APIs in the public sector of EU Countries

Michail Loukeris

Supervisor: Prof. P. Louridas

Department of Management Science and Technology
Athens University of Economics and Business

This dissertation is submitted for the degree of
Management Science and Technology

February 2020

I would like to dedicate this thesis to my loving parents ...

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Michail Loukeris
February 2020

Acknowledgements

And I would like to acknowledge ...

First and foremost, my academic supervisor, Prof. Panagiotis Louridas for supporting me during this semester by pointing me to the right direction in choosing an interesting, both academically and practically, topic for this thesis, as well as for his immediate support whenever I needed it. As well as Prof. Diomidis Spinellis for teaching me what a good research paper looks like.

I would also like to show gratitude to all my colleagues in GRNET, who treated me first as a friend, and taught me all I know when it comes to recruiting and selecting employees. Moreover, I would like to point out Ntora Morfi and George Korfiatis from GRNET and thank them for trusting me and bringing me onboard to their organization.

Finally, I would like to thank my family and friends. Without their patience, encouragement and support all these four years of my studies, this paper would have never been accomplished.

Abstract

Recent technological developments strongly influence the way we lead our life and the way we perform daily activities as shopping and communication. More and more new users are getting access to the Internet and to the newest technological tools, in order to make their life easier and better.

Having perceived the advantages of the use of new technology tools, many governments worldwide have consistently, for the last twenty years, introduced new tools into state operational structure, in order to simplify their public services, make them widely accessible to the public, reduce red tape, offer more efficient services and therefore a more effective governance. Countries like Estonia, the UK, Singapore and Denmark have successfully established fully digitalised public services, including tax paying, transportation, drive licence issuing, and employment procedures.

This research set out to explore the Application Programming Interface (API) landscape in the EU public sector and how APIs could play a role into the digital transformation of governance. More specifically, the aim of this work has been to examine the ability of Web APIs (hereafter "APIs") to assist member states of the EU into enabling their digital transformation. Areas of specific focus include cross-border interoperability between member states and the opportunity for the EU to become involved in developing or advocating API standards.

From a technological point of view an Application Programming Interface (API) refers to a set of clearly defined methods of communication between a service and any other software or components. It is essentially a protocol that allows two applications to interact with each other. In the public sector, APIs enable important functionality and information held in one agency's system or department to be readily available to another without significant and expensive development effort. As well as cross-departmental (agency) access to functionality and information (or even cross-border with a different country's administration), APIs also provide the ability to share information and functionality more widely, i.e, to developers and ultimately to citizens for consumption through web or mobile based applications.

As APIs enable the public sector to create ‘ecosystems’ inside and outside a government system, overcome the restrictions of traditional integration solutions, facilitate the opening of large data sources to citizens and other third parties and support innovation, they seem to play a key role in the digital transformation of efficient governance, generating benefits for the citizen, for business and for the economy.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

At a strategy level, the Tallinn Declaration, signed on the 6th October 2017 [12], confirms the commitment to the vision set out in the EU eGovernment Action Plan 2016-2020 [21] and in the European Interoperability Framework (EIF) [22]. In a time span of five years (2018-2022), steps will be taken towards the implementation of the following principles in EU open organizations: "computerized naturally, comprehensiveness furthermore, availability", "once-just", "dependability and security", "transparency and straightforwardness", and "interoperability as a matter of course", just as national interoperability structures dependent on the European Interoperability Framework (EIF).

In the Declaration the "user-centricity principles for design and delivery of digital public services" is vital. Public administrations and digital public services, upon interaction must fulfill the following requirements: digital interaction, accessibility, security, availability and usability, reduction of the administrative burden, digital delivery of public services, citizen engagement, incentives for digital service use, protection of personal data and privacy, redress and complaint mechanisms. At the same time, the Communication on "Building the data economy" (COM (2017) 9) looks at proven or potential blockages to the free movement of data and presents options to remove unjustified and/or disproportionate data location restrictions in the EU. It also considers the barriers around access to, and transfer of, non-personal machine-generated data and data liability, as well as issues related to the portability of non-personal data, interoperability and standards. In particular, it aims at the development of technical solutions for the reliable exchange and identification of data.

Moreover, the digital transformation of society, business and government is raising issues for a range of policy matters in the EU. As e-government has been in place for the last 20 years is several EU countries including the UK, Estonia etc., it is timely

to explore the interaction between technology and government activities from the perspective of digital government.

Aspects of the digital transformation of government concern the use of web Application Programming Interfaces (hereafter called "APIs")). APIs can be seen as "safe entry ports for new and innovative uses of data" held by companies and potentially, public administrations.

This study is set to explore web APIs as enablers for the digital transformation of governments. While digital transformation of government is much wider than the technologies which can potentially support it, an analysis of the role of APIs in the public sector is highly relevant to illustrate how technology can enable transformation of government. This study examines APIs, and their role in the EU public sector as well as it points out the differences with the private sector and explores future trends with a particular focus on current use of APIs in projects developed by various EU Countries. Specifically, in the first chapter we present a glossary as well as an API overview. Next, we discuss APIs as means to create ecosystems in the public sector and to overcome complex integration. We analyze how it supports open government initiatives and how it promotes innovation as well as some challenges and considerations along with some quantitative metrics. Once we cover all the above we will explore the differences of the public with the private sector in terms of API usage and more specifically we will talk about API availability, business models and disruption as well as making money from APIs. Finally, we will have a look at future API trends. The final chapter includes my internship report.

1.1 Glossary

Term	Definition
API	Application Programming Interface — It is a set of clearly defined methods of communication between the service and any other software or components.
API Ecosystem	The developers, and the users of the application constructs they build through an API, either within a company or on the Internet with business partners, customers, citizens etc.
API Economy	A set of business models and channels — based on secure access of functionality and exchange of data to an ecosystem of developers and the users of the app constructs they build — through an API, either within a company or on the Internet with business partners, customers, citizens etc.
API Versioning	The ability to change without rendering older versions of the same API inoperable.
API Standardisation	A uniform way for APIs to be expressed and consumed, from COM and CORBA object brokers to web services to today's RESTful patterns.
API information control	A built-in means for enriching and handling the information embodied by the API. This information includes metadata, approaches to handling batches of records, and hooks for middleware platforms, message brokers, and service buses. It also defines how APIs communicate, route, and manipulate the information being exchanged.

API portal	A means for developers to discover, collaborate, consume, and publish APIs. To support the overall goal of self-service, these portals describe APIs in a way that represents their functionality, context (the business semantics of what they do, and how they do it), non-functional requirements (scalability, security, response times, volume limits, and resiliency dimensions of the service), versioning, and metrics tracking usage, feedback, and performance. For organizations without mature master data or architectural standards, the API portal can still offer visibility into existing APIs and provide contact information for individuals who can describe features, functions, and technical details of services.
API gateway	A mechanism that allows consumers to become authenticated and to "contract" with API specifications and policies that are built into the API itself. Gateways make it possible to decouple the "API proxy"—the node by which consumers logically interact with the service—from the underlying application for which the actual service is being implemented. The gateway layer may offer the means to load balance and throttle API usage.
API brokers	Enrichment, transformation, and validation services to manipulate information coming to/from APIs, as well as tools to embody business rule engines, workflow, and business process orchestration on top of underlying APIs.
API management and monitoring	A centralized and managed control level that provides monitoring, service level management, System Development Life Cycle (SDLC) process integration, and role-based access management across all three layers above. It includes the ability to instrument and measure API usage, and even capabilities to price and bill charge-back based on API consumption—to internal, or potentially external, parties.

RESTful API	REST stands for "representational state transfer." APIs built according to REST architectural standards are stateless and offer a simpler alternative to some SOAP standards. For example, REST enables plain-text exchanges of data assets instead of using complex WSDL protocols which is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. It also makes it possible to inherit security policies from an underlying transport mechanism. At a high level, these and other simplified approaches can deliver better performance and faster paths to develop, deploy, and triage.

Table 1.1 Glossary

1.2 API Overview

APIs have become a key technological component of modern digital architectures, impacting every sector of the global economy. In the public sector specifically, APIs are a fundamental enabler of the transformation of its operations from analog (manual, paper) to digital.

The purpose of this study is to showcase the major contribution of APIs, when member states are in pursuit of their digital transformation. In order to explore this purpose, our investigation has covered the following topics:

- The current use of APIs in the EU public sector.
- Differences between API use in the public and the private sector.
- The future trends for APIs.
- Aspects of the API Landscape including API Ecosystems, API as low complexity infrastructures, API as components of a business plan.

In summary, in this study web — based research has been used as well as my experience from my internship in GRNET to gather information for analysis of successful but diverse API based case studies from a range of EU countries and sectors.

1.2.1 API Overview

API interaction occurs when one application would like to:

- Request a service from another application.
- Send data to that application.
- Access or query the data held by another application.
- Update data held in that application.

Types of APIs

APIs represent an architectural approach that revolves around providing programmable interfaces to different applications. It is technology agnostic, and creates a flexible, loosely coupled architecture that allows a solution to be made up of components that can more easily be switched in and out. The API approach is also a essential enabler for application developers to create apps that rapidly adapt to end user needs [33].

