A System with an ...

Description

The objective of this project is to take advantage of emerging and existing ……. technologies to develop a system that is targeted towards …

To achieve these aims, this project will be implemented on or with …

The system will provide . . .

. . . as well as providing interactivity and social aspects to enhance their experience.

The project will choose an appropriate architecture . . .

Requirements Engineering

Feasibility Study

1. Are there similar systems to this in the marketplace (in Desk top, Cloud, Android, iOS, Windows …)?

   Identify at least 4 examples of Similar Systems.

   Describe each of them, by gathering information from their websites, "*put directly copied material in quotes and italics*". The more information gathered here will reduce the time required to go back to the websites. Reference the URL for each.

2. Identify the main system features and services provided in the reviewed systems, above. Consider the existing systems and the services they provide. [Reference and copy the URLs]

3. In what ways would users have accomplished the activity (get/use information), when not using a system, an app or online services?

4. Describe a new type of system, The proposed system
   Explain in detail how it might operate for different end users.
   Consider the existing systems that provide similar services to different end users and system adminstrators.
   Take inspiration from the systems identified in section 1 and key features identified in section 2.

5. Who are the stakeholders? How would this new system affect them positively or negatively?

6. What other research would be necessary to ascertain feasibility, Market Research information for marktet size e.g., ownership of smartphones …? (Gartner Research etc.)

7. Make an initial list of **functional** and **non-functional** requirements.

Requirements Elicitation

8. Could observation of exisitng processes or behaviours (Ethnography) be used in this case study? If so, in what way?

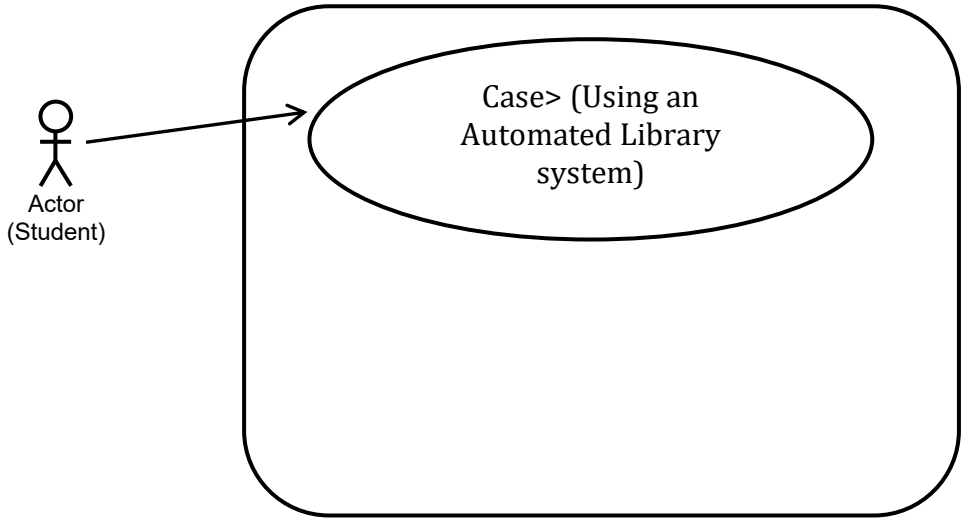
9. Identify a significant stakeholder(s), which will be **interview**ed to get more information on the intended product.
   Justify your choice of stakeholder(s).
   Do up an interview plan and pre-prepare approximately 10 questions.

10. Identify a significant group of stakeholders, which will receive **questionnaires**.
    Justify your choice of stakeholders.
    The questionnaire should have approximately 10 questions.

Requirements Analysis

11. Use the use case template to analyse the proposed system

Draw an initial *use-case diagram* with supporting scenario description for this app (possibly using *StarUML* for the diagram).
The first iteration of the use-case diagram can consist of a single overall use case with supporting main flow and 2 or 3 alternative flows.



System Name
(Automated Library System)

The use case description is developed from analysing the description of the use case. This is the statement of the goal of the use case.

For the first iteration this will be a description of the how the system operates.
Use Cases focus on functional requirements and specific system behaviour.

