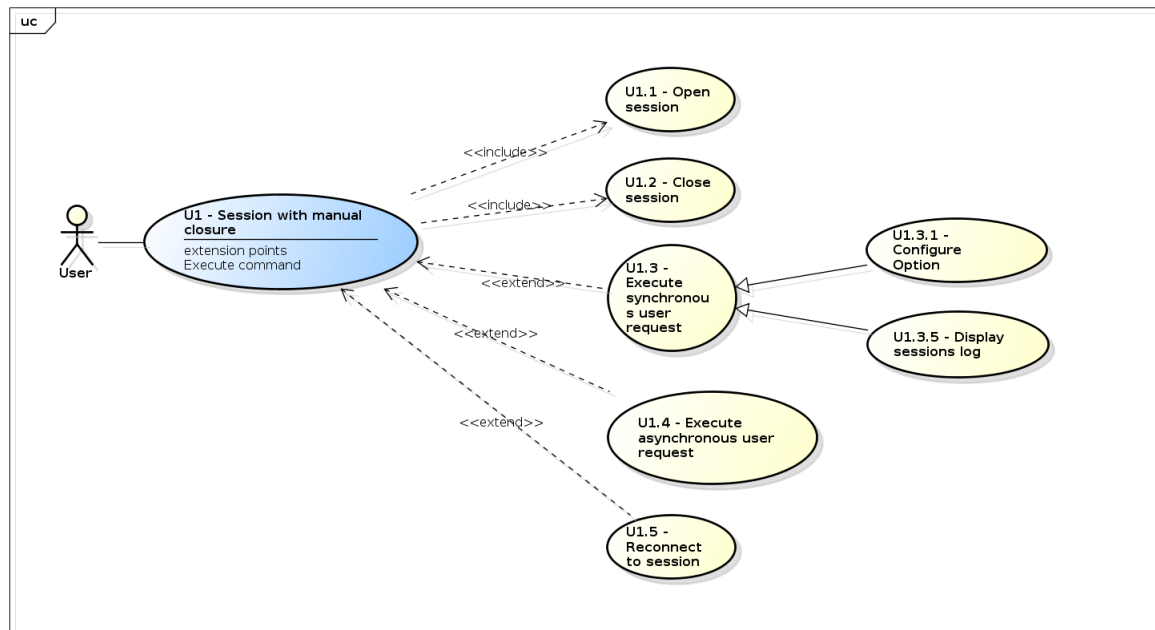


Figure 2.3: UC UMS User Manual

2.2.4 UC UMS User account

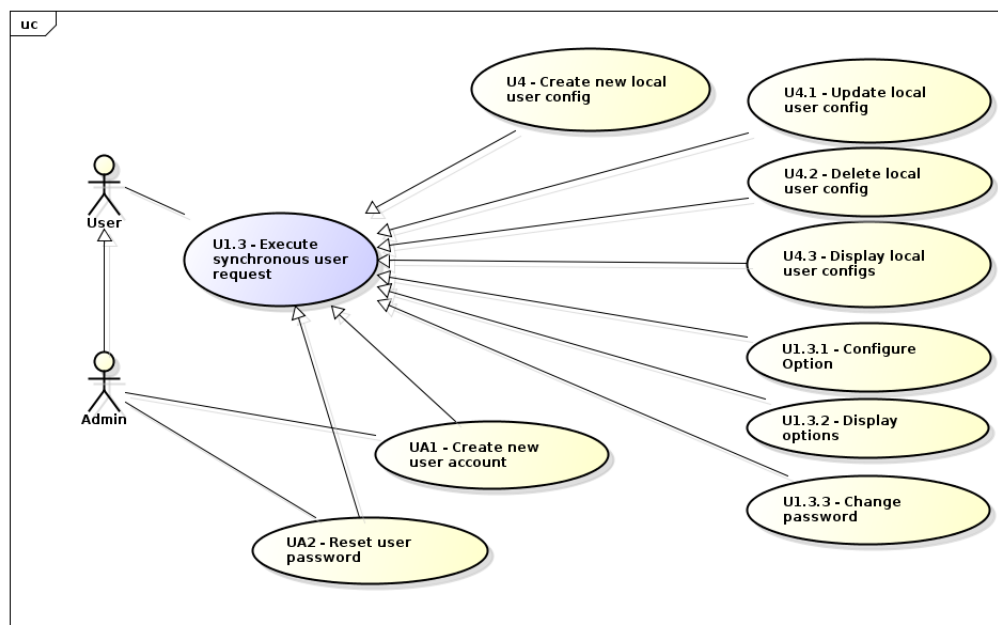


Figure 2.4: UC UMS User account

2.3 Data dictionary

- **Configuration** : The configuration contains all information about the users and machines registered in the database. It does not contain chronological information about the users or the infrastructure (logs, metrics values)
- **Local user config** : The local user config is the description of the given user on a specific machine described in the database
- **Option** : The option is a parameter of the user account that is not mandatory. Default value for each option is defined by the administrator. This features can be used by all VISHNU modules (not only UMS).
- **Password state** : Records the current state of the password of a user: either 'temporary' if the password must be changed next time the user connects to the System, or 'valid' if the password is in a normal state.
- **Root user** : Special user that is pre-configured in the VISHNU system and that has administrator privileges. This user cannot be deleted from the system.
- **Session** : A session is the context in which VISHNU commands are executed (ex: job submission, file transfers). It is created following authentication of a user and lasts until the session is closed either manually or automatically.
- **User account** : The user account is the description in the database of a VISHNU user

Chapter 3

Use cases for Tasks Management System (TMS)

3.1 Use case descriptions

3.1.1 T1 - AsyncCommandOnMachine

Title	T1 - AsyncCommandOnMachine
Summary	User starts an asynchronous command on a given machine
Actors	User
Precondition	- User has an active open session
Postcondition	- The command is in active state until completed - The system log has been updated and contains the request parameters
Base sequence	1. User sends the request with parameters including session id and machine id 2. System checks that the machine id is valid and machine is available 3. System checks that the session id is valid 4. If command parameters contain a file the System verifies that the file is available and readable 5. System returns information to the user about the request status 6. System records request information (time, user, machine, request parameters) in the system log
Branch sequence	5a. T1.1 SubmitJob
