TeamCity Integration

TeamCity is a popular CI/CD tool. Now you can automatically validate your project builds using vREST test suite with TeamCity Server.

> If you are using some other Continuous Integration Server, you can integrate vREST with that in the same way as with TeamCity Server. If you face any issues then do let us know by dropping an email to support@vrest.io

Prerequisite

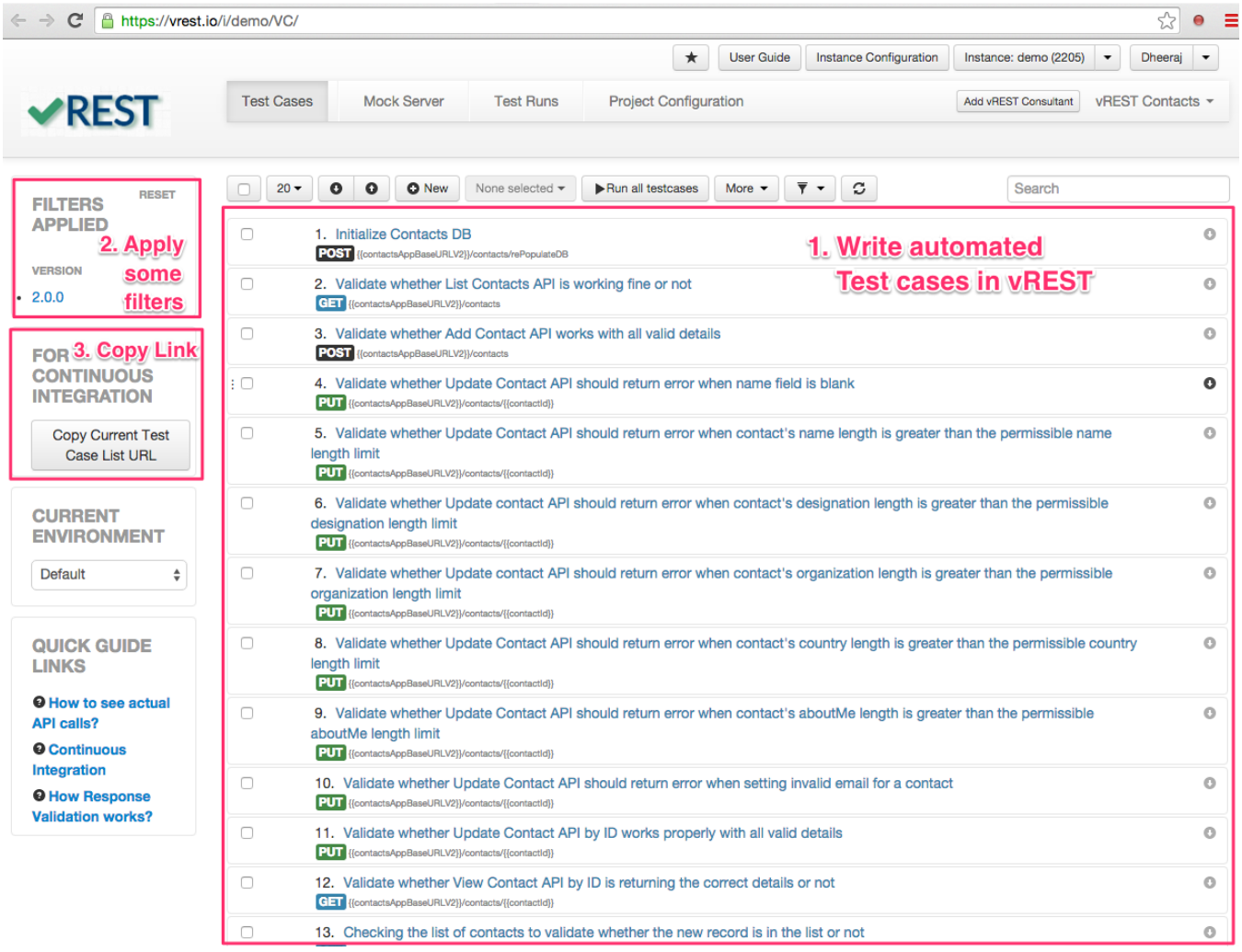
Let's take a sample application (Contacts Application) and step by step guide on how we can integrate vREST with TeamCity Server.

Note: You can find the source code of sample application at Github.

Step by Step Guide:

Step 1: Write your automated test suite in vREST

First you need to write test cases for your test application in vREST to ensure the build quality of your web application. Then copy the test case list URL by clicking on button "Copy Current Test Case List URL" available in Continuous Integration section. You can find the test cases for this sample application in vREST here.



Step 2: [Optional] Learn how vrunner command works and get your hands dirty on it by installing it locally first

First setup the vrunner module by following the guide **Setup / install vrunner**.

