

Дисциплина: Планирование разработки программного обеспечения
Проект: MindBalance (MVP)
Тип продукта: Веб-приложение для мониторинга психологического благополучия сотрудников
Размер команды: 4 человека
Дата подготовки: Декабрь 2025

СОДЕРЖАНИЕ

1. Введение
2. Часть 1: Оценка по трём точкам (PERT)
3. Часть 2: Оценка методом Use Cases (UCP)
4. Часть 3: Дополнительные методы оценки
5. Часть 4: Диаграмма Ганта и критический путь
6. Часть 5: Сетевой график
7. Выводы

ВВЕДЕНИЕ

Назначение документа:

Планирование проекта MindBalance, включающее оценку трудоёмкости несколькими методами, построение диаграммы Ганта с назначением ответственных, определение критического пути и анализ минимальной/максимальной длительности проекта.

Основные характеристики проекта:

- MVP-версия с ключевыми функциями для сотрудников и HR-менеджеров
- Архитектура: Backend + Mobile App + HR-панель + ML-компонент
- Команда: 4 специалиста одинаковой квалификации
 - Аналитик
 - Педант (разработчик)
 - Координатор (DevOps/архитектор)
 - Генератор идей (UX/UI специалист)

Единица измерения: человеко-день (8 часов работы одного специалиста)

ЧАСТЬ 1: ОЦЕНКА ПО ТРЁМ ТОЧКАМ (PERT)

1.1. Методология

Метод PERT (Program Evaluation and Review Technique) используется для оценки трудоёмкости проектов в условиях неопределённости.

Формула расчёта ожидаемой трудоёмкости:

$$E = \frac{O + 4M + P}{6}$$

Где:

- O – оптимистичная оценка (благоприятный сценарий)
- M – наиболее вероятная оценка
- P – пессимистичная оценка (неблагоприятный сценарий)
- E – ожидаемая трудоёмкость

Допущения:

- 4 человека-дня в день (работа всей команды одновременно)
- 1 месяц ≈ 21 рабочий день
- Равномерное распределение нагрузки внутри команды

1.2. Процесс сбора оценок

Требование выполнено: каждую работу независимо оценивали все четыре участника.

Пример работы 1 «Аналитика и требования»:

Участник	O	M	P
Аналитик	9	14	22
Педант	11	15	19
Координатор	10	13	18

Генератор идей	0	11	M	14	P	21
Среднее	10.25	14		20		

Итоговые усреднённые значения: O = 10, M = 14, P = 20

$\text{E}_1 = \frac{10 + 4 \times 14 + 20}{6} = \frac{10 + 56 + 20}{6} = \frac{86}{6} = 14.3 \text{ чел.-дн.}$

1.3. Таблица трёхточечной оценки (усреднённые значения)

Все оценки указаны в человеко-днях.

№	Работа	O	M	P	E (PERT)
1	Аналитика и требования	10	14	20	14.3
2	UX/UI прототипирование	8	12	18	12.3
3	Архитектура системы	8	12	18	12.3
4	Backend: авторизация	6	9	14	9.3
5	Backend: чек-инны	8	12	18	12.3
6	Backend: рекомендации	8	12	18	12.3
7	Backend: HR API	6	9	14	9.3
8	Backend: аналитика/логи	6	9	14	9.3
9	Мобильное: экран чек-иннов	8	12	18	12.3
10	Мобильное: упражнения	8	12	18	12.3
11	Мобильное: рекомендации	6	9	14	9.3
12	Мобильное: настройки	6	9	14	9.3
13	Мобильное: интеграция	5	8	12	8.2
14	HR-панель: дашборды	8	12	18	12.3
15	HR-панель: экспорт	6	9	14	9.3
16	ML: подготовка данных	8	12	18	12.3
17	ML: базовая модель	10	14	20	14.3
18	ML: интеграция в API	8	12	18	12.3
19	QA: интеграционные тесты	8	12	18	12.3
20	Деплой и релиз	6	9	14	9.3
ИТОГО		147	219	330	225.5

1.4. Расчёт общей трудоёмкости

$\text{E}_{\text{total}} = \frac{O_{\text{total}} + 4M_{\text{total}} + P_{\text{total}}}{6}$

$\text{E}_{\text{total}} = \frac{147 + 4 \times 219 + 330}{6} = \frac{147 + 876 + 330}{6} = \frac{1353}{6} = 225.5 \text{ чел.-дн.}$

1.5. Перевод в календарные дни и месяцы

Человеко-дни

Сценарий	Значение
Оптимистичный (O)	147 чел.-дн.
Наиболее вероятный (M)	219 чел.-дн.
Ожидаемый (E)	225.5 чел.-дн.
Пессимистичный (P)	330 чел.-дн.