In the public sector, APIs enable important functionality and information held in one agency's system or department to be readily available to another without significant and expensive development effort. As well as cross-departmental access to functionality and information (or even cross-border with a different country's administration) APIs also provide the ability to share information and functionality more widely, i.e., to developers and ultimately to citizens for consumption through web or mobile based applications.

Although there are many different types of API (see Appendix I), this study is most concerned with web APIs. Web services expose these APIs as endpoints that any internet-enabled language or software can access, in exactly the same way browsers access websites and services [30]. Web APIs deliver requests to the service provider, and then deliver the response back to the requestor, i.e., they are an interface for web applications, or applications that need to connect to each other via the Internet to communicate [32].

Web APIs themselves can be broken down further based on the type of data format that they harness, for example, well known types are Simple Object Access Protocol (SOAP), Remote Procedure Call (RPC) based APIs, and the Representational State Transfer (REST) architectural style. GraphQL – is a data query language growing in popularity and has been adopted by leading social media outlets such as Facebook

and Pinterest [29] as a type of API. While typical REST APIs require loading from multiple URLs, GraphQL APIs get all the data an app developer needs in a single request enhancing speed of response even on slow mobile network connections [29].

Whilst the more traditional APIs are used as integration points within systems hidden from view, Web APIs are often publicly available and can be 'advertised' via API Directory sites online. Tens of thousands [50] are available for developers to deliver consumable information to end users to do everything, from checking traffic and weather, to updating a social media status, or even to make payments.

API Standards

In the geospatial domain, besides existing private companies famous API proposals (e.g. Google Map), the Open Geospatial Consortium (OGC) has created standards to support the exchange of geospatial information [37]. They describe their Web services API standards as an agreed specification of rules and guidelines about how to implement software interfaces and data encodings [38]. Geospatial software vendors, developers and users collaborate in the OGC's consensus process to develop and agree on standards that enable information systems to exchange geospatial information and instructions for geoprocessing. OGC standards are open standards. The OGC interface standards are also available in the REST style, and cover a number of aspects:

- Visualisation standards e.g. Web Map Service (WMS).
- Data Access Standards e.g. Web Feature Service (WFS), SensorThings API.
- Processing Standards e.g. Web Processing Service (WPS).
- Metadata and Catalogue Service Standards e.g. Catalog Service for the Web (CWS).
- The informatics contract between the client code which manipulates normalized data structures of geographic information based on the published API and the library code, e.g. the GeoAPI Implementation Standard.

The standards above are part of the few globally agreed specifications adopted by the Technical Committee 211 of the International Organization for Standardization (ISO). ISO is also known to be working on standards in other sectors, notably in Financial Services with ISO 20022 [31]: however, because of they are work in progress, details about them are still limited. Whilst standards of this formal and specific nature are used in the EU, there is clear evidence that the need for harmonizing APIs lifecycle

has been recognized. For example, the UK Government Digital Service recognized that departments were developing APIs using different tools, platforms and approaches [27], and have set about working with industry to create a set of common principles for API design. The output has been a set of guidelines on how developers working in any UK public sector organisation should build APIs [28] to ensure consistency, and success. These guidelines apply to other countries too. Although they are titled as a 'standard', they are generic, and not exact or specific in the way that an ISO or OGC standard is. Nevertheless, given the fact that government is increasingly using APIs to automate processes and provide citizens with access to new services it is hoped this approach will make integration simpler and faster.

Chapter 2

APIs in the Public Sector

Modern technology in combination with the Internet have permitted people to carry out online most of their daily activities and their transactions. These widespread and rapidly expanding possibilities have increased citizens' and business expectations in their interactions with government.

People are demanding transparency, accountability, access to information and competent service delivery from their governments. They also expect policies and services to be tailored to their needs and address their concerns.

In this section, we will analyze how APIs are used in the public sector. At first we will delve into APIs common uses, such as creators of ecosystems or as means to facilitate integration solutions. Specific examples are described in order to elaborate on APIs use in the public sector.Moreover, we will cover some challenges and considerations as well as examine data on the APIs advertised in one of the most respected API Directories such as ProgrammableWeb as a further indicator of the way APIs are used in the public sector.

Common uses of APIs concern their capacity as creators of ecosystems in the public sectors. Specific examples prove this aspect.

2.1 APIs as a means to create 'ecosystems' in the public sector

API — based ecosystems can be described as the extended interrelationships enabled by developers who build applications that connect various groups of stakeholders to each other via API based solutions that use the Internet to communicate [42].

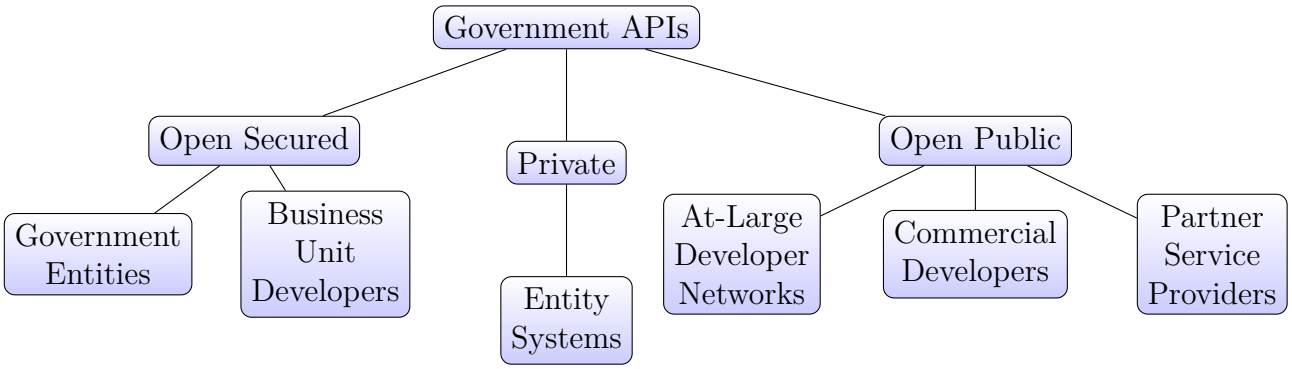

Fig. 2.1 Ecosystems enabled by government APIs [34]

An ecosystem may be created within a government entity, between entities or it may be wider reaching, for example between two governments or between a government,its citizens and third party providers.

The figure above illustrates the way in which APIs are used as well as the typical ecosystem that they facilitate in the public sector.

- **Private – Entity Systems**: These APIs are generally used to facilitate the sharing of data between systems within an entity, avoiding the need for complex point to point integration. Outside of the entity they are not visible to any person or body and are generally in the domain of the IT department. An example maybe a link between an internal administration system and a payroll solution.
- **Open Public – At Large Developer Networks**: Open APIs (i.e., they do not require permission to access them) are the entry point for developers to access large public data sources such as a census information or other similar statistical data, perhaps live sensor data from which to create citizen-facing applications.
- **Open Public – Commercial Developers**: Similarly to Open Public, but developers who are looking to gather openly available data for use, usually, in applications that can be sold. They may add value by combining the data, i.e. data on public transportation networks with location data available on an individual's smartphone to help the citizen make travel choices in real-time. Due to this open access, third-party integration of software is not only easier but also less problematic. Developers have access to the API at all times, so they can ensure that the two-way communication between assorted pieces of software is accurate.

  It is also worth noting the economic stimulation that this can bring. Transport for London's policy of working with major IT players (Google, Apple, Waze etc.)

but allowing their data to be available via the Open Government License has led to the creation of additional economic activity in the order of £100m of direct value and has enabled some 1,000 jobs [10].

- **Open Public/Secured – Partner Service Providers**: The APIs are open to partners often in the private sector which may include healthcare providers for example, who in some member states are interested in sharing healthcare records or confirming eligibility for free or subsidized treatment based on data held by a government organization.

- **Open Secured – Government Entities**: These APIs are available to other government organizations and allow them to share data only after they have authenticated. This supports many of the core tenets of digital government, allowing agencies to gather data on a citizen only once, and then share it securely. For instance this may involve the sharing of citizen data between the agency responsible for income and taxation, and those providing benefits in order that eligibility could be confirmed. Estonia X-Road Platform and Amsterdam City Data are some examples of this.

  Besides not being specifically mentioned in the diagram above, the ability to use APIs is not constrained by sector or geographical boundaries. Open Secured – Government Entities could include an application to application link between governments of different member states. A good example would be the Estonian X-Road Platform which uses APIs to share citizen's healthcare information with Finland.

- **Open Secured – Business Unit Developers**: Similar to the above, but instead of basic inter-entity data sharing, in this case the data is being consumed and then in some way supplemented in order to be useful by developers within a government entity. They are used to create custom applications around internal data assets for entity use.

In summary, the creation of an 'ecosystem' of providers and consumers fosters openness and efficiency and can also spawn the development of innovative service models, some of which may lead to revenue generation for the entities concerned (i.e., mapping data [40]). The ability of APIs to provide access into the governments major operations and data, in turn allows it to realize its objectives of openness, and of delivering efficient, secure, transparent and interoperable citizen centric services. The APIs are, therefore, a significant technological component, which will support the

evolution of public service delivery models, enabling entities to accelerate their shift from eGovernment to Digital Government.

