Functional Specification

Project Name

iPaMS: Integrated Programme and Module System, v1.0

Reference Code

Completion Date

February 2011

Project Summary

There has been considerable work undertaken on defining XML schemas for programme descriptors (XCRi). In addition a parallel project under the Flexible Service Delivery Programme to define schemas for the HEAR (Higher Education Achievement Record) is about to commence. The iPaMS project will build on these pieces of work by defining module templates to complement and extend the programme descriptors and by providing an integrated management infrastructure to manage them and link to the HEAR.

This work is needed as most establishments do not have a central management system for programme descriptors and module templates, resulting in a plethora of inefficient and error prone processes and systems. By delivering a toolkit which can be adapted to suit local needs, all the objectives of the project listed in Section 2 below can be realised.

The application will include version control for academic programme approval processes, and based on SOA principles will integrate with multiple applications for repurposing of the data. Use of the data will include (but not be restricted to) web publication of programme and module information, publication of marketing information, links to VLEs and student record systems.

The toolkit delivered to the community will consist of a web application to manage the application, an open database schema to define the structure, and a set of web services to transfer data to and from the database.

The toolkit will be tested by a full implementation in one of the University’s academic units.

[Extract from iPaMS Project Plan]

Existing System

A number of separate information systems exist at University of Exeter to handle programme and module data. For the purposes of this document a brief summary of each is provided below:

SITS Student Records System

The University of Exeter has implemented the SITS Student Records System, which holds a certain amount of programme and module data and associates it with student records. The system does not hold any descriptive data for programmes or modules such as learning outcomes or assignments but it does maintain a detailed record of a student’s programme and module choices and results across the lifetime of their time at the University. This information is held in tables in the SITS database (generally accessed via screens with the same name) such as PRG, ROU, SCI, SCE, MOD, MAV, TOP, SMO, SMR, SMS. Module diet information (a prescribed set of module choices for a particular route) is held in SITS in the following tables: PDT, PDM and FMC.
University of Exeter Business School: Programmes & Modules system

The University of Exeter Business School has developed a bespoke Programme and Modules management application which is currently in use across all departments (accounting, economics, Finance, Leadership and Management). A full description of the application can be found in Appendix 1.0

University of Exeter College of Humanities: Module system

The College of Humanities at University of Exeter have developed a bespoke Module management application which is currently in use in all former HuSS (Humanities & Social Science) departments. A full description of the application can be found in Appendix 1.1

University of Exeter College of Engineering, Mathematics and Physical Sciences (CEMPS)

CEMPS have a system known as UDB which is used to manage programmes and modules along with other information sets such as staff lists, building information and research grants and finance. A full description of the application can be found in Appendix 1.2

Proposed System

Process Map

iPaMS will provide a highly flexible toolkit for the administration and publication (via iPaMS-WS) of programme and module data. Due to this flexibility it is felt that a brief written description of the general process flow involved is more appropriate in this first phase of development than formal diagrams which may mislead and/or become overly complex in an attempt to define every possible usage scenario.

The second phase of development is likely to incorporate full business and academic workflow and approval mechanisms, and during this phase process flow diagrams would become an essential tool.

The following illustrates a basic process flow in iPaMS, from the perspective of an administrative user:

1. Administrator logs in to iPaMS administrative application using their web browser. If they are successfully authenticated and authorised they will be shown a ‘dashboard’ page appropriate to their level of access.

2. From the dashboard (main menu) the user selects the functional area they wish to work in:
   a. Programmes (including programme structures)
   b. Modules
   c. Organisational structure
   d. Application security
   e. Configuration

3. The user will then see a secondary dashboard page for the chosen area of the system, with facilities to create, search, view, update, delete and archive records; depending on the level of authorisation and association with one or more organisational units (departments).

4. Having selected the required function, the user will be presented with a series of dynamic forms providing all the necessary functionality, for example editing a programme specification or module description. The user would be able to request to see any dependencies of the record they are editing (for example when editing a module view a list of all programmes the module is included on). Once finished the user selects to either save or discard their changes.

5. Any saved changes are persisted to the iPaMS database. One key requirement is to maintain a comprehensive audit trail and version history of programme specifications and module descriptions. To support this facility the user may be prompted to enter into the system some brief notes explaining what changes have been made and why. The system would record these notes alongside the new version of the record with a timestamp and the user’s details.
6. The iPaMS application sends out any configured alerts to other relevant users or groups to make them aware of the changes that have been made.

7. The user could then logout from the application, ending their session, or return to either of the dashboards to perform further operations.

8. Without full business and academic workflow, it is likely that a basic level of approval will be required on certain changes, e.g. to programme specifications and module descriptions prior to the revised version becoming the current ‘live’ record. Approval may be granted by an administrative user who has received an alert that data has been changed, who then logs into iPaMS and can update the record to either approve or reject the changes.

9. Any approved changes to the data in iPaMS are immediately presented through iPaMS-WS to users of any external application, such as websites and smartphone apps. This does not necessarily mean that every modified item of data in iPaMS is visible, for example on certain corporate web pages only marketing information might be presented therefore the impact of any changes depends a great deal on the usage of that data externally.

**Users**

iPaMS users can be split into two high-level groups:

- **Data administrators**
  - Will be expected to login to the system using their main IT account credentials.

- **External data consumers**
  - Programme and module data will be made available to data consumers who are not iPaMS administrators on a read-only basis through the web service layer via an external application such as the organisation’s website or intranet.

   - **Data administrators**
     Limited to staff at the ‘host’ educational institution, i.e. University of Exeter. The class can be further divided into the following:

     - **System administrators**
     - **Department administrators**
     - **Marketing administrators**

     Access to the iPaMS administrative application could vary depending on the IT usage policies of each host organisation; at University of Exeter it is expected that access will be limited users connecting from on campus (including VPN).

