

Comic Stock

Online Comic Book Store

*Functional Requirements Specification*



PRIVATE AND CONFIDENTIAL:

This specification contains information, which is the property of the Entelect Group and Entelect Software (Pty) Ltd. No part of this specification may be reproduced, transmitted or divulged to third parties in any form or by any means, without written permission from the owner.

Date: 5 January 2021

Author(s): Avi Jivan, Riaan du Toit

Version: 0-1

Contents

[1. Distribution List 3](#_Toc60745443)

[2. Version History 3](#_Toc60745444)

[3. User experience research report 3](#_Toc60745445)

[4. Executive Summary 4](#_Toc60745446)

[5. Scope 5](#_Toc60745447)

[5.1 In Scope 5](#_Toc60745448)

[5.2 Out of Scope 5](#_Toc60745449)

[6. Assumptions 5](#_Toc60745450)

[7. Risks 6](#_Toc60745451)

[8. Dependencies 6](#_Toc60745452)

[9. Introduction & background 7](#_Toc60745453)

[9.1 Business Context 7](#_Toc60745454)

[9.2 System Context 7](#_Toc60745455)

[9.3 Process Overview 8](#_Toc60745456)

[10. Feature <xy> 9](#_Toc60745457)

[10.1 As a user, I want to <xy> 9](#_Toc60745458)

[10.1.1 User Interface and acceptance criteria 9](#_Toc60745459)

[10.1.2 Flowchart and system logic etc. 11](#_Toc60745460)

[10.1.3 Data storage requirements 11](#_Toc60745461)

[10.2 As a user, I want to <xy> 12](#_Toc60745462)

[11. Feature <xy> 13](#_Toc60745463)

[12. Contribution Log 14](#_Toc60745464)

[13. General tips for a professional specification 14](#_Toc60745465)

# Distribution List

Table 1: Distribution List

|  |  |  |
| --- | --- | --- |
| Name | Position | Email Address |
| <Name> | <Position> | <Email address> |
| <Name> | <Position> | <Email address> |

# Version History

Table 2: Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author(s) | Version | Changes |
| <Date> | <Author> | <Version> | <Description of changes> |
| <Date> | <Author> | <Version> | <Description of changes> |

# User experience research report

Attach the User Experience (UX) report here.

Here is the template to be used: <PowerPoint template for research report attached here>

# Executive Summary

The executive summary must provide the result/outcome of the specification. Include important information that would allow a reader to quickly determine the effect of this specification. Think of it as the “objective” and “end result” that we want to achieve. Executive summaries will analyze a problem, drawn conclusions, and recommend a course of action in a complete but brief synopsis.

It must be readable as a standalone piece of work; therefore, acronyms must be defined, and figures and tables within the specification must not be cross-referenced.

Be cognisant of the audiences reading this section – for example, product owners, CEOs, and project managers.

# Scope

This section must define the features in scope for this specification as well as the features considered but deemed out of scope.

## In Scope

1. Scope 1
2. Scope 2
3. ….

## Out of Scope

1. Out of Scope 1 – Give a reason if appropriate
2. Out of Scope 2
3. ….

# Assumptions

Not all the requirements you gather will be 100% clear and you may need to make some assumptions in order to design a solution. An assumption is what you believe to be true. These are anticipated events or circumstances that are expected during your project’s life cycle. E.g. “We assume the system will cater for credit card payments”. List the assumptions you make to reach your conclusions. Remember to validate them after you have done the specification.

1. Assumption 1
2. Assumption 2
3. …

# Risks

State all risks or concerns that may affect the delivery of the project. Also state mitigations if available.

