



Collaborative Network for Industry, Manufacturing, Business and Logistics in Europe



D3.7

User Front-End Prototyping for Fast System Adoption

Project Acronym	NIMBLE	
Project Title	Collaboration Network for Industry, Manufacturing, Business and Logistics in Europe	
Project Number	723810	
Work Package	WP3	Core Business Services
Lead Beneficiary	SRFG	
Editor	Oliver Jung	SRFG
Reviewers	Wernher Behrendt	SRFG
Dissemination Level	PU	
Contractual Delivery Date	31/03/2018	
Actual Delivery Date	31/03/2018	
Version	V0.3	

Abstract

This document describes the overall methods applied and tools used for the User Front-End Prototyping during the implementation phase of the NIMBLE platform. Therefore, various suitable approaches will be compared and results will be outlined as implemented for Release 2. Furthermore, an outlook on upcoming features and changes for the following releases will be given. The focus is on enabling small and medium enterprises to move from a traditional web presence to a cloud-aware, IoT-enabled business collaboration ecosystem of European proportions.

NIMBLE in a Nutshell

NIMBLE is the collaboration Network for Industry, Manufacturing, Business and Logistics in Europe. It will develop the infrastructure for a cloud-based, Industry 4.0, Internet-of-Things-enabled B2B platform on which European manufacturing firms can register, publish machine-readable catalogues for products and services, search for suitable supply chain partners, negotiate contracts and supply logistics. Participating companies can establish private and secure B2B and M2M information exchange channels to optimise business work flows. The infrastructure will be developed as open source software under an Apache-type, permissive license. The governance model is a federation of platforms for multi-sided trade, with mandatory interoperation functions and optional added-value business functions that can be provided by third parties. This will foster the growth of a net-centric business ecosystem for sustainable innovation and fair competition as envisaged by the Digital Agenda 2020. Prospective NIMBLE providers can take the open source infrastructure and bundle it with sectorial, regional or functional added value services and launch a new platform in the federation. The project started in October 2016 and will last for 36 months.

Document History

Version	Date	Comments
V0.1	20/03/2018	Initial skeleton version
V0.2	05/04/2018	Added initial versions for all chapters
V0.3	09/05/2018	Draft version ready for review
V1.0	31/03/2018	Finalized

Table of Contents

Abstract	2
NIMBLE in a Nutshell.....	2
Document History	3
Table of Contents	4
List of Figures	5
Acronyms.....	6
1 Introduction	7
2 Base Technology for the UI	8
3 Vital tools and libraries.....	9
4 Core UI Elements and Features.....	10
4.1 Supporting interfaces.....	14
5 Upcoming Features and Changes.....	16
6 Conclusion	17

List of Figures

Figure 1 – Catalogue publishing	11
Figure 2 – Catalogue management	11
Figure 3 – Support button	12
Figure 4 – Dashboard	12
Figure 5 – Explorative search	13
Figure 6 – Text-based search.....	13
Figure 7 – User registration.....	14
Figure 8 – Company member management	14
Figure 9 – Service health check.....	15
Figure 10 – Platform usage dashboard	15
Figure 11 – Issue tracker	15

Acronyms

Table 1: Acronyms table

Acronym	Meaning
HCI	Human-Computer Interaction
NIMBLE	Collaboration Network for Industry, Manufacturing, Business and Logistics in Europe
UI	User Interface
UX	User Experience

1 Introduction

This document includes a technological overview of tools used for realizing the UI throughout the implementation tasks as well as functionalities employed with regard to the iterative User-Centered Design and evaluation tasks.

D3.7 is strongly linked to the User Experience Design for Fast System Adoption (T2.4) which is focused on the analysis and conception/design phase creating a methodological foundation for the choices made and approaches taken in D3.7.

Since the task T3.7 carries on beyond the submission of this document and the UI will be improved and enhanced throughout WP5 (Value-added Business Services for NIMBLE), this document will focus on the state of Release 2 (2018-03-15) and outline the major possible upcoming features and changes for the following releases.

2 Base Technology for the UI

In general, the NIMBLE UI was intended to be developed using a single framework in order to maintain a central codebase and allow developers to easily integrate and improve their modules.