Календарные рабочие дни

Формула: $D = \frac{\text{человеко-дни}}{4}$

Сценарий	Рабочие дни
Оптимистичный	$147 \div 4 = 36.8 \text{ дн.}$
Наиболее вероятный	$219 \div 4 = 54.8 \text{ дн.}$
Ожидаемый (PERT)	$225.5 \div 4 = 56.4 \text{ дн.}$
Пессимистичный	$330 \div 4 = 82.5 \text{ дн.}$

Рабочие месяцы (при 21 рабочем дне/месяц)

Сценарий	Месяцы
Оптимистичный	$36.8 \div 21 = 1.75 \text{ мес.}$
Наиболее вероятный	$54.8 \div 21 = 2.6 \text{ мес.}$
Ожидаемый (PERT)	$56.4 \div 21 = 2.7 \text{ мес.}$
Пессимистичный	$82.5 \div 21 = 3.9 \text{ мес.}$

Итог PERT-оценки

Ожидаемая трудоёмкость проекта:

- 56 рабочих дней ≈ 2.7 рабочего месяца

Коридор сроков:

- Минимум: 1.8 месяца
- Максимум: 3.9 месяца

ЧАСТЬ 2: ОЦЕНКА МЕТОДОМ USE CASES (UCP)

2.1. Диаграмма Use Cases MindBalance (MVP)

Акторы

Актор	Описание
Employee (Пользователь)	Основной пользователь: проходит чек-ины, получает рекомендации, выполняет микро-интервенции
HR Manager	Просматривает агрегированную статистику, скачивает отчёты, управляет параметрами аналитики
ML Analytics System	Внешний интеллектуальный компонент для расчёта рисков выгорания

OAuth / Auth Service	Внешний сервис идентификации (SSO, OAuth 2.0)
Актор	Описание

System Administrator	Управляет деплоем, мониторингом, конфигурацией системы
-----------------------------	--

Use Cases

Блок 1: Пользователь (Employee)

- UC1 – Пройти ежедневный чек-ин
- UC2 – Просмотреть рекомендации
- UC3 – Выполнить микро-интервенцию (упражнение, медитация)
- UC4 – Просмотреть историю состояний
- UC5 – Управлять настройками и профилем

Блок 2: HR-менеджер

- UC6 – Просмотреть дашборд статистики
- UC7 – Скачать отчёт (PDF/Excel)
- UC8 – Настроить параметры отчётности и фильтры

Блок 3: ML-модуль

- UC9 – Получить ML-оценку риска выгорания
- UC10 – Отправить данные для обучения модели

Блок 4: Системные Use Cases

- UC11 – Авторизация через OAuth
- UC12 – Управление пользователями и ролями
- UC13 – Мониторинг логов и состояния системы

2.2. Классификация акторов

Метод UCP задаёт веса для сложности акторов:

Тип актора	Описание	Вес
Simple	Другая система с простым интерфейсом	1
Average	Человек-пользователь или система с протоколом	2
Complex	Сложная система или требует специалистов	3

Классификация для MindBalance:

Актор	Тип	Вес	Обоснование
Employee	Complex	3	Сложный пользовательский интерфейс, требует проактивных уведомлений
HR Manager	Complex	3	Аналитический интерфейс, управление данными, экспорт
ML System	Average	2	Протокол REST API, стандартный формат данных
OAuth / Auth	Simple	1	Стандартный протокол, минимальная интеграция
System Admin	Average	2	Управление через CLI/админ-панель

$$\$\$UAW = 3 + 3 + 2 + 1 + 2 = 11\$\$$$

2.3. Классификация Use Cases

Критерий сложности на основе числа шагов:

Сложность	Описание	Вес
Simple	До 3 транзакций	5

Сложность	Описание транзакций	Вес
Complex	8+ транзакций	15

Use Case Points по методу:

№	Use Case	Тип	Вес	Обоснование
UC1	Чек-ин	Average	10	4-5 шагов: выбор параметра → ввод оценки → сохранение → отправка уведомления
UC2	Рекомендации	Simple	5	2-3 шага: запрос → выборка → отображение
UC3	Интервенции	Average	10	5 шагов: выбор → загрузка → запуск → отслеживание → логирование
UC4	История	Simple	5	3 шага: запрос → фильтрация → отображение
UC5	Настройки	Simple	5	3 шага: отображение → изменение → сохранение
UC6	HR-дашборд	Complex	15	8+ шагов: авторизация → подгрузка данных → агрегация → визуализация → кэширование
UC7	Экспорт	Simple	5	3 шага: выбор формата → генерация → скачивание
UC8	Фильтрация	Average	10	5 шагов: выбор критериев → валидация → пересчёт → кэширование → отображение
UC9	ML-оценка	Average	10	5 шагов: подготовка данных → вызов API → парсинг → логирование → отправка клиенту
UC10	ML-обучение	Complex	15	8+ шагов: сбор данных → подготовка → обучение → валидация → сохранение модели
UC11	Авторизация	Simple	5	3 шага: redirect → callback → создание сессии
UC12	Управление пользователями	Average	10	5 шагов: список → фильтрация → действие → валидация → логирование
UC13	Мониторинг	Average	10	5 шагов: сбор метрик → агрегация → проверка → алERTы → хранение
ИТОГО			125	

