**Terms of reference**

**For the purchase of IT Service Management (ITSM) System**

|  |  |  |
| --- | --- | --- |
| **Item****№** | **List of basic data and requirements** | **Basic data and requirements** |
| 1 | Client | Kumtor Gold Company CJSC |
| 2 | Location | 24 Ibraimov str., Bishkek, Kyrgyz Republic |
| 3 | Purpose | Purchase of IT Service Management (ITSM) System |
| 4 | Functional requirements  | The system shall be able to implement ITIL processes (See Appendix 1) |
| 5 | Number of licenses | Ability to use the ITSM system by 60 agents (simultaneous use by at least 30 agents) |
| 6 | Requirement for the Contractor | Partner status with an official manufacturer of service asset and configuration management software or contracting directly with the supplier. |
| 7 | Licence validity period, technical support period, warranty period for the provision of services. | License validity period shall be minimum 1 year, technical support - minimum 1 year, warranty period for services - minimum 1 year. |

 **Documents to be submitted to participate in the selection process:**

* Description of company's experience;
* Company's registration documents;
* Recommendations from previous clients for the past three years;
* Absence of enforcement proceedings, seized property;
* Absence of arrears on taxes and fees.

|  |  |
| --- | --- |
|  |  |

1. INCIDENT MANAGEMENT
	1. Objectives of the module
2. Building a functional structure of the service for further automatic or manual allocation of responsibility of specialists to resolve incidents and tasks/work orders.
3. Ensuring control over the workload of employees of various support lines and functional teams.
4. Monitoring the progress of all requests.
5. Tracking the time characteristics of the progress of all requests according to the quality parameters defined at the SLA level.
6. Increasing the transparency of all incident resolution processes through a flexible notification and escalation mechanism (hierarchical and functional).
7. Using the Knowledge Base to resolve user requests.
8. Forming a personalized reporting for the purpose of continuous monitoring of key performance indicators of the support service.
9. Communicating with users, including receiving feedback from them in various ways: comments and correspondence on requests and service ratings.
	1. Incident life cycle

The system shall have a tool to implement the entire incident life cycle.

**Identification of incidents**

1. Manual and automatic mode to generate incident records.
2. Users of IT services can create records in the self-service portal.
3. IT staff can create records on behalf of the user.
4. Automatic registration of requests received by e-mail to the support service.
5. Registration of incidents by phone (see Integration).
6. Using customizable web forms.
7. Registration of requests from corporate portals, websites, and external business systems through integration.

**Registration of incidents**

Assigning a unique ID to each incident record.

Recording time and date of the record creation and its subsequent change.

Saving full contact information in the record (initiator's full name, feedback method).

Recording information about the source of the incident communication (person, event, group).

**Categorization and prioritization**

Separating requests into incidents and requests for service.

Ability to change the request type from incident to another request type (request for service, problem, request for change).

**Initial diagnostics**

Fixing the signs of failure and the results of diagnostics of its causes.

**Escalation**

Functional – based on manually defined or pre-configured conditions (target service level and operational level, business priority and support degree).

**Diagnostics and investigation**

Ability to create a request for service from the incident record, as well as establish a link between it and the request for service.

Ability to open a request for change (RFC) from the incident record, as well as establish cause-and-effect relationships between events and requests for change.

**Incident resolution and recovery**

Assigning the persons responsible to resolve the incident (group, department, or employee).

Storing information about the incident status.

All authorized users have access to incident data, while restrictions are imposed on operations with key attributes (priority, status, queue position), which are available only to technical support staff or other responsible persons.

Introducing a resolution with the ability to differentiate viewing for IT specialists/users.

Saving in the incident history complete data on the actions taken in the process of resolving and restoring the service, and the employees who took these actions.

Automatic bulk operations with incident records (classification, creation, and consolidation of records).

Automatic control and monitoring of the response and problem resolution time according to the service level and/or priority.

**Closing the incident**

Recording the incident resolution/closing data, including time and date.

Assessing the user satisfaction with the results of work to eliminate the problem.

* 1. Required functions.
1. Using predefined actions of objects that perform various operations according to certain rules.
2. Separate streams to process common incidents, major and infrastructure incidents.
3. Building hierarchical incident structures (master incident and dependent incidents).
4. Ability to change the request type from "Incident" to "Request for Service", to "Request for Change", to "Problem”.
5. Closing/reopening the incident.
6. Assigning one incident to several IT specialists.
7. Ability to create additional sub-requests - task assignments to different employees and ability to set a deadline for each task.
8. Assigning/reassigning responsible persons.
9. Working with SLA timers (time counter, stop, start, waiting for delivery, etc.).
10. Ability to retrieve a complete list of incident actions to print or email a report.
11. Ability to reserve the incident number before its saving.
12. Indicating the need for initial communication with a user in the general list of incidents.
13. Real-time incident processing statistics.
14. Using the "similar incidents" search to resolve incidents as part of the first line support. The search is conducted by Configuration Item, affected user, category, etc.
15. Based on an incident, entities can be created: a problem or a change (process, task, etc.).
16. Communications between employees of the different support lines as well as users.
17. Setting up notifications by email and in the personal cabinet by various parameters: creation of requests, status changes, assigning requests and setting deadlines, etc.
18. Ability to flexibly filter the list of all incidents by various parameters: status, urgency, assigned employee, etc.
19. Colour highlighting of requests by various parameters (deadlines, priorities, impact on user groups, etc.).
20. Monitoring the entire incident resolution process as it progresses through all stages.
21. Creating an incident form with the ability to determine which fields are mandatory.
22. Calculating labour costs for an incident resolving, an individual task, an employee.
23. When a team leader assigns incidents, it shall be possible to view the employee's current workload for open requests.
24. Ability to automatically assign an incident to an employee based on the request type.
25. The priority is generated based on the urgency set by the user (can be reviewed by the support staff member) and the impact set by the staff member.
26. The system shall track incidents with resolution time overruns, service level discrepancies, according to specified metrics, and provide the ability to view the data.
27. Control of the request completion time
28. Analytics and reporting tool (metrics, dashboards, inbuilt reports, ability to upload data, create custom reports).
	1. Interaction with other modules

