

Унифицированный стандарт спецификации UI v1.0

Комплексная структура для кроссплатформенного проектирования цифровых продуктов

Резюме

Этот документ устанавливает строгий, машиночитаемый стандарт для UI/UX-спецификаций, который устраняет разрыв между дизайнерским замыслом и технической реализацией. Унифицированный стандарт спецификации UI (UUISS) предоставляет детерминированную структуру, обеспечивающую бесшовное сотрудничество между продуктовой, дизайнерской и инженерной командами при соблюдении международных стандартов взаимодействия человека и компьютера.

Стандарт решает критические задачи отрасли: двусмысленность при передаче дизайна в разработку, непоследовательное внедрение доступности, фрагментированное отслеживание аналитики и недостаточную документацию крайних случаев. Применение стандарта обеспечивает организациям измеримые улучшения в скорости разработки, согласованности дизайна и показателях качества продукта.

1. Основы и принципы

1.1 Теоретическое обоснование

Стандарт объединяет международные рамки для обеспечения научной достоверности и отраслевой актуальности. Основы включают ISO 9241-110:2020 (принципы взаимодействия, соответствие ожиданиям пользователей), ISO 9241-210:2019 (человеко-ориентированный процесс проектирования) и WCAG 2.2 Уровень AA (базовый уровень доступности).

Техническая реализация опирается на спецификации W3C DTCG для совместимости дизайн-систем и IFML от OMG для формального моделирования сложных пользовательских сценариев.

1.2 Архитектурная философия

Принят декларативный подход: поведение и состояния UI описываются метаданными, а не кодом. Определены слои: презентация (токены дизайна), композиция (структура), поведение (состояния и взаимодействия), данные (API-контракты и схемы).

Машиночитаемость — обязательное требование. Каждый элемент спецификации содержит метаданные трассировки, связывающие требования, тесты и KPI.

2. Структура спецификации

2.1 Метаданные артефакта

Каждая спецификация начинается с метаданных: идентификатор `UUISS-<Домен>-<Экран>-<Последовательность>`, семантическое версионирование (`MAJOR.MINOR.PATCH`), владельцы (`@product` , `@design` , `@dev`), статусы (`Draft` → `Ready` → `Approved` → `Implemented`), ссылки на внешние системы (Figma, Jira, PRD и др.).

Указываются поддерживаемые платформы с минимальными версиями и примечаниями для каждой.

2.2 Цели и метрики

Определяются ценность для пользователя и бизнес-цели с KPI. Метрики: времена отклика (p50, p90, p95, p99), процент успешных действий, частота ошибок, показатели вовлеченности.

Документируется контекст пути пользователя, входы и выходы экрана, а также связь с принципами ISO.

3. Интеграция дизайн-системы

3.1 Композиция и иерархия

Экран делится на регионы с компонентами. Для каждого региона: ID, обязательность, список компонентов с параметрами. Определяется порядок фокуса для доступности. Все элементы должны быть из утвержденной библиотеки.

3.2 Архитектура токенов дизайна

Токены определяются по W3C DTSG: цвета, размеры, типографика, тени, радиусы, отступы, иконки. Поддерживается двойной режим — встроенные и внешние токены.

4. Поведенческая спецификация

4.1 Управление состояниями

Состояния: default, hover, focus, pressed, disabled, loading, error, success, empty, invalid, masked. Переходы детерминированы. Поддерживаются варианты (светлая/темная тема и др.).

4.2 Шаблоны взаимодействия

Указываются события, условия, действия UI, цели навигации, параметры анимации, аналитические события. Поддерживаются сложные условия логикой И/ИЛИ. Навигация: `route:/path` , `modal:/component` , `https://...` .



5. Требования доступности

5.1 Размеры элементов

iOS: минимум 44pt, Android: 48dp, Web: 24 CSS px (рекомендовано 44).

5.2 Восприятие

Контрастность по WCAG (4.5:1, 7:1). Фокус видим. Поддержка ARIA. Live regions для динамических обновлений. Альтернативные способы ввода.



6. Интернационализация

6.1 Лингвистическая адаптация

ICU MessageFormat для множественного числа и рода. Именованние ключей и запас длины текста (120%, 200%). Поддержка RTL. Настройка тона и культурной уместности.

6.2 Локализация форматов

Числа, даты, время, валюты — по локали и ISO 4217.



7. Интеграция QA

7.1 Критерии приемки

BDD-сценарии на Gherkin. Каждый сценарий имеет ID, название, шаги Given-When-Then, ссылки на требования и тесты.

7.2 Управление крайними случаями

Описываются состояния: пустые, сетевые сбои, HTTP-ошибки, частичные данные, гонки, ошибки ввода. Для каждого — условие, поведение, стратегия коммуникации, механизмы восстановления.

8. Архитектура аналитики

8.1 Таксономия событий

События по схеме `object_action`. Свойства минимальны для приватности. Категоризация PII: анонимные, псевдонимные, чувствительные. Это влияет на хранение и доступ.

8.2 Метрики производительности

Клиент: FCP, LCP, TTI, CLS + бизнес-KPI. Сервер: задержки, ошибки, пропускная способность.

9. Техническая интеграция

9.1 Контракты API

Определяются методы, эндпоинты, схемы запросов/ответов, таймауты, политики повторов, кеш. Схемы версионизируются.

9.2 Адаптации под платформы

iOS — SwiftUI/UIKit, Android — Compose/View, Web — брейкпоинты и progressive enhancement.

10. Методология внедрения

10.1 Стратегия

Постепенное внедрение: новые фичи, пилотные команды, обучение. Интеграция инструментов для экспорта/импорта спецификаций. Проверка в CI.

10.2 Управление

Владение: продукт → требования, дизайнер → UX/UI, разработчик → техническая реализация. Изменения требуют анализа, версии, уведомления. Проводятся аудиты.

11. Заключение

UUISS — шаг вперед в коллаборации дизайна и разработки. Дает скорость (меньше двусмысленности), согласованность (система), качество (учет крайних случаев), доступность (встроенные требования).

Будущее: AR/VR, AI-персонализация, кросс-реальности. Архитектура расширяема и обратно совместима. Успех зависит от сообщества, экосистемы инструментов и обратной связи.

Приложения

- Приложение A: JSON Schema Reference
- Приложение B: Примеры спецификаций
- Приложение C: Матрица совместимости инструментов
- Приложение D: Руководство по миграции
- Приложение E: Чек-лист соответствия

Стандарт поддерживается мейнтейнерами Unified UI Specs и распространяется по лицензии MIT. Направляйте обратную связь или предлагайте изменения через pull request и issues в официальном репозитории.