

Bolzano/Bozen, 25th June 2024

## Market research – SFSCON – SFSCON App

This market research aims to identify one or more partners that can support NOI S.p.A. to update the SFSCON App according to the use cases and feedback emerged after the SFSCON 2023. The SFSCON App is the iOS and Android application dedicated to the South Tyrol Free Software Conference (SFSCON) participants.

**Project name:** IMPACT – Impacting Innovation Through Specialization

**Project code:** EFRE1048

**Project CUP:** J57H23000640009

**NOI AG / S.p.A.**

A.-Volta-Straße 13A  
Via A. Volta, 13A  
I-39100 Bozen / Bolzano  
T +39 0471 066 600  
info@noi.bz.it  
PEC: noi@pec.noi.bz.it  
www.noi.bz.it

Eintragung im Handelsregister  
der Handelskammer Bozen  
Steuernr. & MwSt.Nr.:  
02595720216  
Ges.kapital voll eingezahlt:  
110.740.000 €

Numero d'iscrizione nel registro  
delle imprese presso la Camera  
di commercio di Bolzano  
Codice fiscale e part.  
IVA.: 02595720216  
Capitale sociale interamente  
versato: 172.740.000 €

## Table of contents

Table of contents	2
1. Goal of the market research	4
1.1 Introduction	4
1.2 general architecture	4
1.3 Features	5
1.3.1 SFSCON Application	6
1.3.2 SFSCON APP backend	6
1.3.3 SFSCON dashboard	7
1.4 GIT repositories	7
2. Constraints	7
2.1 Economic exploitation	7
2.2 Invoicing	8
2.3 Work methodology	8
2.4 repository git	10
2.4.1 Documentation	10
2.4.2 Licensing and Reuse compliance	11
2.4.3 Pull request (PR)	11
2.4.4 Commits	12
2.4.5 Deployment	12
2.4.6 Testing	13
2.4.7 API development	13
2.4.8 Access Control List (ACL) management	13
2.4.9 Dockerization	14

2.5 Working place and hour	14
2.5.1 Working Hours	14
2.5.2 Working Place	14
3. Request to the supplier	14
3.1 Project information	14
3.3 Competences	15
4. Documentation	15
5. Deadlines and contacts	15

## 1. Goal of the market research

This chapter aims to explain in more detail about content included in the market research. The aim is to identify one or more partners that can support NOI S.p.A. to develop the SFSCON App according to the use cases and feedback emerged after the SFSCON 2023. The SFSCON App is the iOS and Android application dedicated to the South Tyrol Free Software Conference (SFSCON) participants.

### 1.1 INTRODUCTION

The Development of the SFSCON App started in 2020 with the goal of providing the participant a platform to see the conference's program, bookmark the talks they are interested to and share feedback about talks and speakers.

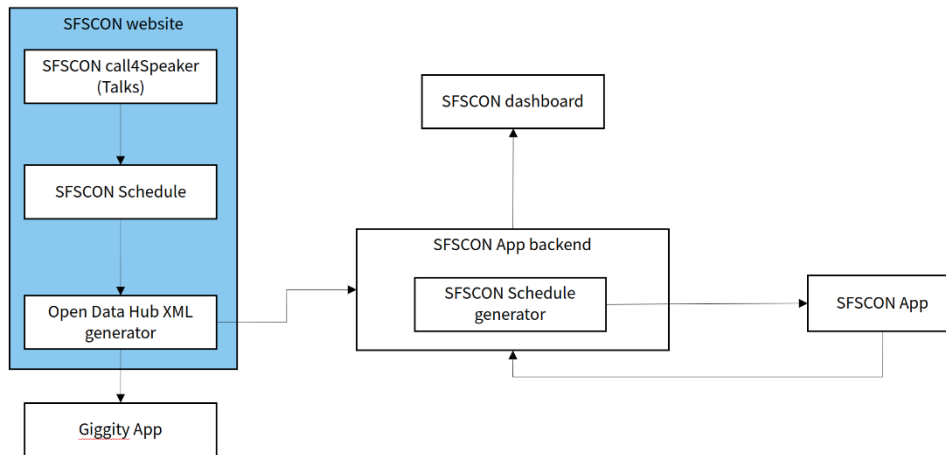
The goals of the SFSCON organization were:

- avoid printing the program;
- easily update the program in case of last minutes changes;
- provide more visibility to the conference by sharing the schedule through an app available in the Apple Store, Google Play Store and also through apps like Giggity (<https://play.google.com/store/apps/details?id=net.gaast.giggity&hl=it&gl=US&pli=1>);
- collect rating of the talks shared by the SFSCON attendees;
- let participants bookmark their favorite talks before the conference and let them be notified when the talk is about to start.

In 2021, 2022 and 2023 the SFSCON app has been used also to manage the entire conference's check-in process and to monitor the participants during the conference by scanning a QR code on the badge at the door entrance of each room. Considering that the SFSCON team doesn't have any more the need to monitor the distribution of participants during the whole conference, the evolution of the Ticketing Systems like Pretix (<https://pretix.eu/>) and the complexity of scanning a QR code at the door entrance of each room, the SFSCON team decided not to include those features in the new version of the app.

### 1.2 GENERAL ARCHITECTURE

Figure 1 shows the general architecture of the entire SFSCON app project including also the components that must be included in the new version of the app.



**Figure 1:** SFSCON App general architecture.

The SFSCON website allows to collect talks proposals through the Call4Speaker section and to generate the event Schedule in the Schedule section. Moreover, the SFSCON website exposes the schedule information in an XML file that other applications (e.g., Giggity, SFSCON App, etc.) can use to generate and render the SFSCON schedule.

The SFSCON backend will:

- retrieve the schedule information from the SFSCON website;
- expose the schedule information that the SFSCON App will consume;
- collect the talk ratings that the SFSCON attendees will input by using the SFSCON App. Please note that the feedback is collected only for statistical purpose and to know which are the speakers and topics that are mostly interesting for the SFSCON participants. For this reason, the collected data must be completely anonymous, therefore there is no need to store the user's data of each talk review/comment;
- expose the ratings data that the SFSCON dashboard will consume;
- collect (and expose) bookmarks that the SFSCON attendees will input in the SFSCON App.

The SFSCON App will take the information to render the SFSCON schedule from the SFSCON App backend and send the talk ratings and the bookmarks to the SFSCON backend and shown on the SFSCON dashboard.

The SFSCON dashboard is a web application that renders the statistics and comments about the SFSCON participants' talks reviews.

### 1.3 FEATURES

This paragraph will include a more detailed description of the functionalities that will be developed or removed in/from the new version of the SFSCON app and its components.

All software components developed within this project must be REUSE (<https://reuse.software/>) compliant.

### 1.3.1 SFSCON APPLICATION

All functionalities of the new version of the SFSCON app will be accessible without the need to login. The new version of the SFSCON App will implement the use cases described below.

- As SFSCON participant I want to search in the schedule section a speaker or a title of a specific talk to easily find the talk I want to bookmark;
- As SFSCON participant I want to bookmark the talks that I am interested in, to define my personal event schedule without the need to login into the app.
- As SFSCON participant I want to access general information (e.g., how to arrive, lunch, coffee break, etc.), to avoid contacting the SFSCON team for generic questions. The application should link the dedicated sections of the website like for example:
  - About: <https://www.sfscon.it/about/>
  - Venue: <https://www.sfscon.it/venue/>
  - Past Editions: <https://www.sfscon.it/programs-archive/>
  - Map: <https://nav.sfscon.it> (In development)
  - Etc.

The final list of pages that must be linked from the App will be defined during the kick-off meeting.

In case of schedule changes the SFSCON App should not delete the personal schedule of each participant. In case that a talk changes name, it should be changed also in all personal schedules that include that talk. In case of talk deletion, the talk must be removed also from all personal schedules that include that talk without touching all other bookmarked talks. In that case the user should receive also a notification of the talk deletion.

The SFSCON app must communicate the bookmarks and talk ratings to the SFSCON App backend in an anonymous way, without any participant information.

