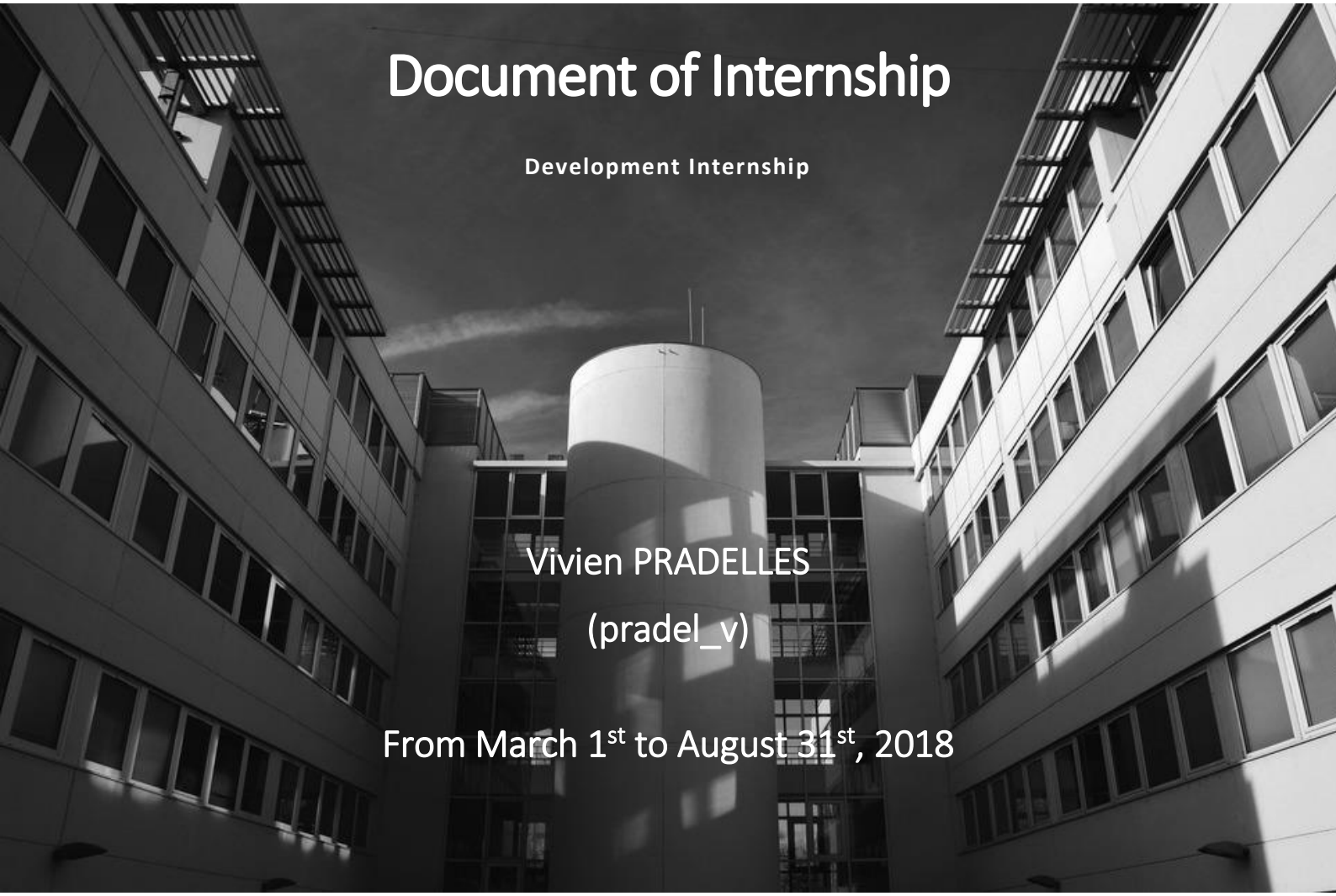


Document of Internship

Development Internship

Vivien PRADELLES

(pradel_v)

From March 1st to August 31st, 2018

This document is a synthesis of my internship at IoT-Valley. It will be composed of three parts; the first part explains to a new employee how the project and the association works, the second part is to convince my superior to give me the responsibility of our current project, and the last part is to highlight to my superior my motivation to undertake a new project in the association.

