NG-STAR Requirements

Author: Irish Medina
Last Updated: August 15, 2014
Please send any questions to irish.m.medina@gmail.com

The following describes the steps that should be taken in order to prepare for the development of NG-STAR.

1. Contact Information

Our clients for this project are Irene Martin (Head, Streptococcus and STI Unit, NML) and Walter Demczuk (Biologist, NML). They can be reached at the following email addresses:

irene.martin@phac-aspc.gc.ca, walter.demczuk@phac-aspc.gc.ca

Questions and Answers from Irene and Walter can be found here:

https://drive.google.com/folderview?id=0B4Vx2hKNcWEaY3htWGVmQ2FvV1k&usp=sharing

My contact information is also located here:

Irish Medina
Student, Unversity of Manitoba, Department of Computer Science
Email Address: irish.m.medina@gmail.com

Developed NG-STAR from January 2014 to August 2014.
There are no other developers in this project, besides help from mentors.

1. Understand the background behind the NG-STAR project:

https://drive.google.com/folderview?id=0B4Vx2hKNcWEaXy1saGRzQnJ1UUU&usp=sharing

https://drive.google.com/folderview?id=0B4Vx2hKNcWEaNmp0NmNzeWcwaXM&usp=sharing

The major topics surrounding the background that are important to understand are antimicrobial resistance (what it is, why does it happen, why is it such a problem), how N. Gonorrhoeae relates to antimicrobial resistance, what MLST (Multi-locus sequence typing) is, why we use MLST, what alleles, sequence types and loci are in MLST.

If you do not have a biology background, it is strongly recommended that you review the concept behind DNA transcription and translation. You should know what genes are and what a genome is. It may be helpful to know what the Human Genome Project is.

https://www.youtube.com/watch?v=_-vZ_g7K6P0 (DNA)
https://www.youtube.com/watch?v=Kzgnl5-8WAk (DNA Transcription, Translation, Replication)
https://www.youtube.com/watch?v=41_Ne5mS2ls (DNA Transcription and Translation)
https://www.youtube.com/watch?v=MvuYATh7Y74 (Human Genome Project)

Here are some additional links to background information that may be helpful:

http://en.wikipedia.org/wiki/Multilocus_sequence_typing

[In AMR (antimicrobial resistance) we don't refer to alleles as “housekeeping genes” since these genes are highly recombinant for bacteria such as N. Gonorrhoeae. Instead, we refer to these genes as “antimicrobial resistant genes”]

There are a few existing MLST web applications, but these are not concerned with antimicrobial resistance. It may be helpful to try out these exiting web application to see how MLST works.

http://pubmlst.org/software/database/bigsdb/
http://www.mlst.net/
http://www.ng-mast.net/ (this is an MLST scheme for N. Gonorrhoeae, however, it only includes 2 genes and these genes are not associated with antimicrobial resistance)

The programming languages used to develop NG-STAR are:

a. Perl
b. HTML
c. jQuery
d. SQL (useful when using the MySQL client or MySQL Workbench, however SQL statements are not present in the data access layer of NG-STAR since we're using the DBIx::Class ORM).
e. Python (only for Selenium WebDriver integration tests).

The development tools used to develop NG-STAR are:

Catalyst MVC Framework – A web application framework.
DBIx::Class (DBIC) – An ORM (Object Relational Mapper).
Bootstrap 3 – A CSS framework.
git – Distributed version control software.
GitLab – a Git repository web-based hosting service (http://gitlab.corefacility.ca)
MySQL – A database management system.
MySQL Workbench – A visual tool for MySQL.
vim – A text editor (optional).
Terminator – A terminal emulator that supports tabs and split views (optional).
Selenium WebDriver – A browser automation tool for integration tests.

No IDE was originally used to develop NG-STAR (vim with Terminator was used). It is up to the developer to use what they want.

2. If you are not familiar with Perl, HTML, jQuery or SQL then you will have to review them. Of course, these can be learned as you progress through the tutorials and as you work on the project.

3. You will need to install all development tools specified (except for Bootstrap 3 which is already included in the NG-STAR GitLab repository, vim and Terminator which are optional and GitLab which can be accessed from http://gitlab.corefacility.ca using your LDAP credentials). Most development tools can be installed using `apt-get` or by consulting installation documentation.

Before installing any software, ensure that you have updated the repositories:

```
sudo apt-get update
```

You must have all required dev tools (gcc, make, ...). To do this, run:

```
sudo apt-get install build-essential
```

To install Catalyst MVC, you can refer to the installation documentation here: http://wiki.catalystframework.org/wiki/installingcatalyst

In general, to install Catalyst you will need:

1. perl 5.8.6 or higher
2. Catalyst::Runtime
3. Catalyst::Devel

First, ensure that you have perl version 5.8.6 or higher by running the following command:

```
perl -v
```

If you don't have perl version 5.8.6 or higher then you can upgrade by running the following commands:

```
sudo cpan
cpan[1]> upgrade
```

To download and install the Catalyst::Runtime and Catalyst::Devel modules, you can use cpan minus. To install cpan minus, run the following command:

```
sudo apt-get install cpanminus
```

To install Catalyst::Runtime and Catalyst::Devel, run the following commands:

```
sudo cpanm Catalyst::Runtime
sudo cpanm Catalyst::Devel
```

To install DBIx::Class, run the following command:

```
sudo cpanm DBIx::Class
```

You should also install all recommend modules for Catalyst using cpanm, as specified here: http://wiki.catalystframework.org/wiki/recommended_plugins

Bootstrap 3 can be found at http://getbootstrap.com/

The NG-STAR repository already includes Bootstrap. We are using the compiled and minified CSS, JavaScript and fonts with no docs or original source files included.