Exception sequence	1a. The TMS server is unavailable - The system returns an error message 2a. The name of the given machine is unknown -The system prints an error message that informs the user 3a. The session id is not valid - The system prints an error message that informs the user. - The user revises the id. 4a. The path to a file parameter is invalid - The system prints an error message that informs user - The user revises the path
Extensions	T1.1 - SubmitJob

3.1.2 T1.1 - SubmitJob

Title	T1.1 - SubmitJob
Summary	User submits a job on a given machine
Actors	User
Precondition	
Postcondition	<ul style="list-style-type: none"> - The job is submitted on the specified machine - The job state and id are recorded on the system's log - The job id is sent to the user
Base sequence	<ol style="list-style-type: none"> 1. The System checks that request parameters contain: <ul style="list-style-type: none"> - job script path - job options 2. The TMS server on the given machine is contacted 3. The job is submitted by the TMS server to the batch scheduler 4. The id of the submitted job is returned to the user
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 1a. Invalid options or script 4a. The batch scheduler server is unavailable 4b. The batch scheduler server rejects the request
Extension of	T1 - AsyncCommandOnMachine

3.1.3 T2 - SyncCommandOnMachine

Title	T2 - SyncCommandOnMachine
Summary	User executes a synchronous command on a given machine
Actors	User
Precondition	- User has an active open session
Postcondition	<ul style="list-style-type: none"> - Request is in completed state - The system log has been updated and contains the request parameters
Base sequence	<ol style="list-style-type: none"> 1. User sends the request with parameters including session id and machine id 2. System checks that the machine id is valid and machine is available 3. System checks that the session id is valid 4. If command parameters contain a file the System verifies that the file is available and readable 5. System returns information containing the results of the request 6. System records request information (time, user, machine, request parameters) in the system log
Branch sequence	<ol style="list-style-type: none"> 5a. T2.1-GetJobs 5b. T2.2-CancelJob 5c. T2.3-ListQueue 5d. T2.4-ListJobs 5e. TA1-SetMachineRefreshRate 5f. TA2-SetMachineEnv