Epitech :
BODIN Théo-Paul
C teacher assistant

IoT-Valley :
RUIZ Bertran
Head of business and Partnership

Summary

Onboarding of a new employee

This part is a presentation of the IoT Rocket project by IoT Valley for a new employee who is joining the development team. It explains the context of the IoT Valley, our teams, why this project exists, the main knowledge about the project, the development team and the organization.

1. IoT Valley presentation

IoT Valley is an association (from French 1901 law) that creates an IoT BtoB ecosystem. This place was created by Sigfox under the name of Tic Valley in 2011 and renamed IoT Valley in May 2015.

This ecosystem was created to answer the need to help entrepreneurs create their IoT startup, in partnership with other businesses that have important processes and don't trust startups. To remedy this situation, Sigfox has created an entity that analyzes the reasons why startups crashed or succeed, help them to create partnerships with big compagnies, analyze the needs of these important companies, and give all the keys to grow a business to the entrepreneurs.

This process allows startups to do business with companies, create and maintain partnerships with them. These companies, when they see the success of IoT Valley collaborations, want to participate in this dynamic.

Another mission of IoT Valley is to accompany entrepreneurs to succeed by creating and developing their startup. In order to do that, they give a certain number of tools such as support to develop their activity on several levels like legal, organizational, commercial, project orientation and project development. This support is free when the startup begins, and with partnership when these works well and make more than 500 000€ of revenue figure.

For this accompaniment, IoT Valley has created the Connected Camp that is composed of 5 teams:

Operate: This team oversees the administration, as well as events, maintenance and rents spaces for startups.

Accelerate: Help startups on all tasks like development, prototyping, management. Develops all the tools needed by IoT Valley. This is also your team!

Acculturate: Create and institute the culture of innovation, sharing with events, meetup, slack administration.

Communicate: Communicate about IoT Valley and its events inside and outside the ecosystem.

Cooperate: Create and Manage partnership with compagnies to create contacts for startups.

2. Team organization

The organization of the work is separated in two period of time: The OKR for a long period of time, and Scrum to resolve OKR on a short period of time.

The OKR (Objective and Key Results) are objectives we want to reach, and Key Results that are little steps to fulfill these objectives (defined by a metric of, for example, 10 events organized or 3 new partnership). Defined on a period of 6 month, this is our long-term objective.

To resolve this, each team uses an agile method with sprints period from 3 weeks to 2 months. This agile method is based on a definition of tasks to do for the end of the period defined during an inter sprint. These tasks answer problematics given by the Product Owner who is not member of the team. Each employee defines the time of sprint execution and time for unexpected tasks. Moreover, they let the Scrum Master organize the sprint with the constraints of each member. The Scrum Master, also member of the team, is the Mentor and manages communication between people of the team and outside. This method has a period of introspection during the inter-sprint to upgrade work methods, adjust the task time estimation to create deadlines who fit with our work capacity.

3. Communication tools

These 5 teams must communicate to work effectively. For that, we have multiple tools to communicate and to organize our work such as Slack, our central tool, a multichannel messaging system. We have two “Slack” (current name given to an instance of Slack) used for communicating internally within the team and one to communicate with startups of the ecosystem of IoT Valley. As time goes on, we have developed some bots Slack like “bangs” who manages an internal money, “onboarding Slack” who is a tutorial of the ecosystem, and more.

Regarding project management, we use essentially Asana, an inexpensive tool who has all the features that we need, such as the timeline to manage the time of our sprints. We use also Taiga for the dev ticket management.

The documentation is another important part. It can be used for creating processes, stock information about an event from one year to another, give explanations on how work certain tools and more. The best tool we have found is Slite, a numerical block note developed by the slack teams that is easy to use.

We also use tools from Google suite. Gmail for the mail part, Google drive for file sharing, Calendar for planning, and the Google authenticator for all apps who permit social connecting.

