

APPLICATION CONFIG FILE GUIDE

Table of Contents

1. Introduction	2
2. Security	3
2.1. Configure and Test Mail Server	3
2.2. Use GitLab As Authentication	5
2.3. Two-factor Authentication	6
3. Database and File Size	7
3.1. Add Database	7
3.2. Upload/Import/Export File Size	7
3.3. Password Reset	8
3.3.1. For First Login	8
3.3.2. Using Login Dialog	8
3.4. Session Timeout Duration	8
4. RML	9
4.1. RML Report Engine Left Panel	9
4.2. Fonts In RML Reports	9
5. ETL	10
5.1. ETL Logs	10
5.2. Apose Extensions for ETL Steps	10

Application Config File Guide

1. Introduction

This guide describes the application configuration file (*application.conf*) in Ambience 2021 software suite. This file is location in the “/etc” folder in Ambience.

The application configuration file is used to configure the parameters and initial settings for Ambience software.

When upgrading the Ambience software suite, it is advisable to compare the new *application.conf* file with the old one to see whether any new items have been added.

The list of parameters that can be configured are listed in the table below.

Security	
Setup email server	Page 3
Setup external authentication	Page 5
Setup Time-based One-time Password (TOTP) Two-factor Authentication (2FA)	Page 6
Password reset for first login	Page 8
Session timeout duration	Page 8
Database and File Size	
Add database	Page 7
Upload/import/export file size	Page 7
RML	
RML report engine left panel	Page 9
Fonts in RML reports	Page 9
ETL	
ETL logs	Page 10
Apose extensions for ETL steps	Page 10

Refer to the following websites for more information:

- <http://docs.elixirtech.com/Ambience/2021/index.html>
- www.elixirtech.com

2. Security

2.1. Configure and Test Mail Server

When identity is added, an email is sent with randomly generated password to the user. When a user wants to change the email or password, a verification is sent via email as well.

If you have not set up an email server, the default behaviour is to store the emails in the "/mail" folder within Ambience. This is usually for diagnosis or debugging purposes. It is recommended to set up a mail server at the start.

Below are two examples of how to set up a mail server.

Example 1: Uses Gmail

1. Gmail allows only OAuth2 authentication without weakening security. Visit <https://console.developers.google.com/apis/credentials> to set up a "clientID" and "clientSecret". Use these to generate a "refreshToken".
2. In the Ambience 2021 root folder, navigate to the "/etc" folder. Open the *application.conf* file using a text editor. In the `elixir.mail` section, edit the following with the information obtained earlier accordingly.

```
elixir.mail {  
  smtp = "gmail"  
  gmail {  
    host = "smtp.gmail.com"  
    port = 587  
    debug = true  
    oauth2 {  
      userName = "xxx@gmail.com"  
      clientId = "XXXX"  
      clientSecret = "YYYY"  
      refreshToken = "ZZZZ"  
    }  
  }  
}
```

3. Save the above edits in the *application.conf* file and start the Ambience 2021 server. New users with valid email addresses can now be created in the Identities module.

Example 2: Uses AWS

1. In the Ambience 2021 root folder, navigate to the “/etc” folder. Open the *application.conf* file using a text editor. In the “*elixir.mail*” section, edit the following:

```
elixir.mail {  
  smtp = "aws"  
  aws {  
    from = "<user@example.com>"  
    host = "<hostname>"  
    dnsResolver = ""  
    port = 465  
    user = "XXXX"  
    password = "YYYY"  
    connectionTimeout = 30000  
    tls = true  
    ssl = true  
    authMechanism = ""  
    debug = false  
  }  
}
```

2. Save the above edits in the *application.conf* file and start the Ambience 2021 server. New users with valid email addresses can now be created in the Identities module.

2.2. Use GitLab As Authentication

The Identities module in Ambience 2021 provides a simple mechanism for authentication (determining who is logging in). If you already have an authentication system, such as an SSO, LDAP or Active Directory, then it is possible to use that as the authentication mechanism. This identity management system is built upon OAuth2, which is what makes it possible to plug in alternate authentication providers.

If an external authentication system is used, the Identities module is not needed and should be removed to avoid confusion.

This section describes the steps to set up GitLab as the authentication method to log into Ambience 2021.

