**Software Development Plan (v 05.29.24)**

**1.** **Введение**

**1.1** **Назначение**

Цель данного документа состоит в описании плана разработки информационной системы "Платформа", а также в определении распределения ролей и обязанностей сотрудников.

**1.2 Область применения**

План разработки программного обеспечения (SDP) описывает этапы разработки, тестирования и развертывания информационной системы на оборудовании заказчика.

**1.3 Определения и аббревиатуры**

<https://docs.google.com/document/d/1Ou6epUB6-zqZc1wfUPYO2CbKB8MVWuJW/edit?usp=drive_link&ouid=103372480510558009952&rtpof=true&sd=true>

**1.4 Ссылки**

* Vision:

<https://docs.google.com/document/d/1E4bIXlkFc2ZZ4bEkHAf8uSEvKtMdK4Rw/edit?usp=drive_link&ouid=103372480510558009952&rtpof=true&sd=true>

* SRS: <https://docs.google.com/document/d/1FPEVQYix8eWcn4zHgGHRQIlBTKPg9gIR/edit?usp=drive_link&ouid=103372480510558009952&rtpof=true&sd=true>
* UC:

<https://docs.google.com/document/d/1WsK6MgHvOmjKYwUbCVXd2ZMJJeIUCtnk/edit?usp=drive_link&ouid=103372480510558009952&rtpof=true&sd=true>

* RL:

<https://docs.google.com/document/d/1wFBYTW8Hgl7Q1R_LKRvdZGVd30RoQ7UR/edit?usp=drive_link&ouid=103372480510558009952&rtpof=true&sd=true>

**1.5 Обзор документа**

1. Введение: смысл данного документа и полезные ссылки.
2. Обзор продукта: назначение проекта, цели его разработки и артефакты, которые будут получены.
3. Организация проекта: структура команды проекта.
4. Процесс управления: примерные сроки и стоимость разработки проекта.

**2.** **Обзор продукта**

**2.1** **Назначение, цели и контекст продукта**

Документ предоставляет заказчику описание информационной системы, включая её бизнес-позиционирование, описание целевой аудитории, функциональные возможности и требования, не углубляясь в технические аспекты.

**2.2** **Влияющие факторы и ограничения**

Основные риски описаны в отдельном документе

<https://docs.google.com/document/d/1wFBYTW8Hgl7Q1R_LKRvdZGVd30RoQ7UR/edit?usp=drive_link&ouid=103372480510558009952&rtpof=true&sd=true>

Другими факторы могут быть:

* Сжатые временные рамки разработки
* Ограниченные ресурсы
* Возможные изменения в требованиях заказчика
* Проблемы при развертывании приложения на хосте закачика
* Потенциальные проблемы с качеством исходного кода

**2.3** **Ожидаемые результаты проекта**

* jar серверное приложение
* sh скрипты сборки и деплоя
* exe файл десктопного приложения
* схемы и документы, необходимые для разработки (OpenAPI спецификация, UseCase диаграмма, макет интерфейса в Figma)

**2.4** **План развития данного документа**

Настоящий документ будет обновляться в случае изменения требований заказчика, а также в следующих случаях:

* Корректировка задач по реализации интерфейсов дизайна в соответствии с изменениями, внесёнными заказчиком
* Пересмотр приоритетов выполнения задач, связанный с изменением их очередности
* Введение новых функциональных требований
* Изменение сроков и этапов проекта
* Обновление технических спецификаций и стандартов
* Изменение состава проектной команды или перераспределение обязанностей

**3.** **Организация проекта**

**3.1** **Структура команды разработки**

* Инячина Диана: Project Manager, Developer, System Architect
* Семенов Александр: Developer
* Кузнецов Вадим: System Analyst, Tester, Developer

**3.2** **Внешние интерфейсы**

Клименков Cергей, Цопа Евгений, Николаев Владимир, Исаев Илья - совет директоров, доверенные лица стейкхолдеров, осуществляют приемочное тестирование

**3.3** **Роли и обязанности**

* Project Manager - планирование и координация проекта
* System Analyst - анализ требований стейкхолдеров, написание ТЗ
* System Architect - проектирование архитектуры ИС
* Developer - разработка серверной и клиентской части ИС
* Tester - тестирование разработанной ИС

**4.** **Процесс управления**

**4.1** **Оценка сроков разработки проекта**

11 месяцев = [февраль 2024; декабря 2024], возможна задержка, в таком случае срок окончания - конец января 2025.