2.2 APIs as a means to overcome complex integration

Many EU countries have built their computing infrastructure over the course of many years, constructing large, complex information systems featuring interfaces to share information from one system to another. Most of these interfaces were point to point and tailored to the needs of a particular project or entity at a given time. As the number of interfaces increased, so did the maintenance burden; the inter-relationships and the data duplication leading to an costly, complex and inefficient architecture [5]. In summary, these legacy government systems and associated business processes increase risk and exacerbate challenges in data sharing and service delivery across the ecosystem.

APIs provide an opportunity, in other words a structural solution, to permit the information within these legacy systems to be exposed with relatively low complexity and investment. They can be connected to legacy systems of record such as ERP systems [19], or citizen records to make the data records directly available helping in this way to bypass the complex interfaces of already existing systems, and allow data sharing to be accomplished more easily. This implies that a well-designed government ecosystem could help so that citizens or businesses will have to provide the same information less frequently (Once Only Principle, OOP).

A characteristic EU example of where API infrastructure is currently being used to overcome the limitations of traditional integration solutions is Estonia's X-Road Platform. It allows citizens to provide ordinary 'private and sensitive' data to public administrations only once, for instance, place of residence or place of birth. The ecosystem also includes private institutions such as banks who can have access in order to perform various functions.

ESTONIA X-ROADS PLATFORM

"X-Road is the backbone of e-Estonia. Invisible yet crucial, it allows the nation's various public and private sector e-Service databases to link up and function in harmony." [18]

X-Road is a government API framework developed by the Estonian Government and licensed under the MIT license. It is also used as a backbone of the Finnish National Data Exchange Layer.Initially built for SOAP/XML web services, it now extends to REST APIs. Rather than requiring governments to develop API management

directly, X-Road provides an API management layer, including an API gateway, which is open-sourced and available to governments worldwide. [25]

The X-Road solution includes a security server to provide authorization for government API access. It also provides central monitoring of API traffic.Apart from the management of APIs, it also provides an aggregation layer in front of multiple databases. This makes easier the creation and delivery of data access APIs.

Since each government service/entity has its own databases they all use X-Road to securely communicate and share 'private and sensitive' data to protect the 'once only' principle of sharing data with government. The service also incorporates many other sectors numbering over 900 organizations and enterprises including those in the banking, health and utility sectors [18]. Whilst they may use the platform to perform functions such as identity verification, powerful use cases such as automated extraction of funds from bank accounts for those failing to keep up to date with taxes are possible.

Given the above, the X-Road itself is a 'very low level engineered application' according to Andrew Kütt [1]. Following certification, an organization deploys an X-Road gateway so that it can hold secure private communications via APIs with other certified organizations that are legally able to share data with it. As a collective set of tools, the e-Estonia services provide the government of Estonia and its partners, including Finland, with a platform on which to innovate and use digital transformation to deliver new services across the globe.

2.3 APIs support open government initiatives

Open Government is the opening up of government processes, proceedings, documents and data for public scrutiny and involvement and is now considered as a integral element of a democratic society [41]. The Open government initiative started in 2009 by Barak Obama [51], after that, a great number of governments embraced open data initiatives. This is based on the belief that greater transparency and public participation can both lead to better policies and services, and also promote public sector integrity, which is necessary to regaining the trust of citizens in the neutrality and reliability of public administrations.

It has been acknowledged that APIs faciltate the opening of huge data sources to citizens and other third parties. The Open Government imperatives have meant that API technology has been exploited outside of the 'IT department', providing access into large open data stores so that developers and their applications and websites can more easily use it. When a government agency publishes an API for their data set,

they open up new and innovative ways to access the data. A developer might develop a mobile or web app to display the data as it is or allow simple queries or automatically depicted in charts.

The most relevant public sector that expose government datasets is The European Data Portal [47] (EDP).

European Data Portal (EDP)

The EDP provides access to 79 different catalogues, most with tens of thousands of open datasets provided by various member state governments. The same site also provides access to over 300 use cases (services or applications) that have been developed using the open data sets available. Some of these applications have been created using APIs to query the EDP.

The access to the Portal is provided by a machine-readable API which enables its users to search, create, modify and delete metadata on the portal. [45] APIs are available both via the Comprehensive Knowledge Archive Network (CKAN) [46] and SPARQL [48] endpoints.

2.4 APIs as a means for innovation

APIs enable new innovative service models which better engage citizens and allow for more efficient delivery of their services. These services no longer have to be provided directly by the agency, partners and citizen developers can use available data to enable new solutions. Smart cities and the vast amount of data produced by sensors supports the development of dynamic platforms and ecosystems providing contextualized, real-time location-based data from IoT or crowdsourcing to business partners and startups giving them opportunities to create new services or improve existing ones.

Transport for London is a successful platform which uses APIs in an innovative way to deliver services.

2.4.1 TRANSPORT FOR LONDON (TfL)

At a recent European conference [10], Transport for London detailed the investment that they had made:

- 200 data elements are made available through an API to about 12,000 developers producing over 600 apps that almost 40% of Londoners use.
- TfL has formed partnerships with major IT players such as Apple (for mobile payment, rental of bikes), Twitter (for pushing alerts out), a two-way data-sharing

agreement with Waze (enriching the app with data from the road network that TfL manages while benefiting from data collected through Waze) and Google (enriching the maps application with real-time data).

- The data can be consumed under the terms of the UK Open Government License with some minimal additions for free. This is done under a statutory requirement as part of UK legislation. Mechanisms are in place to ensure that consumption remains at an acceptable level. There is one single set of data at the base that are both consumed by TfL for its purposes and by third party developers. Developers must give attribution to TfL for the fact that their app includes TfL data.
- In terms of creation of additional economic activity, it has been estimated that this policy generates GBP 100m of direct value and has created over 1,000 jobs.
- All data made available is data that TfL collects anyway for its own purposes. TfL is not collecting additional data just to make it available to third parties.
- Combining data provided by TfL with privately-held data can bring additional innovative ideas (e.g. "Are there correlations between rainfall and collisions involving cyclists?").

2.5 Challenges and Considerations

To a great extent, externally facing public sector APIs involve the movement of data that is sensitive as it usually refers to information regarding a citizen. This creates a number of consistent challenges for government:

- **Regulation** – APIs play an important role in the facilitation of government transparency. A recent EU ruling [52] makes providing transparency into all IT services that will be used in technology projects a requirement in order to receive government funding. APIs are bound to support the technology required for the transparency principle.
- **Further regulatory considerations** - Considerations which must be adhered to when exposing data through any type of interface are the General Data Privacy Regulation [20] (GDPR), the Payment Services Directive (PSD2) [14] and the Public Sector Information Directive (PSI) [13].
- **Security** – APIs share data, services, and transactions in order to create new services. This inherently increases the permeability of an organization's network,

which can expose new vulnerabilities for exploitation. For that reason, APIs must be properly protected to ensure data privacy as well as citizen confidence in terms of service delivery. APIs meant for access to public data should be secured from inappropriate use or abuse such as denial of service. A number of potential security solutions exist. For example the Greek Government API of the Digital Solemn Declaration/ Authorization application uses solutions such as OAuth 2.0 along with OpenId Connect. Other solutions include Certificate based authentication, which are used in conjunction with a wider cyber security strategy and cryptography.

- **Specifications or Standards** - Standards for APIs are available in clusters such as the OGC [39] standard, and the developing ISO standard in Financial Services [31]. However, most organizations are developing APIs based on an internally accepted specification or style guide to ensure consistency, rather than what might normally be recognized as a de facto 'standard'. Every API comes with detailed documentation for consumers which specifies the type of API (RESTful, GRAPHQL, GRPC etc.). There is little intention for further standard development in the aftermath of 'Open Government' [23].

- **Business Models** – The government does not usually charge for data that is publicly owned and it is used for the public good thus not generating much income by users who wish to use that data. Examples of charging mechanisms being in place are limited, one being the UK's Ordnance Survey maps [40], and KLIP (Flanders Underground - Cable and Pipe Information Portal).

2.6 Quantitative assessment of API use in the public sector

It is not easy to reliably quantify the amount of public sector organizations that are currently using APIs internally, but the total number it is estimated to amount to millions [5]. Organizations that create externally facing APIs in order to interact with huge data sources are common worldwide, obvious by the numbers of APIs registered with online API Directories i.e., ProgrammableWeb [49] platform. To lure the maximum amount of developers to leverage the data being exposed, organizations publish their API with high-level technical specification. For that reason, an analysis of a popular directory is likely provide indicative information concerning the number

of EU public sector APIs as well as the sectors and associated public services that they support.

The best known and globally recognized API search directory is ProgrammableWeb [50]. Nordic APIs [3] comments that it is 'exhaustive' and 'comprehensive' and is both hand curated and searchable.

From almost 23,000 listed APIs in ProgrammableWeb (as at December 2019), we selected the 'Government' category which reduced the number searched to 787 (including the deprecated ones). According to our findings only 110 of the 787 Government category APIs advertised on the directory originated from the EU. This could be explained by the fact that ProgrammableWeb is based in the United States and not in Europe. As we can see in the table below most APIs were developed at a National level.