     - **System administrators**
     - Estimated number of users is very low – probably 5 or less. Highest level of user class in terms of system permissions and access. Can access any function within the system and any data. Exclusive operations to include:
       - Create, read, update, archive and delete any item of data in the system that is exposed through the user interface, e.g.:
         - Authorisation (Access control list)
         - Organisational structure
         - Programmes (programme descriptions)
         - Programme structures (programme-module associations)
         - Modules (module descriptions)
       - Perform system configuration tasks, for example setting database / LDAP connection details

     - **Department administrators**
     - Largest group of administrative users, depending on size of organisation and departmental structure could be anywhere between 20-100 users. Can access the following functions and...
data within the system:
- Create, read, update and archive the following data within one or more organisational units (College / school / department)
  - Authorisation
  - Programmes
  - Programme structures (programme - module associations)
  - Modules

- **Marketing administrators**
  Estimated number of users per organisation is under 50. Staff may reside in the central marketing team or be department based. Can access the following functions and data within the system:
  - Create, read and update the following data within one or more organisational units (College / school / department)
    - Programmes:
      - Marketing information
      - Version control: ability to archive details
    - Modules:
      - Marketing information
      - Version control: ability to archive details

- **External data consumers**
  It is anticipated that end user data consumers will access the information held within iPaMS through external applications, such as websites and Smartphone applications which will extract programme and module data via a web service layer.
  End user groups can be defined as:
  - **Current and prospective students:**
    - Undergraduates & Postgraduates
      - Look up detailed information about their programmes and modules.
      - View information about modules to make their choices for the following year
    - International exchange
  - **Staff**
    - Internal
    - External (e.g. prospective employees, staff at related organisations such as UCAS)
  - **General public**
    - Undefined, but based on a flexible web service search facility for programmes and modules, an application which interacts with iPaMS should be able to provide a search facility based a variety of criteria

None of the data consumers will be able to update the data held within iPaMS, and no web service operations enabling updates should be exposed without requiring the intermediary application (e.g. website) to authenticate first. Public access over the internet to this information will be enabled by means of publishing through the University websites, smartphone apps, etc.

**Usage patterns**
It is anticipated that the vast majority of administrative functions (90%+) would be conducted during normal office hours, i.e. Monday-Friday, 8am-6pm. Usage would vary dramatically throughout the year, with a peak of activity over the summer months and into the start of each academic year as departments review and amend existing programmes and modules and add new items. Concurrent access levels would be estimated between 20 users during quiet periods rising to 200 during peak times, based on staffing levels at University of Exeter.

In terms of demand for iPaMS data from external applications via the web service layer, the application should provide a level performance that is able to respond in a timely manner to the anticipated number of requests received. This would of course vary depending on the institution and number of iPaMS-WS client applications. The system will be load tested based on website usage patterns.
Data

The iPaMS application will be primarily concerned with storing, administering and publishing descriptive and versioning data items in the following areas:

- Programmes
- Modules
- Programme structures (programme - module associations)

The intention is that iPaMS will have the capability to become the master data source for each of these areas across multiple departments within an organisation. In support of the primary data items considered above, iPaMS will also be required to store the following supplementary information:

- Organisational structure
- Security model (authentication & authorisation)
- Configuration information

A detailed discussion of the data items required for each area is presented below. This information has been compiled to a large extent from the programme and module descriptions found in this document.

NB: This section does not constitute a definitive list of every database table, field or relationship that may be created in the iPaMS schema; the actual design of the database schema is covered in iPaMS Work Package 4a.

Programmes

The following table details the descriptive programme data that will be held within iPaMS:

<table>
<thead>
<tr>
<th>#ID</th>
<th>Data item(s)</th>
<th>Description and purpose</th>
<th>Format</th>
<th>Marketing version required?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Core programme information</td>
<td>The information in this section defines the core programme attributes that will not change with different versions (revisions) of a programme.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P1</td>
<td>Programme Title</td>
<td>To indicate in broad terms programme scope and content. Adopted in official publications including the Calendar, the prospectus, and on student transcripts.</td>
<td>Free form text</td>
<td>Yes</td>
</tr>
<tr>
<td>P2</td>
<td>Awarding Institution</td>
<td>To identify the universities or other higher education institutions empowered to award the qualification to be received by those successfully completing the programme.</td>
<td>Search or select from drop-down list, with option to add free form text and more than one institution. System configuration should facilitate setting of a default value, e.g. ‘University of Exeter’.</td>
<td>No</td>
</tr>
<tr>
<td>P3</td>
<td>School(s) / College(s) / Teaching Institution</td>
<td>To identify the organisational unit responsible for coordinating the programme.</td>
<td>Drop down box with all relevant departments / institutions listed, from the organisational structure data within iPaMS</td>
<td>No</td>
</tr>
<tr>
<td><strong>P4</strong></td>
<td><strong>UCAS Code</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>----------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(UCAS: Universities &amp; Colleges Admissions Service)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Relevant only for undergraduate and postgraduate programmes where entry is via UCAS. Ensures that the programme articulates with UCAS mechanisms for advertisement and recruitment and with the University Calendar and Prospectus.</td>
<td>Alphanumeric code or N/A for ‘not applicable’.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>No</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Descriptive programme information**  
The information in this section defines the descriptive details of the programme, which may vary to form revisions (versions) of a programme without necessitating creation of a new top-level programme record (though clearly changes beyond a certain threshold would naturally lead to the creation of a new programme).

