myVCF Documentation

Release stable

Oct 09, 2017

Contents

1 Want to try myVCF? 3

2 Documentation contents 5

2.1 How to install myVCF 5
2.2 Setup the application 10
2.3 myVCF features 15
2.4 FAQ 25
2.5 How to cite myVCF 25

Welcome to **myVCF** manual page!

myVCF is a user-friendly platform that helps end-users, without programming skills, to analyze and visualize mutations in an easy and flexible manner. Helping decision making for further downstream analysis.

myVCF will manage VCF (Variant Call Format) files (the standard format for storing NGS mutations data) deriving from different NGS applications (Whole Exome/Genome sequencing, Public database...)

myVCF will help end-users to browse and analyze VCF coming from exome and targeted sequencing projects. myVCF can handle multiple-sample VCF and multiple projects can be created as separate environment in order to manage different VCFs with the same application.

CHAPTER 1

Want to try myVCF?

You can download **myVCF** package from:

- Project homepage (.zip and tar.gz)
- GitHub project (cloning the project)

and follow the instructions contained in the *installation page*

CHAPTER 2

Documentation contents

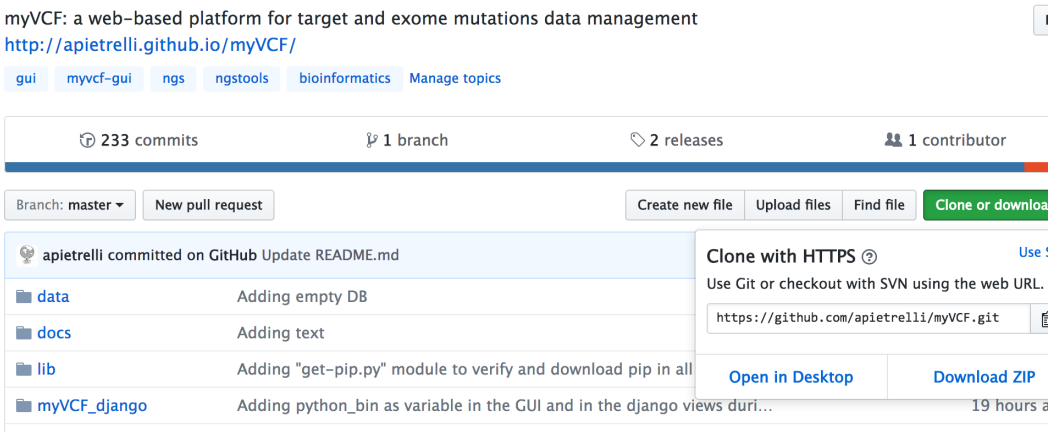
How to install myVCF

Download myVCF

You can download **myVCF** package from:

- **Compressed ZIP package**

1. Go to myVCF homepage
2. Click on Clone or Download button



3. Click on **Download ZIP**
4. Extract the compressed file in your working directory
5. At the end of the process you will have a directory named `myVCF-master/` containing the desktop application

- `git` **command line**

If you have GitHub installed on your computer, you can clone the project directly on your working directory

1. Open the terminal and type:

Note: For MAC users, you can find the terminal app by searching through Spotlight and type `terminal` and click on the application

```
$> cd path/to/working/dir
$> git clone https://github.com/apietrelli/myVCF.git
```

The command will create a directory named `myVCF/` containing the desktop application

Note: To download `git` tool for Unix/MAC operating systems

```
# Ubuntu/Debian Unix OS
$> sudo apt-get install git
# MAC
$> brew install git
```

for Windows users, you can download the git software from the Git homepage and use the same command as for Unix/MAC user by using GitBASH

Warning: Remember the path to myVCF application, as it is necessary for installation and *VCF file loading*

Installation requirements

The application is developed using the Python/Django framework and the sqlite database platform. Please verify the installation of `python2.7` and `sqlite` on your computer.

Python 2.7

myVCF tool is based on **Python 2.7** language. Please verify that you have `python` installed.

If you are not sure or you need to install it, please follow the notes below about the installation depending on your operating system.

Unix (Ubuntu/Debian system)

Using the terminal, install `python2.7` using `apt-get`

```
$> sudo apt-get install python2.7
```

MAC

Open the terminal and install `python2.7` with `brew`

Note: You can find the shell terminal in MAC OS by typing `terminal` in the Spotlight textbox and click on the application.

```
# Terminal application
$> brew install python2.7
```

You can test the installation in the terminal

```
$> python
Python 2.7.5 (default, Mar  9 2014, 22:15:05)
[GCC 4.2.1 Compatible Apple LLVM 5.0 (clang-500.0.68)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>>quit()
```

Windows

You can download the `python2.7` package from the Python project site

Follow the installation process pointing out this two requirements to let myVCF full compatible with your system:

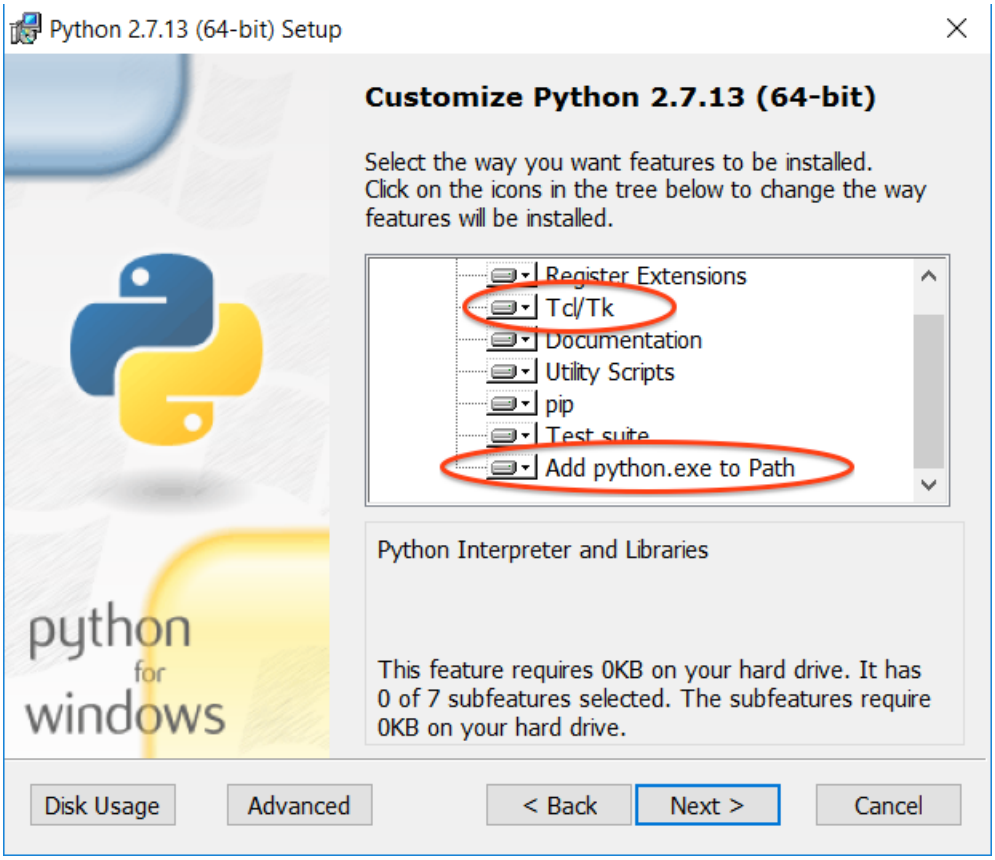
- By default Python2.7 will be installed in `C:\Python27`. Please **DO NOT** modify the Python path and leave the default installation destination directory.

> **Warning:** Please download the **Python2.7** package **NOT** Python3.x

- Please verify that the options:
  - **Add Python to PATH**
  - **Tcl/Tk**

are selected during the installation step.

This will allow the **myVCF_GUI.py** launcher to be functional with no errors.



sqlite

The storage of VCF file has been implemented by using `sqlite` as the backend database. This cross-platform solution allows the end-user to workaround some complex configuration setups which are mandatory with other database system.

Please follow these instructions to install `sqlite` according to your operating system

Unix (Ubuntu/Debian system)/MAC

1. Open the `terminal`
2. Install `sqlite3` package

```
# Ubuntu/Debian Unix OS
$> sudo apt-get install sqlite3
# MAC OS
$> brew install sqlite3
```

3. Launch `sqlite3` from the shell

```
$> sqlite3
SQLite version 3.7.13 2012-07-17 17:46:21
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite>
# Quit from the sqlite3 shell
sqlite> .q
```

Windows

We tested different version of Windows (XP, 7, 10) and in all the Windows systems the `sqlite` library was already installed by default.

