LAUREA
UNIVERSITY OF APPLIED SCIENCES
Together we are stronger

Building a reusable application with Django

Moore, Jonathan Ian

2016 Leppävaara

Laurea University of Applied Sciences
Business Information Technology

Building a reusable application with Django

Moore Jonathan Ian
Degree Programme in Business
Information Technology
Bachelor's Thesis
July 2016

Laurea University of Applied Sciences
Leppävaara
Business Information Technology

Abstract

Moore Jonathan Ian

Building a reusable application with Django

Year 2016 Pages 44

This thesis is project based and is focused on creating an online web application using the open-source web framework, Django. The project object was to take a pre-existing web application then develop it into a reusable fully functional product, which utilises the latest technologies available on the market. The research covers the advantages and disadvantages of using Django and compares it to other available web frameworks. The objective of the investigation was to provide San Francisco Oy with justification for using this particular web framework.

The thesis project was commissioned by the owner of the Dimera Application, San Francisco Oy, which is a digital media company based in Helsinki. Dimera is an abbreviation of Digital Media Ranker, and its creation was the primary intention of this venture.

This thesis gives a descriptive analysis of what the Dimera team achieved during the duration of this project. It covers the implementation of the app and the tools and methods that were used to arrive at the final conclusion. The result of this project was the deliverance of the Dimera application, along with guidance on how to develop and deliver a fully functional Dimera website using the Django framework.

Keywords

Django
Reusability
Programing
Python
HTML
CSS
Framework

Table of contents

1 Introduction

For this project, the assigned team will be building an application with the Django framework. The task is to seek the benefits of building an application with Django, and find out what the framework can offer an application, and how the final product can integrate with other technologies to deliver maximum reusability. Django is built with Python. Which is one of the most commonly used programming languages and can be found on some of the leading websites on the net. The language is slowly losing popularity because of some PHP based web frameworks. This is because PHP offers frameworks where developers can build fully functioning websites with little to no knowledge of programming. But with simple web technologies on the market are people disregarding Pythons good functionality, security and originality and replacing it with cost efficiency.

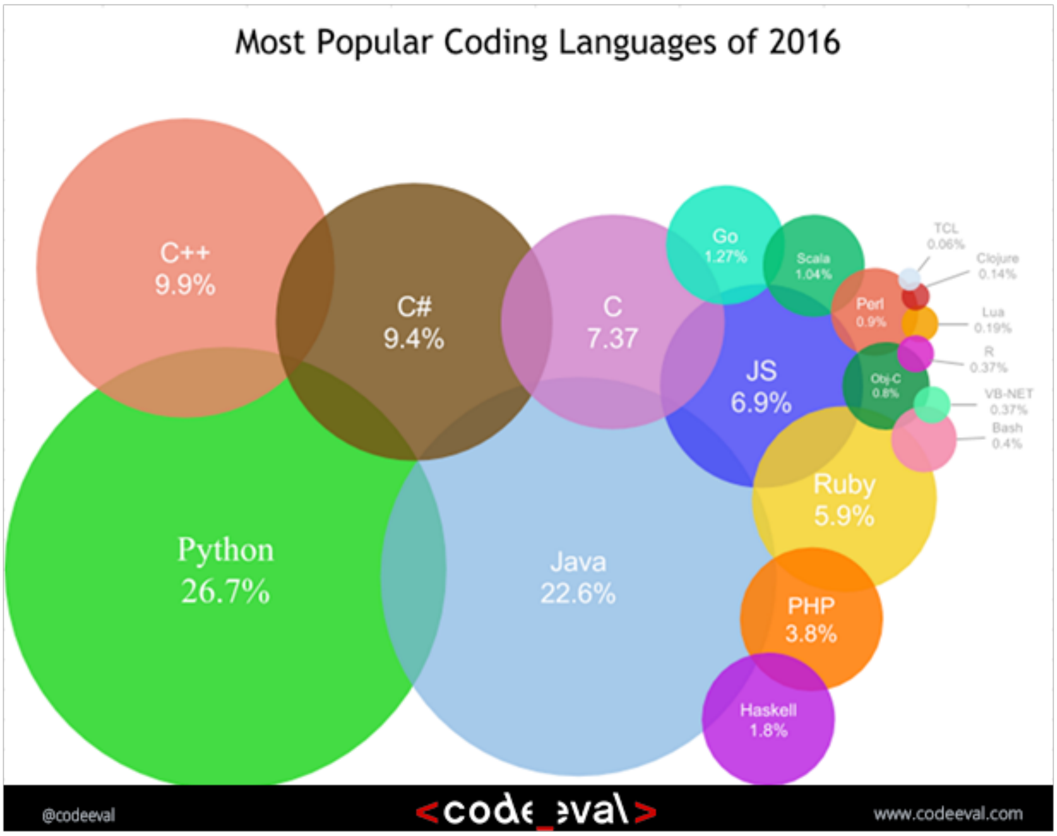


Figure 1: Pythons Popularity

The purpose of this project is to take a pre-existing piece of software used in the Startup100 site and build it into a platform that can be reused in other websites and sold to potential clients. The project consists of three stages; first taking the code from startup100 and improving what's already there, secondly clean the code and make it easy to set up and reuse so it can become a sellable product, then to take the software and test it on a completely new site. This thesis will mostly concentrate on taking the platform and creating an application to be used on new website projects.

This thesis is going to firstly explain the company and project background, and then it will outline the scope and objective of the new project. It will then move on to methodology behind the project and the research that was needed to complete it. Section three goes into detail about the project requirements. It will explain all the technology and tools that have been used to create the Dimera platform and the ShopHunt website. Subsequent to that, it will go into the project implementation explaining what was made during the project and how. It then moves on to a general evaluation of the final product and how the project runs overall, ending with a conclusion where the project will be heading in the future.

1.1 Company background

Juhani Polkka, Ceo of San Francisco Oy, assigned this project. San Francisco Oy is a digital marketing company based in Helsinki, which specialises in services for start-ups. Juhani Polkko founded San Francisco Oy in 2009. They claim to be the first agency in Finland to purely sell their services to start-ups only. Juhani detected a need for start-ups to be listed and ranked. So investors can find trending start-ups within Finland with ease. This is when he decided to make Startup100.

Figure 2: San Francisco Oy Logo

1.2 Project background

The company San Francisco Oy first created Startup100. The idea was to create a trending ranking system that takes data from multiple sources then ranks the data on a monthly basis so users and investors can track which start-ups within Finland are currently most popular. The website was built by a local Helsinki company called Skyhood, who decided to build the website using Django.

Startup100 gives the user of the website the option of checking the latest trending start-ups that are automatically updated on a monthly basis and lists them from number one up until one hundred. The user can then check what the overall top one hundred start-ups in Finland are.

Each start-up on the site has its own profile page where they can display a logo, a description of the company, a list of the products they sell, links to their website and social media profiles and a line chart of the companies financial data.

The website uses Duunitori as an outside service so Start-ups can advertise possible job opportunities, on the Startup100 Jobs page

1.3 Team

Dimera founder Juhani Polkka assigned the project and roles. The team consists of four members, Project manager Semyon Leshchenia, Tero Nurmiluoto as back-end developer, Jonathan Moore as front end developer and content creator, and Oona Colliander as designer and branding.

Name	**Project Position**	**Roles**
Juhani Polkka	Founder and CMO	
Semyon Leshchenia	Project Manager	Project Planner
		Assign Tasks and Roles
		Keep Track of Schedule
		Project Reports
Tero Nurmiluoto	Back-end Developer	Design Back-end
		Program Back-end
		Test and Debug Code
		Document Code and Manual
Jonathan Moore	Front-end Developer	Design Front-end
		Program Front-end
		Design User Interface
		Program User Interface
		Test Debug Code
		Document Code and Manual
Oona Colliander	Designer and Branding	Create Company Identity
		Design and Create Imagery
		Company Branding

Table 1: Project Team and Roles

1.4 Objective

The objective of the project was to take the ranking platform from the pre-existing website Startup100, then update it and develop it into a reusable sellable platform that is complicating for a client to install and rank any digital data of their choice.

The new name for the platform is Dimera. It stands for Digital media ranker. In order for the platform to become sellable, it will have to be made easy to install for the customer and be made adaptable and reusable so it can be used in as many scenarios as possible.