\$\$UUCW = 125\$\$

2.4. Technical Complexity Factor (TCF)

Рассчитывается по 13 техническим факторам (вес 0-5):

$$$$TCF = 0.6 + 0.01 \times \sum (w_i \times score_i)$$$$

Для MindBalance (типичный вес ~ 3-4):

- Распределённая архитектура
- ML-компонент
- Требования к безопасности и приватности
- Высокая доступность

$$$$\sum (w_i \times score_i) = 40$$$$

$$$$TCF = 0.6 + 0.01 \times 40 = 0.6 + 0.4 = 1.0$$$$

2.5. Environmental Complexity Factor (ECF)

Рассчитывается по 8 факторам окружения (вес 0-5):

$$$$ECF = 1.4 - 0.03 \times \sum score_i$$

Факторы:

- Опыт команды с аналогичными проектами: 3
- Знакомство с технологиями: 3
- Стабильность требований: 2
- Мотивация команды: 3
- Конкурентное давление: 2

- Сложность домена: 3
- Новизна архитектуры: 2
- Графики и сроки: 1

$\$ \$ \sum score_i = 19 \$ \$$

$\$ \$ ECF = 1.4 - 0.03 \times 19 = 1.4 - 0.57 = 0.83 \$ \$$

2.6. Расчёт Use Case Points

$\$ \$ UCP = (UAW + UUCW) \times TCF \times ECF \$ \$$

$\$ \$ UCP = (11 + 125) \times 1.0 \times 0.83 = 136 \times 0.83 = 112.88 \$ \$$

2.7. Перевод в человеко-часы и дни

Стандартное предположение: 1 UCP ≈ 20 человеко-часов (используется в отрасли)

$\$ \$ \text{Часы} = 112.88 \times 20 = 2257.6 \text{ ч.-часов} \$ \$$

$\$ \$ \text{Человеко-дни} = \frac{2257.6}{8} = 282.2 \text{ чел.-дн.} \$ \$$

2.8. Длительность проекта при команде из 4 человек

$\$ \$ D = \frac{282.2}{4} = 70.55 \text{ рабочих дней} \$ \$$

В месяцах (21 рабочий день):

$\$ \$ \frac{70.55}{21} = 3.36 \text{ месяца} \$ \$$

2.9. Итог по Use Case Points

По методу UCP длительность проекта MindBalance (MVP):

- 71 рабочий день ≈ 3.4 месяца

2.10. Сравнение PERT и UCP

Метод	Длительность	Человеко-дни	Месяцы
PERT	56.4 дн.	225.5	2.7
UCP	70.55 дн.	282.2	3.36
Разница	+25%	+25%	+24%

Анализ расхождений

Причины различия:

1. PERT недооценивает ML-компонент — метод PERT базируется на трёхточечных оценках отдельных работ, но не учитывает интеграционные сложности
2. UCP более консервативен — использует фактор сложности окружения ($ECF = 0.83$)
3. Компенсация факторов окружения — UCP учитывает опыт команды, стабильность требований, мотивацию

Вывод:

Реалистичная оценка лежит между двумя методами: 60–70 дней (2.9–3.3 месяца)

ЧАСТЬ 3: ДОПОЛНИТЕЛЬНЫЕ МЕТОДЫ ОЦЕНКИ

3.1. Functional Point Analysis (FPA)

Назначение: Оценка сложности информационных систем на основе их функциональности.

3.1.1. Определение функциональных элементов

External Inputs (EI) – Внешние входы (Average = 4 FP)

Пользовательские и системные входы данных:

- UC1: Ввод чек-ина (выбор эмоции, оценка энергии)
- UC5: Изменение настроек профиля
- UC3: Запуск интервенции
- UC8: HR-фильтры отчётности
- UC10: Передача данных в ML-модель
- UC12: Административный ввод (управление пользователями)

$\$ \$ EI = 6 \times 4 = 24 \text{ FP}$

External Outputs (EO) – Внешние выходы (Average = 5 FP)

Отчёты, рекомендации, экспорт данных:

- UC2: Рекомендации (персонализированные подсказки)
- UC3: Интервенции (отправка упражнений)
- UC7: HR-отчёты (PDF/Excel)
- UC6: HR-дашборды (визуализация KPI)
- UC4: История пользователя (временные графики)

$\$ \$ EO = 5 \times 5 = 25 \text{ FP}$

External Queries (EQ) – Внешние запросы (Low = 3 FP)

Запросы без сложной обработки:

- UC4: Запрос истории (фильтрация по датам)
- UC9: ML-прогноз (запрос текущего риска)
- UC2: Список интервенций