<table>
<thead>
<tr>
<th><strong>P5</strong></th>
<th><strong>Programme accredited / validated by</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To identify external bodies that have endorsed the contents and standards of the programme</td>
</tr>
<tr>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>P6</strong></th>
<th><strong>Final Award(s)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To identify the degree, diploma, certificate or other similar formal mark of recognition for successful completion of the programme of study</td>
</tr>
<tr>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>

| **P7** | **FHEQ Level of Final Award(s)**  
(FHEQ: Frameworks for higher education qualifications) |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To indicate how the award level compares with national standards</td>
</tr>
<tr>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>

| **P8** | **QAA Subject Benchmarking Group**  
(QAA: Quality Assurance Agency) |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To indicate how the programme articulates with the QAA’s benchmark statements that make explicit the general academic characteristics and standards of honours degrees in the UK. This is usually not relevant for postgraduate programmes.</td>
</tr>
<tr>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>P9</strong></th>
<th><strong>Programme overview</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To provide a description of the timescale, composition, assessment, and distinctive structural features of the programme with details that may be helpful to current or prospective students and to academic reviewers.</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>P10</strong></th>
<th><strong>Programme duration</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Is the programme available full-time, part-time or both (and if part-time over how many months?) Is modular, distance or blended learning the main mode of study or also available?</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>P11</strong></th>
<th><strong>Start date</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>If not standard October start date</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>P12</strong></th>
<th><strong>Programme description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>150-200 words max covering the programme’s aims and (for PG programmes) the type of student it is designed for (e.g. is work experience required? Is this a professional, post-experience programme?),</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>P13</td>
<td>Distinctive features</td>
</tr>
<tr>
<td>P14</td>
<td>Career opportunities</td>
</tr>
<tr>
<td>P15</td>
<td>Assessment</td>
</tr>
<tr>
<td>P16</td>
<td>Educational Aims of the Programme</td>
</tr>
<tr>
<td>P17</td>
<td>Programme Outcomes and Teaching, Learning &amp; Assessment Methods</td>
</tr>
<tr>
<td>P18</td>
<td>Support for Students and Students’ Learning</td>
</tr>
<tr>
<td>P19</td>
<td>Admissions Criteria</td>
</tr>
<tr>
<td>#</td>
<td>Data item(s)</td>
</tr>
<tr>
<td>---</td>
<td>-------------</td>
</tr>
<tr>
<td>P20</td>
<td>Regulation of Assessment and Academic Standards</td>
</tr>
<tr>
<td>P21</td>
<td>Indicators of Quality and Standards</td>
</tr>
<tr>
<td>P22</td>
<td>Methods for Evaluating and Improving Quality and Standards</td>
</tr>
<tr>
<td>P23</td>
<td>Programme structure</td>
</tr>
</tbody>
</table>

**Versioning information**

The information in this section is designed to provide a flexible framework for revising and identifying programme descriptions between changes.

<table>
<thead>
<tr>
<th>#</th>
<th>Data item(s)</th>
<th>Description and purpose</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>P24</td>
<td>Date of Last Revision</td>
<td>To indicate when the programme specification was last formally approved</td>
<td>Date format – automatically populated and updated by the system.</td>
</tr>
<tr>
<td>P25</td>
<td>Latest modifier</td>
<td>Identifies the member of staff who last modified the record</td>
<td>Automatically populated and updated by the system.</td>
</tr>
<tr>
<td>P26</td>
<td>Version number</td>
<td>Incrementing arbitrary identifier</td>
<td>Automatically populated and updated by the system.</td>
</tr>
<tr>
<td>P27</td>
<td>Comments</td>
<td>Allow the user to give a general description of the changes made to the programme record</td>
<td>Free text</td>
</tr>
</tbody>
</table>

**Programme Structures**

A programme structure defines the content of a programme for a particular academic year, across all stages (years) of the programme and should include the following items of data in a repeating table-like structure within iPaMS:

<table>
<thead>
<tr>
<th>#</th>
<th>Data item(s)</th>
<th>Description and purpose</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>P51</td>
<td>Academic year</td>
<td>Identify the academic year this programme structure applies to.</td>
<td>Standard academic year format, e.g. 2010/11</td>
</tr>
<tr>
<td>P52</td>
<td>Stage</td>
<td>Used to group a selection of modules within the program structure for a particular academic year / stage (e.g. Economics - 2010/11, Stage 1).</td>
<td>Number within a configurable range (for most programmes 1-4)</td>
</tr>
</tbody>
</table>
PS3 Modules
Define the mandatory and optional modules for a particular academic year / stage, making iPaMS should allow users to easily build up a flexible list of modules, allowing search and selection of each module from the iPaMS system, stating whether they are mandatory or optional with the flexibility to create ‘option groups’ i.e. either module A or module B or module C.

PS4 Credits
The system should allow for input of the total credits required for each academic year / stage. Number within a configurable range (e.g. 0 to 120). iPaMS should also show a ‘running total’ of the credit value of the modules added to the program structure, to inform the user when the target value has been met.

Modules
The following table details the descriptive module data that will be held within iPaMS:

<table>
<thead>
<tr>
<th>#ID</th>
<th>Data item(s)</th>
<th>Description and purpose</th>
<th>Format</th>
<th>Marketing version required?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Core module information</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The information in this section defines the core module attributes that will not change with different versions (revisions) of a module.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M1</td>
<td>Module Code</td>
<td>Unique identifier for module.</td>
<td>The exact format of this code changes between institutions, but in general a module code will be a unique (within an organisation) alphanumeric string under 10 characters in length indicating the organisational unit (e.g. Business School), the level, an ID and optional suffixes to distinguish modules that are otherwise identical based on some specific criteria (e.g. mode of delivery). iPaMS should allow the configuration of an input mask (regular expression) on the module code to allow each institution to enforce their own format.</td>
<td>No</td>
</tr>
<tr>
<td>M2</td>
<td>Module Level</td>
<td>To specify the level at which the module’s learning outcomes are set according to the university’s Levels &amp; Awards Framework</td>
<td>A number or letter(s) from the following list: 0, 1, 2, 3, M, TD.</td>
<td>No</td>
</tr>
<tr>
<td>M3</td>
<td>Module Title</td>
<td>Indicator of module scope and content. Adopted in official publications including the Calendar, the prospectus, the ECTS Guide, and student transcripts. Re-named versions of existing modules must always have a new module code. Transcripts are generated from module codes. If the title changes without changing the code, then the module of study is retrospectively altered for all students, leading to inaccurate information being held and produced.</td>
<td>Free form text, normally 50 characters or less so that it can be accommodated on a student’s transcript.</td>
<td>Yes</td>
</tr>
<tr>
<td>M4</td>
<td>Credit</td>
<td>A quantified means of expressing equivalence of learning.</td>
<td>A number, always to correspond with the 15/30 credit framework (excluding dissertations) for PG module or any multiple of 5 for an UG module.</td>
<td>No</td>
</tr>
<tr>
<td>M5</td>
<td>ECTS</td>
<td>The European Credit Transfer Scheme allows for the transfer of credit between European Universities. The ECTS value must be half of the module credit value (#M4). This is applicable to all modules.</td>
<td>A number - half the credit value.</td>
<td>No</td>
</tr>
<tr>
<td>M6</td>
<td>Other</td>
<td>Certain institutions may use different units of measurement for module value, such as points.</td>
<td>A number with an assigned unit (e.g. points)</td>
<td>No</td>
</tr>
</tbody>
</table>