The Dimera platform was used and tested on a fresh website developed by the team, that will allow users to search for whichever product they wish to purchase, and the software will then list and rank all the E-commerce stores within Finland that sell the particular product which they were searching for. The purpose of this website is so users can find the best store in which to buy a product. The website will generate income through marketing each store on the list. In order to become top of the ranked list, each company will have to gain good feedback from customers, plus increase the user traffic from their website and social media pages. The current name for the new website is ShopHunt

1.5 Scope

This thesis will go into the background of the project and tools used to shape the Dimera platform. It will go into depth about how the Dimera platform will be made reusable, and how Dimera will be implemented into a new site.

The projects deliverables will be to craft a finished fully functional app that can be reused in as many scenarios as possible. The three main components of the project will be to take the pre-existing app from Startup100 and develop it further into a sellable product. This will involve cleaning the code and adding more features to the app. The app will be called Dimera. The next stage was to try the Dimera app in a different scenario, tracking and ranking different kinds of data from different sources. The project was started in February 2016 and will end in November 2016.

1.6 Limitations

As with all projects, time is always a limitation, as the project needs to be complete by November 2016. To pull everything together at the finishing stage will be a challenge. Deciding exactly what the finished deliverable is can also be tricky as projects of this kind can evolve as the development phase goes further. A schedule has been conceived and is followed but evolves on a weekly basis as more updates are built, and there is always the possibility that certain complications will arise.

Reusability is the biggest issue that the Dimera project faces and is the main focus point of this thesis. For the application to become lucrative it will have to be made easier to install on as many platforms as possible. The application will also hit a larger target audience if it can analyse as many different data types as possible. The team's overall knowledge on the target market could limit the final projects reusability.

The project had difficulties at the beginning when it came to finding freelance developers to join the team. The team that produced Startup100 can no longer assist with the project because of other projects that they are working on. Finding the right developer for Django became problematic because the Django community in Finland is small. We tried to outsource the position to other countries but found the communication level for this kind of a project through information technology was insufficient. A Ruby developer was contacted and a PHP developer to see if they could cooperate with the project, but after a trail period with the framework they both decided they didn’t have the expertise. Eventually two developers that can help with Dimera were found. This could mean in the future it could be troublesome to find the right developers to further develop the app if the current developers become unavailable.

1.7 Project Components

In this section, there is an in-depth description of the three main components of the project, Startup100, Dimera and Shop hunt. It looks at the past in the form of Startup100 the present in the Dimera Application, and the future ShopHunt website.

1.7.1 Dimera

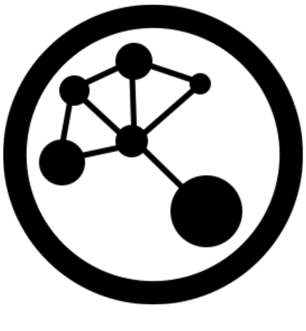


Figure 3: Dimera Logo

Dimera is the application that is used to create Startup100 and ShopHunt. It is short for digital media ranker. Dimera takes data from numerous data sources and ranks the data at timed intervals to produce up to date tables of ranked data. Dimera can be used to show which products, organizations or people are trending at any particular moment. To give the reader a better understanding of Dimera I will refer to everything that Dimera tracks as objects.
It can track any data that is made public on the web. This can include financial data, social media, and web page views or even when a certain thing is mentioned on the web. Dimera gives you two options; you can have tables that show what is trending. This means that it will display the objects in the table that have seen the biggest increase in whichever data you are tracking. So if you are tracking companies social media pages and website to see who is being viewed the most, while also tracking who is being mentioned the most in online media, it will show you which companies have had the most online activity within a particular time frame. Secondly, you can have the overall table displaying companies that have had the most online activity.

Every object that Dimera tracks has its own profile page where you can choose to display whatever data is necessary about that specific object. So if you were tracking companies, on their profile page there can be displayed, their position in the overall ranking, the company logo, a description of the company, links to their website and social media pages and a list of all the products they sell. The information on the profile pages can be adapted to display whatever information will be relevant to the objects that Dimera is tracking.

1.7.2 Startup100

Figure 4: Startup100 Logo

Startup100 was first established in 2013 to give San Francisco Oy an overall view of all the start-ups within Finland. The website was made by Skyhood which is a web development company in Helsinki that specialises in the Django web frame. The initial set up was made rapidly to cut costs, as they never knew if the idea would be successful. The website was originally made to aid San Francisco Oy marketing company, but soon gained interest from investors that were looking for new companies to invest in, and start-ups that want to promote their companies and see where they stand in the country compared to other start-ups. The main trending feature gained the most success as it gains interest from not just investors and start-up companies, but also people in the Finnish start-up community as a whole.

Startup100 has given the development team the beginning building block for the Dimera web application. It Tracks data from multiple data sources and gives the user the opportunity to track the top trending start-ups in Finland and the top start-ups in Finland overall. The way it analyses data will be the same for all other Dimera websites. The idea is to build on what is already in Startup100 so it can be used to track and analyses as many different types of data as possible. Startup100 has company profile pages where signed up users can view financial data on the start-ups along with a wide variety of other basic information about the company. Startup100 gives the start-ups the opportunity to advertise jobs on the page, and it is a paid service that uses an outside service Duunitori.

Because of the quick speed the original Dimera platform was built in, there are many bugs in the code, and it is also difficult for the site to run locally for further development and to install it onto other websites or servers. After the Dimera platform has been improved and upgraded with advanced features it will be integrated with Startup100 to improve the site.

The plan to upgrade the existing platform into Dimera, which will run smooth and have a more efficient install process, came about after other companies in other countries offered to pay for the platform. There is interest from multiple companies to build the application to track trending start-ups in Sweden, Denmark, and the Czech republic. When the Dimera application is complete it will be marketed internationally to track start-ups and whatever else companies, organisations or the public would like to track and rank.

1.7.3 ShopHunt

ShopHunt is a prototype website that is currently being built; the plan is to build it to test the Dimera app in a different scenario. The deadline for this project is not following the same timeline as the Dimera app, but is planned for completion sometime in the near future. Testing the Dimera application with a different data source than is being used for Startup100 will help us to gain an understanding to whether or not Dimera will work in various arrangements.

The idea behind ShopHunt will be to rank E-commerce stores within Finland and rank the stores so customers can find which stores are trending. The main difference between Startup100 and ShopHunt is that ShopHunt will not just rank the E-commerce stores by which company is best. But it will give the user the option to search for a product and the site will list all the stores selling that individual product, in a ranked trending order. We hope customers will find this beneficial because they will be able to find the best store to buy a product. We also hope this will give E-commerce stores a competitive edge, as each company will compete to climb the ranking list. It will also give small independent stores the opportunity to try making it onto the trending list, which will help them to gain more access.

Each store on the site will also have a similar profile page to the Startup100 website, but will focus more on the products that each web store sells. This will add a new feature to the Dimera platform because you will be able to see ranked and trending products as well as companies.

2 Methodology and research

The research techniques used in this thesis are correlation and regression analysis. It looks into the relationships between different objects, in this case how different software and technologies interact with each other. For the topic in question, Django, there are little published resources available. This is because Django is community run and most of the resources available come in the form of starter tutorials. This research covers reasons why and why not. Why to use Django and why not to use Django. Then it looks into how certain things can be accomplished.

2.1 The Purpose of Dimera

The purpose of Dimera is to rank trending digital media. The purpose from the investor’s point of view is for the Dimera application to create income by becoming a sellable product that will generate profit.

The Dimera applications can be used to rank any analytical data as long as the data can be found publically on the web. Most social media websites and browser analytics is open to the public. Although if you want certain analytics from social media profiles a code will have to be handed over by the owner of that particular profile.

2.1.1 Dimeras target audience

For the Dimera application, there will be two target audiences. The first will be potential clients that will purchase the Dimera applications. The second will be the end user that will use the application to search for precise ranked data.

The clients that will purchase the Dimera application is expected to be companies that want a replication of the Startup100 platform for a different country. But it could also be companies that feel the need to see trending analytics of different data types. Initially, the target audience is investors that can use the tool to find companies and products and see how the companies' data and ranking compare to other companies and products. Many companies see it as a valuable asset to see analytical data about their particular market to give them a competitive edge.

As Dimera grows and is further developed to be less costly to install onto a website the demographic may change. Bloggers could see value in the application to rank their favourite sports stars, celebrities or fashion objects to see who or what is trending. People tend to be very interested in seeing data ranked in order from best to last.

The target audience for a website made with Dimera will vary widely because it's up to the client to decide what data to rank.

2.1.2 Marketing Dimera

Marketing Dimera will be the responsibility of the San Francisco Oy team. They will have to adopt the best techniques to market the application, focusing on the correct target group.