$\$ \$ EQ = 3 \times 3 = 9 \text{ FP}$

Internal Logical Files (ILF) – Внутренние логические файлы (Average = 10 FP)

Основные сущности БД:

- Пользователи (users, roles, permissions)
- Журнал чек-инов (check_ins)
- Настройки и профили (user_settings, preferences)
- История рекомендаций (recommendation_history)
- Агрегированная статистика (analytics_summary)

$\$ \$ ILF = 5 \times 10 = 50 \text{ FP}$

External Interface Files (EIF) – Внешние интерфейсные файлы (Low = 7 FP)

Интеграции с внешними системами:

- OAuth/Auth-система (стандартный протокол)
- ML-модель как внешняя подсистема (REST API)

$\$ \$ EIF = 2 \times 7 = 14 \text{ FP}$

3.1.2. Итоговое количество функциональных точек

$\$ \$ UFP = EI + EO + EQ + ILF + EIF$

$\$ \$ UFP = 24 + 25 + 9 + 50 + 14 = 122 \text{ FP}$

3.1.3. Value Adjustment Factor (VAF)

14 факторов, оценка 0–5 для каждого. Средняя оценка по MindBalance: 3

$\$ \$ VAF = 0.65 + 0.01 \times (14 \times 3) = 0.65 + 0.42 = 1.07$

Факторы:

- Требования к надёжности (высокие)
- Производительность (средние)
- Восстановляемость (высокие)
- Безопасность (высокие)
- Сложность функций (средняя)
- Повторное использование кода (среднее)
- Удобство установки (среднее)
- Удобство изменения (среднее)
- Удобство поддержки (среднее)
- Требования к концентрированному использованию (низкие)
- Требования к мобильности (высокие)
- Требования к интерфейсу (высокие)
- Требования к параллельной обработке (средние)
- Требования безопасности базы данных (средние)

3.1.4. Adjusted FP

$$\$ \$ AFP = UFP \times VAF = 122 \times 1.07 = 130.54 \text{FP} \$ \$$$

3.1.5. Перевод в человеко-ресурсы

Стандарт: 1 FP ≈ 10 человеко-часов

$$\$ \$ \text{Часы} = 130.54 \times 10 = 1305.4 \text{ч.- часов} \$ \$$$

$$\$ \$ \text{Человеко-дни} = \frac{1305.4}{8} = 163.2 \text{чел.-дн.} \$ \$$$

При команде из 4 человек:

$$\$ \$ D = \frac{163.2}{4} = 40.8 \text{ рабочих дней} \approx 1.94 \text{ месяца} \$ \$$$

Итог FPA

Длительность по FP Analysis: - 41 рабочий день ≈ 2.0 месяца

3.2. COCOMO II (Constructive Cost Model – Early Design)

Назначение: Оценка проектов на ранних стадиях проектирования в человеко-месяцах.

3.2.1. Размер проекта (Lines of Code)

Перевод FP → KLOC:

Используется стандартное соотношение: 1 FP ≈ 50 LOC

$$\$ \$ KLOC = AFP \times 0.05 = 130.54 \times 0.05 = 6.53 \$ \$$$

Интерпретация: Проект MindBalance (~6.5K строк кода) – это малый/средний проект.

3.2.2. Scale Drivers (5 факторов масштабирования)

Модель COCOMO II Early Design использует 5 масштабирующих факторов:

Фактор	Уровень	Коэффициент	Обоснование
Precededness (Прецеденты)	Nominal	3.72	Команда имеет опыт с веб-приложениями
Development Flexibility (Гибкость разработки)	Nominal	3.04	Гибкие требования, итеративный подход
Architecture / Risk Resolution	Low	4.24	Высокие риски ML-интеграции
Team Cohesion (Сплочённость команды)	Nominal	3.29	Опытная группа
Process Maturity (Зрелость процесса)	Low	4.24	Нет формальных процессов, первый MVP

$$\$ \$ E = 0.91 + 0.01 \sum w_i = 0.91 + 0.01 \times (3.72 + 3.04 + 4.24 + 3.29 + 4.24) \$ \$$$

$$\$ \$ E = 0.91 + 0.01 \times 18.53 = 0.91 + 0.1853 = 1.095 \$ \$$$

3.2.3. Effort Adjustment Factor (EAF)

Коэффициент корректировки усилий на основе 17 факторов (RCPX, RUSE, PDIF, PREX, PERS, FCIL, SCED).

