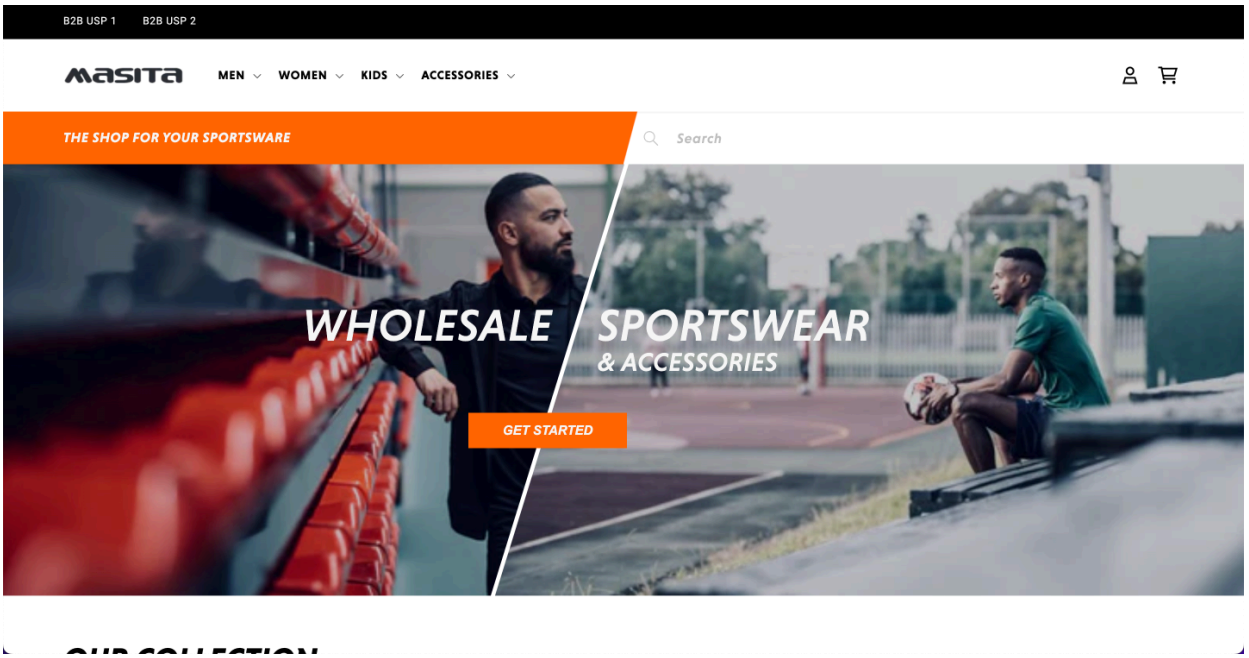
Project plan

Graduation internship project

Masita



Date	**:**	**28-02-2024**
Version	**:**	**1.1**
Status	**:**	**Definitive**
Author	**:**	**Luc Swinkels**

Version

Version	Date	Author(s)	Amendments	Status
0.1	26-02-2024	Luc Swinkels	First draft	Draft
1.0	27-02-2024	Luc Swinkels	First definitive version	Definitive
1.1	28-02-2024	Luc Swinkels	Processed feedback v1 Robin	Definitive

Table of contents

1. Project Assignment

1.1 Context

Masita, a leading Dutch sportswear brand, had a Magento webshop developed a few years ago. However, it turned out that there were a lot of different requirements and desires that were built as extensions to Magento by various developers, including Moonly. Additionally, Masita worked with other parties for tasks such as order picking and shipping, the ability to customise items, and selling products on other channels (such as bol.com, etc.). To integrate these software systems with the Magento webshop, a number of connections were written.

Due to insufficient consideration for future-proofing, scalability, and the Magento framework as a basis when creating these integrations, they did not function properly, leading to more and more cracks in the system. Whenever an external integration was updated, the entire order process behind the store would come to a halt.

Ultimately, Masita chose to cease operations, and the webshop was taken offline due to escalating problems with the website. Currently, these integrations no longer work, and they have not been updated for over a year.

Last year, Masita was acquired by the current owner, and they are in the process of completely redesigning their webshops. Subsequently, Moonly was asked to start developing a B2B store so that they could begin selling items to businesses.