The Dimera application will be marketed using a WordPress website. The website will have a focus on getting the functionalities of the Dimera application across to the potential client. Along with the WordPress website, there will be social media pages like Facebook, Twitter and LinkedIn. Using the web to market the Dimera application will be the cheapest and efficient method. On the release date of the Dimera application, a press release will also be sent out to all of San Francisco Oys journalist contacts.

2.2 Technical Research

This section looks into the technology that the Dimera app runs on, Django. It looks into reasons why developers would use Django, and what disadvantages and advantages the technology offers. Dimera currently has issues when installing and running on the hosting environment and the local virtual environment. It will also examine the possible techniques to achieve an easier install process.

2.2.1 Building Dimera with Django

The initial reason for building the app with Django is because Django packages and libraries simplify the development process and save the developer time in delivering the final product. Django has a rich library of pre-made tools so the developer doesn’t have to reinvent the wheel.

2.2.2 Technical Advantages of Django

After Researching the benefits and advantages of Django it usually comes back to one thing, and that's not that it's an advantage to use Django, but it is an advantage that Django is built using Python. Python is considered an easy language to program with. This is because you can create more functions with fewer lines of code.

The main advantage of Django is that it delivers a pre-made ORM, which allows you to concentrate on the user interface and less on the backend code, which comes in installable packages. It creates a light model, view, and controller. Unlike other frameworks that add a lot of bloated syntax, that will not be used in your final application. It lets you choose the exact requirements and plugins your want to use in the app. Because Django produces so little unused code, web applications assembled with the framework usually have cleaner code and are more human readable, which makes it easier to understand and edit for the developer.

All Django packages in the Django library are entirely free to the developer and there is a lot to choose from to save the developer from trying to reinvent programs that are already made. It's up to the developer to choose what fits their projects requirements best. When packages are chosen Django Packages usually come quite light, so as long as the developer knows what they are doing it is simple from them to edit each package into a unique applications to give some originality to each website.

Django also offers excellent search engine optimisation. This is because it delivers human readable URLs. It not only makes the URLs easier to read for humans but also for search engines as they usually search for keys words in each individual search done by a user.

Django offers a high level of security; this is because it hides all source code in the server then dynamically generates web pages. This means all the code in the server is translated into HTML and CSS so when the user tries to view it they will not see the original code but an interpretation made by the browser. Not even the original HTML or CSS will be viewable because it is stored in a secure template folder then translated over to the browser. When a user uses a developer tool to view the source code, they will only see an interpretation of what the browser believes to be the code.

Django Admin package delivers the developer a pre-made admin panel that can easily be made into a simple content management system, which can even be used to style and add any page and content with ease. This is great for when you hand the end website over to the customer because they can edit page content with very few technical skills.

2.2.3 Technical Disadvantages of Django

The initial disadvantage that became common for the Dimera project was finding good Django Developers. This is because there isn't a big following of Django developers in Finland. This is because more and more developers are training in easier languages to understand like PHP. This is because content management frameworks like WordPress and Drupal have become so favourable for so many websites in recent years.

A lot of developers consider Django to be old fashioned because the developer has to still make certain parts of a website that they wouldn't have to if they used a more modern framework. They tend to opt out of having more originality and better security for frameworks that focus more on styling and design.

A lot of modern developers don't have the knowledge to set up the backend requirements locally or on the server in order for a Django Application to run a browser. This is because a Django website requires you to build a lot in the backend for an application to run. This should give the developer more control over the final project. But developers often take short cuts by using PHP that will automatically run on any browser. This is a disadvantage for Django and Python who could consider updating their technologies to automatically run on a browser.

It is said that Django should give each developer the option to easily devise websites with originality, but its packages sometimes have the reverse effect as developers install packages and decide to keep them exactly the same. This is good if a developer wants something to be created at a quick pace, but because the Django community largely creates the Django packages, Django can at times fall behind other technologies. This is because Django progresses when the users give back to the community.

2.2.4 Pre existing technologies that can make Dimera easier to install, in a one-click install setup.

After testing the install process of Dimera with a variety of different of developers it became apparent that Dimera had a difficult and time-consuming install process. This is because most the developers questioned didn’t have previous experience with Django. An application becomes available to more people and is reused more if a wider population of developers can setup the application, and this is why it becomes important to Dimeras success to have a straightforward install action.

To get the Dimera application to operate successfully it comes down to package management. There are a few steps when running locally that must be taken. After download of the application, the developer will have to create a Virtualenv within the applications folder. This is where all Django and Python packages will be installed to prevent them from interacting with packages on the developer’s computer. Installing Python packages onto a computer could affect the way that other applications perform, so it is essential that the Python packages are isolated within a virtual environment.

The Dimera application comes with a requirement.txt file that contains all the Python and Django packages needed for the app to run. They will need to be installed onto the virtual environment in order for the application to work. Using Pip you can install the requirements text in one command. When the virtual environment is running insert the command “$ pip install -r requirements.txt” This will automatically install all the packages in one action. This comes with one issue, packages that are no longer supported or available for download. If this occurs you will have to install the packages individually to see which packages don’t perform. Once they are defined you can search for their update and install them instead. These issues are apparent to an experienced Python developer but if the developer is not, proper instructions or tutorials can be hard to find.

The next step will be to set up the PostgreSQL database locally and adjust the file paths so the application can run. Then the migrations will have to be applied to the Virtualenv.

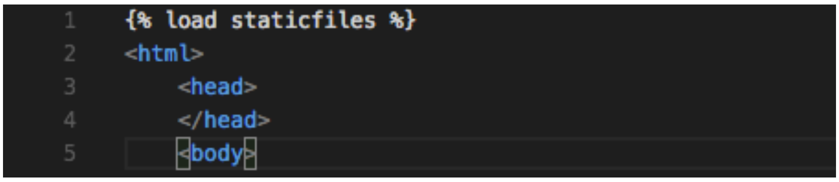
If issues arise you can edit the settings.py file to show you the trackback on the browser, which will locate the issue within your packages and files. This can be achieved by setting DEBUG = false to DEBUG = True in the settings.py file. This must be changed back to DEBUG = False before pushing the application live because it will display hidden file information on the browser.

The simplest way of making the install process understandable was to document actions and create a manual.

2.2.5 Integrating Django with other Web Frameworks

Django doesn’t integrate well with other prominent frameworks such as WordPress or Drupal. That's not to say that it's not possible, though. If a client was to ask for the Dimera application on their WordPress website, they could do it by setting up two separate hosting sites and insert the Dimera tables into a WordPress page. The problem is that it can only be done with two independent servers. WordPress and Django run on completely different programming languages, so to get a full integration can be problematic.

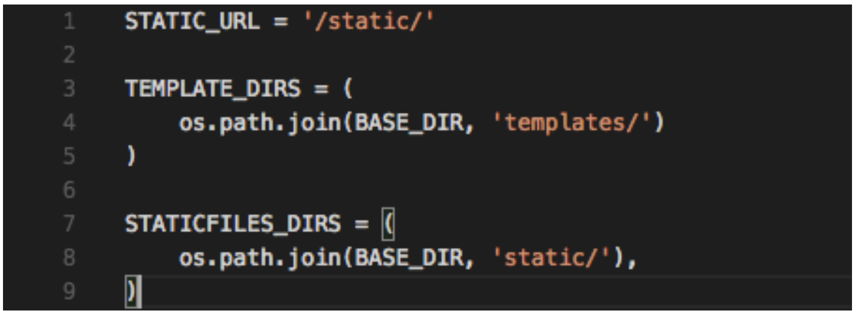
Although you may not be able to execute Django on frameworks such as WordPress and Drupal, that's not to say that other technologies can’t be integrated with Django. Because Django uses pure HTML files to display objects on the browser, it can make good use of technologies such as JavaScript which means you can use Node.js JQuery and AngularJS and so on. All of these technologies will run within isolated Django template files and Django static files. When placed in the correct folders the developer can insert the technologies into the HTML files using the correct file paths. All that is required after that is to load the static files into the HTML file by inserting this command:



```
{% load staticfiles %}
<html>
    <head>
    </head>
    <body>
```

Figure 5: Loading Django Static Files to HTML

As you can see the {% load staticfiles %} is inserted above the first <html> tag. This is because it is Django code and not HTML. The developer will then need to tell Django that the static files exist by declaring their path in the settings.py file. The same way all files need to be declared in the settings.py file.