To avoid the need to have a login or input the order/ticket ID, the bookmarked talks must be stored on the mobile phone or browser storage.

### 1.3.2 SFSCON APP BACKEND

The SFSCON backend will get the SFSCON schedule information from the SFSCON Website. The SFSCON website will expose the schedule in an open format (frab XML file). The backend will then prepare the schedule information and expose it for the application.

The SFSCON App backend will also get from the SFSCON app the talks' ratings shared by the participants and prepare it to be consumed from the SFSCON dashboard. The

reviews will be collected by the backend in an anonymous way, without the need to do the check if a user sends more than one review.

### 1.3.3 SFSCON DASHBOARD

The SFSCON Dashboard is a web application that the SFSCON team will use to evaluate the participants ratings and export the data collected through the SFSCON app. The new version of the SFSCON dashboard will implement the use case described below;

- As a SFSCON organizer I want to see also the number of ratings of each talk, to better interpret the ranking of it.
- As SFSCON organizer I want to see a summary of the statistics (e.g., number of ratings received, ratings, etc.) about the single talks, to evaluate it and improve next year's schedule.
- As SFSCON organizer I want to see a summary of the statistics of the bookmark selected by the SFSCON participants, to evaluate which talks are the most interesting;
- As SFSCON organizer I want to export the raw data about the collected feedback in CSV format, to use the tools I am used to elaborate the data.

### 1.4 GIT REPOSITORIES

The SFSCON app is an open-source project and at the following links you can find the source code of each component included in the present market research:

- SFSCON app: <https://github.com/noi-techpark/it.sfscon.a.p.p>
- SFSCON backend: <https://github.com/noi-techpark/it.sfscon.app.backend>
- SFSCON dashboard: <https://github.com/noi-techpark/it.sfscon.app.webadmin>

## 2. Constraints

In this section are listed and described the constraints that the service provider must follow to work with NOI on this project.

### 2.1 ECONOMIC EXPLOITATION

Where the creation of material subject to proprietary rights, including copyrights, sui generis data rights, and related rights, including solely of photographs, industrial design, all rights of economic exploitation arising from achieved results are reserved to NOI S.p.A., excepting those expressly excluded when the order is placed.

Further, if the material includes a software development project, all source code from libraries or other modules used in the realization of an assignment and belonging to a third party must be released under an Open Source license ([opensource.org/licenses](https://opensource.org/licenses)) in a manner compatible with the scope of the "outbound" software license, without requirement for adaptation, addition, cancellation or requests for permission from

third parties on the part of NOI S.p.A. In the absence of any expressly indicated license, the terms of the GNU GPL v3 licence shall apply. The use of material belonging to third parties must be expressly declared at the time of the offer or be easily and immediately understandable from the description of the project. In the event that code is developed during the realization of this assignment, NOI S.p.A. will initiate a Git repository on which the supplier must develop and publish the source code.

If the material consists of data, creative works (drawings, literary works, cinematographic works, figurative art, photographs), industrial design or other material which are subject in whole or in part to the proprietary rights of a third party, the use of such material is permitted provided it is licensed under conditions compatible with the license under which said material will be published, if indicated. If no license is indicated, the material will be subject to conditions compatible with the Creative Commons Zero (CC0) license.

## **2.2 INVOICING**

The invoicing of the activities concluded by the supplier will be sent to NOI S.p.A via electronic invoice only after the outputs produced have been successfully tested by NOI S.p.A. Before to proceed with the testing of the outputs, the supplier must provide to NOI S.p.A.:

- the entire documentation.
- if code development is planned, the code must be uploaded to the Git repository provided by NOI S.p.A.
- in the case of multimedia contents (e.g., photos, videos, illustrations, documents), the service provider must upload it on specific platforms (e.g., Vimeo, Flickr, etc.) and provide the source files or open versions through appropriate file hosting services indicated by NOI S.p.A.

