|  |  |
| --- | --- |
| Министерство образования Республики Беларусь | |
| Учреждение образования | |
| БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ | |
| ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ | |
|  | |
|  | |
| Факультет компьютерных систем и сетей | |
| Кафедра программного обеспечения информационных технологий | |
| Дисциплина: **Название дисциплины (АББРЕВИАТУРА)** | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
| **ОТЧЁТ** | |
| по лабораторной работе № **X** | |
|  | |
| Тема работы: **Название темы** | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
| Выполнил: | Фамилия И.О. |
|  | гр. **XXXXXX** |
|  | Вариант **X** |
|  |  |
| Проверил: | Фамилия И.О. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Минск **202X** | |

# Вопросы к заказчику

После ознакомления с перечнем требований был составлен следующий список вопросов:

1. Приложение может поддерживать лишь определённые версии операционных систем, так как каждая из них предлагает отличающийся набор функций (API), который требует написания разных версий программы для разных платформ. Разработка приложения под устаревшие версии нецелесообразна. Вопрос: какие версии каких операционных систем приложение должно поддерживать в первую очередь?
2. Какие именно языки должно поддерживать приложение?
3. Должно ли «Отчество» являться обязательным параметром?
4. Должно ли приложение поддерживать функциональность по вводу и сохранению полной информации и о жене, и об отце, и о детях?
5. Поддержка какого количества детей должна быть реализована в приложении?
6. Какой формат даты (американский, постсоветский и т.д.) должен быть реализован в приложении?
7. Какой должна быть вёрстка календаря?

# Ответы заказчика

1. Приложение должно поддерживать Windows 2000 – Windows 11.
2. Приложение должно поддерживать английский язык (обязательно), русский язык (опционально). Остальные языки допустимы.
3. «Отчество» должно являться обязательным параметром для русскоязычной версии программы.
4. Приложение поддерживать функциональность по вводу и сохранению полной информации и о жене, и об отце, и о детях.
5. Должна быть поддержка ста детей, поскольку встречаются феноменальные семьи.
6. Формат даты должен быть постсоветским (число, месяц, год).
7. Календарь должен быть ячеистым: несколько строк с числами.

# Учебный тест-план

## Суть и основные цели проекта

Создание и просмотр корректных баз данных о семьях в удобном виде.

## Требования, подвергаемые тестированию

Список критически важных аспектов (smoke test):

1. Открытие, закрытие приложения на OS Windows NT 5.Х, 6.Х, 10.Х.
2. Создание, сохранение и открытие базы данных.
3. Сохранение данных при сохранении-открытии базы данных.

Список второстепенно важных аспектов (critical path test):

1. Одновременный запуск двух и более инстанций приложения.
2. Открытие страницы общих данных при запуске.
3. Доступ к панели инструментов на всех страницах.
4. Выделение таблицы детей мышью.
5. Возможность сохранения неполной базы данных.
6. Сохранение базы данных в новых файлах.
7. Сохранение базы данных в виде «1 файл на 1 семью».
8. Поддержка нескольких языков (русский, английский).
9. Ввод текста в поля для ввода.
10. Выбор даты на календаре.
11. Добавление, удаление и редактирование детей.

Список желательно корректных аспектов (расширенное тестирование):

1. Отсутствие опечаток.
2. Единообразие оформления.
3. Применение в пользовательском интерфейсе общепринятых практик вёрстки.
4. Блокировка ввода букв в качестве роста и возраста.
5. Максимально возможное количество детей.
6. Загрузка баз данных из файлов с другим расширением.
7. Сохранение базы данных под именем уже существующего в этом месте файла.
8. Ввод и сохранение возраста с помощью калькулятора.

## Требования, не подвергаемые тестированию

В связи с уточнением требований были исключены следующие аспекты:

1. Работа на OS Windows 9x.
2. Работа на MS-DOS и других консольных OS.
3. Работа на других графических OS (кроме OS Windows NT 5.Х – 10.Х).

