Requirements Specification

Databases and Web Applications Laboratory (LBAW)
Bachelor in Informatics Engineering and Computation (L.EIC)

Sérgio Nunes
Dept. Informatics Engineering
FEUP · U.Porto

Outline

- Information Systems Development
- EBD Component
- A1. Project Presentation
- Requirements Analysis
- A2. Actors and User Stories

Information Systems Development

Information Systems Development

- Core activities in information systems development.
  - **System analysis**: identify the problems, establish information requirements;
  - **System design**: create design specifications;
  - **Programming**: translate specifications to software program code;
  - **Testing**: assess if the system is able to produce the right results;
  - **Deployment**: changing from the old system to the new system;
  - **Production and maintenance**: operate and evaluate the system.

Development Methodologies

- **Traditional systems life cycle (i.e. waterfall)**:
  - development is organized in formal stages, with a strong separation between end users and developers;
  - relevant for large complex systems with tight control over the development process.
- **Prototyping**:
  - rapid building of inexpensive experimental systems for end users to evaluate;
  - iterative process that actively promotes systems design changes;
  - relevant for contexts with high uncertainty;
  - more likely to produce systems that fulfill user requirements.
- **Agile development** breaks large projects into small sub-projects to be completed in short time periods using iteration and continuous feedback.

Development in LBAW

- In LBAW we adopt a structured development approach, organized in a series of sequential stages with well-defined outputs (artifacts).

- This approach is justified due to the focus on learning and calendar restrictions.

- In professional settings, web-based information systems development is usually associated with agile methodologies, with early and regular end-users involvement in the process.

Requirements Specification

Requirements Specification Component

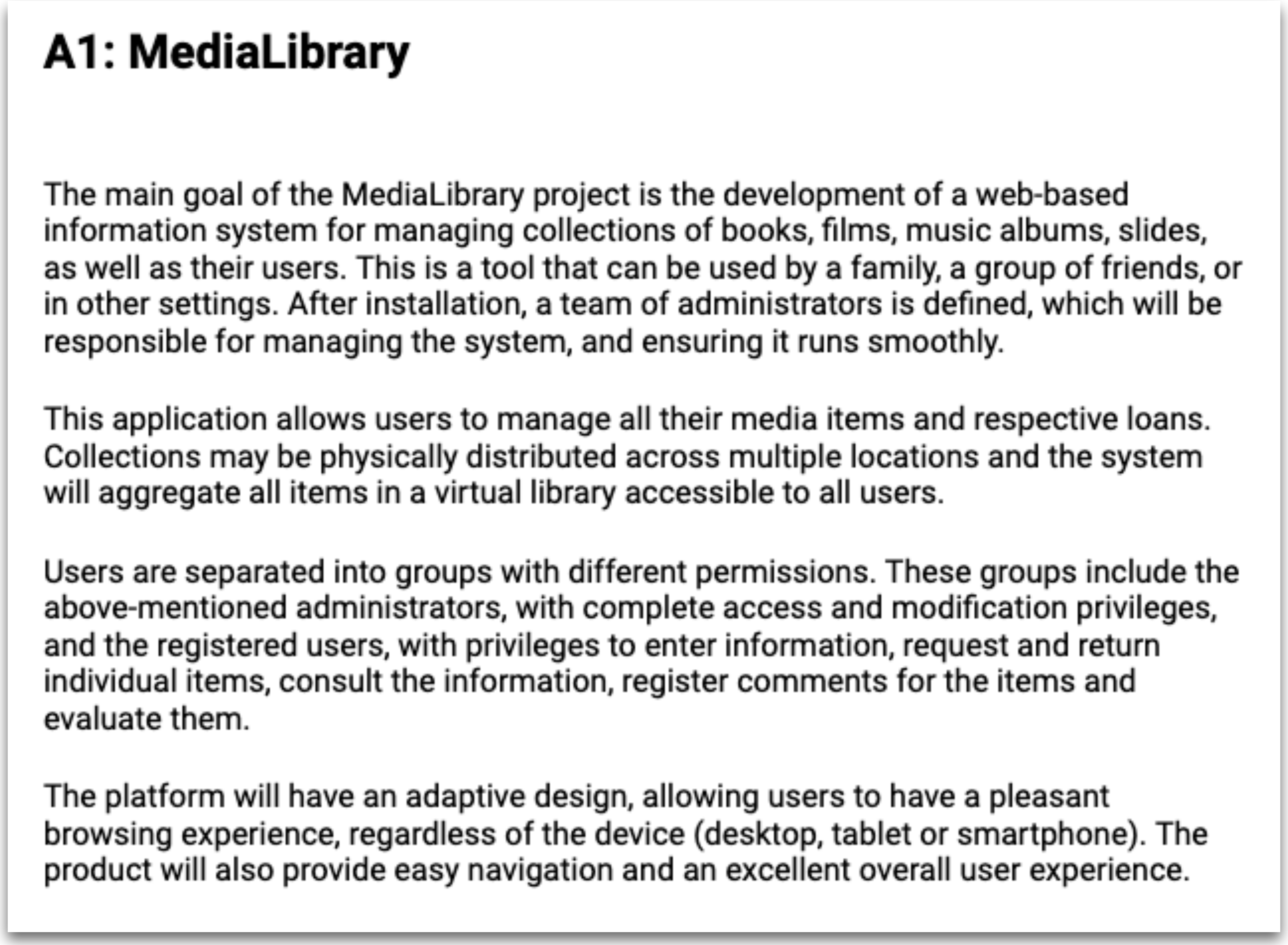
- *This component groups the artifacts related to the Requirements Specification of the system to be developed during the project.*
- *These artifacts are usually included in the Requirements Document of a system and developed in the initial phase of analysis by a team that includes all project stakeholders.*
- The ER component includes three sections:
    - A1. Project Presentation
    - A2. Actors and User Stories
    - A3. Information Architecture

A1. Project Presentation

A1. Project Presentation

- This artifact introduces the context and motivation of the project, and briefly describes the web information system to be developed. It includes the goals of the project and lists the main features that should be supported, together with the identified access groups.

