

# Требования к проекту “Stolovka”

<b>1. Введение</b>	<b>2</b>
Цели	2
Соглашения о терминах	2
Предполагаемая аудитория	3
Масштаб проекта	3
Ссылки на источники	4
<b>2. Общее описание</b>	<b>5</b>
Видение продукта	5
Функциональность продукта	5
Классы и характеристики пользователей	5
Студент:	5
Работник столовой на выдаче:	6
Менеджер столовой:	6
Среда функционирования продукта	6
Рамки, ограничения, правила и стандарты	6
<b>3. Функциональность системы</b>	<b>7</b>
1. Приложение «Stolovka» для ВШЭ СПб	7
Описание и приоритет	7
Причинно-следственные связи, алгоритмы	7
Функциональные требования	7
2. Приложение для сотрудников.	9
Описание и приоритет	9
Причинно-следственные связи, алгоритмы	9
Функциональные требования	9
3. Сервер с базой данных.	10
Описание и приоритет	10
Причинно-следственные связи, алгоритмы	10
Функциональные требования	10
4. Набор средств разработки.	11
Описание и приоритет	11
Причинно-следственные связи, алгоритмы	11
Функциональные требования	11
<b>Нефункциональные требования</b>	<b>12</b>

Требования к производительности	12
Требования к сохранности (данных)	12
Критерии качества ПО	12
Требования к безопасности системы	12
<b>Прочие требования</b>	<b>13</b>

# 1. Введение

## 1.Цели

Данный документ подробно описывает внешние и внутренние требования к проекту по разработке и внедрению приложения «Stolovka» (далее «проект»), сценарии использования и ограничения, необходимые для полного документирования проектного решения для всех причастных к его осуществлению.

## 2.Соглашения о терминах

Приложение	Приложение «Stolovka» для платформ Android/iOS, реализующая указанный функционал, без привязки к конкретному ВУЗу.
Онлайн версия	Веб-версия приложения, по функционалу аналогичная вышеуказанным платформам.
Конкретная версия приложения	Отдельное приложение «Stolovka», реализованное для конкретного ВУЗа, либо для отдельного кампуса ВУЗа. Например: версия для столовой ВШЭ СПб на Кантемировской.
Программа сотрудников	Отдельная программа на Android с доступом к базе данных конкретной версии приложения и текущим заказам, которую используют сотрудники в столовой.
Меню	Обновляемый список блюд конкретной версии приложения с названиями, ценами, изображением и возможной дополнительной информацией, показываемый пользователю при открытии приложения.
База данных	Обновляемая база данных конкретного приложения, в которой хранится список блюд с вышеуказанными параметрами и другую информацию, хранимую на онлайн сервере с доступом для менеджера столовой, полное описание смотри в функциональных требованиях.
Пользователь	Человек, использующий конкретную версию приложения.
ВШЭ СПб	Главный кампус (Кантемировская 3) филиала Высшей Школы Экономики в Санкт-Петербурге

Менеджер столовой (менеджер)	Управляющий столовой (непосредственно в ней работающий), который следит за меню, организывает работу сотрудников в конкретный день. Могут быть разные люди (в зависимости от дня).
Внедрение	Изменение системы работы столовой под новое приложение, обучение её сотрудников, реклама и прочие мероприятия, благодаря которым готовое приложение смогут активно использовать пользователи.
Разработчики	Группа из трёх людей (Александр Гладков, Мария Тигина, Максим Зуев), которым принадлежит авторство идеи приложения и которые непосредственно занимаются её разработкой и администрированием.
Сервер	Онлайн машина с базой данных конкретной версии приложения.
ВУЗ	Высшее Учебное Заведение, университет.
Корзина	Временное хранилище выбранных товаров перед их заказом

### 3.Предполагаемая аудитория

Предполагаемая аудитория – студенты бакалавриата/магистратуры ВШЭ СПб, сотрудник столовой на выдаче и менеджер столовой.