**Descriptive module information**

The information in this section defines the descriptive details of the module, which may vary to form revisions (versions) of a module without necessitating creation of a new core module record (though clearly changes beyond a certain threshold would naturally lead to the creation of a new module).

<p>| M7 | Description | Short overview of the module | Free form text, normally no more than 150 words. Should be able to format content, e.g. using a HTML WYSIWYG (what you see is what you get) editor. | Yes |
| M8 | School(s) / College(s) / Teaching Institution | To identify the organisational unit(s) responsible for teaching the module. | Drop down box with all relevant departments / institutions listed, from the organisational structure data within iPaMS. System should allow for the selection of multiple administrative units, e.g. college or school, department or discipline, centre or group). | No |
| M9 | Lecturer(s) | Publicises the roster of teaching staff | Search or select from a drop down list of appropriate staff, ideally drawn directly from a central IDM system (not iPaMS). iPaMS would then just store the unique identifier for the staff member against the module. Allow selection of multiple staff and identification of module co-ordinator plus notes field. | No |</p>
<table>
<thead>
<tr>
<th>Module Relations</th>
<th>Pre-requisites</th>
<th>Prerequisites are requirements for prior learning used in curriculum design to ensure progression or to produce pathways in a module portfolio.</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Co-requisites</td>
<td>Co-requisites are requirements for concurrent or prior learning used in curriculum design to ensure coherence or progression in a module portfolio within a stage.</td>
<td>Search or select from a drop-down list of module codes / titles and allow multiple selections / notes field.</td>
<td>No</td>
</tr>
<tr>
<td>Exclusions</td>
<td>Exclusions identify modules that cannot be taken in conjunction with each other.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Predecessors</td>
<td>Module(s) which this module is designed to replace</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Successors</td>
<td>Module(s) which are designed to replace this module once it has been succeeded</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Duration of Module</td>
<td>To indicate the time-span of the module’s teaching and learning activities.</td>
<td>A number, representing a time-span in units of weeks or semesters (drop-down or radio group selection for units).</td>
<td>Yes</td>
</tr>
<tr>
<td>Total Student Study Time</td>
<td>To publicise the amount and composition of student workload associated with the module, to demonstrate that the management of this workload has been fully considered in the module’s design, and to promote parity in workload among modules of equal credit value.</td>
<td>A number, representing a total in units of hours with an indication of the intended allocation among various teaching and learning activities, including contact time and private study. System should support business rules, e.g. at University of Exeter the total hours will be ten times the credit value (#M4) of the module</td>
<td>No</td>
</tr>
</tbody>
</table>
| Assignments and Assessments | To publicise the formal workload associated with the module (formative and summative); to demonstrate proper module design by specifying linkages between the student’s attainment of the module’s learning outcomes and the assessment components; to provide information on feedback; to promote parity among modules of equivalent credit value. | Table format allowing addition of multiple rows, each row containing the following columns:  
  - Formative or % contribution (e.g. 100% of final mark)  
  - Form of assessment (e.g. examination)  
  - Size of assessment (number plus units, e.g. 3 hours)  
  - ILOs assessed, providing lookup back to ILOs assigned in #M19  
  - Feedback method  
  - Notes | No |
| M18 | Aims | Aims are about teaching and the management of learning. The rationale for the module should be given, and a description of what the module addresses. Aims should give students a short description of the teaching intentions for the module | Free form text, normally no more than 150 words. Should be able to format content, e.g. using a HTML WYSIWYG (what you see is what you get) editor. | No |
| M19 | Intended Learning Outcomes | An Intended Learning Outcome is a statement of what a student is expected to know, understand and be able to do at the end of a period of learning and how that learning is to be demonstrated and/or represented. | Under three section headings: • module specific skills • discipline specific skills • personal and key skills iPaMS should allow for search and selection of ILO skills from within the database (and allow users to add new skills), normally up to a maximum of 10 ILOs in total for the module. | No |
| M20 | Learning/Teaching Methods | To publicise the nature of the teaching activities and learning environment; to demonstrate proper module design by specifying linkages between training and the assessment of the student’s attainment of the module’s learning outcomes; and to demonstrate that student learning is supported by assessment and feedback. | Free form text, normally no more than 150 words. Should be able to format content, e.g. using a HTML WYSIWYG (what you see is what you get) editor. | No |
| M21 | Syllabus Plan | To publicise the topics covered in the module and the organisation of the teaching activities. | Free form text including a list of topics, usually in chronological order of study, sometimes broken down into units of weeks. Should be able to format content, e.g. using a HTML WYSIWYG (what you see is what you get) editor. | No |
| M22 | Indicative Basic Reading List | To publicise the reading materials that are important or essential for those studying the module; to demonstrate the academic foundation of the module. | Free form text or formatted for compliance with bibliographical schema such as DocBook. Normally a short list of books or articles in reference format (author, date, title, publisher). If a core text or textbook exists, this should be indicated. Lists should be indicative, rather than a full bibliography. | No |