A1. MediaLibrary Example



A1: MediaLibrary

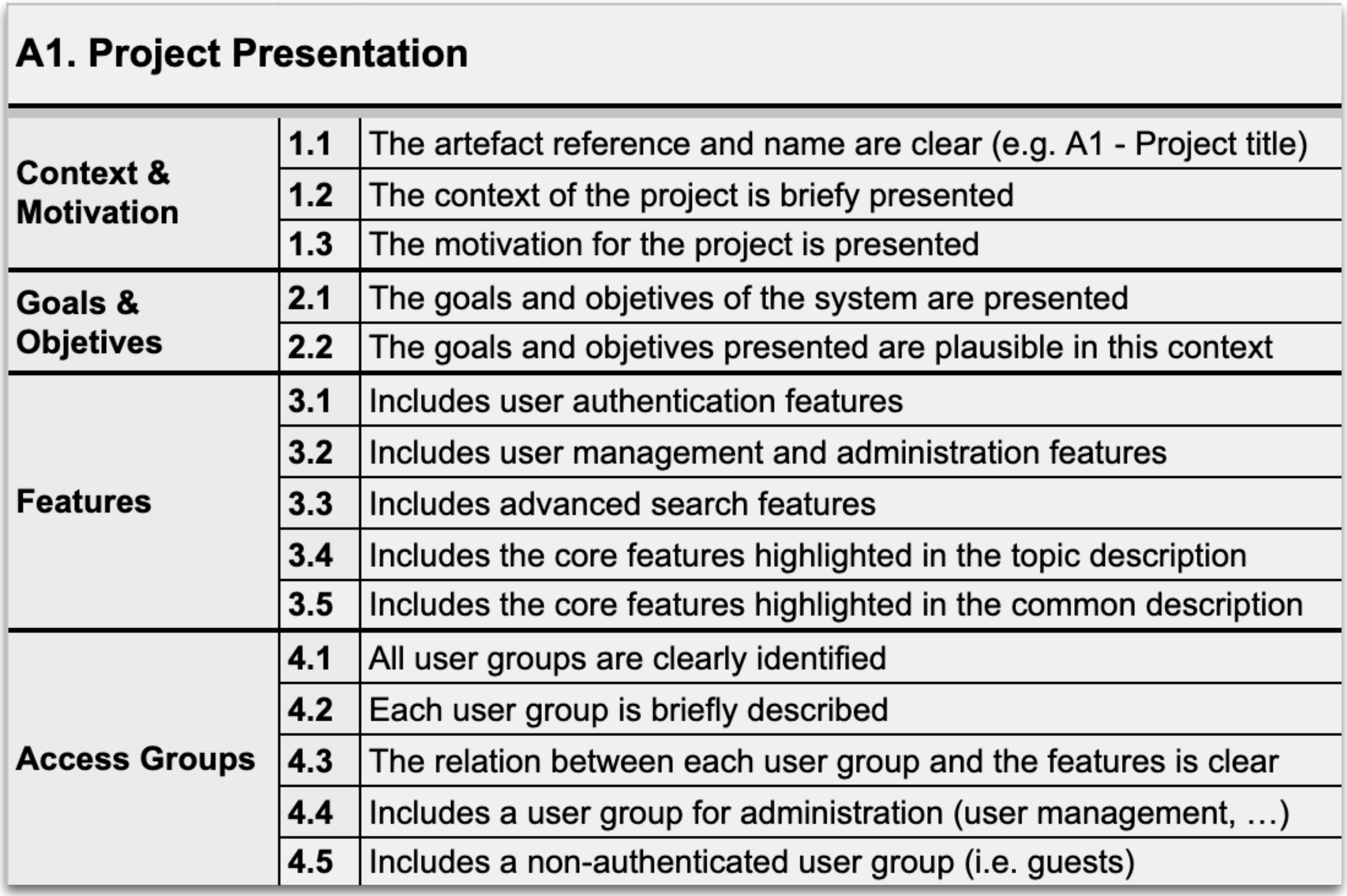
The main goal of the MediaLibrary project is the development of a web-based information system for managing collections of books, films, music albums, slides, as well as their users. This is a tool that can be used by a family, a group of friends, or in other settings. After installation, a team of administrators is defined, which will be responsible for managing the system, and ensuring it runs smoothly.

This application allows users to manage all their media items and respective loans. Collections may be physically distributed across multiple locations and the system will aggregate all items in a virtual library accessible to all users.

Users are separated into groups with different permissions. These groups include the above-mentioned administrators, with complete access and modification privileges, and the registered users, with privileges to enter information, request and return individual items, consult the information, register comments for the items and evaluate them.

The platform will have an adaptive design, allowing users to have a pleasant browsing experience, regardless of the device (desktop, tablet or smartphone). The product will also provide easy navigation and an excellent overall user experience.

A1. Checklist



A1. Project Presentation

Context & Motivation	**1.1**	The artefact reference and name are clear (e.g. A1 - Project title)
	1.2	The context of the project is briefy presented
	1.3	The motivation for the project is presented
Goals & Objetives	**2.1**	The goals and objetives of the system are presented
	2.2	The goals and objetives presented are plausible in this context
Features	**3.1**	Includes user authentication features
	3.2	Includes user management and administration features
	3.3	Includes advanced search features
	3.4	Includes the core features highlighted in the topic description
	3.5	Includes the core features highlighted in the common description
Access Groups	**4.1**	All user groups are clearly identified
	4.2	Each user group is briefly described
	4.3	The relation between each user group and the features is clear
	4.4	Includes a user group for administration (user management, …)
	4.5	Includes a non-authenticated user group (i.e. guests)

A2. Actors and User Stories

Usage Modeling

- You need to understand how people will work with your system.
- An important part of software development is to explore the requirements.
- Usage modeling explores how people work with a system.
- Common usage modeling techniques include: use cases, user stories, features.

- LBAW: obtain user stories from given features, plus propose new set of features.

Established Features

- A collection of features are established for all themes:
  - LBAW Project Themes
- A set of common features for all projects:
  - 0. Common Requirements
  - Includes: user profiles, search, notifications, administration, help, etc.
- And specific features for each theme, e.g.:
  - 1. Collaborative News
  - 2. Social Network
  - ...
- Also includes supplementary requirements, common and theme-specific.

A2. Actors and User Stories

- This artifact contains the specification of the actors and their user stories, serving as agile documentation of the project requirements.

- Based on both the common and the theme-specific functional requirements, each group is expected to identify and describe the actors and the user stories for their project.

- The full implementation of the features identified as required corresponds to a grade of 18 (in 20). **The additional 2 points are for features proposed by each group, considering the quality and innovation of these features.**

- LBAW Project Themes

Actors

- An Actor is any entity that interacts with the system being specified.
- This may include persons (not just the end-user) and entities external to the system.
- Actors are always external to the system being modeled.

