v02 20.11.2017

концепция

Устав онлайн

История изменений

| Дата | Версия | Описание | Автор |
| --- | --- | --- | --- |
| 20.11.2017 | 2.0 | Общее описание процесса | имТ |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Оглавление

[1 Введение 3](#_Toc499553352)

[1.1 Цель документа 3](#_Toc499553353)

[1.2 Определения и сокращения 3](#_Toc499553354)

[1.3 Связанные документы 3](#_Toc499553355)

[2 Описание проекта 4](#_Toc499553356)

[2.1 Общее описание проекта 4](#_Toc499553357)

[2.2 Цели проекта 4](#_Toc499553358)

[2.3 Ожидаемый результат 4](#_Toc499553359)

[3 Пользователи 4](#_Toc499553360)

[3.1.1 Администратор 4](#_Toc499553361)

[3.1.2 Уставщик 4](#_Toc499553362)

[3.1.3 Наборщик 4](#_Toc499553363)

[3.1.4 Гость 5](#_Toc499553364)

[4 Общее описание процесса 5](#_Toc499553365)

[4.1 Составление правил 6](#_Toc499553366)

# Введение

Документ подготовлен на этапе проведения предпроектных работ по формированию предложения и содержит описание решаемой в рамках проекта бизнес-задачи, а также концепцию архитектуры предлагаемого решения.

## Цель документа

Документ служит основанием для определения состава работ и первоначальной оценки проекта. Бизнес-требования к системе и общие принципы построения архитектуры решения, включенные в концепцию, должны быть согласованы с представителями рабочей группы заказчика.

На следующих этапах выполнения проектных работ концепция используется для проработки детального ТЗ и управления изменениями требований.

## Определения и сокращения

**Пользователь** – Пользователь системы.

**Роль** – Роль, назначаемая Пользователям.

**Администратор** – Роль, позволяющая редактировать Пользователей и утверждать или отклонять Заявки Пользователей.

**Заявка** – Заявленное действие Пользователей, которое требуется заверения Администратором

**Уставщик** – Роль, позволяющая создавать, редактировать и удалять Уставы

**Устав** – Корневая сущность, описывающая особенности богослужений для Шаблона Устава, храма или монастыря

**Шаблон Устава** – Предустановленный Устав, имеющий типовые характеристики богослужений

**Версия Устава** – Набор Правил для формирования последовательности богослужений. Относится к Уставу.

Может быть в трех состояниях:

* Черновик
* Опубликован
* Архивный

**Владелец Устава** – Пользователь, создавший Устав

**Редактор Устава** – Пользователь, получивший права на редактирование Устава

**Статус валидности Версии Устава** – Статус корректного оформления Правил Устава.

Может быть в состояниях:

* Не проверен
* В процессе проверки
* Не валиден
* Валиден

**Правило Устава** – Сущность, определяющая последовательность богослужения для дня

**Правило Минеи** – Сущность, определяющая последовательность богослужения для дня годового круга

**Правило Триоди** – Сущность, определяющая последовательность богослужения для дня пасхального круга

**Сообщение Устава –** Сообщение, относящееся к Уставу. Формируется фоновой службой при выполнении отложенных задач.

## Связанные документы

| Название | Ссылка на документ | Описание |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Описание проекта

## Общее описание проекта

Заказчик: мужской монастырь Николо-Берлюковская пустынь, Далее Монастырь.

Исполнитель: …

## Цели проекта

Целью проекта является разработка и внедрение системы, позволяющей формировать и просматривать последование богослужений на каждый день года, которые могут быть уникальными для каждого храма или монастыря РПЦ.

## Ожидаемый результат

Результатом работ является развернутое на площадке заказчика решение, функциональность которого соответствует требованиям, зафиксированным в проектной документации на этапе проектного обследования и формирования Технического задания.