If you have troubles in launching myVCF application, follow this procedure to install the `sqlite` necessary files.

1. Go to the `sqlite` web site https://sqlite.org/download.html and download precompiled binaries from the Windows section.
   - **sqlite-dll-win32-x86-*.zip**

   or

   - **sqlite-dll-win64-x64-*.zip**

> **Warning:** Check what Windows version you have installed (32 or 64 bit) on your computer to correctly download the right `sqlite3` package from the web site
>
> To check your system version click on:
>
> **Start > Control panel > System**
>
> and check the version.

2. Unpack the `.zip` file and follow the default installation instructions

Python library dependecies

Now that all the major components have been installed, lets proceed with the last step of the installation process regarding Python library dependencies.

Install packages with myVCF_GUI

The easiest way to satisfy the myVCF Python dependencies is to use the myVCF GUI.

1. Open the GUI menu by double-clicking the icon relative to your system for launching the GUI
2. Click on the button **"Install packages"**

3. The system will install all the dipendencies to start **myVCF** properly

Install packages with terminal

If the `python2.7` installation doesn't fail, you should have also installed `pip` which is the Python command for library installation.

Now we are going to install all the dependencies using just one command-line using `pip`

- **Unix (Ubuntu/Debian system)/MAC**

1. Open the terminal
2. Go to `myVCF/` directory
3. Execute this command:

```
pip install -r requirements.txt
```

Verify the installation by typing:

```
python manage.py shell
```

If you see something like..

```
Python 2.7.5 (default, Mar  9 2014, 22:15:05)
Type "copyright", "credits" or "license" for more information.
>>>
```

..everything went well! :) Now exit from the python shell.

```
>>> quit
```

- **Windows**

1. Open the MS-DOS prompt (`cmd.exe`)

Note: To open CMD shell in Windows click on

Start > type on the search box "**cmd**" > click on **cmd.exe**

---

2. Go to the `myVCF/` directory
3. Execute this command:

```
# MS-DOS Prompt
$> C:\Python27\python.exe pip -m install -r requirements.txt
```

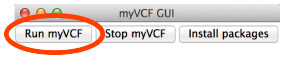
Warning: If you followed the *Python 2.7 Windows installation chapter*, you should have all the Python command in `C:/Python27/`

Launch the application

Finally, you're ready to start the application:

With GUI

- Open the **myVCF** GUI:
  1. Double-click on **myVCF_GUI.py** (Windows)
  2. Double-click on **myVCF_launcher** (MAC/Unix)
- Click on **"Run myVCF"**



- Wait few second for browser loading the homepage

Note: If you are on Windows and the double-click on **myVCF_GUI.py** does not open the application, try to open the file with

Right mouse click -> Open with -> Choose default program

and browse the directory to find Python executable `python.exe` in `C:/Python27/`

With Terminal

```
# UNIX on terminal
$> cd path/to/myVCF/
$> python manage.py runserver

# Windows on MS-DOS cmd
$> cd C:\path\to\myVCF\
$> C:\Python27\python.exe manage.py runserver
```

Visit http://127.0.0.1:8000/ in your browser to see how it looks.

Setup the application

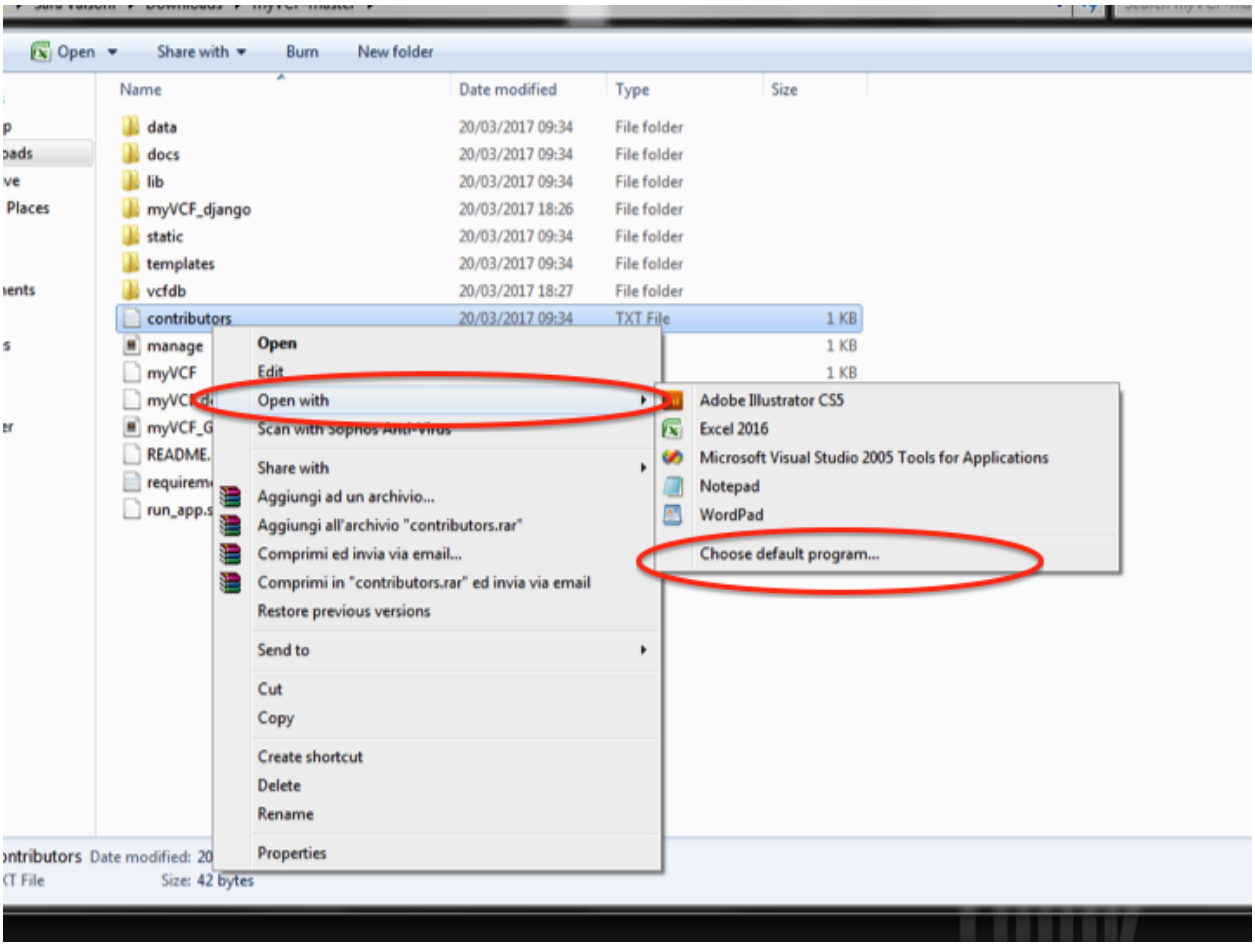
Now you are ready to load all your VCF files and start to analyze your data with myVCF.

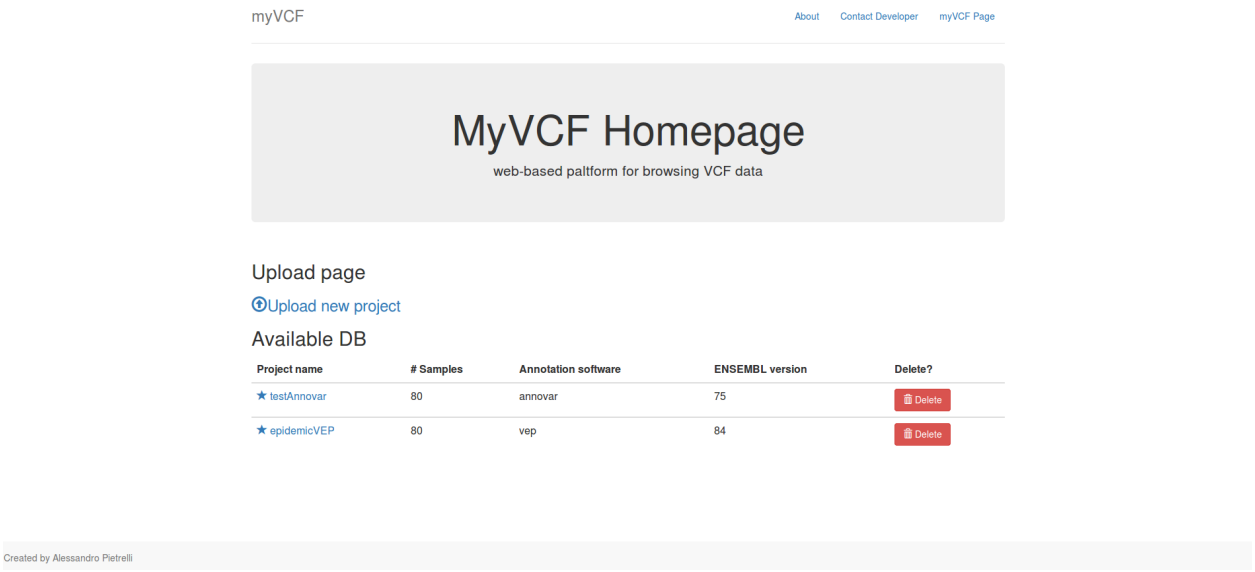
myVCF is designed for human annotated VCF files, but it accepts any type VCF coming from different species with or without annotations.

