**INTRODUCTION**This is the Test Plan for the qaAuto application, User settings section . This plan will address only those items and elements that are related to the User settings, both directly and indirectly affected elements will be addressed. The primary focus of this plan is to ensure that the User settings section provides the same level of information and detail as the current system while allowing for improvements and increases in data acquisition and level of details available. The User settings section will have five sections of testing, Currency, Units of distance, Change email, Change password, Remove account.

**TEST ITEMS**

* [Currency settings](https://confluence.ithillel.com/display/HA/User+settings#Usersettings-Currency)
* [Units of distance settings](https://confluence.ithillel.com/display/HA/User+settings#Usersettings-Unitsofdistance)
* [Change email](https://confluence.ithillel.com/display/HA/User+settings#Usersettings-Changeemail)
* [Change password](https://confluence.ithillel.com/display/HA/User+settings#Usersettings-Changepassword)
* [Remove account](https://confluence.ithillel.com/display/HA/User+settings#Usersettings-Removeaccount)

**APPROACH**

Testing for the User settings section will consist of functionality testing to verify basic functions such as currency selection, email  and password changes.

* check the design of the User settings section for compliance with requirements
* check the chosen currency will change in the next tabs: [User garage#Addanexpense](https://confluence.ithillel.com/display/HA/User+garage#Usergarage-Addanexpense) and [User fuel expenses#Fuelexpensestable](https://confluence.ithillel.com/display/HA/User+fuel+expenses#Userfuelexpenses-Fuelexpensestable)
* check the validity of the email address
* check that the system validates the correctness of the entered email and password
* check that the user receives an error message if the data is entered incorrectly
* check that the user receives confirmation of the successful change of email and password
* check that the user can delete the account

**ENTRY/EXIT CRITERIA**

**entry criteria**

* All necessary dependencies and resources, such as section requirements, design, and documentation, are available to the testing team.
* The application has been developed and pre-release tested by the developers.
* All necessary testing tools are configured and ready to use.
* The test data that will be used during testing is prepared and available

**exit criteria**

* All planned test cases have been executed and passed successfully.
* The number of detected and fixed defects is less than the specified limit for this testing phase.
* All necessary reports and documentation have been prepared and verified.
* There are no critical/blocked defects.
* Testing has been completed within the specified time frame.

**ENVIRONMENTAL NEEDS**

The following elements are required to support the overall testing effort at all levels within the User settings section:

1. minimum amount of RAM, number of processor cores.  
2. support for Windows, macOS, Linux, and Chrome, Firefox, and Safari browsers

**RESPONSIBILITIES**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | TM | PM | Dev Team | Test Team | Client |
| Acceptance test Documentation & Execution | x | x |  | x | x |
| System Design Reviews | x | x | x | x | x |
| Detail Design Reviews | x | x | x | x |  |
| Test procedures and rules | x | x | x | x |  |
| Screen & Report prototype reviews |  |  | x | x | x |
| Change Control and regression testing | x | x | x | x | x |

The development team leader will be responsible for the verification and acceptance of all unit test plans and documentation.  
The project manager/test manager is responsible for all test plans and documentation.  
The entire project team will participate in the review of the system and detail designs as well as review of any change requests that are generated by the user or as a result of defects discovered during development and testing.

**RISKS AND CONTINGENCIES**

* Incompatibility between parts of the system, incorrect response to certain scenarios.
* Lack of stability of the program, its crash, can lead to a negative user experience.
* Performance issues that may cause delays or limitations in the operation of the software product.
* Product requirements may be misunderstood or incorrectly implemented, which will lead to a mismatch of customer expectations.