# Пользователи

### Администратор

|  |  |
| --- | --- |
| Представители | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Описание | Администратор системы |
| Ответственность | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Цели | Управление пользователями системы, распределение ролей и доступа. Формирование общих шаблонов последовательностей богослужения. |
| Комментарий | - |

### Уставщик

|  |  |
| --- | --- |
| Представители | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Описание | Пользователь системы, имеющий доступ к формированию правил и последовательностей богослужения. |
| Ответственность | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Цели | Получение инструмента для формирований правил и последовательностей богослужения. |
| Комментарий | - |

### Наборщик

|  |  |
| --- | --- |
| Представители | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Описание | Пользователь системы, имеющий доступ к наполнению БД богослужебными текстами. |
| Ответственность | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Цели | Получение доступа к редактированию и добавлению богослужебных текстов. |
| Комментарий | - |

### Гость

|  |  |
| --- | --- |
| Представители | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Описание | Внешний пользователь системы, имеющий доступ к чтению последовательностей богослужений. |
| Ответственность | Представители группы пользователей не участвуют в этапе предпроектного обследования. |
| Цели | Получение доступа к последовательностям богослужений. |
| Комментарий | - |

# Общее описание процесса

Объектом автоматизации является Устав православного богослужения.

Правила соединения служб определены в книге Типикон.

Здесь декларируются Знаки служб – элементы системы градации торжественности служб.

Существует несколько книг, определяющих содержание богослужения. Каждая из них отвечает за свой временной период:

* Минея – книга, в которой описаны службы на каждый день года.
* Триодь – книга, в которой описаны службы Триодного цикла, зависящего от даты Пасхи (изменяющейся каждый год).
* Октоих – книга, в которой описаны службы недельного круга. Состоит из 8 частей (8 гласов), каждая из которых определяет службы на каждый день недели (начиная с воскресенья и заканчивая субботой). С Пасхи начинается 1-ый глас, в следующее воскресенье – 2-ой, и так далее. После 8-ого гласа снова начинается 1-ый.
* Часослов – книга, описывающая службы суточного круга.
* Псалтирь – книга, состоящая из 150 Псалмов. Каждый Псалом состоит из определенного количества стихов. Псалмы объединены в 20 Кафизм, которые используются в последовательностях богослужений. Некоторые из Псалмов используются в последовательностях богослужений отдельно.

Каждый храм (монастырь) имеет свои особенности богослужения, поэтому предполагается наличие множества Уставов для конкретных храмов или монастырей, которые должны основываться на одном из нескольких предустановленных Уставов (например: Полный, Монастырский, Приходской).

**Наборщик** вносит данные служб (содержание книг Минея, Триодь, Октоих, Часослов, Псалтирь). Эти данные являются общими для всех Уставов.

**Уставщик** формирует Правила соединения служб Устава, основываясь на одном из предопределенных Уставов, а также на собственных особенностях богослужения.

**Уставщик** должен определить следующие Правила:

* На каждый день года (ссылаясь на конкретную службу из Минеи);
* На дни Триодного цикла (ссылаясь на конкретную службу Триоди), наличие правил для всех дней Триодного цикла не является обязательным условием;
* На седмичные, субботние и воскресные дни (правила для использования Октоиха);
* Для каждого Знака служб.

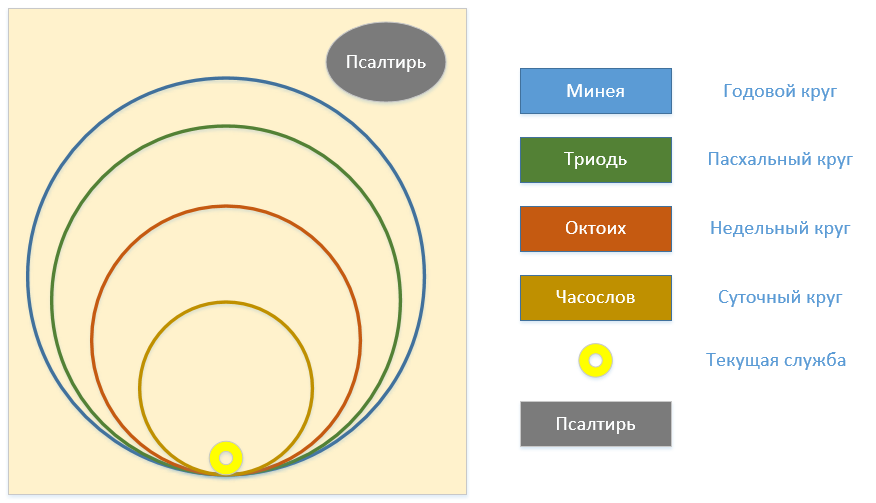
**Гость** имеет возможность просматривать конечный результат последовательностей богослужений конкретного Устава, набранных **Наборщиком** и определенных правилами **Уставщика**. Всё это он должен делать оффлайн.

