



FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

NERDWIRED

Sistemas de Informação

Mestrado Integrado em Engenharia Informática e Computação

Ana Macedo - up201909572@fe.up.pt
André Moutinho - up201707291@fe.up.pt
Leonor Sousa - up201705377@fe.up.pt
Luís Fernandes - up201706910@fe.up.pt
Sérgio Dias - up201704889@fe.up.pt

Project Overview

Our project consists in the development of a **Mobile Application** to support the **management** of all the **logistic activity** related to **NerdWired**, a company that sells **computer components** to final customers.

The app contemplates the **purchase** and **sales** process, handles **stock replenishment** and **allocation** to **warehouse** and assists the **picking** process of a sales order.

It has mainly 2 types of users: **managers** and **pickers**.

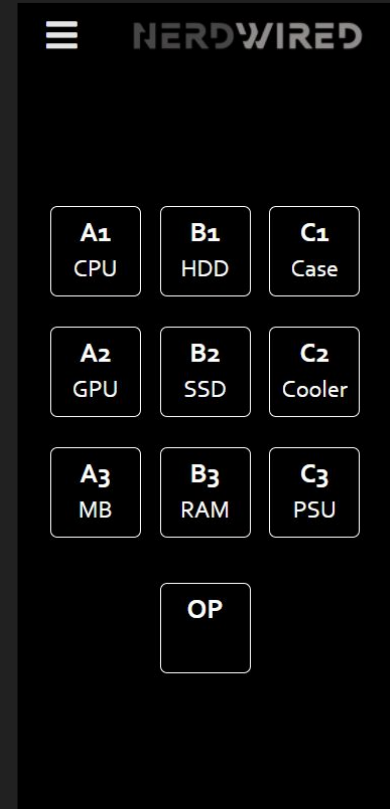


Fig 1 - Home page (warehouse overview)

Context

When orders from **suppliers** arrive at the **warehouse**, their products are immediately stored in their respective **subsections**. As the customers make orders, the picker picks the respective products, based on **picking waves** that can aggregate products from different orders.