The steps are as follows:

1. Create an account in GitLab.
2. In GitLab, add Ambience 2021 as an application under your user.
Note that the URL callback should be <http://hostname:1740/authclient>. This is consistent with the setting in the *application.conf* file.
3. Change the hostname on your machine to point to the proper endpoint (i.e., the added application).
4. Add the user into Users module with the same name that was created in the GitLab server.
5. Go to Ambience root directory and go to the “/etc” folder. Open the *application.conf* file using a text editor.
6. Make the following changes in the `elixir.sso.client` section.

```
elixir.sso.client {
  cookie-name = "elx-amb"

  cookie-same-site = "Lax"
  openid-field = "name"
  openid-scope = "openid email"
  service-definition {
    elxssso {
      authorization = "https://<gitlab-host>/oauth/authorize"
      token = "https://<gitlab-host>/oauth/token"
      userinfo = "https://<gitlab-host>/oauth/userinfo"
      logout = "${sso-server-baseurl}"/simple-ssologout"
      debug = false
      client {
        id = "[Your Application ID]"
        secret = "[Your secret]"
        endpoint = "${sso-client-baseurl}"/authclient"
      }
    }
  }
}
```

7. Save the *application.conf* file and restart the Ambience 2021 server.
8. Now Ambience 2021 suite can be logged in using the GitLab account.

2.3. Two-factor Authentication

Ambience 2021 supports Time-based One-time Password (TOTP) Two-factor Authentication (2FA). By default, 2FA is disabled in the *application.conf* file. To enable 2FA, edit the *application.conf* file in two areas:

1. Under the *simple-server* section, change `show-totp = false` to `true`. This is to allow the login dialog to include 2FA.

```
simple-server {
  clients {
    ambience {
      secret = "171ccf22-670a-43c2-ac79-05c44bf305e3"
      redirect = "${sso-client-baseurl}"/authclient"
      #login-page = "" # set resource file here to use a custom
login page for this client
      landing-page = "http://"${host}":"${port}"/"
      name = "Elixir Ambience"
      show-totp = true
    }
  }
}
```

2. Add a new line in the *application.conf* file. This will allow User Settings module to include 2FA setup, in which users can set up their own 2FA.

```
ambience.user-settings.enable-panel.totp = true
```

3. Database and File Size

3.1. Add Database

When MongoDB is installed, by default the database is named “eno”. You can add another database into MongoDB and include it the *application.conf* file so that the Ambience 2021 is able to access it.

Two areas need to be edited.

1. The first area allows user to read and write to the database via datasets and import/export modules. In the location indicated in bold, replace it with your new database name.

```
#Where datasets can be read from and written to via datasets and
import/export modules
ambience.datasets.databases = ["eno", "NewDatabaseName"]
ambience.import.databases = ["eno", "NewDatabaseName"]
ambience.export.databases = ["eno", "NewDatabaseName"]
```

2. The other area allows the user to include the new database so that Ambience can access it in MongoDB. In the `elixir.data.mongodb` section, add a new line as indicated in bold, replacing the database name.

```
elixir.data.mongodb {
  ...
  default {
    connectionString = "mongodb://"${mongodb}":27017"
    ...
    database {
      eno = "eno"
      NewDatabaseName = "NewDatabaseName"
    }
  }
}
```

3.2. Upload/Import/Export File Size

The current file size limit is set to 150 MB in the *application.conf* file. This is to prevent user from slowing down the server by uploading or exporting huge files.

You can increase or decrease the size file limit by editing the value in bold.

```
# Used by UploadDownload and ImportExport Modules
akka.http.server.parsing.max-to-strict-bytes = 150m

# Used by UploadDownload Module and ImportExport Modules
akka.http.server.parsing.max-content-length = 150m
```


3.3. Password Reset

3.3.1. For First Login

When the user logs in with the randomly generated password (i.e., first login), they will be forced to change the password immediately. This can be disabled by editing the setting in the *application.conf* file. In the `elixir.Identity` section, edit the `changePassword = true` to `false`.

```
elixir.Identity {  
  ...  
  on-reset {  
    changePassword = true  
  }  
}
```

3.3.2. Using Login Dialog

Any user can reset their own password using the login dialog. The new password will be sent to the user's email.