As a result, Angular¹ was chosen as the framework for implementing the UI because it is one of the most commonly used open-source frameworks for web development, is actively developed and improved by Google since 2010 and offers various important features out of the box, such as:

- Deployment for web, mobile web, native mobile and native desktop from a single codebase
- High performance and scalability using RxJS² data models
- Compatibility with various major integrated development environments (IDEs)
- Component-based code splitting
- Data binding
- Unit test support by Karma³

Angular is based on TypeScript⁴ – a superset of JavaScript⁵ that compiles to plain JavaScript. The main advantages of using TypeScript are enhanced IDE support, type inference and the support of modern ECMAScript⁶ standards that get compiled down to older JavaScript versions depending on browser support.

While there are plenty of similar frameworks (e.g. React⁷, Meteor⁸) Angular seemed to offer the best integration with other development tools, cover the biggest part of the tool chain and part of the development team was already familiar with it.

¹ <https://angular.io>

² <http://reactivex.io>

³ <https://karma-runner.github.io/2.0/index.html>

⁴ <https://www.typescriptlang.org>

⁵ <https://www.javascript.com>

⁶ <https://www.ecma-international.org/ecma-262/7.0/index.html>

⁷ <https://reactjs.org>

⁸ <https://www.meteor.com>

3 Vital tools and libraries

In the following chapter all vital tool and libraries used during the UI development will be listed and briefly summarized.

Bootstrap

The most important library in use is Bootstrap⁹. Bootstrap is a responsive, mobile-first toolkit for arranging and styling UIs that offers various components. For proper integration with Angular ng-bootstrap¹⁰ was included into the project. The most important features of Bootstrap are:

- Pre-defined, adjustable stylesheets and color schemes taking visual impairments (e.g. Deuteranopia) into account
- Grid-based layouts with responsive breakpoints adjusting the UI to the device resolution
- Ready-made components, such as alerts, buttons, dropdowns, forms, progress bars, tooltips, etc.

Karma

As mentioned before Angular offers unit test support by Karma out of the box. Karma allows for automated testing on real devices, remote control of the whole workflow via command line or IDE, debugging via IDE and integration with continuous integration tools, such as Jenkins¹¹.

Selenium

In addition to Karma Selenium¹² is used for testing and automation. Selenium ultimately is a browser automation tool for defining interactions and workflows. This allows for detecting various errors that are not visible on the code level, e.g. inaccessible UI elements. Furthermore, Selenium can be used to generate workflows for data generation and demonstration purposes.

⁹ <https://getbootstrap.com>

¹⁰ <https://ng-bootstrap.github.io>

¹¹ <https://jenkins.io>

¹² <https://docs.seleniumhq.org>

4 Core UI Elements and Features

This chapter contains all Angular modules in use with brief explanations regarding the corresponding functionality and UIs.

Business Process Execution

Angular module name:

bpe

Functionality:

Execute business processes (e.g. order, negotiation) by communicating with the Camunda¹³ API.

Interfaces:

The bpe module currently has no UI but is used by other modules (e.g. simple-search). An interface for customizing business processes was integrated at an early stage but is currently on hold.

Catalogue

Angular module name:

catalogue

Functionality:

Upload and maintain product (and service) catalogues.

Interfaces:

¹³ <http://camunda.com>

Figure 1 – Catalogue publishing

A catalogue can be publishing in the “Publish” section using single product input forms or uploading multiple product Excel files.

Figure 2 – Catalogue management

A catalogue can be maintained in the “Catalogue” section by using the edit/delete functionalities and search filters.

Common

Angular module name:
common

Functionality:

Offer functions that shall be available and uniform across all modules (e.g. error handling, internationalization, feedback).

Interfaces:



Figure 3 – Support button

A globally available support button for reporting issues to the technical support team.

Dashboard

Angular module name:

dashboard

Functionality:

Provide an overview of all relevant information and processes to the user (aka. landing page).

Interfaces:

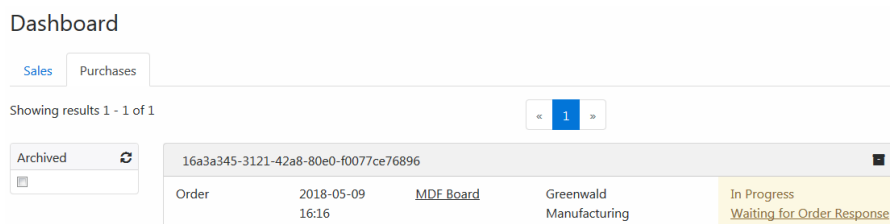


Figure 4 – Dashboard

Provides a manageable overview of ongoing business processes.