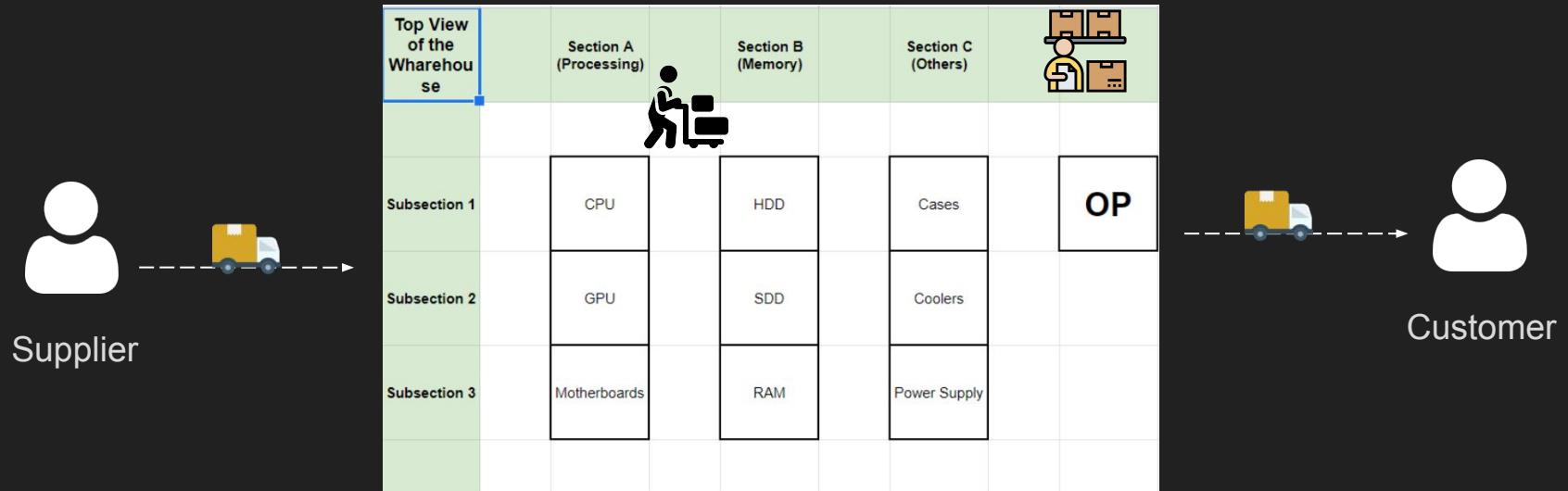


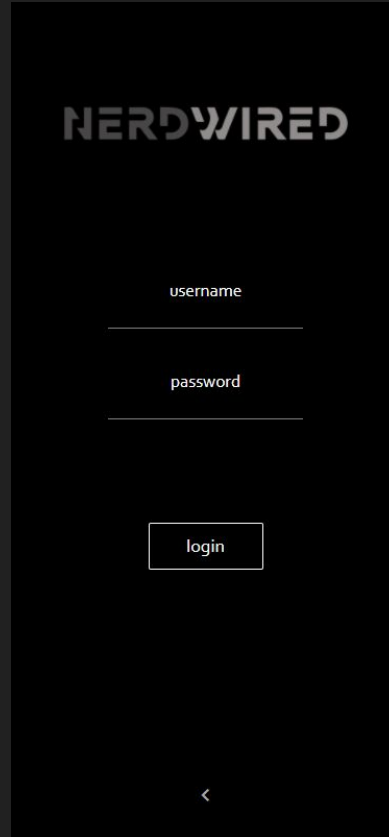
Fig 2 - Business process scheme of the target company

Features

Authentication

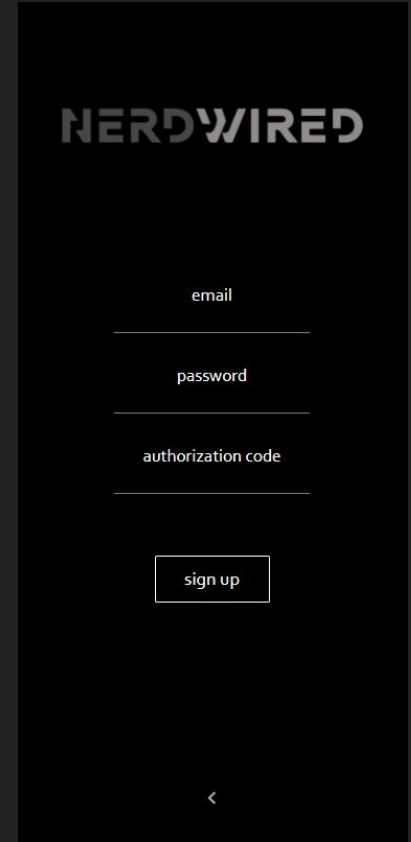
Login - signup

- Email / password based;
- Firebase Authentication.



A dark-themed mobile app login screen. At the top, the text "NERDWIRED" is displayed in a bold, white, sans-serif font. Below the header, there are two input fields: the first is labeled "username" and the second is labeled "password", both in a small, white, sans-serif font. Each label is positioned above a horizontal white line representing the input field. At the bottom of the form, there is a white rectangular button with the text "login" in a small, dark, sans-serif font. At the very bottom of the screen, there is a small, white, left-pointing chevron icon.

Fig 3 - Login screen



A dark-themed mobile app signup screen. At the top, the text "NERDWIRED" is displayed in a bold, white, sans-serif font. Below the header, there are three input fields: the first is labeled "email", the second is labeled "password", and the third is labeled "authorization code", all in a small, white, sans-serif font. Each label is positioned above a horizontal white line representing the input field. At the bottom of the form, there is a white rectangular button with the text "sign up" in a small, dark, sans-serif font. At the very bottom of the screen, there is a small, white, left-pointing chevron icon.