3.4. Session Timeout Duration

Ambience software has inactivity session timeout of 15 minutes. This timeout duration can be changed in the *application.conf* file. In the `ambience.web` section, edit the `session-timeout = 15 minutes` to the desired duration.

```
ambience.web {  
  ...  
  session-timeout = 15 minutes  
  enforce-single-session = false  
  ...  
}
```

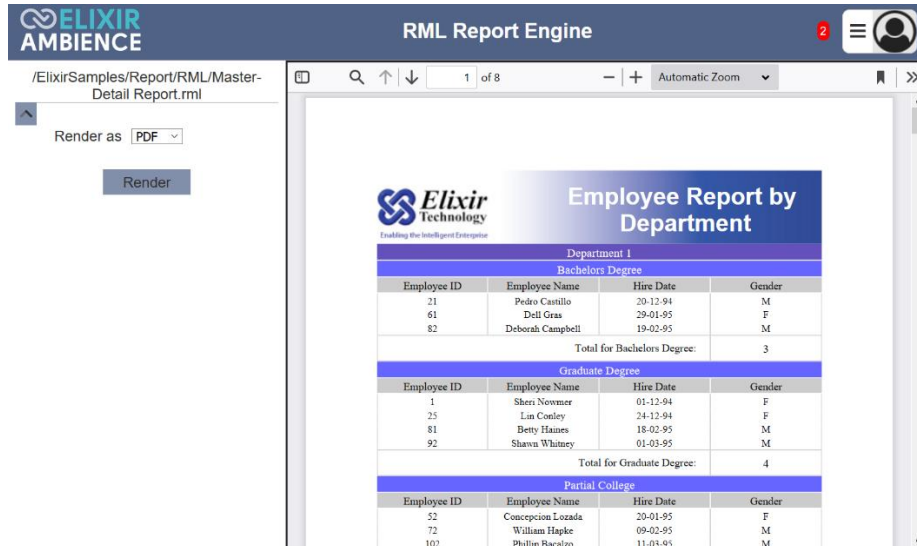
To disable the timeout, use the value `never`.


4. RML

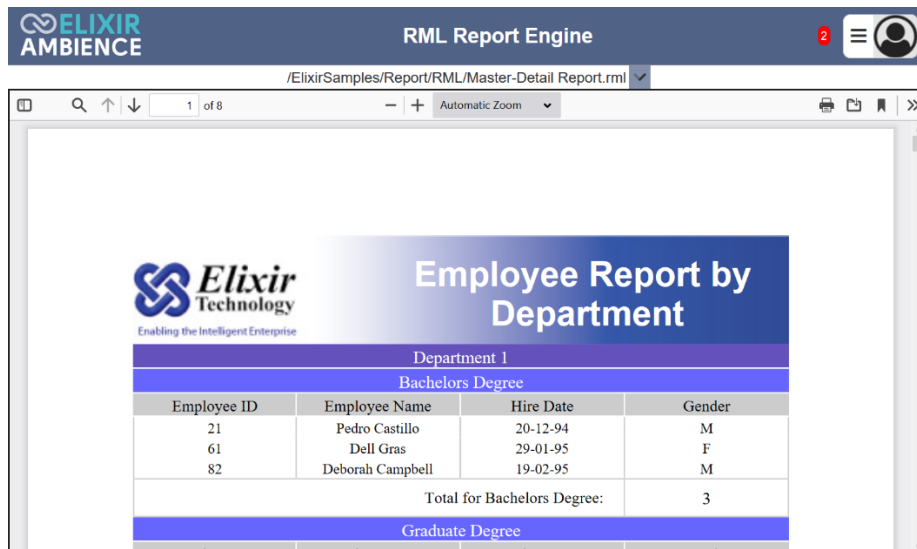
4.1. RML Report Engine Left Panel


When generating a RML report, the report by default fills the entire area in the RML Report Engine page. You can enable the parameters on the left of the page. To do so, add the following line in the *application.conf* file.

```
ambience.rml-engine.ui-left-parameters = ["application/pdf"]
```



You can change the pdf to other types if your browser supports viewing them. You can toggle the left panel off by clicking on the  icon.



To toggle on, click on the  icon on the top row.

4.2. Fonts In RML Reports

You can use extra fonts (non-OS) in the RML reports. To do so, add an extra line in the *application.conf* file.

```
elixir.rml.fonts.path = "<pathname>/Fonts"
```

5. ETL

5.1. ETL Logs

By default, the ETL job logs is created as the job runs. These logs allow you to observe the behaviour of the jobs. But these logs may not be useful for production systems as they may run the ETL jobs 24 hours a day, thus generating huge volumes of log. This may slow down the system.

You can turn off the job log by adding the following into the *application.conf* file.

```
ambience.etl.logging.write-to-server-log: false
ambience.etl.logging.write-to-job-log: false
```

5.2. Apose Extensions for ETL Steps

To enable the Apose extensions for ETL steps, you require a separate license. Add the following into the *application.conf* file.

```
ambience.docx-engine.aspose {
  enabled = true
  words-license = "<pathname>/Aspose.Words.Java.lic"
}
```