## Составление правил

Для каждого календарного дня правила составления последовательностей богослужений складываются из нескольких факторов. Необходимо учитывать особенности:

1. Рядовой службы годового круга из Минеи;
2. Службы из Триоди, если календарный день попадает в Пасхальный круг;
3. Дня недели и гласа из Недельного круга Октоиха;
4. Конкретной службы Суточного круга, описание которой находится в Часослове.

Схематично можно изобразить вышеуказанные особенности в следующей диаграмме:



В качестве примера можно указать рядовую ситуацию: необходимо составить службу утрени 13 марта 2017 года.

1. 13 марта совершается память преподобного Василия исповедника (берем информацию из Минеи).
2. Это понедельник 3-ей седмицы Великого поста (берем информацию из Триоди).
3. Также это понедельник седмицы 5-го гласа (информация из Октоиха).
4. Служба утрени описывается в Часослове.
5. На утрени по понедельникам Великого поста читаются 4-ая, 5-ая и 6-ая кафизмы из Псалтири (эта информация определяется в Знаках служб).

# Архитектура решения

## Регистрация пользователя

1. Новый пользователь регистрируется через сайт typicon.online, вводя данные:

- логин

- пароль

- подтверждение пароля

1. Система регистрирует пользователя без назначения ему Роли.

## Подача заявки на назначение Ролей

**Доступ**

Доступно только для зарегистрированных пользователей.

**Алгоритм**

1. Зарегистрированный пользователь заходит на страницу Подачи заявок. Страница находится в настройках аккаунта Пользователя.

В таблице Пользователю отображаются все поданные им заявки со статусами «Создана», «Выполнена», «Отклонена».

1. Для пользователя без определения Роли возможна подача следующих заявок:
   1. Заявка на назначение Роли Уставщика. Доступно текстовое поле для комментария.
   2. Заявка на назначение Роли Редактора. Доступно текстовое поле для комментария.
2. Пользователь подает одну из заявок на назначение Роли. Заявка получает статус «Создана».

## Назначение Ролей пользователям

**Доступ**

Доступно для пользователей с ролью «Администратор».

**Алгоритм**

1. Администратор на странице необработанных заявок получает поданные от зарегистрированного Пользователя Заявки.
2. Выбирая Заявку в окне редактирования можно либо ее утвердить, либо отклонить с возможностью ввода комментария.
3. В случае утверждения Пользователю назначается заявленная им Роль. Заявка получает статус «Выполнена».
4. В случае отклонения Заявка получает статус «Отклонена».

## Просмотр списка Уставов

**Доступ**

Доступно для пользователей с ролью «Уставщик» и «Администратор».

**Алгоритм**

1. Выводится таблица со списком Уставов.
   1. Для Администратора – это все Уставы
   2. Для Уставщика – это Устава, где он Создатель или Редактор
2. Поля модели Устава:
   1. Id
   2. Name (FirstOrDefault)
   3. Status (WaitingApprovement | Draft | Published)

WaitingApprovement – создан, но нет ни одной Версии Устава

Draft – есть Версия Устава в статусе Draft

Published – Есть версия Устава в статусе Published

## Подача заявки на создание Устава

**Доступ**

Доступно для пользователей с ролью «Уставщик» и «Администратор».

**Алгоритм**

1. На странице подачи заявок Пользователь выбирает заявку «Создать Устав».
2. В странице создания Заявки Пользователь вносит данные об Уставе:

* Полное наименование (многоязычное)
* Краткое наименование (многоязычное)
* Координаты для отображения на карте (?)
* Изображение (аватар)
* …
* Шаблон Устава, на основании которого будет создаваться данный Устав

1. Сохраняя форму, создается Заявка на создание Устава со статусом «Создана». Также создается Сущность Устава (без версии Устава с вложенными Правилами) с указанными данными об Уставе.

## Создание Устава

**Доступ**

Доступно для пользователей с ролью «Администратор».

**Алгоритм**

1. Два способа инициации:
   1. Администратор на странице необработанных заявок получает поданные от Пользователем Заявки на создание Уставов.
   2. Администратор на странице просмотра списка Уставов выбирает действие «Утвердить» создание Устава на выбранном элементе.
2. Выбирая Заявку в окне редактирования можно либо ее утвердить, либо отклонить с возможностью ввода комментария.
3. В случае утверждения запускается в очередь Задача на клонирование Устава.