Scale (EU Coutries)	**APIs**	**Number of APIs**
City	- Transport for London - City of Helsinki Service Mapping	12
Regional	- The Statistical Institute of Catalonia - Open Greater Manchester	7
National	- Denmark Central Business Register (CVR) - Where Does My Money Go (UK budget spend)	71
International	- Openspending - World Government Data	7
EU	- Open Patent Services - VAT - OrganiCity Permissions - It's Your Parliament EU Data - Nephics European VAT Number Validation - iTranslate4.eu - European Union Legislation	12

Table 2.1 ProgrammableWeb EU API Analysis

The majority of these APIs provide access to open data sources for developers to use in order to create applications for commercial use while others focus more on citizens and democracy.

2.7 Summary

APIs expose data in a very cost effective way through both private and public ecosystems, which developers can consume to generate benefits for citizens, business and for the economy. The amount of APIs is growing rapidly each year as illustrated in online API Archives. This proves the value that they add for the public sector across a variety of use cases.

Chapter 3

Differences with the Private Sector

In this chapter we will compare the way the private sector exploit APIs, and how the public sector also exploits them.

3.1 API availability

As said before, although it is hard to quantify existing APIs as many of them are internal and unadvertised, externally available APIs are to a certain degree registered with API directories such as ProgrammableWeb [49], or RapidAPIs [2]. According to a survey carried out by Deloitte [5] the public sector may have slower growth than the private sector which is also believed to be slowing, or maturing in terms of underlying technology and potential. According to this survey, across global markets, public-sector API adoption lags and they suggest that this is due to ongoing Open Government guidelines that require longer time frames for organizing and executing larger scale API transformation initiatives [5]. However, as discussed earlier in this work, a huge amount of government data is being made available for consumption by citizen and commercial developers.

3.2 Private API Usage

Things that can be done with private APIs include [55]:

- Building internal apps for company use around a microservices model.
- Creating a shared pool of data and assets that allows teams to collaborate faster and easier.

- Strengthening partnerships, allowing partners to test out integrations, and streamline technical integrations.
- Streamline inbound and outbound marketing data collection, simplifying layered technology stacks via APIs.
- Building customer-facing apps with internal assets.

3.3 Business Models and Disruption

APIs have great transformative power to disrupt business, in conjunction with other technologies such as mobile and cloud.APIs are fundamental to the digital disruption in the commercial space, especially in retail, entertainment and social media [4]– probably to a far greater extent than government has been disrupted today.

- First Utility have shown that APIs having destructive potential to alter the electric utility industry within the UK. They help users easily switch utility providers, aided by an API that enables customers to receive quotes and sign up for their service. In this way, their API is disrupting a whole industry.
- The impact that Netflix had on Blockbuster made possible by Netflix's internal APIs, which handles two billion requests a day, and enables Netflix to develop and package new services for different platforms at speed [9].
- Amazon has required that all data-based communication between departments be done via API, naturally positioning Amazon to lead disruption in a world where APIs are becoming more and more ubiquitous. Amazon's disruption of the book industry was closely followed by providing access to their cloud via APIs creating a new business now worth 160 billion. [24]

The disruption of government may be the result of the fact that private sector, or third sector providers can integrate with government platforms via APIs to expose and use data to develop new and better service delivery models. Architectures will become modular and flexible so that they can be agile and responsive to changing demands from the ecosystem. In this way, differences with the private sector use of APIs will converge.

3.4 Making money from APIs

APIs are becoming more and more important in terms of revenue generating activities for business. In a recent survey of IT decision makers, Mulesoft, recently acquired by Salesforce [35], a vendor of integration software found that 50% of large enterprises (10,000+ employees) surveyed were making more than $10 million a year from API initiatives [43]. In the public sector, generating income from the provision of data that is publicly owned, and is being used for the public good, has rarely attracted fees. Examples of charging mechanisms being in place are the UK's Ordnance Survey maps and KLIP (Flanders Underground - Cable and Pipe Information Portal) which charges map requestors to have a digital map of utility services generated for a specific location.

Governments might need to start considering financial aspects of APIs as cost pressures become significant and might decide to adapt so as to make money from ecosystems. The above can be achieved through a cost per API call model where, for instance transport data is used by developers to build commercial applications. However, it is more likely for governments to have revenue as part of a service delivery ecosystem with the private sector that provides efficiency and cost saving. Such collaborations could be developed with cities and local governments, insurance and health private sectors in order to create ecosystems that deliver innovative solutions through applications.

3.5 Summary

To date, governments in order to fulfill their mission have harnessed the power of the API in order to make data more open and available to their citizens, and to themselves. The benefits range from increasing transparency, to enhanced efficiency of the existing service models.On the other hand, the private sector has harnessed APIs for a more transformative and disruptive end, developing to completely different business models, such as those which have made Netflix and Amazon great.

Chapter 4

The future trends for API use in the Public Sector

In this section, we will point out what the future trends of APIs in the public sector may be in a time span of 5 years.

- **Growth Rate** - There are some clues that the growth of APIs has decelerate to a certain extent [4]. However, although the number of APIs may not be growing at the pace that was anticipated a few years ago, their use and the ecosystems that they support continue to grow.
- **Digital Government Platform growth requires APIs** - Predictions on the future trends in Digital Government from research companies such Forrester suggest that Digital Government Platforms software will become more common in the next 3-5 years [25]. Digital Government Platforms require APIs as the integration mechanism to transfer data between component systems and therefore governments will continue to invest in switching from a service-oriented architecture (SOA) to a modular one (MASA) which is a new architectural model introduced by Gartner which stands for Mesh app and service architecture. It reflects what has emerged over the last five years, as organizations have experimented several digital projects. In the race to digital transformation, MASA focuses on enabling rich, fluid and dynamic connections of people, processes, services, content, devices and things. exploiting APIs and micro-services. [11]
- **Government will invest in Intelligent Things requiring APIs** – It is expected that governments will go on increasing investments in intelligent solutions, across many sectors — from defense, policing, waste management, health, agri-

culture and smart communities [16] in order to enhance service delivery quality, and efficiency. Sensor and video networks, intelligent drones, fleets of automated vehicles, and robotic devices will become fundamental to government service delivery capability and serve as a real-time data source for government, using APIs to move data among IT systems and layers. It is expected that the next progression will see the environment composed of many physical things with both sensor and computation capabilities, will make the technology direction pervasive and invisible [44]. Applications will be capable of communication, cooperation, and negotiation with each other. Unlike general applications, agents will be designed with targets to be fulfilled on behalf of its users. That is, agents will take appropriate actions efficiently towards its environment over the P2P protocol. For example, an agent can be designed to read a patient's biometrics from a patients wearable sensor devices such as a smartwatch and adjust thermostats to heat or cool a patient's room accordingly. In this way,the new platform is not limited to a certain set of devices, and it opens many possibilities over the P2P protocol to produce novel (multi-agent) applications that enrich the idea of ubiquitous computing [54].

- **Industry 4.0** – Industry 4.0 is the new wave of Digital Transformation in the industry, driven by recent developments in Cloud, AI + Analytics, IoT and API technologies. Nevertheless, the challenges of this transformation go beyond the mere acquisition and assimilation of new technologies. Part of the vision of Industry 4.0 consists of overcoming these barriers, enabling the flow of information and allowing coordinated actions among employees, systems, machines, and external collaborators. This is possible with the use of APIs that – in a standardized way and applying security mechanisms – enable the agile and secure integration between systems and devices. [26]

- **APIs as products** – APIs are products and as such should have a product lifecycle from conception and development through to withdrawal. In a similar manner given the fact that APIs are a technology to implement and forget government IT departments will need to communicate with third parties such as developers that depend on them, monitor their usage and withdraw them when needed (i.e versioning).

- **API Standards** – Not having to redesign an API due to its modular capabilities is cost saving and compelling for the public sector. The ability for applications and data sources to be able to connect without the need for a bespoke API takes

us one step closer to the ubiquitous platform of open data. However, it is of utmost importance for the developer community, to know the standards of an API, and then getting to know the specifications of the API usually through the developer portal of an API provider.

- **Citizen developers and Open APIs** - With open APIs citizens can easily make use of open data, or improve already existing applications which leverage it. Hackathons will become more widespread as a way for the public sector to interact with people. This helps member states in their aim to conduct significant user research before releasing any citizen facing services. Moreover data changes could potentially come faster than if we were to wait for the vendor to implement them. This process is very similar to open-source software, which is widely used and very helpful for developers [36].

- **AI and ML-based APIs** - Using predictive analytics APIs to combine big data, embedded, visual, spatial/location, text, web, network and mobile information has evolved to include natural language processing especially within context per Business Intelligence trends [7]. New sources of real-time data help with trend detection for faster responses to the intelligence. For example, Netflix now offers interactive content that simulates the Choose Your Own Adventure format most notable in a "Black Mirror" special [8]. Another example is how Axway partnered with Elastic Beam to leverage an AI API [6].

4.1 Conclusion

The future of digital government seems deeply linked to the use of the APIs as enablers. As the technological demands of digital government move forward, it appears that APIs are well positioned to keep pace, and provide the access points needed to enable fast and secure data sharing to support government's needs from law and order i.e., police data [15], to healthcare and the environment. As with all aspects of technology, the use and development of APIs will evolve over time.