Now that Moonly is working on completing this B2B store, we want to think ahead and commission a study on the development of a brand new B2C store.

The current problem analysis is as follows: Masita is currently unable to sell products to B2C customers because all integrations are disabled on the current website (masita.com, which is often inaccessible), and it is not possible to order products. Additionally, this website, having not been updated for over a year, often experiences downtime and has low speed. Since the Magento store is currently unusable, and Masita does want to sell products again, we need to explore how a new B2C store can be realised.

1.2 Goal of the project

Design challenge

Design a new e-commerce platform that enables Masita, a leading Dutch sportswear brand, to sell products to individual customers (B2C) in a seamless and reliable manner, considering past integration issues and system failures.

What problem am I trying to solve?

The problem revolves around Masita's dysfunctional Magento webshop, which stopped working due to complex integration requests from Masita that weren't possible within Magento (i.e. external plugins that weren't being updated).

This caused Masita to shut down its operations and take the webshop offline. Now, Masita wants to restart sales but needs a new, reliable B2C store to replace the old one, which can handle integrations better, offer a better user experience, and ensure smooth online sales to individual customers.

Desired situation
The desired situation is for Masita to have a new, functional B2C webshop that drives new users to make purchases. The webshop should be technologically solid and able to handle integrations from other sales channels (like bol.com), as well as possible future expansions.

1.3 The assignment

Summary
The assignment is to research which technologies can and should be used for the project, create user-friendly designs that are optimised for UX and conversion, validate these designs with user tests, then develop these validated designs into a working prototype by using the earlier researched technologies.

Full assignment
Moonly and Masita have decided to collaborate on a new B2B and B2C store so that Masita can resume selling their products. Since it was important for Masita to start selling to businesses first, a B2B store is currently in development. In the future, a B2C store will also need to be developed.

For the B2C store, Masita has provided several requirements. There must be the ability to sell printed shirts, customers must be able to manage orders, and it should be possible to sell products through external stores.

Since Moonly wants to avoid the same pitfalls and problems encountered with the Magento store when developing this new store, we want to first conduct research into the best solution for developing this new store. This research should not only consider Masita's various requirements but also focus on how the B2C shop can encourage customers to start ordering from Masita again. While the B2B shop primarily focuses on functionality for retailers, the B2C shop needs to be optimised for UX and conversion to help Masita regain its position in the market.

Additionally, the results of this research are intended to be translated into a proof of concept/prototype of a B2C store so that Moonly can develop a new B2C shop with a validated foundation.

The research will be conducted with Moonly as the client and has been initiated by Moonly in preparation for developing the B2C shop. Masita is not the client in this case (though they may be a stakeholder).

Functional requirements
- The webshop should be able to have integrations for other channels such as bol.com.
- The webshop should have a custom “kit building” integration where users can create custom products that they can purchase.
- The webshop should be available to users through an online website.
- The webshop should be available on multiple device types (mobile & desktop).
- E-commerce fundamentals (purchasing a product) should be available and working.
- Users should be able to create an account so that the data can be used for analytics and other marketing purposes (as well as other features like easier returns).

Non-functional requirements
- The webshop should be user-friendly and optimised for conversion.
- The webshop should be SEO friendly and technologically solid (fast loading speeds).

1.4 Scope

The project includes:	The project does not include:
Designing wireframes	Developing the webshop database infrastructure (back-end)
Setting up user tests	Researching which hosting solution should be used to host the website (and hosting the project itself)
Creating a low-fidelity prototype	
Developing the webshop storefront (front-end) (high-fidelity prototype)	
Research on available webshop technologies	
Research on UX & conversion optimization	
Research on Masita's technical requirements (channel integrations and custom kit builder)	
Research on Magento issues from previous webshop	
Advisory report	