Для MVP с ограничениями по времени и команде:

$$\$ \$ EAF = 1.25 \$ \$$$

(Выше стандартного 1.0 из-за сложности ML-компоненты и требований к производительности)

3.2.4. Итоговое вычисление COCOMO II

$$\$ \$ PPM = 2.94 \times (KLOC)^E \times EAF \$ \$$$

$$\$ \$ KLOC^E = 6.53^1 \times 1.095 \approx 7.3 \$ \$$$

$$\$ \$ PPM = 2.94 \times 7.3 \times 1.25 = 26.8 \text{чел.-месяцев} \$ \$$$

Перевод в рабочие дни:

$$\$ \$ \text{Дни} = 26.8 \times 21 = 562.8 \text{чел.-дней} \$ \$$$

При команде из 4 человек:

$$\$ \$ D = \frac{562.8}{4} = 140.7 \text{ рабочих дней} \approx 6.7 \text{ месяцев} \$ \$$$

Итог COCOMO II

Длительность по COCOMO II:

- 141 рабочий день ≈ 6.7 месяца

⚠ Примечание: Метод COCOMO II завышает оценку из-за учёта сложности распределённой архитектуры, требований к безопасности и наличия ML-компонентов. Может использоваться как оценка «с запасом».

3.3. Сравнение всех четырёх методов

Метод	Рабочие дни	Месяцы	Человеко-дни	Характеристика
PERT (3-точки)	56.4	2.7	225.5	Оптимистичный
Use Case Points	70.5	3.4	282.2	Сбалансированный
FPA Analysis	40.8	1.9	163.2	Недооценивает ML
COCOMO II	140.7	6.7	562.8	Консервативный
ДИАПАЗОН	41–141	1.9–6.7	163–563	Разброс 3.5x

Выводы по методам оценки

1. PERT и UCP согласуются:

- Оба дают оценку 56–71 дней (2.7–3.4 месяца)
- Различие ~25% объясняется учётом факторов окружения в UCP
- Рекомендуемый диапазон для планирования: 60–70 дней

2. FPA недооценивает:

- Функциональные точки не учитывают ML-сложность
- Для проектов с ML-компонентами использовать нельзя
- Результат (41 день) – нижняя граница

3. COCOMO II завышает:

- Модель консервативна для малых проектов
- Хорошо работает для масштабных систем (1000+ LOC)
- Использовать как верхнюю границу риска: до 6.7 месяцев

4. Применимость методов:

Метод	Применим для MindBalance?	Примечание
PERT	ДА	Основной метод, надёжный
UCP	ДА	Хороший для проверки, +25% резерва
FPA	УСЛОВНО	Только для backend, не для ML
COCOMO II	УСЛОВНО	Как верхняя граница риска

Итоговый вывод:

- Ожидаемая трудоёмкость: 56–71 рабочих дней (2.7–3.4 месяца)
- Реалистичный диапазон с буфером: 70–85 дней (3.3–4.0 месяца)
- Максимальная граница (с рисками): 140+ дней (6.7+ месяцев)

ЧАСТЬ 4: ДИАГРАММА ГАНТА И КРИТИЧЕСКИЙ ПУТЬ

4.1. Основные принципы планирования

Трудоёмкость vs. Длительность:

- Трудоёмкость (по PERT) = 56 дней при полной параллелизации
- Фактическая длительность = больше из-за зависимостей между работами

Структура команды (4 человека):

Роль	Основные ответственности	Компетенции
Аналитик (A)	Требования, Data Science, ML	Бизнес-анализ, SQL, Python
Педант (П)	Backend API, авторизация, QA	Backend, DevOps, тестирование
Координатор (К)	Архитектура, Mobile, интеграция	Mobile, системный дизайн
Генератор идей (Г)	UX/UI, Frontend, HR-панель	Design, Frontend, React

Логика Ганта:

1. Аналитика и архитектура – последовательно (зависимости)
2. Backend, Mobile, ML – параллельно (независимы)
3. HR-панель – параллельно с backend
4. QA и интеграция – после основной разработки
5. Деплой – финальная фаза

4.2. План работ с PERT-длительностями

№	Работа	Длительность (дн.)	Зависимости	Ответственный
1	Аналитика и требования	14	–	Аналитик
2	UX/UI прототипирование	12	1	Генератор идей
3	Архитектура системы	12	2	Координатор
4	Backend: авторизация	9	3	Педант
5	Backend: чек-инны	12	4	Педант
6	Backend: рекомендации	12	4	Аналитик
7	Backend: HR API	9	5	Координатор
8	Backend: аналитика/логи	9	5	Аналитик
9	Mobile: чек-инны	12	3	Генератор идей
10	Mobile: упражнения	12	3	Генератор идей
11	Mobile: рекомендации	9	6	Координатор
12	Mobile: настройки	9	6	Педант
13	Mobile: интеграция	8	11,12	Координатор
14	HR-панель: дашборды	12	3,7	Координатор
15	HR-панель: экспорт	9	8,14	Генератор идей
16	ML: подготовка данных	12	3	Аналитик
17	ML: базовая модель	14	16	Аналитик
18	ML: интеграция в API	12	17,5	Педант
19	QA: интеграционные тесты	12	13,15,18	Педант
20	Деплой и релиз	9	19	Координатор

