**UI Test Cases:**

1. **Verify Application Title**: Ensure that the title of the application on the left-hand side of the header displays 'Lander'.
2. **Verify Navigation Links**: Click on each link in the middle section (Dashboard, Plot Size, Investments, Claims) and verify that they navigate to the respective pages without any errors.
3. **Verify User Project Information**: Verify that the name of the project 'Hofgut Neuhausen' is displayed on the right-hand side of the header.
4. **Check the dropdown**: Test the functionality of the project dropdown, ensuring that it displays related projects and clicking on a project updates the displayed project.
5. **Verify User Initials Icon**: Check that the black circle containing the user's initials is displayed correctly. Test the dropdown attached to it, ensuring it shows options like 'My Profile', 'Logout', etc.
6. **Verify Project Details**: Ensure that the 'Project' section on the left displays the correct project name and its size (for eg. 300.3 ha). Also, check that the 'Landtypes' section on the right displays the icons and their descriptions correctly.
7. **Verify Edit Project and Send to Sponsor Buttons**: Test the functionality of the 'Edit Project' button and 'Send to Sponsor' button, ensuring they perform the intended actions without errors.
8. **Verify My Plot Cards**: For each row in the 'My Plot' section, verify that the image, name, size, type icon, data completion percentage, and information icon are displayed correctly. Test the functionality of the information icon to ensure it provides the correct details for each row.
9. **Verify Add Plot Button**: Click on the '+Add Plot' button and verify that it opens a form or a modal to add a new plot.
10. **Verify Footer Links**: Click on the links in the footer ('Imprint', 'Privacy', etc.) and verify that they navigate to the respective pages.

**UI Edge Cases:**

1. **Empty Project Name**: Test the UI when the project name is empty. Verify that appropriate validation messages are displayed, and the user cannot proceed until a valid project name is provided.
2. **Long Project Name**: Test the UI with a very long project name to check how the application handles and displays lengthy project names.
3. **No Land types Available**: Verify the behaviour of the UI when there are no land types available to display in the 'Landtypes' section.
4. **No Plots Available**: Test the UI when there are no plots available for the selected project. Ensure the application displays a relevant message.
5. **Maximum Rows in My Plot**: Verify how the UI handles displaying a large number of rows in the 'My Plot' section, up to the maximum supported by the application.

**UI Negative Cases**:

1. **Invalid Login Credentials**: Test the UI with invalid login credentials to ensure proper error messages are displayed when attempting to log in.
2. **Unauthorized Access**: Attempt to access pages or functionalities without proper authorization. Verify that unauthorized users are redirected to the login or denied access.
3. **Incorrect Data Input**: Provide incorrect or invalid data while adding a new plot using the '+Add Plot' button. Check that the application handles the input properly and displays appropriate error messages.
4. **Incorrect URL Navigation**: Manually enter invalid URLs or page paths to check how the UI handles incorrect navigation attempts.

**API Test Cases:**

1. **Verify API Endpoints**: Ensure that all the required API endpoints exist and are functional for the UI components, such as getting project details, plot information, user information, etc.
2. **Verify Response Data**: Test the API responses for correctness, including proper project data, plot details, and user information. Verify that the response formats match the expected schema.
3. **Verify Data Completeness API**: For the 'Data Complete' feature, test the API that calculates the data completion percentage for each plot.
4. **Verify Add Plot API**: Test the API endpoint responsible for adding new plot information. Check that the data is correctly stored in the database.

**API Edge Cases:**

1. **Empty Project Data**: Test the API with empty project data or no data for specific project-related endpoints.
2. **Null or Missing Data Fields**: Verify the API's behaviour when sending requests with null or missing data fields.
3. **Max Data Length**: Test the API with data that reaches the maximum allowable length for various fields.

**API Negative Cases**:

1. **Invalid API Endpoints**: Attempt to access non-existent API endpoints or endpoints with incorrect HTTP methods. Verify that the API returns proper error responses.
2. **Invalid Authentication Tokens**: Test the API with invalid or expired authentication tokens. Ensure the API denies access and returns the appropriate authentication error.

**Database Test Cases:**

1. **Verify Project Data in DB**: Verify that the correct project 'Hofgut Neuhausen' with the corresponding size (300.3 ha) is present in the database.
2. **Verify Plot Data in DB**: Check that the data for each plot displayed in the 'My Plot' section corresponds accurately with the data stored in the database.
3. **Verify Data Completeness in DB**: Test the data completeness calculations in the database, ensuring the percentages are accurate and up-to-date.

**Database Edge Cases:**

1. **Maximum Project Size**: Test the database with a project size that reaches the maximum allowable limit.
2. **Large Number of Plots**: Verify the database's performance and handling of a large number of plot records.

**Database Negative Cases:**

1. **Data Inconsistencies**: Introduce data inconsistencies in the database, such as invalid plot information or incorrect project associations. Check how the application handles such cases.

**Performance Test Cases:**

1. **Verify UI Responsiveness**: Test the UI responsiveness under different network conditions and ensure the pages load within acceptable time frames.
2. **Verify API Response Time**: Measure the response time of critical API endpoints to ensure they meet performance criteria.
3. **Load Testing**: Perform load testing to determine the application's behavior under heavy user loads and identify any performance bottlenecks.

**Performance Edge Cases:**

1. **High User Load**: Test the application's performance under peak load conditions to ensure it can handle the maximum number of concurrent users.
2. **Slow Network Connection**: Simulate slow network conditions to evaluate the application's response time in such situations.

**Performance Negative Cases:**

1. **Performance Degradation**: Continuously perform actions or API calls to test for performance degradation over time.
2. **Large Data Volume**: Test the application's performance with a large volume of data, such as a high number of plots or projects.