For more information about **not-annotated** or **non-human** VCF file, please follow this *link*

myVCF manages **annotated VCF** files with specific fields that are mandatory in order to load and visualize the data correctely.

To verify if your `.vcf` file is compatible with myVCF, please read the following section.


Open
Share with
Burn
New folder
Name
Date modified
Type
Size
data
docs
lib
myVCF_django
static
templates
vcfdb
contributors
manage
myVCF
README.
Open
Edit
Open with
Share with
Aggiungi ad un archivio...
Aggiungi all'archivio "contributors.rar"
Comprimi ed invia via email...
Comprimi in "contributors.rar" ed invia via email
Restore previous versions
Send to
Cut
Copy
Create shortcut
Delete
Rename
Properties
Adobe Illustrator CS5
Excel 2016
Microsoft Visual Studio 2005 Tools for Applications
Notepad
WordPad
Choose default program...


myVCF
About
Contact Developer
myVCF Page
MyVCF Homepage
web-based paltform for browsing VCF data
Upload page
Upload new project
Available DB
Project name
Samples
Annotation software
ENSEMBL version
Delete?
testAnnovar
80
annovar
75
Delete
epidemicVEP
80
vep
84
Delete
Created by Alessandro Pietrelli

VCF fields and requirements

myVCF can read VCF files deriving from **Annovar** or **VEP** annotation systems. These software are the most common tools used for VCF annotation after the SNP calling step.

Note: If you are not sure if your VCF file respect the mandatory field and requirements, try to load it by following the *Load new data section*

Let's define which are the mandatory fields that a VCF must contains for myVCF

- Since myVCF is a tool to browse and visualize mutations genotyped with NGS technologies, the VCF file **must** contain at least 1 genotyped sample

See example below:

```
...
##contig=<ID=17,length=81195210,assembly=b37>
##contig=<ID=18,length=78077248,assembly=b37>
##contig=<ID=19,length=59128983,assembly=b37>
##contig=<ID=20,length=63025520,assembly=b37>
##contig=<ID=21,length=48129895,assembly=b37>
##contig=<ID=22,length=51304566,assembly=b37>
##contig=<ID=X,length=155270560,assembly=b37>
##contig=<ID=Y,length=59373566,assembly=b37>
##contig=<ID=MT,length=16569,assembly=b37>
##INFO=<ID=Func_ensGene,Number=.,Type=String,Description="Func_ensGene annotation␣
↪provided by ANNOVAR">
##INFO=<ID=Gene_ensGene,Number=.,Type=String,Description="Gene_ensGene annotation␣
↪provided by ANNOVAR">
##INFO=<ID=GeneDetail_ensGene,Number=.,Type=String,Description="GeneDetail_ensGene␣
↪annotation provided by ANNOVAR">
##INFO=<ID=ExonicFunc_ensGene,Number=.,Type=String,Description="ExonicFunc_ensGene␣
↪annotation provided by ANNOVAR">
#CHROM        POS     ID      REF     ALT     QUAL    FILTER  INFO    FORMAT  Sample1
1     762273  rs3115849       G       A       123.7   LowQual AC=2;AF=1;AN=2;Func_
↪ensGene=ncRNA_exonic;Gene_ensGene=ENSG00000225880;GeneDetail_ensGene=.;ExonicFunc_
↪ensGene=. GT:AD:DP:GQ:PL  1/1:0,63:63:99:1550,188,0
```

This is part of a VCF file in which one sample has been genotyped (Sample1) for one mutation.

- For **Annovar** annotated VCF files, the mandatory fields would be:
  1. Gene_ensGene
  2. ExonicFunc_ensGene
- For **VEP** annotated VCF files, the mandatory field would be:
  1. CSQ

  This field is added by default during VEP annotation

Note: To verify the necessary fields for the annotation part, you should see in the HEADER part of the VCF file the following lines:

```
# Annovar fields
##INFO=<ID=Gene_ensGene,Number=.,Type=String,Description="Gene_ensGene annotation␣
↪provided by ANNOVAR">
##INFO=<ID=ExonicFunc_ensGene,Number=.,Type=String,Description="ExonicFunc_ensGene␣
↪annotation provided by ANNOVAR">
```

```
# VEP: CSQ field
##INFO=<ID=CSQ,Number=.,Type=String,Description="Consequence annotations from Ensembl␣
↪VEP. Format: Allele|Consequence|IMPACT|SYMBOL|Gene|Feature_
↪type|Feature|BIOTYPE|EXON|INTRON|HGVSc|HGVSp|cDNA_position|CDS_position|Protein_
↪position|Amino_acids|Codons|Existing_variation|DISTANCE|STRAND|VARIANT_CLASS|SYMBOL_
↪SOURCE|HGNC_ID|CANONICAL>
```

or copy the VCF in `myVCF/data/VCFs/` directory and try to load the VCF through the *Upload page*

How to annotate your VCF

If you don't have the genomic/transcript annotation for your VCF file, or the VCF is not suitable for myVCF please consider to annotate it using the following instructions.

How to install the annotation tools

Annovar

The installation of Annovar is very well-described on ANNOVAR Manual pages

Since *Annovar* is a `perl` script, the software can be run on different operating systems including Unix and Windows.

VEP

The installation of VEP is described on VEP main page

Please follow the instructions below to install the software based on your operating system.

Unix (Ubuntu/Debian system)/MAC

For UNIX/MAC users, there is a tutorial available that describes the download and the installation steps in a simple manner.

Windows

Please follow these instuctions to install and configure VEP for Windows.

Note: The easiest way is the **Cygwin** installation procedure.

Launch the code for annotation

Here we reported the *minimum code* to run a correct annotation thai is compatible with myVCF. The tutorial contains both the *Annovar* and *VEP* annotation procedure.

For **Windows users** please launch the commands using **Cygwin** downloaded in the previous section or **CMD shell** (*find CMD*)

Annovar

1. Download the ENSEMBL transcript reference database required for myVCF compatibility.

```
# Download the ensembl DB (example: hg19)
# buildver = hg19/hg38 depending on what reference assembly you used during the read␣
↪mapping
table_annovar.pl -downdb -webfrom annovar -buildver hg19 ensGene

## Optional but useful annotation
# dbSNP147
table_annovar.pl -downdb -webfrom annovar -buildver hg19 avsnp147
# dbnsfp30a - non-synonimous variants annotation compendium (it takes lot for␣
↪download)
#http://annovar.openbioinformatics.org/en/latest/user-guide/filter/#ljb42-dbnsfp-non-
↪synonymous-variants-annotation
table_annovar.pl -downdb -webfrom annovar -buildver hg19 dbnsfp30a
```

2. Launch the annotation process. The command line is based on Annovar tutorial.

```
# Launch the annotation
table_annovar.pl example/ex2.vcf humandb/ -buildver hg19 -out myanno -remove -
↪protocol ensGene,avsnp142,dbnsfp30a -operation g,f,f -nastring . -vcfinput
```

Note: To download additional databases to enrich the annotation for your mutations, please see this link and modify the **Launch the annotation** command line by adding the name of the database in `-protocol` and an `f` in `-operation` for every database you want to add.

