



Plunner
Test Report Document
Version 1.1

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Revision History

Date	Version	Description	Author
07/01/2016	1.0	Initial Version	Denis Kuryshov, Giorgio Pea
20/01/2016	1.1	Final checking and revision	Giorgio Pea

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Table of Contents

[Revision History](#)

[Table of Contents](#)

[1. Introduction](#)

[1.1 Purpose of this document](#)

[1.2 Intended audience](#)

[1.3 Scope](#)

[1.4 Definitions and acronyms](#)

[1.5 Support documentation](#)

[2. End to end manual tests](#)

[2.1 General tests](#)

[2.3 Member](#)

[2.3 Planner](#)

[3. Unit tests](#)

[3.1 Backend](#)

[3.1.1 Member/Planner](#)

[3.2 Frontend](#)

[4. Other tests](#)

[4.1 Responsiveness tests](#)

[4.2 Browser support](#)

Plunner	Version: 1.1
Test Report Document	Date: 20/01/2016

1. Introduction

1.1 Purpose of this document

The purpose of this document is to report the set of tests that team behind the project has made in order to guarantee the achievement of all the requirements of the project itself and its quality. These tests have been implemented following the ATP document.

1.2 Intended audience

The intended audience for this document is:

- **The team behind the project:** since this document describes the different tests the team has performed to guarantee the correctness and the quality of the product, in case of future improvements and development it will be used as starting point for new tests and quality checkings
- **The supervisors of the project:** the information contained in this document can be used by the supervisors to understand the testing process of Plunner, and so to make suggestions to the team and have an idea of the final quality of the project
-

1.3 Scope

This document aims to show the different types of tests the team has performed against the application, tests that follows the plan presented in the ATP document. The focus is on end to end manual test, in order to verify the correctness and quality of the system as a whole. nevertheless other kind of tests are described and documented.

1.4 Definitions and acronyms

Abbreviation, acronym, term	Definition
ATP	Acceptance Test Plan

In addition to the indicated abbreviations and definition see the RDD's definitions and acronyms

1.5 Support documentation

- Acceptance Test Plan Document

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

2. End to end manual tests

Here follows a detailed list of the most important end to end manual tests accordingly to the ATP document.

2.1 General tests

Test	T1.1
Test Name	Test empty fields or empty required fields of a form
Test Description	Test if trying to submit a form without filling any of its fields or at least one of its required fields produces errors that are signaled to the user
Prerequisite(s)	The application is installed and is running
Input Specification	The user clicks the button of the current view that submits the form in that view (usually the button with a blue background) without filling any of the form's fields or without filling at least one of the form's required fields
Output Specification	An HTTP 422 "Unprocessable Entity" response is submitted by the backend to the browser, the body of this response contains the indication of the required data to be sent. The fields that hold the non inserted data are outlined in red and below them a red error message appears prompting: "Insert a [data to be inserted in that field]"
Pass/Fail	PASS FOR EVERY VIEW

Test	T1.2
Test Name	Test invalid data inserted in a form view
Test Description	Test if trying to submit a form filled with invalid data produces errors that are signaled to the user
Prerequisite(s)	The application is installed and is running

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Input Specification	The user clicks the button of the current view that submits the form in that view (usually the button with a blue background) having filled the form with invalid data
Output Specification	An HTTP 422 “Unprocessable Entity” response is submitted by the backend to the browser, the body of this response contains the indication of the invalid data sent. The fields that hold the invalid data are outlined in red and below them a red error message appears prompting: “Insert a valid [data to be inserted in that field] ”
Pass/Fail	PASS FOR EVERY VIEW

2.2 Organization/Registered organization

Test	T2.1
Test Name	Test Login with non matching credentials
Test Description	Test if trying to login by entering non matching credentials (credentials that do not correspond to any registered organization) produces errors that are signaled to the organization
Prerequisite(s)	The application is installed and is running
Input Specification	The organization navigates to the organization sign in section of the website, inserts valid non matching credentials and clicks the “sign in” button
Output Specification	An HTTP 422 “Unprocessable Entity” response is submitted by the backend to the browser, the body of this response contains an error message that indicates the absence of any user with the inserted credentials in the database. On the top of the “sign in” button a red error message appears prompting “These credentials do not match our records”
Pass/Fail	PASS