**4.2** **План проекта**

*4.2.1* *План фаз*

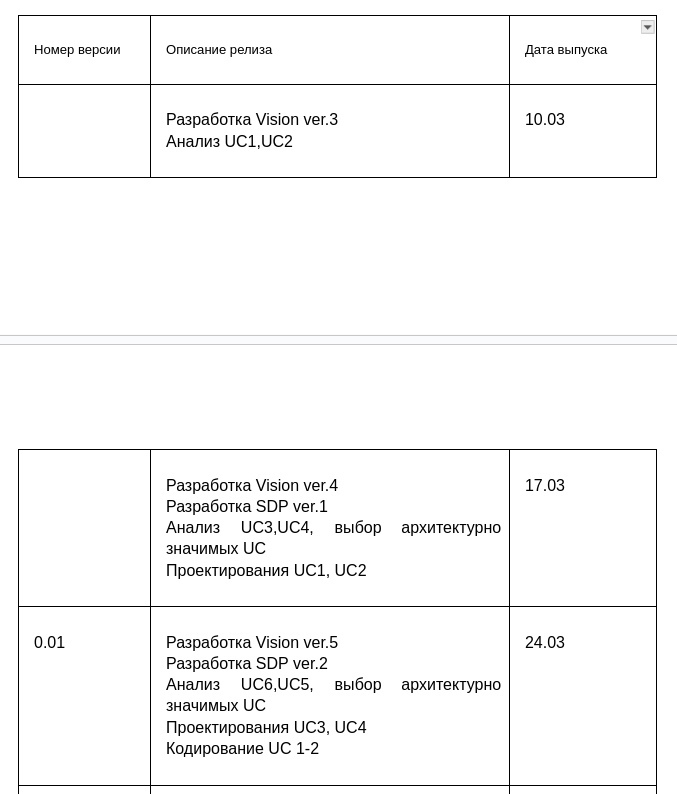
Разработка ведется по "классической" каскадной модели

| **Фаза** | **Количество итераций** | **Начало** | **Конец** |
| --- | --- | --- | --- |
| Определение требований | 1 | 6 февраля 2024 | 1 марта 2024 |
| Анализ требований | 1 | 2 марта 2024 | 31 мая 2024 |
| Проектирование | 2 | 1 июня 2024 | 20 сентября 2024 |
| Разработка | 2 | 1 июля 2024 | 1 ноября 2024 |
| Тестирование | 1 | 2 ноября 2024 | 1 декабря 2024 |
| Внедрение | 1 | 2 декабря 2024 | 29 декабря 2024 |

| **Фаза** | **Описание** | **Вехи** |
| --- | --- | --- |
| Определение требований | Определение того, что из себя представляет данная система, чек и как она ограничена | Vision |
| Анализ требований | Уточнение требований поведения ИС и создание плана жизненного цикла | SRS, UC, SDP, BC, Gloss, RL |
| Проектирование | Создание артефактов, необходимых для разработки на основе ТЗ | UX/UI design, API schema, DB schema |
| Разработка | Создание билда, содержащего всё определенную в ТЗ функциональность | Полнофункциональный билд проекта |
| Тестирование | Поиск и устранение неисправностей | Полнофункциональный и стабильный билд проекта |
| Внедрение | Развёртывание на оборудовании заказчика | Развернутое приложение, доступное пользователям |

*4.2.2* *Релизы*

*пример для правок*

**

| **Номер версии** | **Описание релиза** | **Дата выпуска** |
| --- | --- | --- |
|  | Разработка Vision 0.1  Анализ UseCase | 11 марта 2024 |
|  | Разработка Vision 0.2  Разработка SDP | 14 марта 2024 |
| 0.1-prealpha | Прототип системы на основе архитектурно-значимых UseCase | 19 июня 2024 |
| 0.4-alpha | Прототип системы на основе архитектурно-значимых UseCase (front, back, DB) (внутренний билд) | 1 октября 2024 |
| 0.9-beta | Полнофункциональный билд проекта | 12 ноября 2024 |
| 1.0-release | Полнофункциональный и стабильный билд проекта | 1 декабря 2024 |
| 1.1-lts | Полнофункциональный и стабильный билд проекта на оборудовании заказчика | 29 декабря 2024 |