Explorative (Semantic) Search

Angular module name:

explorative-search

Functionality:

Search for products with certain characteristics and features based on the underlying taxonomies.

Interfaces:

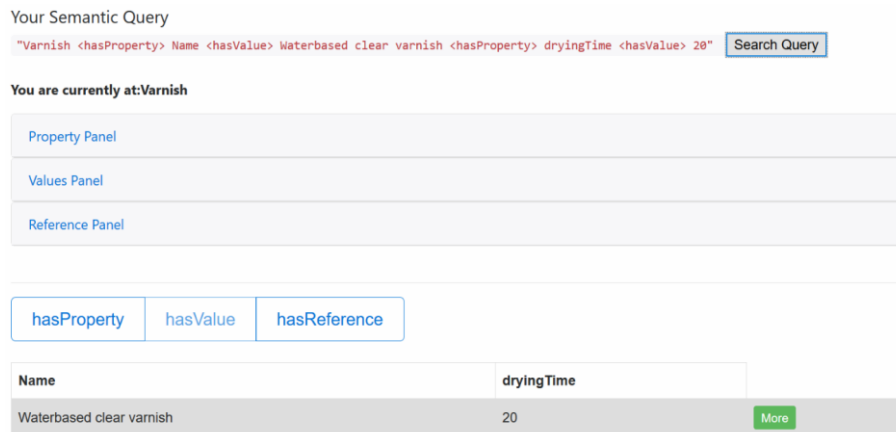


Figure 5 – Explorative search

Allows for using a simplified semantic search interface filtering products by taxonomy category, properties, values and references based on SPARQL¹⁴.

Simple (Text-based) Search

Angular module name:
simple-search

Functionality:
Provide a classic text-based faceted search interface.

Interfaces:

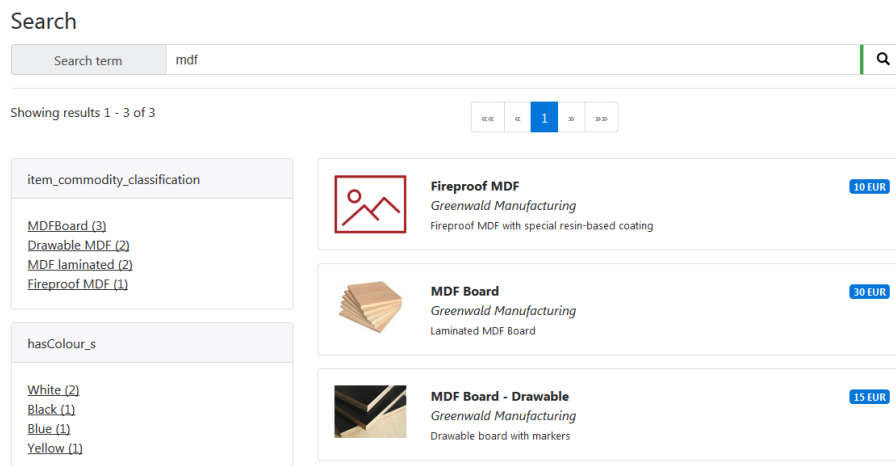


Figure 6 – Text-based search

A text-based search interface that allows for applying filters extracted from existing facets based on a Solr¹⁵ index.

User Management

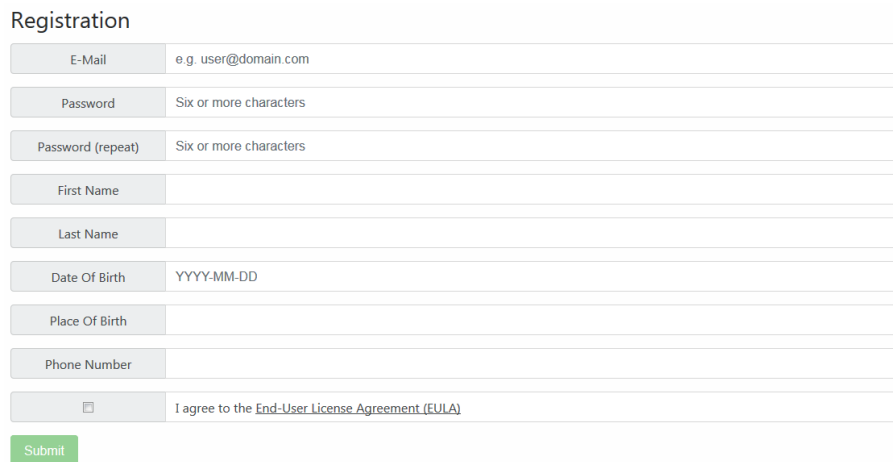
¹⁴ <https://www.w3.org/TR/sparql11-query>