VEP

A simpler approach than Annovar, can be to use the following command. Once downloaded the tool and the human assembly containing the annotations, launch this command to annotate your VCF file

```
perl variant_effect_predictor.pl -i example.vcf --cache --force_overwrite --vcf -o␣
↪example_VEP.vcf
```

At the end of the process you will have file named `example_VEP.vcf` with all the information suitable for myVCF

Load Data

In myVCF package there are two annotated VCF files that you can use for a trial run. This can be done by loading these files directely from the myVCF upload page by clicking on the dropdown menu **VCF File**:

- mini_annovar.vcf (annotated with *Annovar*)
- mini_vep.vcf (annotated with *VEP*)

These files reported ~ 1000 mutations in 80 samples and they are stored in `/path/to/myVCF/data/VCFs`

1. Copy/move the VCF files you want to load into myVCF in `/path/to/myVCF/data/VCFs`
2. Launch the application (See how to *launch the app*) and load http://127.0.0.1:8000/ in your browser
3. Click on the **Upload new project** link in the myVCF homepage

4. Give a name to the project and select the VCF to load

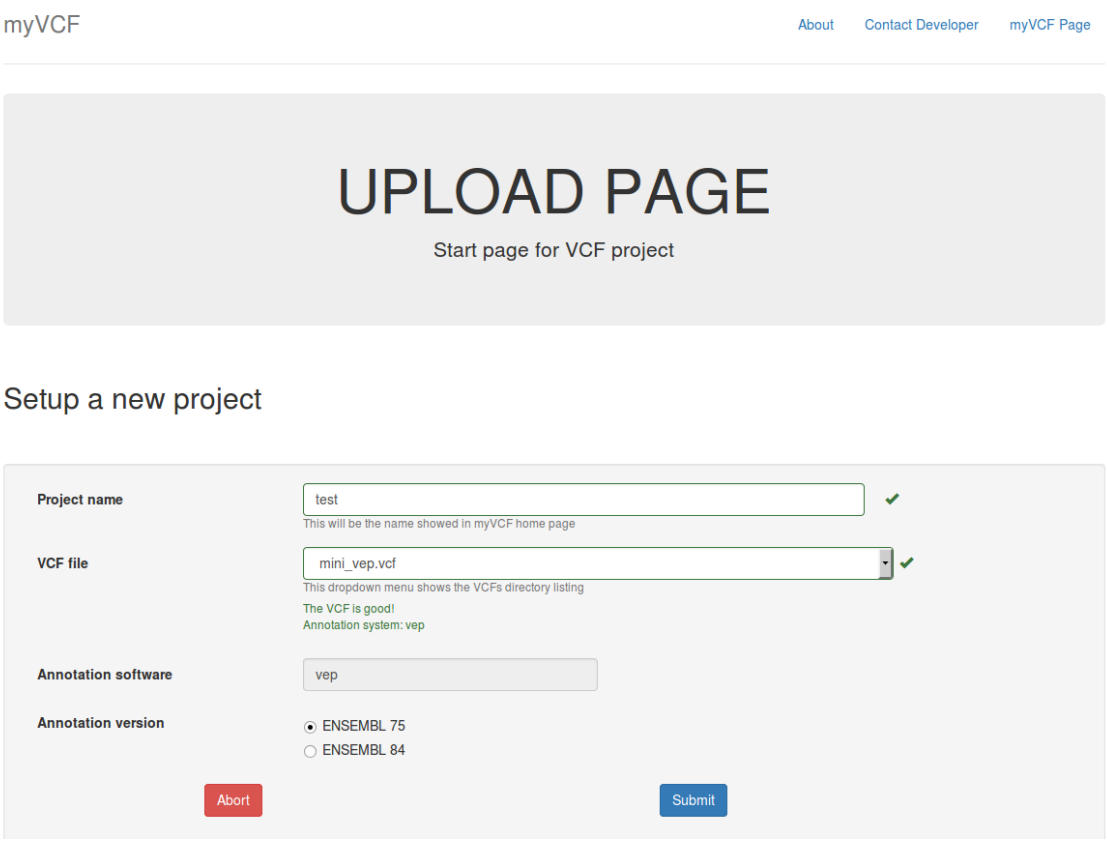


Fig. 2.1: New project upload page example. `mini_VEP.vcf` is the mutation file to upload into `test` project using `ENSEMBL75` as the transcript reference.

Note: If you don't find your VCF in the dropdown menu, please verify that you have copied the file into the directory `myVCF/data/VCFs` and restart the application

---

5. Click on submit button for saving the project

Warning: Don't panic if the saving process takes very long time. Do **NOT refresh** the page until the **Upload completed** page will appear.

myVCF features

myVCF is designed as a tool for browsing and visualizing mutational data coming from NGS technologies, including Whole-Exome and -Genome sequencing as well as target resequencing.

Several features have been implemented to help the end-user in the navigation and the exploration of his project. In the next paragraphs you will find the description of principal features available in myVCF.

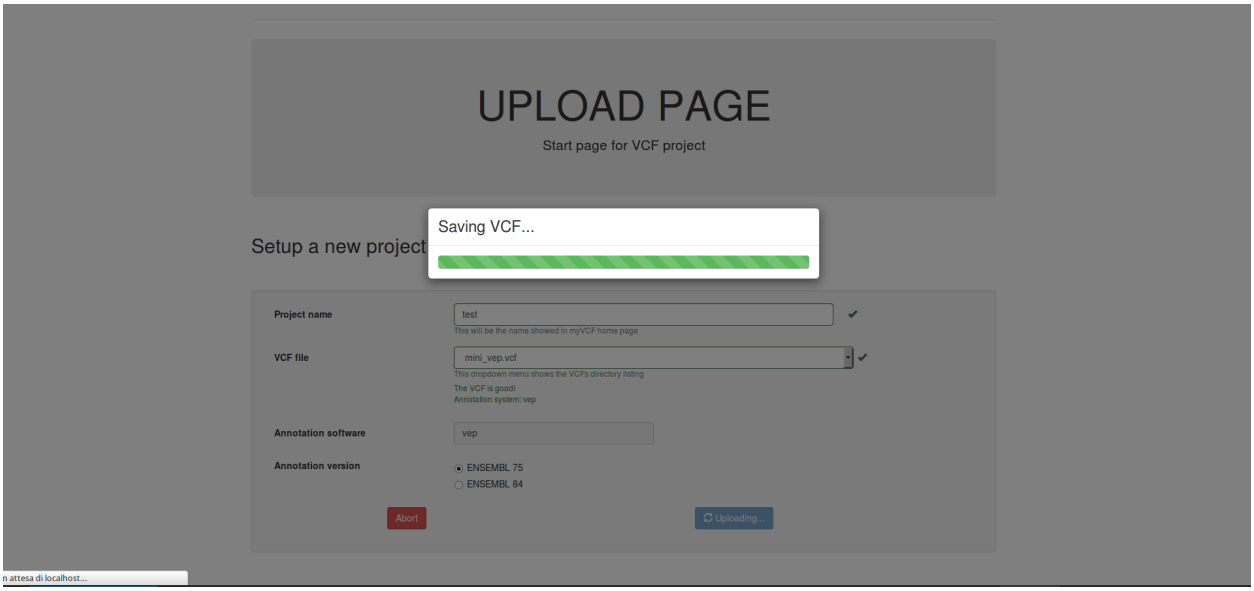


Fig. 2.2: The storing process of very large VCF files (above 50 MB) will take a long time.

How to query a project database?

The search engine in myVCF is very versatile. Once you are in a project homepage, you can query the database by searching for:

1. Gene name (Official Gene Symbol)
2. Genomic region (1:20000-200100)
3. dbSNP ID (rs324239)
4. Variant (1-456783-456783-A-T)

Gene/Region view

Basic gene/region search will generate a **Gene page** composed by:

- **Table** containing the mutations found in the gene/region
- **Mutation plot** showing the distribution of the mutations grouped by their functional consequence.

