

Sailor - Pioneers of the digital entrepreneurship

Architecture and Design

Revision History

Version	Modifier	Date	Description of Change
0.1	A. D'Onofrio	14/04/2020	Initial rough draft.

Table of Content

1. Introductionpag. 2
2. Design Goalspag. 4
3. System Behaviourpag. 5
4. Use Casespag. 6

1. Introduction

This document describes the architecture and design for the Sailor platform, developed by the Flipped Company. Sailor is a web-based platform designed to help non-tech people to embrace and to embark on a career in the digital world. The project is basically a *service arbitrage* platform.

Service Arbitrage (or dropservicing), is a business model that involves three parts:

- The Client, or *buyer*, is the person that lands on our website and decides to buy one of our services.
- The Supplier, or *seller*, is the person/company that sells a particular product/service.
- Us, or *Sailor*, is the entity that sits in the middle, it's purpose is to "bring together" the two main parts. Basically when a client decides to rely on one of our bundles and buys one, our system scans our database and finds the right Supplier for the required product/service and proceeds to submit the order to them (*the seller*). Once the Seller completes the task that it's been given it will send the completed product back to us. Since the bundles include many different products we will repeat this process as many times as needed. The Client will receive a full pack that includes all the required products when all the Suppliers provide the ordered service.

Using this platform will be advantageous for everyone, especially the client, since he will not have to worry about anything in the process, after he buys a particular bundle he's sure that he will receive it in a few days (depends on the supplier).

For this project, the major stakeholders are:

1. Users and Customers: They want an intuitive platform that will help them choose the perfect and most suitable service for their needs. They also want assurance that they will receive the ordered products.
2. Developers: They want an application that is robust and easy to maintain, they also want to be able to update and upgrade it in the most efficient way possible.
3. Suppliers: They will not actually be in touch with the platform, they are already offering a service somewhere else on the Internet, we'll just buy it from them and then resell to our client after putting all the required products together.

The structure of this project is fairly simple, we'll give a more detailed description of the following components:

1. Landing Page: The landing page has to be very simple and intuitive, with a very modern design. It has to be very supportive to the client since remember, our target customers are people that don't have the required skills to produce all the assets needed to start an online business. The page has to include a client support section that will send messages directly to us.
2. Purchase Trip: This section of the platform will be used everytime a client decides to purchase a bundle. It will include a description of what the selected bundle offers, the approximate time of delivery, the price, the occasional option to choose a faster delivery time (with a price surplus) and the payment method.

To purchase a service, the client has to make an account, this will be necessary because he'll then be able to track down his order by looking at his orders status section. He will be able to send tickets to us explaining the issues he might be encountering. Finally the completed order will be delivered and made available on his personal profile.
3. Dashboard: This page will be reserved for the staff. The purpose of this page is to have a complete overview of the overall platform status. This means that it will show all the tickets, the pending orders and everything will contain the informations about the Client enriched with the timeframe references. A colored status bar will also be available for a more immediate feedback, it will show how far an order has gone and how much time is left before delivery. The staff will finally be able to send the completed bundle to the client through this page.

2. Design Goals

The following points will follow as guidelines for the developers. Following them should result in a very intuitive page that will successfully fulfill its purposes.

1. Emphasis: giving emphasis to the most important components will result in a easier to use interface.
2. Balance and Alignment: Even though a symmetrical design could be boring it is probably the better choice for our target audience since it will be visually pleasing and more "trustful" to non-expert eyes.
3. Contrast: Everything has to be extremely easy to read, we don't want to have parts that can stress our clients, lighter colors have to be placed onto dark backgrounds and black filters have to be applied on the images.
4. Repetition: repetition unifies and strengthens a design. Using this principle will give a sense of trust and tranquility that will probably feel the user more inclined to use our services.
5. Proportion: grouping related items can give them importance, hence contribute to a more professional look.
6. Movement: we need to control the elements in a way that allows the eye of the visitor to follow a flow towards our call to action (*sign in, buy, comment etc...*)
7. White Space: the page doesn't have to be completely packed with elements, we'll use our white space to create a visual hierarchy and organization. Our brain associates white spaces around an element with importance and luxury, hopefully it will push the visitor into becoming a client.

3. System Behavior

The system itself will be pretty straight-forward and no shenanigans whatsoever. Our client will be able to register into the platform, purchase products and services, check the status of his orders and remain in contact with the support staff.

3.1 Registration

In order to sign up, the future user simply has to:

- click the "sign-up" button on the top-right corner of the page
- fill up the form with his informations
- wait for the confirmation email
- finalize his account by confirming it through email

3.2 Login

In order to log in, the user has to:

- click the "sign-in" button on the top-right corner of the page
- insert email and passwords

3.3 Purchase

In order to purchase a service, the user has to:

- sign into his/her account
- go to the "services" section
- choose a bundle
- click on the "get started" button

On the checkout page:

- choose all the desired products
- accept the terms and conditions
- choose a payment method (PayPal)
- click "Purchase"
- confirm via email

3.4 Order Checkup

In order to check up on the order, the user has to:

- sign into his/her account
- go to his/her profile

- click onto "My Orders"

3.5 Order Delivery

In order to complete an order, the user has to:

- sign into his/her account
- go to his/her profile
- click onto "My Orders"
- click onto the desired order (with the status "ready")
- download the content

4. Use Cases

These use cases are guidelines for the developers, they should be followed but are not mandatory.

4.1 Registration / Login

- display the required form
- send the informations to the database
- start a session
- redirect to user profile

4.2 Purchase

- gather up the required informations through html forms
- verify payment
- complete the order with status, delivery time etc...
- start the "forward order" procedure
- redirect to user profile

4.3 Forward Order to supplier

- use the informations gathered from the user order to forward the order to the different suppliers
- setup order status and delivery times with the informations recieved by the supplier