Клонируется опубликованная Версия Устава из указанного Шаблона Устава со всеми свойствами и вложенными коллекциями Правил и добавляется к коллекции Версий у Сущности Устава. Версии задается номер «1» и статус «Черновик».

1. По ее окончанию Заявка получает статус «Выполнена».
2. В случае отклонения Заявка получает статус «Отклонена».
3. Если Задача была инициирована не по Заявке, то Создателю Устава посылается сообщение об успешном создании Устава.

## Первоначальная настройка Устава

**Доступ**

Доступно для пользователей с ролью:

* «Администратор»
* «Уставщик» в правах Владельца Устава или Редактора.

Описание

Функционал для реализации в дальнейшем.

При создании

**Алгоритм**

1. После того

## Добавление задачи на публикацию Версии Устава

**Доступ**

Доступно для пользователей с ролью:

* «Администратор»
* «Уставщик» в правах Владельца Устава или Редактора.

**Алгоритм**

1. Пользователь на странице редактирования Сущности Устава выбирает действие «Опубликовать версию Устава».
2. Если Статус валидности Черновика Устава равен «В процессе», Пользователю возвращается сообщение   
   *«В настоящее время идет проверка валидности Черновика Устава. Повторите операцию позднее.»*
3. Если Статус валидности Черновика Устава равен «Не валиден», Пользователю возвращается сообщение   
   *«Черновик Устава имеет ошибки в заполнении. Исправьте их и повторите операцию.»*
4. Если Статус валидности Черновика Устава равен «Не проверен» или «Валиден», Задача на публикацию Устава отправляется на выполнение в фоновом режиме.

## Публикация Версии Устава

**Доступ**

Выполняется в фоновом режиме.

**Алгоритм**

1. Задача стартует.
2. Находит Версию Устава в статусе Черновик.
3. ~~Система проверяет Статус валидности Версии Устава.~~
4. ~~Если Статус валидности «Не проверен», создается задача на проверку валидности версии Устава.~~

~~Данная задача перезапускается.~~

1. ~~Если Статус валидности «В процессе», данная задача перезапускается.~~
2. ~~Если Статус валидности «Не валиден», задача завершается со статусом «Неудача».~~

~~В Устав отправляется сообщение о том, что Версия Устава не валидна.~~

~~Не отправляем, так как это уже должна сделать задача проверки валидности~~

1. ~~Если Статус валидности «Валиден», задача выполняет свой функционал.~~
2. Удаляется вложенная коллекция с годами переходящих праздников.
3. Указанная Версия Устава клонируется со всеми свойствами и вложенными коллекциями Правил и добавляется к коллекции Версий у Сущности Устава. Клонированной Версии Устава задается следующий номер после номера Указанной Версии Устава и статус «Черновик».
4. Указанной версии Устава задается статус «Опубликована».
5. Актуальной опубликованной Версии Устава задается статус «Архивный».
6. Удаляются все Выходные формы Устава.
7. В Устав отправляется сообщение об успешной публикации Версии Устава.
8. Задача завершается со статусом «Успешно».

## Проверка валидности Версии Устава

**Доступ**

Выполняется в фоновом режиме.

**Алгоритм**

1. Задача стартует.
2. Статусу валидности Версии Устава задается значение «В процессе».
3. Выполняется проверка валидации для всех вложенных элементов следующих коллекций:

* Правила Минеи
* Правила Триоди
* Общие правила
* Знаки служб
* Кафизмы
* Явные добавочные Правила.

1. В случае нахождения ошибок в указанных выше элементах, таковые добавляются в коллекцию ошибок с такими свойствами для каждого элемента:

* Наименование элемента
* Ссылка на ключ элемента
* Полное наименование ошибки

1. По окончанию проверки если ошибок не было найдено, Статус валидности Версии Устава задается в значение «Валиден».
2. Если ошибки были найдены, Статус валидности Версии Устава задается в значение «Не валиден», а коллекция ошибок в сериализованном виде задается в свойство Версии Устава «Сумма валидации» (ValidationSummary).
3. В Версию Устава добавляется сообщение об успешности/неуспешности проверки валидности.
4. Задача завершается.