There are many other tools for many different specific uses: Trello, and Teamgant for some project management, VosFactures for invoices for accounting, Spendesk for budget team management, ForestAdmin for self-developed app monitoring…

4. IoT Rocket project

a) Project context

IoT Rocket is a tool offered by IoT Valley who gives a good technical base to startups. In fact, entrepreneurs don't always have the knowledge in development or don't have time to create a user interface (easy for the eyes) for their products.

For that, after having examined what all new startups develop, the accelerate team has designed a tool with a generic base containing a user interface, user management, multi-customer management, data visualization, alert gesture, ERP connection, Payments, …

This project must have a generic architecture because startups need to be able to add functionalities and to take the result for development base. There must be the least possible maintenance, it must be easy to upgrade and inexpensive (with the least paid service to work).

b) Team context and organization

The development team (who belongs to the accelerate team) is composed of 8 people. The Product Owner is Simon. He creates the project idea with Bertran and follows the project development.

To create a user-friendly application, Marisella is a young UX (User eXperience) and designer for our applications.

Cyril, newest in the team, is an experienced developer. Comfortable with agile method, he is the team Scrum Master and developer full-stack (oriented to front-end development for team needs). He is assisted by Amir, a front-end student developer and Colin an ex back-end developer who has begun to be full-stack.

Nicolas is a young Backend developer. He has designed the back-end of the two last projects of the IoT Valley. He is assisted by me. He works closely to Cyril for design features.

The last teammate is Kevin, a back-end developer who works remotely from Australia. He is the oldest of the dev team. He sometimes works on isolated feature, bug fix, solution search, or on projects outside of the main project.

This is a newly created team. The last teammate has arrived just before the start of the development of IoT Rocket new ambitious project.

Each day, at 9:15 AM, the dev team meets to talk about their actions of the day before, difficulties they have encountered, and their objectives of the day. This meeting allows us to help a teammate when he is having big difficulties in his work. Each Thursday at 2 PM, we have a developer weekly meeting to talk about the current project progress, time estimation, and next sprint feature. This weekly meeting is an addition to the general weekly meeting with all team of IoT Valley It exposes what each member has accomplished, and the tasks for the week to come. This meeting also allows to make announcements and organize future events.

c) Development tools and languages

This part is important because it explains what tools and languages the development team uses and why. You must be familiar with development tools to be performant and understand the process of development.

- Development tools:

In addition to IoT Valley communication, the development team uses specific tools. Even if you can choose your computer, operating system and your source code editor, you must use the same code versioning control, code virtualization, ticket manager and other.

First, we need design and feature to develop. InVision is a good instrument to test and to share design within the team. Marisella is always working one sprint in advance (version 2) of the team (version 1) to create, design and model the next version of the application. It allows the development team to be familiar with the next design and make best technical choice to allow feature of the next sprint.

After the design is set, to coordinate the team for a good development, we organize (similarly to other teams of IoT Valley) our Sprint on Asana that has timeline and it is easily understandable by non-developers. In addition to Asana, we have Taiga for error ticketing (also named issues) on current and other projects. It permits during bug fixing period (at the end of each sprint) to assign tasks to each developer according to his responsibility of the issue.

For code sharing and versioning, we use Git with GitLab. We separate development in branches. A branch is a copy of the development code (also called branch Develop), it is also a copy of the master code as well as the code developed during the sprint. The master code is used on a production server. The Develop Branch is sometimes set on the pre-production server to test the new features developed.

When you create a new branch for a new feature, you must set a name who describes clearly what you are looking for. When you apply modifications locally, you can send your code on this branch. You must name your commit (current name given to a patch) with a specific pattern to have a clean history of code. This pattern is composed of a specific name (feat for new feature, fix for fixing issue, refactor for code refactoring or chore for update config), a sentence starting by an infinitive verb which describes your modification. When you have finished the feature, tested, and documented your code, you can create a Merge Request on GitLab. This is a code reviewed by your peers and mainly participates to clean your code and check bugs. The code review allows to easily share best practices into the team. After the review by peers, the Merge Request can be merged or refused with comments on what needs to be changed, fixed, or documented.