Once you setup this module on your local machine, you can execute the vREST test cases by executing the following command:

```
vrunner --email=<EMAIL_ID> --password=<PASSWORD> --url=<URL OF vREST Test Cases> --logger=xunit [--
filepath=<absolute log file path>] [--env=<ENVIRONMENT>]
```

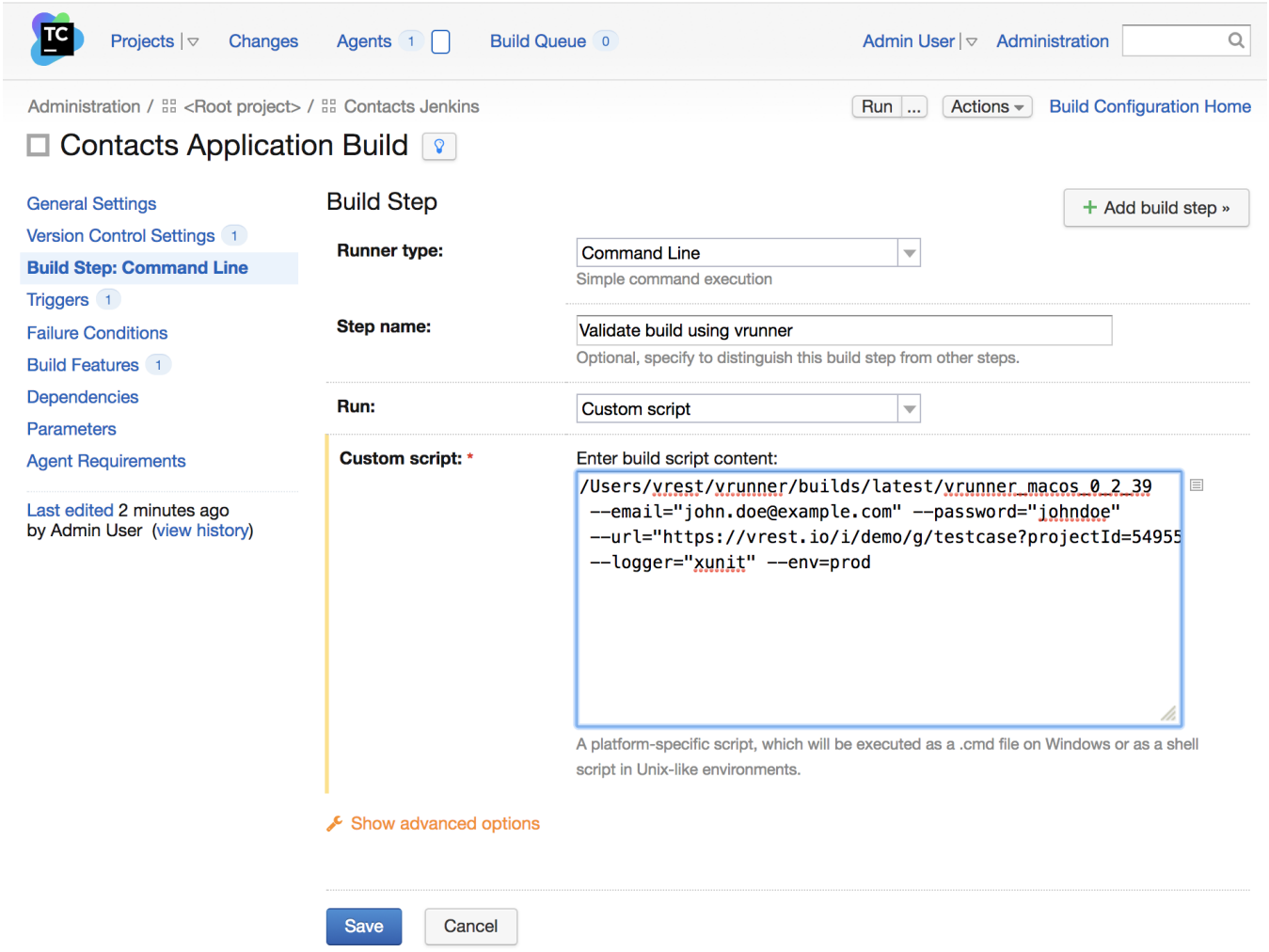
Options

- **email**: Email ID through which you have registered on vREST
- **password**: Password of your vREST account
- **url**: Provide the test case list URL here. You can find the test case list URL by going to your vREST instance and select Test Cases tab. Now click on button "Copy Current Test Case List URL" available in Left hand side, below the "Filters section". Provide the copied URL in this option.
- **logger**: Use xunit here. Reporter xunit will export the test case report in xml junit style, which can be used in any continuous integration server to publish the reports.
- **filepath:** Absolute path of the log file, into which execution process and result logs will be dumped. If path/file is not present, tool will try to setup that path, and create file automatically. Please note that if file already exists, that will be overwritten. By default it will be the `vrest_logs/logs.xml` in current directory.
- **env**: [Optional] Provide the environment name which you have defined in vREST (Case-sensitive).