Integration with request for service management tools.

Integration with Knowledge Base management tools.

Ability to create a request for change (RFC) from the Incident Management module.

Integration with release and deployment management tools.

1. REQUEST FOR SERVICE MANAGEMENT
	1. Objectives of the module
2. Forming a centralized portal focused on the provision of services and oriented to the user of these services.
3. Speeding up the completion of requests for service.
4. Optimizing the request completion costs.
5. Monitoring and planning staff workload to complete requests for service.
6. Ensuring communication between staff from different departments to reduce the time spent working on the request.
7. Monitoring and documenting the progress of the requests.
8. Ensuring transparency of the work on the requests by being able to monitor the entire process.
	1. Request for service life cycle.

The system shall have a tool to implement the entire request for service life cycle.

**Registration of requests for service**

1. Users of IT services can create records in the self-service portal.
2. IT staff can create records on behalf of the user.
3. Registration of requests by phone.
4. Automatic registration of requests received by e-mail to the support service.
5. Registration of requests from corporate portals.

**Categorization and prioritization**

Defining the category of requests based on a customisable directory with a hierarchical structure.

Prioritizing requests according to the set priorities in manual mode or pre-configured conditions.

Automatic priority calculation based on SLA information, configuration item type, etc.

Ability to adjust the priority during the request processing with fixing the changes for subsequent audit and reflection in reports.

Ability to change the request type from incident to another request type (request for service, problem, request for change).

**Approval of the request for service**

Forwarding the request for approval automatically according to pre-configured rules or manually by the responsible employee.

**Delegation of authority**

Forwarding the request for approval to the replacement employee in the event of the supervisor's absence from work.

**Completion of the request for service**

Assigning the persons responsible to complete the request (group, department, or employee).

Ability to create a request for service from the incident record, as well as establish a link between it and the request for service.

Storing information about the status.

Saving in the request history complete data on the actions taken in the process of completing the request, and the employees who took these actions.

Bulk operations with records (classification, creation, and consolidation of records).

Automatic control and monitoring of the response and completion time according to the service level and/or priority.

**Closing the request for service**

Recording the request completion/closing data, including time and date.

Assessing the user satisfaction with the results of the request.

* 1. Required functions.
1. Creating the request for service records and storing the detailed information about the request.
2. Ability to change the request type from "Request for Service" to "Incident", to ‘"Request for Change”.
3. Ability for users to view a description of the services available to them when creating a request for service on the self-service portal.
4. Ability for users to view/use the available knowledge bases when registering a request for service on the self-service portal.
5. Recording the date and time of the request for service record creation and its change.
6. Categorization of requests.
7. A tool to agree and approve a request before its completion.
8. Recording information about urgency, impact degree and priority in the request record, ability to change these attributes during the request completion.
9. Providing the user with detailed information about the request progress status.
10. Automatic routing of requests to executors: employees, groups, external organizations.
11. Using templates for the most typical requests for service.
12. Storing information about the request closing category.
13. Tools to conduct request for service analyses to identify trends.
14. Tools to conduct user satisfaction surveys.
15. Ability to resume processing of a previously closed request.
16. Automatic rejection or confirmation of the decision by the system member using the self-service portal or by e-mail.
17. Communications between employees of the different support lines as well as users.
18. Setting up notifications by email and in the personal cabinet by various parameters: creation of requests, status changes, assigning requests and setting deadlines, etc.
19. Ability to flexibly filter the list of all requests by various parameters: status, urgency, assigned employee, etc.
20. Colour highlighting of requests by various parameters (deadlines, priorities, impact on user groups, etc.).
21. Monitoring the entire process as it progresses through all stages.
22. Creating request forms with the ability to determine which fields are mandatory.
23. Calculating labour costs for a request, an individual task, an employee.
24. When a team leader assigns requests, it shall be possible to view the employee's current workload for open requests.
25. Ability to automatically assign a request to an employee based on the request type.
26. The system shall track requests with resolution time overruns, service level discrepancies, according to specified metrics, and provide the ability to view the data.
27. Control of the request completion time
28. Analytics and reporting tool (metrics, dashboards, inbuilt reports, ability to upload data, create custom reports).
	1. Interaction with other modules

Integration with incident management tools.

