Software Requirements Specification

version 1.0

Kinana Joudeh
Reema Riyad
Muath Juady

Supervised by:
Dr. Abdalrazzaq Natsheh

A project report submitted in partial fulfillment of the requirements for B.Sc. degree in Computer Science

An-Najah National University
2019 / 2020

Table of Contents:

1. Introduction

This document is a Software Requirement Specification (SRS) for the DyNotify Web Application. This is the initial draft for the SRS and it will be used for the extensions. This document is prepared by following IEEE conventions for software requirement specification.

In the age of the world wide web, people are given many options from where to purchase a product or book a service. Big and small e-commerce websites and services providers are constantly competing to acquire and retain customers of all interests.

Products for example, may be available through many sites (e.g: Amazon, eBay, AliExpress, ...etc). With the large number of sites, comes a huge number of choices that may exhaust the user. In other words, the time and effort needed to search each one individually for the best deal adds an extra layer of inconvenience. Therefore, this product aims to act as a search engine that optimizes the search process, and return relevant results from multiple sources.

This same product provider may offer discounts and deals on certain occasions, but it's rather inconceivable to customers to check every now and then for those deals in order to get a satisfactory price where it fits the current budget.

With those problems in mind, constantly changing prices and the time and effort needed for manually searching and tracking products and deals is where we come in.

To address these common issues, we created a web application which provides facilities and tools for both customers and business owners to, first, search all major websites for products and services with a couple of clicks and display all of them on the same page in a structured format set by custom filters. Second, The means to track and monitor specific changes to those products and services for price variations among others. Third, to check each site in specific, or section in general for custom curated deals and discounts that are generated by our artificial Intelligence machine learning recommendation system models.

1.1. Purpose

Dynotify is a SaaS web service where users can search, check and track various products and services from dozens of sites. In addition, users can get notified via email,

SMS or push notification whenever a change is detected on the webpage. For example, a product price drop on Amazon or Flight ticket discount on an airline site.

The aim of this document is to specify complete description of the dynamic notification system to be developed.

1.2. Intended Audience

Therefore, intended reader groups for this software requirement specification are customers, business holders, partners, *developers and project managers and supervisors.*

Through this document, the workload needed for development, validation and verification will ease. To be specific, this document is going to describe functionality, external interfaces, performance, attributes and the design constraints of the system which is going to be developed.

1.3. Project Scope

This project website is divided into two main sections:

1. WordPress content management system on the home page (dynotify.com) which has some overviews of the product images, features, competitor comparisons, prices, and blog posts of the services we provide. This section is intended mainly to attract new customers to the web applications through marketing techniques like SEO and to inform the visitors about product details and use cases.
2. The main web application on a subdomain (app.dynotify.com), which has three main uses:

    First, enabling the customer to search for any product/service across major websites with a single click and returns a structured list of results according to some filters which saves the user time and effort from manually searching each site individually. (i.e: searching for “iPhone 11” on Amazon, eBay, Aliexpress, BestBuy, …).

    Second, browsing a list of deals categorized according to major websites and sections (top deals, daily deals, popular deals, …), which keeps the user up to date with the latest discounts and deals. (i.e: Today deals from Amazon, eBay, Aliexpress, BestBuy, …).

Third, whether a user searches for a product/service or finds a deal, he can choose to monitor that product or service for any changes such as price drop, then get alerted by a notification via email, SMS or push notification.

Currently, there are some websites that provide similar services, that are limited. On the other hand, we aim to provide a wider range of additional features that organizes the user's interests, widen his perspective and draw his attention to things that he may not be aware of.

1.4. Glossary

Term	Definition
DyNotify	The name of the main web application.
SaaS	Software as a Service, which the software is licensed on a subscription basis.
Customers	The everyday person who seeks to purchase a product for his house or a flight ticket to go on a holiday.
Business Holders	The travel agencies or local exchange offices who wish to get the latest updates and prices on flights, hotels and currencies.
Partners	Big players such as Amazon or small companies. The latter provide products or services that our customers can browse through our site.
Angular	Angular is a framework for building mobile and desktop web applications led by Google.
Firebase	Firebase is Google's mobile platform that helps you quickly develop high-quality apps and grow your business.
Firestore	Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase.
Google Functions	Google Cloud Functions makes it easy for developers to run and scale code in the cloud and build event-driven serverless applications.
IEEE	Institute of Electrical and Engineering.
Web Scraping	Web data extraction is a technique employed to extract data from websites whereby the data is extracted and saved to a local file in the computer or to a database.