USE CASE	<number>	<Name of Use Case> <the name is the goal as a short active verb phrase>
Description of Goal in Context	<a brief statement of the goal for this Use Case in context > <Where, when, by Whom>	
Preconditions	<what we expect is already the state of the system> <list>	
Post Conditions, Success End Condition	<the state of the system upon successful completion>	
DESCRIPTION of Scenario	< The long description of the scenario to be analysed. The use case description is a more complete description identifying behaviour, it comes from the requirements gathering>	

Main Flow		
Step	**Action**	**Alternate**
n.1	<put here the steps of the scenario from trigger to goal delivery, and any clean-up after>	
n.2	**<…>**	
n.3	**<…>**	
n.4	**<…>**	
n.5	**<…>**	
n.6	**<…>**	
EXCEPTIONS or ERROR Flow **Description**		
Step	**Branching Action** < Exception number m of Use Case n>	**Alternate**
n.m.1	< condition causing exception> <Action, steps of scenario to goal delivery>	
ALTERNATIVE or VARIATION Flow **Description** <condition causing alternative> <list of variation>		
Step	**Branching Action**	**Alternate**
n.m.1	<Action, steps of scenario to goal delivery>	
n.m.2	<Action, steps of scenario to goal delivery>	

Non-functional Requirements for each use Case can be added in the Table below. Only consider 2 or 3 non functional requiremnts from the list in the table.

Non-functional requirements, management issues and decisions required to be made, can be identified in the following table.

From the table below, **choose <u>a limited number</u> of appropriate non-functional requirements** relevant to the Use Case.

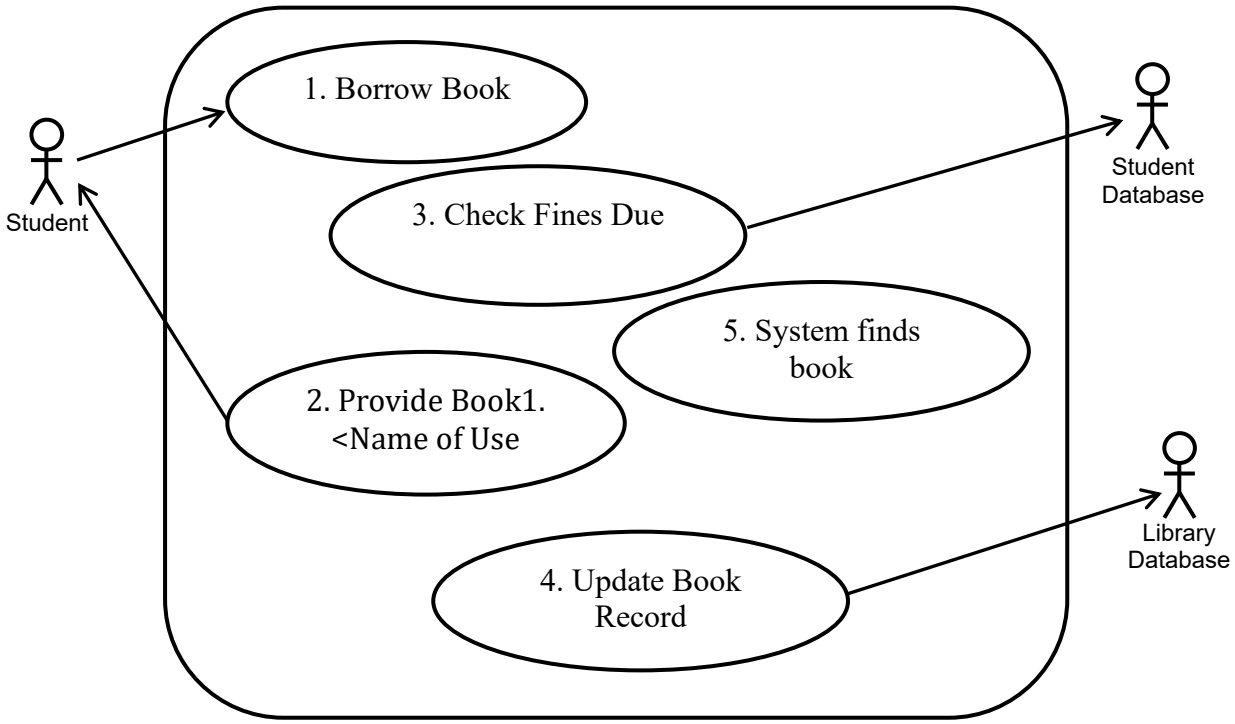
Non Functional Requirements can be categorised as