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Test	T2.2
Test Name	Test Login with matching credentials
Test Description	Test if trying to login by entering matching credentials (credentials that correspond to any registered organization) produces an authorizing popup and a redirect to the registered organization dashboard
Prerequisite(s)	The application is installed and is running
Input Specification	The organization navigates to the organization sign in section of the website, inserts valid matching credentials and clicks the “sign in” button
Output Specification	An HTTP 200 success response is submitted by backend to the browser, the body of this response contains the authentication token for the organization. In the top of the page a popup with a spinner inside, and with the word “Authorizing” appears and the registered organization is redirected to its dashboard
Pass/Fail	PASS

For the organization registration the tests are the same as above but the website section involved is the “organization sign up” section, there are no credentials, instead we talk about registration information, the “sign in” button is replaced by the “sign up” button and the popup mentioned in the test T.2.2 contains the word “Registering” instead of “Authorizing”.

Test	T2.3
Test Name	Test member creation
Test Description	Test if adding a member using the “add member” popup and valid non already in use credentials actually creates a new organization member that can be seen in the organization dashboard
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization and of the “add member” popup

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Input Specification	The organization fills the fields in the “add member” popup with valid non already in use credentials and clicks the add button
Output Specification	A popup with a spinner and the word “Adding” appears and then the organization can see in its dashboard the freshly created member. The backend has successfully stored the new member data in the database.
Pass/Fail	PASS

Test	T 2.4
Test Name	Test member information
Test Description	Test if all the members of an organization appear in the “members” section of the organization’s dashboard
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization
Input Specification	The organization navigates to the “members” section of its dashboard
Output Specification	Every member of the organization is listed with its name and the group it belongs to, this information has been successfully retrieved from the database
Pass/Fail	PASS

Test	T 2.5
Test Name	Test member information detail
Test Description	Test if all the information of an organization’s member is displayed in the “member information detail” section of the organization view
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered

Plunner	Version: 1.1
Test Report Document	Date: 20/01/2016

	organization
Input Specification	The organization navigates to the “members” section of its dashboard, clicks the arrow in the tile relative to one of its members and is redirected to the “members information detail” section
Output Specification	In the above mentioned section the member’s name, email address and joined groups are shown. This information has been successfully retrieved from the database.
Pass/Fail	PASS

Test	T 2.6
Test Name	Test member deletion
Test Description	Test if the deletion of a member using the “delete member” button of the “member information detail” section in edit mode actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization and of the “member information detail section” in edit mode
Input Specification	The organization clicks the “delete member” button
Output Specification	A popup with a spinner and the word “Deleting member” appears and then the organization is redirected to its dashboard; here, in the “members” section, the deleted member is no more visible. The deleted member is no more present in the database
Pass/Fail	PASS

Test	T 2.7
Test Name	Test removing a member from a team T

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Test Description	Test if removing a member from a team using the links present in the “member information detail” section in edit mode actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization in a “member information detail” section in edit mode
Input Specification	The organization clicks the “delete from team” link near the word “T” under the team's subsection
Output Specification	A popup with a spinner and the word “Removing from team” appears, then the organization is returned to the “member information detail section”, here the left team does not appear anymore. In the organization dashboard the tile that corresponds to the member under analysis does not show anymore the left team and in the “team information detail” section relative to the left team the member mentioned above is not present anymore. The member is no more associated in the database to the team T
Pass/Fail	PASS

Test	T 2.8
Test Name	Test editing a member information
Test Description	Test if editing a member information by using the “save changes” button in the “member information detail” section in edit mode actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization in a member information detail section in edit mode

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Input Specification	The organization edits the information of the member(name, email and password), using the fields that the “member information detail section” in edit mode shows, and clicks the save changes button
Output Specification	A popup with a spinner and the word “Saving changes” appears and then the organization is returned to the “member information detail” section, here the new information is visible. In the organization's dashboard the tile that matched the member in analysis has been updated with the new data. The information stored in the database relative to the member in analysis has been successfully updated with the new inserted values
Pass/Fail	PASS