Exception sequence	1a. The TMS server is unavailable - The system returns an error message 2a. The name of the given machine is unknown -The system prints an error message that informs the user 3a. The session id is not valid - The system prints an error message that informs the user. - The user revises the id. 4a. The path to a file parameter is invalid - The system prints an error message that informs user - The user revises the path
Extensions	T2.1 - GetJob T2.2 - CancelJob T2.3 - ListQueue T2.4 - ListJobs TA1 - SetMachineRefreshPeriod TA2 - SetMachineEnv

3.1.4 T2.1 - GetJob

Title	T2.1 - GetJob
Summary	User requests the TMS server to get information about a job
Actors	User
Precondition	
Postcondition	
Base sequence	1. The Systems checks the job id 2. The TMS server on the given machine is contacted 3. The TMS server asks job information to the batch scheduler server 4. The user receives job information
Branch sequence	
Exception sequence	1a. The job id is invalid 3a. The batch scheduler server is unavailable 3b. The batch scheduler server rejects the request
Extension of	T2 - SyncCommandOnMachine

3.1.5 T2.2 - CancelJob

Title	T2.2 - CancelJob
Summary	The user cancels a job
Actors	User
Precondition	
Postcondition	- The job is canceled on the specified machine - The job state and id are removed to the system's log - An information is sent to the user
Base sequence	1. The systems checks the job id 2. The TMS server on the given machine is contacted 3. The TMS server cancels the job 4. The user receives a message
Branch sequence	
Exception sequence	1a. The job id is invalid 3a. The batch scheduler server is unavailable 3b. The batch scheduler server rejects the request
Extension of	T2 - SyncCommandOnMachine

3.1.6 T2.3 - ListQueue

Title	T2.3 - ListQueue
Summary	User lists all queues or classes of a specific batch scheduler
Actors	User
Precondition	
Postcondition	<ul style="list-style-type: none"> - The system collects the informations on each queue or classes - The system records request parameters in system's log - The system send the list contained the information on all queues to the user.
Base sequence	<ol style="list-style-type: none"> 1. The TMS server on the given machine is contacted 2. The TMS server asks queues or classes information to the batch scheduler server 3. The System sends all queues list to the user
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 2a. The batch scheduler server is unavailable 2b. The batch scheduler server rejects the request
Extension of	T2 - SyncCommandOnMachine

3.1.7 T2.4 - ListJobs

Title	T2.4 - ListJobs
Summary	User lists all jobs submitted
Actors	User
Precondition	-User has an active open session
Postcondition	<ul style="list-style-type: none"> - The System sends information on all jobs to the user - The System records request parameters in the system's log
Base sequence	<ol style="list-style-type: none"> 1. The TMS server on the given machine is contacted 2. The TMS server asks job information to the batch scheduler server 3. The System sends full information on all jobs to the user
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 2a. The batch scheduler server is unavailable 2b. The batch scheduler server rejects the request
Extension of	T2 - SyncCommandOnMachine

3.1.8 TA1 - SetMachineRefreshPeriod

Title	TA1 - SetMachineRefreshPeriod
Summary	The admin sets the refresh period of output and error file content
Actors	Admin
Precondition	
Postcondition	- The refresh period value is stored by the system
Base sequence	<ol style="list-style-type: none"> 1. System saves the refresh period for the given machine. 2. System applies the refresh period to all current jobs and future requests
Branch sequence	
Exception sequence	<ol style="list-style-type: none"> 1a. Refresh period value is too short (minimum value : see technical requirements) <ul style="list-style-type: none"> - System returns an error message

Extension of	T2 - SyncCommandOnMachine
--------------	---------------------------

3.1.9 TA2 - SetMachineEnv

Title	TA2 - SetMachineEnv
Summary	The admin sets an environment variable
Actors	Admin
Precondition	
Postcondition	- Environment variable is set
Base sequence	1. System saves the environment variable for the given machine. 2. System applies the environment variable to all current jobs and future requests
Branch sequence	
Exception sequence	
Extension of	T2 - SyncCommandOnMachine