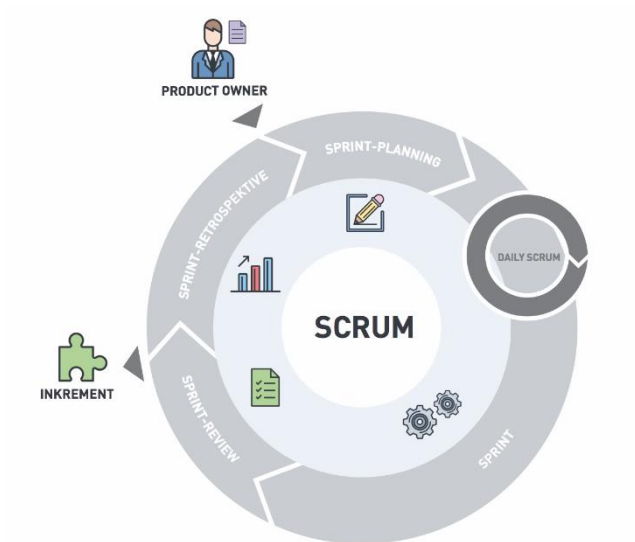
All invoices must include that the transaction is subject to the Split Payment discipline as mentioned in the art.17-ter del DPR 633/197 and must be issued exclusively in electronic format (Unique Office code: T04ZHR3).

## **2.3 WORK METHODOLOGY**

The SFSCON is an event organized by the Tech Transfer Digital of NOI S.P.A. Since the main project of the Tech Transfer Digital is the Open Data Hub, for the development of the SFSCON App the team will follow the same work methodology as in the Open Data hub Project. This paragraph will include more details about the work methodology.

The development of the activities covered by this market survey will follow the agile method (scrum). Two weeks sprint sessions are scheduled, unless otherwise agreed, during the kick-off meeting with the core team of NOI S.p.A.





**Figure 3:** the SCRUM methodology.

The software development will take place in three phases/environments:

- **development environment:** this environment is on supplier's infrastructure and is used during the development of the software components.
- **testing environment:** on infrastructure made available from NOI Techpark. This environment is used to test the new working versions of the software components. For the publication of the new versions a Continuous Integration (Jenkins) pipeline will be developed by the NOI team. For this reason, the new versions of the code will have to be "committed" to a dedicated Git Repository according to the instructions provided by the team of the NOI Techpark.
- **production environment:** on infrastructure made available from NOI Techpark. After the testing phase, as soon as the software produced is considered sufficiently stable, the software will be integrated into the production environment. Also, this process is managed automatically with Continuous Integration pipelines.

To coordinate the project NOI S.p.A. will use a Kanban Board in GitHub. Each functionality or issue will be described by NOI S.p.A. in GitHub and put on the Kanban Board. The Kanban Board will have the following columns:

- **Backlog:** contains all issues that are on hold and have to be discussed during the next sprint meeting with the supplier;
- **ToDo:** contains all issues that must be concluded in the actual sprint;
- **In Progress** contains all issues where that are working in progress.
- **To Review:** contains all issues that NOI Techpark has to review and that has to be reviewed during the sprint meeting.

All issues in the Kanban, apart from the one in Backlog, have to be assigned to the user that must make the next step (e.g., the issues in ToDo will be assigned to the developer who has to develop the functionality, the issue in To Review will be assigned to the tester, etc.). The supplier will have access to the project Kanban board and will have to check it regularly.

To allow the NOI S.p.A. team to properly review and test the code, for each issue in the ToDo lane the service provider has to send a pull request to the development Branch of the repository at least 5 working days before the sprint meeting.

To allow a better integration of the systems already in use by NOI Techpark it is required to implement all software components, where possible, using the technologies that are already in use by the Open Data Hub (<https://opendatahub.com/>) project.

## 2.4 REPOSITORY GIT

The source code must be uploaded to the Git repositories provided by NOI Techpark. During the upload, the service provider must take particular attention to the following aspects:

- do not commit usernames or passwords. NOI Techpark uses GitHub Actions to build the code which implements password ingestion based on special keywords in the source code;
- well document the code describing at least:
  - the general architecture of the system;
  - the list of the licenses of all the libraries used;
  - the installation process;
  - all other useful information for people who want to fork or install and use the project.

