**DrankenAutomaat**

**Inleiding**

De drankenautomaat is een simulatie die vanuit gebruikersstandpunt bestaat uit twee delen; één gericht naar klanten en één gericht naar onderhoud.

Het klant gedeelte stelt de voorkant van een drankenautomaat voor. Dranken zoals koffie, thee, soep,… kunnen besteld worden . Afhankelijk van welke drank besteld wordt kan sterkte en/of grootte van de drank ingesteld worden. Betaling gebeurt met behulp van een bankkaart.

Het onderhouds gedeelte is een voorstelling van de achterkant van de drankenautomaat. Dit gedeelte is enkel toegankelijk voor onderhoudspersoneel. Hier kunnen onderhoudstaken worden uitgevoerd zoals het bijvullen van de ingredienten en instellen van een maximum quantity en alarm quantity. Ingredienten kunnen worden bijgevuld tot de maximum quantity. De alarm quantity is een instelling per ingredient die gevolgen heeft voor een notification service. Deze service zorgt er voor dat notifications worden verstuurd als ingredient quantities onder hun alarm quantity komen.

De backend is een standalone applicatie en de frontend is een webapplicatie die gedeployed kan worden in een webserver die java server pages en servlets ondersteund. Voor de reference setup is gekozen voor JBoss 4.2.3. De frontend communiceert met de backend via xml over TCP/IP. Beide delen kunnen op afzonderlijke computers worden geïnstalleerd zolang een TCP/IP verbinding mogelijk is.