Table 3: Risks

|  |  |  |
| --- | --- | --- |
| Number | Risk | Mitigation |
| 1 | This is the first Risk | This is the action taken to mitigate the effects of the risk |
| 2 | This is the second Risk | This is the action taken to mitigate the effects of the risk |

# Dependencies

What are the things you rely on to make the solution work? Do you need other systems to be ready or some component or system to function before this solution can work? Are there any dependencies on the business process to make the design work? Dependency is a broad software engineering term used to refer when a piece of software relies on another one

A dependency is something you can manage, that is, if you act you can resolve the dependency and ensure the feature is a success. If there is nothing you can do about it, then it is a fact or a constraint, not a dependency. E.g. “The application front end cannot be built until the data is made available” or “We cannot start development until the front end has been designed”.

# Introduction & background

Introduce the problem being solved and the solution specified. Provide the context to the business domain and background of the project.

Note that this section is aimed for the implementation team and needs to provide sufficient context to the problem being solved.

## Business Context

Provide business context. List business rules relevant to the problem statement and the specified solution.

Table 4: Business Rules

|  |  |  |
| --- | --- | --- |
| Number | Business Rule | User Story/Feature |
| 1 | This rule governs the behaviour of the business | <Cross-reference> |
| 2 | This rule also governs the behaviour of the business | <Cross-reference> |

## System Context

Explain the system context accompanied by a context diagram.

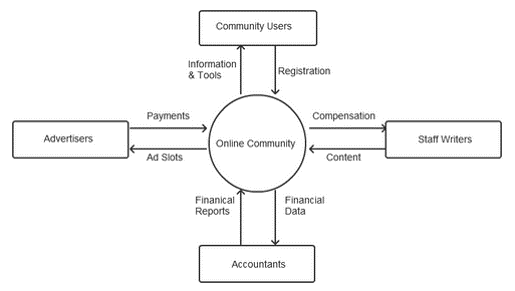


Figure 1: Context Diagram

## Process Overview

Explain the process flow accompanied by a process flow diagram.