Test	T2.9
Test Name	Test team creation
Test Description	Test if adding a team using the “add team” popup actually creates a new organization team that can be seen in the section “teams” of the organization’s dashboard
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization and of the “add team” popup
Input Specification	The organization fills the fields in the “add team” popup with a non already in use name and with a description, selects at least one its members as components of the team and at least one of its members as planner of the team and clicks the “add” button
Output Specification	A popup with a spinner and the word “Adding group” appears and then the organization is redirected to its dashboard, here a tile that matches the freshly created group with its name, description and planner is now present. This information is also visible in the “team information detail” section along with all the

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

	members of the team. The new team has been successfully saved in the database
Pass/Fail	PASS

Test	T3.0
Test Name	Test team deletion
Test Description	Test if the deletion of a team using the “delete team” button in the “team information detail” section in edit mode actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization and in the “team information detail” section in edit mode
Input Specification	The organization clicks the “delete team” button
Output Specification	A popup with a spinner and the word “Deleting team” appears and then the organization is redirected to its dashboard, here, in the “teams” section, the deleted group is no more visible. The team has been successfully deleted from the database
Pass/Fail	PASS

Test	T3.1
Test Name	Test removing a member from a team
Test Description	Test if removing a member from a team using the trash icons near the group members listed in the “team information detail” section in edit mode, actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

	organization and of the “team information detail” section in edit mode
Input Specification	The organization clicks the trash icon near a listed group member
Output Specification	A popup with a spinner and the word “Removing from team” appears and then the organization is returned to the “team information detail” section, here the removed group member is no more visible. In the organization’s dashboard, in the members section, the tile matching the removed member does not show anymore the team from which the member has been removed, the same is for the “member information detail” section. The member is no more associated in the database with the team
Pass/Fail	PASS

Test	T3.2
Test Name	Test editing team information
Test Description	Test if editing team information by using the “save changes” button in the “team information detail” section in edit mode, actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization and of the “team information detail” section in edit mode
Input Specification	The organization edits the information of the group, using the fields that the “team information detail” section in edit mode shows, and clicks the “save changes” button
Output Specification	A popup with a spinner and the word “Save changes” appears and then the organization is returned to the “team information detail” section, here the information visible has changed accordingly to edits of the organization; the same thing is valid for the tile that matches the group in the organization’s

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

	dashboard under the “teams” section. In the database, the team’s information has been successfully updated with the new inserted values
Pass/Fail	PASS

Test	T3.3
Test Name	Test planner change
Test Description	Test if changing a planner of a team using the “change planner” popup, actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization and of the “change planner” popup accessible from the “team information detail” section
Input Specification	The organization selects a new member as a planner using the list of members provided and clicks the “change” button
Output Specification	A popup with a spinner and the word “Saving changes” appears and then the organization is returned to the “team information detail” section. In this section the newly selected planner is visible, the same thing is valid for the tile that matches the team in the organization’s dashboard under the “teams” section, In the database the team has been successfully associated with the new inserted Planner
Pass/Fail	PASS

Test	T 3.6
Test Name	Test adding Members to a team
Test Description	Test if adding Members to a team via the “add to team” popup actually performs the action

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Prerequisite(s)	The application is installed and is running, we are in the context of a logged registered organization and of the “add to team” popup accessible from the “team information detail” section
Input Specification	The organization selects the members it wants to add to the group using the list of members provided and clicks the “add” button
Output Specification	A popup with a spinner and the word “Adding members” appears and then the organization is returned to the “team information detail” section. In this section the new members are visible. In the organization’s dashboard, under the “members” sections, the tiles that match the added members show the team name; this is also valid for the “member information detail” section. In the database the team has been successfully associated with the new Members
Pass/Fail	PASS

2.3 Member

Test	T3.7
Test Name	Test importing a schedule using CalDAV with correct existing credentials
Test Description	Test if importing a schedule from an external service based on CalDAV, providing correct existing CalDAV credentials using the “import schedule” popup actually performs that action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged member and of the “import schedule popup”
Input Specification	The members inserts correct CalDAV credentials (username, password, CalDAV url) clicks the “get calendars” button and select at least one of the listed CalDAV calendars
Output Specification	A popup with a spinner and the word “Importing schedule” appears and then the