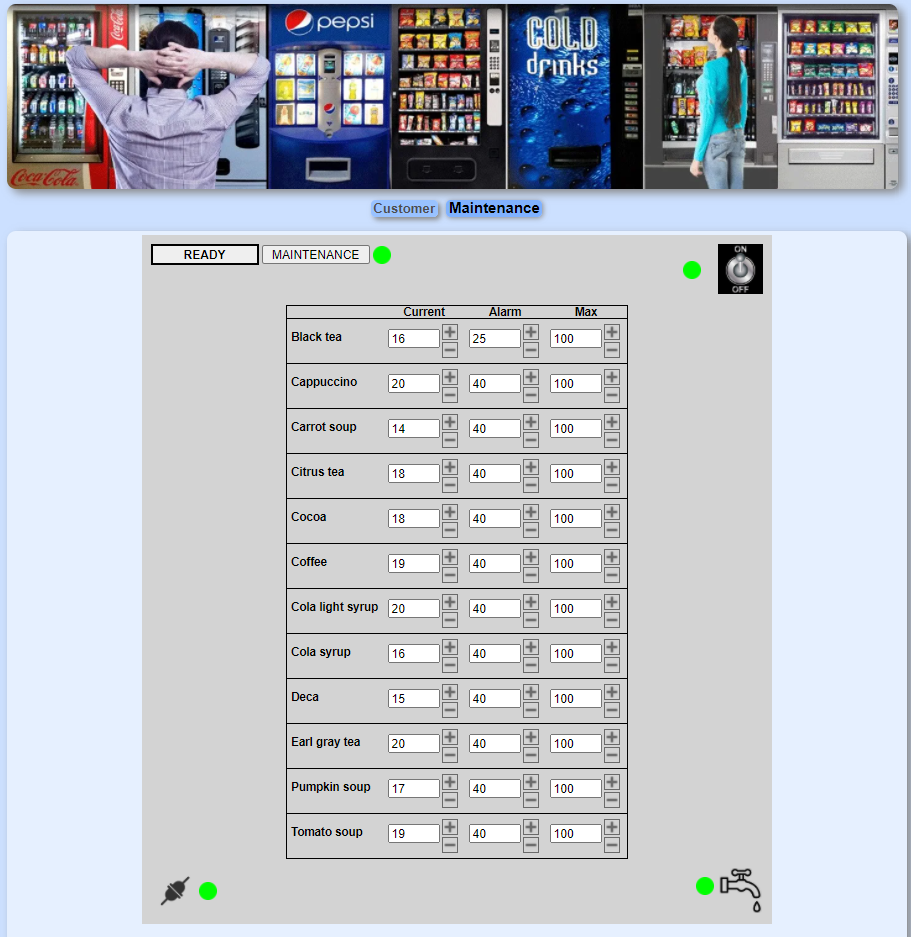
**GUI**

**Klant**



* Bovenaan wordt aangegeven door een status message en een groen of rood lampje of het mogelijk is gebruik te maken van de drankenautomaat:
  + In operation : de automaat is in werking
  + Out of order : het is momenteel niet mogelijk de automaat te bedienen
* Onder de status message kan door te klikken een bankkaart worden ingestoken of verwijderd. De afbeelding en een rood of groen lampje geven aan of een bankkaart is ingebracht. Zonder bankkaart zullen geen drankjes bested kunnen worden.
* De knoppen rondom laten toe een drankje te selecteren. Al ser niet genoeg ingredienten beschikbaar zijn kan het zijn dat sommige knoppen gedisabled zijn. Afhankelijk welk drankje geselecteerd wordt verschijnen verschillende opties :
  + Size : de grootte van het drankje (regular, large en xxl)
  + Flavour : de sterkte van het drankje :
    - + : regular
    - ++ : strong
    - +++ : very strong
* Met behulp van de Prepare button kan de opdracht worden gegeven een drankje te bereiden.
* Een animatie geeft de bereiding weer. Een beker wordt getoond met behulp van een fade in effect. Als de beker volledig zichtbaar is met een groen lampje er naast is het drankje klaar en kan het weggenomen worden.

**Onderhoud**



* Deze pagina is enkel toegankelijk na het ingeven van geldige user credentials van een onderhoudsman (dmc007 / password).
* Links bovenaan kan de drankenautomaat in twee verschillende modi geset worden. De twee verschillende modi zullen enkel beschikbaar zijn als het water en de elektriciteit zijn aangesloten en de drankenautomaat is ingeschakeld :
  + READY : Een groen lampje geeft aan dat de automaat zich in deze modus bevindt.Het klantgedeelte is actief en onderhoud kan momenteel niet uitgevoerd worden
  + MAINTENANCE : Een oranje lampje geeft aan dat de automaat zich in deze modus bevindt. Onderhoud kan momenteel worden uitgevoerd en het klantgedeelte is momenteel gedisabled (Out of order)
  + Een niet brandend lampje geeft aan dat de READY modus en de MAINTENANCE modus niet beschikbaar zijn omdat bijvoorbeeld de elektriciteit niet is ingeschakeld.
* Rechts bovenaan kan de drankenautomaat uit of aan gezet worden. De schakelaar geeft aan of de drankenautomaat aan of uit staat. Ook een groen lampje geeft aan dat de drankenautomaat aanstaat. Een niet brandend lampje geeft aan dat de drankenautomaat is uitgeschakeld.
* Links onderaan kan de drankenautomaat worden aangesloten op het elektriciteitsnetwerk. Een afbeelding van een stekker geeft aan of de drankenautomaat is aangesloten. Ook een groen lampje geeft aan dat de drankenautomaat is aangesloten. Een niet brandend lampje geeft aan dat de drankenautomaat niet is aangesloten.
* Rechts onderaan kan de drankenautomaat worden aangesloten op het water. De afbeelding van een kraantje geeft aan of de automaat is aangesloten of niet. Ook een groen lampje geeft aan dat de drankenautomaat is aangesloten. Een niet brandend lampje geeft aan dat de drankenautomaat niet is aangesloten.
* De tabel in het midden van de pagina kan in MAINTENANCE mode gebruikt worden om ingredienten aan te vullen of te verwijderen, de maximum hoeveelheid en het alarm level per ingredient in te stellen. De + en – knopjes laten toe de hoeveelheden te verhogen of te verlagen. De current quantity kan niet meer zijn dan de maximum quantity. De alarm quantity stelt in hoe lag de current quantity kan zakken voor er notification mails verstuurd worden naar de onderhoudsploeg om aan te geven dat de automaat moet worden bijgevuld.