As Open Data Hub (<https://opendatahub.com/>) we have created some boilerplate repositories for the most common project type (es. Java project, Web Component, .Net Core project, etc.). In case you are starting a new project from scratch, before starting your project please look for the boilerplate that best fits your project and use it to initialize your repository.

### 2.4.1 Documentation

While you are documenting your code, please consider that the official language of the all project managed (like for example Open Data Hub and SFSCON) is English. So, the entire documentation, including the comments in the code, must be in English. Moreover, you must observe the following guidelines:

- use the right boilerplate of the README.md if exists;
- use only markdown or text (no binaries, no PDF, etc.);
- should be so detailed that a third person, without any connection to the developers can setup the project, run it and develop it further;

- Java Doc and similar tools for other languages should be as complete as possible;
- add the author tags incl. emails;
- README.md should be a good description of the project and should also have usage instructions (boilerplate does not consider that). Mainly because tools like **\*\*npm\*\*** use it as homepage for each project

In general, the documentation of the project (e.g., readme file, license file, etc.) should be done to allow third parties developers, who do not know anything about the project, to understand the whole project and replicate, install or modify it without the need to contact NOI S.p.A. Therefore, the documentation (README.md) should include also:

- a brief description that allows the user to understand the overall goal and functionalities of the project;
- longer and detailed description that includes also:
  - description of the distinct parts of the repository/application;
  - description of distinct parts of the project (also other repositories, if existing, and a link to them) and how this application is part of the overall project;
  - external services/code/framework/software that are used including their license and copyright information;
- detailed development setup instructions (including testing);
- detailed deployment setup instructions.

#### *2.4.2 Licensing and Reuse compliance*

In respect to the licensing and copyright information, the service provider must follow the guidelines defined by the Reuse project:

<https://reuse.software/>

The service provider must provide code where the Reuse linter passes without errors and the licenses must all be compatible with each other.

#### *2.4.3 Pull request (PR)*

As mentioned in the previous paragraphs the service provider, before each sprint meeting, will deliver the source code by making a Pull Request to the Development Branch of the repository Git provided by NOI S.p.A. at the beginning of the project. In general, the service provider must observe the following guidelines to make the pull requests:

- at the beginning of each sprint the service provider will open a Pull Request (PR) with a prefix [WIP] as Work in Progress;
- during the sprint, the service provider must regularly push the commits to that PR to allow NOI S.p.A. to monitor the status of the project;

- at the end of the sprint (at least 5 days before the sprint meeting) the service provider will close and send the Pull Request.

NOI S.p.A. will analyze the Pull Request before the meeting and eventually send feedback to the service provider. The minimal requirements for a Pull Request to get accepted are:

- the documentation must exist and be as complete as possible in respect to the status of the project;
- commits must not contain credentials or any other sensible data;
- contributions (e.g., documentation, comments, etc.) must be in English;
- merge conflicts must be resolved by the contributor;
- all Continuous Integration verifications must pass;
- Pull Request branches should have a linear history, that is, they should not contain merge commits.

During the development cycles the pull request comments and in general the issues and the dedicated Kanban board on GitHub (original repository) must be tracked by the service provider. The discussion about issues, pull requests, and other specific comments on the code development will be managed on GitHub in the project repository and NOT through email. That also involves moving user stories to the corresponding column in the Kanban and assigning them to the right user.

#### *2.4.4 Commits*

These paragraphs contain some guidelines that the service provider should follow while implementing the project:

- commits should contain a single thing/feature, not be too big and specially they should not be a combination of unrelated features or bug-fixes;
- each commit must be described: present tense and active (e.g., "Add logging to commons" not "commons will get logging now" and not "Added logging").

#### *2.4.5 Deployment*

For the deployment of the project NOI S.p.A. will use its CI/CD infrastructure, for this reason it is important that the service provider includes in the documentation of the project the information about how the application should be deployed or updated by a CD pipeline. Therefore, the documentation should point out the following things:

- What parameters must be configured? Which ones are secrets and which are not?
- What services must be used? (e.g., PostgreSQL database, S3, etc.)
- What steps must be made to package the application/project so that it can be copied to the server?
- What steps must be made on the server after deploying? (ex. Database migrations executing with special command)
- What must be adjusted on the server only once? (ex. cron job, shared folder).