**Versioning information**
The information in this section is designed to provide a flexible framework for revising and identifying module descriptors between changes.

| M20 | Academic year | Indicates the academic year this version of the module applies to | Standard academic year format, e.g. 2010/11 | No |
| M21 | Semester / term | Indicates the semester or term this version of the module applies to | Selectable radio group or drop-down for ‘semester’ or ‘term’ and a numerical field | No |
Organisational structure

The table below details the organisational structure information that iPaaS will need to hold. This data will of course exist in a multitude of other systems such as IDM, Student Records, HR and potentially the iPaaS WS layer could be used to populate the data within the iPaaS database, although given the (relatively) static nature of the data it may be more straightforward to enter it manually. It is envisaged that this data will be used in most other areas of iPaaS to form structural associations, e.g. between a department administrator and a set of programmes or modules to which they have access.

<table>
<thead>
<tr>
<th>#ID</th>
<th>Data item(s)</th>
<th>Description and purpose</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS1</td>
<td>Unit code</td>
<td>Short code identifying the organisational unit (OU).</td>
<td>Free text up to a configurable maximum length</td>
</tr>
<tr>
<td>OS2</td>
<td>Short name</td>
<td>Short name of OU.</td>
<td>Free text</td>
</tr>
<tr>
<td>OS3</td>
<td>Full name</td>
<td>Full name of OU.</td>
<td>Free text</td>
</tr>
<tr>
<td>OS4</td>
<td>Parent OU</td>
<td>Optional. Defines a parent-child relationship between OUs, e.g. ‘College of Humanities’ is the parent of ‘Drama department’</td>
<td>Drop-down list of OUs already entered in iPaaS.</td>
</tr>
</tbody>
</table>

Security model

The table below details the authentication and authorisation information that iPaaS will need to hold. It is important to state at this point that the objective is to provide a security model for iPaaS that is as flexible as possible, for example allowing the creation of arbitrary user groups with unique permission sets and any number of users. Realising this objective is ambitious; security models are complex and a challenge to get right but in order for iPaaS to be as practical as possible to the maximum number of different institutions it is important that we aim high in this area.

<table>
<thead>
<tr>
<th>#ID</th>
<th>Data item(s)</th>
<th>Description and purpose</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>SM1</td>
<td>User identifier (username)</td>
<td>Some form of identifier for a user, most likely the username from their institutional IT account.</td>
<td>Short text field, probably under 16 characters in length.</td>
</tr>
<tr>
<td>SM2</td>
<td>User credential (password)</td>
<td>Optional. The default approach should be to authenticate users against a central identity management system – iPaaS would therefore not be required to store passwords. Would only be used for the creation of ‘local’ iPaaS accounts.</td>
<td>Short text field, probably under 16 characters in length. Usual confidentiality measures in place.</td>
</tr>
<tr>
<td>SM3</td>
<td>Organisational unit</td>
<td>Identify the OU(s) that this user is associated with.</td>
<td>Selectable list of OUs in iPaaS.</td>
</tr>
<tr>
<td>SM4</td>
<td>Group membership</td>
<td>Identify the group(s) that this user belongs to</td>
<td>Selectable list of iPaaS user groups</td>
</tr>
<tr>
<td>SM5</td>
<td>Permission assignment</td>
<td>Identify all permissions assigned to this user individually. A user’s complete permission set should be a union of individually assigned permissions and those inherited from their group membership.</td>
<td>Selectable list of iPAMS permissions (e.g. ‘Modify Programme Marketing information’)</td>
</tr>
</tbody>
</table>
Groups

| SM6 | Group name | Description name of the user group – e.g. ‘Department of English Marketing Administrators’ | Short text field |
| SM7 | Users | Identify all the current members of a user group | Selectable list of iPaMS users. |
| SM8 | Organisational unit | Identify the OU(s) that this user is associated with. | Selectable list of OUs in iPaMS. |
| SM9 | Permission assignment | Identify all permissions assigned to this group. All members inherit the group’s permissions. | Selectable list of iPaMS permissions (e.g. ‘Modify Programme Marketing information’) |

Configuration Information
The details of this section will be fleshed out in the technical specification, precisely because much of the configuration information in the system is of a distinctly technical nature and concerns the environment in which the iPaMS application is being installed and run. The table below gives a high level overview of this information.

<table>
<thead>
<tr>
<th>#ID</th>
<th>Data item(s)</th>
<th>Description and purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>Database connection</td>
<td>Allows the system administrator to configure the details of the database to which the iPaMS system will connect.</td>
</tr>
<tr>
<td>C2</td>
<td>IDM integration (IDM: Identity Management)</td>
<td>Configuration of optional external systems to be used for authentication, e.g. LDAP, OpenSSO etc.</td>
</tr>
<tr>
<td>C3</td>
<td>Application environment</td>
<td>Specify whether the application is running in production, staging, test or development mode.</td>
</tr>
<tr>
<td>C4</td>
<td>System log files</td>
<td>Specify the location of various system log files.</td>
</tr>
</tbody>
</table>

Out-of-scope for iPaMS Phase 1 development
The following list details data that is either considered permanently out of scope for iPaMS or that may be incorporated into the application as part of future development:

- Support in the iPaMS application for Flexible Combined Honours which offers a degree structure that lets students study a range of subjects (FCH - University of Exeter terminology but other institutions provide similar opportunities)
- System should allow for multiple module titles, e.g. long, short, marketing version.
- iPaMS will not store any student data – i.e. it will not duplicate the programme and module diet information held within a system such as SITS.
- iPaMS will not store programme and module learning materials – i.e. it will not duplicate the resources held within a VLE such as Moodle or WebCT.
- Whilst the application will include the facility to search for and view versions of a programme or module at any stage in its history, the initial development phase to February 2010 will not implement workflow for these items (although it will not incorporate any features that could hinder workflow integration in the future).