**Installatie**

Zowel frontend als backend hebben een Java runtime environment nodig : jre 1.6 or higher.

**Backend**

Om de backend te installeren volstaat het de zipfile DrinksMachine.zip te unzippen. De resulterende filestructuur ziet er dan als volgt uit:

* DrinksMachine
  + Lib
    - Log4j.jar
    - Mail.jar
  + Log
  + Application.properties
  + drinksMachine.jar
  + log4j.properties
  + run.bat

De lib folder bevat twee libraries :

* log4J : gebruikt door de applicatie voor logging
* Mail : gebruikt door de applicatie om mails te sturen

In de log folder wordt gelogd.

In application.properties kan de applicatie geconfigureerd worden.

drinksMachine.jar bevat de gecompileerde broncode.

Log4J.properties bevat configuratie voor het gebruikte logging framework.

Met behulp van run.bat kan de applicatie gestart worden.

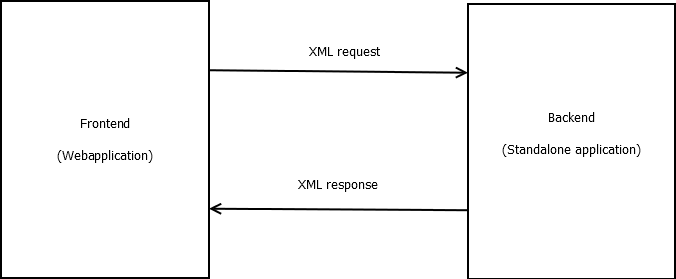
**Frontend**

Om de frontend te installeren volstaat het om de zipfile jboss-4.2.3.GA.zip te unzippen. Dit is de webserver die kan opgestart worden met behulp van \jboss-4.2.3\bin\run.bat. De frontend is gedeployed in de webserver, dus het opstarten van de webserver resulteert automatisch in het opstarten van de frontend. De webserver omvat te veel files om allemaal te bespreken. Enkele relevante zal ik hier extra duiden:

* \jboss-4.2.3.GA\server\default\conf\props\DrinksMachine-roles.properties : user role configuratie in verband met de onderhouds pagina authenticatie
* \jboss-4.2.3.GA\server\default\conf\props\ DrinksMachine-users.properties : user credentials configuratie in verband met de onderhouds pagina authenticatie
* \jboss-4.2.3.GA\server\default\conf\application.properties : frontend configuratie
* \jboss-4.2.3.GA\server\default\conf\jboss-log4j.xml : logging configuratie
* \jboss-4.2.3.GA\server\default\conf\login-config.xml : configuratie in verband met de onderhouds pagina authenticatie
* \jboss-4.2.3.GA\server\default\deploy\DrinksMachineClient.war : de webapplicatie. Dit is een war, een web archive. Hierin bevindt zich de gecompileerde broncode, jsp paginas, css files, javascript files en web applicatie gerelateerde xml configuratie files.

**Implementatie**

**Communication**



De twee componenten communiceren met elkaar via XML over TCP/IP. De messages worden gevormd door twee java classes: request en reply.

1. De frontend instantieert een request en serialiseert dit naar XML.
2. De frontend stuurt de XML door een socket naar de backend.
3. De backend ontvangt de XML en deserialiseert dit naar een request object.
4. De backend behandeld het request en instantieert een reply.
5. De backend serialiseert de reply naar XML en stuurt dit door het de socket naar de frontend.
6. De frontend ontvangt de XML, deserialiseert dit naar een reply object en behandelt de reply.

**Logging**

The log4j framework is used for logging.

* Frontend :
  + \jboss-4.2.3.GA\server\default\log\boot.log : the webserver startup is logged in this location
  + \jboss-4.2.3.GA\server\default\log\server.log : once the webserver is started logging ends up in this location
  + \jboss-4.2.3.GA\server\default\conf\jboss-log4j.xml : logging can be configured here. Currently the loggin threshold is set to INFO and two appenders are configured : a file appender and a console appender.
* Backend :
  + \DrinksMachine\log : logging ends up in this location
  + \DrinksMachine\log4j.properties : logging can beconfigured here. Currently the loggin threshold is set to INFO and two appenders are configured : a file appender and a console appender.