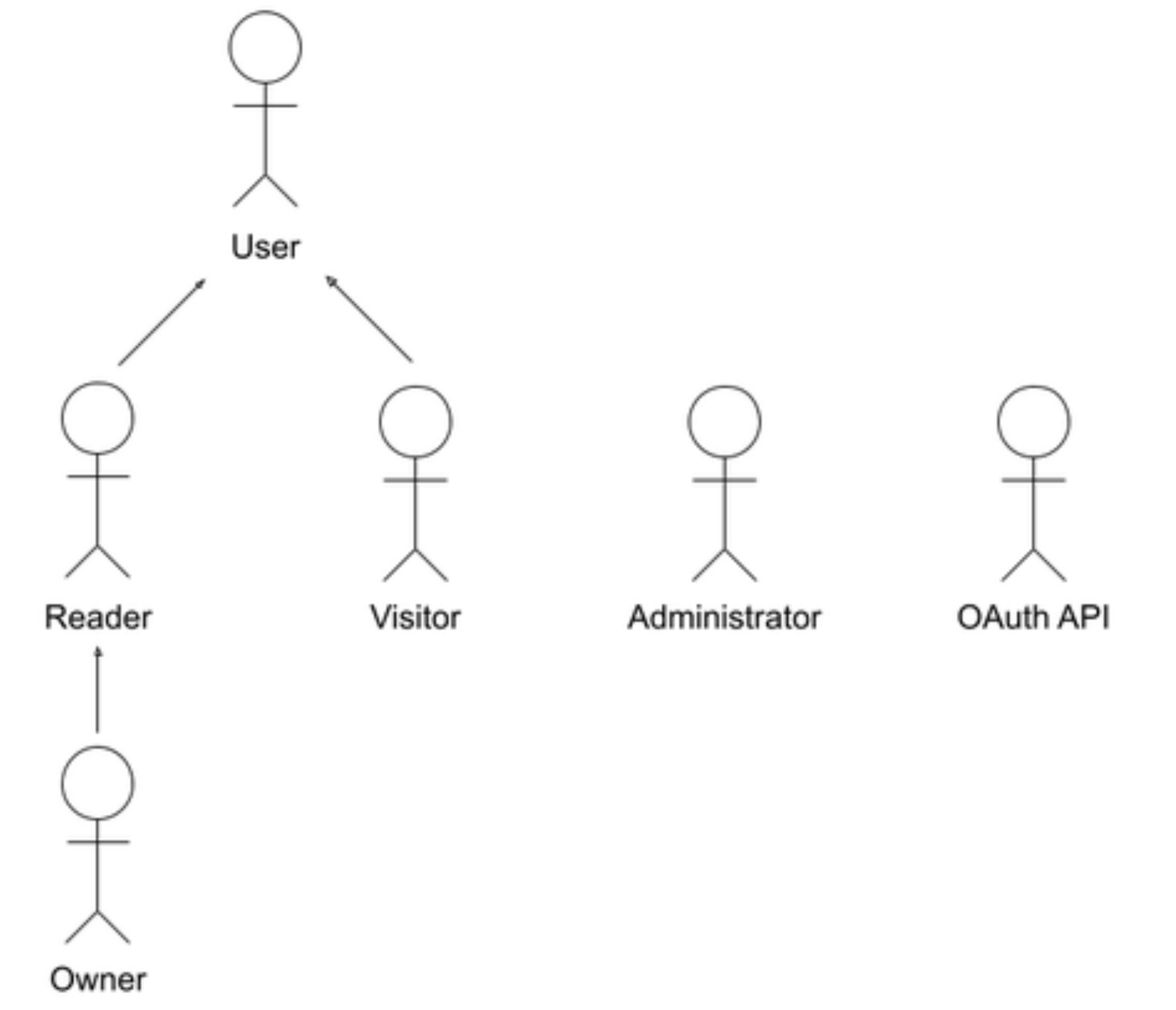


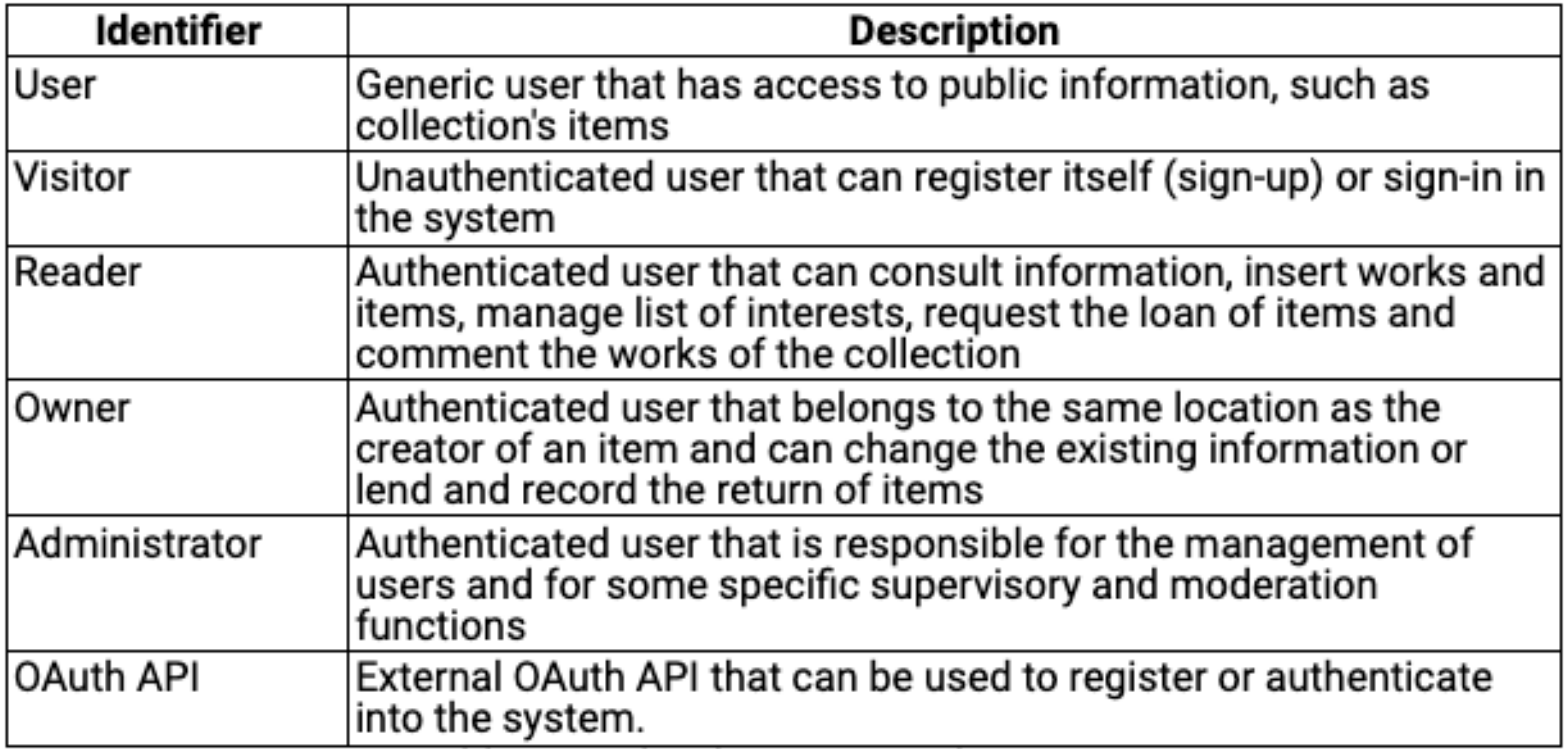
Figure 1: MediaLibrary actors.