Reporting Requirements
The exact scope of the required reporting functionality is impossible to define as each institution, organisational unit, user or group may have separate requirements. Reports could be produced from iPaMS data in a variety of ways:

1. At a basic level it is anticipated that reports would be made available within the system to view programme and module information in a flexible way which supports the selection of different fields and allows ordering by arbitrary combinations of those fields, for example ‘module codes,
titles and marketing overview ordered by latest year of revision’. Full system-wide ad-hoc reporting is not a required feature of iPaMS.

2. However, an institution would of course be free to use their existing business intelligence and reporting platforms to connect to and work with the iPaMS database, either directly or via the web service layer.

3. Bespoke reports could be built by external application developers, connecting to iPaMS-WS and transforming the data retrieved to meet a specific set of requirements. Publishing data in XML format through iPaMS-WS enables a rich variety of possibilities, such as support for different formats (web-based, PDF, etc.).

Use Cases

High level use case diagrams have been completed to model the interaction between each user group (known as an actor in a use case diagram) and the various functional areas and operations (use-cases) within the iPaMS system. To aid readability the use case diagrams have been separated into packages, such as ‘Programme management’, Module management’ etc, with an additional use case package diagram which aims to show the interactions between the different areas of the system. These diagrams are not intended to show every single system function down to the lowest level, but rather to provide an understandable representation of how various users will interact with the main operations:

- iPaMS use case package diagram

iPaMS Admin Application use case diagrams:

- Programme management
- Module management
- Programme structures
- Application security
- Organisational structure
- Configuration management

Target Platform(s)

The primary target platform for the iPaMS admin application is any standards compliant web browser running on a Windows, Mac or Linux operating system. Whilst it is not possible to list all combinations of browser / operating system the following will definitely be supported:

- Google Chrome (Windows, Mac, Linux)
- Firefox v2+ (Windows, Mac, Linux)
- Internet Explorer 7+ (Windows)
- Safari (Mac, Windows)

Defining target platforms for the iPaMS-WS component is more difficult. The advantage of providing publishing and updating through SOAP and REST style web services is that a multitude of client application types can be supported.

Features

The main features of iPaMS can be identified from the details provided in the Users, Data and Use Cases sections above and this section provides a summary list and further expansion and explanation of certain features, categorising them as essential or desirable for the first phase (to February 2011) of iPaMS development:

Essential

iPaMS Admin Application

1. General navigation and usage:
   - System should support form-based login via a web browser.
   - Users should be able to logout of the system from any page.
• The user interface, which is discussed further below, should remain consistent across all functional areas within the admin application, meaning that the navigation at all levels should appear in the same location and the style should remain the same throughout.

2. Login securely to the iPaMS web application and view dashboard screen containing links to functional management areas appropriate to a user's organisational unit, group membership and permissions. Functional areas of iPaMS are listed below.

3. Programme descriptions (including marketing information):
   • Search, read, create, update (revise version), archive, delete (sys admin only), copy.
   • View revision history

4. Programme Structures (per programme, per academic year):
   • Create, read, update, add and remove modules, stages, set extensions, copy. Versioning for programme structures is covered by point #2 above, as a structure relates to a single programme description.

5. Module descriptions (including marketing information):
   • Search, read, create, update (revise version), archive, delete (sys admin only), copy.
   • View revision history

6. Organisational structure:
   • View, edit, add, delete organisational unit and assign parent record if required (e.g. create relationship between 'College of Humanities' and 'English department')
   • Create and remove associations between programmes, module, users or groups and organisational unit(s).

7. Permissions:
   • View permissions.
   • Assign or revoke permissions to users and groups
   • Create, update and delete permissions. The implementation of this feature is dependent upon the facilities available within the chosen development framework. It may only be possible to configure a static set of permissions based on the current database structure.

8. Users:
   • Search for and view users by name, group, organisational unit, assigned permissions
   • Create, update and remove users. Depending on the configuration, creating a new user may rely on looking up an existing record in some external system such as a LDAP-enabled directory.

9. Groups:
   • Search for and view groups by name, organisational unit, assigned permissions.
   • Create, update and remove groups.
   • Add or remove users from groups.

10. Alerts and prompts:
    • The system should provide a flexible framework for configuration of email alerts to be sent to specified users or groups of users when certain actions are performed in the system. For example, the system may be configured to send an email to all College of Humanities Marketing administrators when a change is made to a programme within the same organisational unit.
    • Permission to create new alerts would be granted initially by a system administrator.

Desirable
This section details features that either:
• Exist in the wish-list category for Phase 1 development of iPaMS
• Will need to be incorporated into the requirements for an extension beyond February 2011 into Phase 2.
• May be developed as part of an external system which consumes data through iPaMS-WS.
1. Approval workflow (process tracking)
   - Programme specifications and Module descriptions
   - New approvals, withdrawals, amendments
   - Business and Academic approval
   - Objective is to replace manual ‘follow the flowchart’ workflow whereby MS Word documents are created, emailed around for changes or comments and then finally input into a database driven application if approved.
   - System would need to map existing business procedures to flexible workflow model within iPaMS.
   - In increasing order of complexity to implement and convenience to the end user, approval workflow could take the following forms in iPaMS:
     a. In addition to the proposed version control fields defined for programmes and modules as outlined in the Data section above, additional fields would be provided to support a step-by-step approval process, but users would still be required to work manually through flowcharts maintained outside of iPaMS.
     b. An external workflow system could be used to automate the manual approval process, with hooks provided into iPaMS at each stage
     c. The approval process itself is built into iPaMS.

