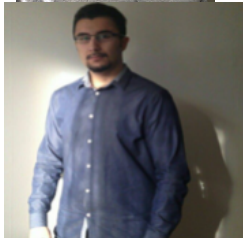


DAT076 / DIT126
Grupp 3
Webshop (BigStore)



Robin Rehnberg - rerob@student.chalmers.se



Ömersahin Keskin - omers@student.chalmers.se



Mojtaba Ataie - ataiem@student.chalmers.se



Allan Ridha - allanr@student.chalmers.se

Github link: <https://github.com/Marwix/WebAppProject>

1. *Introduction*

This project is a general store that sells everything. The products are displayed using a simple grid system. The user interface is flexible and able to be displayed on a large range of devices without issues.

On the homepage where the products are displayed the user is able to add products they want to a cart. However, to actually proceed to the cart the user must log in if they have an account or register if they want to continue. Once the user has created an account or signed in they can proceed to the checkout page where they are able to continue with their purchase. Before completing the purchase the customer is able to provide a coupon code (assuming there is an active one) to receive a reduced price.

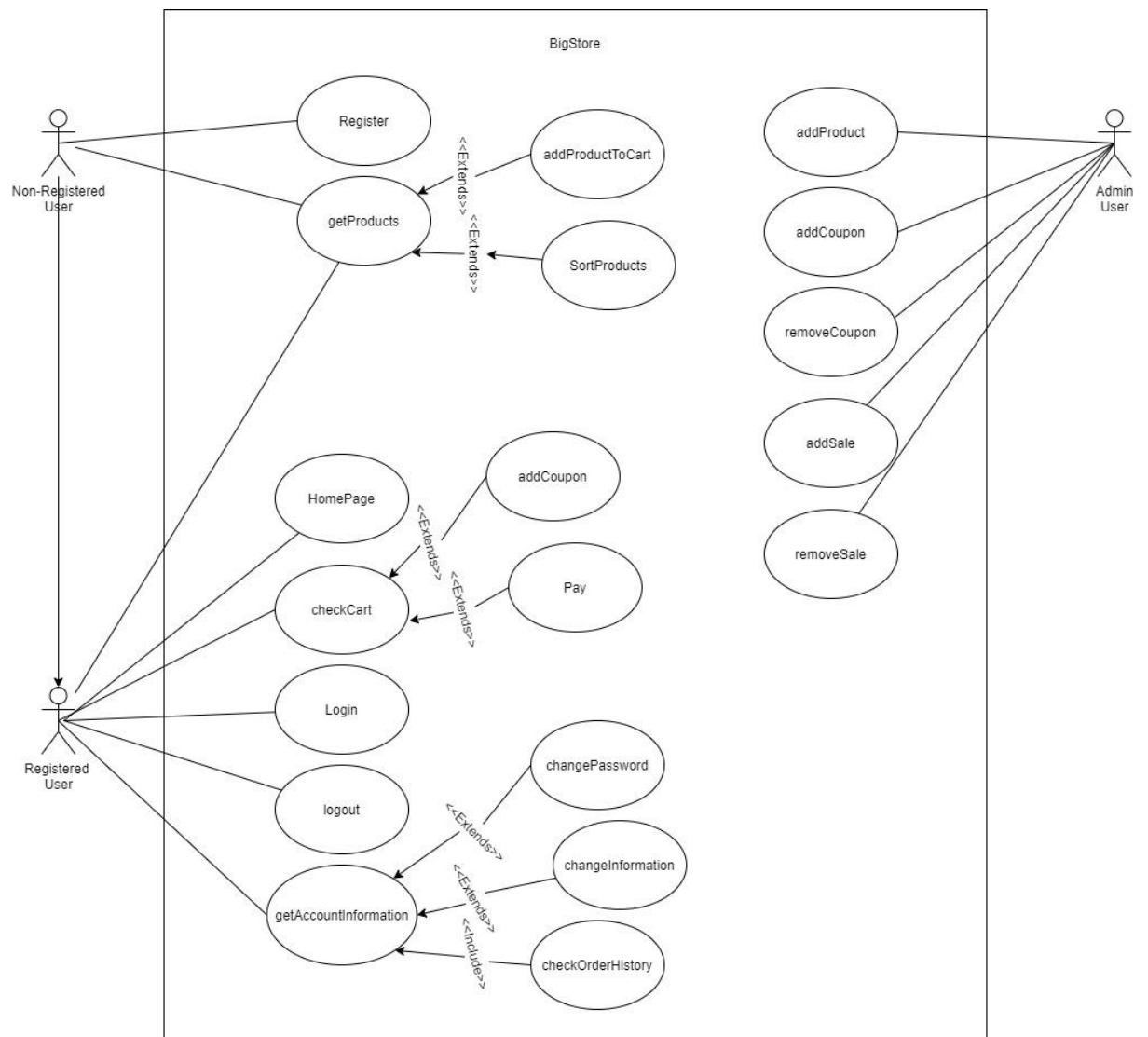
In addition to the shopping procedure there is also an account page. On the account page the customer is able to change the stored shipping information along with their email and password. At the very bottom there is also a purchase history tied to the currently signed in customer.