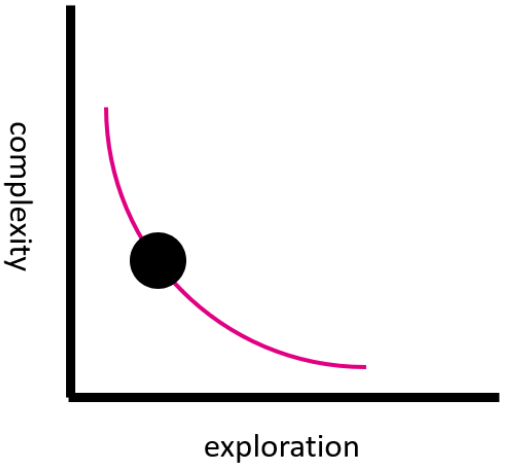
1.5 Conditions

- The designs should be made in Figma, so that they are readily available for free and adhere to modern design/prototyping standards.
- The code should be stored in a repository in the Moonly Software GitHub organisation.
- Code commits (Git) should adhere to conventional commit standards (see here).
- Documentation should be written / made available in Google docs.
    - Useful information should also be documented internally in the Moonly Notion knowledge base, so that it can be referred to later for future projects.

1.6 Finished products

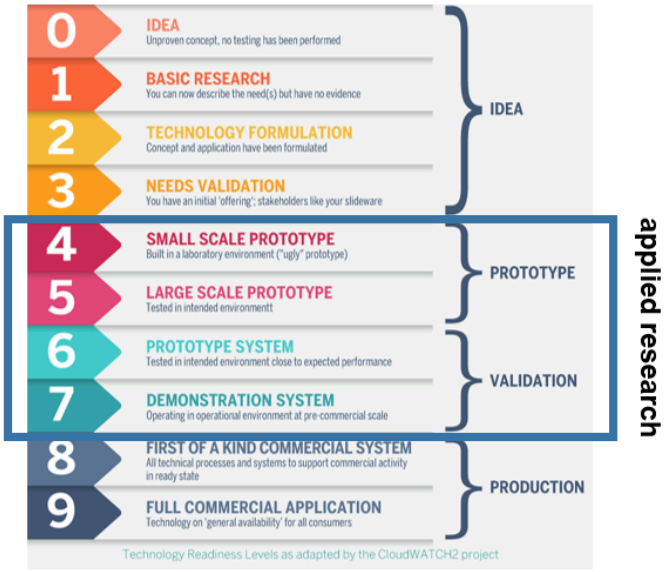
- High-fidelity prototype (live webshop)
    - Testing document - User testing high-fidelity prototype (live webshop)
    - Research document - Magento issues
    - Research document - Available webshop technologies
    - Research document - Masita's technical requirements (channel integrations and custom kit builder)
    - Low-fidelity prototype (figma design)
        - Testing document - User testing low-fidelity prototype (figma design)
        - Wireframes (figma design)
        - Research document - UX & Conversion optimization
- Advisory report

Complexity-exploration matrix



On the complexity-exploration matrix the project is fairly balanced, with a small focus going towards complexity as most of the techniques that could be used are complex but well-documented (possible front-end solutions like Shopify, React), however since there are requirements for custom integrations like connections with channels like bol.com and a custom kit builder, there is lots of room for exploration and experimentation.

Technology readiness levels



The technology readiness level of the desired end product is 6. It is possible to make a push towards 7, but since e-commerce platforms have lots of small unexpected functionalities to take into account, it is smart to start off the MVP as a 6.

1.7 Research questions

Design challenge

- Design a new e-commerce platform that enables Masita, a leading Dutch sportswear brand, to sell products to individual customers (B2C) in a seamless and reliable manner, considering past integration issues and system failures.

Main question:

- How can a new, user-friendly B2C webshop be built and validated by research?

Sub questions:

- Which problems did the old magento-based B2C-webshop face and how can I avoid them?
- Which technologies can and should be used to develop the new webshop?
- How can Masita's technical requirements be integrated?
- How can I make sure the new webshop is user-friendly and optimised for conversion?

2. Approach and Planning

2.1 Approach

For this project, I will be using agile (scrum) methodologies by working in 2-week sprints, with each sprint having a retrospective. I am also logging all my tasks in Monday and Jira per sprint. Since this is an individual project, I will not be having stand-ups, but I will partake in the company-wide stand-up every Monday to inform my colleagues of how the project is progressing.

Test approach
Testing will be done by performing relevant tests that will be defined later on the devices mentioned in the test environment chapter (see here).