Chapter 5

Conclusion

This study set out to explore the API landscape in the EU public sector. The purpose of the study has been to identify areas where APIs are enablers of governments digital transformation. Areas of specific focus include aspects such as sources of open data, differences between APIs in the private sector and the future trends of APIs.

The report provides a useful baseline overview of APIs, considering what they are used for, the different types of API that can be leveraged, and the API standards that exist. A glossary of terms and API types in the appendices provide further resources for the target audience. The report then goes on to consider how APIs are used in the public sector. The findings showed that APIs are used by the public sector to help them achieve their goals in four main ways:

- Enabling ecosystems.
- Overcoming complex integration of large systems.
- Supporting open government initiatives.
- Enabling innovation and economic growth.

The use of APIs has its challenges too. This study highlighted security and enhanced EU regulation around privacy as considerations for API owners. An API is another gateway into a computer network and associated data, and requires the security features and ongoing maintenance that such an interface deserves.

The lack of standards does in some way hinder interoperability both internally and externally to government agencies. It is forcing organizations to develop their own set of guidelines to ensure alignment, and this is something that the UK Government have recently released to all API developers [28]. However, the use of API gateways, and

the predominance of RESTful architectures is in some way diluting the pressure for a standard.

Differences with the private sector were also considered. The report found that to date, government has harnessed the power of the API to make data more open and available to their citizens, and to themselves. The benefits range from increasing transparency, to enhanced efficiency of the existing service models. The private sector has harnessed APIs for a more transformative and disruptive end, giving rise to completely different business models, such as those which have made Netflix and Amazon leaders in their field.

Our research also considered the future of government, which will be to some extent built on the API as a key enabler. As the demands of government move forward, it appears that APIs are well positioned to keep pace, and provide the access points needed to enable fast and secure data sharing to support government's needs from law and order, healthcare and the environment.

Our study provides brief examples of governemnt solutions with APIs at their core such as Estonia's X-Road, UK's TfL and Greece's Digital Solemn Declaration/ Authorization issuing system.

We truly believe that our study has given some useful insight in how APIs as a technology can contribute to Governments digital transformation.

Chapter 6

Report

The goal of this report is to give an overview of the things that I have interacted with, throughout my internship, analyzing thoroughly both the deliverables and the theory behind the technologies used.

6.1 Company Description

GRNET S.A. provides Internet connectivity, high-quality e-Infrastructures and advanced services to the Greek Educational, Academic and Research community, aiming at minimizing the digital divide and at ensuring equal participation of its members in the global society of knowledge. Additionally, GRNET develops digital applications that ensure resource optimization for the Greek state, modernize public functional structures and procedures, and introduce new models of cooperation between public bodies, research and education communities, citizens and businesses.

GRNET provides advanced services to the following sectors: Education, Research, Health, Culture.

GRNET operates under the auspices of the Greek Secretariat for Research and Technology / Ministry of Education, Research and Religious Affairs.

Advanced E-Infrastructures and Innovative Services

The backbone interconnects more than 100 institutions including all universities and technological institutions, research centers, public hospitals, museums and libraries, as well as the Greek School Network, with speeds up to 26*10 Gbps through its high-speed, high-capacity infrastructure of long-term leased fiber that spans across the entire country.

GRNET is present in global networking for research and education, representing Greece in GÉANT. GÉANT is Europe's leading collaboration on network and related infrastructure and services for the benefit of research and education, contributing to Europe's economic growth and competitiveness.

Large Scale Computing Services for the Researchers - Cloud Computing

GRNET offers innovative Cloud Computing services that are available via the Infrastracture as a Service model, under the brand name "~okeanos". By using "~okeanos", any academic user can create a multi-layer virtual infrastructure and instantiate virtual computing machines, local networks to interconnect them, and a reliable storage space within seconds. Thousands of academic users have already utilized virtual machines in the course of their research, experimental, educational or other activities. The Cloud Computing infrastructure and services of GRNET have been made available to the pan-European R&E community via the "okeanos-global" service.

Hellenic High Performance Computing Infrastructure

The "ARIS" national high-performance computing infrastructure provides state-of-the-art supercomputing capabilities to the Greek scientists. The system went into pilot operational phase in June 2015 and it is available for productive use to all researchers and scientists across Greek universities, technological institutions and research centers. The system enables the implementation of scientific and technical large-scale applications, with GRNET guaranteeing its smooth operation offering comprehensive end-user support.

ARIS is based on IBM's NeXtScale platform, incorporating the Intel ® Xeon ® E5 v2 processors, (Ivy Bridge) and it provides computational power that reaches 170TFlops (trillion floating point operations per second). With a total of 426 compute nodes, it offers more than 8500 processor cores (CPU cores) interconnected through FDR Infiniband network, a technology offering very low latency and high bandwidth. In addition, the system will offer about 1 Petabyte (quadrillion bytes) of raw storage, based on the IBM General Parallel File System (GPFS). The system software allows developing and running scientific applications and provides several pre-installed compilers, scientific libraries and popular scientific application suites.

Enhancing the Use of ICT and Access to Digital Content

GRNET coordinates a series of initiatives aimed at creating e-infrastructures and services that can facilitate organizing, describing and promoting digital content of educational, research, geospatial, and environmental as well as cultural topics. These actions contribute to the vision of creating a virtual horizontal infrastructure of digital repositories, which is available from universities, research centers, museums, libraries and other institutions in the country and Europe and facilitates the preservation, sharing and exploitation of digital content by businesses and the society. The ultimate goal is to enhance the use of digital content and services from researchers, teachers, staff of public bodies and SMEs, as well as other types of communities involved in the production, processing, and use of digital knowledge.

Environmental Policies

GRNET implements green technologies in its networking and computing infrastructure, in an attempt to reduce its yearly greenhouse gas emissions' footprint. In order to achieve that, environmental regulation, laws and codes of practice are highly regarded when assessing standards of environmental performance.

To this respect, GRNET's environmental policy is based on the following lines of action:

- upgrades of networking & computational infrastructure with energy efficient equipment,
- deployment of an energy consumption monitoring infrastructure for real time measurements in the network Points of Presence (PoPs) and Data Centers,
- participation in research activities for the design of energy-aware mechanisms in the operation and control of the network,
- improvement of energy efficiency in GRNET data centers (low Power Usage Effectiveness – PUE values) through the application of innovative energy-aware techniques,
- increase of environmental awareness within the Greek research and academic community through the dissemination of "green" best practices,
- increase in the use of videoconferencing tools to achieve commute trip reduction,

- minimization of environmental pollution through the reduction, reuse or recycling of materials.

GR-IX

GRNET operates GR-IX, the Greek Internet Exchange. GR-IX constitutes an important national infrastructure as it interconnects significant players of the Greek Internet, such as Internet Service Providers, Content providers etc., and facilitates the exchange of IP traffic among them.

The goal of GR-IX is to improve the connectivity, quality and speed of the Greek Internet and, at the same time, reduce the cost of accessing it.

6.1.1 Internship Goal

The general goal of this internship, is gaining experience in Software Engineering through the exposure to the complete lifecycle of Software Development, understanding how each step impacts the rest. The main focus is towards the coding aspect of System Development Life Cycle (SDLC), writing reusable, clean and well documented software.

6.2 Internship Basic Characteristics

6.2.1 Organic Structure

The Organic Structure consists of the Management Bodies, as provided for in Presidential Decree 29/1998, as amended and in force, the Support Structures and the Directorates. The Board of Directors of GRNET SA, which determines the direction in which the Company moves and takes all final decisions, is: The General Assembly (GA), the Board of Directors (SA), the Chairman and Chief Executive Officer and the Deputy Chairman BoD. The organizational structure of GRNET SA.

6.2.2 Board of Directors

The Board of Directors is the highest administrative body of GRNET SA. and is competent to decide on any matter relating to the administration and the achievement of the Company's aims, except for the matters relating to the exclusive functions of the General Assembly. The Board of Directors mainly shapes the Company's strategy and development policy, secures the main sources of funding of GRNET SA while supervising and controlling the management of its assets. It also cultivates and

promotes the national and European dimension of the Company. The BoD has eight members and his term of office is four years. The Chairman is a member of the BoD, as well as the Company's supreme executive. It carries out the scientific supervision of the functions of GRNET SA, presides over all its Directorates and Services and directs its work, while proposing to the Board on policy issues.

6.2.3 Directorate of Administrative and Financial Management

The Directorate of Administrative Functions and Financial Management assists the Board of Directors and the Chairman in their work and supports the Company in relation to the following: It supports Administrative, Operational and Secretarial Management of the Company, its Management Bodies, Human Resources and its Projects. Technologically it supports the Company's Management Bodies, and Human Resources on Internal ICT Infrastructure. It processes the Company's supplies. It supports the Company in managing all funded projects. It carries out the accounting management and financial monitoring of the Company and the financed Projects.

6.2.4 Directorate of Electronic Infrastructure

The Directorate of Electronic Infrastructure has the responsibility to operate, upgrades, install, modify and supervise the performance of the productive infrastructure and services, maintenance and costing of its operation, the management of the GRNET support center and infrastructure support teams, contacts with contractors, suppliers and subcontractors.