Fig 4 - Signup screen

Features

Authentication

User permissions - 2 different roles

Picker



Executes picking waves, **collecting** products that are ordered by clients and bringing them to the Out Point. Inputs feedback throughout the **picking** process.

Manager



Triggers the generation of **picking waves** and assigns them to the pickers. Guides the picking process when necessary and reads the feedback provided by pickers, taking actions accordingly.

Features

Stock management

Listing

- List of **products** stored in each **warehouse section**;
- Information about **reference** and **stock** level of each product.



The screenshot shows a web application interface with a dark theme. At the top, there is a hamburger menu icon and the text 'NERDWIRED'. Below this, the section 'A1 CPU' is highlighted. A table with three columns: 'Ref', 'Name', and 'Stock' is displayed. The table contains eight rows of product data. At the bottom of the table, there is a left-pointing arrow icon.

Ref	Name	Stock
01006	Intel Core i7-9700K	30/50
01001	AMD Ryzen 5 3600	25/50
01007	Intel Core i7-10700K	50/50
01003	AMD Ryzen 7 3700X	50/50
01005	Intel Core i5-10400	50/50
01004	AMD Ryzen 9 3900X	50/50
01002	AMD Ryzen 5 3600X	50/50
01008	Intel Core i9-9900K	50/50

Fig 5 - Stock listing (per section)

Features

Stock management

Adjustment

- Once a **Picking Wave** is **finished**, the **stock** of each picked item is **updated**;
- **Supplier orders'** note are read and **stock** increases automatically.

Features

Orders

Listing

- List of **client orders**;
- List of **supplier orders**.



Clients' Orders			
Client	Order	Date	Status
0003	ECL.2020.8	2020/12/10	WFP
0004	ECL.2020.5	2020/12/10	WFP
0004	ECL.2020.6	2020/12/10	WFP
0001	ECL.2020.2	2020/12/04	Picking
0004	ECL.2020.7	2020/12/10	WFP
0001	ECL.2020.3	2020/12/09	Picking
INDIF	ECL.2020.1	2020/11/30	Picking
0003	ECL.2020.4	2020/12/10	WFP
0013	ECL.2020.9	2020/12/10	WFP
0003	ECL.2020.10	2020/12/12	WFP

Fig 6 - Client orders listing



Suppliers' Orders		
Supplier	Order	Date
Amazon PT	ECF.2020.3	2020/12/10
Intel	ECF.2020.4	2020/12/10
Amazon PT	ECF.2020.2	2020/12/10
Corsair	ECF.2020.6	2020/12/10
NZXT	ECF.2020.9	2020/12/10
HyperX	ECF.2020.7	2020/12/10
Asus	ECF.2020.10	2020/12/10
MSI	ECF.2020.11	2020/12/10
Amazon PT	ECF.2020.1	2020/12/06
AMD	ECF.2020.5	2020/12/10
Seagate	ECF.2020.8	2020/12/10

Fig 7 - Supplier orders listing

Features

Orders

Details

- Visualization of details such as **products** that are part of an **order** and their respective **ref** and **quantities**.

Ref	Loc	Name	P/Qty
04004	B1	Seagate SkyHawk 8TB SATA III	10
04005	B1	Western Digital Gold 4TB	13
04003	B1	Seagate IronWolf 6TB SATAIII	7
05002	B2	SSD Samsung 870e 1TB	5
05006	B2	SSD Corsair MP400 4TB	17
05005	B2	SSD Corsair MP400 2TB	8
02008	A2	AMD Radeon RX 5700 XT	9
01001	A1	AMD Ryzen 5 3600	6
07004	C1	Cooler Master H500	15
03003	B3	Corsair Vengeance RGB	2