№	Работа	Длительность (дн.)	Зависимости	Ответственный
---	--------	--------------------	-------------	---------------

4.3. Матрица распределения ресурсов по дням

Нагрузка на каждого человека (примерно равномерна):

Специалист	Фаза 1 (1-38)	Фаза 2 (39-74)	Фаза 3 (75-86)	Фаза 4 (87-95)	Всего
Аналитик (А)	14	38 (Рекомендации, ML)	—	—	52
Педант (П)	—	38 (Backend, Mobile, ML int.)	12	—	50
Координатор (К)	12	41 (Архитектура, Mobile, HR, Backend)	—	9	62
Генератор идей (Г)	12	32 (Mobile, HR-панель)	—	—	44

Общее: ~208 чел.-дней (близко к плану 225.5 из PERT)

4.4. Критический путь

Определение критического пути:

Последовательность работ, определяющих минимальное время завершения проекта. Любая задержка на критическом пути отодвигает конец проекта.

Расчёт критического пути

Прослеживаем цепочки зависимостей:

Путь 1 (Аналитика → UX → Архитектура → Backend → QA → Деплой):

1. Аналитика и требования (14) → конец день 14
2. UX/UI (12) → конец день 26
3. Архитектура (12) → конец день 38
4. Backend: авторизация (9) → конец день 47
5. Backend: чек-инны (12) → конец день 59
6. HR API (9) → конец день 68
7. QA: интеграционные (12) → конец день 86
8. Деплой (9) → конец день 95

Критический путь: $\text{CP} = 14 + 12 + 12 + 9 + 12 + 9 + 12 + 9 = 89$ дней

Задачи критического пути:

№	Задача	Длит.	Начало	Конец	Марж
1	Аналитика и требования	14	1	14	0
2	UX/UI	12	15	26	0
3	Архитектура	12	27	38	0
4	Backend: авторизация	9	39	47	0
5	Backend: чек-инны	12	48	59	0
6	Backend: HR API	9	60	68	0
7	QA: интеграционные	12	75	86	0
8	Деплой	9	87	95	0

Резерв (марж) = 0 — эти задачи не имеют временного буфера.

4.5. Минимальная и максимальная длительность

Минимальная длительность

Критический путь: 89 дней (4.2 месяца)

Это минимум при условии:

- Никаких задержек на критических задачах
- Полная параллелизация
- Никаких непредвиденных рисков

$\$D_{min} = 89 \text{ дней} \approx 4.2 \text{ месяца}$

Максимальная длительность

Возможные риски, добавляющие время:

Риск	Добавляется	Обоснование
Интеграция ML (перекол)	+10 дн.	ML-модель может потребовать доработки
Интеграция Mobile-Backend	+8 дн.	Несовместимость API
Безопасность/приватность	+7 дн.	Требования GDPR, penetration testing
Деплой и production issues	+5 дн.	Stability issues, incident response
Буфер на неопределённость	+10 дн.	Неизвестные неизвестности

$\$D_{max} = 89 + 10 + 8 + 7 + 5 + 10 = 129 \text{ дней} \approx 6.1 \text{ месяца}$

Вероятностная оценка

Сценарий	Дней	Месяцев	Вероятность	Пояснение
Оптимальный	89	4.2	20%	Идеальное исполнение
Номинальный	100	4.8	50%	Реалистичный сценарий
Пессимистичный	129	6.1	20%	С накоплением рисков

4.6. Обоснованные выводы

Длительность проекта

Трудоёмкость vs. Длительность:

- Трудоёмкость (PERT): 56 дней – общее количество работы, если вся команда работает одновременно
- Календарная длительность: 89–95 дней – фактическое время выполнения проекта

Причина различия: Зависимости между работами не позволяют полностью параллелизировать весь проект.

Критический путь

Критический путь определяется цепочкой:

$\$ \text{Требования} \rightarrow \text{UX/UI} \rightarrow \text{Архитектура} \rightarrow \text{Backend авторизация} \rightarrow \text{Backend чек-инны} \rightarrow \text{Backend HR API} \rightarrow \text{QA} \rightarrow \text{Деплой}$

Каждая задача на этом пути критична: задержка на 1 день отодвигает окончание на 1 день.

Параллельные пути с резервом

ML-компонент и Mobile имеют резерв ~20 дней. Это означает:

- Можно перераспределять ресурсы, если возникнут проблемы с backend
- ML-интеграция должна быть готова к дню 74 (но завершение проекта – день 95)

Оптимизация сроков

Возможные способы сокращения сроков:

Мера	Экономия	Сложность
Параллелизировать UX и архитектуру	5 дн.	Высокая (больше синхронизации)
Упростить HR API (минимальный функционал)	3 дн.	Средняя (меньше функций)
Предварительно подготовленные ML-данные	3 дн.	Низкая
Использовать готовые компоненты Mobile	7 дн.	Средняя (зависимости)
Максимум возможной экономии	~15 дн.	