*4.2.3* *Штатное расписание*

| **Роль** | **Ставка** |
| --- | --- |
| System Analyst | 625 |
| System Architect | 940 |
| Developer | 940 |
| Tester | 440 |
| Project Manager | 1250 |

*4.2.4* *Ресурсы проекта*

[*https://docs.google.com/spreadsheets/d/1iUdcPr\_2PLi3WvS7pNlAUiCz9PQC9xNirVKt4u9E3s0/edit?usp=sharing*](https://docs.google.com/spreadsheets/d/1iUdcPr_2PLi3WvS7pNlAUiCz9PQC9xNirVKt4u9E3s0/edit?usp=sharing)

4.2.5.1 План по сотрудникам

| **Роль** | **Навыки** | **Фазы** | **Cтавка** | **ЗП (руб)** |
| --- | --- | --- | --- | --- |
| System Analyst | Роль системного аналитика ведет и координирует выявление требований и моделирование вариантов использования, определяя функциональные возможности системы и разграничивая систему; например, установление того, какие действующие лица и варианты использования существуют и как они взаимодействуют. | * Определение требований * Анализ требований | 625 | 100 000 |
| System Architect | Роль архитектора программного обеспечения ведет и координирует техническую деятельность и артефакты на протяжении всего проекта. Архитектор программного обеспечения устанавливает общую структуру для каждого архитектурного представления: декомпозицию представления, группировку элементов и интерфейсы между этими основными группами. Также берет на себя роль конструктора базы данных, определяет таблицы, индексы, представления, ограничения, триггеры, хранимые процедуры, табличные пространства или параметры хранения, а также другие конструкции базы данных, необходимые для хранения, извлечения и удаления постоянных объектов. | * Проектирование * Разработка | 940 | 150 000 |
| Developer | Роль исполнителя отвечает за разработку и тестирование компонентов в соответствии с принятыми стандартами проекта для интеграции в более крупные подсистемы. Когда тестовые компоненты, такие как драйверы или заглушки, должны быть созданы для поддержки тестирования, разработчик также несет ответственность за разработку и тестирование тестовых компонентов и соответствующих подсистем. | * Разработка * Тестирование * Внедрение | 940 | 150 000 |
| Tester | Роль тестировщика отвечает за основные действия по тестированию, которые включают в себя проведение необходимых тестов и регистрацию результатов этого тестирования. Это охватывает:  - Определение наиболее подходящего подхода к реализации для данного теста  - Реализация индивидуальных тестов  - Настройка и выполнение тестов  - Регистрация результатов и проверка выполнения теста  - Анализ и восстановление после ошибок выполнения | * Тестирование * Внедрение | 440 | 70 000 |
| Project Manager | Роль менеджера проекта распределяет ресурсы, формирует приоритеты, координирует взаимодействие с клиентами и пользователями и, как правило, позволяет команде проекта сосредоточиться на правильной цели. Менеджер проекта также устанавливает набор методов, обеспечивающих целостность и качество артефактов проекта. | * Определение требований * Анализ требований * Проектирование * Разработка * Тестирование * Внедрение | 1250 | 200 000 |
| **Итого:** |  |  |  | 670 000 руб/месяц |

4.2.5.2 План поиска сотрудников

Сотрудники будут искаться среди тех, кто потерял работу из-за закрытия проекта, по причине академа или отчисления собригадников, и невозможности в одиночку продолжать проект. При расширении команды потребуется расширение бюджета.

*4.2.6* *Бюджет*

| **Сумма (руб)** | **Объект** | **Описание** |
| --- | --- | --- |
| 0 | Оборудование | Оборудование предоставляет заказчик |
| 670 000 | Зарплата | Расчёты в таблице |
| 0 | Аренда офиса | Коворкинг университета/онлайн |
| 670 000 |  | |

**4.3** **Мониторинг и контроль проекта**

Мониторинг и контроль проекта включают в себя планирование и реализацию мер для отслеживания хода выполнения проекта, контроля над соблюдением расписания, бюджета и требований, а также обеспечения качества результата.