If the signed in user is an Admin account they gain access to an additional area of the account page, the admin tools. The admin tools simply contain a button redirecting them to a page with tools only available to administrators.

The admin page allows admin users to add new products, add new coupons, remove old coupons and create/remove single product sales.

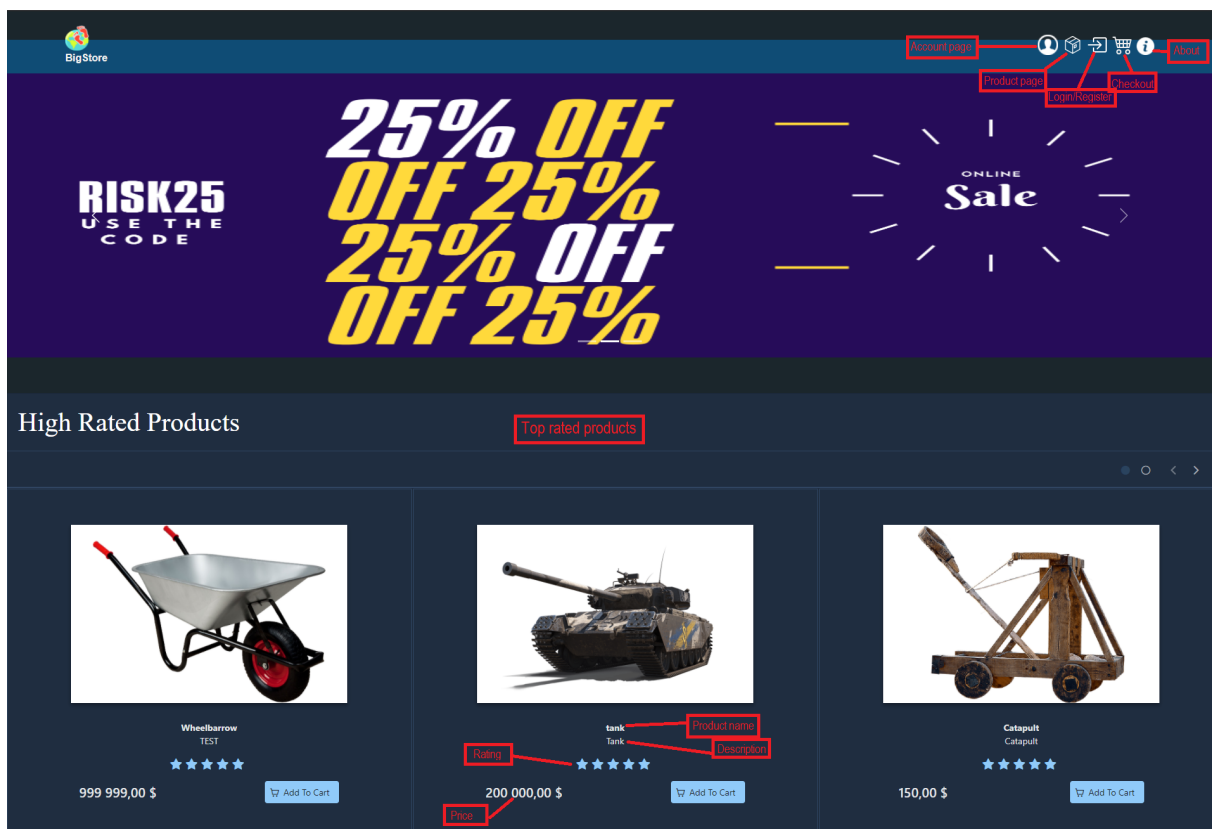
The website's theme is different shades of blue in addition to the PrimeFaces theme "Vela".