Because we allow developers and customers (startups who use it) to use any operating system of their choice, we use Docker for code virtualization. This tool offers multiple possibilities for a developer such as running the same application on different operating systems. It allows any developer to test locally (on his computer) the application with the same configuration than the production server. The application (that I will describe in the next section) is composed of different parts, each part is developed, and it runs in a different docker instance. It provides more security and allows us to run each part in a different server (if it's needed). Docker-compose is a docker instance manager proposed by Docker. It manages multiple Dockers and saves the deployment configuration of each one.

- Development languages:

One of our main choice criteria is that the language must be easy to learn or commonly used by startups that want to be able to upgrade our solution. It must be durable in time to not be obsolete during the development of our startups and create security failure.

Each part is different, and it has a language adapted to its needs. For the front-end, we use NodeJs with ReactJs Framework. It is a great couple for our needs because it is really performant, asynchronous (can handle multiple inputs at same time) and it can be used to make mobile applications with ReactJs cousins (React Native, a framework to make Android and iOS apps with the same code).

Python is a language commonly used by startups for bootstrap idea. It creates backend as it is very easy to learn. It is for this reason that we use this language to make the backend of IoT Rocket. Django is a modular framework liked by the developer community that allows lots of modules to replace many repetitive tasks such as Django Rest Framework to do RestFull Api.

To stock data, we use PostgreSQL. This is a performant database, that is easy to use and to deploy because it is already available in docker, has a Django module to link with the project and is already used by lots of startups.

To manage, we use a SAAS (Solution As A Service) named ForestAdmin. It's simple to maintain, easy to access and to use by startups. It's also my main part of the IoT Rocket. It's composed by front-end, that is maintained by the Forest Admin company, and a connector to Database where we can do data gesture.

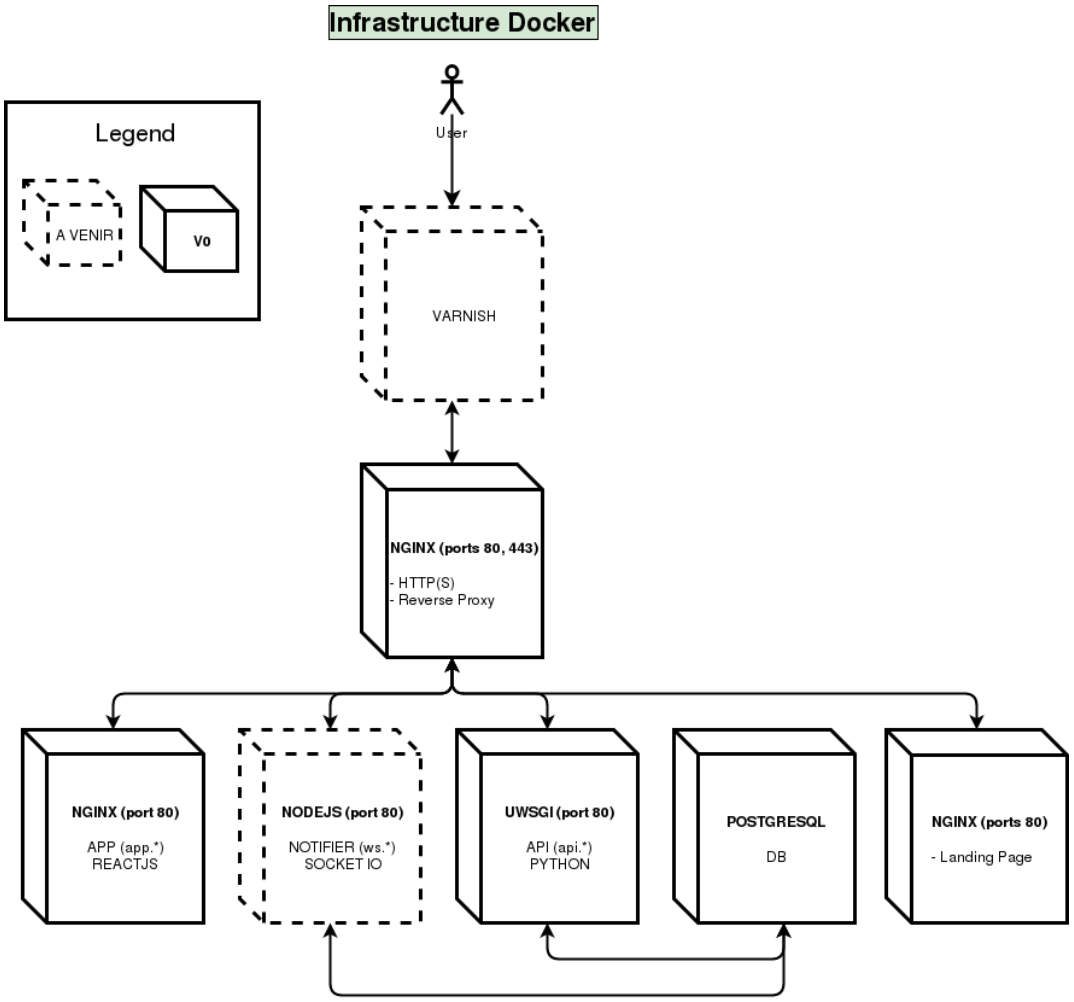
For the landing page of the app, we have used NodeJs with VueJs framework because it's light and does not have unnecessary functionalities. The landing page is only here to redirect users to the correct service and expose error pages.