### 4.Масштаб проекта

Конкретная версия приложения должна охватывать аудиторию столовой одного кампуса ВУЗа, для ВШЭ СПб это порядка 4000 студентов, и обслуживаться одним менеджером столовой.

В первую очередь предполагается разработка приложения под ВШЭ СПб и его внедрение при непосредственном участии разработчиков.

В дальнейшем предполагается интеграция приложения в неограниченное число российских ВУЗов при минимальной поддержке разработчиков (разработка конкретной версии, выдача рекомендаций по внедрению, установка сервера).

## **5.Ссылки на источники**

## **2. Общее описание**

### **1. Видение продукта**

Продукт является некоторым аналогом существующих онлайн сервисов по заказе еды, с узкой спецификацией – столовые ВУЗов.

### **2. Функциональность продукта**

Выбор студентом еды в меню приложения.

Заказ выбранной еды на определённую перемену в столовой.

Онлайн оплата или оплата на месте.

Получение индивидуального кода для выдачи.

Хранение обновляемой базы данных с блюдами.

Возможность сотруднику посмотреть текущие заказы и обновить их статус.

### **3. Классы и характеристики пользователей**

#### **Студент:**

Предполагается ориентированное на студентов с высокой лояльностью к современным технологиям, а именно активное использование смартфонов, их приложений, использование интернета и социальных сетей. Предполагается, что у студента есть опыт или представления об использовании приложений-сервисов для предоставления каких-либо реальных услуг (заказ еды/назначение встречи и т.д.).

### **Работник столовой на выдаче:**

Типичный сотрудник столовой - человек, возможно, не имеющий никаких или практически никаких представлений о работе приложений смартфонов, имеет минимальный опыт работы за компьютером. С неохотой готов к изменению рабочих функций.

### **Менеджер столовой:**

Человек с высокими коммуникационными и организационными навыками, имеющий достаточные представления о работе компьютеров и организации электронного обмена данных. Способен воспринять необходимость смены организации рабочего процесса и регулировать работу в новых условиях.

## **4.Среда функционирования продукта**

Конкретная версия приложения должна быть реализована для платформ Android и IOS, при успешном запуске может быть реализована веб версия.

База данных должна являться SQL.

На платформе PC для операционной системы Ubuntu должен быть реализован набор средств для разработки конкретных версий приложения, SQL базы данных для серверов.

Программа для сотрудника на выдаче должна работать на базе Android устройства(планшета) с доступом к сканеру.

Программа для менеджера на платформе PC.

## **5.Рамки, ограничения, правила и стандарты**

Система предполагает возможность оплаты с использованием онлайн

сервисов, поэтому приложения должно удовлетворять всем правилам оплаты в интернете. Также предполагается заключение контракта с банком для проведения операций онлайн.



## **3. Функциональность системы**

### **1. Приложение «Stolovka» для ВШЭ СПб**

#### **1. Описание и приоритет**

Android/IOS приложение для смартфонов, которое используют студенты для заказа еды. Высокий приоритет, так как без него проект не имеет смысла.

#### **2. Причинно-следственные связи, алгоритмы**

Приложение, работающее с SQL базой данных на сервере в онлайн режиме и предоставляющее его в GUI режиме студентам.

#### **3. Функциональные требования**

- Регистрация в системе по студенческому билету или по пропуску сотрудника.

Для этого должен быть осуществлён доступ к базе данных студентов/сотрудников ВУЗа для проверки корректности введенных данных. При регистрации пользователь указывает свой логин и пароль, по которым он в дальнейшем сможет входить в приложение, список логинов и хешей паролей должен храниться в базе данных сервера.

В случае неверно введенных данных пользователю должно быть отказано в регистрации с соответствующим сообщением.

- Вход в аккаунт.

Пользователь вводит свой логин и пароль, которые сверяются в хешем соответствующего пароля на сервере базы данных.

После успешного входа в аккаунт, пользователь будет автоматически входить в него при запуске приложения, должна быть возможность выйти из аккаунта и зайти в другой. При каждом входе должно проверяться, что численность студента/сотрудника в ВУЗе ещё актуально.