¹⁵ <https://lucene.apache.org/solr>

Angular module name:
user-mgmt

Functionality:
Provide interfaces and tools for enabling proper authentication of users and management of user and company data.

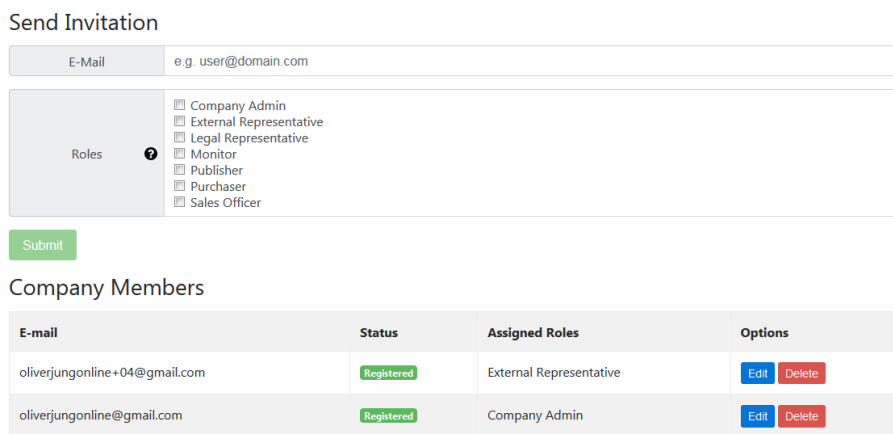
Interfaces:



The registration form includes the following fields: E-Mail (with placeholder 'e.g. user@domain.com'), Password (with hint 'Six or more characters'), Password (repeat) (with hint 'Six or more characters'), First Name, Last Name, Date Of Birth (with placeholder 'YYYY-MM-DD'), Place Of Birth, and Phone Number. At the bottom, there is a checkbox for 'I agree to the End-User License Agreement (EULA)' and a green 'Submit' button.

Figure 7 – User registration

Enables a user to register on the platform.



The 'Send Invitation' form has an E-Mail field (placeholder 'e.g. user@domain.com') and a 'Roles' section with a list of roles: Company Admin, External Representative, Legal Representative, Monitor, Publisher, Purchaser, and Sales Officer. Each role has a checkbox. A green 'Submit' button is at the bottom.

The 'Company Members' table is as follows:

E-mail	Status	Assigned Roles	Options
oliverjungonline+04@gmail.com	Registered	External Representative	Edit Delete
oliverjungonline@gmail.com	Registered	Company Admin	Edit Delete

Figure 8 – Company member management

Allows the legal representative (or members with management rights) of a company to manage the roles and rights of company members on the platform.

4.1 Supporting interfaces

In addition to the UI presented to the user there are some external supporting interfaces set up for platform managers and administrators to monitor and maintain platform activities.

Platform Services		
Service Name	Last Success	Status
Eureka (CATALOG-SEARCH-SERVICE)	Never	BAD
Eureka (FRONTEND-SERVICE)	09/May/2018 13:50:21 (UTC)	GOOD
Eureka (BUSINESS-PROCESS-SERVICE)	09/May/2018 13:50:21 (UTC)	GOOD
Eureka (ERROR-NO-SERVICE (to show missing service))	Never	BAD
Eureka (CATALOGUE-SERVICE-SRDC)	09/May/2018 13:50:21 (UTC)	GOOD
Eureka (IDENTITY-SERVICE)	09/May/2018 13:50:21 (UTC)	GOOD
Eureka (GATEWAY-PROXY)	09/May/2018 13:50:21 (UTC)	GOOD
Data-Channels-Service	Never	BAD <input type="button" value="Restart"/>
ERROR-SERVICE (to show failure)	Never	BAD <input type="button" value="Restart"/>
Nimble Eureka	09/May/2018 13:50:21 (UTC)	GOOD
Messaging-Service	Never	BAD <input type="button" value="Restart"/>
Registration Service	09/May/2018 13:50:21 (UTC)	GOOD