Итоговый диапазон после оптимизации: 75–95 дней (3.6–4.5 месяца)

ЧАСТЬ 5: СЕТЕВОЙ ГРАФИК

5.1. Определение сетевого графика

Сетевой график – динамическая модель производственного процесса, которая:

- Отражает технологическую зависимость между работами
- Показывает последовательность выполнения комплекса работ
- Связывает их свершение во времени
- Учитывает затраты ресурсов
- Выделяет критические (узкие) места проекта

5.2. Правила построения сетевого графика

1. Одно исходное событие – начало проекта
2. Одно завершающее событие – конец проекта
3. Прямые стрелки – работы между событиями
4. Не более одной работы между двумя событиями – иначе вводят фиктивные работы
5. Нет замкнутых циклов – граф должен быть ациклическим
6. События нумеруются – левое событие < правое событие

5.3. Построение сетевого графика для MindBalance

Структура событий

События: 0 – Начало проекта 1 – Завершение аналитики 2 – Завершение UX/UI 3 – Завершение архитектуры 4 – Завершение backend авторизации 5 – Завершение backend основных модулей 6 – Завершение ML-интеграции 7 – Завершение mobile разработки 8 – Завершение HR-панели 9 – Завершение QA 10 – Конец проекта (деплой)

Представление в виде таблицы зависимостей

Работа	Предшественник	Начало события	Конец события	Длительность
Аналитика	Начало	0	1	14
UX/UI	Аналитика	1	2	12
Архитектура	UX/UI	2	3	12
Backend: авторизация	Архитектура	3	4	9
Backend: чек-инны	Авторизация	4	5	12
Backend: рекомендации	Авторизация	4	5	12

Backend: HR API Работа ML: подготовка	Чек-инны Предшественник Архитектура	5 Начало события 3	5a Конец события ба	9 Длительность 12
ML: модель	Подготовка	6a	6b	14
ML: интеграция	Модель	6b	6	12
Mobile: чек-инны	Архитектура	3	7a	12
Mobile: остальное	Чек-инны	7a	7	29
HR-панель	Архитектура	3	8	21
QA	Mobile, ML, HR	7,6,8	9	12
Деплой	QA	9	10	9

5.4. Расчёт ранних и поздних сроков

Ранний срок свершения события $t_p(i)$

Это самый ранний момент, к которому завершаются все работы, входящие в событие.

Прямой проход (forward pass):

События	t_p (ранний)	Обоснование
0 (Начало)	0	По определению
1	$0 + 14 = 14$	Аналитика завершена
2	$14 + 12 = 26$	UX/UI завершена
3	$26 + 12 = 38$	Архитектура завершена
4	$38 + 9 = 47$	Backend авторизация завершена
5	$47 + 12 = 59$	Backend чек-инны завершены (макс из двух путей)
5a	$59 + 9 = 68$	Backend HR API завершена
6a	$38 + 12 = 50$	ML подготовка завершена
6b	$50 + 14 = 64$	ML модель завершена
6	$64 + 12 = 76$	ML интеграция завершена
7a	$38 + 12 = 50$	Mobile чек-инны завершены
7	$50 + 29 = 79$	Mobile остальное завершено
8	$38 + 21 = 59$	HR-панель завершена
9	$\max(76, 79, 59) + 0 = 79$	QA начинается когда все три модуля готовы
10	$79 + 12 + 9 = 100$	Финальная дата проекта

Критический путь находится по событиям где $t_p = t_n$

Поздний срок свершения события $t_n(i)$

Это максимально допустимый момент свершения события без нарушения конечного срока.

Обратный проход (backward pass) с $t_n(10) = 100$:

События	t_p (поздний)	Резерв $R(i) = t_p - t_r$
10	100	0
9	$100 - 12 - 9 = 79$	0
7	$79 - 0 = 79$	0
8	$79 - 0 = 79$	20
6	$79 - 0 = 79$	3
6b	$79 - 12 = 67$	3
6a	$67 - 14 = 53$	3
7a	$79 - 29 = 50$	0
5a	$79 - 9 = 70$	2
5	$79 - 12 = 67$	8
4	$67 - 12 = 55$	8
3	$55 - 9 = 46$	8
2	$46 - 12 = 34$	8
1	$34 - 12 = 22$	8
0	0	0

5.5. Критический путь из сетевого графика

События критического пути (где $R(i) = 0$):

$\$ \$ 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 7a \rightarrow 7 \rightarrow 9 \rightarrow 10 \$ \$$

Соответствующие работы:

1. Аналитика (14)
2. UX/UI (12)
3. Архитектура (12)
4. Mobile: чек-инны (12)
5. Mobile: остальное (29)
6. QA (12)
7. Деплой (9)

Общая длительность критического пути:

$\$ \$ CP = 14 + 12 + 12 + 12 + 29 + 12 + 9 = 100 \text{ дней} \$ \$$

Примечание: В этой версии критический путь проходит через Mobile (т.к. Mobile: остальное = 29 дней). В исходном плане Mobile была разбита на отдельные компоненты. Используя исходную разбивку (Mobile: упражнения 12 + интеграция 8), критический путь сокращается до 89 дней.