В случае неудачи, пользователь получает сообщение об ошибке и получает возможность ввести данные заново.

- Просмотр меню, доступного для заказа. Данные загружаются из базы данных на сервере.

В случае проблем с соединением (с сетью интернет или с базой данных) пользователю должно показываться сообщение об ошибке.

- Добавление блюд в корзину в приложении.

Данная информация хранится локально на устройстве пользователя.

- Заказ выбранных в корзине блюд, при этом указывается время, на которое выбирается заказ (перемена в ВУЗе или пара), доступный список времён загружается из базы данных, а также тип оплаты заказа: онлайн или при получении. В первом случае должна происходить онлайн оплата заказа. По информации из базы данных пользователю должна быть показана информации о загруженности столовой в данный момент с возможным предложением заказать еду на другое время, должно проверяться наличие блюда (не закончилось ли оно).

Успешный заказ отправляется на базу данных в список текущих заказов с кодом заказа – уникальным набором из цифр и букв, позволяющим идентифицировать (проверить) автора заказа. Этот же код показывается студенту как код для подтверждения заказа. Не должно быть возможности «угадать» код или ошибиться, тем самым получив чужое блюдо.

В случае любых проблем с соединением, отсутствия подтверждения

от базы данных в течение 5 секунд, заказ отменяется, пользователь получает сообщение об ошибке.

- Возможность отменить текущий заказ, перенести его на другое время. В данном случае соответствующим образом обновляется база данных.

Отмена должна происходить не ранее чем за одну пару до заказа. В случае любых проблем или отказа от отмены, пользователь получает предупреждение о невозможности отказа.

## **2. Приложение для сотрудников.**

### **1. Описание и приоритет**

Android приложение для устройств с крупным экраном, клавиатурой и мышью, которое используют сотрудники для работы с базой данной заказов. Высокий приоритет, так как без него проект не имеет смысла.

### **2. Причинно-следственные связи, алгоритмы**

Приложение, работающее с SQL базой данных на сервере в онлайн режиме и предоставляющее его в GUI режиме работникам столовой.

### **3. Функциональные требования**

- Регистрация аккаунта главного управляющего столовой, подтверждается разработчиками при выдаче приложения ВУЗу, который в дальнейшем самостоятельно регулирует доступ сотрудников к этому аккаунту.

Все проблемы с регистрацией аккаунтов решаются непосредственно через разработчиков.

- Регистрация и удаление в системе аккаунтов менеджеров столовой, подтверждается главным управляющим.
- Регистрация работников на выдаче, выполняется менеджером столовой.
- Обновление общей информации, такой как время перерывов в ВУЗе, выполняется главным управляющим, записывается в базу данных.
- Обновление текущего меню менеджером столовой. Менеджеру показывается графический интерфейс, который взаимодействует с базой данных сервера.
- Просмотр списка текущих заказов работником на выдаче и менеджером столовой. Данные загружаются из базы данных сервера. Данные сохраняются локально на устройстве. В случае разрыва соединения, сотрудник должен продолжать выдачу загруженных заказов. Клиенту должно быть отказано в новом заказе, если данные не могут быть переданы на программу сотрудника.
- Обновление статуса заказа – поступил на сервер, готов к выдаче, оплачен, выдан, отменён. Выполняется работником на выдаче.
- Идентификация заказа по коду. Должно быть проверено наличие данного кода в текущем списке заказов, при успешном подтверждении работник на выдаче выдаёт заказ пользователю.
- Обновление меню на текущий день. Для этого менеджеру столовой выдаётся список блюд, доступный в базе данных, она из них выбирает блюда и информацию об их количестве на текущий день.
- Обновление информации о блюдах в базе данных. Менеджер столовой может добавлять блюда, изменять их цены.

### **3. Сервер с базой данных.**

#### **1. Описание и приоритет**

SQL База данных, с которой взаимодействуют оба приложения. Высокий приоритет.