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

	member is returned to its dashboard, here the tile that matches the imported schedule is now visible with its information(name, state, sync-datetime), this information is also displayed in the “edit imported schedule” popup. The imported schedule’s data has been successfully inserted in the database, the imported schedules’ synchronization cron job has been successfully updated to include the new imported schedule
Pass/Fail	PASS

Test	T3.8
Test Name	Test importing a schedule using CalDAV with non existing or incorrect credentials
Test Description	Test if importing a schedule from an external service based on CalDAV, providing non existing or incorrect CalDAV credentials using the “import schedule” popup produces errors that are signaled to the member
Prerequisite(s)	The application is installed and is running, we are in the context of a logged member and of the “import schedule popup”
Input Specification	The members inserts non existing or incorrect CalDAV credentials (username, password, CalDAV url) clicks the “get calendars” button
Output Specification	An HTTP 422 “Unprocessable entity” response error is submitted from the backend to the browser, the body of the response contains an error message about the incorrect credentials. No calendar is listed in the popup and an error message in red appears signaling to the member the error
Pass/Fail	PASS

Test	T3.9
Test Name	Test editing imported schedule with correct or existing CalDAV credentials

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Test Description	Test if editing an imported schedule using the “edit imported schedule” popup actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged member and of the “edit imported schedule” popup
Input Specification	The members changes the name of the schedule, the CalDAV name of the schedule, the CalDAV credentials and the state of the schedule(enabled/disabled)
Output Specification	A popup with a spinner and the word “Saving changes” appears and then the member is returned to its dashboard, here the tile that matches the imported schedule shows the updated information, this information is also displayed in the “edit imported schedule” popup. In the database the imported schedule has been successfully updated with newly inserted information
Pass/Fail	PASS

Test	T4.0
Test Name	Test editing imported schedule with incorrect or non existing CalDAV credentials
Test Description	Test if editing an imported schedule using the “edit imported schedule” popup with incorrect or non existing CalDAV credentials actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged member and of the “edit imported schedule” popup
Input Specification	The members changes the name of the schedule, the CalDAV name of the schedule, the CalDAV credentials and the state of the schedule(enabled/disabled). The new CalDAV credentials inserted are incorrect or non existing
Output Specification	An HTTP 422 “Unprocessable entity” response

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

	error is submitted from the backend to the browser, the body of the response contains an error message about the incorrect credentials. A popup with a spinner and the word “Saving changes” appears and then the member is returned to “edit imported schedule” popup, here an error message is shown to the member
Pass/Fail	PASS

Test	T 4.1
Test Name	Test imported schedule deletion
Test Description	Test if deleting an imported schedule using the “delete” button in the “edit imported schedule” section actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged member and of the “edit imported schedule” popup
Input Specification	The member clicks the “delete schedule button”
Output Specification	A popup with a spinner and the word “Deleting schedule” appears and then the member is returned to its dashboard, here no tile matches the deleted schedule. In the database no more data exists relative to the schedule
Pass/Fail	PASS

Test	T 4.2
Test Name	Test schedule composition
Test Description	Test if composing a schedule by inserting a name for the schedule and a series of busy timeslots and pressing the “save schedule” button in the context of the “compose schedule” section actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged member in the “compose schedule section”

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

Input Specification	The member inserts a name for the schedule, selects the status of the schedule(enabled/disabled), drags at least one busy timeslot for the schedule in the calendar view in the section, and clicks the “save schedule” button
Output Specification	A popup with a spinner and the word “Creating schedule sch” appears and then the member is returned to its dashboard, here, in the “schedules” section the a tile relative to the created schedule is visible; clicking on that tile redirects the Member to “composed schedule detail” section in which the information about the schedule is visible and editable. The created schedule has been inserted successfully in the database
Pass/Fail	PASS

Test	T 4.2
Test Name	Test meetings view and meeting detail view
Test Description	Test if all the meetings to be planned and planned for every team a Member belongs to, are visible in the “meetings” section of the Member’s dashboard
Prerequisite(s)	The application is installed and is running, we are in the context of a logged Member in the “meetings” section of its dashboard
Input Specification	The Member navigates to the “meetings” section of its dashboard
Output Specification	All the to be planned and planned meetings for all the teams a Member belongs to are visible, with their information, as tiles in the “meetings” section of the Member’s dashboard, clicking on one of them will open a popup with a detail view of the clicked meeting’s information
Pass/Fail	PASS