You can install git by running the following command:

```
sudo apt-get install git
```

GitLab can be accessed from http://gitlab.corefacility.ca
To install mysql client and mysql server, run the following command:

```
sudo apt-get install mysql-client mysql-server
```

To install MySQL Workbench, run the following command:

```
sudo apt-get install mysql-workbench
```

Optionally, you can install vim and Terminator by running:

```
sudo apt-get install vim
sudo apt-get install terminator
```

To install Selenium WebDriver, follow these instructions:

You will need pip, which is a python package manager (similar to cpan in perl). To install pip, run the following command:

```
sudo apt-get install python-pip
```

Install Selenium WebDriver by running the following command:

```
sudo pip install selenium
```

For more information, refer to these instructions here: http://selenium-python.readthedocs.org/installation.html

Since we are using the Chrome driver in our integration tests, you will need to download and install that as well.

Download Chrome driver here:

https://sites.google.com/a/chromium.org/chromedriver/home

Choose version 2.10 (or the latest version)

Go to the directory that contains the Chrome driver file that you downloaded.

Unzip the Chrome driver file. You should get a file called 'chromedriver'.

You must move 'chromedriver' to /usr/bin. To do this, run the following command in the directory where 'chromedriver' is:

```
sudo mv chromedriver /usr/bin
```

4. Learn the Catalyst MVC Framework by following the tutorials provided on their website: https://metacpan.org/pod/Catalyst::Manual::Tutorial::02_CatalystBasics

It is highly recommended that you work through these tutorials before proceeding:

2 Catalyst Basics
3 More Catalyst Basics
4 Basic CRUD
5 Authentication
6 Authorization
9 Advanced CRUD (FormHandler)
https://metacpan.org/pod/Catalyst::Manual::Tutorial::09_AdvancedCRUD::09_FormHandler
You may want to try installing and running a FormHandler example:
https://github.com/gshank/formhandler-example
10 Appendices (includes a tutorial on how to switch from SQLite to MySQL).

You may skip the other sections as they include features that are not utilized in the NG-STAR project.

5. Learn all other development tools and programming languages. Here is some useful information:

a. DBIx::Class has a fairly difficult learning curve. It is recommended that you read through the documentation provided below and understand what an ORM (Object-relational mapper) is. The reason that we use an ORM is to avoid SQL Injection attacks (since an ORM generates parametrized SQL statements from object-oriented code), and more importantly provides an easy way to switch out different database management systems.

Here is a very good presentation:
http://www.slideshare.net/ranguard/dbixclass-beginners-presentation

More introductory documentation:
https://metacpan.org/pod/DBIx::Class::Manual::Intro
https://metacpan.org/pod/DBIx::Class::Manual::Example
http://search.cpan.org/~jrobinson/DBIx-Class-Tutorial-0.0001/lib/DBIx/Class/Tutorial/Part1.pod
http://search.cpan.org/~jrobinson/DBIx-Class-Tutorial-0.0001/lib/DBIx/Class/Tutorial/Part2.pod

The DBIx::Class Cookbook is very useful during development as it provides many examples on usage:

https://metacpan.org/pod/DBIx::Class::Manual::Cookbook

b. If you are not familiar to web development (pertaining to the client-side), then these tutorials on Codecademy may be useful to follow:

http://www.codecademy.com/tracks/web (optional)
http://www.codecademy.com/tracks/jquery (some portions of NG-STAR include jQuery)
http://www.codecademy.com/skills/make-a-website (this tutorial includes a section on how to use Bootstrap)
http://www.codecademy.com/skills/make-an-interactive-website (a good tutorial on how to use jQuery)

c. For MySQL, it should be adequate to know basic SQL commands (such as insert, delete and update). When using the MySQL command line client, you will need to know how to log in (as a user and as root), be familiar with the commands `show databases;` and `use table_name_here;`, as well as creating users with the appropriate privileges.

d. Instead of writing out SQL statements and running it as a script to generate database tables (which is error prone), we use MySQL Workbench to create ER Diagrams and subsequently generate database tables based on this model by “Forward Engineering” our model.

http://dev.mysql.com/doc/workbench/en/wb-forward-engineering-live-server.html

If you want to make changes to your database schema, you can do so by making changes to its associated ER Diagram. You can generate this ER Diagram by “Reverse Engineering” your database.

http://dev.mysql.com/doc/workbench/en/wb-reverse-engineer-live.html

If you already have an ER Diagram and you want to update it, you can do so by "Synchronizing".

http://dev.mysql.com/doc/workbench/en/wb-database-synchronization.html

If you would like to query the database from MySQL Workbench (instead of from the MySQL command line client), you can do so by going to Database in the Toolbar and clicking on Query Database.

http://dev.mysql.com/doc/workbench/en/wb-sql-editor-query-panel.html
More examples of using MySQL Workbench is provided in the Allele Database Tutorial.

5. For git and GitLab, you should be familiar with what vesion control software is.

Here is a good git tutorial:
https://www.atlassian.com/git/tutorial/git-basics

Here is a good git cheat sheet:
http://rogerdudler.github.io/git-guide/

6. We use Selenium WebDriver using the Python binding for Integration Tests.

Documentation is provided here:
http://selenium-python.readthedocs.org/index.html

It is recommended that you learn how to use Selenium WebDriver after you have learned everything else.