6.2.5 Information Services and Application Development Division

The Information Services and Application Development Division is responsible for the full life cycle of the projects and applications deployed internally in the company, from the planning phase to the maintenance process. It also owns the information services that the company has to the users of the wider educational and research community. It helps define the company's strategy in the field of ICT and its implementation. It also participates in the submission of selected proposals and in the execution of national and European projects.

6.2.6 My Role

In GRNET my role is that of a backend developer. As a backend developer I primarily design and develop APIs. Specifically, I worked a project called Dilosi. The Project Dilosi is a project carried out by GRNET on behalf of the Greek Government and its main purpose is digitalize the process of issuing a **solemn declaration** as well as an **authorization**. This project aims to change the way Greek citizens go about issuing a formal declaration which is going to KEP. There are over 10 developers working on this project. Out of these developers some of them are backend developers, some are frontend and the rest are responsible for the CI/CD(Continuous Integration / Continuous Development), scaling and deployment of the app. Dilosi started on early October and it is scheduled to go live by the end of March. Regarding the technologies used on this project, the frontend is developed using REACT Framework [53] while the backend is based on Django Framework [17]. As for the CI/CD as well as our Version Control System we use our own GitLab server. Currently as a database we use PostgreSQL but it is very possible that we will migrate to Oracle as this is what the government uses currently. For building the application as a whole (frontend, backend, database) we use Docker. Regarding the way we operate, we use Scrum which is an agile process framework for managing complex knowledge work. Scrum consist of daily meetings in our subgroups, as well as weekly meetings. Every couple of weeks we schedule a retrospective event where we discuss how the past weeks were, what was wrong, what was right, what to fix and propose ways to strengthen the team.

Required Skills

In order to be a backed developer I had to be able to perform well under pressure. I had to be able to produce high quality, maintainable and reusable code using Python Language. I had to be able to develop RESTful web services, have good knowledge of UNIX based systems and be familiar enough with Git. Moreover, I had to be able to communicate ideas, meet deadlines and overall be an active team member. Furthermore, I needed to understand and cooperate with developers from other subgroups such as frontend developers. Being able to use the terminal efficiently is a big must as well as be familiar with concepts such as ssh and public/private keys. Another important requirement is to be able to use third party libraries and be able to read through their documentation to see what fits the needs of the project.

6.3 Tasks

In Dilosi I was assigned many tasks all of which I completed successfully. Specifically,

- I assisted in designing the database models as well as most of the API.
- I designed programatically the template of the new digital solemn declaration and authorization based on the original paper ones.
- I developed the logic of creating/submitting a Declaration.
- I developed the SMS Verification functionality.
- I designed and implemented the My / Entity Inbox which is the inbox of either an agent that has a pending declaration or a citizen that has been authorized by someone else.
- I designed partially and implemented the Actions feature which basically includes all the possible actions an Entity or a User can perform on a declaration (accept/reject/revoke etc.)
- I implemented the backend infrastructure for administrators to add / edit new Declaration Templates e.g Zero Income Declaration Template and Template Types e.g., Editing the "Solemn Declaration" Template Type footnotes.
- I have written most of the backend tests which are mostly integration tests.
- I have also implemented the sharing functionality declarations and authorizations have in order for agencies and citizens to receive them in their inboxes.
- I have developed all the action logging infrastructure of the backend.

6.4 Results

Overall my results were pretty good which is the main reason I was hired as a full time backend developer for various national projects such as the platform for digital driver license issuing or the Pre School enrollment platform. Specifically, regarding the above tasks all the aforementioned featured I was asked to implement have now been merged in our development branch and are about to go into production one Dilosi goes live. Three of the major tasks I has to complete were the new digital templates of solemn declaration and authorization, all the backend API tests and and the implementation of My/Entity Inbox.

Design of digital solemn declaration and authorization template

This was one very important and challenging tasks as I had to first print the original paper application an take a ruler, measure each box, its margins etc. and try and recreate the same design digitally. For this project I worked along with Prof. Panagiotis Louridas at a separate repository from the rest of dilosi project. The newly created template compared to the original ones should also include a qr code containing the reference code of the application as well as being digitally sealed with the digital government seal. Another challenge I face was to implement a way to shrink the text to fit the boxes as I couldn't the text length anyone would insert so I had to this dynamically. I have spent over 50 hours over the course of 20 days on this task and it is being maintained until now with various improvements and bugfixes. As mentioned before this is an extremely important task not only for the company and the projects but for all of the Greek citizens, as without it after completing a digital solemn declaration fields nothing would happen. Fortunately, there were no delays compared to the initial planning. Everything went smooth and this piece of code is currently merged into the development branch. The way I worked was by pushing commits everyday through Git tou GitHub initially and the merge it to GitLab.

API Tests

Another extremely import task I managed to complete was the develop the full testing infrastructure of the API. Specifically, this was most probably the most important and longest in duration task I was assigned. What I had to do was to write integration tests to test the flow of the application as a whole. Unfortunately. this is not a one time task. As more code and features get added to the application more tests have to be written to ensure that the application, features and their functions operate as intended. Imagine what would happen if there were no tests and our untested application would go into production and being used by 11 million Greek citizens? Therefore I consider this task extremely important. Regarding the task itself I have written over 1000 lines of code and about 25 tests with multiple functions each, which test from the creation of a declaration to accepting or rejecting an inbox. The special thing about these test is that they are not unit tests but integration tests which means that they connect to each other testing a use case and a flow. Again on this task I commited every day through Git to GitLab. I have spent on this task over 100 hours in a course of my 4 month internship and I expect to spent many more hours as code gets integrated in our codebase. Some challenges I faced were that I had to go through the Django

testing documentation in order to write the script that would execute the tests as well as trying to get the coverage as high as possible. Some issues that emerged were that when the CI/CD team tried to integrate our test suite to GitLab so it runs every time someone pushes a commit, they couldn't do it successfully do the test having third party dependencies mostly with memcache so I had to refactor many things in our tests in order to make it work.

My / Entity Inbox

This task is a fairly recent task I was assigned. It basically involved creating all the backend infrastructure for different government entities e.g., DOI, EFKA etc. as well as citizens to be able to be notified about incoming declaration and authorization as well as being able to interact with them. Something like an email inbox but for declarations and authorizations. Again this is a very important task as without it there wouldn't be any point in having digital declarations if there was not a way to send them accordingly. The way this feature operates is that when someone picks a recipient while completing the fields of the declaration there are 3 scenarios. First scenario is that he picks a recipient from the list of verified recipients. These are agencies that have been verified by the Greek government. In that case the declaration automatically goes to the Inbox of that recipient as has a pending state. Second scenario is picking an entity only by its name. In that case since we cannot easily assign this name to a verified entity due to conflicts in the name (lowercase letters, uppercase letters, spelling mistakes etc.) we create the declaration but won't put it to anyone's inbox. Third scenario is that someone picks an entity by both its name and email or just email. In that case although the entity might not exist and to being verified having the unique email is a way to create that entity's inbox and put that declaration there. At the same time we notify that email that there is a pending declaration for them and when the entity logs in to review the declaration then it will get verified. This case usually refers to schools and a popular declaration would be the No Religion declaration many parent do for their kids. Regarding this task i have spent about 80 hours over the course of about 5 days. Again, the way I worked was to push commits everyday to our application repository in GitLab. Some challenges I face was the how to design the feature in order to fulfill most cases and scenarios as well as how to develop it efficiently. Of course this feature came with its own set of tests which were added to our tests suite. Regarding the initial scheduling there were 3 days of delay due to me being sick at that time.

6.5 Time / Project Scheduling

In the following table I mention the task I was assigned as discussed earlier along with the duration of each one in days. I should mention that if a task states a number of days it doesn't mean that I worked on this task for X consecutive days, but instead it could be a sum of multiple hours. A day is considered to be 8 hours. Some durations have an asterisk which means that although this task is initially completed it might get improvements and bugfixes.