2. A list of use cases

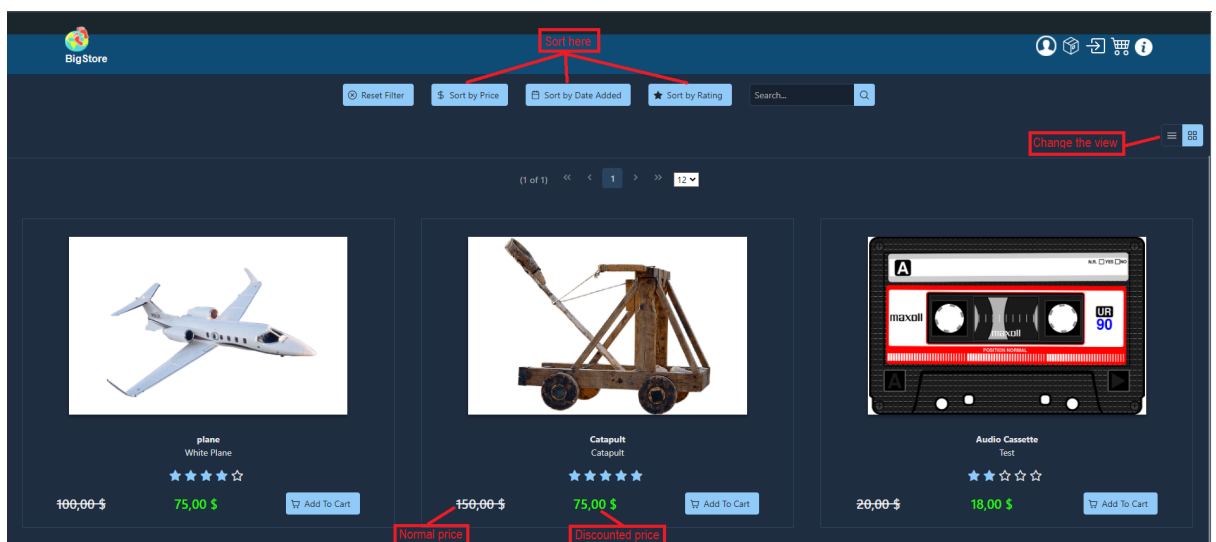


3. User manual

Home page:



Product page:



Account page:

Account Information

Your account info

Email *
test123@mail.se

Firstname *
test123@mail.se

Lastname *
test123@mail.se

Phonenumber *
123456789

Address *
Test.gata

City *
Göteborg

Zip *
41111

Save changes

Change Password

Change your password here

Old Password *

New Password *

Repeat New Password *


Change Password





Order history

Order nr. 3 (2021-03-07)	Amount
Item	75.0
GameStop	75.0
Total:	75.0
Order nr. 2 (2021-03-07)	Amount
Item	100.0
GameStop	100.0
Total:	100.0
Order nr. 1 (2021-03-07)	Amount
Item	75.0
GameStop	75.0
plane	75.0
Total:	150.0

Your previous purchases

If you are not logged in, the account info and checkout page will look like the picture below.





Sorry you need to login/register to view cart.

[Click here to register/login](#)

Checkout Page:

The screenshot shows the checkout page of BigStore. It includes a coupon code field, email address field, and a delivery address form. A cart table is also visible, showing items like 'plane' and 'GameDay'. Annotations highlight key elements: 'Get discount on total price if you have a coupon code' points to the coupon code field; 'Your email' points to the email address field; 'Your delivery information' points to the delivery address form; 'Cart with items' points to the cart table; 'Total price' points to the 'Total to pay: \$200.0' text; and 'Pay here' points to the 'Pay now' button.

CHECKOUT

COUPON CODE:
Code:

EMAIL ADDRESS:
Email:

DELIVERY ADDRESS:
Firstname:
Lastname:
Address:
City:
Postcode:

CART:

Items	Amount	Color	Price	
plane	1	Blue	1 x 190.0	X
GameDay	1	Red	1 x 10.0	X

Total to pay: \$200.0

By placing your order you agree to our Terms and Conditions, privacy and return policies. You also consent to some of your data being stored by BigStore, which may be used to make future shopping experiences better for you.

About Page:

The screenshot shows the 'About' page of BigStore. It contains a description of the shop, a list of authors, a purpose statement, and a GitHub link with the GitHub logo.

About

BigStore is an e-commerce shop where you can buy items such as stocks, games, planes and tanks.
To buy items, please go to Products and select your items and pay in the checkout page.

Authors

Allan Ridha
Mojtaba Ataie
Ömer Sahin
Robin Rehnberg

Purpose

Made for the project course Web Development (DAT076). All rights reserved!

Github link

Admin Page:

Add new product

Product and image name *

Price *

Stars (1-5) *

Color *

Measurements (Width * Length * Height) *

Weight *

Product description *

Add product

Add new Sale

Product ID *

Discount for specific product

Sale *

Discount percentage

Add coupon

1-5 of 5 records

<< < 1 > >> 10 ▾

Product ID	Product name	Price Multiplier	Remove sale
101	plane	0.75	✕
102	Catapult	0.5	✕
103	Audio Cassette	0.9	✕
104	tank	0.0	✕
151	Wheelbarrow	0.0	✕

1-5 of 5 records

<< < 1 > >> 10 ▾

Add new Coupon

Coupon Code *

Coupon for all available products

Multiplier *

Discount percentage

Add coupon

Coupon Code	Multiplier
-------------	------------

No records found.