*4.3.1* *План управления требованиями*

Цель: Обеспечить выполнение всех заявленных требований к продукту и вовремя выявлять и корректировать отклонения.

Процедуры:

1. Регулярные встречи с заказчиком:

* Формат: звонки или живые встречи
* Частота: 1 раз в 2 недели
* Участники: Project Manager, представители заказчика, ключевые члены команды разработки.
* Цель: сверка текущего статуса проекта с требованиями, обсуждение изменений и уточнений.

1. Документирование требований:

* Постоянный контроль за требованиями в Google Docs
* Постоянное обновление документации по требованиям на основе встреч с заказчиком

*4.3.2* *План управления расписанием*

Цель: Обеспечить выполнение проекта в установленные сроки.

Процедуры:

1. Определение рамок фаз:
   * Мягкие рамки: внутренние дедлайны для контроля за выполнением отдельных задач и фаз проекта.
   * Жёсткие рамки: сроки, определённые готовностью стейкхолдеров (преподавателями) принимать разработанную ИС.
2. Инструменты управления расписанием:
   * Notion: для создания и ведения календарей задач.
   * Telegram: для оперативного общения и уведомлений.
   * Google Docs: для совместной работы над документами и расписанием.
3. Мониторинг и корректировка расписания:
   * Еженедельные встречи команды для обсуждения прогресса и корректировки расписания.
   * Использование диаграмм Ганта для визуализации прогресса.

*4.3.3* *План управления бюджетом*

Цель: Обеспечить контроль и эффективное распределение бюджета проекта.

Процедуры:

1. Управление бюджетом:
   * Контроль бюджета осуществляется Project Manager.
   * Регулярное отслеживание расходов и сравнение с бюджетным планом.
2. Распределение зарплат:
   * Равное распределение зарплат между членами команды.
   * Обеспечение социальной справедливости и равенства.
3. Отчётность по бюджету:
   * Ежемесячные отчёты о расходах и бюджетировании.
   * Регулярное обсуждение финансового статуса на встречах команды.

*4.3.4* *План управления качеством*

Цель: Обеспечить высокое качество конечного продукта на всех этапах его разработки.

Процедуры:

1. Контроль качества:
   * Постоянная оценка артефактов-отчётов на каждой фазе жизненного цикла.
   * Тестирование GUI на прохождение UseCase невовлечёнными пользователями.
   * Ручное тестирование GUI и API на соответствие требованиям.
2. Инструменты тестирования:
   * Использование систем автоматического тестирования (например, Selenium для GUI, Postman для API).
   * Ведение журналов тестирования и исправления дефектов.
3. Оценка качества:
   * Проведение регулярных встреч для обсуждения результатов тестирования.
   * Корректировка процессов разработки на основе полученных данных.

*4.3.5* *План отчетности*

Цель: Обеспечить своевременное и полное информирование всех участников проекта о текущем состоянии и прогрессе.

Процедуры:

1. Формат отчетности:
   * Документы в формате docx, оформленные согласно шаблонам RUP.
   * Включение всех необходимых данных: статуса задач, прогресса по фазам, результатов тестирования, финансовых отчетов.
2. Периодичность отчетности:
   * Сокращённый спринт: отчёт каждая неделя.
   * Исключение: июль-август 2024.
3. Распространение отчетов:
   * Отправка отчетов всем заинтересованным сторонам (стейкхолдерам, членам команды) через электронную почту или системы управления проектом.
   * Архивирование отчетов для дальнейшего использования и анализа.

**4.4** **План управления рисками**

Смотрите RL: <https://docs.google.com/document/d/1wFBYTW8Hgl7Q1R_LKRvdZGVd30RoQ7UR/edit?usp=drive_link&ouid=103372480510558009952&rtpof=true&sd=true>

**4.5** **План завершения проекта**

На сервере Helios развёрнута разработанная ИС, реализована вся функциональность (можно выполнить каждый из UseCase-ов), отсутствуют критические баги, препятствующие этому. Команда приёмочного тестирования ставит "зачёт" продукту.