PLANE APP RISK ANALYSIS

[1. Brief Description of SUT 2](#_531oso1c9hd6)

[2. Attributes, Components and Capabilities 4](#_4i89rh2ket21)

[2.1 Attributes 4](#_s62rslwbhxc5)

[2.2 Components 5](#_4ojphkuu3cjp)

[2.3 Capabilities 6](#_4yxetgeoqc9h)

[3. High-level Risks 7](#_gp8mguv92or2)

[4. Risk Analysis 8](#_phi830labv2e)

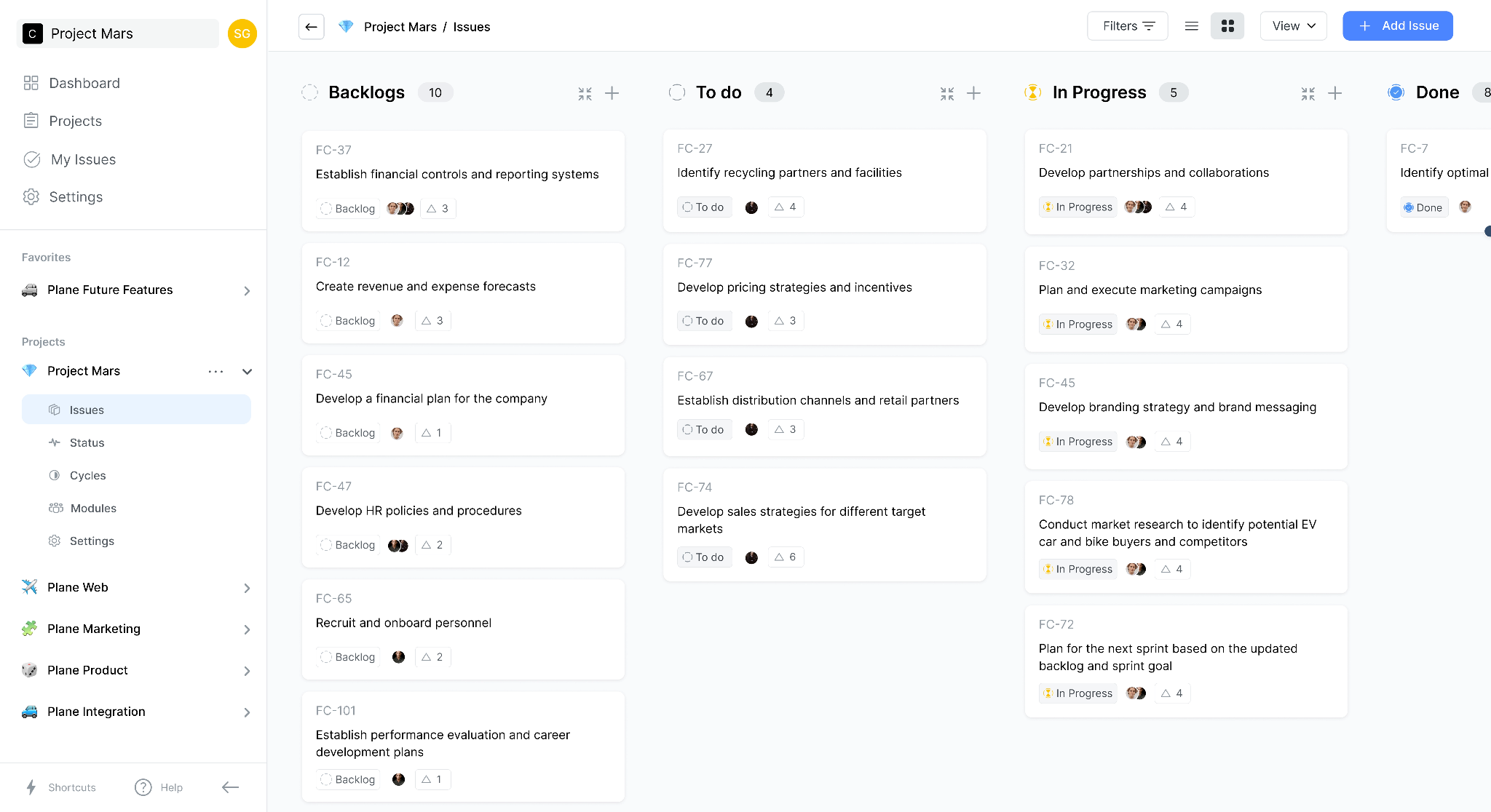
[References](#_hz6y06rtn9kv)23

## Brief Description of SUT

Plane is an open-source software development tool to manage issues, sprints, and product roadmaps. It is positioned as a Jira alternative.

**Application website:** [**plane.so**](https://plane.so/)**.**

**Documentation:** [**docs.plane.so**](https://docs.plane.so/)



**Main selling points**

* Simple UX
* Customization using Cycles and Modules
* Integrations, such as GitHub, Slack; Importers to migrate from other products
* Self-hosted and cloud versions

**Main features**

- Issue Planning and Tracking

- Issue Attachments

- Layouts

- Cycles

- Command Panel

- GitHub Sync and other integrations

- AI Assistant Notepad

- Team Members

**Tech stack**

- Frontend: JavaScript / Vue.js / Tailwind CSS

- Backend: Python / Django/ / Model-View-Controller (MVC) pattern

- Web Server: Gunicorn - The API code uses Gunicorn as the web server. Gunicorn is a Python WSGI (Web Server Gateway Interface) HTTP server that allows the application to handle multiple concurrent requests efficiently.

- Database: PostgreSQL

- Authentication: JWT

- Integrations (cloud version only): GitHub, Slack, Jira importer (coming soon)