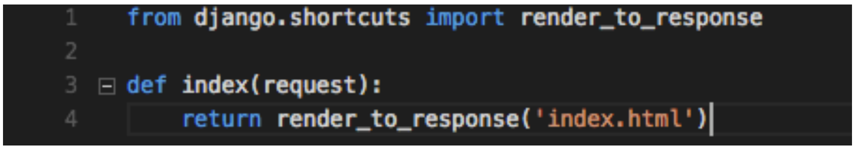
```
STATIC_URL = '/static/'

TEMPLATE_DIRS = (
    os.path.join(BASE_DIR, 'templates/')
)

STATICFILES_DIRS = (
    os.path.join(BASE_DIR, 'static/'),
)
```

Figure 6: Declaring Templates and Static Files Path

Then all the developer needs to do is rendered the HTML file within the views.py file; this is so Django knows to display it on the browser. Inserting the following code into the views.py file can do this:



```
from django.shortcuts import render_to_response

def index(request):
    return render_to_response('index.html')
```

Figure 7: Rendering index.html Page For Browser

After this is set up HTML files will work in the same way as they would in any other frameworks with the added benefit that you can use Django packages and commands. It must be remembered that each HTML file will need to be rendered within the views.py file in order for it to run.

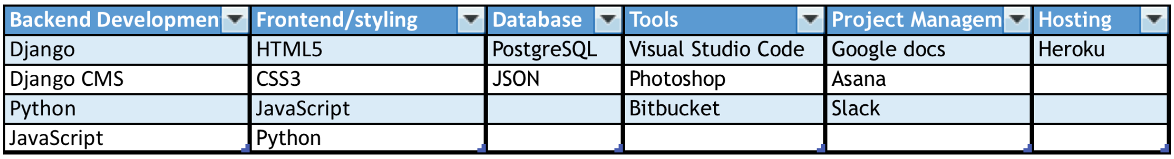
Because of the static folder, it gives the Django developer a lot of flexibility when choosing which technologies they would like to use within their web application. As long as it works with HTML it should work with Django.

3 Project Requirements

To run any web application or software there can be a huge amount of resources and requirements needed. Section Three of this report lists all resources and requirements need for the Dimera Project.

3.1 Technology Used

In this section of the report, there is a descriptive paragraph taken from reliable sources that are leaders in all the programming languages and technologies used within the project and necessary in order for a Dimera website to run locally and live. There is a short description and background of the technology, followed by an in-depth description on how the technology applies to the project.



Backend Developmen	Frontend/styling	Database	Tools	Project Managem	Hosting
Django	HTML5	PostgreSQL	Visual Studio Code	Google docs	Heroku
Django CMS	CSS3	JSON	Photoshop	Asana	
Python	JavaScript		Bitbucket	Slack	
JavaScript	Python				

Table 2: Project Tools and Technologies

3.1.1 Django - Web framework

Figure 8: Django Logo

Django is a Python Web framework that was built to encourage fast development and clean, pragmatic design. It is free and open source. (Django Software Foundation. 2016)

Dimera uses Django as its main web framework. Django gives the developers the possibility to create anything that is requested with only the imagination and skill levels of the developers as a limitation. It also gives the developer access to a large library of pre-made tools to speed up development.

3.1.2 Django CMS - Content Management System

Figure 9: Django CMS Logo

For the Dimera project, we aren't using an outside source Django CMS. This is because the default Django Admin fulfils all the CMS requirements needed for the project. It is also easily edited to function and look the same as the Django CMS's on the market. The Django Admin is easy to install and edit to fill your requirements and is also free. If a client was to find a CMS that they like the look of it can easily be installed at a later date.

The Django Admin interface used on Dimera is for the site administrator to authenticate sign ups to the sites, and to also add now companies of products along with all their data to the site.

3.1.3 Python - Backend Development

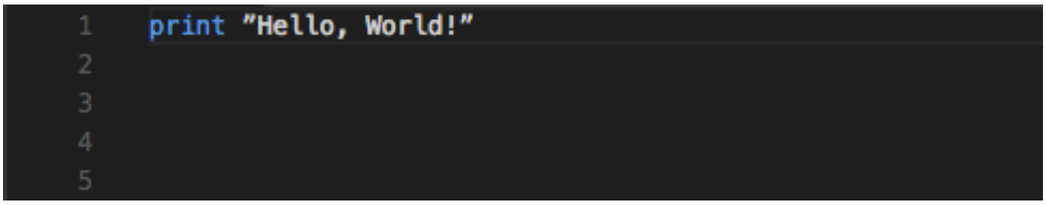
Figure 10: Python Logo

Python is an object-oriented, high-level coding language, which uses dynamic semantics. It is particularly useful for speedy application development because of its high-level data structures and dynamic typing and binding. Python also has a simple syntax, which emphasises readability, which reduces cost in the development and maintenance process. (Python Software Foundation. 2016)

Django uses Python as its main coding language; this is why it has to be used in the Dimera application. Python is the backbone of the whole application. It connects the database to the front-end. It also tells the database and the front-end which way it is supposed to function. Python is used to manipulate the data in the database to produce the ranked and trending tables. It is the communicator between the front and back-end.

Python is somewhat simpler than other languages because the code its self is lighter than some of its counterparts like Java and C++. In the below figure there is written three programs in Python, Java and C++. They all function the same and print the end result of: Hello, World!

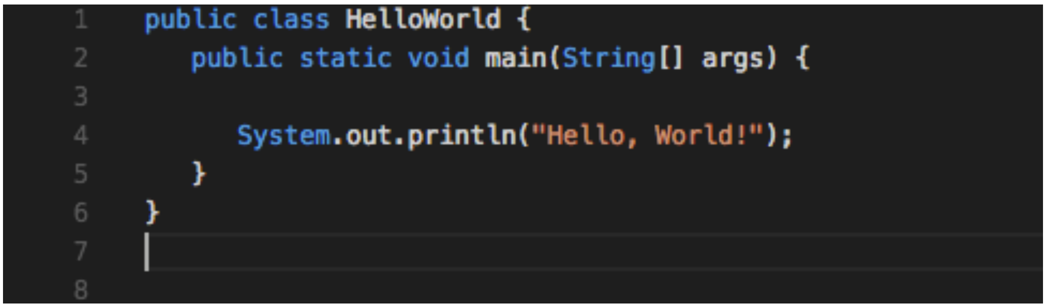
Python:



```
print "Hello, World!"
```

Figure 11: Python Print Syntax

Java:

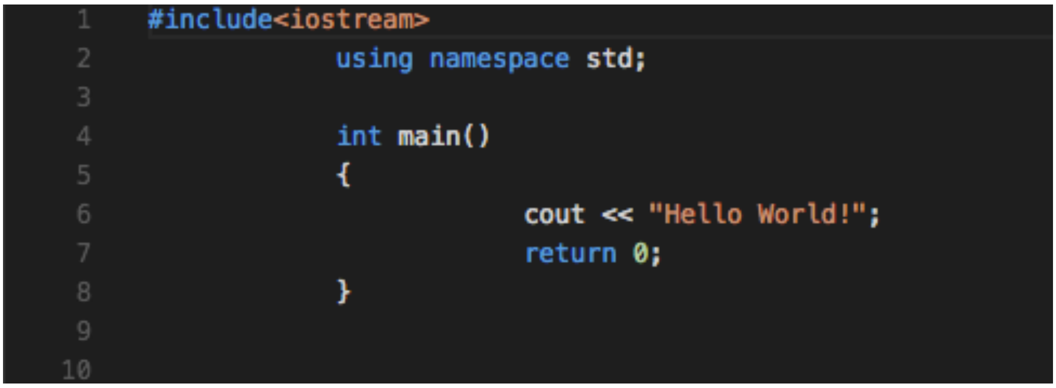


```java
public class HelloWorld {
   public static void main(String[] args) {

      System.out.println("Hello, World!");
   }
}
```

Figure 12: Java Print Syntax

C++:



```cpp
#include<iostream>
            using namespace std;

            int main()
            {
                        cout << "Hello World!";
                        return 0;
            }
```

Figure 13: C++ Print Syntax

As can been seen from the figure above it is clear that Python is shorter. Python may be shorter but can still be used to function in the same ways as other programming languages. This can save time and resources when working on a project because the code isn't syntax heavy.

3.1.4 PostgreSQL – Database

Figure 14: PostgreSQL Logo

PostgreSQL is an open source relational database management system. A team of volunteers that offer the technology free of charge develops it. It has been developed to handle large quantities of data that make it useful for the development of any project. (Rouse, M. 2006)

PostgreSQL doesn't differ much from other SQL technologies but was chosen for this project because it has the ability to handle large amounts of data. PostgreSQL was designed to handle a high volume of data. PostgreSQL can also save money because it is free to use with no licensing costs, which will stay the same no matter how big a database becomes.