4. **Design**

Each webpage (.xhtml) comes with an additional controller and view class. Whenever a webpage needs access to outside information it will have to be retrieved from the view or controller class. The view classes simply hold data while the controller classes can manipulate it or provide the view with new data.

Frameworks and libraries used:

JSF: Used to construct the webpages and its components.

Jakarta

PrimeFaces - Many of the original HTML elements were replaced by similar tools in the PrimeFaces library

Lombok: Library that provides developers with annotations to automatically add features such as getters, setters, equals methods and more.

JUnit: Library to enable testing in a java environment.

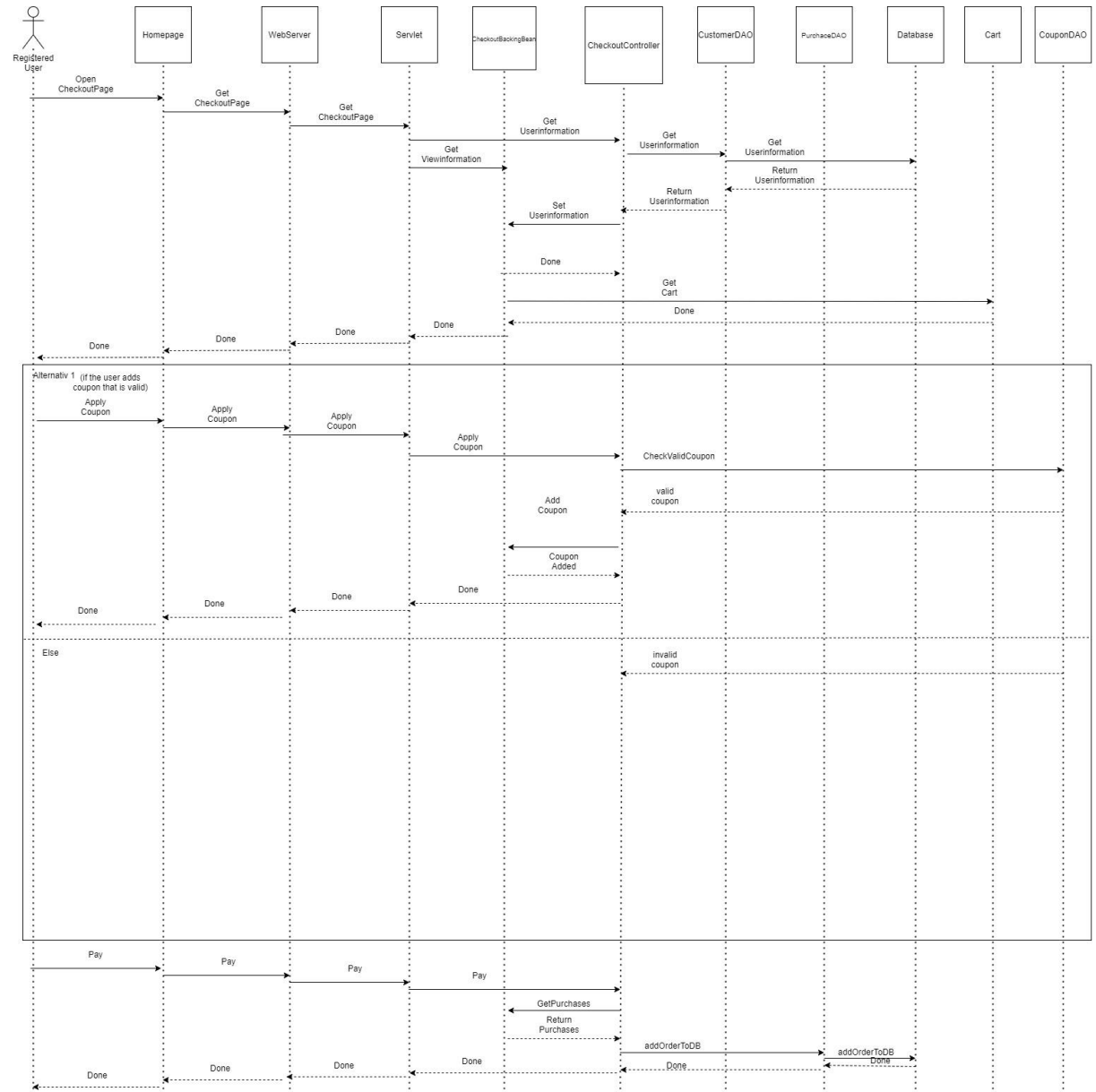
Arquillian: Tool used to run the tests.