## Attributes, Components and Capabilities

| 1. *What application properties need verification?* 2. *The* ***adjectives*** *and adverbs that describe why a user would use the service in the first place.* 3. *Recommended: 10-12 items max.* |
| --- |

### 2.1 Attributes

1. Available (main features functioning as expected)
2. Customizable (cycles, modules, views, pages, states, settings: user, project, workspace)
3. Responsive (data volume testing)
4. Reliable (stored data CRUD)
5. User-friendly, simple and intuitive (UI)
6. Secure (authentication, roles, permissions)
7. Collaborative (invitation, roles, permissions)
8. Extensible (Integrations work as expected)
9. Compatibility (hosted versions can be installed and used on different platforms)
10. Easy to setup (self-hosted)

### 2.2 Components

| 1. *What the* ***nouns*** *that name components of the system under test?* 2. *What the parts of the application that require testing attention* 3. *Recommended: 10-20 items.* |
| --- |

* Version
  + Self-hosted
  + Cloud
* Authentication
* Roles, permissions
* Workspace
  + Dashboard
  + Projects
  + Analytics
  + Issues
  + Notifications
  + Settings
* Project
  + Issue /Issues
  + Cycles / Cycle
  + Modules / Module
  + Views / View
  + Pages / Page
  + Settings
  + Members management
* Keyboard shortcuts / Command palette
* Menu
* Themes
* Docs

### 2.3 Capabilities

| 1. *What purpose does the application serve for users?* 2. *The verbs describing what capabilities users receive (actions that get translated into test cases)* |
| --- |

Capabilities will be listed in Test Plan document.

