Risk Register

Date of last review:

ID	Description of Risk	Impact	Risk Reponse	Risk Level
1	App crashes or freezes due to unhandled exceptions or memory leaks.	Poor user experience and increased app uninstalls.	Implement thorough testing and crash reporting tools like Firebase Crashlytics.	High
2	Notifications not triggering as expected on all platforms.	Users miss reminders, defeating the app's core purpose.	Use reliable notification packages and test across devices.	High
3	Compatibility issues across different OS versions.	Unexpected bugs on some user devices.	Test app on a wide range of devices and OS versions.	Medium
4	Data not syncing properly with the backend database.	Loss of user input, inaccurate tracking data.	Use robust API design and offline-first strategies.	High
5	Inadequate hydration goal personalization.	Reduced user engagement and app abandonment	Implement customizable goals and smart suggestions.	Medium
6	Poor internet connectivity affecting app performance.	Inability to sync or fetch data.	Support offline mode with local storage and sync on reconnect.	Medium
7	Security vulnerabilities in database or user data handling.	User data leaks, legal and trust issues.	Use encryption, secure APIs, and follow best practices for user data protection.	High
8	Push notifications blocked by battery optimization settings.	Users miss important reminders.	Provide clear onboarding instructions to whitelist app.	Medium
9	App store rejection due to policy non-compliance.	Delays in release and potential feature rework.	Review guidelines thoroughly before submission.	Medium
10	User retention drops after initial use.	Low daily and monthly active users, leading to decreased revenue potential.	Gamification (streaks/rewards), periodic engagement prompts.	Mediuem
11	Timezone/date handling bugs.	Incorrect daily resets or reminders.	Use UTC, test timezone changes, intl package.	Medium
12	UI inconsistencies across different screen sizes or orientations.	Confusing UX and poor aesthetics.	Use responsive design and extensive UI testing.	Medium
13	Slow load times due to inefficient frontend code.	User frustration, possible churn.	Optimize rendering and use performance profiling tools.	Medium
14	Incorrect logic in water intake calculations.	Users may get misleading intake recommendations.	Double-check formula implementations and validate results.	High

15	Unauthorized access to admin endpoints.	Security breach or data manipulation.	Use role-based access control and API authentication.	High
16	Missing validations on user input fields.	Unexpected behavior, potential crashes.	Implement frontend and backend validation rules.	Medium
17	Unclear error messages shown to users.	Confusion and lack of trust.	Design user-friendly error handling and display.	Low
18	Data inconsistency between local storage and server.	Wrong progress shown to users.	Use sync flags and conflict resolution mechanisms.	High
19	Data corruption during sync or updates	Users see incorrect or missing data	Implement transactional updates, versioning, and data backups	High
20	Inadequate test coverage leading to undetected bugs	Production issues post-release	Establish test case coverage thresholds, automate CI/CD testing	High
21	Deployment failure or rollback issues	App becomes temporarily unusable	Implement safe CI/CD pipelines and blue-green deployments	Medium
22	Accessibility non-compliance (e. g., screen reader support)	Alienation of users with disabilities	Follow accessibility standards (WCAG) during design & dev	Medium
23	Overly frequent reminders causing notification fatigue	Users disable reminders or uninstall app	Allow user to set frequency and quiet hours	Medium
24	Market saturation with similar apps	Reduced user acquisition and retention	Highlight unique features (personalization, gamification), run competitor analysis	Medium
25	Monetization strategy fails or gets user backlash (e.g., ads, premium tier)	Low revenue, app uninstalls	Validate pricing models via beta testing, gather user feedback	Medium
26	Non-compliance with GDPR or data privacy laws	Legal action, loss of user trust	Get legal review, add consent flows and data deletion options	High
27	Use of copyrighted assets (fonts, icons, music)	Legal takedowns or app store rejection	Use licensed assets only, track asset usage	Medium
28	Key team member leaves the project	Delays and knowledge gaps	Maintain documentation and code handover protocols	Medium
29	Missed deadlines due to underestimation	Late launch, budget issues	Use buffer time, Agile sprints, and realistic planning	Medium
30	Backend cost scaling with user growth	Budget overrun	Use scalable architecture and monitor cloud costs	Medium
31	Negative app reviews due to bugs or design flaws	Poor rating, less visibility in app stores	Have fast bug fix cycles and responsive support	Medium

32	Lack of user feedback loop	Features don't align with user needs	In-app surveys, reviews analysis, feature request form	Medium

Risk owner	Notes
Development Team	Prioritize critical crash fixes in updates.
QA & Development Team	Special focus on Android background restrictions.
QA Team	Include emulators and physical device testing.
Backend Developer	Add data integrity checks and retry mechanisms.
Product Team	
Development Team	Use local caching
Backend Developer	Regularly check security measures.
Development Team	Educate users about enabling notifications.
Project Manager	Prepare all required documentation and assets.
Product Team	Track analytics
QA Team	
Frontend Developer	Use layout widgets properly in Flutter.
Frontend Developer	Lazy load assets and components.
Backend Developer	Include unit tests for calculation modules.

Backend Developer	Audit API access regularly.
Fullstack Team	Use schema validation and client-side checks.
Frontend Developer	Include toast/snackbar explanations.
Fullstack Team	Regularly reconcile local vs. server state.
Backend Developer	Include automatic data validation, rollback mechanisms, and periodic backups.
QA Team	Integrate unit, integration, and end-to-end tests; track coverage metrics regularly.
DevOps	Use staging environments and automated smoke tests before going live.
UI/UX Designer	
Product Team	Offer customizable notification settings and adaptive scheduling based on user behavior.
Business Analyst / Marketing Team	Focus on clear value proposition and continuous user experience improvements.
Product Manager	Monitor feedback on monetization features and remain flexible with pricing models.
Legal Team / Backend Developer	Ensure data flows and user interactions are audited and legally reviewed pre-release.
Design Team	Maintain an asset license repository and audit new additions during design reviews.
Project Manager	Schedule regular knowledge-sharing sessions and maintain centralized documentation.
Project Manager	Use historical estimates and monitor velocity in sprints to recalibrate timelines.
Backend Developer / Finance Team	
Support Team / QA Team	

Product Team	Incorporate passive (analytics) and active (surveys) feedback in the feature roadmap.