3.1.5 Heroku - Cloud Server

Figure 15: Heroku Logo

Heroku is a cloud platform server system, which offers integrated data services and a powerful ecosystem. It is used to deploy and run applications. (Salesforce. 2016)

Heroku is a cloud server where Startup100 is running live; we also use Heroku to run test sites on during the project. Heroku uses Git to directly communicate with private local virtual environments to make it simple and quick to upload full applications to the web. It has a nice interface that allows you to see all the metrics and analytics of a running application. Although the Dimera teams use Heroku to host their sites, it will be completely up to the client to decide where they want to host their sites. But they will have to find a host that supports Python 3.5 in order for the Dimera application to run. This is a complication that will hopefully be changed in the future when the application becomes more reusable.

3.1.6 HTML5 – Site Layout

Figure 16: HTML5 Logo

HTML is an abbreviation of Hyper Text Mark-up Language. It is used as the standard mark-up language for developing web technologies. (Refsnes Data. 2016)

HTML is used to build the structure of content and objects on all web pages. To display any Python, database or JavaScript objects and data on a web browser it will have to be inserted and displayed in an HTML file. PostgreSQL may be storing the data and Python will be manipulating the data and giving it rules, but to display it on a web page in a table it will still need to use an HTML table tag with CSS styling to do so. For the Dimera websites, we have used a base.html file that will be inserted into every HTML page on the web app. the Base file contains the navigation menu, the page header, and footer bar. This ensures that they will be the same on every page adding consistency; it saves us time in the development process so the developer doesn't have to edit every HTML file when a change is made to the header or footer. For the Dimera platform, it was decided to make the HTML files as simple as possible so they are easy to develop into any layout a client would request.

3.1.7 CSS - Front-end Development and Styling

Figure 17: CSS3 Logo

CSS is an abbreviation of Cascading Style Sheets. CSS creates rules for how HTML elements are displayed and styled on to a screen, paper or other media. (Refsnes Data. 2016)

CSS3 is the backbone of all styling done on the websites. It directly tells HTML content how to display and act. CSS tells objects how to act in the front-end only and doesn't add many functions to the site apart from styling. In the Dimera application we use three CSS pages to give different styling for different sized screens, this is to ensure maximum optimisation between device platforms. The CSS page being used will be changed depending on the size of the devices screen size. Dimera is optimised to use a different CSS file for Desktop computers, tablet devices and smartphone devices.

3.1.8 JavaScript - Front-end Development

Figure 18: JavaScript Logo

JavaScript is an agile, interpreted, object-oriented programming language that uses first class functions. It usually is used to run on the client side of the web and is used to design how web pages should behave after user events. (Mozilla Developer Network and individual contributors. 2016)

JavaScript is only used partially in the front-end of Dimera apps. Its gives the developer the opportunity to give animation to web pages. It is fundamental to give a page the most up to date styling techniques on the web. It is a great tool for manipulating objects on the front-end of the web pages. If used effectively it can make the web page more interactive for the user and grab their attention.

3.1.9 JSON - Test Database

Figure 19: JSON Logo

JSON stands for “JavaScript Object Notation”. It is a data interchange format, which has been created to make data easy for developers to read and right. (JSON. 2016)

JSON is only used in the development phase of creating a Dimera web application, in the local environment. While styling a web page it is easier to see the final outlook and check for display bugs when the data tables and profile pages are full of data. JSON was used to create a large amount of fake data so the developer can see what the final web application will look like when it’s full of data. When the application is complete and goes live on a site JSON will no longer be required.

3.1.10 Visual Studio Code

Figure 20: Visual Studio Code Logo

Visual Studio Code is an efficient, agile source code editor developed by Microsoft. It is free to download and is open source. It is a useful tool that keeps development environments clean and efficient. (Visual Studio Code 2016)

All Dimera source code is edited using Visual Studio Code. The tool is free and very straightforward to use. It can handle a wide variety of programming languages that makes it the perfect tool for developers. The text editor is considered lightweight compared to other source Code editors on the market. It gets straight to the point and lets you start working on your code with no confusion or unnecessary features. Other source code editors may have some beneficial features like they might suggest code to the developer as they type. But if you feel comfortable working with a language it isn't needed.

3.1.11 BitBucket - Git repository

Figure 21: Bitbucket Logo

Bitbucket is a free web-based hosting service for projects using mercurial or Git. Bitbucket is popular in the Django community because it is built with the same Python Framework. (Bitbucket 2016)

When starting the project Github was used as the Git repository for the project. When the team took on a new Django developer he made the suggestion to move to Bitbucket. Bitbucket and Github work in pretty much the same way but Bitbucket is the preferred tool in the Django community because it is built with Django and Python. Bitbucket is also ideal for small teams because it is free to groups of up to eight people.

3.1.12 Photoshop - Image Editing

Figure 22: Photoshop Logo

Photoshop is a popular Image editing software that owned by Adobe Systems Inc. The software allows the user to manipulate digital photos in a wide variety of ways, such as crop, resize, colour changes etc. (WebFinance Inc. 2016)

Photoshop has been used throughout the project for all images editing and formatting. It plays a major role in styling the sites so the end product is not just functional but also pleasing to the eye. A website that is presented well gives the user a good first impression and will open up the user's curiosity.

3.2 Project Management Tools

With every project, management is essential to make sure the project meets its objectives, timeframe, and budget. To make sure the demands were met for the Dimera project a series of tools were used for scheduling, documenting, planning and communicating. Although the tools listed in this section don't aid in creation and development of the Dimera application they are still fundamental for an organisation to produce the best possible outcome.

3.2.1 Google Docs - Project documentation

Figure 23: Google docs Logo

Google docs are a web-based word processor, which manages and stores the user's documents on the cloud. It allows users to write and edit documents on the web browser from any Internet connection. (Techopedia Inc. 2016)

Google is the word-processor of choice for the San Francisco Oy and Dimera Team. It is used for all documents. Project plans for the project were written on Google docs and shared with the team. Google docs are also used for the creation of all the content that is displayed on the front-end of all Dimera websites. Because Google docs are directly connected with the team's email accounts it becomes effortless to share docs with team members and offers a function that lets you see which teams members have written each part of the doc or if anyone edits any parts of the doc.

3.2.2 Asana – Project Schedule

Figure 24: Asana Logo

Asana is an application created by Facebook co-founder Dustin Moskovitz, which is designed to help teams track and manage their work. (Asana (software). 2016)

Asana is used by the Dimera project team to create an overall schedule and milestone list. Using Asana the team is able to see what tasks are complete and which tasks need to be done. Tasks can also be assigned individually to team members and with a deadline that needs to be followed. The app was helpful during the project because it would send notifications to phones when a task's deadline is approaching.

3.2.3 Slack – Communications

Figure 25: Slack Logo

Slack is a cloud-based communications tool, designed for team collaboration. (Slack (software). 2016)

Slack is the main tool of communication for the Dimera Project. It became useful because you could send personal messages as well as create group chats to share information and files. Slack offers a service that makes it efficient for keeping communications organised because you are able to make a separate chat group for each subject or task. Meaning communication channels don't become overcrowded with different material.

4 Project Implementation

For this section of the report, it will go into what was done for the Dimera project as a whole. It will not go into detail about the exact code itself because some of the material is a company secret. It will explain what processes were complete during the project but not so much into how each process was made. This is to ensure that Dimera retains its competitive edge by holding its secrets.

4.1 Project Planning

Below is a quick overview of the project plan. Not all completed tasks have been noted because of the way the project has evolved. At times tasks have been issued on the fly and completed at a quick pace. This is why there is only noted down the main tasks assigned by the project manager

Goals:

- Create Dimera platform
- Launch the E-commerce site (ShopHunt)
- Update S100
- Document Process

Dimera:

- Extract the application from Startup100
- Update the app with the latest technologies like the latest version of Django and Python 3.
- Clean and document code
- Simplify the install process
- Debug and test the application
- Simplify Django data entry process, in order to make the application able to handle different data types.
- Add new ranking calculations.

- Automate company, product and user registration.
- Create a manual about the Dimera set up.

Startup100

- Apply the new updated version of Dimera to Startup100
- Modernise the websites front-end.
- Improve company search methods
- New and improved features on company profiles
- Allow users to view trending companies in tables of ranked companies by category. E.g. banking and finance, business software, travel, games etc.
- Fix all bugs, clean code and document.