***Test planning should put us in a position to know exactly what tests must be written [1].***

## High-level Risks

| # | RISK | QUALITY AREA | LIKELIHOOD  1..5 | IMPACT  1..5 | RISK SCORE  1..25 |
| --- | --- | --- | --- | --- | --- |
| 1 | UI functional & customization failures | functional | 4 | 4 | 16 |
| 2 | "Collaboration" and perissions features problems | functional, security | 3 | 3 | 9 |
| 3 | Performance issuues | performance | 4 | 4 | 16 |
| 4 | Poor usability | usability | 4 | 3 | 12 |
| 5 | Installation/update problems | installability | 4 | 4 | 16 |
| 6 | Integrations with/migration from other tools | functional | 3 | 3 | 9 |
| 7 | Self-hosting problems | functional, compatinility | 4 | 5 | 20 |

## Risk Analysis

| # | RISK | QUALITY AREA | ANALYSIS | LIKELIHOOD  1..5 | IMPACT  1..5 | RISK SCORE  1..25 |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Self-Hosted version issues with setup via Docker |  | Failure to launch/ complete installation  Emails not functional  Install via Docker fails  MacOS  Linux  Windows  Raspberry pi | 3 | 5 | 15 |
| 2 | Self-Hosted version missing or has poor documentation | usability | Missing docs  Poor docs  Outdated docs | 5 | 2 | 10 |
| 3 | Issue with updating self-hosted version | installability | Unable to update app  Data is corrupted after update  Critical functionality is broken | 4 | 5 | 20 |
| 4 | Login Security | security | Invalid login not allowed  Limit number of attempts  Password hash is used  Reset password link expires | 2 | 3 | 6 |
| 5 | Failure to CRUD issue | functional | Create simple issue  View issue  Update issue  Delete issue | 4 | 4 | 16 |
| 6 | Issues related to views | functional | List view  Kanban view  Calendar view  Spreadsheet view  Gantt Chart View  Filters  Views | 4 | 3 | 12 |
| 7 | Analytics is not accurate | functional | Various issue states  Most issues created by user  Most issues closed by user  Custom Analytics - Project  Custom Analytics - Measure  Custom Analytics - Dimension  Custom Analytics - Group | 4 | 2 | 8 |
| 8 | Failure to add workspace | functional | Create single new workspace  Create new workspace with each size (6)  Create multiple workspaces | 2 | 4 | 8 |
| 9 | Failure to customize workspace - manage members | functional | Members - Add new members  Members - Change member role  Members - Remove member | 3 | 4 | 12 |
| 10 | Failure to customize workspace - general settings | functional | General - Update  Logo  Name  Description  Size | 2 | 2 | 4 |
| 11 | Failure to customize workspace - change plan (cloud version) | functional | Billing - Change plan | 2 | 3 | 6 |
| 12 | Failure to customize workspace - configure, import/export | extensible | Configure  GitHub integration, Import/Export  Jira integration, Import/Export  Slack integration | 4 | 2 | 8 |
| 13 | Failure to add project | functional | Create single new project  Create public project  Create private project | 2 | 4 | 8 |
| 14 | Failure to customize project - general settings | customizable | General - Update  Cover photo  Icon&Name  Description  Network: Public/Private  Identifier | 3 | 3 | 9 |
| 15 | Failure to customize project - manage members | customizable | Members - Add new members  Members - Change member role  Members - Remove member  Project lead - assign  Default assignee - set | 3 | 4 | 12 |
| 16 | Failure to customize project - features, states, labels | customizable | Features - toggle  Cycles,  Modules,  Views,  Pages,  Inbox  State - add, remove  State - rename  Label - CRUD | 3 | 4 | 12 |
| 17 | Failure to customize project - integrations | extensible | Configure  GitHub integration  Slack integration | 4 | 2 | 8 |
| 18 | Failure to create estimate | functional | Create estimate | 2 | 2 | 4 |
| 19 | Automation failure | functional | Auto-close inactive issues  Auto-archive closed issues | 3 | 1 | 3 |
| 20 | Poor accessibility | accessibility | Insufficient text size, color contrast  Interactive elements are accessible through the keyboard | 5 | 2 | 10 |
| 21 | Command palette failure | usability | Trigger by Cmd+K  Search failure  Unable to execute command via palette | 3 | 3 | 9 |
| 22 | Slow response (performance issues) | performance | When viewing multiple issues page load takes too long  When viewing single issue page load takes too long  When updating issue it takes too long | 4 | 4 | 16 |
| 23 | Failure to update issue with various inputs | functional | Issue title  Issue description  Attachmentt (various supported types)  Update state  Update priority  Update parent  Update Cycle | 4 | 4 | 16 |
| 24 | Failure to add comment in issue |  | Add comment | 3 | 4 | 12 |
| 25 | Poor Usability - confusing | usability | Confusing UI  Not intuitive UI | 4 | 3 | 12 |
| 26 | Poor Usability - inconsistency, spelling errors | usability | Inconsistent language  Grammar mistakes  Inconsistant use of fonts. colours | 4 | 1 | 4 |
| 27 | Incorrect Handling of Roles/Permissions/Access | collaborative,  security | Owner  Admin  Member  Guest  External user | 4 | 2 | 8 |
| 28 | Compatibility issues/Themes | compatibility | Chrome  Edge  Firefox  Safari  Screen size | 2 | 3 | 6 |
| 29 | Documentation Issues | usability | Missing docs  Unclear docs  Outdated | 5 | 2 | 10 |
| 30 | UI issues on mobile devices | compatibility | iPhone  iPad  Samsung  Surface tablet | 3 | 2 | 6 |
| 31 | AI assistant is not functional | functional | AI assistant is not responding | 3 | 2 | 6 |

1. Risk Mitigation

| # | RISK | RISK SCORE,  PRIORITY | MITIGATION | COMMENTS |
| --- | --- | --- | --- | --- |
| 1 | Issue with updating self-hosted app | 20 | Regression testing  API testing | Automated  Intensity: medium |
| 2 | Slow response (performance issues) | 16 | Regression testing  Performance testing  Acceptance Testing | Automated  Intensity: medium |
| 3 | Failure to update issue with various inputs | 16 | Regression testing  API testing  Exploratory testing  Acceptance Testing | Automated  Intensity: high |
| 4 | Failure to CRUD issue | 16 | Regression testing  Integration testing  Unit testing  UI Testing  API testing | Automated  Intensity: high |
| 5 | Self-Hosted version issues with installation/setup via Docker | 15 | Review docs regularly  Monitor feedback from community via GitHub, Discord  Acceptance testing  Regression testing  Compatibility testing | Manual  Automated  Intensity: medium |
| 6 | Issues related to views | 12 | Regression testing  Integration testing  E2E Testing  Unit testing  UI Testing  API testing | Automated  Intensity: medium |
| 7 | Failure to customize workspace - manage members | 12 | Regression testing  Integration testing  E2E Testing  Unit testing  UI Testing?  API testing  Exploratory testing | Automated  Intensity: medium |
| 8 | Failure to customize project - manage members | 12 | Regression testing  Integration testing  E2E Testing  Unit testing  UI Testing?  API testing  Exploratory testing | Automated  Intensity: medium |
| 9 | Failure to customize project - features, states, labels | 12 | Regression testing  Integration testing  E2E Testing  Unit testing  UI Testing?  API testing  Exploratory testing | Automated,  Manual  Intensity: medium |
| 10 | Failure to add comment in issue | 12 | Regression testing  Integration testing  E2E Testing  Unit testing  UI Testing  API testing  Exploratory testing | Automated,  Manual  Intensity: medium |
| 11 | Poor Usability - confusing | 12 | Usability testing  Exploratory testing  Engaging with the community on Discord channel, Twitter, Linkedin, YouTube, reviewing GitHub issues | Automated  Intensity: medium |
| 12 | Self-Hosted version missing or has poor documentation | 10 | Review docs regularly  Monitor feedback from community via GitHub, Discord | Manual  Automated alerts  Intensity: low |
| 13 | Poor accessibility | 10 | Accessibility testing  Exploratory testing | Manual  Automated alerts  Intensity: medium |
| 14 | Documentation Issues (other than self-hosted docs) | 10 | Review docs regularly  Monitor feedback from community via GitHub, Discord | Manual  Automated alerts  Intensity: low |
| 15 | Failure to customize project - general settings | 9 | Regression testing  Integration testing  E2E Testing  Unit testing  UI Testing  API testing | Automated  Intensity: medium |
| 16 | Command palette failure | 9 | Regression testing  UI Testing | Automated  Intensity: low |
| 17 | Analytics is not accurate | 8 | Regression testing  UI Testing  API testing  Exploratory testing | Automated  Manual  Intensity: low |
| 18 | Failure to add workspace | 8 | Regression testing  UI Testing  API testing  Exploratory testing | Automated  Manual  Intensity: low |
| 19 | Failure to customize workspace - configure, import/export | 8 | Regression testing  UI Testing  API testing  Exploratory testing | Automated  Manual  Intensity: low |
| 20 | Failure to add project | 8 | Regression testing  E2E Testing  Smoke testing  UI Testing  API testing  Exploratory testing | Automated  Manual  Intensity: low |
| 21 | Failure to customize project - integrations | 8 | Regression testing  E2E Testing  UI Testing  API testing  Exploratory testing | Automated  Manual  Intensity: medium |
| 22 | Incorrect Handling of Roles/Permissions/Access | 8 | Regression testing  E2E Testing  UI Testing  API testing  Exploratory testing | Automated  Manual  Intensity: high |
| 23 | Login Security | 6 | Security testing  API testing  Exploratory testing | Automated  Manual  Intensity: medium |
| 24 | Compatibility issues/Themes | 6 | Usability testing  UI testing  Visual testing  Exploratory testing  Engaging with the community on Discord channel, Twitter, Linkedin, YouTube, reviewing GitHub issues | Automated  Manual  Intensity: low |
| 25 | UI issues on mobile devices | 6 | Usability testing  UI testing  Visual testing  Exploratory testing  Engaging with the community on Discord channel, Twitter, Linkedin, YouTube, reviewing GitHub issues | Automated  Manual  Intensity: low |
| 26 | AI assistant is not functional | 6 | Regression testing  API testing  Integration testing | Automated  Intensity: low |
| 27 | Failure to customize workspace - change plan (cloud version) | 6 | Regression testing  API testing  Integration testing | Automated  Intensity: low |
| 28 | Failure to customize workspace - general settings | 4 | Regression testing  UI testing  E2E testing  API testing  Integration testing | Automated  Intensity: low |
| 29 | Failure to create estimate | 4 | Regression testing  UI testing  E2E testing  API testing  Integration testing | Automated  Intensity: low |
| 30 | Poor Usability - inconsistency, spelling errors | 4 | Usability testing  UI testing  Exploratory testing  Engaging with the community on Discord channel, Twitter, Linkedin, YouTube, reviewing GitHub issues | Manual  Intensity: low |
| 31 | Automation failure | 3 | Regression testing  API testing  Integration testing | Automated  Intensity: low |

| **1** | **Likelihood Factors** |  |
| --- | --- | --- |
|  |  |  |
|  | *Unlikely* |  |
|  | *Low likelihood* |  |
|  | *Moderate likelihood* |  |
|  | *High likelihood* |  |
|  | *Very high likelihood* |  |
|  |  |  |
| 1.1 | Complexity of the feature/software: Complex software systems can increase the likelihood of risks due to the potential for more intricate interactions and dependencies. |  |
| 1.2 | Quality of requirements: Inadequate or ambiguous requirements can lead to misunderstandings, resulting in higher chances of risks during development and testing. |  |
| 1.3 | Skill and experience of the development team: The expertise and experience of the team can impact the likelihood of risks. A team with limited knowledge or experience may be more prone to making mistakes and overlooking potential issues. |  |
| 1.4 | Availability and accuracy of documentation: Poor documentation can lead to misunderstandings, confusion, and errors, increasing the likelihood of risks. |  |
| 1.5 | Time pressure and project deadlines: Tight schedules and aggressive deadlines can lead to rushed development and testing, increasing the likelihood of risks being overlooked. |  |
| 1.6 | Use of new or unfamiliar technologies: The use of unfamiliar technologies or tools can increase the likelihood of risks due to potential lack of understanding or experience. |  |
| 1.7 | Third-party dependencies: Reliance on external libraries, APIs, or services introduces additional risks, as issues in these dependencies can impact the software product. |  |
| 1.8 | Frequency of changes or updates: Frequent changes or updates to the feature/software can introduce new risks or amplify existing ones due to the dynamic nature of the development process. |  |
|  |  |  |
| **2** | **Impact Factors** |  |
|  | low |  |
|  | medium |  |
|  | high |  |
|  | critical |  |
|  |  |  |
| 2.1 | Criticality of the software: The potential impact of a risk is higher if the software is critical to the business or end-users. For example, if the software is used in healthcare or financial systems, any failure or data breach could have severe consequences. |  |
| 2.2 | User base and usage frequency: The number of users and the frequency of usage can determine the impact of a risk. If a large number of users rely on the software frequently, any disruption or failure can have a significant impact on productivity or customer satisfaction. |  |
| 2.3 | Financial impact: Risks that can result in financial losses, such as lost revenue or increased costs, should be considered high impact. |  |
| 2.4 | Regulatory compliance: Non-compliance with legal or industry regulations can have severe consequences, including legal penalties, reputation damage, and business disruptions. |  |
| 2.5 | Data integrity and security: Risks that can compromise data integrity, confidentiality, or availability should be considered high impact, especially if sensitive or personal information is involved. |  |
| 2.6 | Customer satisfaction and reputation: Risks that can negatively impact customer satisfaction, such as poor performance or usability issues, can damage the reputation of the software product and the organization. |  |
| 2.7 | Business continuity: Risks that can lead to prolonged system downtime or loss of critical functionality can significantly impact business operations and continuity. |  |
| 2.8 | Safety and legal implications: For features/software products used in safety-critical systems (e.g., aviation, medical devices), risks that can result in physical harm, injury, or legal liabilities should be considered high impact. |  |
|  |  |  |
| **3** | **Risk Score Ranges** |  |
|  | 0..6 | low |
|  | 7..12 | medium |
|  | 13..18 | high |
|  | 19..25 | critical |
|  |  |  |
| **4** | **Testing Types** |  |
| 4.1 | Acceptance Testing |  |
| 4.2 | API Testing |  |
| 4.3 | Compatibility testing |  |
| 4.4 | E2E Testing |  |
| 4.5 | Engaging with the community on Discord channel, reviewing GitHub issues |  |
| 4.6 | Engaging with the community Twitter, Linkedin, YouTube |  |
| 4.7 | Exploratory Testing |  |
| 4.8 | Feature testing |  |
| 4.9 | Integration Testing |  |
| 5 | Performance testing |  |
| 5.1 | Production Validation Testing  (PVT Smoke Test) |  |
| 5.2 | Regression Testing |  |
| 5.3 | Reviewing and updating documentation before each release |  |
| 5.4 | Security testing |  |
| 5.5 | Static Analysis |  |
| 5.6 | System Testing |  |
| 5.7 | UI Testing |  |
| 5.8 | Unit Testing |  |
| 5.9 | Usability testing |  |

## References

1. [Google Testing Blog: The 10 Minute Test Plan](https://testing.googleblog.com/2011/09/10-minute-test-plan.html)
2. [Google Sheets: Risk Analysis - Plane App](https://docs.google.com/spreadsheets/d/1bPsRtZfOYbaWbRAxnGnu2q8dFIjIbyDA69Ii303hJR8/edit#gid=1551150708)