Infrastructure Services		
Service Name	Last Success	Status
Postgres-Database	09/May/2018 13:50:15 (UTC)	GOOD
Message Hub Nimble	09/May/2018 13:50:15 (UTC)	GOOD
Object Store Nimble	09/May/2018 13:50:15 (UTC)	GOOD

Figure 9 – Service health check

This HTML-only UI allows for monitoring the health status of various platform and infrastructure services and provides quick restart buttons in case of failures.

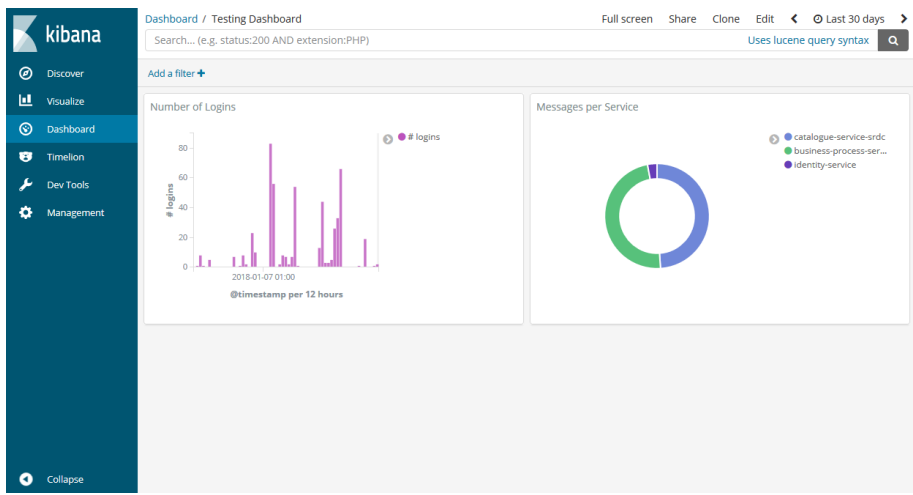


Figure 10 – Platform usage dashboard

This Kibana¹⁶-based UI shows some basic platform metrics (e.g. number of logins and messages sent per service) with different visualization modes.

nimble-platform Issue Tracker			
Overall			
Open issues (8)		Closed issues (16)	
business-process-service			
Open issues (2)		Closed issues (1)	
Creation	Assignees	Labels	Milestone
9.2.2018 by n/a		affects user interface enhancement user experience	Release 3.0 (1.6.2018)
9.2.2018 by n/a		affects user interface new feature user experience workshop bremen	Release 3.0 (1.6.2018)

Figure 11 – Issue tracker

This HTML-only UI aggregates all open and closed issues for all repositories on Github.

¹⁶ <https://www.elastic.co/products/kibana>

5 Upcoming Features and Changes

There are close to 50 issues assigned to Release 3.0 (2018-06-01). Apart from bug fixes and deployment changes they mostly target the following features and change requests:

- Unifying and improving of the UIs (layouts, icons, views). At a later stage there might be an additional framework, such as Material¹⁷, included for more components and improved mobile support
- Improvement of the business process execution (thread-based, guided representations instead of a step-by-step approach)
- Introduction of data integration and communication services
- Introduction of a structured contracting functionality that combines company terms, business processes and data channels
- Make all labels human readable and provide pre-defined options where feasible

¹⁷ <https://material.angular.io>

6 Conclusion

As stated in D2.4 we are following a user-centered iterative design process. This naturally results in new feature requests by users after each release. Those feature requests are then aligned with the overall platform, prioritized and converted into issues for the next release. This approach will be followed throughout WP3.

WP5 (Value-added Business Services) on the other hand will address issues outside of the core functionalities and provide UIs or other defined interfaces for integrating those value-added services within the central UI. The simplest solution might provide a redirect link to an external UI using single sign-on (SSO) while a more sophisticated approach might provide means of dynamically building “NIMBLE-native” UIs via configuration files and APIs.

Finally, further supporting interfaces will be provided for platform managers in order to analyze the platform usage and adapt to user needs.