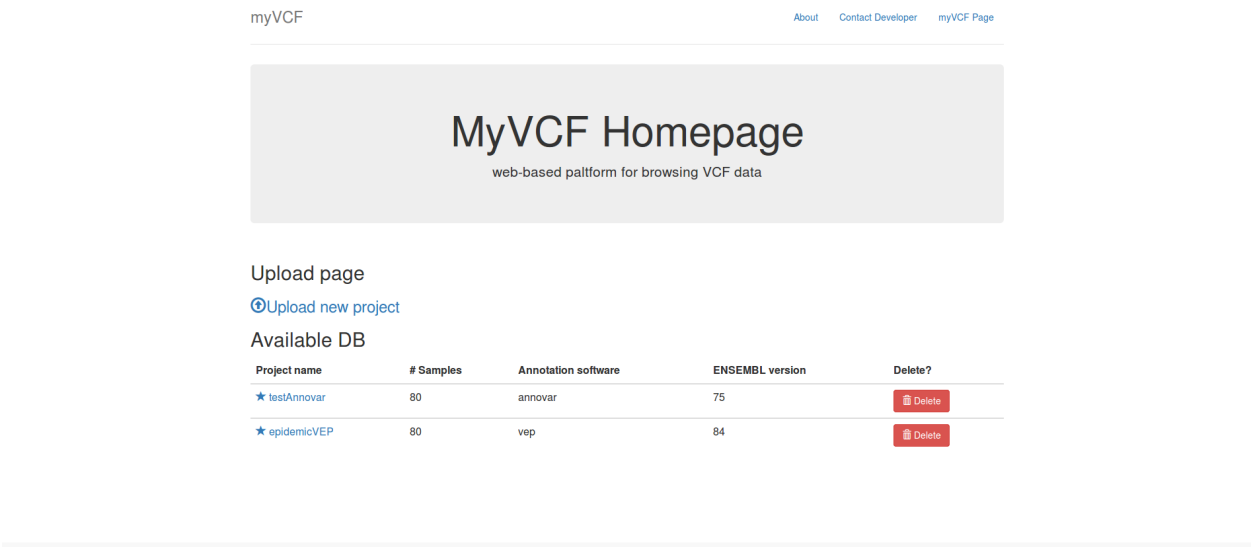
Here we described a simple gene search example

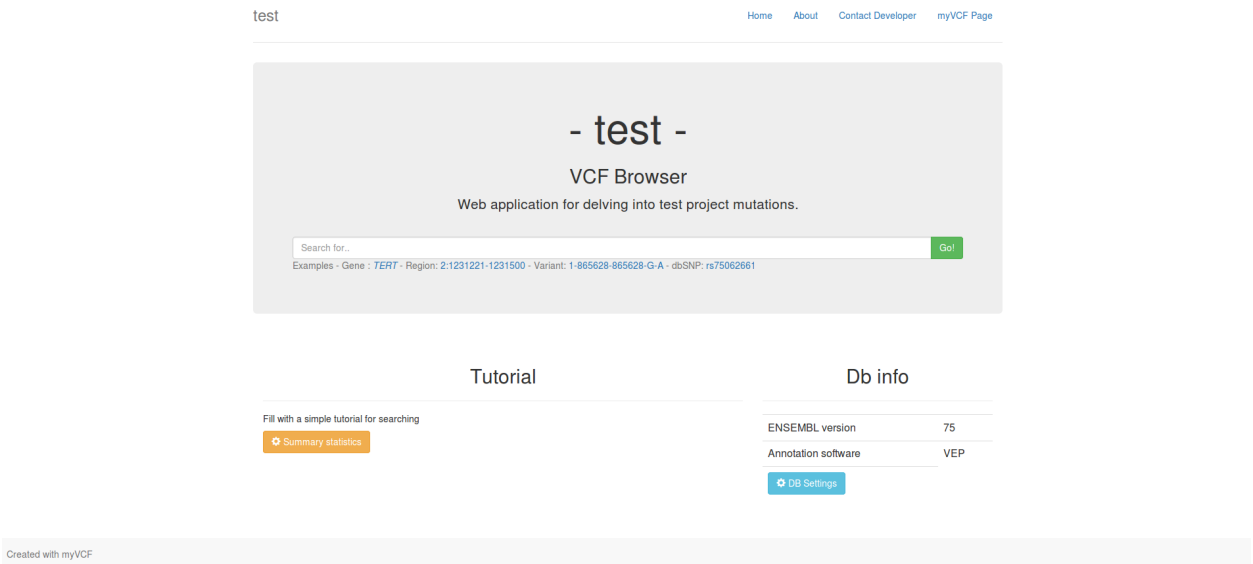
Example for SAMD11 gene search:

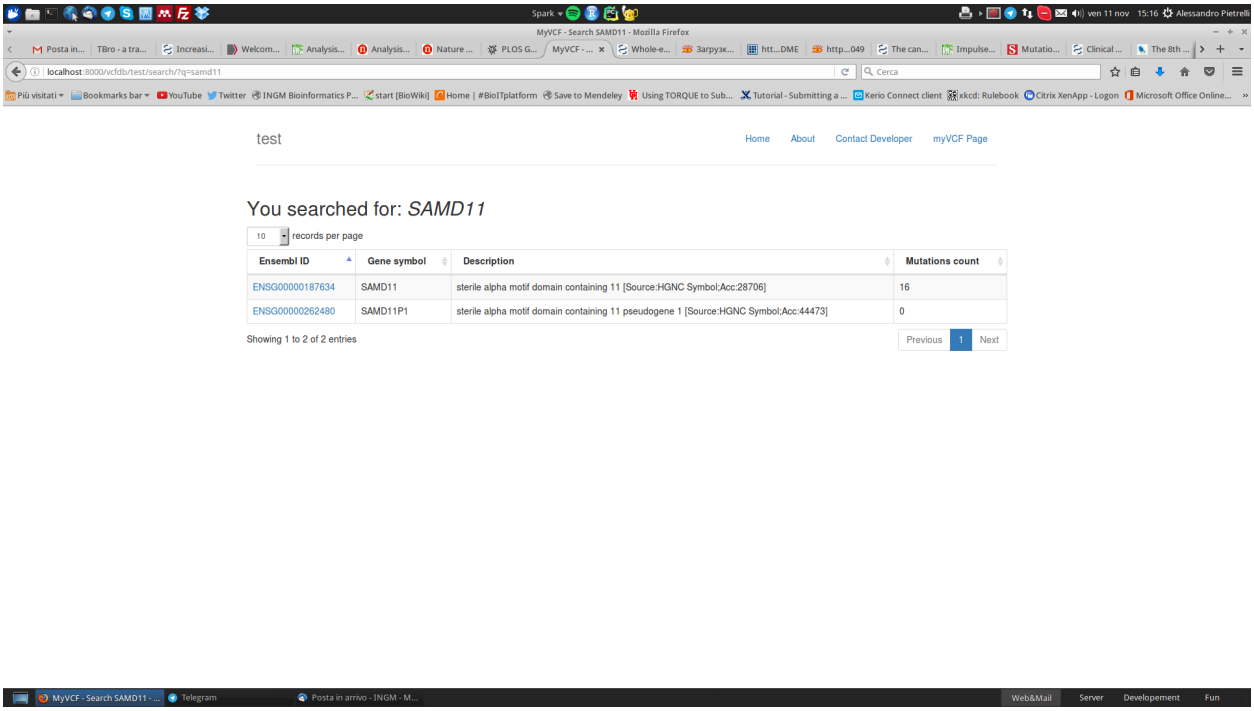
- Launch myVCF application (see how to *launch here*)
- Click on the project name you want to explore
- Fill the text box with `SAMD11` and click **GO!**

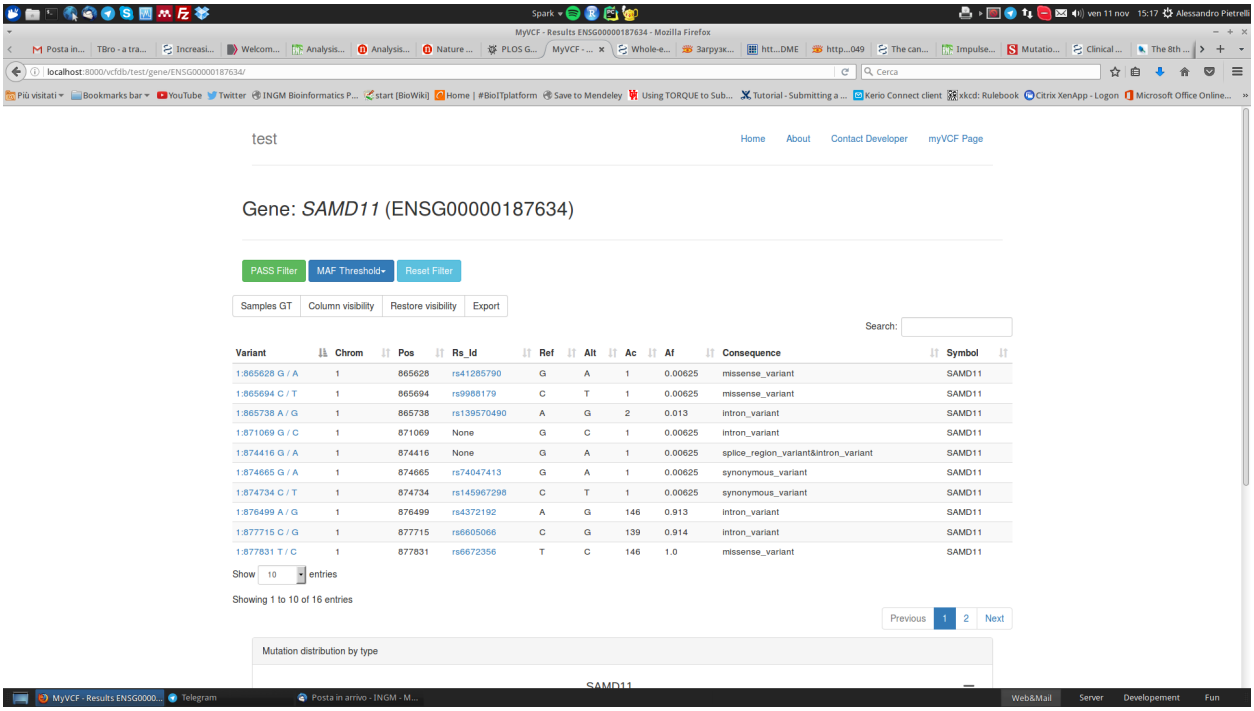
We searched for `SAMD11` gene. The system will output all genes containing the name you searched for. So in this case, togheter with `SAMD11`, the pseudogene `SAMD11P1` is also reported.