3.1.10 TA3 - LaunchTmsServer

Title	TA3 - LaunchTmsServer
Summary	The administrator launches the VISHNU TMS server on a given machine
Actors	Admin
Precondition	- The Vishnu server software (TMS Module and dependencies) is installed on the machine - The machine is configured in the Vishnu system database - The batch scheduler processes are up and running on the same machine - The network connection between the machine and the Vishnu database server is up and running
Postcondition	- The TMS server is up and running - A server log has been created
Base sequence	1. Admin connects to the machine as vishnu user 2. Admin updates the Vishnu configuration if necessary (database server hostname and credentials, DIET configuration, Batch scheduler configuration) 3. Admin launches the Vishnu TMS Server executable 4. System checks the connections to its peers within the Vishnu platform 5. System retrieves the list of active jobs (not completed jobs) that were launched on the same machine 6. System checks that all the active jobs (from previous step) are still running on the batch scheduler, and eventually updates the job status (for ex. from waiting to running, or from running to finished) 7. System returns a status message to the administrator
Branch sequence	
Exception sequence	4a. A connection to a Vishnu peer is down. System returns an error message and stops 6a. The batch scheduler does not recognize some job ids. In this case the System updates the job status to completed.

3.2 Use case diagrams

3.2.1 UC TMS Overview

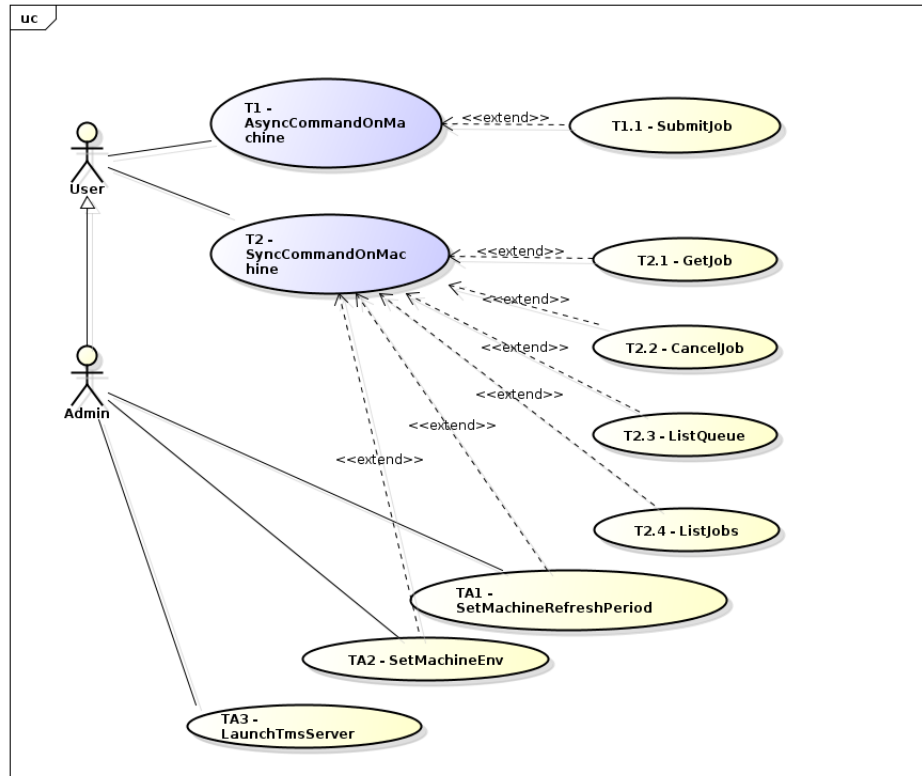


Figure 3.1: UC TMS Overview

3.3 Data dictionary

- **Batch Scheduler** : A batch scheduler is a distributed resource manager that enables to allocate at best the resources to the jobs on a machine according to user needs (the needs are specified by the user by batch directives (batch options) in file or command line).
- **Job** : A job is a sequence of instructions (included batch scheduler directives) written to launch and to perform by a specified batch scheduler.
- **Job id** : A job id allows to identify the job in the batch scheduler system.
- **JobPath** : A jobPath is the path to the file (script) containing the instructions (batch directives or job characteristics, job execution command) of the job.
- **Queue ou Classe** : A queue or class allows to associate the resource limits (CPU wallclock time, CPU memory) and execution priorities of the jobs.
- **TMS** : Task Management System