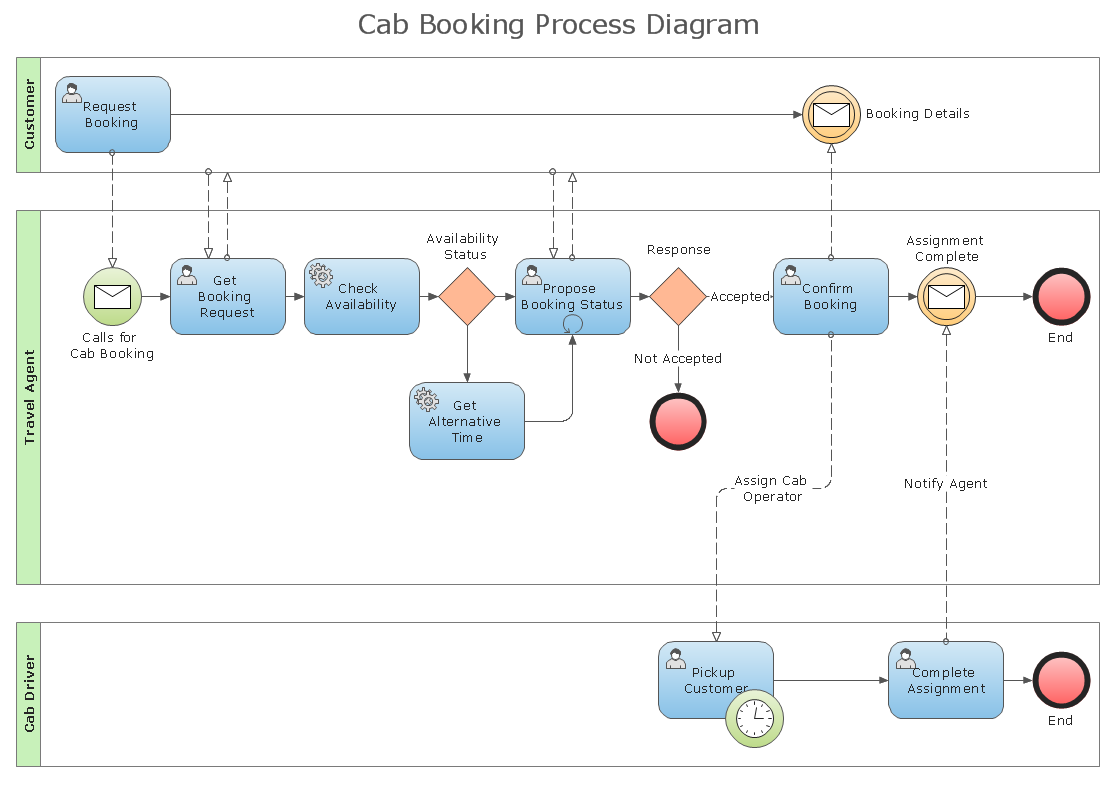


Figure 2: Process flow diagram for Process X

# Feature <xy>

State the feature being documented. Break up the given feature into multiple user stories.

Context (if needed) for this feature.

## As a user, I want to <xy>

### User Interface and acceptance criteria

Explain the user journey using screen designs for a mobile application using mock-ups. Include key designs as part of this main document.

Also include a link to the prototype.

#### Wireframe for “Step 1” in the user journey

What is happening in the wireframe illustrated in Figure 3?

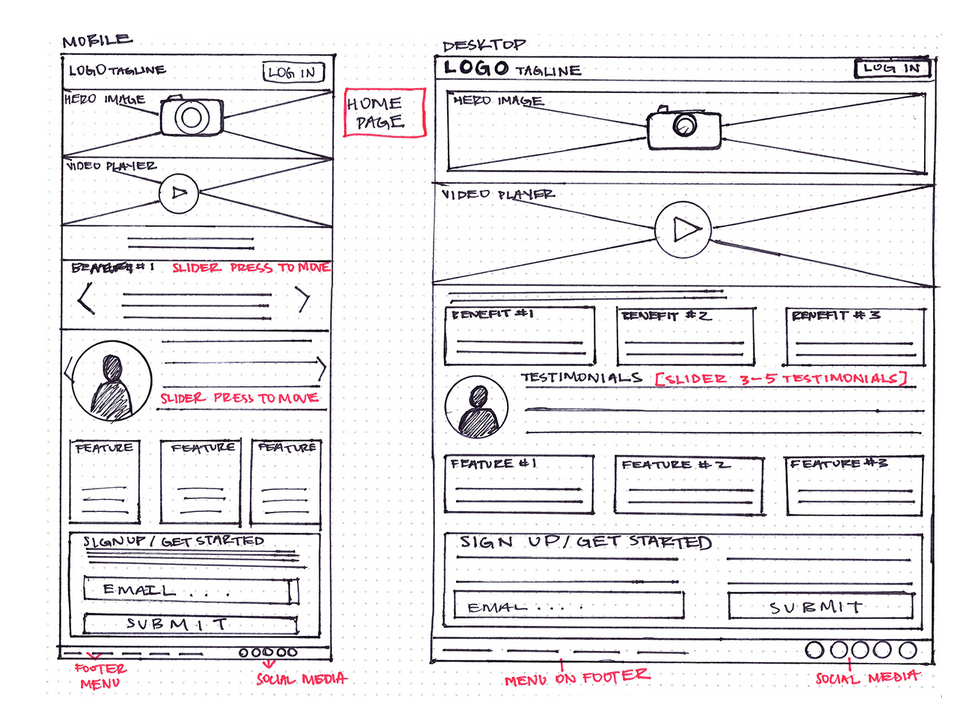


Figure 3: Wireframe 1

#### Acceptance Criteria for Wireframe “Step 1”

You can write a paragraph or summarize it in a table. A table may make for easier referencing when testing.