2. Withdrawal of non-progressed programme and module changes

3. Full ad-hoc reporting on iPaMS data, or inclusion of bespoke complex reports to meet the requirements of a particular institution.

4. Automatic production of HEAR transcript from within iPaMS (requires Integration

The iPaMS Web Wervice module (iPaMS-WS) will provide the single access point for external applications requiring integration with iPaMS. It is anticipated that integration with other key systems within a HE organisation will be required, including student records (e.g. SITS) and virtual learning environments (e.g. Moodle).

Where iPaMS requires a feed of data from an external application, the functionality would be made available through one or more SOAP or REST web service methods and the structure and format of the data would be validated against the iPaMS XML Schemas. Only well-formed, validated XML documents would be accepted on incoming data feeds.

In cases where external applications need to consume data that is mastered in iPaMS, the web service layer would again be used to provide such data through a selection of methods (e.g. getProgrammes, getModules). However, in the case of outgoing data feeds the format of the information could be specified by the client, to allow for greater flexibility in use. For example, the web service may provide data feeds in XML, CSV or JSON (JavaScript Object Notation) formats, depending on the needs of the client.

On the other side, the iPaMS project plan also covers development of SITS and Moodle interfaces which would interact with iPaMS-WS. In environments where the SITS StuTalk application was available it would be possible to configure SITS to accept and process XML data from iPaMS-WS given the openly available XML schema definitions and appropriate levels of authentication, if applicable. Without StuTalk it may be necessary to develop a bespoke interface to consume and process data from iPaMS-WS.

In terms of integration with Moodle, it is anticipated that one of two approaches will be taken; either the web service tools that should become available in Moodle v2 would be used, alternatively the iPaMS project will cover development of a plugin for Moodle.

Expandability

It is understood that the business requirements of each HE institution differ in respect to the management and use of programme specifications and module descriptions. In order to maximise the applicability of the iPaMS toolkit to a variety of needs, this functional specification
is deliberately lightweight in terms of defining specific requirements for any single institution. The objective is to provide a toolkit that is flexible but still meets certain core requirements deemed essential for any institution.

User Interface

Screen shots are under development.

Time Scale

The iPaMS Project Plan document details the key milestones for the design, development, testing, implementation and evaluation of the iPaMS and iPaMS-WS application toolkit.

Document Version Control

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Author</th>
<th>Sections Changed</th>
<th>Amendment Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/11/2010</td>
<td>1.0</td>
<td>Tom Bunting</td>
<td>All</td>
<td>Initial draft</td>
</tr>
<tr>
<td>15/11/2010</td>
<td>1.1</td>
<td>Tom Bunting</td>
<td>Programmes and Modules data, marketing fields, desirable features</td>
<td>Added 'Marketing version required' column to programmes and modules data. Split Modules data into logical groupings (as per suggested changes to module XML schema), expended on workflow requirements</td>
</tr>
</tbody>
</table>

Signed:                                Signed:                                  Date:  
__________________  ____________________  ________________

Development Lead            User/Client Representative
Appendix

Existing systems

Appendix 1.0:
University of Exeter Business School, current programme and module system

1. Name of system
   Not sure it has a name! It resides within what is imaginatively called the Website admin area, along with a bunch of other systems for displaying website/intranet content such as events, publications etc. – generally anything which was not easily portable across to T4

2. URL(s) – both of ‘admin’ application and any ‘consumers’
   Website admin area
   
   | http://business-school.exeter.ac.uk/admin/ |
   | programmes - http://business-school.exeter.ac.uk/admin/programmes |
   | modules - http://business-school.exeter.ac.uk/admin/modules |
   | Also ties in with |
   | Past exam papers http://business-school.exeter.ac.uk/admin/past_exam_papers/ |
   | Module questionnaires http://business-school.exeter.ac.uk/admin/module_questionnaires/ |
   | Staff profiles http://business-school.exeter.ac.uk/admin/staff/view/ |

External website

There may be many of these, but I’ll try and include at least an example of each type

Programmes

| http://business-school.exeter.ac.uk/programmes/undergraduate/accountingandfinance/ba_af/ |
| (marketing information) |
| http://business-school.exeter.ac.uk/programmes/undergraduate/accountingandfinance/ba_af/programmestructure/ |
| (programme structure) |

Modules

| (marketing information) |
| (full module spec) |

(approximately 271 in total)
Programme lists

http://business-school.exeter.ac.uk/programmes/international/moduleinformationandacademicissues/

http://business-school.exeter.ac.uk/programmes/undergraduate/specifications/ (and a pg version exists, shows latest pdf spec for each)

Module lists

http://business-school.exeter.ac.uk/programmes/undergraduate/modules/ (and a pg version exists)

Staff profiles

http://business-school.exeter.ac.uk/about/whoswho/index.php?web_id=Richard_Jackson&tab=teaching (pulls taught modules info from latest module specs)

Intranet

http://intranet.exeter.ac.uk/business-school/staff/education/modules/

http://intranet.exeter.ac.uk/business-school/staff/education/modules/module/index.php?id=664

http://intranet.exeter.ac.uk/business-school/staff/education/modules/discontinuedmodules/

http://intranet.exeter.ac.uk/business-school/staff/education/modules/modulesawaitingaccreditation/

http://intranet.exeter.ac.uk/business-school/staff/education/modules/modulesbysemester/

http://intranet.exeter.ac.uk/business-school/staff/education/modules/reviews/

http://intranet.exeter.ac.uk/business-school/staff/education/modules/reviews/form/

http://intranet.exeter.ac.uk/business-school/staff/education/programmes/

http://intranet.exeter.ac.uk/business-school/staff/education/programmes/programme/index.php?id=230

http://intranet.exeter.ac.uk/business-school/students/undergraduate/modules/ (and a pg version exists)

http://intranet.exeter.ac.uk/business-school/students/undergraduate/modules/module/index.php?id=664

http://intranet.exeter.ac.uk/business-school/students/undergraduate/modules/pastexampapers/

3. Platform (PHP, JSP etc)
   PHP/MySQL

4. College / departments covered
   The Business School (includes depts Accounting, Economics, Finance, Leadership, Management)

5. Main admin users / groups (e.g. ‘Programme administrators in Business School’). Approximate numbers of users would also be good.
Marketing team: 5 users

These guys are responsible for the marketing blurb, and one gets an email alert when any of the users in the education group below make a change or addition to any programme or module – on either the marketing blurb or the specification itself.