Payara: Open source server to run the website.

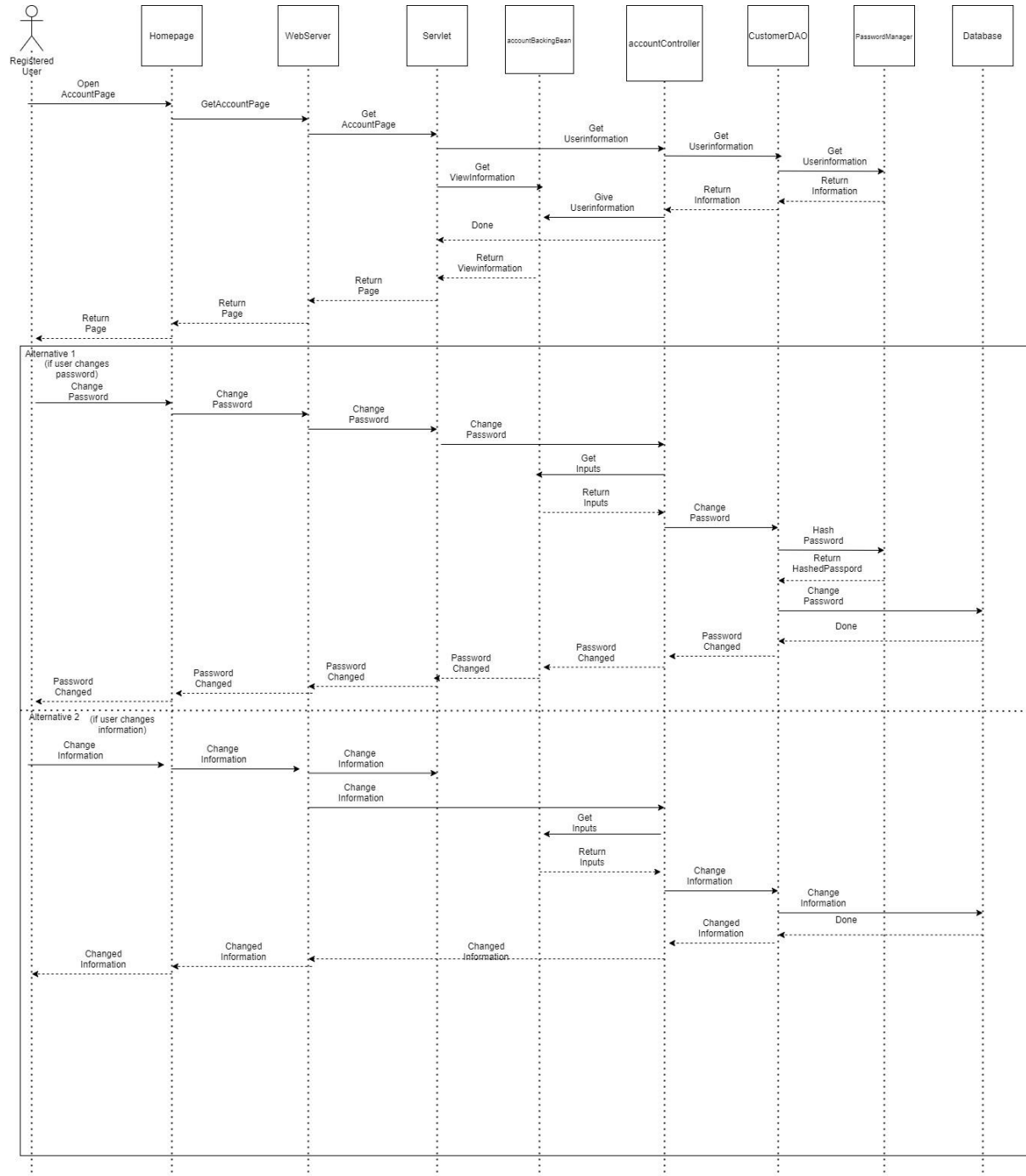
Apache Derby: Used to make the databases.