5.6. Резервы и буфер проекта

Задачи с нулевым резервом (критические)

Эти задачи должны выполняться точно в график:

- Аналитика (резерв 0)
- UX/UI (резерв 0)
- Архитектура (резерв 0)
- Backend авторизация (резерв 0) – или Mobile чек-инны (резерв 0)

Задачи с положительным резервом

Эти задачи могут быть отодвинуты без влияния на финальный срок:

Задача	Резерв	Может быть отодвинута на
ML-интеграция	3 дн.	3 дня
HR-панель	20 дн.	3 недели
Backend HR API	2 дн.	2 дня

ВЫВОДЫ

Резюме оценок трудоёмкости

Метод	Трудоёмкость	Длительность	Применимость
PERT (3-точки)	225.5 чел.-дн.	56.4 дн. (2.7 мес.)	Основной метод
Use Case (UCP)	282.2 чел.-дн.	70.5 дн. (3.4 мес.)	Проверка, буфер
FP Analysis	163.2 чел.-дн.	40.8 дн. (1.9 мес.)	Недооценка для ML
COCOMO II	562.8 чел.-дн.	140.7 дн. (6.7 мес.)	Верхняя граница

Рекомендуемая оценка: 60–70 дней (2.9–3.3 месяца) ← пересечение PERT и UCP

Длительность проекта с учётом зависимостей

Аспект	Значение	Примечание
Минимальная	89 дн. (4.2 мес.)	Критический путь
Номинальная	95 дн. (4.5 мес.)	Базовый план Ганта
Реалистичная	100 дн. (4.8 мес.)	С буфером на риски
Максимальная	129 дн. (6.1 мес.)	Пессимистичный сценарий

Критический путь

Последовательность критических задач:

```
$$\text{Требования} \rightarrow \text{UX/UI} \rightarrow \text{Архитектура} \rightarrow \text{Backend авто} \rightarrow \text{Backend чек-инны} \rightarrow \text{Backend HR API} \rightarrow \text{QA} \rightarrow \text{Деплой}$$
```

Свойства:

- Любая задержка на критическом пути откладывает конец проекта
- ML-компонент и Mobile имеют резерв 15–20 дней
- HR-панель имеет резерв 20 дней

Рекомендации по управлению проектом

1. Мониторинг критического пути

Еженедельный контроль:

- Статус каждой критической задачи
- Ранние сигналы об отставании
- Готовность информации для следующей этапа

Критические вехи (gates):

- День 14: Завершение требований → готовность к UX/UI
- День 26: Завершение UX/UI → готовность к архитектуре
- День 38: Завершение архитектуры → старт backend/mobile/ml
- День 59: Завершение backend основной части → QA

2. Управление резервом

Параллельные пути с резервом:

- ML может быть отодвинуто на 3–15 дней без влияния на срок
- HR-панель может быть отодвинута на 20 дней
- Использовать этот резерв для компенсации задержек на критическом пути

3. Риск-менеджмент

Риск	Вероятность	Влияние	Смягчение
Задержка в ML	Средняя	Высокое	Параллелизировать подготовку данных
Проблемы интеграции	Средняя	Высокое	Раннее тестирование API
Изменение требований	Высокая	Среднее	Фиксация требований на день 14
Проблемы Mobile	Низкая	Среднее	Использовать готовые компоненты

4. Распределение ресурсов

Рекомендуемое распределение (4 человека):

Неделя 1–2 (Дни 1–14): Аналитик (100%) **Неделя 2–3 (Дни 15–26):** Генератор идей (UX/UI) (100%) **Неделя 3–6 (Дни 27–59):** Все специалисты (параллельная разработка) **Неделя 7–9 (Дни 60–86):** Педант (QA и интеграция), остальные – доработки **Неделя 9–10 (Дни 87–95):** Координатор (деплой), остальные – поддержка

Заключительные рекомендации

1. Трудоёмкость проекта: ~56–71 рабочих дня (консенсус PERT и UCP)
2. Календарная длительность: ~89–100 дней (4.2–4.8 месяцев) с учётом зависимостей
3. Критический путь: 89 дней – это абсолютный минимум
4. Буфер риска: +10–20 дней рекомендуется для непредвиденных ситуаций
5. Итоговый план: 95–110 дней (4.5–5.2 месяцев) для надёжного выполнения

Подготовлено: Команда Teamatix

Дата: Декабрь 2025

Статус: Готово к согласованию с заказчиком