- To display the mutation list for `SAMD11 - ENSG00000187634` just click on the **ENSEMBL Gene ID** link and you will be directed to the **SAMD11 gene page**


myVCF
About
Contact Developer
myVCF Page
MyVCF Homepage
web-based paltform for browsing VCF data
Upload page
Upload new project
Available DB
Project name
Samples
Annotation software
ENSEMBL version
Delete?
testAnnovar
80
annovar
75
Delete
epidemicVEP
80
vep
84
Delete
Created by Alessandro Pietrelli


test
Home
About
Contact Developer
myVCF Page
- test -
VCF Browser
Web application for delving into test project mutations.
Search for..
Go!
Examples - Gene : TERT - Region: 2:1231221-1231500 - Variant: 1-865628-865628-G-A - dbSNP: rs75062661
Tutorial
Fill with a simple tutorial for searching
Summary statistics
Db info
ENSEMBL version
75
Annotation software
VEP
DB Settings
Created with myVCF


test
Home
About
Contact Developer
myVCF Page
You searched for: SAMD11
10 records per page
Ensembl ID
Gene symbol
Description
Mutations count
ENSG00000187634
SAMD11
sterile alpha motif domain containing 11 [Source:HGNC Symbol;Acc:28706]
16
ENSG00000262480
SAMD11P1
sterile alpha motif domain containing 11 pseudogene 1 [Source:HGNC Symbol;Acc:44473]
0
Showing 1 to 2 of 2 entries
Previous
1
Next


test
Home
About
Contact Developer
myVCF Page
Gene: SAMD11 (ENSG00000187634)
PASS Filter
MAF Threshold
Reset Filter
Samples GT
Column visibility
Restore visibility
Export
Search:
Variant
Chrom
Pos
Rs_Id
Ref
Alt
Ac
Af
Consequence
Symbol
1:865628 G / A 1 865628 rs41285790 G A 1 0.00625 missense_variant SAMD11
1:865694 C / T 1 865694 rs9988179 C T 1 0.00625 missense_variant SAMD11
1:865738 A / G 1 865738 rs139570490 A G 2 0.013 intron_variant SAMD11
1:871069 G / C 1 871069 None G C 1 0.00625 intron_variant SAMD11
1:874416 G / A 1 874416 None G A 1 0.00625 splice_region_variant&intron_variant SAMD11
1:874665 G / A 1 874665 rs74047413 G A 1 0.00625 synonymous_variant SAMD11
1:874734 C / T 1 874734 rs145967298 C T 1 0.00625 synonymous_variant SAMD11
1:876499 A / G 1 876499 rs4372192 A G 146 0.913 intron_variant SAMD11
1:877715 C / G 1 877715 rs6605066 C G 139 0.914 intron_variant SAMD11
1:877831 T / C 1 877831 rs6672356 T C 146 1.0 missense_variant SAMD11
Show 10 entries
Showing 1 to 10 of 16 entries
Previous
1
2
Next
Mutation distribution by type
SAMD11

You can filter the mutations by using the *Filter buttons*

- **PASS Filter** - Only PASS mutations will be showed. This filter acts on the `FILTER` field in the VCF file
- **MAF Threshold** - Only mutations with an Allele Frequency (AF) lower than the MAF threshold you have selected will be reported. This filter acts on the `AF` field in the VCF file.
- **Reset Filters** - Reset all filters. All mutations will be displayed.

You can also modify the visualization aspect by using the following *Display buttons*

- **Samples GT** - All the genotypes of the semples (stored in the VCF file) will be showed in the table
- **Column visibility** - Toggle On/Off the columns by selecting them from a dropdown menu
- **Restore visibility** - Restore the default column visualization
- **Export** - Save the table in different formats including `XLS`, `PDF` and `CSV`

> **Hint:** The function to export the table will recapitulate the browser visualization. If the Sample genotype columns are showed in the table, they will be exported in the file.

Note: This visualization (Gene view) and all the entire features described in this paragraph are available to search for **Gene** (as in the example), **Region** and **dbSNP ID**

Variant view

Variant view directly connectes the single variant with the additional information contained in the VCF file uploaded and stored in myVCF database.

The variant page links additional information about the allele frequency of the searched variation by interrogating all the principal population frequency database:

- **ExAC**
- **ESP**
- **1000Genomes**

Data from those database will be automatically displayed in the page.

Example for variant search:

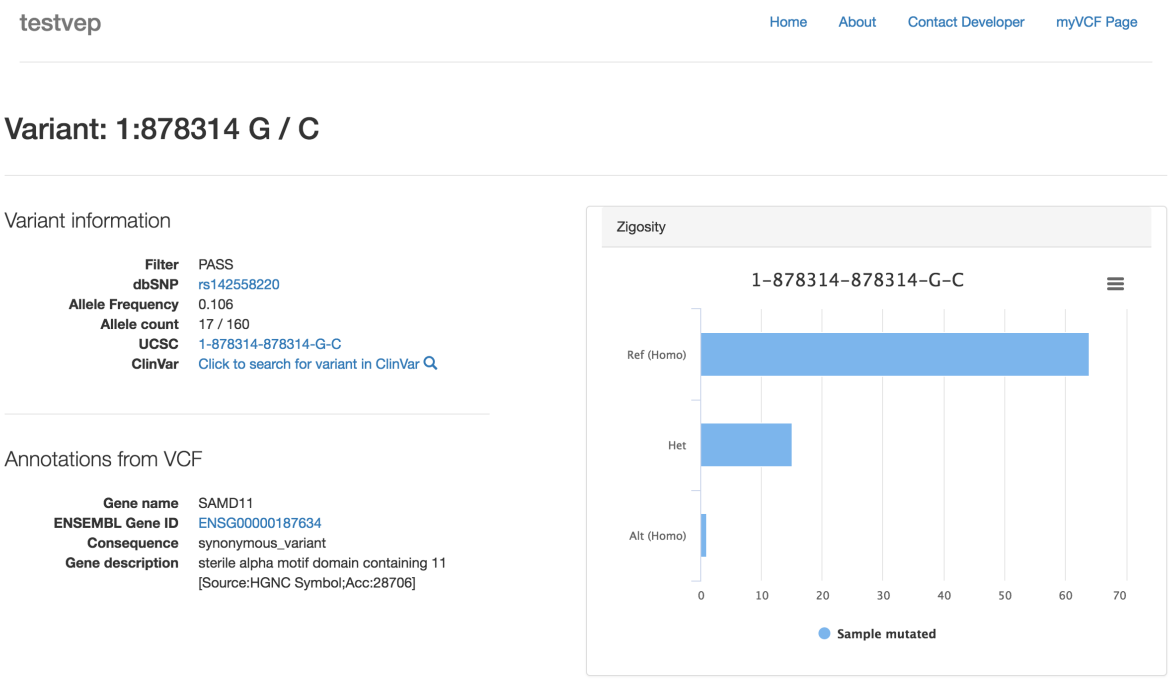
- You can search directely for single variant by using the format:

  CHR-Position-Position-Ref-Alt

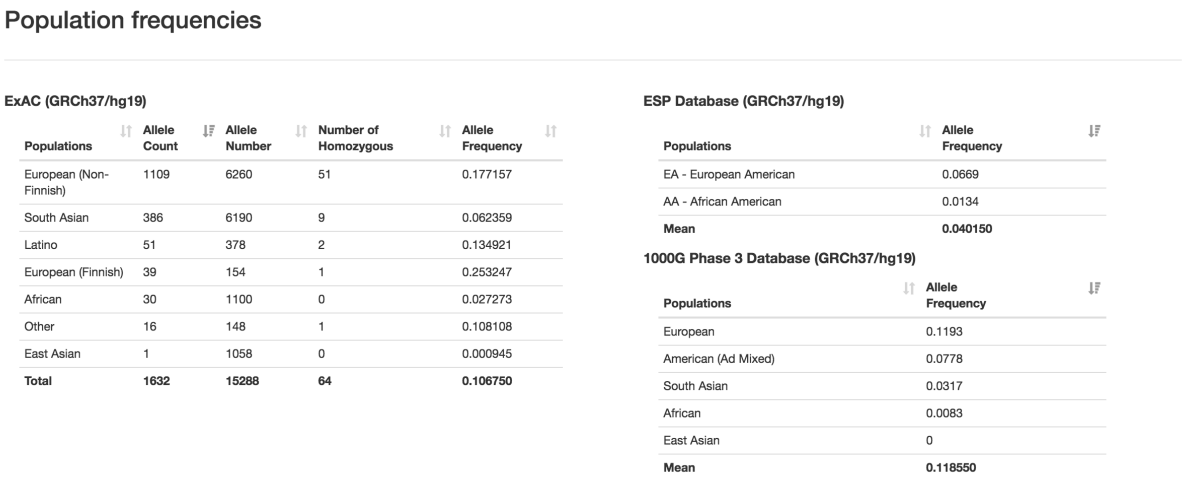
  from the project home page.

In this example we are going to search for the **1-878314-878314-G-C** variant.

- If the variant exists in the VCF file, the **variant page** will retireve information from the VCF regarding:
  - **Variant quality**
  - **Variant annotation**
  - **Zigosity distribution** across samples



- In the bottom part of the **variant page**, you will find the variant frequency distribution according to major public databases.



Population frequencies

ExAC (GRCh37/hg19)

Populations	Allele Count	Allele Number	Number of Homozygous	Allele Frequency
European (Non-Finnish)	1109	6260	51	0.177157
South Asian	386	6190	9	0.062359
Latino	51	378	2	0.134921
European (Finnish)	39	154	1	0.253247
African	30	1100	0	0.027273
Other	16	148	1	0.108108
East Asian	1	1058	0	0.000945
Total	**1632**	**15288**	**64**	**0.106750**

ESP Database (GRCh37/hg19)

Populations	Allele Frequency
EA - European American	0.0669
AA - African American	0.0134
Mean	**0.040150**

1000G Phase 3 Database (GRCh37/hg19)

Populations	Allele Frequency
European	0.1193
American (Ad Mixed)	0.0778
South Asian	0.0317
African	0.0083
East Asian	0
Mean	**0.118550**

Important: Since all the linked public database are mapped on **GRCh37/hg19** human assembly, if you load and query variation from GRCh38 assembly the frequency showed won't be correct!

Hint: Every variation in the **gene table view** (described before) is a link to its variant page.

Note: Internet connection is needed to retrieve the frequency information from public databases.

VCF metrics summary

myVCF can also generate a global VCF summary report considering several metrics and information.

You can generate this report by clicking on the *Summary button* 

Hint: The first time you load the summary statistics the process will take several minutes, especially for exome/genome projects. All following loadings will be very fast thanks to the system saving in the cache that speeds-up the process. Cache memory will be removed once the application is closed.

The VCF quality report consists of several statistics and plots all-in-one page. You can export separately each plot as single images.

Here, some example of the statistics generated:

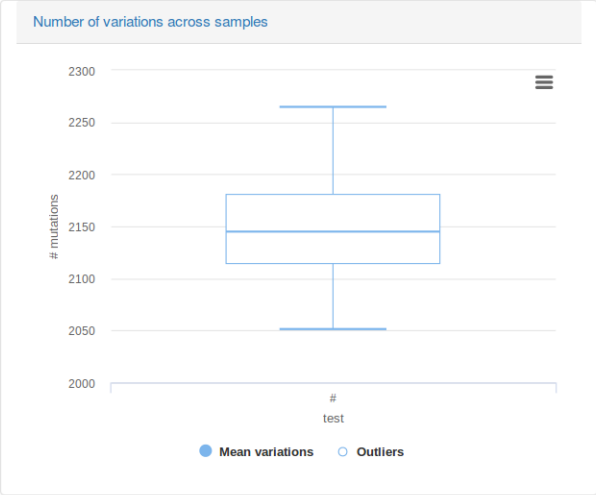
- Number of variants and the distribution of mutation across samples
- Variant quality distribution
- Variant distribution across chromosomes stratified by functional consequence
- Variant functional consequence distribution as pie chart