5. Application flow

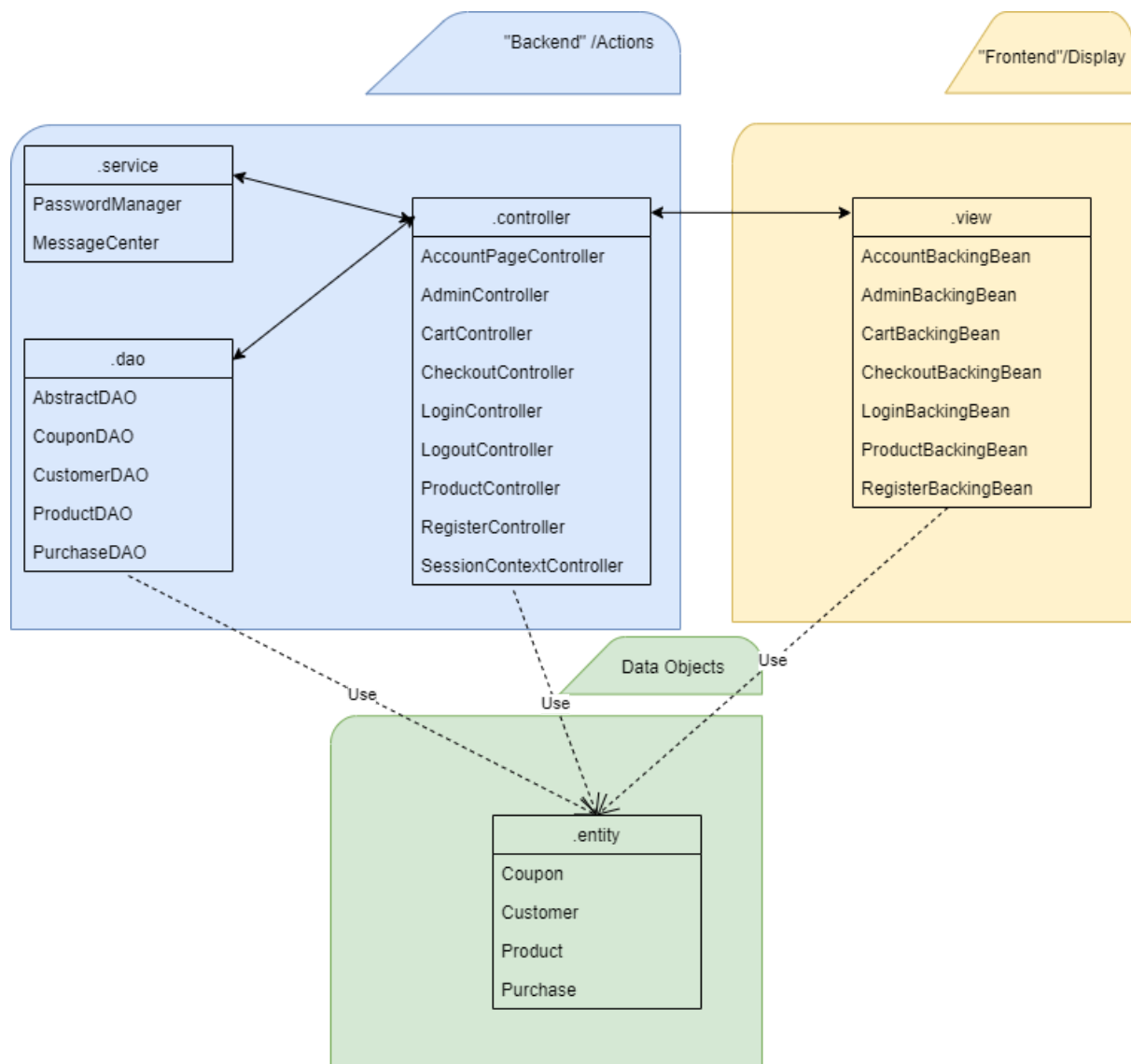
getUserInformation



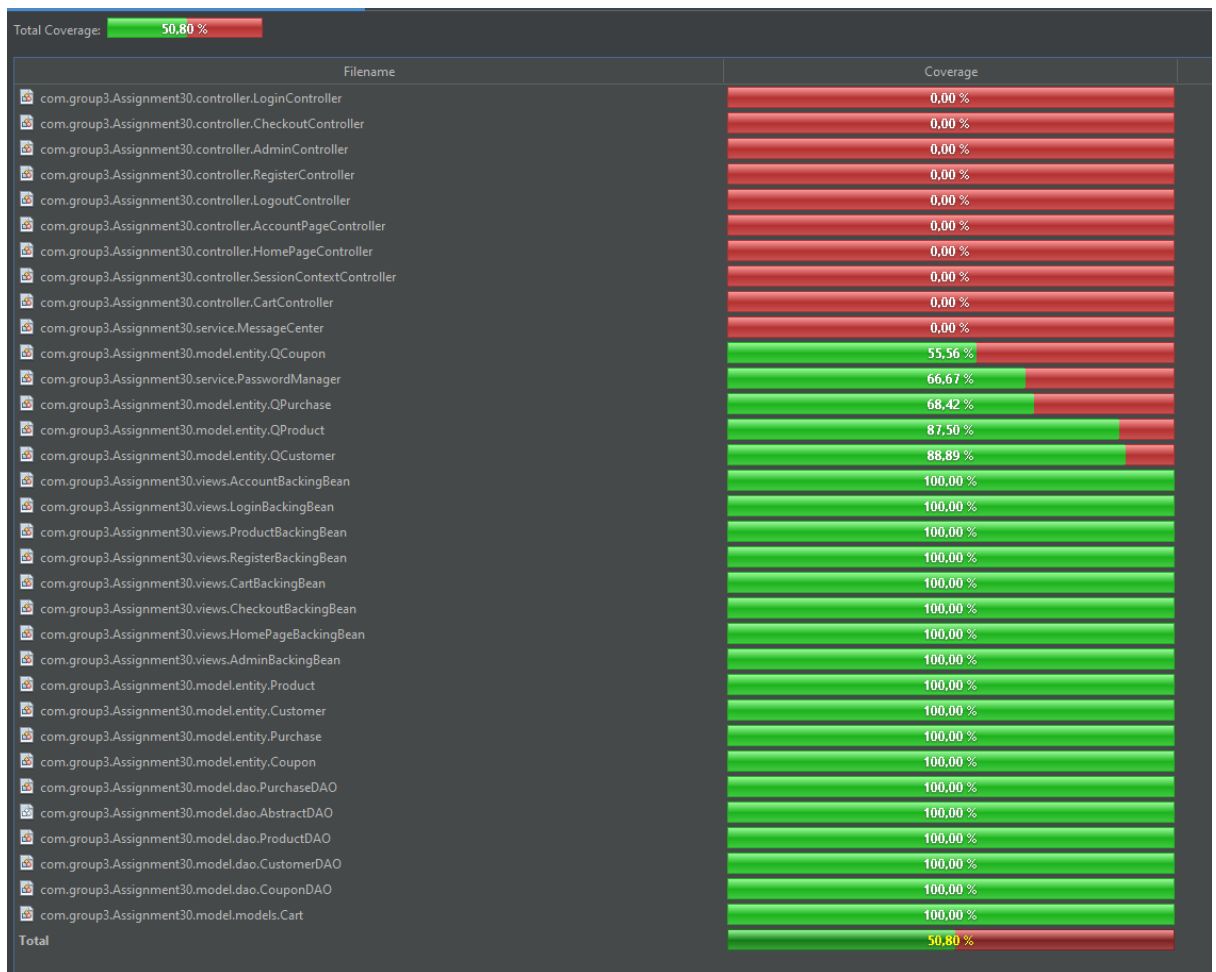
CheckCart



6. Package diagram



7. Show a code coverage report