Ability to create a request for change (RFC) from the Request for Service Management module.

Integration with release and deployment management tools.

Integration with configuration management system.

Integration with the service catalogue module to support creation of a request for service by the user directly from the service catalogue, and to create and maintain links between the request for service records and the service catalogue.

1. KNOWLEDGE MANAGEMENT
	1. Objectives of the module
2. Organizing the creation and maintenance of a knowledge base.
3. Using the knowledge base materials in various management processes.
	1. Required functions.
4. Generating a unique ID for each record/item in the knowledge base.
5. Ability to place attached files of any formats (messages, electronic documents, tables, multimedia content, etc.) in database records.
6. A single structured entry method using forms to create new records.
7. Creating and maintaining links between database records.
8. Automatic recording of the information about the author, data owner, creation date, etc. in the database records.
9. Search for data in the knowledge base records by various parameters (topic, owner, date, keywords, etc.).
10. Search for content that is stored in different formats.
11. Tools to create and maintain FAQs for clients and users.
12. Tools to classify data in the database.
13. Ability to submit an article for approval prior to publication.
	1. Interaction with other modules

Ability for the incident management personnel to create database records.

Ability for the problem management personnel to create database records.

Quickly creating a knowledge base record from a request for change record (RFC) and establishing a link between records.

Integration with the Configuration Management Database (CMDB) to support links between knowledge base records and configuration item records.

1. CHANGE MANAGEMENT
	1. Objectives
2. Assessing the impact of changes on business processes by analyzing the risk level and technical consequences of the changes.
3. Increasing the percentage of efficiency through increased change control.
4. Reducing the time required to implement changes.
5. Clear monitoring and detailed reports on the change progress.
6. Controlling all stages of the change and customization process and thereby reducing the risks associated with the change implementation.
7. Optimizing the prioritization of requests for change and thereby ensuring support of the most critical business services.
8. Reducing the volume of support calls by minimizing disruption associated with changes.
9. Planning upcoming changes in consultation with stakeholders, controlling responsibility to implement changes.
10. Keeping records of the costs for preparing and implementing changes.
11. Analyzing the results of changes.
	1. Required functions.
12. Assigning a unique ID to each request for change record (RFC).
13. Ability to create incident-based changes.
14. Recording the date and time of the request for change record creation and its subsequent change.
15. Monitoring and tracking changes throughout their life cycle (from initial assessment to closing).
16. Separating changes into types and the ability to customize the processing of changes of each type.
17. Categorizing changes by impact and priority.
18. Documenting the progress and results of the change approval procedure.
19. Ability to reject a request for change by a special role.
20. Ability to customize the powers of the Change Advisory Board (CAB) members depending on their role.
21. Indicating that the change has been approved.
22. Tools to conduct a preliminary assessment of a change.
23. Generating a change schedule with appropriate access controls to reflect all approved changes and notifications to users and IT staff.
24. Generating a schedule of expected service downtime.
25. Monitoring the availability of a tested rollback or recovery plan when approving a change.
26. Reminding of the readiness to check completed changes.
27. Recording information about changes that have already been checked.
28. Recording the closing date of changes. Ability to specify the closing category.
29. Supporting creation and use of change templates.
30. Supporting the process to implement standardized changes.
31. Tools to analyze, fine tune and plan proposals for changes.
32. Supporting the link between the change and its feasibility study and risk and requirements documentation.
33. Manual and automatic allocation of responsibility to an employee or team.
34. Ability to divide the change into stages.
35. Calculating planned and actual time spent.
36. Approving changes via e-mail or the self-service portal.
37. Utilizing complex approvals for important changes.
38. Ability to link an RFC to one or more incidents and issues related to a given change.
39. One or more Configuration items can be linked to the RFC, which will be affected by this change.
40. Complex Change Management. A process is linked to the RFC, which consists of a sequence of levels, and these are broken down into tasks. Additionally, each level can have sub-processes.
41. Availability of a tool to automate the change processes. Different employees or teams can be automatically assigned to different stages of the change process.
42. Availability of a change calendar that provides a graphical representation of all planned and ongoing changes makes it easy to track potential conflicts resulting from the intersection of change implementation stages.
43. Ability to suspend the change progress for selected systems at pre-set dates (scheduled audits, etc.) - "Change Freeze".
44. Analytics and reporting tool (metrics, dashboards, inbuilt reports, ability to upload data, create custom reports).
	1. Interaction with other modules

Establishing and maintaining links between change records and incident records.

Establishing a link between problem/known error records and change records.

Ability to coordinate and schedule release and deployment activities using change management tools.

1. SERVICE CATALOGUE MANAGEMENT
	1. Required functions.
2. Centralized monitoring of all services.
3. Supporting the life cycle of services, from service creation to decommissioning.
4. Organizing searches to make it easier for users to place an order.
5. Ability to publish services on the self-service portal with the ability to restrict access to company's users, groups, or divisions.