Identify Actors

- Ask the following questions to help you identify actors:
    - Who is the main customer of your system?
    - Who obtains information from this system
    - Who provides information to the system?
    - Who installs the system?
    - Who operates the system?
    - Who shuts down the system?
    - What other systems interact with this system?
    - Does anything happen automatically at a preset time?
    - Who will supply, use, or remove information from the system?
    - Where does the system get information?

From: The Object Primer (Chapter 5)

Description of Actors



Identifier	Description
User	Generic user that has access to public information, such as collection's items
Visitor	Unauthenticated user that can register itself (sign-up) or sign-in in the system
Reader	Authenticated user that can consult information, insert works and items, manage list of interests, request the loan of items and comment the works of the collection
Owner	Authenticated user that belongs to the same location as the creator of an item and can change the existing information or lend and record the return of items
Administrator	Authenticated user that is responsible for the management of users and for some specific supervisory and moderation functions
OAuth API	External OAuth API that can be used to register or authenticate into the system.

Table 1: MediaLibrary actors description.

User Stories

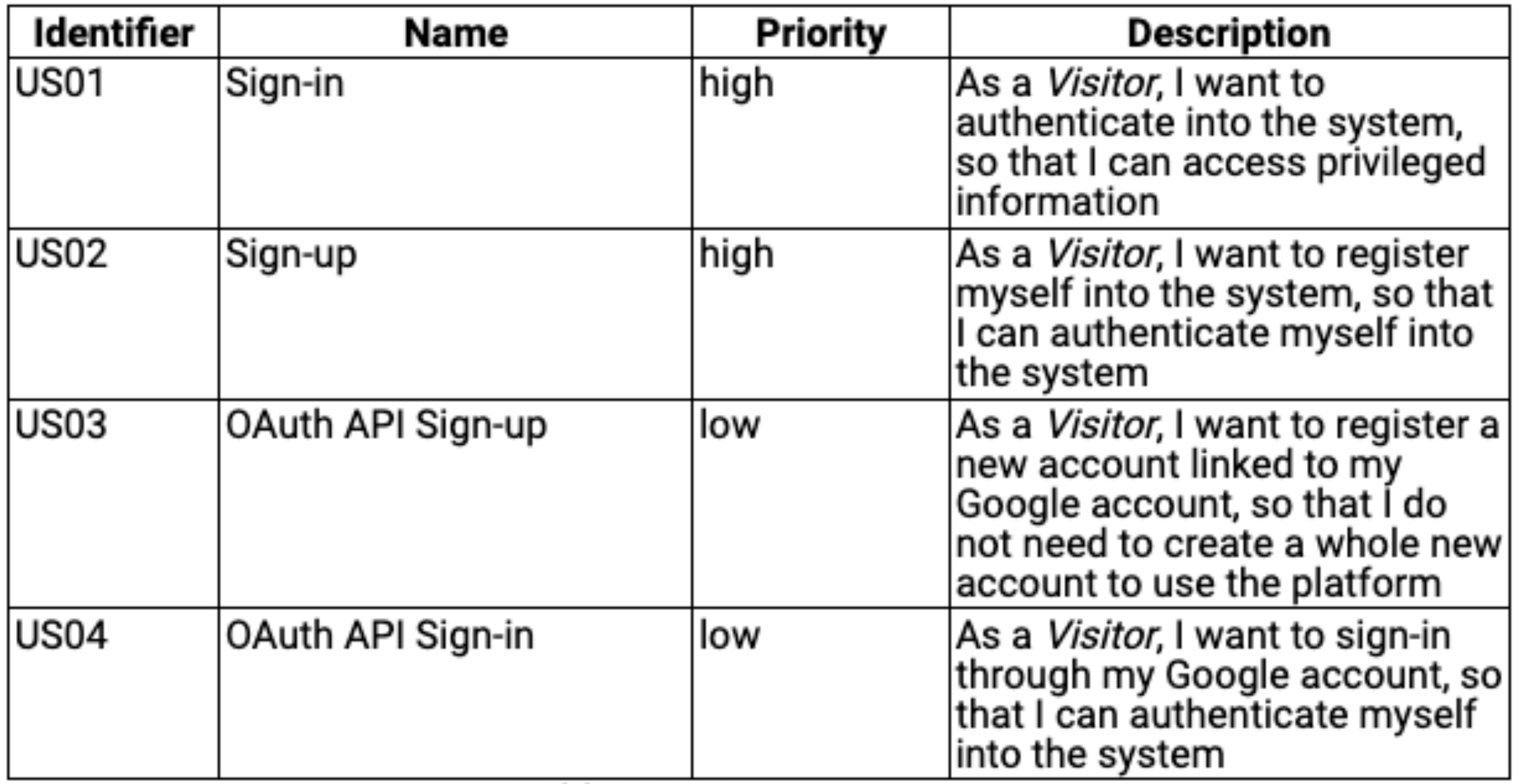
- A User Story is a high-level definition of a requirement.
- US are brief descriptions of a potential interaction with the system by one of its users.
- US are focused only on interaction requirements and not on the technical aspects of the design of the system.
- US have a priority: high or essential, medium or conditional, and low or optional.
- A team of expert analysts also includes an estimate of the effort required for its implementation.