To manage user connection to any application (backend, front-end, homepage, …) we use a Nginx reverse proxy. This ensures security by connection filtering and enrich request.

d) IoT Rocket architecture

The project architecture must be simple to be understandable by startups. If you understand, it is because we have succeeded in designing IoT Rocket.

- Global project architecture:



Credit: Cyril ALFARO – CTO of IoT Valley

This is the up to date global project architecture. You can see in the legend that we have some parts coming soon ("a venir"). The architecture hasn't been modified from the initial version ("V0").

As I have explained before, we have one docker container by applications. Now, you can see all the docker containers represented by cubes.

The first brick, varnish, (not developed) is an HTTP request handler who saves server response in a cache. It increases server performance for recurrent customer requests.

Next, the second brick is the reverse proxy. This is the module who redirects, the enriched user request, to the good application, with security check. This brick is important because it's the principal entry point to the IoT Rocket for the user.

When the user goes on the logging page or wants a new workspace, he goes on the Landing Page. On this page, the user can find or create his workspace. To resume, this brick handles all the requests that don't have specified workspaces and redirects them to the front-end application when a workspace is found.

The user is now on the APP brick (front-end). This page manages graphics, maps, and the user page. This is an important part for startups because it is the brand of the company for their customers. This implies that this brick must be easily customizable with startup brands. It must also be modulable to add new functionalities developed in the API brick (back-end)

As I explained before, the app brick exposes the response given by the API in a nice and understandable manner. This part must manage user permission (with ACL logic), store front data, and ask the startups API to get device data. The devices data aren't store in IoT Rocket because some startups already have developed device management or must make specific development in other languages. To store front data, the API has a unique access to the Database.

The Database is also in a docker container. It provides more security because we manage all access in the docker configuration. This database use PostgreSQL.

Furthermore, we have another part that is not represented in this scheme, as it is for the project's administration only. This is Forest Admin, a standalone interface that administrates IoT Rocket. It manages all devices, groups, permissions, dashboards and graphs… This is also the only way to add devices. This interface is only accessible by startups to administrate workspaces. For the customers, the administrators and owner can administrate directly on the app.

The last brick, not developed, is the socket. It allows backend to send data to front-end when updates are detected on the database. This is our next challenge, and it's also your first mission. We will describe this mission with you during your onboarding.

All these parts are stocked in different docker containers that form a local network only accessible by the reverse proxy. This architecture is scalable because you can run each container on a different computer. Using docker container allows good stability because if one-part crashes, the server and other parts won't, and you only must restart the container. IoT Rocket is very easy to duplicate for startups or developers thanks to Docker-Compose that deploys all the bricks automatically with one command.

- Our data generator:

As I explained before, we do not stock device data in IoT Rocket project for multiple reasons. To test and demonstrate our project to customers, we have developed an API with fake data. This part is more important than you might think because it is our only way to test our application and to know what the IoT Rocket needs are.

This project is based on the same docker deployment architecture than IoT Rocket with an API, data generation application and database.