ShopHunt

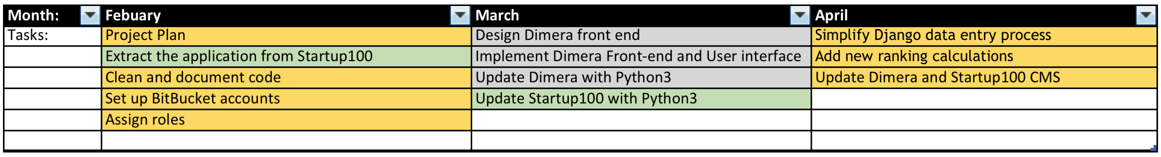
- Design site layout
- Create company identity
- Build front-end of the site and user interface.
- Implement the Dimera application into the ShopHunt website and set it to rank the correct data sources.
- Set up back-end for ShopHunt website
- Collect and install E-commerce data for the initial launch of the website.

We are now at the closing stage of the first two stages of the project, Dimera Application development and Startup100 improvements. Development of the ShopHunt website has begun and the first prototype has been created. Although the ShopHunt part of the project runs on a different schedule and is set to be complete in the near future.

4.2 Project Timeline

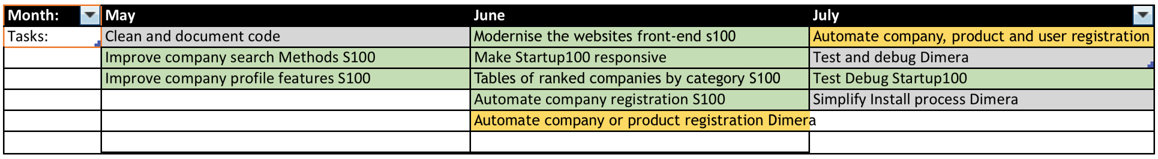
The Dimera project started in February 2016 the project hasn't had a strict timeline to follow but has more worked towards getting everything ready to present to the investor before November 2016. Tasks were assigned through Asana, some of which had a completion date. Many tasks did not have a recorded completion but a time was usually verbally agreed upon as the project developed and evolved.

Below is a list of tasks that was completed during the implementation of the Dimera project. The tasks are put into months of completion according to dates found on Asana. Start dates for each task are unavailable because Asana hasn't been set to record those dates.



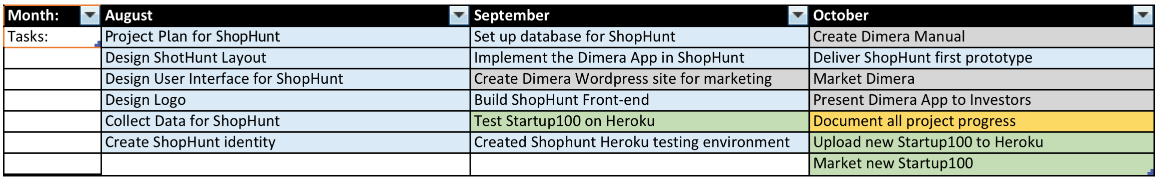
Month:	Febuary	March	April
Tasks:	Project Plan	Design Dimera front end	Simplify Django data entry process
	Extract the application from Startup100	Implement Dimera Front-end and User interface	Add new ranking calculations
	Clean and document code	Update Dimera with Python3	Update Dimera and Startup100 CMS
	Set up BitBucket accounts	Update Startup100 with Python3	
	Assign roles		

Table 3: Timeline, February, March, and April



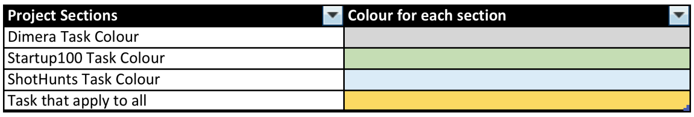
Month:	May	June	July
Tasks:	Clean and document code	Modernise the websites front-end s100	Automate company, product and user registration
	Improve company search Methods S100	Make Startup100 responsive	Test and debug Dimera
	Improve company profile features S100	Tables of ranked companies by category S100	Test Debug Startup100
		Automate company registration S100	Simplify Install process Dimera
		Automate company or product registration Dimera	

Table 4: Timeline, May, June, and July



Month:	August	September	October
Tasks:	Project Plan for ShopHunt	Set up database for ShopHunt	Create Dimera Manual
	Design ShotHunt Layout	Implement the Dimera App in ShopHunt	Deliver ShopHunt first prototype
	Design User Interface for ShopHunt	Create Dimera Wordpress site for marketing	Market Dimera
	Design Logo	Build ShopHunt Front-end	Present Dimera App to Investors
	Collect Data for ShopHunt	Test Startup100 on Heroku	Document all project progress
	Create ShopHunt identity	Created Shophunt Heroku testing environment	Upload new Startup100 to Heroku
			Market new Startup100

Table 5: Timeline, August, September, and October



Project Sections	Colour for each section
Dimera Task Colour	⬜
Startup100 Task Colour	🟩
ShotHunts Task Colour	🟦
Task that apply to all	🟨

Table 6: Timeline Colour Code

4.3 Project Development

This section of the report will go into the development of the Dimera application. It will take a look into each development phase of setting up and developing the application.

4.3.1 Backend Development

The Majority of the application Dimera is back-end code. For this project, we updated the whole application that was taken from Startup100 and updated it with the latest technologies. This includes the latest version of Django and the latest version of Python. Mostly the application didn't need a lot of work doing to it in the backend because it was already created. It just needed further tweaking so the application could handle different data types. For instance, the Startup100 website ranks companies, but the ShopHunt website ranks companies and products. A new source of data had to be discovered, because stores usually have all the running social media accounts and website, but products usually do not.

One distinct feature of Dimera is the company sign up process. It now offers companies more interactivity because they are able to edit their own profile pages within Dimera applications. Before companies had to contact the website admin about new updates to their profile pages, but now companies can ask for login credentials for their profile pages and update it themselves. To ensure vandalism isn't done by the companies the website admin will get a notification every time and an update is made. The website admin will then have to verify the data before it goes live to the site. User interaction has been further developed by allowing new companies to register on the website. Before companies would have to fill in the Google form in order to get on the site. Then the website admin would have to enter the new company and its data, onto the site manually. This was quite time-consuming and not cost-effective. So now the companies can enter their data into the registration form, and all the admin needs to do is verify the company and it will automatically add all the new data.

Dimera uses the same process as Startup100 to rank the data, and produce analytical data. This is why the application took a short time to develop. The preliminary set up of the Dimera application has been refined to not have as many requirements as Startup100. The fewer requirements for the Dimera application, the fewer the updates needed to be installed every time a Dimera project is started.

4.3.2 Database Development

For the development of the database, it will need to be pulled from Heroku and pushed onto the developer's local machine. PostgreSQL will need to be running on the local machine. The data itself can then be edited and viewed using a variety of PostgreSQL applications. For this project, we used Postico for database development. You will have to set the path on a Virtualenv so the Dimera application can find the data. The data will then need to be migrated into the app. Once the application is up and running locally, data can be easily installed into the database by using the Dimera Admin page on the browser.

4.3.3 Front-end Development

Dimera comes with easy HTML and CSS file templates that can easily be edited. It was designed so the developer can quickly set up a basic prototype of a Dimera site. It includes a base.html file, which has a premade template that holds a simple footer, header, navigation, table styling and a page content tag. This base.html file is then inserted into the index.html page using Django syntax {% extends "web/base.html" %}. This base.html file then can be inserted into every HTML file that is created for a Dimera site to speed up development and to make sure each web page gets the same styling. Within the Dimera application, you will find a static folder, which contains three other folders. JavaScript, Img and CSS. All folders are empty apart from the CSS folder that contains three CSS files. One is called base.css and is used for all the general CSS used on the base.html file. There is then admin.css, which contains lots of CSS, which is the interface for the Dimera application Content management and administration pages. The third CSS file is style.css, which can be used for styling any aspect the developer, chooses to work with. The idea behind the three CSS files was to give the developer organisation. All CSS files are then inserted into the base.html file so they will not have to be added to any of the other HTML files.

Although there is a lot of code on the front-end of Startup100 and ShopHunt, the front-end of Dimera has been kept simplistic. The idea behind this is to give the developer a fresh, easily manageable beginning whenever a new Dimera application starts. This will ensure that all Dimera applications have originality and their own identity. If a client was to ask if they could make and manage Startup100 for a different region and wanted it to use the same styling it wouldn't be too demanding on the developer to insert the Startup100 HTML, CSS and JavaScript files into the Dimera application.