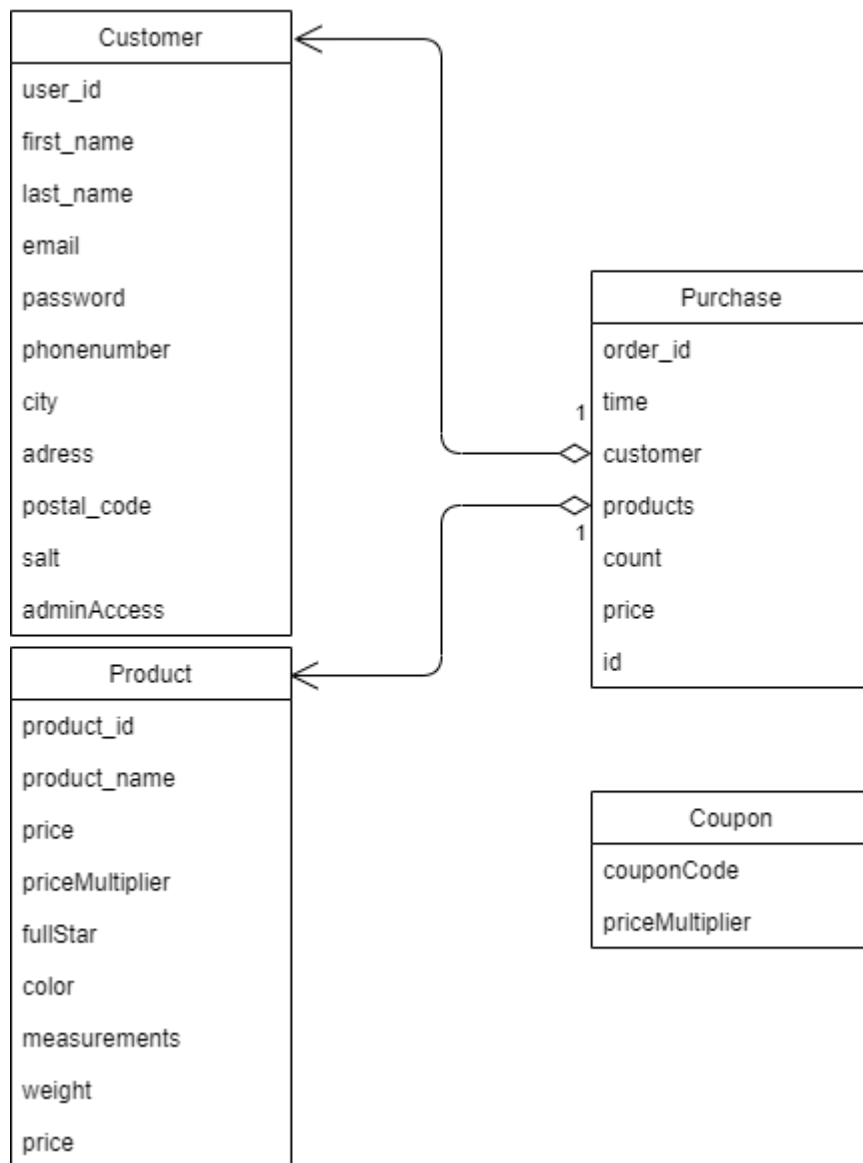
PasswordManager is at 66,67% because it contains two exceptions that will only be thrown if the underlying libraries are modified.