Defining Priorities

- You should define the priorities according to the following rules.
- **High** priority should be used for user stories that must be implemented in the vertical prototype (A8). The features related to these user stories are identified with an asterisk in the themes’ presentation.
- **Medium** priority should be used for user stories that must be implemented in the final product (A9). These include the mandatory features described in each theme (underlined) plus the features that you introduce for the remaining 10% (innovation).
- **Low** priority should not be used unless you want to explore more advanced or extra features. You can identify these features with “low” priority and decide later if they are to be implemented. There is no penalty if these are not implemented.

User Stories Template

- Each user story must follow the template:
  - As a [user], I want [function], so that [value]
- Do not forget the three "w"s — who, what, why.



Identifier	Name	Priority	Description
US01	Sign-in	high	As a *Visitor*, I want to authenticate into the system, so that I can access privileged information
US02	Sign-up	high	As a *Visitor*, I want to register myself into the system, so that I can authenticate myself into the system
US03	OAuth API Sign-up	low	As a *Visitor*, I want to register a new account linked to my Google account, so that I do not need to create a whole new account to use the platform
US04	OAuth API Sign-in	low	As a *Visitor*, I want to sign-in through my Google account, so that I can authenticate myself into the system

Table 2: Visitor user stories.

Supplementary Requirements

Supplementary Requirements

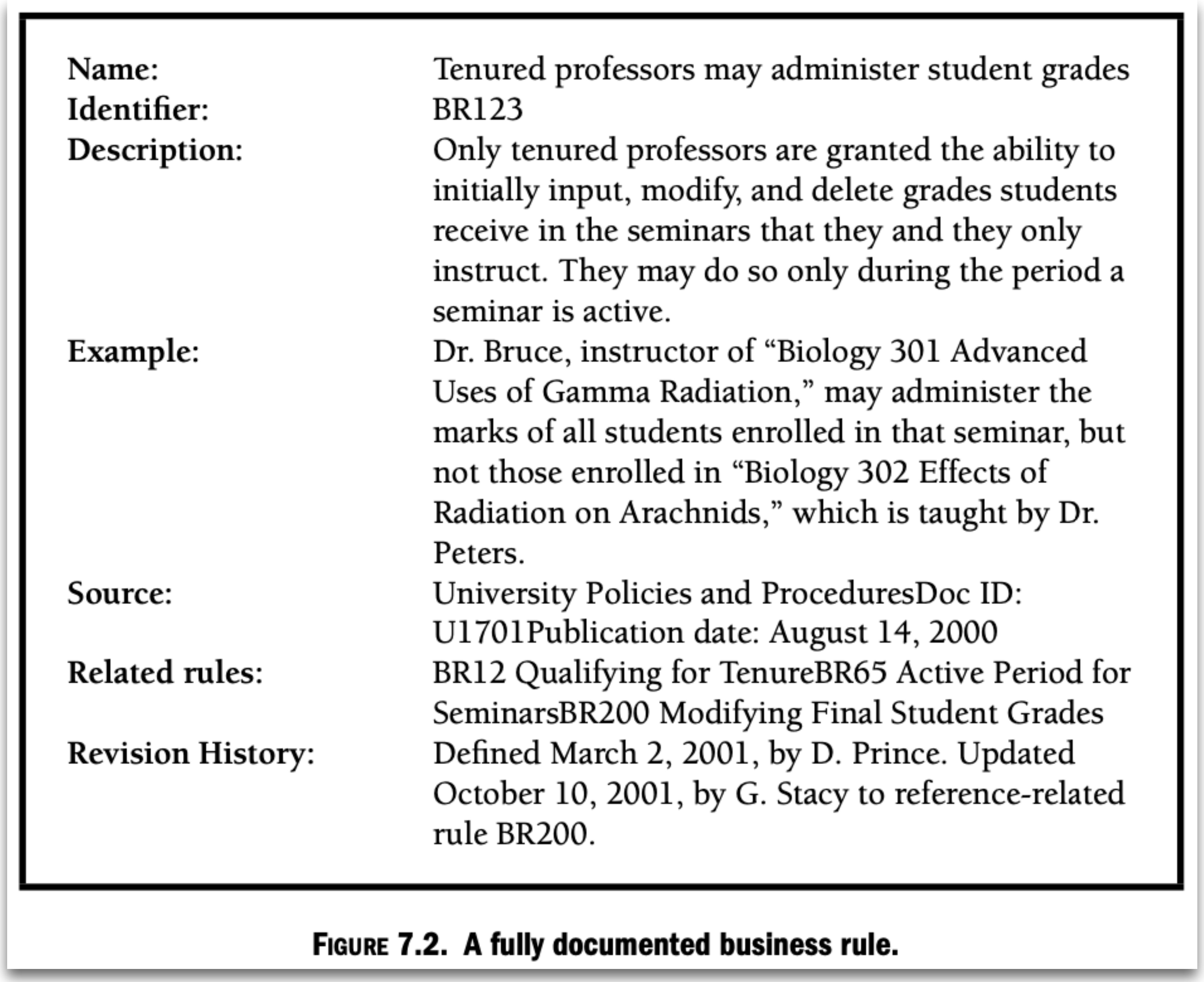
- The supplementary requirements are also presented as a section, and includes business rules, technical requirements, and other restrictions on the project.

- A business rule defines or constrains one aspect of the business, with the intention of asserting business structure or influencing business behavior.

- Technical requirements are concerned with the technical aspects that the system must meet, such as performance-related issues, reliability issues and availability issues.

- A restriction on the design limits the degree of freedom in the search for a solution.

Example Business Rule



Name: Tenured professors may administer student grades

Identifier: BR123

Description: Only tenured professors are granted the ability to initially input, modify, and delete grades students receive in the seminars that they and they only instruct. They may do so only during the period a seminar is active.

Example: Dr. Bruce, instructor of “Biology 301 Advanced Uses of Gamma Radiation,” may administer the marks of all students enrolled in that seminar, but not those enrolled in “Biology 302 Effects of Radiation on Arachnids,” which is taught by Dr. Peters.