The API exposes a GraphQL endpoint to get device data. It allows to only get the fields that we need. The data generation application generates data based on public API such as OpenWeatherMap.org. The database is the same as IoT Rocket.

e) IoT Rocket challenges

Now, I will introduce our difficulties on this project. This list is not exhaustive but resumes our principal obstacles.

- Newly created team

  The IoT Valley, as we know it, is only three years old and the budget at the time did not allow a development team. Today, the association has found funding to grant development of complex applications or tools. It is for this reason that most developers have less than one year of presence in the association.

  This problematic implies that we must set all development tools depending on each previous developer experience. We must learn to communicate and develop together.

- Modular project

  One of our main challenges is to make the project modular, this will allow startups to update IoT Rocket with their own modules. Furthermore, it will be easy to understand for beginners because startups don't always have the skills or time to learn about our solution.

  To resolve this difficulty, we make documentation about the project, modules and the way to upgrade this base.

- Multi databases

  Startups who use our solutions have multiple customers, we need to separate data from different customers and be GDPR compliant. Using one database per customer allows to remove easily or share the data of a company to itself. In addition, the database connection must be dynamic because startups can't be restarting servers when they add new customers. Restarting servers implies that other customers can use the solution during the restarting, and it is bad for user experience. Consequently, we must adapt our tools such as Django models or forest admin.

5. Other projects

On the side of IoT Rocket, we develop many tools for the teams or IoT Valley environment. In fact, during your onboarding, before you integrate the development team on IoT Rocket, we will give you a little side project.

For example, I will explain what my onboarding project is. I develop a connector between three tools that we use. I connect a form generator named Typeform that offers an integrated payment solution with another service named Vos Factures, that manages the invoices of the IoT Valley. This connecter permits to alleviate the accounting service.

The IoT Valley has developed an e-learning platform. The registration requires a subscription with payment. The connector allows to automatically subscribe to this platform. I have developed this project alone with NodeJS (the language that I was the most comfortable with at the time).

The team has made other projects such as bangs (money into IoT Valley ecosystem), Citymeo slack (a connector between slack and Citymeo to share information), Share bot (bot who adds LinkedIn, twitter, and slack share button on link in slack), and more…

My motivation for leading the project

This part is a cover letter to convince my superior to give me the lead on the current project. I will describe what I have done during my internship on this project and present the skills that would make me a good leader for this project.

PRADELLES Vivien
222 chemin de lapujade
31200 Toulouse

Cyril ALFARO
CTO at IoT Valley
231 Rue Pierre et Marie Curie
31670 Labège

Subject: Application for the position of Lead project manager of IoT Rocket

Dear Mr. Alfaro,

Your job position as the Lead project manager at the IoT valley has caught my attention and I am interested to continue to work on the IoT Rocket project.

For the end of my internship, we are finishing the version 2 of the IoT Rocket. It was a pleasure for me to work with you and the team to create this project from scratch. At the beginning, I only developed with JavaScript languages and during this adventure I have learned Python.

Upon my arrival, I worked on the Forest Admin connector alone to allow startups to administrate the solution. The principal challenge for this part has been to manage the connection to multiple databases dynamically. This difficulty was resolved with a retro engineering of the Forest Admin library and a good knowledge of JavaScript.

I also worked on the API of IoT Rocket in Python with Nicolas. I have made the mailing system and integrated Mailjet and worked with Nicolas on the development of the ACL module. It is a complex module who manages user rights according to his access and was the main challenge of the initial version of IoT Rocket. I also worked with the front-end team to create endpoints that they needed for their work.

Kevin, who works from Australia, has developed the Startup API example. I worked with him to create a connector GraphQL in IoT Rocket. Due to jet lag, the mission was a little bit difficult. But with good communication and documentation, we have collaborated and succeeded efficiently this task.

I really liked to follow and help front-end development for technical choice or back-end feature development. It is a field of work that I want to continue to discover in a near future and I think that Amir could train me to work with the front-end team.

Now, as of the 31st of August 2018, I have finished my internship and Simon has offered me to integrate development team. It's with pleasure that I accepted to stay with you in this adventure of the IoT Valley. In the future, the development team will have to work on other projects to help startups. You will most certainly manage these new innovative projects.