QEntity - Remain at sub-100% because quite frankly we do not know how to test them further and since they are generated files we do not consider them top priority. (???) Duger detta? vet inte hur jag ska förklara Q klasserna

Controllers

We didn't try our controller since we couldn't test it with what we had. To download graphene complicated everything even more since it stopped working instead of putting time to make graphene work we used the time to clean our code and move methods to correct places.

8. **Model diagram** - An UML diagram of the model



9. *Responsibilities*

Html

404.xhtml - Robin&Ömer
LoginPage.xhtml - Robin&Ömer
accountPage.xhtml - Robin&Ömer
adminPage.xhtml - Robin&Ömer
checkoutpage.xhtml - Allan some small assists from Ömer and Robin.
ProductPage.xhtml - Mojtaba + search&filter by Robin&Ömer
HomePage.xhtml - Mojtaba
aboutPage.xhtml - Allan
loginregister.xhtml - Robin&Ömer
logoutPage.xhtml - Robin&Ömer
paymentResult.xhtml - Allan

Controllers

AccountPageController - Robin&Ömer
AdminController - Robin&Ömer
cartController - Allan&Robin&Ömer
CheckoutController - Allan&Ömer&Robin
LoginController - Robin&Ömer
LogoutController - Robin&Ömer
ProductController - Mojtaba with some help from Ömer.
RegisterController - Robin&Ömer
SessionController - Robin&Ömer (From
<http://mrbool.com/working-with-session-and-filter-in-jsf/34569>)

DAO

AbstractDAO - Everyone
CouponDAO - Robin&Ömer
CustomerDAO - Allan&Mojtaba
ProductDAO - Robin&Ömer
PurchaseDAO - Robin&Ömer

Entity

Coupon - Allan
Customer - Everyone
Product - Robin&Ömer
Purchase - Robin&Ömer

Service

PasswordManager - Robin&Ömer
MessageCenter - Robin&Ömer

Views

AccountBackingBean - Robin&Ömer
AdminBackingBean - Robin&Ömer
CartBackingBean - Robin&Ömer with help from Allan.
CheckoutBackingBean - Allan
ProductBackingBean - Mojtaba with help from Ömer.
RegisterBackingBean - Robin&Ömer

BackingBeanTest

AccountBackingBeanTest - Robin&Ömer
HomePageBackingBeanTest - Mojtaba
AdminBackingBeanTest - Robin&Ömer
CartBackingBeanTest - Robin&Ömer
CheckoutBackingBeanTest - Allan
LoginBackingBeanTest - Robin&Ömer
ProductBackingBeanTest - Allan&Ömer
RegisterBackingBeanTest - Robin&Ömer

DAOTest

CouponDAOTest - Robin&Ömer
CustomerDAOTest - Allan&Mojtaba
ProductDAOTest - Robin&Ömer
PurchaseDAOTest - Robin&Ömer

EntityTest

CustomerTest - Robin&Ömer
ProductTest - Robin&Ömer
PurchaseTest - Robin&Ömer

ModelTest

CartTest - Robin&Ömer

ServiceTest

PasswordManagerTest - Robin&Ömer

CSS

primeflex folder is from (<https://github.com/primefaces/primeflex>)

js

bootstrap js file is from (<https://getbootstrap.com/>)