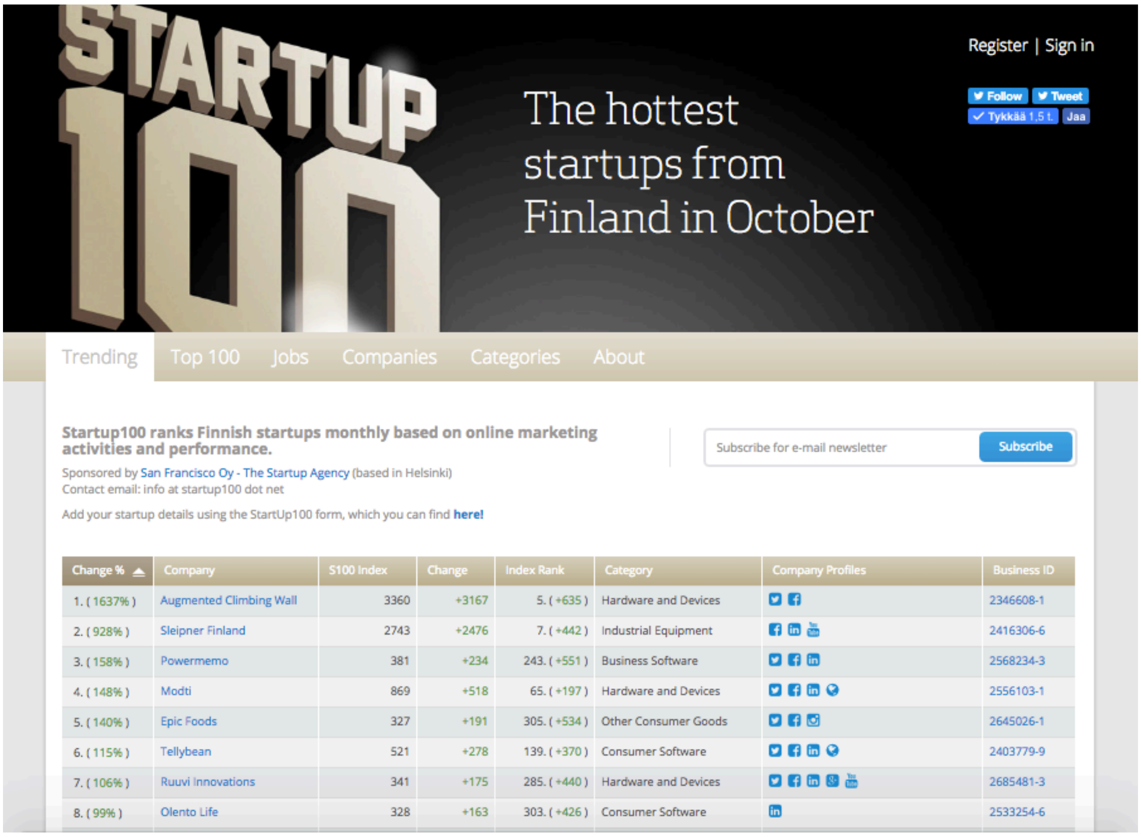
All HTML pages that are created have to be declared in the settings.py file and then rendered in the views.py file in order to run on the browser, but apart from that working on the front-end of Django is similar to other technologies.

4.3.4 User Interface Development

Dimera uses clean HTML5 and CSS3 templates for its user interface, this gives the developer freedom when creating a new Dimera based website. The style and layout of the website will be decided by each individual client of Dimera. Giving the user the option to choose their own user interface gives the Dimera app flexibility. Dimeras UI is constructed with HTML5, CSS3 and JavaScript, but it will also give the client the option to choose which technology they prefer.

One of the main goals, when it came to developing the user interface of Dimera was to make sure that the application is responsive to all screen sizes and browsers. Updating the front-end of Startup100 to be responsive was the first task performed on the front-end of Startup100.

Below is pictured Startup100 and ShopHunts interface to show the possible interfaces of Dimera.



The hottest startups from Finland in October

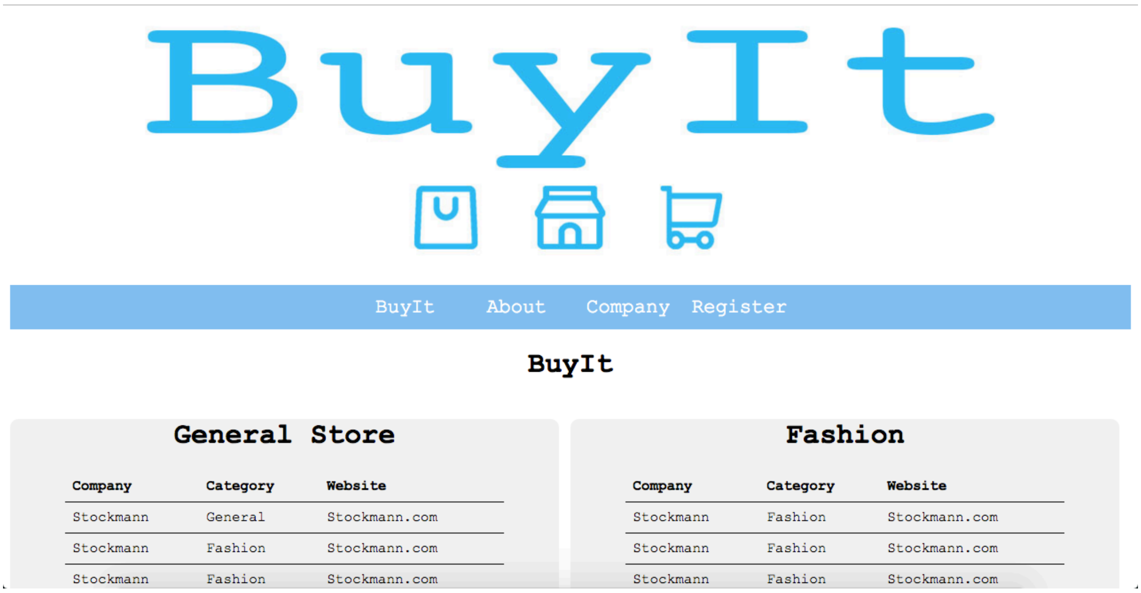
Startup100 ranks Finnish startups monthly based on online marketing activities and performance.

Sponsored by San Francisco Oy - The Startup Agency (based in Helsinki)
Contact email: info at startup100 dot net

Add your startup details using the StartUp100 form, which you can find **here!**

Change %	Company	S100 Index	Change	Index Rank	Category	Company Profiles	Business ID
1. (1637%)	Augmented Climbing Wall	3360	+3167	5. (+635)	Hardware and Devices		2346608-1
2. (928%)	Sleipner Finland	2743	+2476	7. (+442)	Industrial Equipment		2416306-6
3. (158%)	Powermemo	381	+234	243. (+551)	Business Software		2568234-3
4. (148%)	Modti	869	+518	65. (+197)	Hardware and Devices		2556103-1
5. (140%)	Epic Foods	327	+191	305. (+534)	Other Consumer Goods		2645026-1
6. (115%)	Tellybean	521	+278	139. (+370)	Consumer Software		2403779-9
7. (106%)	Ruuvi Innovations	341	+175	285. (+440)	Hardware and Devices		2685481-3
8. (99%)	Olento Life	328	+163	303. (+426)	Consumer Software		2533254-6

Figure 26: Startup100 Desktop Version



BuyIt

General Store

Company	Category	Website
Stockmann	General	Stockmann.com
Stockmann	Fashion	Stockmann.com
Stockmann	Fashion	Stockmann.com

Fashion

Company	Category	Website
Stockmann	Fashion	Stockmann.com
Stockmann	Fashion	Stockmann.com
Stockmann	Fashion	Stockmann.com

Figure 27: ShopHunt Homepage, Desktop Version

The Interface for ShopHunt and Startup100 worked well on the desktop but not so well on mobile. This is because on the small screens the menus were to narrow and became difficult to smoothly interact with. To overcome this a completely different menu was applied for mobile. This was accomplished by applying different styles to the CSS file for different screen sizes making the site responsive.