Source: University Policies and ProceduresDoc ID: U1701Publication date: August 14, 2000

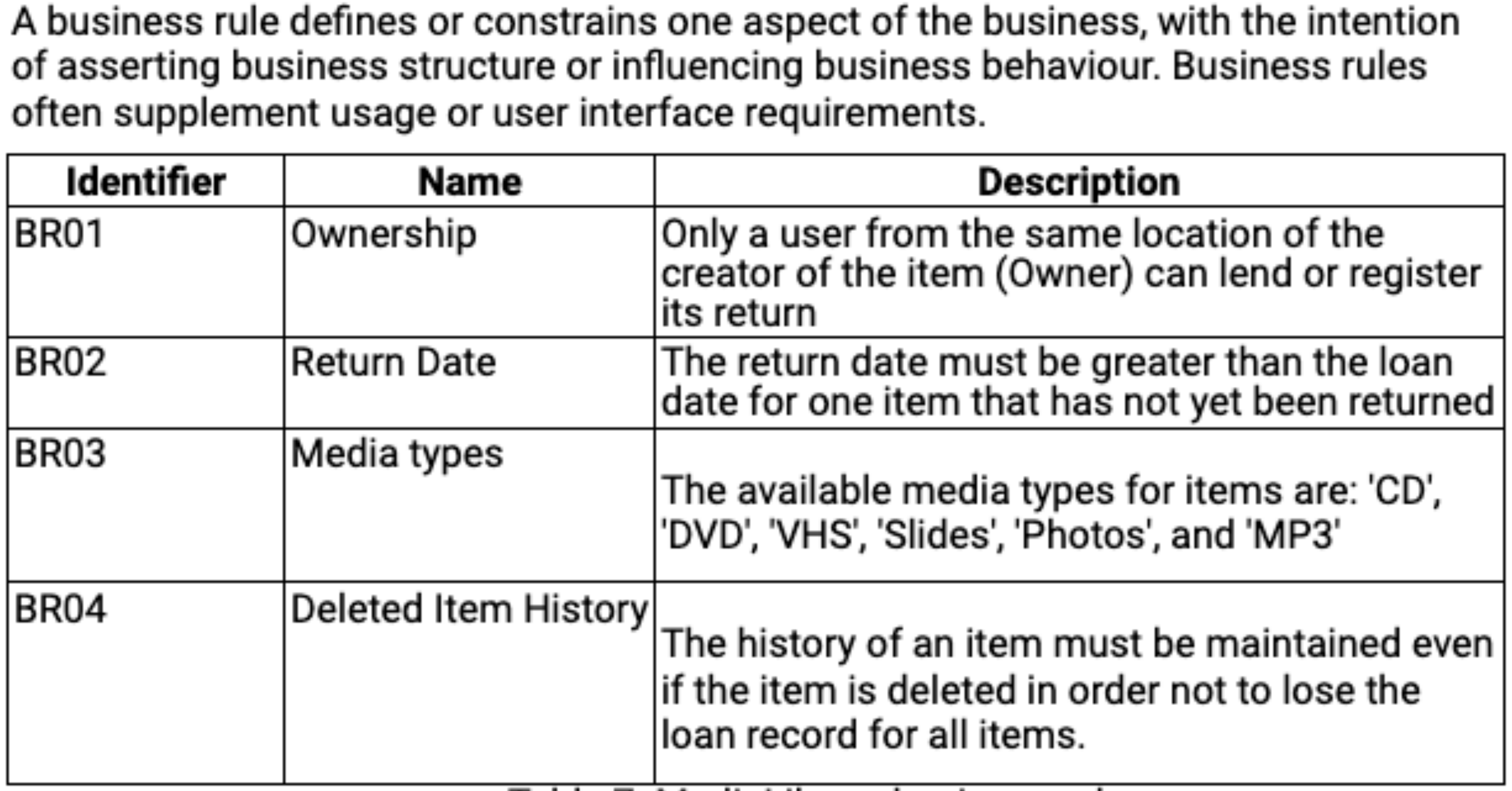
Related rules: BR12 Qualifying for TenureBR65 Active Period for SeminarsBR200 Modifying Final Student Grades

Revision History: Defined March 2, 2001, by D. Prince. Updated October 10, 2001, by G. Stacy to reference-related rule BR200.

FIGURE 7.2. A fully documented business rule.

Business Rules

- The supplementary requirements are also presented as a section, and includes business rules, technical requirements and other restrictions on the project.



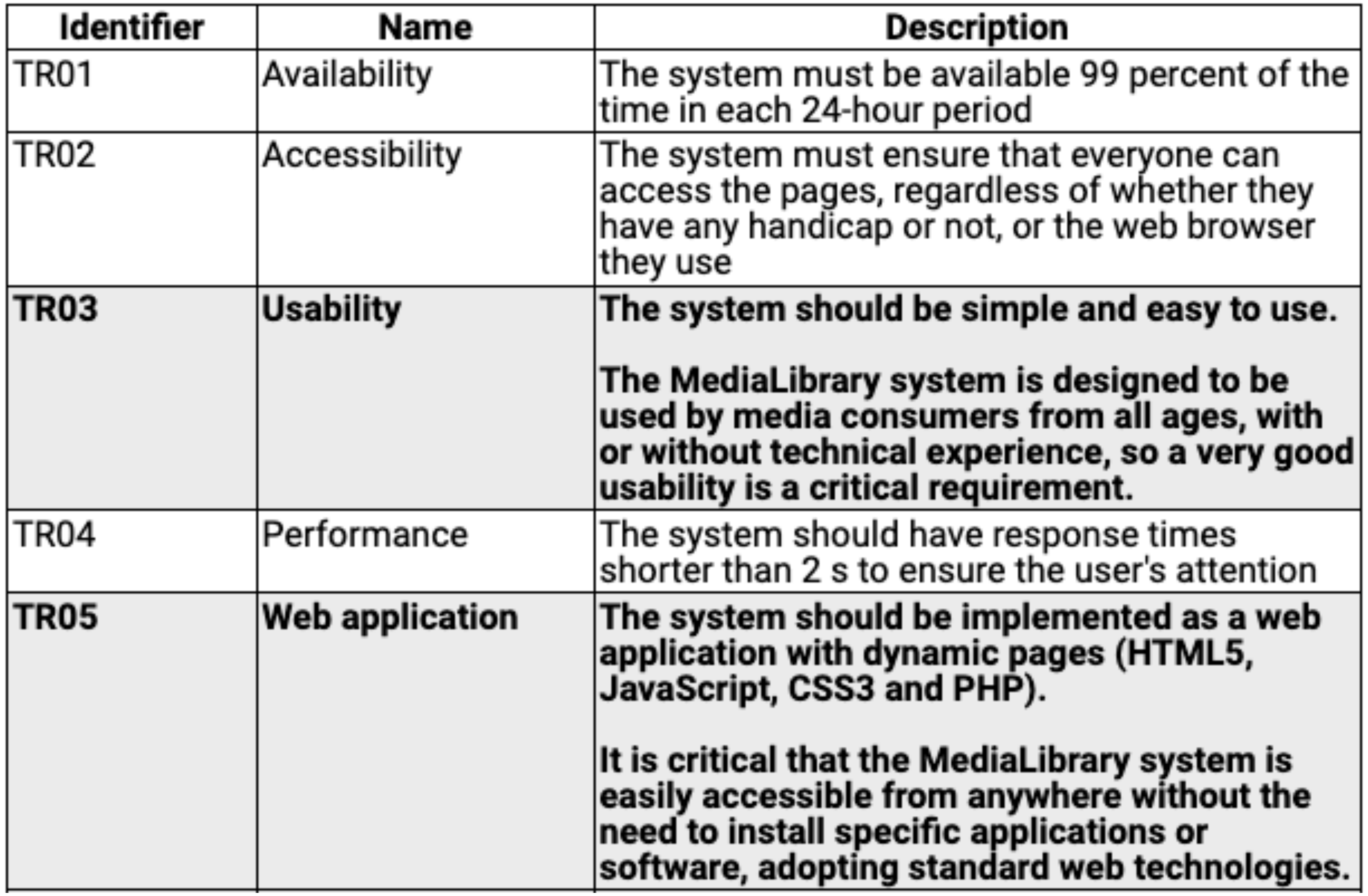
A business rule defines or constrains one aspect of the business, with the intention of asserting business structure or influencing business behaviour. Business rules often supplement usage or user interface requirements.