Step 3: Add vrunner command in the build step in TeamCity Build Configuration

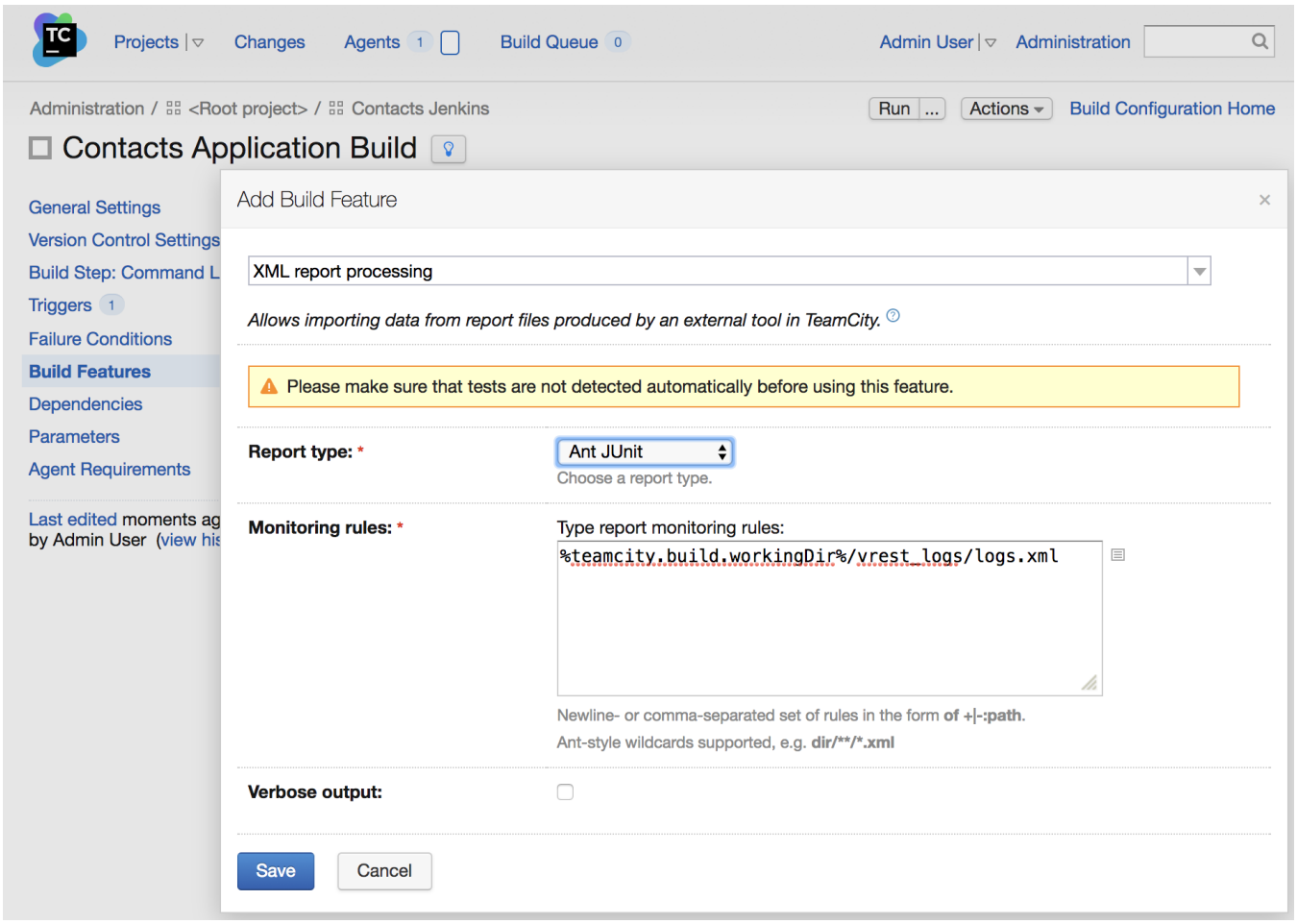
In TeamCity Build Configuration, select "**Build Step: Command Line**" and then click on button "**Add Build Step**" and just fill the form and add the custom script shown in the snapshot below in the field "**Custom Script**". Please replace the path to vrunner binary, email, password, env, url parameters of the command according to your configuration.

Note: Make Sure every % symbol in the URL must be escaped with another % symbol.



Step 4: Publish reports in TeamCity Server

By default, **vrunner** command writes the report in '**vrest_logs/logs.xml**' if logger '**xunit**' is used. You may change this path by providing the **filepath** option in the vrunner command. In TeamCity Build Configuration, select "**Build Features**" and then click on button "**Add Build Feature**" and specify the form as shown below:


Projects
Changes
Agents 1
Build Queue 0
Admin User
Administration
Administration / <Root project> / Contacts Jenkins
Run ...
Actions
Build Configuration Home
Contacts Application Build
General Settings
Version Control Settings
Build Step: Command L
Triggers 1
Failure Conditions
Build Features
Dependencies
Parameters
Agent Requirements
Last edited moments ag
by Admin User (view his
Add Build Feature
XML report processing
Allows importing data from report files produced by an external tool in TeamCity.
Please make sure that tests are not detected automatically before using this feature.
Report type: *
Ant JUnit
Choose a report type.
Monitoring rules: *
Type report monitoring rules:
%teamcity.build.workingDir%/vrest_logs/logs.xml
Newline- or comma-separated set of rules in the form of +|-:path.
Ant-style wildcards supported, e.g. dir/**/*.xml
Verbose output:
Save
Cancel

That's it.