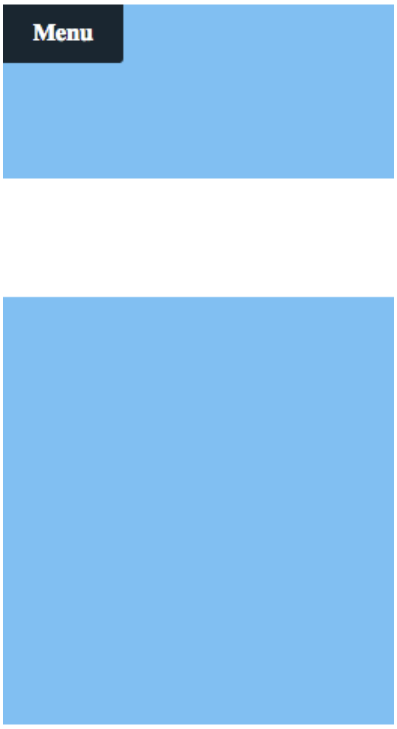


Figure 28: ShotHunt, Mobile Menu Inactive

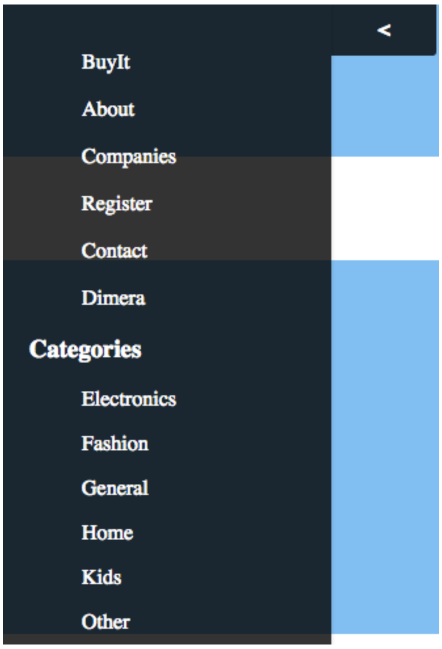


Figure 29: ShopHunt, Mobile Menu Active

The mobile menu uses simple JavaScript to allow the menu to appear hidden or in view. This greatly improves usability and navigation throughout the site, the menu tab will stay in view on the top left of the mobile screen at all times.

4.4 Testing

Testing for the Dimera Application was done both on the local virtual environment and then live on a Heroku test server. Everything is checked for bugs; data retrieval is checked to see if it is being ranked correctly. All links and styling were then checked on all the major browsers and different screen sizes.

The San Francisco Oy Website checklist was also used as a guide. The checklist is presented below

Code:

- Site is responsive
- Site is compatible with the major browsers
- Site code is error free
- The is no broken link

Visibility:

- Domain redirection
- There is a sitemap
- There is robot.txt
- There is a favicon and Apple icons
- There are social media metadata tags
- There are no spelling errors
- All pages are optimised for the keywords

Analytics:

- There is a Google Webmaster tool account
- There is GTM or analytics script installed
- There are views and filters
- There are goals
- There are Google RLSA (Remarking lists for search ads

Adwords:

- There are Campaigns
- There are ad groups
- Ads are optimised for keywords

The development team will only be focusing on the first two sections, code and visibility. Analytics and Adwords will be the responsibility of the San Francisco Oy marketing team.

4.5 Platform Management

Platform management has been assigned to the project manager Semyon Leshchenia and San Francisco Oy employee Olli-Joonas Juuso. It is their responsibility to manage all data through the Content Management System. This responsibility includes checking all companies and products on the Startup100 site and ShopHunt site are up to date and correct. After a new company registers to the websites it with be their responsibility to verify the data before it will be delivered to the live website to be ranked by Dimera.

After the sites have been fully complete it will be their responsibility to manage the site and report to the developers if issues arise.

4.6 Hosting

All Dimera websites owned by San Francisco Oy are hosted on Heroku. Prospective clients will be able to choose their hosting environment, but Dimera will still suggest that they should host on Heroku because it supports Python 3.5.

To deploy the Dimera application on Heroku firstly the developer will need to install the Heroku tool belt on their local machine. Then when they cd into their application folder locally they will have to create Heroku within the application folder by running the command: create heroku, they can then upload the application by running the command: git push heroku master. This will then push all the code into Heroku. The application needs a Procfile to tell Heroku how to run the application. Dimera comes with a Procfile included to ease this process although it may require changing depending on how the developer modifies certain aspects in Dimeras code. From here the developer will have to set up Dimera on Heroku in the same way they would set up Dimera in their local environment. This includes installing all the Django packages from the requirement.txt file, which is achieved by running the same command as the developer would run while creating the Virtualenv, which is, pip install -r requirements.txt, but change pip with heroku so the command changes to: heroku install -r requirements.txt. Then the database will need to be migrated to Heroku.

The process of uploading the application can be challenging and not always so straightforward. So during the project, it was of utmost importance to document the process. There is a step-by-step guide to installing and running the Dimera application on Bitbucket. This may not be convenient if the developer decides to change any of the code or Django packages found on Dimera during the development stage of a new website.

4.7 Marketing

Although marketing doesn't fit under the scope of the project and this task will be the responsibility of the San Francisco Oy marketing team, it is still a major player in the Dimera applications success. It will be of utmost importance for the team to aim their marketing at the correct audience.

5 Evaluation

The Dimera application is now ready to be marketed to potential clients, the San Francisco Oy team will execute this task. It has been delivered to the investors along with the digitalmediaranking.com WordPress website which will act as the main hub for marketing the product.

So far the project has been successful and the two main objectives have been concluded, which are Dimera and Startup100. The Dimera application has been completed, and is already being marketed to potential clients. Startup100 has been updated with the Dimera improvements and adopts the latest technologies. Testing of the Dimera application in different scheme is under production in the form of ShopHunt. All tasks have finished within the allocated timeframe.

The Dimera application complied with the goals of becoming a reusable data ranker, which is able to handle and analyse a wide variety of digital media data. Dimeras anticipated success now lies in the hands of the San Francisco Oy marketing and sales team. Return on investment will be achieved when potential clients have purchased the application. There is also a plan to gain capitol through online marketing when the new ShopHunt website is released.

Future development on Dimera will include searching for a solution to making the Dimera application installable in one click. This was a side goal of the project that wasn't fully achieved. The install process was made understandable through logical documentation and tutorials of the Dimera application. But to ease and speed up the development process on new Dimera based websites will save both time and resources if the Dimera application is to become a success.

6 Conclusion

On the first of November, the results of this project will or have been delivered to the investors. It includes a fully developed Dimera Application, improvements and new features to the Startup100 website which utilises the new and updated technologies used in Dimera, and the new prototype website ShopHunt which runs on Dimera. The project will now be moved on to the San Francisco marketing team for them to sell the application. Further development will be done to the ShopHunt website as it is still under construction and will be released to the public in the near future. All testing and debugging has been completed on the Dimera application, but it will continue to be monitored in the future because only time can uncover certain issues. Documentation has been complete and can be downloaded after a purchase of the Dimera application along with guides on how to install the application and how to run the admin interface. The Application will go on sale with two sales brackets. One for the use of the application on your website, but with no viewing right of the code meaning the application will be running on San Francisco OY owned servers. The other price range will be for full developer mode where the code can be purchased and developed in whichever way the buyer decides. This option will be more expensive unless they choose to franchise the Startup100 brand within a different country of their choice.

References

Asana (software). 2016. Last modified 15 October 2016. Accessed 23 September 2016.
https://en.wikipedia.org/wiki/Asana_(software)

Bitbucket. 2016. Last modified 16 September 2016. Accessed 20 September 2016.
https://en.wikipedia.org/wiki/Bitbucket

Django Project, Homepage. 2016. Accessed 20 September 2016.
https://www.djangoproject.com/

JSON. 2016. Introducing JSON. Accessed 12 October 2016.
http://www.json.org/

Mozilla Developer Network and individual contributors. 2016. What is JavaScript? Accessed 12 October 2016.
https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript

Refsnes Data. 2016. CSS Introduction. Accessed 24 September 2016.
http://www.w3schools.com/css/css_intro.asp

Refsnes Data. 2016. HTML Introduction. Accessed 24 September 2016.
http://www.w3schools.com/html/html_intro.asp

Rouse, M. 2006. PostgreSQL. Accessed 21 September 2016.
http://whatis.techtarget.com/definition/PostgreSQL

Salesforce. 2016. The Heroku Platform. Accessed 24 September 2016.
https://www.heroku.com/platform

Slack (software). 2016. Last modified 24 October 2016. Accessed 23 September 2016.
https://en.wikipedia.org/wiki/Slack_(software)

Techopedia Inc. 2016. Definition - What does *Google Docs* mean. Accessed 23 September 2016.
https://www.techopedia.com/definition/13626/google-docs

Visual Studio Code. 2016. Last modified 22 October 2016. Accessed 20 September 2016.
https://en.wikipedia.org/wiki/Visual_Studio_Code

WebFinance Inc. 2016. Photoshop. Accessed 24 September 2016.
http://www.businessdictionary.com/definition/Photoshop.html

What is Python? Executive Summary. 2016. Accessed 20 September 2016. https://www.python.org/doc/essays/blurb/

Figures

Tables