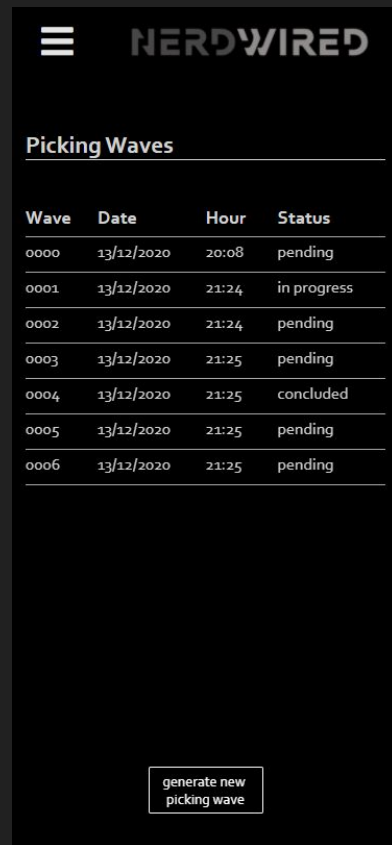
Fig 8 - Order details view

Features

Picking waves

Listing

- Visualization of the generated **picking waves**;
- **Manager** can visualize all **picking waves**;
- A **picker** can only visualize the **picking waves** that were **assigned** to them.



The screenshot shows a web application interface for 'NERDWIRED'. At the top left is a hamburger menu icon. The title 'NERDWIRED' is in a bold, sans-serif font. Below the title, the section 'Picking Waves' is highlighted with a light blue background. Underneath this section is a table with four columns: 'Wave', 'Date', 'Hour', and 'Status'. The table contains seven rows of data. At the bottom right of the interface is a button labeled 'generate new picking wave'.

Wave	Date	Hour	Status
0000	13/12/2020	20:08	pending
0001	13/12/2020	21:24	in progress
0002	13/12/2020	21:24	pending
0003	13/12/2020	21:25	pending
0004	13/12/2020	21:25	concluded
0005	13/12/2020	21:25	pending
0006	13/12/2020	21:25	pending

generate new picking wave

Fig 9 - Picking waves listing

Features

Picking waves

Assigning

- A **manager** can assign a specific **picker** to carry out a **picking wave**.

Picking Wave 5		
Picker:	Select a picker	Status: pending
Ref	picker	P/Qty
	picker2	
	picker3	
A11		
01001	AMD Ryzen 5 3600	0 / 4
01001	AMD Ryzen 5 3600	0 / 1
A21		
02001	Nvidia GeForce GTX 1060 6GB	0 / 1
02004	Nvidia GeForce RTX 3090	0 / 3
B31		
03001	HyperX Impact 16GB DDR4	0 / 8
03006	HyperX Fury 64GB(4x16GB) DDR4	0 / 2
B32		

Fig 10 - Picking wave details (manager view)

Features

Picking waves

Details

- List of **products** that are part of the **picking wave**, grouped by warehouse sections;
- **Quantity** to pick for each item;
- The order that warehouses appear in indicates the **route** to follow.

Picking Wave 5		
Picker:	Select a picker	Status: pending
Ref	picker	P/Qty
	picker2	
	picker3	
A11		
01001	AMD Ryzen 5 3600	0 / 4
01001	AMD Ryzen 5 3600	0 / 1
A21		
02001	Nvidia GeForce GTX 1060 6GB	0 / 1
02004	Nvidia GeForce RTX 3090	0 / 3
B31		
03001	HyperX Impact 16GB DDR4	0 / 8
03006	HyperX Fury 64GB(4x16GB) DDR4	0 / 2
B32		
<		

Fig 10 - Picking wave details (manager view)

Features

Picking waves

Processing

- **Picker** can update the picked amount of each product as they execute the **picking wave**;
- **Picker** can submit a **textual note** if additional information needs to be provided regarding a **picking wave** (e.g. exceptions on inventory).

NERDWIRED

Picking Wave 1

Ref	Name	P/Qty
A12		
01006	Intel Core i7-9700K	5 / 5

Tudo ok!

< Submit

Fig 11 - Picking wave input screen

Features

Picking waves

Delivery note

- A **delivery note** is automatically generated when all items of an order are picked.

Goods receipt

- A **good receipt** is automatically generated when an **order** from suppliers arrives to the warehouse.

Features

Picking waves

Automatic generation

- The **managers** only have to input the amount of products for the **Picking Wave**. The **composition** of the Picking Wave is created **automatically**!