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

2.3 Planner

Test	T 4.3
Test Name	Test meeting creation
Test Description	Test if creating a meeting by specifying a title, a description, a duration and a series of possible meeting times and by clicking the “create meeting” button in the “create meeting” section actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged planner in the “create meeting “ section
Input Specification	The Planner inserts a title, a description, a duration and a series of possible meeting times for a meeting. The Planner then clicks the “create meeting” button
Output Specification	A popup with a spinner and the word “Creating meeting” appears and then the member is returned to its dashboard, here, in the “meetings” section the a tile relative to the created meeting is visible; clicking on that tile redirects the Member to “edit meeting” section in which the information about the meeting is visible and editable. The created meeting has been inserted successfully in the database
Pass/Fail	PASS

Test	T 4.3
Test Name	Test non planned meeting editing
Test Description	Test if editing the information relative to a non planned created meeting using the “edit meeting” section of the Planner’s view actually performs the action
Prerequisite(s)	The application is installed and is running, we are in the context of a logged planner in the

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

	“edit meeting“ section
Input Specification	The Planner edits the title, the description, the duration and the possible meeting times and clicks the “save changes” button
Output Specification	A popup with a spinner and the word “Saving changes” appears and then the Planner is returned to its dashboard, here, in the “meetings” section the tile relative to the updated meeting is visible with the new inserted information; clicking on that tile redirects the Member to “edit meeting” section in which the information about the meeting has been updated. In the database the information about the edited meeting has been accordingly updated
Pass/Fail	PASS

Test	T 4.3
Test Name	Test planned meeting editing
Test Description	Test if the editable information about the meeting is only restricted to title and description
Prerequisite(s)	The application is installed and is running, we are in the context of a logged planner in the “edit meeting“ section
Input Specification	The Planner tries to edit the duration of the meeting, the possible meeting times, the title and description of the meeting and clicks the “save changes” button
Output Specification	The web interface of the “edit meeting” section of the Planner’s view denies the editing of the duration of the meeting and there’s no calendar like view for changing the possible meeting times, only the title and description of the meeting are editable and the consequent behavior of the application is the same as the one described in the previous test. In the case that someone forces the editing using HTTP requests an HTTP 422 “unprocessable entity” response is submitted by the backend to the

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

	browser specifying an error message
Pass/Fail	PASS

3 Unit tests

3.1 Backend

Here follows a list of backend unit tests divided by actor, the terminology used is different from the one present in this document: Company → Organization, Employee → Member, Group → team. The behavior tested by the below unit tests is clear from their title, so only for a small part of them a more detailed description is given.

3.1.1 Member/Planner

1. Auth
 - 1.1. Login
 - 1.2. ErrorLogin
 - 1.3. ResetPassword
 - 1.4. ErrorResetPassword
 - 1.5. ThrottlesLogins
 - 1.6. NoThrottlesLogins
2. Calendars
 - 2.1. Calendars
 - 2.1.1. Create
 - 2.1.2. Delete
 - 2.1.3. UpdateNoCalDAV
 - 2.1.4. Show
 - 2.1.5. ShowNotMine
 - 2.1.6. CreateCalDAV
 - 2.1.7. UpdateCalDAV
 - 2.1.8. GetExternalCalendars
 - 2.1.9. Index
 - 2.1.10. ErrorIndex
 - 2.2. Timeslots
 - 2.2.1. Create
 - 2.2.2. Delete
 - 2.2.3. Update
 - 2.2.4. Show
 - 2.2.5. ShowNotMine
 - 2.2.6. AllIndex
 - 2.2.7. CurrentIndex
 - 2.2.8. ErrorIndex