API	An application programming interface is an interface or communication protocol between different parts of a computer program intended to simplify the implementation and maintenance of software

1.5. References

1.4.1 IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

1.4.2 https://dynotify.com & https://app.dynotify.com

1.6. Overview of Document

In this document, there are two more sections. The next one, Overall Description contains an overview of what the product should do and how it the users will interact with it. Further, the section also describes the environment in which the system will operate. Also, it discusses the constraints on the system implementation and design. Some assumptions and dependencies are also suggested.

The third section of this document, Requirement Specification, is written for both users and system developers. So, it is written in multiple levels of detail. Further, it addresses, in detail, the functional and non-functional requirements.

2. Overall Description

We are going to focus on describing the system in terms of product perspective, product functions, user classes and characteristics, assumptions and dependencies on the following section of this document. Next, we will address specific requirements of the system, which will enclose external interface requirements, requirements of the system, performance

2.1. Product Perspective

This product is a web based application that provides a simple mechanism for users to search and track products and services across major sites.

The following are the main features that are included in this product:

- User account: The system allows the user to create their accounts in the system and provide features of updating and viewing profiles.
- Search: The system allows the user to search for a specific products or services across major websites that provide them, and returns a structured list of data from all those sites.
- Monitor: The system allows the user to choose to monitor any product or service out of the search results. In order to be notified upon specific changes like price drop.
- Recommendations: The system displays sections containing a list of deals (Today Deals, Top Deals, Popular Deals, Recent Deals)

2.2. User Characteristics and Use cases

There are only three types of users that interact with this system, referred to as normal customers, business owners and administrators, and they can use the system for quite different purposes.

2.2.1. Normal customers can search - through our website - for any type of product or service that the partners offer on their own sites instead of searching each one individually. In addition, customers can also request to track the page of the product that he intends to buy, or the service to use, and under some criteria that he can specify via some custom filters, he will receive a notification when criteria is met.

2.2.2. Business owners use case is similar to normal customers, the difference is that they can search and monitor huge amount of products and services, imported from a list in text files, excel sheets or databases that they specify. Then, A notification is sent in a frequent basis set by the user.

2.2.3. The administrator use cases are divided into two sections:

Depending on the sections that were detailed in the product scope above:

1. WordPress administration panel from the home page (dynotify.com) which offers the following use cases and features:
    1: Website pages creation and management
    2: Blog posts creation and management
    3: Media files upload and management
    4: SEO and marketing tools
    5: Chat support feature
    6: Website traffic statistics and insights

2. Main administration panel from the WebApp (app.dynotify.com) which offers the following use cases and features:

    1. User and billing management
    2. Chat support via chat & tickets system
    3. Users statistics and insights
    4. Monitoring system management
    5. Recommendation system management

2.3. Operating Environment

The system is composed of multiple parts:

1. The main domain (dynotify.com) which will be hosted on a Cloud Virtual Private Server (VPS) for delivering the website to users.
2. A CDN (Content Delivery Network) will be used to minimize the latency between the user's location and the server location which decreases the overall page load time. In addition to an extra layer of security against different kinds of attacks and an overall website caching for increasing website delivery speed.
3. The scripts, which will be written in the Python language, will be hosted on a serverless computing network known as (Google Cloud Functions), this step ensures super fast execution of these scripts without a single point of failure and minimizes operation cost substantially.
4. Firebase will be used as a backend service for user authentication and management, and hosting the entire web application including the database on the subdomain (app.dynotify.com).

2.4. Design and Implementation Constraints

- The application front-end shall be developed using Angular framework as a web application using the nebular UI library and design system.
- The application back-end services (user authentication, database, ...etc.) shall be developed and managed by the firebase platform to shorten development time and improve scalability.

- The application scripts which are written in python shall be implementing web scraping techniques and API calls to fetch the required data for products and services from the partners websites. Because this process is quite heavy on resources, it will be hosted on a serverless architecture (Google Cloud Functions) to reduce the operational cost and increase scripts execution speed and reliability.
- The web scraping scripts shall be implementing many techniques to ensure reliability and to ensure the return of the correct data required, such techniques include: rotating proxies and browser user agents to avoid getting banned, time delay between each request.
- The changing layout of the partners websites will cause issues when scraping the required page, to solve this issue we will need to do a thorough testing and quality assurance on many different types of products and services for every single partner website to ensure that all cases are covered and the correct data is returned.
- The email and SMS notification systems shall be implemented using a third party services “sendgrid.com” and “plivo.com” respectively, this will eliminate common issues when sending mass amount of emails and to ensure SMS message delivery to the desired recipient, this will increase reliability and scalability and reduce the overall operational cost.