- Product related
- Organisation related, process and approaches set by the company
- External, imposed by outside bodies

RELATED INFORMATION	Use Case: 1	Using the system to borrow Book
Priority:	<how critical to your system/organization>	
Product: Performance	<Process and memory capacity, throughput, response time>	
Product: Efficiency	<level of memory usage, processor usage etc.>	
Product: Reliability	<likelihood of correct operation over a period of time>	
Organisation: Standards	<company standards for development, documentation etc.>	
Organisation: Delivery	<approach to deploying systems to users>	
External: Legislation	<Privacy, data protection, data retention rules, safety etc.>	
External: Ethical	<appropriate usage: is it appropriate for the target user>	
Frequency	<how often it is expected to happen>	
Channels to actors	<e.g. interactive, static files, database, timeouts>	
OPEN ISSUES	<list of issues awaiting decision affecting this use case>	
Due Date	<date or release needed>	
…any other management information…	<as…needed>	

More Systems Analysis

12. Develop a second iteration in a separate word report consisting of 4 or 5 use cases.
    Each use case requires a use case narrative describing the scenario analysis.
    Each use case should have 2 or 3 exception or alternative flows



Requirements Specification Matrix

13. From the requirements analysis identified with the use Case analysis identify key functional requirements.

    There should be 6 to 10 easily identifiable feature or requirements that can be listed in this matrix.
    All the features (that will become use cases) identified previously need to be included, review section 7 for the additional features

Req ID	Name of Req	Description	Priority	User Contact

System Modeling

From the systems analysis and the requirements table, identify additional features and actors.
Normally this would be done through additonal Use Case models (diagrams and narratives).
At this stage the aim is just to list what would be needed to complete the model in a list by reviewing the requirements table and the systems analysis models.

14. List all the potential *actors* in this system.

15. List the potential *use-cases* in this system.

Validation and Verification of Requirements

16. Test Case Planning.

    Develop test cases for the main use cases, the abstracted **high priority** functional requirements, identified in Iteration 2 of the Use Case Analysis.

    Develop at test case for the **most important** non-functional requirements for the highest priority Use Cases.

    Use the test case template to create initial Use Acceptance Test plans that will permit users and developers to agree the system will have been developed as specified by the requirements

    Consider the test plan as a user guide or user manual for non-technical novice users of the system

Test Case Number: 1
Test Case Name: Verify Borrow Book
Related Use Case Name: Borrow Book Number: **1**
Purpose: **Confirm the main flow for Borrow book**
Procedure Steps: (Guided by Main flow or other flows of Use case) 1. Tester will present valid identification 2. System indicates borrower is authenticated 3. Tester enters a valid book name, that is available 4. System indicates book is available 5. Tester asks for book 6. . . . 7. . . . 8. 9. 10. 11. 12. 13.
Expected Results: **All steps worked as expected for the main flow**

Completing the Feasibility Study

Review Previous Versions

17. Before you complete the final submission of a feasibility report, review and update the non-functional requirements, if necessary.

18. Review and, if necessary, Update the requirements Specification Matrix, section 13, and identify the high level core system features

Update Requirement Specification (RS) & set of Use-Case Diagrams (UCD) with narratives

19. Consider the Use case Model to ensure that core/key functionality has been addressed in the analysis and modelling process.

    Add comments here on what might need to be done to address any omissions or corrections.

    a. Do any of your use-cases need to be broken down further i.e., is there is too much functionality in one use-case?
    b. Update the potential Use Case list in section 15 as necessary.
    c. Update the Requirement Specification table with additional requirements as necessary.

Prototype

20. Create an initial prototype of the proposed system.

    Such as: sketch the home page/starting page of the system. Then take a photo of it and insert the photo into the document.

Functional & Non-Functional Test-Cases

21. Write additional test-cases (using the test-case template) for each of three abstracted **high priority** *functional* requirements (one test-case per requirement/use case).

22. Write additional test-cases (using the test-case template) for each of the two **most important** *non-functional* requirements (one test-case per requirement).