Table 5: Acceptance Criteria for Wireframe 1

| Content | Acceptance Criteria |
| --- | --- |
| Photo of User | Image of logged-in user should be displayed |
| Hero Image | Cycle through images for current month’s specials |
| Video Player | Play a 5 second intro as loop to entice users to click on it |
| Actions | Acceptance Criteria |
| Log In | Route to Login Page |
|  | Execute Login Use Case |
| Submit | Execute Signup Use Case |
|  | Display confirmation message |
|  | Route to Page “ABC” |
| etc |  |
| UI Controls | Acceptance Criteria |
| Email | Email input box validation on regular expression for email addresses |
|  | 1000-character input limit |
| Select Gender | Radio buttons |
|  | One option must be selected |
| Error Handling | Acceptance Criteria |
| Submit Fails | Display user friendly message “Could not register, please try again later” |
| Login Page load fails | Display user friendly message “Oops, something went wrong, please try again later” |

### Flowchart and system logic etc.

Provide a visual representation (using a process flow) and narrative of how the user interacts with the system in the context of this user story. This process flow must tie together the screens, with the implementation of business rules and requirements and data storage requirements.

### Data storage requirements

Define the data storage requirements, if any. Note that this section may not be required if data must not be stored.

Data storage functional requirements are independent of their implementation. For example, if it were a requirement to store a record of all comic books in a warehouse, then Table 6 may be one way of specifying such a requirement.

Table 6: Storing comic books available for purchase

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ISBN | Title | Author(s) | Condition | Price |
| 978-3-16-148410-0 | My title | John Doe, Jane Smith | New | R 24.99 |
| 978-3-16-513610-0 | Another title | Jane Smith | New | R 50,00 |
| 978-3-16-873210-1 | A title of a book | Ted Someone, Jane Smith | Used | R 9,58 |

Note that it does not specify the design of the database. Therefore, concepts like (primary) keys and joins are not specified.

Sometimes, additional information (like the maximum length of a name allowed when a user creates an account) may need to be specified. This may need to be specified in this section or it could be specified, for example, in the system logic section. This will depend on the context for which the requirement is being written.

## As a user, I want to <xy>

Repeat the concepts from section 10.1 for each user story that is part of this feature.

# Feature <xy>

Repeat the concepts from section 10 for each feature that is part of this specification.

# Contribution Log

|  |  |  |
| --- | --- | --- |
| Date | Name | Change(s) |
| 06/01/2020 | Gertjie Dysel | I did the use cases because UML is easy |
| 07/01/2020 | Frikkie Mossel | I did the wireframes because I like art |
| 07/01/2020 | Francel De Milliers | I did the editing and business rules because I am the BA in the group… |

# General tips for a professional specification

1. Use cross-references and captions for tables and figures in the specification. Tables have captions above, and figures have captions below
2. Repeat table headers across pages, and do not allow a single row to split across pages
3. Define acronyms before using them. Note that acronyms must be redefined in the executive summary since it can be considered a standalone document
4. Be consistent. Use the tools provided by Word to apply formatting throughout the document. Avoid applying styling to individual elements since it is a maintenance nightmare when making changes
5. Write in the third person
6. Strive to keep your specification as short as possible since stakeholders are busy. A longer specification is not more impressive
7. Consider the audience of the specification when writing. Some stakeholders may be less technical than others. Generally, avoid pseudocode
8. Read sentences out loud to check that they flow well
9. Justify paragraph text since it looks more professional than ragged edges
10. Include the version number of the specification in the file name so that it need not be opened just to check
11. Refresh the table of contents, figure and table numbers, and cross-references regularly (CTR A, then F9 by default). Always make sure it is up to date before sending out the specification
12. Use the “show/hide” functionality (in the “home” tab of Word) to clean up unnecessary blank spaces. Remember to hide these before sending out the specification
13. Read through the headings while viewing the navigation pane (within the “view” tab in Word). Check that they are sufficiently descriptive of the sections they represent and that there aren’t any untidy blank sections
14. Always review spelling and grammar. You are a professional now