1. **Please create TC design to cover this functionality.**

**Date passed YYYY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Current Date  (YYYY-MM-DD) | User input | Expected calculated result | Comment |
| "Current year" | 2019-03-01 | 2019 | 2019-01-30 | Simple case to get half of the period passed. |
| 2020-03-01 | 2020 | 2020-01-31 | Long year (366 days) |
| 2019-01-01 | 2019 | 2019-01-01 | First day of the year (for testing boundary case) |
| 2020-12-31 | 2020 | 2020-07-01 | Last day of the year |
| 2020-04-29 | 2020 | 2020-02-29 | Result is 29th February (long year) |
| 2020-02-29 | 2020 | 2020-01-30 | Today is 29th February (long year) |
| "Previous year" | 2020-03-01 | 2019 | 2019-12-31 | Previous year, result should be 31 December of the previous year. |
| 2020-01-01 | 2019 | 2019-12-31 | First day of the year |
| 2020-12-31 | 2019 | 2019-12-31 | Last day of the year |
| 2020-02-29 | 2019 | 2019-12-31 | Today is 29th February (long year) |
| 2021-02-02 | 2020 | 2020-12-31 | Long year in result |
| "-2 or more year" | 2020-03-01 | 2018 | 2018-06-30 | Exactly -2 years, result should be 30th June of the entered year |
| 2020-03-01 | 2017 | 2017-06-30 | More than 2 years |
| 2020-01-01 | 2017 | 2017-06-30 | First day of the year |
| 2020-12-31 | 2017 | 2017-06-30 | Last day of the year |
| 2020-02-29 | 2017 | 2017-06-30 | Today is 29th February (long year) |
| 2023-02-02 | 2020 | 2020-06-30 | Long year in result |
| "Future year" | 2020-03-01 | 2021 | 2021-06-30 | Exactly next year |
| 2020-03-01 | 2030 | 2030-06-30 | Simple future year |
| 2020-01-01 | 2021 | 2021-06-30 | First day of the year |
| 2020-12-31 | 2021 | 2021-06-30 | Last day of the year |
| 2020-02-29 | 2021 | 2021-06-30 | Today is 29th February (long year) |
| 2019-02-02 | 2020 | 2020-06-30 | Long year in result |