**Documentation**

Code documentatie werd gegenereerd met javadoc. Twee zipfiles met documentatie zijn beschikbaar :

* backendDocumentation.zip : kan na unzippen bekeken worden in een browser door index.html te openen
* frontendDocumentation.zip : kan na unzippen bekeken worden in een browser door index.html te openen

**Authentication**

Username/password authenticatie wordt gebruikt om het openen van de achterkant van drankenautomaat met behulp van een sleutel door onderhoudspersoneel te simuleren. Hiervoor wordt BASIC configuratie gebruikt. Op verschillende locaties werd dit geconfigureerd :

* DrinksMachineClient\WebContent\WEB-INF\jboss-web.xml
* DrinksMachineClient\WebContent\WEB-INF\web.xml
* \jboss-4.2.3.GA\server\default\conf\login-config.xml
* \jboss-4.2.3.GA\server\default\conf\props\DrinksMachine-roles.properties
* \jboss-4.2.3.GA\server\default\conf\props\DrinksMachine-users.properties

**Web technology**

Requests vanuit de browser worden in de web applicatie afgehandeld met behulp van servlets. De html pagina’s maken gebruik van Java Server Page technology zodat scriptlets gebruik gemaakt kan worden van scriptlets om het aanmaken van de html pagina’s te vergemakkelijken. De gebruiker interacties in de browser worden doorgestuurd met behulp van javascript en XMLHTTPRequests. Naar dit laatste word took wel eens verwezen als AJAX : Asynchronous Javascript and XML. Deze technologie laat het toe requests te sturen naar de webserver en ook replys te krijgen zonder de pagina te submitten. Dit resulteert in een vlottere user experience.

**Configuratie**

Zowel frontend als backend zijn extern configureerdbaar.

* Frontend :
  + \jboss-4.2.3.GA\server\default\conf\application.properties :
    - Host : de host waarop de backend bereikbaar is
    - Port : de gebruikte poort voor de communicatie tussen frontend en backend
* Backend
  + \DrinksMachine\application.properties :
    - Port : de gebruikte poort voor de communicatie tussen frontend en backend
    - Path : het pad naar de persistence store
    - To : het emailadres waarnaar eventuele notifications verstuurd worden
    - enableNotifier : enable of disable het versturen van email notifications
    - password : het paswoord van de te gebruiken email account
    - sleep : het aantal uren tussen het eventueel versturen van notification emails

**Webserver**

Als webserver werd gekozen voor JBoss 4.2.3 GA. Deze webser is een veel gebruikte open-source Java EE gebaseerde application server runtime platform en wordt ondersteund door Red Hat. Voor educational purposes is deze webserver free to use.

**Persistence**

Om de status van de drankenautomaat te bewaren na het afsluiten van de software had een database dienst kunnen doen. De bij te houden data is echter niet van die omvang dat dit nodig is. Het gebruik maken van JAXB heeft gezorgd voor een lightweight oplossing. Dit is een module te vinden in Java 1.6. Het laat toe java objecten te serializeren en deserializeren van en naar xml. De xml kan indien gewenst, zoals in dit geval, opgeslagen worden op de harde schijf.

**Notifier**

De drankenautomaat bidet de mogelijkheid email notifications te laten versturen als de hoeveelheid van ingredienten onder een in te stellen threshold komen. Dit is in te stellen op de onderhoudspagina. Op deze manier kunnen de ingredienten tijdig worden aangevuld en komen klanten niet voor een lege drankenautomaat te staan. Het versturen van de mails maakt gebruikt van javax.mail. Dit is een gebruiksvriendelijke module te vinden in Java 1.6. Al wat nodig is, is een werkende emailaccount en beperkte configuratie om mails te kunnen versturen.

**Dankwoord**

Hierbij wil ik nog mijn leraars bedanken met in het bijzonder Jan Beurghs zonder wie deze drankenautomaat niet mogelijk zou geweest zijn.