Activity / Task	**Duration in Days**
Designing of Models / Database Schema	4
Designing and Developing Digital Solemn Declaration and Authorization pdf Templates	20*
Developing of API Tests	100*
Implementation of the SMS Verification System	3
Implementation of the Creation/Submission of a Solemn Declaration	17
Implementation of Sharing Functionality	2
Refactoring Codebase	40*
Desging and Implementation of My / Entity Inbox basic flows	10
Design and Implementation of Templates / Template Types / Fields	63*
Database Stress Test Script	3
Implementation of Entity / User Actions	32

Table 6.1 Internship Activities / Tasks along with their duration

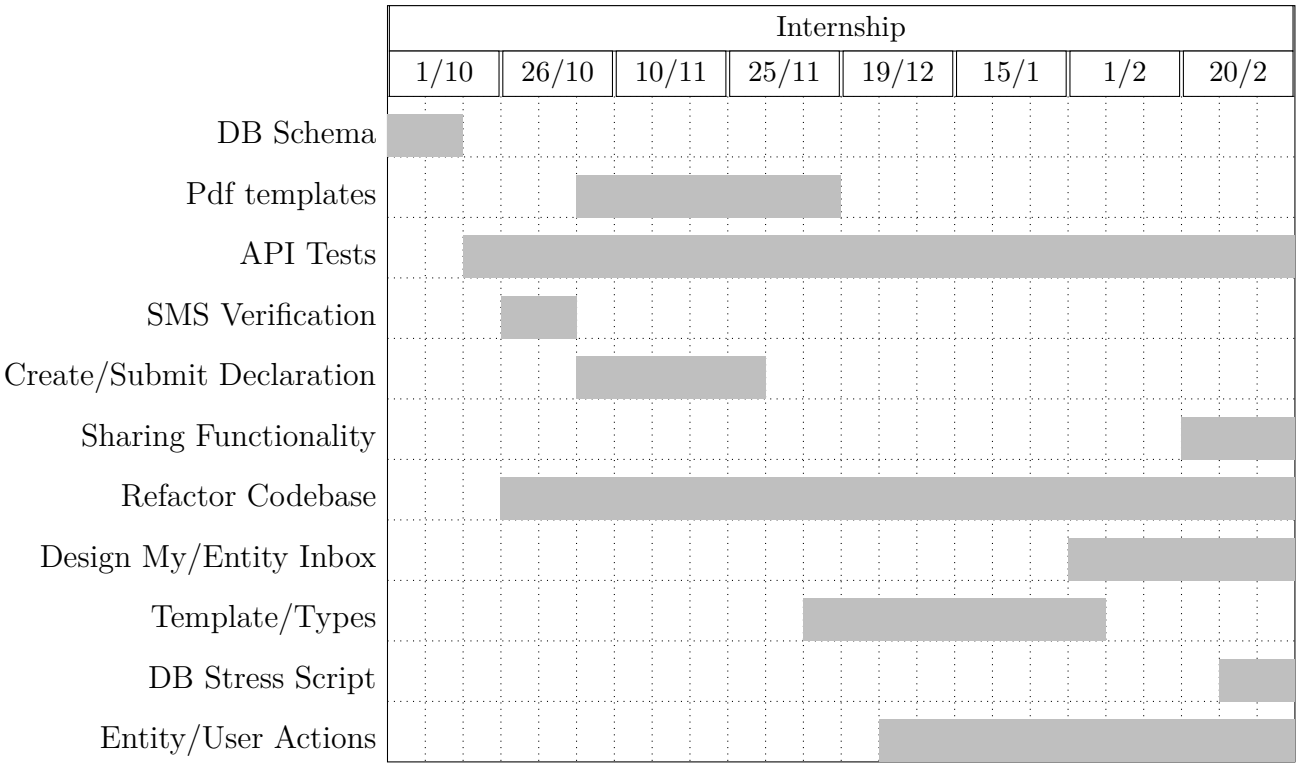

Fig. 6.1 Internship Tasks Gantt Chart

6.6 Skills

During my internship I used many skills which I learned from AUEB while being an undergraduate student. In the table that follows I present those skills along with the course that taught me, the methods and some example in which is used them.

Skills	**Method**	**Example**
Database Management Systems	Microsoft SQL Server	I used this skill to query the SQLite and PostreSQL Database of dilosi application.
Software Engineering in Practise	Software Design Patterns,	Developed high quality, maintainable, modular and testable code
Cloud Applications	Django Framework	Since Dilosi is based on Django Framework, this was the most important skill. All the code I have written the past months is within Django context
Project Management	Work Scheduling	It helped me by prioritize work in such a way so there are no delays
Programming 2	JUnit, Git	I used git every day to publish my work as well as Integration tests to test the API.
Personal Skills Development	Interpersonal Communication	It helped my communicate better with my coworkers and understand each other while being respectful and tolerant.
Optimization Methods in Management Science	Analysis and design of optimization methods	It helped to better schedule my assignments.
Algorithms and Data Structures	Python, Algorithms	I used Python as my main development Language every day.

Table 6.2 University Skills that proved useful

6.7 Comments

The past 4 months have been great. From day one I felt very welcomed in GRNET. The on-boarding package included a desk to work and MacBook Pro along with an external monitor, a parking slot, lunch from a catering and a very good salary. All developers I worked with were very welcoming and eager to help, respectful and tolerant to mistakes. The department I worked within is very well organized, with weekly team meetings, 1-1 meetings with the supervisor and monthly written report. Every couple months we host a retrospective event where we share our thought for the past months and propose ways to become better at a personal and team level. At first I was anxious and stressed that I wouldn't be able to perform good enough bearing in mind that my coworkers are very experienced and have worked in great companies in the past but as very soon I came to understood that this wasn't the fact because what I was assigned to do is exactly what I have been taught all these years in university so in a sense I was preparing for the past 4 years for this job. The past four months for me have been mostly learning new things ranging from technologies, to coding languages, to developing new interpersonal skills to understanding how a working environment operates. I have master my GIT skills, I have started developing in VIM Editor, I have familiarized with Docker, Kubernetes and CI/CD. If I had to change something in my Department I would change the fact that there isn't a single source of truth. We use at least 3 platform to post our issues and our TODOs and there is not much vertical communication in terms of requirements and scheduling. I would rather propose having a single platform where everything is posted there from issues to news regarding the Greek government requirements and so on. Something else i would like to point out as I believe it is very important especially for developers is their work to be widely used. At GRNET I had a very rare opportunity of developing application for the Greek government, in other words an application that will be used by 11 million Greek citizens. Overall, I am extremely happy and grateful for this opportunity and once again I would like to thanks my university for preparing me accordingly all these years, Prof. Panagiotis Louridas for supporting me every step of the way, my coworkers for welcoming me and helping me everyday to become a better software engineer and AUEB's Internship Office for providing me with the opportunity to work at such a great company.

References

[1] Andrew Kutt, L. A. (2017). Estonia's modern digital governance: A conversation with andres kütt and liina areng. [online] https://www.csis.org/podcasts/strategic-technologies-podcast/estonia%E2%80%99s-modern-digital-governance-conversation-andres-k%C3%BCtt [Last accessed : 2019-12-29].

[2] API, R. (2019). Find and connect to the world's top apis. [online] https://rapidapi.com/ [Last accessed : 2019-12-29].

[3] Apis, N. (2019). Api discovery: 15 ways to find apis. [online] https://nordicapis.com/api-discovery-15-ways-tofind-apis/ [Last accessed : 2019-12-29].

[4] Arnaud Lauret, B. C. A. K. (2016). The api economy. [online] https://19yw4b240vb03ws8qm25h366-wpengine.netdna-ssl.com/wp-content/uploads/theapieconomy.pdf [Last accessed : 2019-12-29].

[5] Arun Perinkolam, L. V. C. (2017). Api imperative: From it concern to business mandate. *Tech Trends 2018*. [online] https://www2.deloitte.com/us/en/insights/focus/tech-trends/2018/api-program-strategy.html [Last accessed : 2019-12-29].

[6] Axway partnership with EAlastic Beam (2019). Axway partners with elastic beam to leverage artificial intelligence for api security. https://www.axway.com/en/company/media/2018/press-release-axway-partners-elastic-beam-leverage-artificial-intelligence-api. Accessed: 2019-12-29.

[7] BI Trends 2020 (2019). Top business intelligence trends 2020. https://bi-survey.com/top-business-intelligence-trends. Accessed: 2019-12-29.

[8] Black Mirror Special (2019). "black mirror" special. https://www.youtube.com/watch?v=bTnJORFcDbY&feature=youtu.be. Accessed: 2019-12-29.

[9] Blog, N. T. (2012). The secret to amazons success internal apis. https://netflixtechblog.com/embracing-the-differences-inside-the-netflix-api-redesign-15fd8b3dc49d. Accessed: 2019-12-29.

[10] Borschette, A. (2017). Data access and transfer with a focus on apis and industrial data platforms. [online] https://ec.europa.eu/information_society/newsroom/image/document/2017-32/report_final_for_web_

C285AA6E-0C77-373C-999BF6DFBCC3F995_46252.pdf [Last accessed : 2019-12-29].

[11] Castellani, S. (2017). What is masa – mesh app and service architecture? https://apifriends.com/api-management/what-is-masa/. Accessed: 2019-12-29.

[12] Commission, E. (2017). Tallinn declaration on egovernment. [online] http://ec.europa.eu/newsroom/document.cfm?doc_id=47559 [Last accessed : 2019-12-29].

[13] Commission, E. (2019a). European legislation on the re-use of public sector information. [online] https://ec.europa.eu/digital-single-market/en/european-legislation-reuse-public-sector-information [Last accessed : 2019-12-29].

[14] Commission, E. (2019b). Payment services (psd 2) - directive (eu) 2015/2366. [online] https://ec.europa.eu/info/law/payment-services-psd-2-directive-eu-2015-2366_en [Last accessed : 2019-12-29].

[15] DATA.POLICE.UK (2020). Police api documentation. https://data.police.uk/docs/. Accessed: 2019-12-29.

[16] David W. Cearley, M. A. (2016). Gartner identifies the top 10 strategic technology trends for 2017. [online] https://www.gartner.com/en/newsroom/press-releases/2016-10-18-gartner-identifies-the-top-10-strategic-technology-trends-for-2017 [Last accessed : 2019-12-29].

[17] Django (2019). Django project. https://www.djangoproject.com/. Accessed: 2019-12-29.

[18] E-Estonia (2019). X-road. [online] https://e-estonia.com/solutions/interoperability-services/x-road/ [Last accessed : 2019-12-29].