test

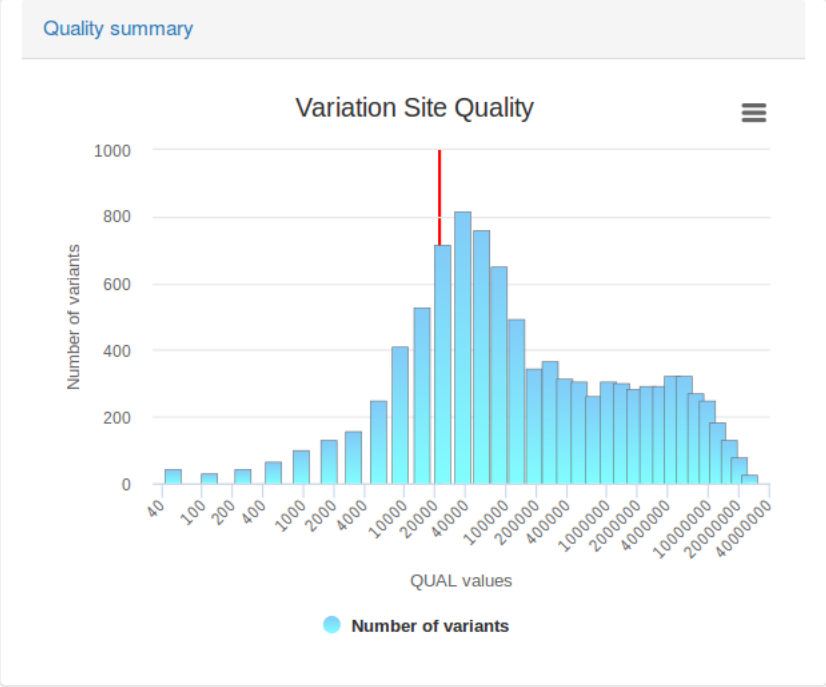
Home About Contact Developer myVCF Page

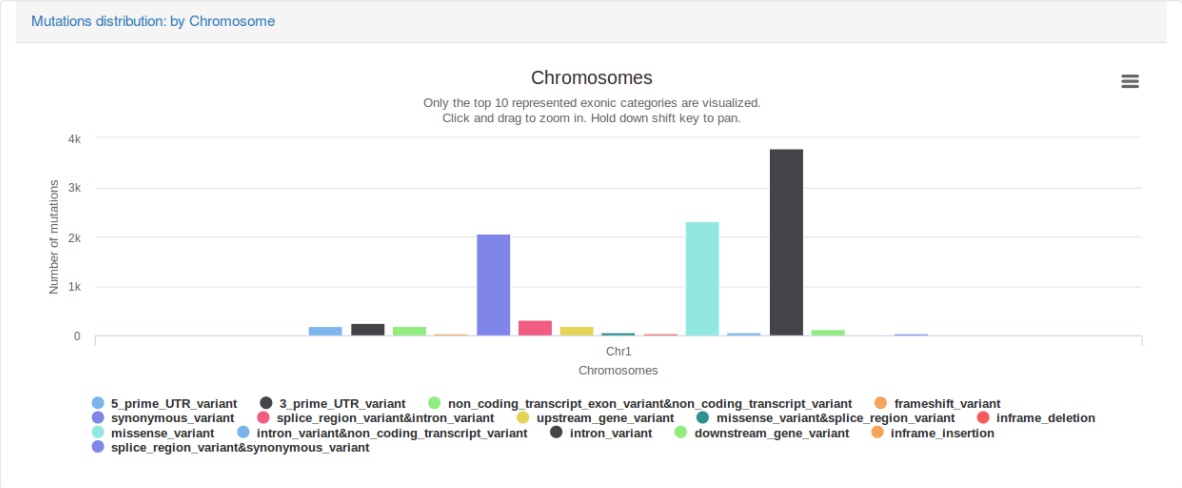
Summary statistics

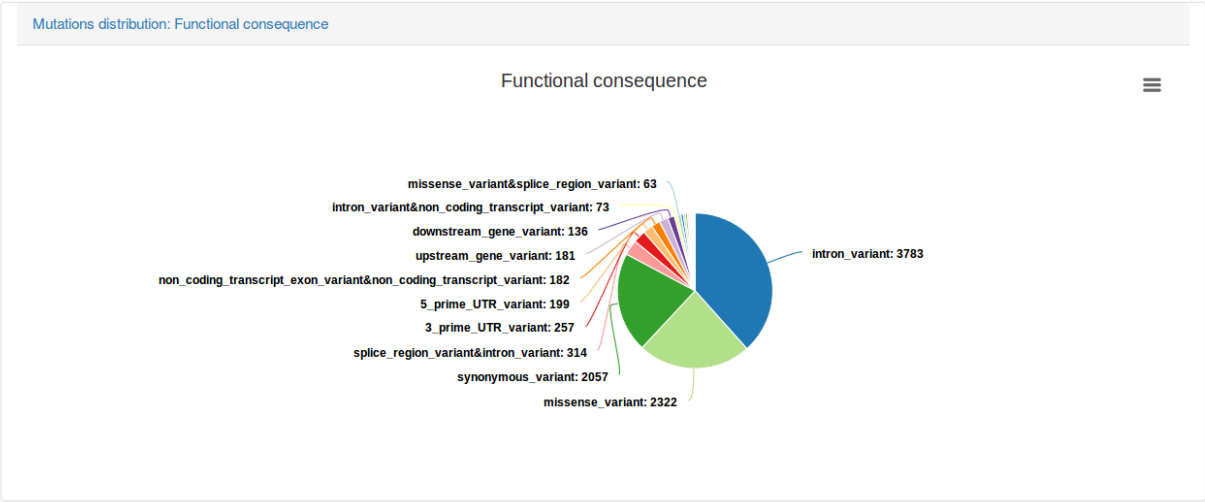
Project name	test
Software annotation	vep
Number of samples	80
Number of records	9857



Summary statistics VCF




Mutations distribution: by Chromosome
Chromosomes
Only the top 10 represented exonic categories are visualized.
Click and drag to zoom in. Hold down shift key to pan.
Number of mutations
4k
3k
2k
1k
0
Chr1
Chromosomes
5_prime_UTR_variant
3_prime_UTR_variant
non_coding_transcript_exon_variant&non_coding_transcript_variant
frameshift_variant
synonymous_variant
splice_region_variant&intron_variant
upstream_gene_variant
missense_variant&splice_region_variant
inframe_deletion
missense_variant
intron_variant&non_coding_transcript_variant
intron_variant
downstream_gene_variant
inframe_insertion
splice_region_variant&synonymous_variant


Mutations distribution: Functional consequence
Functional consequence
missense_variant&splice_region_variant: 63
intron_variant&non_coding_transcript_variant: 73
downstream_gene_variant: 136
upstream_gene_variant: 181
non_coding_transcript_exon_variant&non_coding_transcript_variant: 182
5_prime_UTR_variant: 199
3_prime_UTR_variant: 257
splice_region_variant&intron_variant: 314
synonymous_variant: 2057
missense_variant: 2322
intron_variant: 3783

Add sample groups

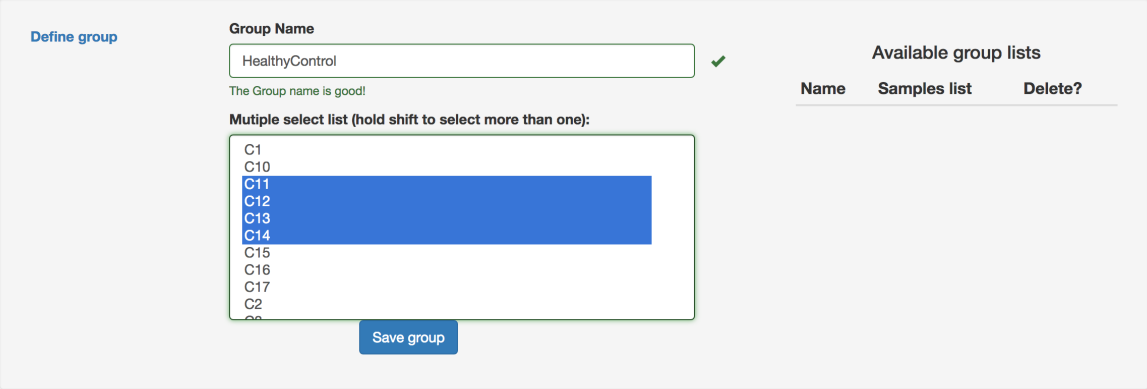
Most of the times, exome and target sequencing projects, are performed to understand the genetic difference between two or more group of samples that belong to a particular phenotype or hold some features of interest according to clinical data.

With myVCF you can easily define samples groups in order to filter and export mutations that are present only in certain samples defined by the group.

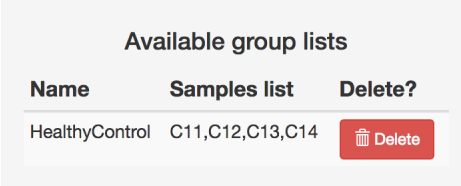
Hint: This feature is available **only for human-based and annotated projects**

To define and add groups in specific project, follow these steps:

1. Click on *DB settings page* from the project homepage
2. Go to **Setup Groups** section
3. Define a *group name* and select the *sample ID* that you want to include in the group



4. Save group by clicking on **Save group** button
5. You can verify the correct group definition by looking at the *Available group lists* table.



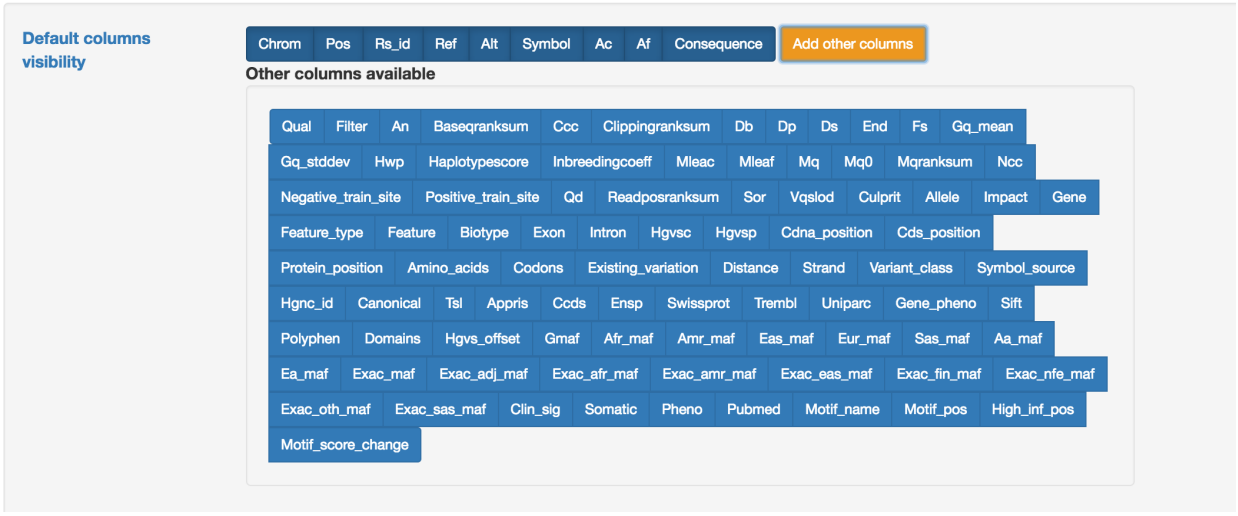
Now you can apply filters on mutations/region results by your sample group definition.

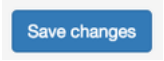
Change default columns view

By default myVCF visualizes a set of columns in the gene/region view composed by the principal annotation given by the VCF file.

You can change the default view by accessing to the *DB settings page* 

You will be redirected to the preferences page and you can select which columns will be displayed in the Gene/Region table.



To save the column view modified by the user, click on *Save changes* 

FAQ

1. What if my VCF file is not annotated or is not human-based?

Don't worry! myVCF can handle VCF from any type and automatically detect the format of your file.

You can still upload the VCF, create the project, query for available region according to your species chromosomes names and export the mutation and the genotype data.

Since the application is designed for human-based VCF, some of the available features such as, gene query or alelle frequency/in-silico predictors annotation, will be disabled.

How to cite myVCF

Paper under review!!

Update will be available soon. Fingers crossed :)