#### **2. Причинно-следственные связи, алгоритмы**

#### **3. Функциональные требования**

- Запись информации о: временах перерывов в ВУЗе, общем меню блюд, меню блюд на текущий день, текущий список заказов, логины и хеши паролей студентов, сотрудников столовой (разных уровней доступа).
- Возможность выполнять запросы к базе данных от обоих приложений в режиме онлайн.
- Сохранение истории всех заказов в течение как минимум последнего месяца.

### **4. Набор средств разработки.**

#### **1. Описание и приоритет**

Архитектуры приложения, позволяющая разработчикам быстро внедрять конкретную реализацию приложения под новый ВУЗ. Средний приоритет.

#### **2. Причинно-следственные связи, алгоритмы**

Должно планироваться при разработке архитектуры приложения, но конкретная реализация актуальная после первичного внедрения приложения на территории ВШЭ СПб и сбора статистики использования.

### 3. Функциональные требования

- Создание новой конкретной версии приложения.  
Разработчики должны иметь возможность внести минимальные изменения в дизайн, прописать иконки, стилизованные под новый вуз, а также прописать протоколы взаимодействия с базой данных сотрудников и сервером на базе вуза, регистрация управляющих аккаунтов. Все прочие детали backend не должны затрагиваться для успешной реализации нового приложения. Должна быть возможность зарегистрировать новый продукт в магазинах приложений app store, google play.
- Возможность получить схему базы данных и принцип построения сервера для установки всего этого на базу нового университета.

## **4. Нефункциональные требования**

### **1. Требования к производительности**

- Сервер должен обрабатывать объёмы данных порядка 100 блюд в меню, 10000 студентов, одновременные запросы от порядка 100 студентов и 3 менеджеров, 1 управляющего столовой.
- Сервер физически находится на территории ВУЗа или в пределах города, должен находиться в режиме онлайн весь рабочий день ВУЗа.
- Сервер должен храниться на одной машине средней производительности.
- Удобство интерфейса имеет наибольший приоритет, затем простота настройки, затем производительность.

### **2. Требования к сохранности (данных)**

- Все персональные данные должны храниться в строго зашифрованном виде и удовлетворять закону о хранении персональных данных.
- Должна храниться история заказов в указанном периоде, порядка 1 месяца.
- Информация о зарегистрированных пользователях и прочих данных должна храниться без ошибок записи, передаваться по сети интернет по защищённому протоколу, гарантирующему доставку данных без потерь/ошибок (TCP).

### **3. Критерии качества ПО**

- Удобство: при выборе 5 случайных студентов 4 из них, должны назвать приложение удобным.
- Остальные критерии качества описаны в разделе “Прочие требования”.

## **4. Требования к безопасности системы**

- Доступ к логинам студентов не должен иметь никто, кроме разработчиков, даже при наличии копии базы данных.



## 5. Прочие требования

- Разработка приложения для нового ВУЗа и сопутствующие мероприятия по внедрению должны занимать не более 8 часов работы трёх разработчиков -- не учитывая время на согласование дизайна и т.п.
- После внедрения приложения на базе нового ВУЗа, разработчики не должны принимать никакого участия в его дальнейшем сопровождении, помимо административных/профилактических/ремонтных работ по отдельному договору.
- Установить систему должен уметь человек, не являющийся разработчиком, имеющий соответствующее образование, при поддержке разработчиков по сети интернет (для установки в других городах). Должно проверяться соответствующими реальными тестами -- установка приложения при письменном и онлайн инструктаже разработчика.
- Настройка сервера должна занимать порядка двух часов работы специалиста.
- Приложение работника должно быть интуитивно понятно для человека, не знакомого с работой за компьютером. Обучение работе должно занимать не более получаса. Должно проверяться тестированием интерфейса на пользователях с характеристикой, схожей с указанной как “работник на выдаче”.
- Работник должен иметь возможность исправить допущенную ошибку, отменить недавно сделанное действие.