For all these reasons, I would appreciate if you would train me to become the Lead role for this project. In fact, you have been able to see during this project and my internship that I am perseverant faced to difficult problematics, I communicate well with colleagues and I have the capacity to learn quickly new technologies. In addition, I have good knowledge of all the aspects of this project and I can help and train startups on any points regarding this subject.

Finally, it would be a springboard for my career to manage a team on a project of this size. Indeed, I have never had the opportunity to work as team leader. Nevertheless, it is a goal I have set for myself, as it will allow me to develop new skills, such as management, problem solving and diplomacy. Undeniably, it will give me a first-hand experience in management, which is a training I will be able to develop even more in the future.

Thank you for taking the time to consider this application and I look forward to hearing from you in the near future.

Yours sincerely,

Vivien Pradelles

Create my dream project

This part is a cover letter to convince my superior to give me the responsibility of my dream project. I will describe this project, its context, my opinion about it and why I think that I can manage this project.

PRADELLES Vivien
222 chemin de lapujade
31200 Toulouse

Simon VACHER
Head of operation at IoT Valley
231 Rue Pierre et Marie Curie
31670 Labège

Subject: Application for the position of Developer at the IoT valley

Dear Mr Vacher,

Your job position as a Developer at the IoT valley has caught my attention and I am interested to work on your new project within your entity.

As you know, we research many different startups at the IoT valley and we offer to accelerate them and their project. Generally, it is hard to find startups, as they do not communicate a lot about their resistance, and mostly when they start even though it is crucial.

This situation could be resolved by developing a system that goes through different platforms on the internet, for example newspapers, websites regarding company creation and development, or even startup directories, and through this search we will find out which startups need help. This will allow us to be the first incubator to offer them a partnership, as it is known that most of the time startups affiliate with the first incubator that contacts them.

This project will be able to use as a base the technologies that we already have, such as Python Django that is mastered by all the members of the dev team and offers many modules allowing to browse and search the internet for the information we need. This project is for non-commercial purposes (it is not going to be sold). For this reason, instead of developing the frontend, we can also forest admin, which is very easy and quick to put in place.

This project, as well as easing us the path to find startups to develop our network, will also allow us to gather statistics on already existing and referenced startups. We will be able to extract information that interests us, for example the reasons why a startup fails, or what funding they have found to support their project. All this information will help us to guide our startups to success.

During my internship at the IoT valley, I was able to learn a lot through the other members of the team. I was able to acquire a real sense of teamwork, knowledge in development, project management, and self-training.

Moreover, with these 6 months experience I have developed my level in Python and more generally in backend design. In addition, learning about forest admin was a very good opportunity, as it is a very performant tool to expose a database.

At EPITECH, I have learned a lot about information research, a skill that I was able to use during my professional experiences and at the IoT valley. This competence will be useful in the context of this project and help me be efficient during its development.

Working at the IoT valley, has evidently showed me the importance of networking. This awareness allows me to be in the dynamic of “l’important dans la réussite d’une startup c’est l’humain” (the

essence of the success of a startup is the human element). Indeed, I have acquired a better understanding of the implications of creating links between startups, compagnies and the incubator. I can work more efficiently by being in the dynamic of the IoT values of creating, supporting and developing startups and their network. These values are the essence of the project.

I look forward to discussing my application for this position with you. Thank you for your time and thoughtful consideration. I hope to hear back you from you soon.

Yours sincerely,

Vivien Pradelles

Conclusion

After 6 months of internship at the IoT valley, I would like to thank my colleagues and members of my team for their insight and implication during this time. This opportunity has allowed me to create professional and personal relationships that I appreciate deeply.

I have learned a lot during this period and have been dedicated to my work. Indeed, my reward was to be offered a position at the IoT valley after my internship.

I am thankful to have been at the core of innovation. Indeed, working at an incubator has let me be in contact with entrepreneurs, constantly developing and using their creative intelligence to invent truly modern and innovative project and concepts. This has influenced me, and for my future career I wish to work in new technologies to reinvent the world.