2.5. Assumptions and Dependencies

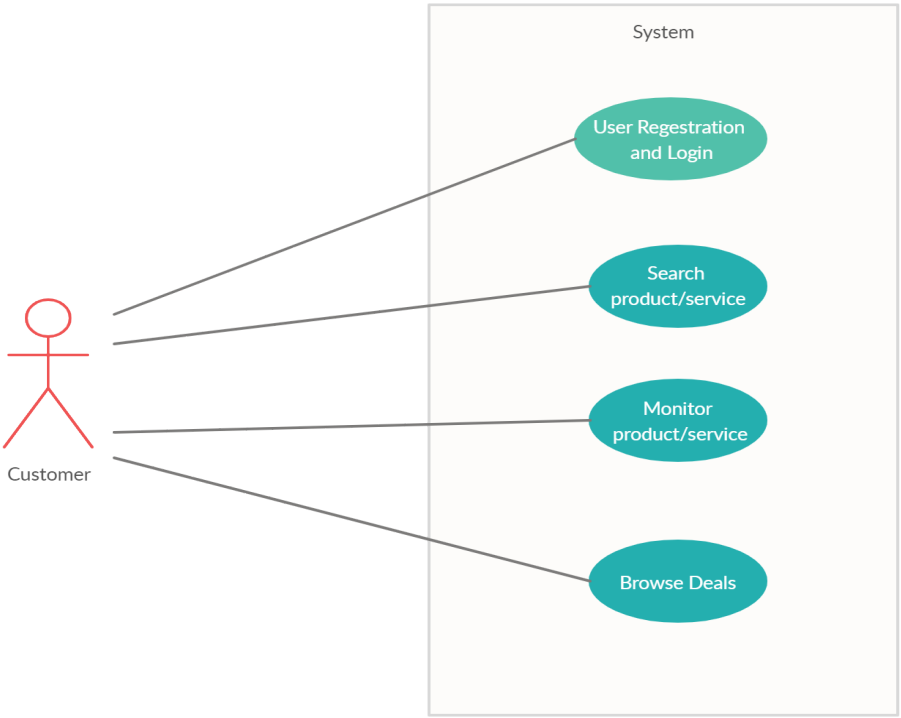
- The main domain running WordPress content management system (dynotify.com) is hosted on a Virtual Private Server (VPS) on a third party service provider, in case of server failures or resources limitation, it will cause issues with site delivery and speed to our site visitors.
- The subdomain running the web application (app.dynotify.com) is hosted on the firebase platform, in case of resources limitation or database connection issues, it will cause issues with web application usage and speed to our site visitors.
- The application python scripts will be running on a serverless architecture (Google Cloud Functions), the execution time and reliability are dependent on the availability of this service and resources.
- The system depends on the availability of the partner websites to fetch the products or services data, in case of unavailability or any other factor that might affect the source website, the web scraping script or the API service will fail to return the required data.

- The web scraping script are limited by the layouts which are coded and tested beforehand on the partners website, in case a new layout is detected, the web scraping script will fail to return the desired output.
- The API service depends on the partner website and in some cases are limited by them, in case where the limit is reached, we will not be able to use this API service for some time until the limits are reset.
- The email notification service depends on a third party website “sendgrid.com”, in case of any issues with the third party service, our email delivery will be affected.
- The SMS notification service depends on a third party website “plivo.com”, in case of any issues with the third party service, our SMS delivery will be affected.

3. Requirements Specification

3.1. User and System Requirements (Functional Requirements):

3.1.1. User Requirements:



General Use Case Diagram:

3.1.1.1. User Registration and login:

There will be a sign up page so that user can register an account. User must specify some information asked during registration. Upon validation, registration will be completed, and user will be notified.

There will be a login page so that the user can type into his login information, and login to the system. Login information will be email address and password specified in the registration process. In addition, user can login via his facebook, google account, twitter.

3.1.1.2. Search a Product or a Service:

Via name or url, user can search:

i. Products

ii. Flights

iii. Hotels

iv. Car Rental

v. Games

vi. Movies & TV

vii. Music & Video

viii. Stocks

ix. Currencies

x. Courses

xi. Websites Services

xii. Custom URL

Each has its own search engine and associated filters.

After running python scripts for searching, results are displayed as a set of cards with a brief description for the user to look at.

3.1.1.3. Monitor a Product or a Service:

When the user searches for something that he is interested in, as noted in section 3.1.1.2. He can choose whatever result he wants, to monitor upon specific changes like price drop. Then upon change, user is notified via email or SMS or even via a push up notification.

3.1.1.4. Browse Deals:

Today Deals: in this tab, a list of the most searched products or services for the current day are displayed.

Popular Deals: in this tab, a list of the most searched products or services for the current month are displayed.

Top Deals: in this tab, a list of the most converting & profitable products or services are displayed.

Recent Deals: in this tab, a list of the most relevant recent searched products or service from all the users are displayed.