**Date passed YYYY+MM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Current Date  (YYYY-MM-DD) | User input | Expected calculated result | Comment |
| "Current year" + "Current month" | 2019-02-02 | 2019-02 | 2019-02-02 | Simple scenario for getting current date in result |
| 2019-01-01 | 2019-01 | 2019-01-01 | First day of the year |
| 2019-12-31 | 2019-12 | 2019-12-31 | Last day of the year |
| 2020-02-29 | 2020-02 | 2020-02-29 | Today is 29th February (long year) |
| "Current year" + "Previous month" | 2019-03-02 | 2019-02 | 2019-02-15 | Simple case ( result must be 15th date of provided month) |
| 2019-01-01 | 2018-12 | 2018-12-15 | First day of the + year change |
| 2019-12-31 | 2019-11 | 2019-11-15 | Last day of the year |
| 2020-02-29 | 2020-01 | 2020-01-15 | Today is 29th February (long year) |
| "Current year" + "Current month -2 or more" | 2019-05-01 | 2019-03 | 2019-03-15 | Exactly -2 months ( result must be 15th date of provided month) |
| 2019-05-01 | 2019-02 | 2019-02-15 | More than -2 months |
| 2019-01-01 | 2018-05 | 2018-05-15 | First day of the year + year change |
| 2019-12-31 | 2019-05 | 2019-05-15 | Last day of the year |
| 2020-02-29 | 2019-05 | 2019-05-15 | Today is 29th February (long year) |
| "Current year" + "future month" | 2019-03-02 | 2019-04 | 2019-04-15 | Exactly next month |
| 2019-03-02 | 2019-08 | 2019-08-15 | Future month |
| 2019-01-01 | 2019-12 | 2019-12-15 | First day of the year |
| 2019-12-31 | 2020-02 | 2020-02-15 | Last day of the year + year change |
| 2020-02-29 | 2020-05 | 2020-05-15 | Today is 29th February (long year) |
| "Previous year" + "any month" | 2019-03-02 | 2018-05 | 2018-12-15 | Previous year + future month |
| 2019-03-02 | 2018-02 | 2018-02-15 | Previous year + past month |
| 2019-03-02 | 2018-03 | 2018-03-15 | Previous year + same month |
| 2019-03-02 | 2018-01 | 2018-01-15 | Previous year + first month |
| 2019-03-02 | 2018-12 | 2018-12-15 | Previous year + last month |
| 2020-02-29 | 2019-06 | 2019-06-15 | Today is 29th February (long year) |
| 2019-01-01 | 2018-03 | 2018-03-15 | First day of the year |
| 2019-12-31 | 2018-03 | 2018-03-15 | Last day of the year |
| 2021-02-02 | 2020-02 | 2020-02-15 | Result year is long year |
| "-2 or more year" + "any month" | 2021-05-02 | 2019-02 | 2019-02-15 | Exactly -2 years |
| 2019-05-02 | 2015-07 | 2015-07-15 | Future month |
| 2019-05-02 | 2015-02 | 2015-02-15 | Past month |
| 2019-05-02 | 2015-05 | 2015-05-15 | Same month |
| 2019-05-02 | 2015-01 | 2015-01-15 | First month |
| 2019-05-02 | 2015-12 | 2015-12-15 | Last month |
| 2020-02-29 | 2015-05 | 2015-05-15 | Today is 29th February (long year) |
| 2019-01-01 | 2015-01 | 2015-01-15 | First day of the year |
| 2019-12-31 | 2015-12 | 2015-12-15 | Last day of the year |
| 2025-05-05 | 2020-02 | 2020-02-15 | Result year is long year |
| "Future year" + "any month" | 2019-05-02 | 2030-06 | 2030-06-15 | Future month |
| 2019-05-02 | 2030-02 | 2030-02-15 | Past month |
| 2019-05-02 | 2030-05 | 2030-05-15 | Same month |
| 2019-05-02 | 2030-01 | 2030-01-15 | First month |
| 2019-05-02 | 2030-12 | 2030-12-15 | Last month |
| 2020-02-29 | 2030-02 | 2030-02-15 | Today is 29th February (long year) |
| 2019-01-01 | 2030-06 | 2030-06-15 | First day of the year |
| 2019-12-31 | 2030-07 | 2030-07-15 | Last day of the year |
| 2018-05-05 | 2020-08 | 2030-08-15 | Result year is long year |

**Other cases to try**

|  |
| --- |
| Submit missing values (e.g. only month, without year) |
| Input (edit) date after it is already picked with calendar widget |
| Input invalid characters |
| Input non existing date (e.g.13th month, 35th date) |
| Input 29th February for short year |
| Check boundary values (if exists) |
| Input not accepted format (e.g. YYYY+M) |

1. **Please define high-level automation strategy (short overview);**

If it is possible to provide entered date to the backend via API, most of calculation verifications should be done by API tests (sending date values and getting calculated values).

With UI testing (e.g. Selenium) verifications should mostly focus on visual aspects and check:

* If fields, labels are visible, enabled/clickable;
* Calendar widget can be opened/ closed;
* Date can be picked by calendar widget;
* Date can be entered/ updated/ deleted by user input and widget;
* After submitting only part of date, value in UI is showed correctly (user must see his/her entered value not the calculated one);

Also need to clarify if input validation is handled in frontend or backend, to check accordingly with API or UI test.

1. **Please list most likely problems you could expect during such functionality testing.**

To fully test date calculation functionality, it is needed to have ability to change/set “current date” (simulate different values for System.now).

In the given calculation logic examples there are cases which, from user perspective, could be one case (e.g. "Current year" + "Previous month" and "Current year" + "Current month -2 or more"), because from user perspective they are just “Current year + past month” with the same calculation logic. But if cases are calculated separately in backend they also should be separately tested to ensure coverage.