Identifier	Name	Description
BR01	Ownership	Only a user from the same location of the creator of the item (Owner) can lend or register its return
BR02	Return Date	The return date must be greater than the loan date for one item that has not yet been returned
BR03	Media types	The available media types for items are: 'CD', 'DVD', 'VHS', 'Slides', 'Photos', and 'MP3'
BR04	Deleted Item History	The history of an item must be maintained even if the item is deleted in order not to lose the loan record for all items.

Table 7: MediaLibrary business rules.

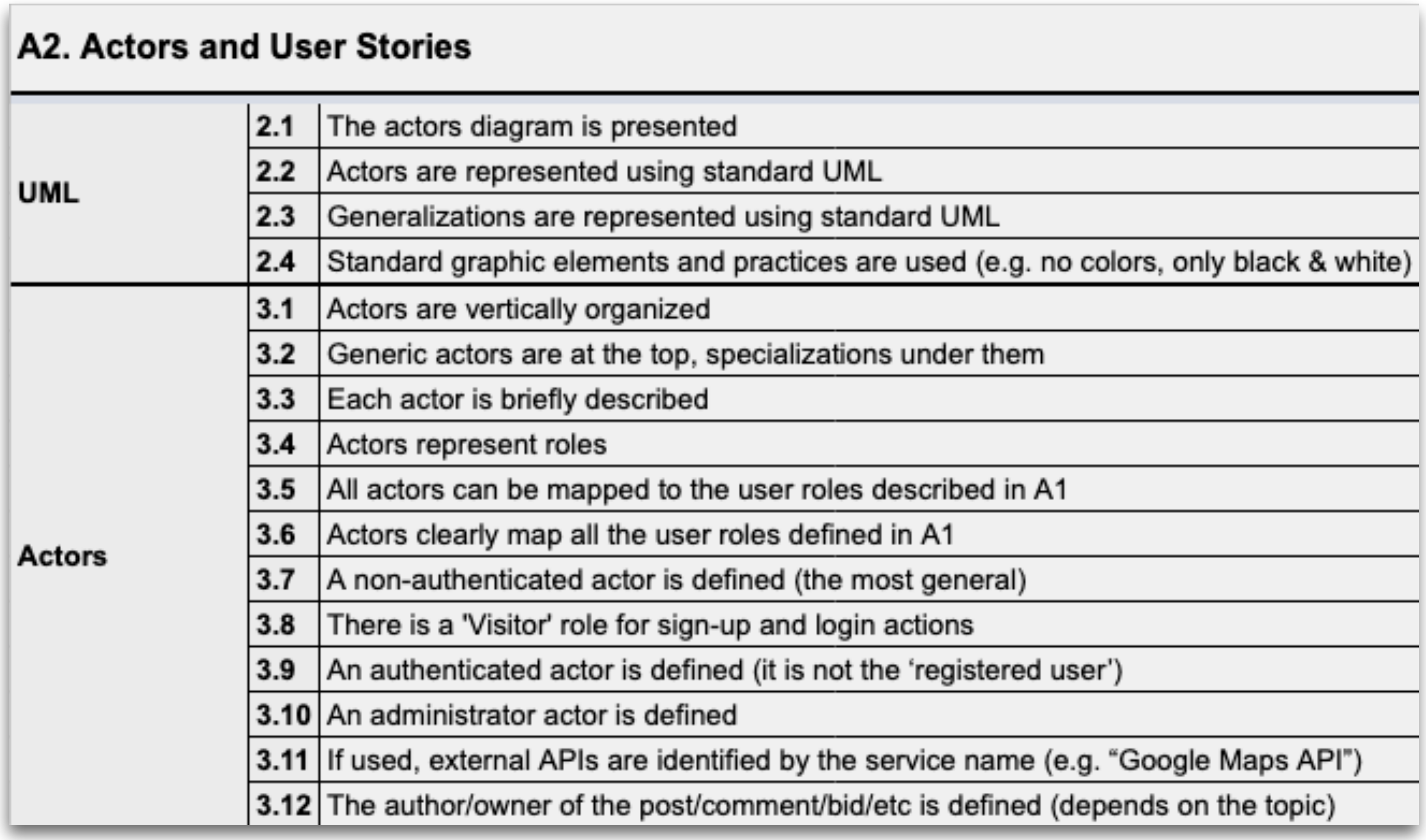
Technical Requirements

- Highlight and justify the three most critical technical requirements for your project.