## Тестовая стратегия

После составления перечня тестов были составлены следующие методы их осуществления:

1. Smoke test: выполняется вручную тестировщиком посредством взаимодействия последнего с графическим интерфейсом пользователя;
2. Critical path test: выполняется вручную тестировщиком посредством взаимодействия последнего с графическим интерфейсом пользователя;
3. Расширенный тест: выполняется вручную тестировщиком посредством взаимодействия последнего с графическим интерфейсом пользователя.

В связи с отсутствием физического и юридического доступа к исходному коду рассматриваемого ПО будет применяться метод чёрного ящика.

Поскольку, если программное средство работает на более старой версии OS Windows, то оно гарантированно будет работать и на более новой версии (принцип обратной совместимости), то тестирование необходимо начинать с работы на самой старой из запланированных для тестов операционных систем.

## Критерии качества

После анализа тестовой стратегии были составлены следующие критерии качества и связанные с ними действия:

1. Качество считать приемлемым, если соблюдены следующие условия:

* 100% тестов рода smoke test пройдены успешно;
* 87.5% тестов рода critical path пройдены успешно;
* 100% критических ошибок исправлено.

1. Процесс нового тестирования начинать по окончании процесса сборки новой версии приложении.
2. Процесс тестирования продолжать по окончании исправления более 50% найденных при прошлом тестировании приложения ошибок.
3. Процесс тестирования завершать по окончании исправления более 90% найденных при прошлом тестировании приложения ошибок.

## Ресурсы

1. Программные: виртуальная машины для следующих операционных систем:

* OS Windows NT 5 (2000 или XP) 32-bit.
* OS Windows NT 6 (Vista, 7, 8 или 8.1) 64-bit;
* OS Windows NT 10 (10 или 11) 64-bit.

1. Аппаратные: одна стандартная рабочая станция – персональный компьютер с Windows 10/11, 8 GB RAM, i7 3GHz.
2. Человеческие: один студент-тестировщик БГУИР ФКСиС ПОИТ.
3. Временные: 16 рабочих дней (128 рабочих часов).
4. Финансы: согласно одобренному бюджету.

## Расписание

1. 20 февраля: планирование тестовых испытаний;
2. 28 февраля: анализ требований;
3. 9 марта: доработка требований;
4. 21 марта: формирование чек-листов;
5. 3 апреля: создание тест-кейсов;
6. 17 апреля: поиск и документирование дефектов;
7. 30 апреля: отчётность о результатах тестирования;
8. 7 мая: автоматизация тестирования.

## Роли и ответственность

Студент-тестировщик: планирование тестовых испытаний, анализ требований, доработка требований, формирование чек-листов, создание тест-кейсов, поиск и документирование дефектов, отчётность о результатах тестирования, автоматизация тестирования.

## Оценка рисков

1. Персонал (низкая вероятность): если студент-тестировщик недоступен, студент отчисляется, а тестирование осуществляется другим студентом-тестировщиком с таким же вариантом задания.
2. Время (низкая вероятность): заказчик указал крайние сроки выполнения этапов, желательно выполнять каждый этап на несколько дней раньше, чтобы иметь время на исправление непредвиденных ошибок.
3. Другие риски: другие конкретные риски отсутствуют.

## Документация

1. Требования. Ответственный – студент-тестировщик, дедлайн – 9 марта.
2. Тест-кейсы и отчёты. Ответственный – студент-тестировщик,   
   дедлайн – 17 апреля.
3. Итоговый отчёт о результатах. Ответственный – студент-тестировщик, дедлайн – 30 апреля.

## Метрики

Доля успешных тестов P – это количество успешно пройденных тестов U, разделённое на количество всех тестов V:

Минимально допустимые пороги успешных тестов:

1. P = 10% - начальная стадия проекта;
2. P = 50% - основная стадия проекта;
3. P = 90% - конечная стадия проекта.