1. **Перечень технологий:**
   1. Язык программирования: Python
   2. Среда разработки: PyCharm
   3. Сторонние библиотеки: PyQt, PyWinStyles, PyInstaller
   4. Система контроля версий: Git
   5. Сервис совместной разработки: GitHub
   6. Система автоматизации тестирования: PyTest
   7. ???
2. **Обоснование:**
   1. Язык программирования: Python - мультипарадигменный язык программирования, направленный на повышение производительности разработчика и переносимости написанных на нём программ, подходящий для любого направления разработки, активно развивающийся сообществом, а также хорошо интегрированный с системой распространения пользовательских библиотек
   2. Среда разработки: PyCharm - среда разработки, являющаяся частью экосистемы JetBrains, включающая в себя весь необходимый функционал для разработки на Python, начиная от интерпретатора и визуального отладчика и заканчивая интегрированным каталогом библиотек и широким выбором плагинов, позволяющих повысить эффективность и удобство разработки
   3. Библиотеки:
      1. PyQt - библиотека, позволяющая в Python использовать графический фреймворк Qt, поддерживающая практически весь его обширнейший функционал, в том числе удобное разделение логики и интерфейса, виджеты, стили и прочее. Помимо этого PyQt также включает в себя Qt Designer - дизайнер графического интерфейса, что в совокупности делает библиотеку удобной для быстрого прототипирования
      2. PyWinStyles - библиотека, позволяющая задать окну программы Windows визуальный стиль из предложенных на выбор ровно двумя строчками кода, при этом внося в дизайн приложения серьёзные изменения, делающие его привлекательнее и интереснее для конечного пользователя
      3. PyInstaller - библиотека, позволяющая собрать все файлы и зависимости, необходимые для работы приложения, в один запускаемый файл, удобный для распространения и запуска на других системах без необходимости дополнительной установки какого-либо программного обеспечения, одной командой
   4. Система контроля версий: Git - распределенная система контроля версий, основанная на двухступенчатом принципе внесения изменений, что удобно для удалённой разработки, быстра и надёжна, удобна для небольших файлов
   5. Сервис совместной разработки: GitHub - облачный сервис для хостинга и совместной разработки, поддерживающий систему контроля версий Git, не имеющий каких-либо весомых преимуществ по сравнению с аналогами в предполагаемой области разработки, *~~кроме того, что создан не украинцами, как GitLab~~*~~,~~ за исключением имеющегося опыта работы с этим сервисом
   6. Система автоматизации тестирования: PyTest - фреймворк Python, позволяющий писать простые и удобные для восприятия и запуска тесты, содержащий в себе полноценную систему тестирования логики приложения, а также имеющий большое сообщество и поддержку плагинов, в том числе для тестирования графических интерфейсов
3. **Декомпозиция:** [**[Вторая лабораторная]**](https://lucid.app/lucidchart/d2860b01-1e75-4576-acb5-3d8049187506/edit?viewport_loc=-512%2C44%2C3061%2C1689%2CHWEp-vi-RSFO&invitationId=inv_d3f46c5d-4f0d-4ea5-996e-bf64d2bad71c)
   1. Main:
      1. Запуск приложения
   2. Типа controller?:
   3. Типа view:
      1. Отрисовка пользовательского интерфейса
      2. Передача вводимых данных
   4. TournamentManager (Менеджер турниров) (оно и ниже – модель):
      1. Хранение списка объектов типа Tournament
      2. Манипуляция объектами типа Tournament: создание, изменение, удаление
   5. Tournament (Турнир):
      1. Установка, хранение и передача:
         1. организационных данных о турнире
         2. Участников турнира
         3. победителя турнира
         4. турнирной Сетки
   6. Bracket (Сетка):
      1. Установка, хранение и передача:
         1. типа турнирной сетки
         2. списка Матчей
      2. Установка результата Матча
   7. Match (Матч):
      1. Установка, хранение и передача:
         1. номера текущего раунда
         2. номер матча в раунде
         3. участников матча
         4. результата матча
   8. Participant (Участник):
      1. Установка, хранение и получение имени (названия) участника
   9. DatabaseManager (Менеджер базы данных):
      1. Взаимодействие с базой данных: запись и чтение Турниров

Создание репозитория:

