Products

This page covers the things every team member needs to know in order to effectively contribute at Unicis.

Product roadmap

The Unicis product roadmap is transparent and can be found via the Unicis Feedback portal. However, strategy and priority information is handled internally on a Strategic Level and later reflected in the internal GitLab repository and on the Feedback portal.

What are product groups

Unicis organizes product development efforts into separate, cross-functional product groups, including product designers, developers, and quality testers. Product groups are grouped by business goal and designed to operate in parallel. Every product group is responsible for security and privacy by default and by design, performance, stability, scalability, database migrations, release compatibility, usage documentation, contributor experience, and support escalation. At Unicis, anyone can contribute and work across product groups.

Current product groups

Product group	Goal (value for customers and/or community)
Unicis Apps	Reach maturity Atlassian Apps in the GRC solution category.
Unicis Platform	Reach maturity in the GRC product category.

Unicis Apps group

Responsibility	Person(s)
Product Designer, Engineering Manager, Product Manager, Quality Assurance	Predrag - CEO and Founder
Developer	Yaroslav

Unicis Platform group

Responsibility	Person(s)
Product Designer, Product Manager, Quality Assurance	Predrag - CEO and Founder
Engineering Manager	Peter - CTO
Developer	Yaroslav and Abdul

Change Management

Unicis manages changes to the system in a controlled and systematic way. This helps ensure the reliability and integrity of our services. Each change is documented, evaluated for its impact, and approved before implementation.

When there is a pull request (PR), each code change or logic should be reviewed by the lead or someone else. The reviewer is required to provide feedback regarding the change review, including whether it has been tested and utilizing the appropriate abbreviations:

Short Messages	Meaning
LGTM	Looks Good To Me
ACK	Acknowledgement, i.e. agreed/accepted change
NACK/NAK	Negative Acknowledgement, i.e. disagree with change and/or concept
RFC	Request For Comments, i.e. I think this is a good idea, let's discuss
WIP	Work In Progress, do not merge yet
AFAIK/AFAICT	As Far As I Know / can tell
IIRC	If I Recall Correctly
IANAL	"I am not a lawyer", but I smell licensing issues
IANASP	"I am not a security & privacy expert", but I smell security and privacy issues

Planned and unplanned changes

Planned changes are scheduled and managed through our standard change management process. Unplanned changes, such as emergency fixes, are handled promptly but still require documentation and a post-implementation review to ensure no negative impact on the system.

Breaking changes

Breaking changes are those that could disrupt or alter existing functionality. These changes require extensive communication with stakeholders, extensive testing, and detailed migration guides to help users adapt to the new system.

API changes

Changes to our API can have a negative impact on everyone using our products. Always make sure these changes are compatible with previous versions. Clear versioning and detailed release notes are provided to guide users through the transition.

Drafting

Drafting means creating user stories, specifications, and designs. This phase captures the ideas and requirements, which are then refined and tested before they are implemented.

User story

A user story tells the story of a feature or function from the point of view of the end user. It has rules that say what needs to be done to make the story complete.

Defining "DONE"

"DONE" means that a user story or task has met all the requirements, passed all the tests, and is ready for release. It requires coding, testing, documentation, and approval from relevant stakeholders.

Outages

Outages are unexpected downtimes or problems with the service. Our team quickly fixes the problem and checks everything to make sure it doesn't happen again.

Status Page Our status page is used to communicate with our customers and integrates with our internal Element chat channel `Alert`.

Scaling Unicis

Scaling is adding more capacity to the system to handle more load. This includes improving performance, adding resources, and improving infrastructure to ensure seamless service for a growing user base.

Testing

Testing is important for making sure our product works well. This includes testing everything from the beginning to the end to make sure that new changes don't cause problems or break things.

Version support

We offer support for multiple versions of our software to help users who can't upgrade right away. Clear timelines and deprecation notices are given for phasing out old versions.

Release testing

Before a new version is released, it undergoes rigorous testing to identify and fix any issues. This includes regression testing, performance testing, and user acceptance testing.

Bug communication

Effective communication about bugs includes detailed reporting, status updates, and resolutions. Users are promptly informed of any issues affecting their experience.

Features

Features are improvements or new features we add to our products. Each feature goes through a lifecycle from request to release, ensuring it meets user needs and maintains system integrity.

Feature request

Users and other stakeholders are welcome to submit feature requests via the Unicis Feedback portal. These requests are reviewed, prioritized, and scheduled based on their impact and feasibility.

Feature communicate

Clear and open communication about features includes status updates, expected timelines, and any changes in scope. This keeps users updated and involved throughout the development process in the Unicis Feedback portal.

Feature accepted

A feature request is accepted once it has been reviewed, approved, and scheduled to be implemented. This means it has met the requirements and will be worked on in the future, and it is tagged in the portal as `Planned`. During development, we label the feature as `Started` in the feedback portal.

Feature released

A feature is considered released when it is fully developed, labeled as `Completed`, tested, and deployed to the production environment. Release notes and documentation are provided to help users understand and utilize the new functionality.

Quality

Quality is a top priority at Unicis. We maintain high standards through rigorous testing, code reviews, and continuous improvement practices to deliver reliable and effective solutions.

Bug status

Bug status indicates the current progress of a bug through its lifecycle, from reported to resolved. Clear status updates help users and developers track and prioritize issues effectively.

Bug reproduced

A bug is marked as reproduced once it has been successfully replicated by the QA team. This step is crucial for diagnosing and fixing the issue accurately.

In development

In development status means that a user story, feature, or bug fix is currently being worked on by the development team. This includes coding, initial testing, and code review.

In QA

In QA status indicates that a user story, feature, or bug fix is undergoing quality assurance testing. This phase ensures that the work meets all acceptance criteria and is free from defects.

High priority user stories and bugs

High priority user stories and bugs are those that have a significant impact on users or system performance. These are addressed urgently to ensure minimal disruption and maximum satisfaction.

Wireframes

Wireframes are visual representations of a user interface. They help in planning the layout and functionality of a feature before actual development begins, ensuring clarity and alignment among stakeholders.

Meetings

Meetings at Unicis are structured to enhance collaboration and communication. They include user story discovery sessions, design reviews, weekly bug reviews, and group weeklies.

User story discovery

User story discovery meetings involve brainstorming and defining user stories. This collaborative process ensures that all relevant requirements and perspectives are considered.

Design

Design meetings focus on creating and reviewing the visual and functional aspects of new features. This includes wireframes, mockups, and prototypes to ensure the best user experience.

Monthly bug review

Monthly bug review meetings are held to assess the status of reported bugs, prioritize fixes, and track progress. This ensures timely resolution and maintains system stability.

Group weeklies

Group weekly meetings provide a platform for team members to share updates, discuss challenges, and plan for the coming week. These meetings foster teamwork and alignment.

OKRs together

OKRs (Objectives and Key Results) are set and reviewed collaboratively to ensure that team goals align with company objectives. Regular check-ins keep everyone focused and on track.

Scrum at Unicis

Scrum is the agile framework used at Unicis to manage development. It promotes iterative progress, continuous

feedback, and rapid adaptation to change.

Scrum board

The Scrum board is a visual tool used to track the progress of tasks within a sprint. It includes columns for user stories, tasks, in-progress work, and completed items.

Sprints

Sprints are time-boxed iterations, typically two weeks long, during which specific work items are completed. Each sprint begins with planning and ends with a review and retrospective.

Community/Outside contribution

Community and outside contributions are encouraged and valued at Unicis. We provide clear guidelines and support for external developers to contribute to our projects, fostering innovation and collaboration.

unicis, sprints, scrum, community, communication, OKR, bug, review, code, meeting, agenda

From:
https://handbook.unicis.tech/ - **Unicis Handbook**

Permanent link:
https://handbook.unicis.tech/pub/company/products

Last update: **25.09.2024 12:26**