Education/student services/QA: 5 users

That’s 10 users in total, plus occasional office temps.

6. Databases involved (either for pulling data from, e.g. DWAI / SITS or storing data in)
None apart from our stand-alone MySQL tables, but does require link with staff profiles system for taught modules links to work to/from staff web profiles

7. Data items covered (programmes / modules / both / ‘diets’)
Programmes and modules both included in 4 db tables (one for ‘top level’ prog info, one for ‘top level’ module info, one for yearly prog structures and one for yearly module descriptors).

Programme structures are simply a free text field with links to modules hard coded in the WYSIWYG editor. Hard coded links cause some problems if prog structures appear in different places (e.g. intranet) so not ideal.

There is no separate db table linking modules and programmes but module admin page can tell administrators which modules are include in which current programme structures by simply searching text for module codes. Not particularly sophisticated but it does the job!

8. Brief description (few paragraphs at most) of the purpose and principle operations performed by the application, e.g. maintaining module descriptors. If any standards are adhered to (e.g. for programme or module specs) it would be useful to mention these
In addition to things covered above such as general maintenance (add, edit, delete) of module & programme descriptors and their ‘top level’ information (Code, title, credits etc.) and marketing blurb, there are some other features of our system...

- Can import a module spec from a word document by converting to docbook.xml format using OpenOffice
- Copy a module/programme spec – useful when small changes are required so you don’t have to re-enter all the data
- Extend a module/programme spec for another academic year (and allow people to roll it back when they change their mind!)
- Email alerts to marketing team when education team do something (detailed in Q5). Some sort of marketing blurb probably ought to be mandatory but could cause problems for a module which needs urgently adding and no-one has written the blurb. Could be taken from the ‘aims’ section and reviewed later I guess?
- Programmes admin locks db entry from other users while someone else is editing it (intended to extend this to modules too when there is time)

Appendix 1.1
University of Exeter College of Humanities Module management system – details:

1. Name of system: undefined.

2. URL(s) – both of ‘admin’ application and any ‘consumers’
   Admin – https://intranet.exeter.ac.uk/humanities/staffonly/modules/
   Consumers (for example):
3. **Platform (PHP, JSP etc)**  
   PHP & MySQL

4. **College / departments covered**  
   Humanities (all the former HuSS departments, and we’re starting to use it with the former SALL departments as we complete their redesigned sites)

5. **Main admin users / groups**  
   Access limited to System administrator and Web Marketing Officer (WMO). Previously in the days of HuSS the school marketing manager also had access to add module “blurbs”. The intention was/is that department admins would have access also so they can manage the data for their modules.

6. **Databases involved**  
   It draws on the data from SITS stored in the DWAI database (dwai.si_mod and dwai.si_mav) and uses the DWAI module description table (dwai.mods_mds) to identify which modules have a description. Additional data is stored in the Humanities database – module blurbs, groupings etc

7. **Data items covered**  
   Modules (lists, groups within lists, and blurbs)

8. **Brief description**  
   It gives us the facility to set up and manage different lists of modules to appear on the web, and control how modules are grouped within those lists. It allows an additional module summary, not included in the formal module description, to be included for marketing purposes. It will link to the module description in DWAI, or a PDF depending on what’s available for a particular module.

---

**Appendix 1.2**

University of Exeter College of Engineering, Mathematics and Physical Sciences Programme & Module management system – details:

1. **Name of system**: UDB

2. **URL(s) – both of ‘admin’ application and any ‘consumers’**:  
   https://udb.emps.ex.ac.uk. This is for editing the data. The data is then dynamically displayed on our College website using php scripts.

3. **Platform (PHP, JSP etc)**: MySQL database with PHP front end.

4. **College / departments covered**: EMPS: Engineering, Maths, Computing, Physics, CSM

5. **Main admin users / groups**: The principle admin users are student services administrators (3) in EMPS. Additionally the programme coordinators and module leaders have access to their own programme/module. Technical support staff also have access.

6. **Databases involved**: This is a standalone system. It is all in a single database, but there are many tables to allow information to be kept across multiple years. We also include reading lists, lecturer loading information and the ability to upload resources (e.g. files with important information).

7. **Data items covered**: The UDB is a comprehensive system. As mentioned, it covers programmes & modules, but it is also our main user database for displaying staff lists (profiles/photos), building information (rooms/telephones/maps), research grants & finance, mark entry (student marks for coursework and normalising them and presenting them anonymously), publications for postgrads, project posters, and a permissions system so that we can enable certain users to access certain parts of the UDB. Linking in to this we also have email lists which obtain student data from DWAI and enable staff members to send emails to modulecode@emps.ex.ac.uk, programme@emps.ex.ac.uk, tutorialgroup@emps.ex.ac.uk and to yeargroup@emps.ex.ac.uk.
8. **Brief description:** I think I have covered this already. The system is an electronic copy of the official programme and module descriptors and allows linking between the two. The official RTFs that are required centrally are generated from the UDB. The setup is such that a programme contains many years of data, with the top level data (code/lecturers available) applying to all years within that programme. The modules are the same. Additionally we have “user friendly” data included so as to present something on the website that is more interesting than the dry specs! We also have semester/term information and the credits of each module to enable us to automatically list the modules that are available within a programme and also to provide lists of “optional modules totally X credits”.