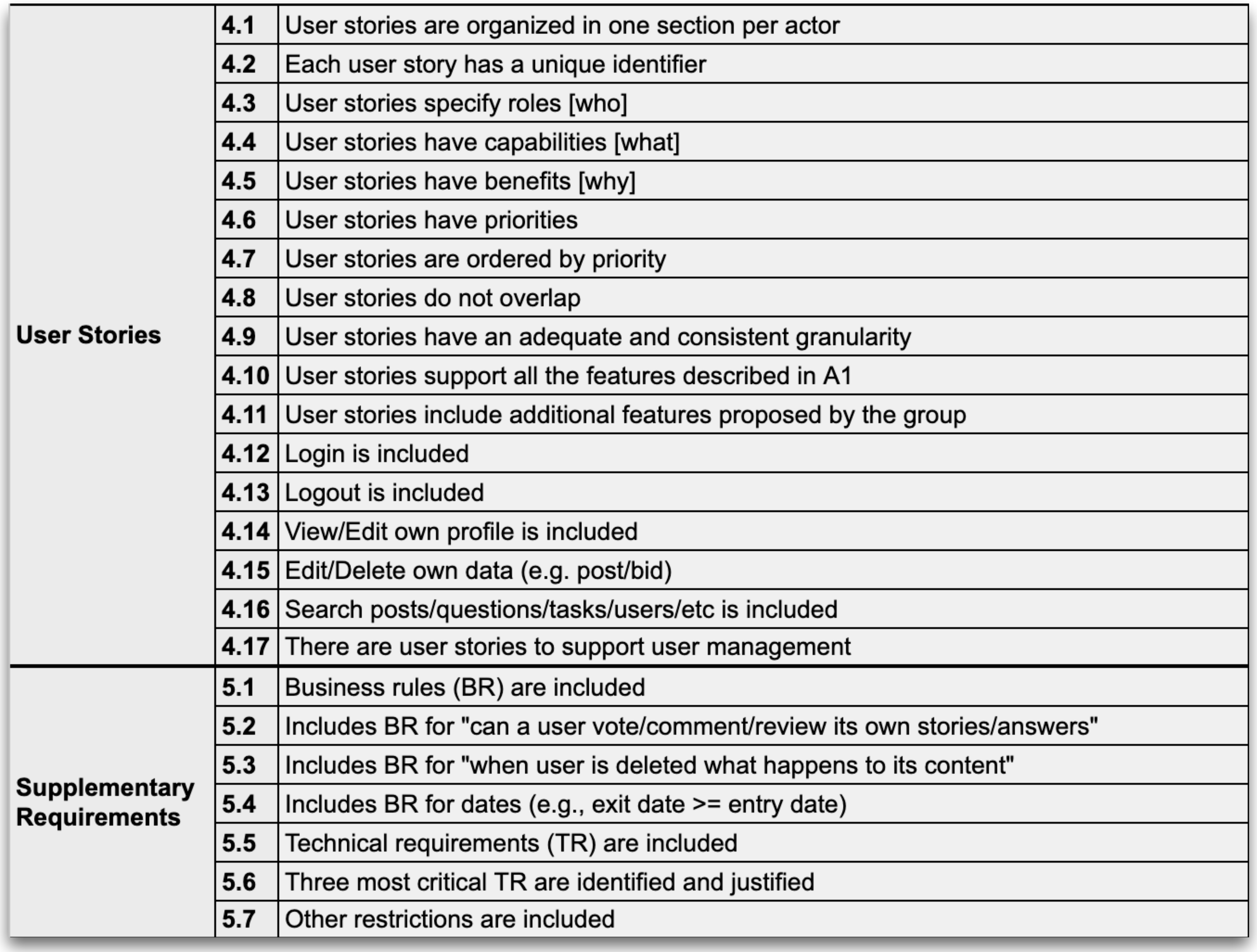


Identifier	Name	Description
TR01	Availability	The system must be available 99 percent of the time in each 24-hour period
TR02	Accessibility	The system must ensure that everyone can access the pages, regardless of whether they have any handicap or not, or the web browser they use
TR03	**Usability**	**The system should be simple and easy to use.** **The MediaLibrary system is designed to be used by media consumers from all ages, with or without technical experience, so a very good usability is a critical requirement.**
TR04	Performance	The system should have response times shorter than 2 s to ensure the user's attention
TR05	**Web application**	**The system should be implemented as a web application with dynamic pages (HTML5, JavaScript, CSS3 and PHP).** **It is critical that the MediaLibrary system is easily accessible from anywhere without the need to install specific applications or software, adopting standard web technologies.**

A2. Checklist



A2. Actors and User Stories		
UML	2.1	The actors diagram is presented
	2.2	Actors are represented using standard UML
	2.3	Generalizations are represented using standard UML
	2.4	Standard graphic elements and practices are used (e.g. no colors, only black & white)
Actors	3.1	Actors are vertically organized
	3.2	Generic actors are at the top, specializations under them
	3.3	Each actor is briefly described
	3.4	Actors represent roles
	3.5	All actors can be mapped to the user roles described in A1
	3.6	Actors clearly map all the user roles defined in A1
	3.7	A non-authenticated actor is defined (the most general)
	3.8	There is a 'Visitor' role for sign-up and login actions
	3.9	An authenticated actor is defined (it is not the 'registered user')
	3.10	An administrator actor is defined
	3.11	If used, external APIs are identified by the service name (e.g. "Google Maps API")
	3.12	The author/owner of the post/comment/bid/etc is defined (depends on the topic)



User Stories	4.1	User stories are organized in one section per actor
	4.2	Each user story has a unique identifier
	4.3	User stories specify roles [who]
	4.4	User stories have capabilities [what]
	4.5	User stories have benefits [why]
	4.6	User stories have priorities
	4.7	User stories are ordered by priority
	4.8	User stories do not overlap
	4.9	User stories have an adequate and consistent granularity
	4.10	User stories support all the features described in A1
	4.11	User stories include additional features proposed by the group
	4.12	Login is included
	4.13	Logout is included
	4.14	View/Edit own profile is included
	4.15	Edit/Delete own data (e.g. post/bid)
	4.16	Search posts/questions/tasks/users/etc is included
	4.17	There are user stories to support user management
Supplementary Requirements	5.1	Business rules (BR) are included
	5.2	Includes BR for "can a user vote/comment/review its own stories/answers"
	5.3	Includes BR for "when user is deleted what happens to its content"
	5.4	Includes BR for dates (e.g., exit date >= entry date)
	5.5	Technical requirements (TR) are included
	5.6	Three most critical TR are identified and justified
	5.7	Other restrictions are included

References

- The Object Primer - Agile Model-Driven Development with UML 2.0
  Scott Ambler. Cambridge University Press, 3rd Edition, 2004

- Management Information Systems - Managing the Digital Firm
  Kenneth C. Laudon, Jane P. Laudon. Pearson, 15th Edition, 2017