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

- 3. Employee
 - 3.1. Index
 - 3.2. ErrorIndex
 - 3.3. Update
- 4. Groups
 - 4.1. Show
 - 4.2. ShowGroupNotInTheSameCompany
 - 4.3. Index
 - 4.4. ErrorIndex
- 5. Meetings
 - 5.1. Show
 - 5.2. ShowGroupNotInTheSameCompany
 - 5.3. Index
 - 5.4. IndexCurrent
 - 5.5. ErrorIndex
- 6. Planners
 - 6.1. Groups
 - 6.1.1. Show
 - 6.1.2. ShowGroupNotManagedByMe
 - 6.1.3. Index
 - 6.1.4. IndexCurrent
 - 6.1.5. ErrorIndex
 - 6.2. Meeting Timeslots
 - 6.2.1. EmployeeViewIndex
 - 6.2.2. GetNonPlannerInAGroup
 - 6.2.3. Show
 - 6.2.4. EmployeeViewShow
 - 6.2.5. Create
 - 6.2.6. PlannedCreate
 - 6.2.7. EmployeeCreate
 - 6.2.8. Update
 - 6.2.9. EmployeeUpdate
 - 6.2.10. Destroy
 - 6.2.11. PlannedDestroy
 - 6.2.12. EmployeeDestroy
 - 6.2.13. Index
 - 6.3. Meetings
 - 6.3.1. CreateMeeting
 - 6.3.2. IndexAllMeetings
 - 6.3.3. IndexCurrent
 - 6.3.4. ErrorIndexNoMeetings
 - 6.3.5. ShowMeeting
 - 6.3.6. ShowNonExistingMeeting

Planner	Version: 1.1
Test Report Document	Date: 20/01/2016

- 6.3.7. ShowOtherGroupsMeeting
- 6.3.8. PlannerDeleteMeeting
- 6.3.9. EmployeeDeleteMeeting
- 6.3.10. GetNonPlannerInAGroup
- 6.3.11. DeleteNonExistingMeeting
- 6.3.12. DeleteOtherGroupsMeeting
- 6.3.13. UpdateExistingMeeting
- 6.3.14. UpdatingOtherGroupsMeeting

3.1.2 Organization

- 1. Auth
 - 1.1. NewCompany
 - 1.2. ErrorNewCompany
 - 1.3. Login
 - 1.4. ErrorLogin
 - 1.5. ResetPassword
 - 1.6. ErrorResetPassword
 - 1.7. ThrottlesLogins
 - 1.8. NoThrottlesLogins
- 2. Company
 - 2.1. Index
 - 2.2. ErrorIndex
 - 2.3. Update
- 3. Employees
 - 3.1. Show
 - 3.2. Create
 - 3.3. SameEmailDifferentCompany
 - 3.4. Delete
 - 3.5. DeleteNotMine
 - 3.6. Update
 - 3.7. Index
 - 3.8. ErrorIndex
- 4. Groups
 - 4.1. Employees
 - 4.1.1. OkStore
 - 4.1.2. Store422
 - 4.1.3. Store403
 - 4.1.4. Delete
 - 4.1.5. DeleteEmployeeNotMine
 - 4.1.6. DeleteGroupNotMine

Plunner	Version: 1.1
Test Report Document	Date: 20/01/2016

- 4.1.7. Delete404
- 4.1.8. Index
- 4.1.9. ErrorIndex
- 4.2. Groups
 - 4.2.1. IndexAllGroups
 - 4.2.2. ErrorIndexNoCompany
 - 4.2.3. ShowSpecificGroup
 - 4.2.4. TryToShowOtherCompaniesGroup
 - 4.2.5. CreateNewGroup
 - 4.2.6. CreateDuplicateGroup
 - 4.2.7. SameGroupDataDifferentCompanies
 - 4.2.8. ErrorCreateGroup
 - 4.2.9. Delete
 - 4.2.10. DeleteNotMine
 - 4.2.11. Update

3.2 Frontend

1. LoginServiceTest
2. LogoutServiceTest
3. ErrorHandlingServiceTest

4. Other tests

4.1 Responsiveness tests

The web interface of Plunner is responsive at least for resolution $\geq 320\text{px} \times 480\text{px}$ (iPhone 4 resolution). This result has been obtained through manual tests using Google Chrome's device mode util.

4.2 Browser support

Plunner supports all modern browsers:

Plunner	Version: 1.1
Test Report Document	Date: 20/01/2016

- Internet Explorer version ≥ 9
- Mozilla Firefox version ≥ 4
- Google Chrome ≥ 19
- Apple Safari ≥ 6
- Opera ≥ 15

These results derive from the tools the tools and the framework used for the web interface of Plunner.