2.2 Research methods

- Which problems did the old magento-based B2C-webshop face and how can I avoid them?
    - Expert interview
        - I will conduct an expert interview with Bruno Delsing, a developer who worked on the previous magento-based webshop to figure out what the problems were.
    - Root cause analysis
        - By looking at what could have caused these problems thoroughly, I can help determine which root causes to avoid.
- Which technologies can and should be used to develop the new webshop?
    - Available product analysis
        - By analysing which technologies are available to me and comparing their features, pros and cons, I can evaluate what the best choice is for this project.
    - Expert interview
        - I will conduct an expert interview with Robin Galema, who has years of experience with building e-commerce platforms.
- How can Masita's technical requirements be integrated?
    - Available product analysis
        - Since there is no active knowledge within Moonly of how to integrate Masita's wishes, I will be conducting available product analysis research to see which options I can use to achieve this and how.
    - Literature study
        - I will look at online resources, blogs, developer forums to find information about things like custom product builders and channel integrations for e-commerce platforms (Masita's requirements) to gain insight in this.
- How can I make sure the new webshop is user-friendly and optimised for conversion?
    - Usability testing
        - I will set up usability tests with a range of users to test my designs for UX and conversion by seeing how they walk through the design.
    - Expert interview
        - I will conduct an expert interview with Rik Gommans, one of Moonly's in-house developers with an eye for design and UX to see how I can ensure a quality UX.
    - Prototyping
        - I will build a prototype that can be tested for usability.
    - Design pattern research

- I will look at popular design patterns used by direct competitors and similar companies, to determine how to drive conversion rate up.

2.3 Breakdown of the project / time plan

Orientation / problem definition

Sprint 0 - week 2 + 3 (Week 9 & 10 - 26 februari t/m 8 maart)

- Write project plan
- Set up scrum board with tasks and sprints

Analysis / research

Sprint 1 - week 4 + 5 (Week 11 & 12 - 11 maart t/m 22 maart)

- Research magento issues
- Research technical requirements Masita (channel integrations and custom kit builder)

Sprint 2 - week 6 + 7 (Week 13 & 14 - 25 maart t/m 5 april)

- Research available webshop technologies
- Research UX & conversion optimization
- Start wireframe design

Design

Sprint 3 - week 8 + 9 (Week 15 & 16 - 8 april t/m 19 april)

- Design low-fidelity prototype (figma)
- Set up user testing plan for low-fidelity prototype
- Test prototype
- Make adjustments based on test results

Development

Sprint 4 - week 10 + 11 (Week 17 & 18 - 22 april t/m 2 mei)

- Set-up MoSCoW model for high-fidelity prototype features
- Start development high-fidelity prototype (must-haves)

Sprint 5 - week 12 + 13 (Week 19 & 20 - 6 mei t/m 17 mei)

- Continue development (must-haves)

Sprint 6 - week 14 + 15 (Week 21 & 22 - 20 mei t/m 31 mei)

- Continue development (should-haves)
- Test must-haves

Sprint 7 - week 16 + 17 (week 23 & 24- 3 juni t/m 14 juni)

- Finalise development (should-haves / could-haves)

Sprint 8 - week 18 + 19 (week 25 & 26 - 17 juni t/m 28 juni)

- Document transfer
- Work through backlog from previous sprints if needed

This breakdown does not include my portfolio and other tasks relevant to FHICT. In sprint 0, I will be setting up my portfolio and by sprint 1 it should be ready for me to add content to it. At the end of every sprint, I will take some time to update the content of the portfolio. I plan to have my portfolio fully finished by the end of sprint 7 (week 16-17).

In the last sprint, I will also be working on preparing my end presentation for the internship.

Each sprint will also include a retrospective to reflect on the process.

3. Project Organization

3.1 Team members

Name + e-mail	Role/tasks	Availability
Luc Swinkels luc@moonlysoftware.com	Student / designer / developer	5 days a week (in person / online)
Robin Galema robin@moonlysoftware.com	Company supervisor / development supervisor / product owner (available for design & development feedback)	5 days a week (in person / online)
Luuk Derkx l.derkx@fontys.nl	Teacher supervisor	5 days a week (in person / online)
Rik Gommans rik@moonlysoftware.com	Front-end developer/designer (available for design feedback)	5 days a week (in person / online)
Ryan Smith ryan@moonlysoftware.com	Front-end developer (available for fallback advice when Robin is not available)	5 days a week (in person / online)
Arie Kegel (Masita) -	Stakeholder / contact (not directly involved with this project yet, Robin is product owner)	-

3.2 Communication

Communication with company supervisor

Every 2 weeks I have a meeting (Monday, 2PM) with my company supervisor to discuss general internship and project progress and other possible issues and questions I need help with. For any urgent questions, I can contact my company supervisor directly as we are both available 5 days a week, either in person or through online channels such as Slack.