3.1.2. System Requirements

3.1.1. User Management

3.1.1.1 User Authentication

Product will be used via a web browser. Each user will have his own experience, and he must be logged in to the server to access his activities. Hence, first-time users must perform complete registration process. To register to the system, user must specify some information asked during registration. After validation, registration will be completed, and user will be informed.

There will be a sign in page so that users can type into their login information, and login to the system. Login information will be email

address and password specified in the registration process. In addition, user can login via custom credentials, emails, or social media. Server let through the user if the given credentials are correct. If specified information is not matched, an error dialog will be shown. Otherwise, the user will be redirected to his personal dashboard.

When user forgets his password, he can request a new one from the system by specifying his email.

3.1.2. Search a product

This step requires the user to enter a product name in “Search” input field, after entering the product name, the python script for APIs and web scraping are called immediately in the Google Cloud Functions architecture to be executed, a JSON file will be the output of this execution to be returned to the front end to be displayed as cards.

Many websites options are given as a choice (e.g. Amazon, eBay, AliExpress, ...etc.) and by default “Amazon” is chosen. When the user clicks on any other website, the python script for the chosen site will be fired to be executed.

3.1.3. monitor a product

As a result of the previous step, a list of cards are displayed when searching for a product. Whenever the user clicks on any desired card, a popup window appears enabling the user to check further information about this product and given the option to either check the product original page on the source website or to set a custom criteria to track and monitor this product for price changes. In the event those criteria are met, a notification will be sent to the user email or phone number.

3.1.3 Smart detection for monitoring prices

From the previous step, an option is given to the user to either set a custom criteria for monitoring and tracking a specific product, or let our system choose whatever price range and frequency checks is adequate (Smart Detection).
This system works as follows:

- Takes the product url as input
- Checks the price with 6 hours interval

- If all the first 4 checks prices are equal or difference is less than 2%, Increase the frequency checks to 12 hours instead of 6
- This process is repeated until the max frequency checks interval is 60 hours
- When the frequency checks interval is greater than 40 hours, a 2 small checks 6 hours apart are made randomly within the 40 hours, if the price differs decrease the checks frequency by half.
- If the price drops from the original price by more than 7% (also dependent on the product price), it fulfills the required condition to send the price drop notification.
- Another option where it meets the alert condition where the current price this day is the cheapest from the previous 2 weeks or months by a difference of a min of 4%
- Register each check time frequency and price in the databases to be displayed as a graph when needed or to be learned from for future checks

3.1.4 Similar suggestions model

When monitoring a product, an option is given to receive suggestions of similar products, this gives the users a list of cheaper products from all sites that are the same or very similar to the original product he requested the monitoring for. This system is executed as follows:

Products Data Collection Stage:

- Checks the product url page if it contains similar products section to be collected
- Searches in all e-commerce websites with the original product title, the output search results product are collected
- The output result is a large list of products containing a URL, title, price, image for each product.

Ranking Stage:

- A score number is given to each product to determine its relevance to the original product
- The score increases by a certain number when the main product title are similar (fuzzy match > 85%) to the original product title.
- The score increases by a certain number the price is close or in a reasonable lower range of the original product price.
- The score increases by a certain number the original product image is the same or very similar to the original product image.

An example of this system:

The end result for a price alert of "PlayStation 4" on Amazon is: all PlayStation 4 listings on all sites ranked by relevance and which are cheaper than the original product price and with a ranking score larger than a certain number.

3.2. Detailed Non-Functional Requirements:

3.2.1. Security and Privacy:

The system maintain fine level of security against unauthorized access, authorization and authentication is provided by Firebase. because each user has an account that holds personal information, and activities.

In addition, the system website includes an SSL certificate which enables the website to run securely on https protocol. This process is essential when enabling the billing system later on.

3.2.2. Efficiency and Usability

This system ensures a high level of efficiency which was accomplished through many levels of testing and iteration of the user experience flow. A satisfactory goal was achieved by minimizing the number of clicks needed to reach the end goal and displaying each stage at a visual intuitive interface.

To increase the efficiency, a BOT was designed in this system by the name of “DYNO”, this bot displays information and examples of any part of the application which the users hover the mouse cursor on. This decreases the need for users to request help from the system administrators as frequent questions or inquiries will be displayed by the bot.

3.2.3. Scalability and Performance

The system was designed with scalability in mind, Firebase platform is built for performance and scalability with it’s Firestore database architecture, it can scale to hundreds of thousands of simultaneous connections without any issues.

Google Cloud Functions was chosen to ensures super fast execution of the backend python scripts without a single point of failure and to minimizes the operation cost substantially. In addition, using this service eliminates the usual issues with normal servers like resources limitation and increases reliability.