It is done based on multicriteria, aiming to:

- Maximize the number of **finished orders** after the **Picking Wave** is executed;
- Process **older orders** faster;
- Select a set of products that allow for an **efficient route**.

Features

Picking waves

Route calculation

- Based on the generated product set, we calculate the shortest route to visit every collection point, starting at the **Entry Point** to the warehouse and ending at the **Out Point**, where orders are dispatched.
- Takes into account the warehouse shape, exploiting this knowledge during the **shortest path** searching;
- Always returns a possible **route** even if a certain amount of time is reached before the algorithm executes completely;

Features

Picking waves

Warehouse plant

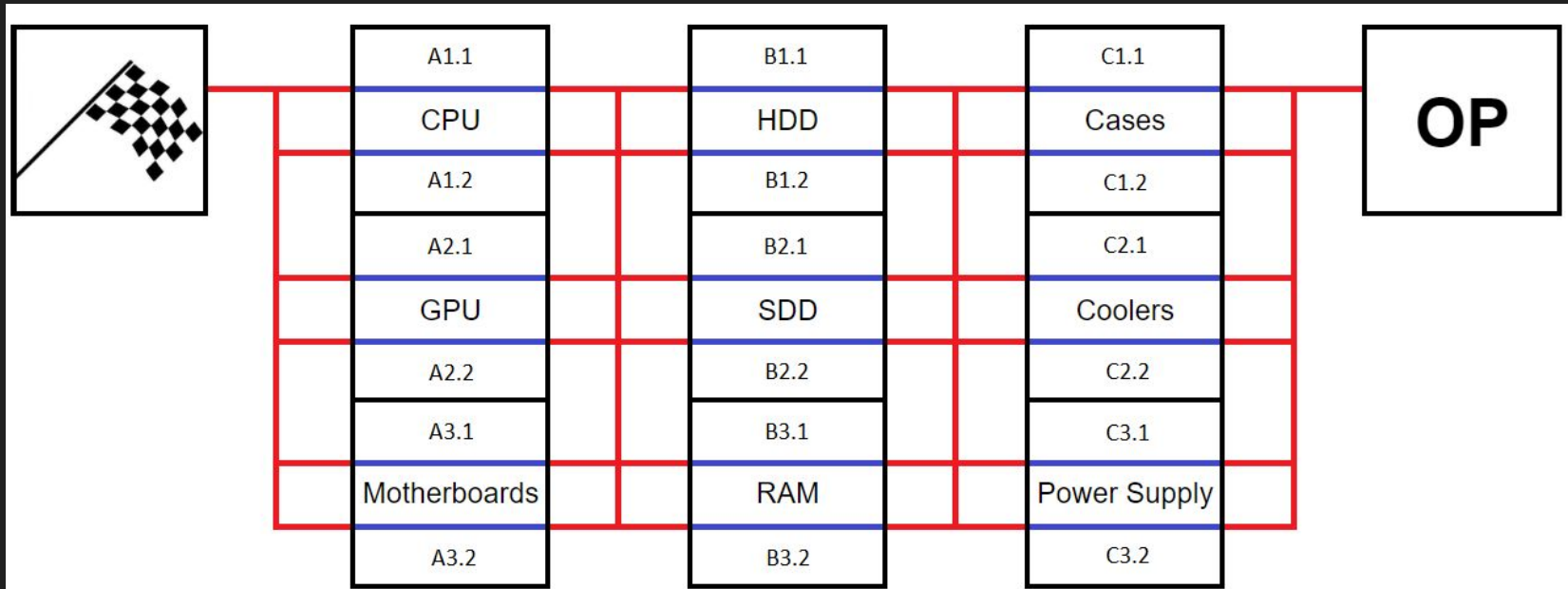


Fig 12 - Warehouse plant

System Architecture & Technologies



Jasmin (cloud management software)

React Native (framework used for building the app)



React Native



Firebase

Firebase (database provider)

Demo

Navigation through the app