Communication with teacher supervisor

Every week I have a meeting (Friday, 3PM) with my teacher supervisor to discuss general internship and project progress. If me and my teacher supervisor feel like weekly meetings are too often, we can reschedule these to be bi-weekly instead. For urgent questions or concerns I can contact my teacher supervisor on online channels like Microsoft Teams or e-mail.

Communication with stakeholders

While it may look like Masita is the stakeholder, they are currently ***not yet*** involved in the project.

Masita gave this project pitch to Moonly and Moonly has decided it would be best suited as my internship project, instead of directly taking it on as an active project. This means that I will not be directly reporting or communicating with Masita, but more so with Robin Galema as product owner. When the project is finished, Moonly will evaluate what is needed to complete the project for full-scale launch and hold contacts with Masita accordingly once my internship is completed.

3.3 Test environment

CI/CD

Ideally, the code for this project would be stored in a repository on the existing Moonly Software GitHub organisation, which has already been integrated with automated CI/CD tasks that deploy to an online environment when new code is pushed to the repository. This environment is set up with a production and development environment, so both can be tested on separate domains.

However, since it is currently unknown which technologies are going to be used, it is better to evaluate which hosting platform is best suited for this when these technologies are known.

Testing

Testing will be conducted on the previously mentioned development environment.

As this code is accessible to any device with a working browser and internet connection, I will be running tests on the following devices:

- MacOS Laptop (my personal Macbook Pro, M1 2020 13")
    - The storefront will be tested on multiple browsers:
        - Google Chrome, Safari, Firefox
    - The storefront will be tested on multiple resolutions:
        - Macbook (2560×1600), FHD (1920×1080), QHD (2560×1440)
- Windows laptop (Moonly testing device, unknown)
    - Using the Microsoft Edge browser
- iOS phone (my personal iPhone 15 Pro)
    - Using the Google Chrome and Safari browser
- Android Phone (Moonly testing device, Google Pixel 6)
    - Using the Google Chrome browser

When all tests have passed on all devices and browsers, code can be sent to production upon review from my company supervisor by using pull requests.

3.4 Configuration management

Git

As previously mentioned, code will be hosted within the Moonly Software GitHub organisation in a separate repository. This repository will have the following branching structure:

- **Master** (production, protected and can only be pushed to after an approved pull request by my company supervisor)
- **Development** (default) (development, where the last round of testing will happen before being merged into production)
- **[JRA-444] feature-name** (for specific features, these will be merged into development after being completed and tested)

Jira

Jira will be used to keep track of development progress. To assist with this, I will be using Jira's built-in Git integration by naming my feature branches after a user story in Jira like so: **jra-1-as-a-user-i-want-to-purchase-a-product**. With these integrations I can easily track what feature a branch belongs to.

Conventional commits

To maintain code quality and reduce confusion in large codebases, I will be following the conventional commit guidelines when pushing code to the Git repository. With these guidelines, every commit message will have a predefined type, subject, and style as seen here.

4. Finance and Risks

4.1 Cost budget

There are no known software or hardware costs tied to the project. There is a possibility that certain plugins for the chosen e-commerce platform could cost money, however since it is still unknown which platform is being used and which plugins would be needed, it is not possible to have a predefined budget or costs.

If there are costs associated with necessary software, they will be evaluated on a case to case basis.

The hardware solutions that are needed to realise the project (hosting and testing machines) are already in place and therefore don't need to be included in the budget.

4.2 Risks and fall-back activities

Risk	**Prevention activities included in plan**	**Fall-back Activities**
Laptop breaks	Documentation stored on cloud-services (Google Docs), and code stored on GitHub	Use temporary device from Moonly or personal computer
Robin Galema (company supervisor) absent/ill	None	Contact Ryan Smith for front-end help if needed
Student absent/ill	None	None (evaluate with student if needed)
Office unavailable to work in	None	Work from home (remote)
Luuk Derkx (teacher supervisor) absent/ill	None	None (evaluate with Fontys if needed)