#### 2.4.6 Testing

All projects should include unit tests and the minimal requirements for the service provider are:

- setup a test infrastructure;
- write unit tests to cover the most key features;
- the minimal test coverage should be 20%;
- tests should cover own business logic (even if minimal) and not third-party API's / libraries.

Finally, a test-driven development is appreciated.

#### 2.4.7 API development

In case that within the project it is foreseen also the development or the change of APIs, the service provider should observe the following guidelines:

- all API calls must be documented in the README.md;
- Swagger UI should be used;
- in case of errors the API should return to the consumer valid and descriptive error messages;
- the API should be RESTful, if possible, but, in case of need, other formats will be considered. In case of non RESTful APIs the service provider should present to NOI S.p.A. enough documentation to allow NOI S.p.A. to decide whether to go on with the modern technology or stick to RESTful;
- the API must include also:
  - Response codes,
  - HTTP methods,
  - validity errors,
  - logging: JSON format for production and plain-text for local development written to stdout.

#### 2.4.8 Access Control List (ACL) management

In case that the project foresees Access Control List management, the service provider should observe the following guidelines:

- every login to a webapp needs ACL;
- the passwords must be complex enough to be secure;
- OAuth 2.0 standard is required Session management for webapps should be present, logout after an inactivity time (the length of the inactivity time depends on the single projects and must be agreed with NOI S.p.A.)

As an Access Management tool NOI S.p.A. uses Keycloak (<https://www.keycloak.org/>) instance. More details are available at the following links:

<https://github.com/noi-techpark/documentation#oauth>

#### 2.4.9 Dockerization

NOI S.p.A. is using Docker (<https://www.docker.com/>) to automate the deployment of the application and we strongly recommend to:

- use docker for local development;
- keep local docker setup, staging and production as similar as possible (these will be provided and updated by the NOI S.p.A. team);
- use environmental variables to configure different stages (i.e., .env files).

### 2.5 WORKING PLACE AND HOUR

#### 2.5.1 Working Hours

The execution of the works that involve collaboration with the staff of NOI Techpark or other entities involved in the project must be carried out within a timeframe ranging from 9.00 to 12.00 and from 15.00 to 17.00. Depending on the needs, different times may be agreed via email between the service provider and the entities involved.

#### 2.5.2 Working Place

The meetings that will be agreed during the project will take place, according to the needs of the project team, online or in the NOI Techpark offices:

- Via Alessandro Volta, 13, Bolzano.

Any expenses that the supplier will have to incur to reach these locations will not imply an additional cost for NOI Techpark. In any case, any travel costs that the supplier will have to incur to ensure the natural performance of the project activities (e.g., extraordinary coordination meetings, interventions that require presence on site, development activities to be carried out in agreement with the one or more entities / suppliers involved in the project, etc.) cannot be billed to NOI Techpark.

## 3. Request to the supplier

### 3.1 PROJECT INFORMATION

The supplier must include in all documents (e.g., offer, invoice, etc.) the following information:

**Project name:** IMPACT – Impacting Innovation Through Specialization

**Project code:** EFRE1048

**Project CUP:** J57H23000640009

### 3.3 COMPETENCES

Considering the activities included in this market research, the service provider who wants to send an estimation should provide a list of references. For these types of activities:

- web development, regarding data visualisation platforms;
- IOS,Android and webapp application development;
- App backend development.

## 4. Documentation

To participate in this market research, we kindly ask you to provide the following documentation:

- a short company description that also includes a list of references in similar projects;
- a short description of the team that will be assigned to the project including;
- a short description of the competences of each team member;
- the hourly rate of each team member included in the project team;
- the total effort estimation in terms of time and costs;
- a short project plan that includes the list of the project milestones and its deadlines.

## 5. Deadlines and contacts

The service providers who are interested in participating in this market research will have to present their estimation by the **10<sup>th</sup> of July 2024**.

In case of any question please contact:

Stefano Seppi

Email: [s.seppi@noi.bz.it](mailto:s.seppi@noi.bz.it)