[19] ERP (2019). Erp integration - a united application architecture. [online] https://www.mulesoft.com/resources/esb/erp-integration-application-architecture [Last accessed : 2019-12-29].

[20] EU General Data Protection Regulation (2019). Gdpr portal: Site overview. [online] https://www.eugdpr.org [Last accessed : 2019-12-29].

[21] European Commission (2016). Communication: Eu egovernment action plan 2016-2020 - accelerating the digital transformation of government. [online] https://ec.europa.eu/digital-single-market/en/news/communication-eu-egovernment-action-plan-2016-2020-accelerating-digital-transformation [Last accessed : 2019-12-29].

[22] European Commission (2017). The new european interoperability framework. [online] https://ec.europa.eu/isa2/eif_en [Last accessed : 2019-12-29].

[23] European Payments Council (2017). How can application programming interface standardisation be achieved in the context of the revised payment services directive? [online] https://www.europeanpaymentscouncil.eu/news-insights/insight/how-can-application-programming-interface-standardisation-be-achieved-context [Last accessed : 2019-12-29].

[24] Evangelist, A. (2020). The secret to amazons success internal apis. https://apievangelist.com/2012/01/12/the-secret-to-amazons-success-internal-apis/. Accessed: 2019-12-29.

[25] Finnerty, B. (2018). A digital government technology platform is essential to government transformation. [online] https://www.gartner.com/en/documents/3534849/accelerate-data-centric-digital-government-with-a-proact [Last accessed : 2019-12-29].

[26] Gimenes, N. (2018). What is masa – mesh app and service architecture? https://sensedia.com/en/api/why-do-iot-and-the-industry-4-0-need-apis/. Accessed: 2019-12-29.

[27] Government Digital Service (2018). Identifying the challenges of designing cross-government apis. [online] https://technology.blog.gov.uk/2018/01/29/identifying-the-challenges-of-designing-cross-government-apis/ [Last accessed : 2019-12-29].

[28] Gov.uk (2019). Api technical and data standards. [online] https://www.gov.uk/guidance/gds-api-technical-and-datastandards [Last accessed : 2019-12-29].

[29] Graphql (2012). Graphql. [online] https://graphql.org/ [Last accessed : 2019-12-29].

[30] Insider, G. T. (2014). Government agencies leading app economy using innovation drive economic growth. [online] https://federaltechnologyinsider.com/government-agencies-leading-app-economyusing-innovation-drive-economic-growth/ [Last accessed : 2019-12-29].

[31] ISO (2018). Iso standard for apis. [online] https://www.iso20022.org/news/apis-and-iso-20022-whitepaper [Last accessed : 2019-12-29].

[32] Karen C.Simon Mueller (2017). Api. [online] https://www.definition.net/define/api [Last accessed : 2019-12-29].

[33] Kiran Somalwar, C. A. B. D. (2014). Api-fication core building block of the digital enterprise – white paper. [online] https://www.hcltech.com/white-papers/systems-integration/api-fication-core-building-block-digitalenterprise [Last accessed : 2019-12-29].

[34] Lacheca, D. (2016). Accelerate data-centric, digital government with a proactive api program. [online] https://www.gartner.com/en/documents/3534849/accelerate-data-centric-digital-government-with-a-proact [Last accessed : 2019-12-29].

[35] Mulesoft (2019). Salesforce signs definitive agreement to acquire mulesoft. [online] https://www.mulesoft.com/press-center/salesforce-acquisition [Last accessed : 2019-12-29].

[36] Mygov.scot resources (2019). Digital first service standard. [online] https://resources.mygov.scot/standards/digital-first/ [Last accessed : 2019-12-29].

[37] OGC (2019). Ogc® standards and supporting documents. [online] http://www.opengeospatial.org/standards [Last accessed : 2019-12-29].

[38] OGC e-Learning (2019). Introduction to ogc standards. [online] http://cite.opengeospatial.org/pub/cite/files/edu/ogcstandards/text/services-ogc.html [Last accessed : 2019-12-29].

[39] Open Geospatial Consortium (2017). Ogc® open geospatial apis - white paper. [online] http://docs.opengeospatial.org/wp/16-019r4/16-019r4.html [Last accessed : 2019-12-29].

[40] Ordnance Survey (2019). Os places api. [online] https://developer.ordnancesurvey.co.uk/os-places-api [Last accessed : 2019-12-29].

[41] Organisation for Economic Co-operation and Development (2019). Open government. [online] http://www.oecd.org/gov/open-government/ [Last accessed : 2019-12-29].

[42] Paolo Malinverno, M. K. (2017). From apis to ecosystems: Api economy best practices for building a digital platform. [online] https://www.gartner.com/en/documents/3760363/from-apis-to-ecosystems-api-economy-best-practices-for-b [Last accessed : 2019-12-29].

[43] Pearlman, S. (2016). How much is an api worth? [online] https://blogs.mulesoft.com/biz/api/how-much-is-an-api-worth/ [Last accessed : 2019-12-29].

[44] Peiris, P. (2017). How iot is strengthening ubiquitous computing (part 2). [online] https://dzone.com/articles/howiot-is-strengthening-ubiquitous-computing-part-2 [Last accessed : 2019-12-29].

[45] Platform, E. P. S. I. (2016). Understanding the european data portal. [online] https://www.europeandataportal.eu/sites/default/files/2016_understanding_the_european_data_portal.pdf [Last accessed : 2019-12-29].

[46] Portal, E. D. (2019a). Api version 3. [online] http://www.europeandataportal.eu/data/en/api/3 [Last accessed : 2019-12-29].

[47] Portal, E. D. (2019b). Use cases. [online] https://www.europeandataportal.eu/en/using-data/use-cases [Last accessed : 2019-12-29].

[48] Portal, E. D. (2019c). Virtuoso sparql query editor. [online] http://www.europeandataportal.eu/sparql [Last accessed : 2019-12-29].

[49] ProgrammableWeb (2019a). Apis, mashups and the web as a platform. [online] https://www.programmableweb.com/ [Last accessed : 2019-12-29].

[50] ProgrammableWeb (2019b). Apis, search the largest api directory on the web. [online] https://www.programmableweb.com/apis/directory [Last accessed : 2019-12-29].

[51] The White House President Obama Archives (2019). Open government initiative. [online] https://obamawhitehouse.archives.gov/open [Last accessed : 2019-12-29].

[52] Union, E. (2019). Eu funding. [online] https://europa.eu/european-union/about-eu/funding-grants_en [Last accessed : 2019-12-29].

[53] Walke, J. (2013). Reactjs. https://reactjs.org/. Accessed: 2019-12-29.

[54] Wikipedia (2019). Ubiquitous computing. [online] https://en.wikipedia.org/wiki/Ubiquitous_computing [Last accessed : 2019-12-29].

[55] Wodehouse, C. (2016). Public apis vs. private apis: What's the difference? [online] https://www.upwork.com/hiring/development/public-apis-vs-private-apis-whats-the-differenc [Last accessed : 2019-12-29].

Appendix A

API Types

API Type	Data Formats	Description
Web APIs	SOAP over HTTP/S	SOAP is a protocol that defines the communication method, and the structure of the messages. The data transfer format is XML. A SOAP service publishes a definition of its interface in a machine-readable document, using WSDL – Web Services Definition Language.
	XML-RPC over HTTP/S	XML-RPC is an older protocol than SOAP. It uses a specific XML format for data transfer, whereas SOAP allows a proprietary XML format. An XMLRPC call tends to be much simpler, and to use less bandwidth, than a SOAP call.
	JSON- RPC over HTTP/S	JSON-RPC is similar to XML-RPC, but uses JSON instead of XML for data transfer.

	REST over HTTP/S	REST is not a protocol, but rather a set of architectural principles. Some of the characteristics required of a REST service include simplicity of interfaces, identification of resources within the request, and the ability to manipulate the resources via the interface. The most commonly-used data format is JSON or XML. Often the service will offer a choice, and the client can request one or the other by including "json" or "xml" in the URL path or in a URL parameter. In a well-defined REST service, there is no tight coupling between the REST interface and the underlying architecture of the service. This is often cited as the main advantage of REST over RPC (Remote Procedure Call) architectures.
	GraphQL	GraphQL is a data query language developed internally by Facebook in 2012 before being publicly released in 2015. It provides an alternative to REST and ad-hoc webservice architectures. While typical REST APIs require loading from multiple URLs, GraphQL APIs get all the data an app developer needs in a single request enhancing speed of response even on slow mobile network connections.
Library based APIs	JavaScript APIs, TWAIN, Twilio	To use this type of API, an application will reference or import a library of code or of binary functions, and use the functions/routines from that library to perform actions and exchange information.

Class-based APIs (object oriented) – a special type of library based API	Java API	These APIs provide data and functionality organised around classes, as defined in objectoriented languages. Each class offers a discrete set of information and associated behaviours, often corresponding to a human understanding of a concept.
Object remoting APIs	CORBA	These APIs use a remoting protocol, such as CORBA – Common Object Request Broker Architecture. Such an API works by implementing local proxy objects to represent the remote objects, and interacting with the local object. The same interaction is then duplicated on the remote object,via the protocol.