

Система за продажба на недвижими имоти



Изработили: Светослав Касабов 21621648

Джан Мехмедов 21621637

Специалност: СИТ Курс: III група: 4

Contents

[Техническо задание на проекта 3](#_Toc167197932)

[Бюджет 3](#_Toc167197933)

[Предмет на техническото задание 3](#_Toc167197934)

[Начален екран 3](#_Toc167197935)

[Екран за вход 4](#_Toc167197936)

[Екран за регистрация 5](#_Toc167197937)

[Основен екран за избор на прожекция на филм 6](#_Toc167197938)

[Екран за закупуване на билети 7](#_Toc167197939)

[Обхват на разработката 7](#_Toc167197940)

[Изисквания към програмния продукт 8](#_Toc167197941)

[Общи изисквания 8](#_Toc167197942)

[Обосновка на избора на система за контрол на версиите 8](#_Toc167197943)

[Причина 8](#_Toc167197944)

[Трудности при използването 8](#_Toc167197945)

[Настройки 9](#_Toc167197946)

[Съпоставка 9](#_Toc167197947)

[Обосновка на избора за система за управление на проекти: 9](#_Toc167197948)

[Причина за избора; 9](#_Toc167197949)

[Трудности при използването; 10](#_Toc167197950)

[Настройки, нужни за работата с него; 10](#_Toc167197951)

[мобилна/desktop/web версия; 10](#_Toc167197952)

[Използвани функционалности. 10](#_Toc167197953)

[Обосновка и разяснения към избраната методология за разработка на софтуера 10](#_Toc167197954)

[Предимства: 11](#_Toc167197955)

[Недостатъци: 11](#_Toc167197956)

[Описание на стъпките по реализацията: 12](#_Toc167197957)

[Екип 12](#_Toc167197958)

[Определяне на роли 12](#_Toc167197959)

[Разделяне на задачи 12](#_Toc167197960)

[Определяне на срокове 13](#_Toc167197961)

[Програмни средства за реализацията 13](#_Toc167197962)

[Стъпки, зависещи от конкретния подход за разработка 13](#_Toc167197963)

[Жалонни точки 13](#_Toc167197964)

[Timeline 14](#_Toc167197965)

[Microsoft Project 14](#_Toc167197966)

[Календар 14](#_Toc167197967)

[Диаграма на Гант 15](#_Toc167197968)

[Ресурсен лист 15](#_Toc167197969)

[Проследяване на завършеност на задачите 15](#_Toc167197970)

[Диаграма на свършената работа 15](#_Toc167197971)

[Диаграма на бюджет 15](#_Toc167197972)

[Таблица на бюджет 15](#_Toc167197973)

[Диаграма на ресурси 15](#_Toc167197974)

[Бизнес правила: 15](#_Toc167197975)

[Обекти и техните атрибути 16](#_Toc167197976)

[Изводи 16](#_Toc167197977)

# Техническо задание на проекта

## Бюджет

### Въведение

Създаването на ефективна система за продажба на недвижими имоти изисква добре планиран бюджет. Този бюджет обхваща всички аспекти на разработката, поддръжката, маркетинга, персонала и инфраструктурата, необходими за успешното функциониране на платформата.

#### 1. Разработка и Поддръжка на Софтуера

Разработка:

Брой разработчици: 3

Цена на час на разработчик: 15 USD

Часове за разработка: 225 часа (средна стойност)

Обща цена за разработка: 6,750 USD

Поддръжка:

Месечна цена за поддръжка: 500 USD

#### 2. Маркетинг и Реклама

Онлайн маркетинг:

Месечен бюджет: 1,000 USD

Офлайн маркетинг:

Месечен бюджет: 500 USD

#### 3. Персонал и Обучение

Брой служители:

Програмисти: 3

Маркетинг специалисти: 1

IT персонал: 1

Администратори: 1

Средна месечна заплата на служител:

1,200 USD

Обща месечна сума за заплати:

(3 програмиста + 1 маркетинг специалист + 1 IT персонал + 1 администратор) \* 1,200 USD = 7,200 USD

#### 4. Инфраструктура и Оборудване

Сървъри и хостинг: 5000USD за 2

Месечни разходи: 5000 USD (средна стойност)

Офис оборудване:

Еднократна сума: 6,000 USD

#### 5. Юридически и Административни Разходи

Месечни разходи:

200 USD

#### 6. Резервен Фонд

Месечна вноска в резервния фонд:

2,000 USD

### Общ бюджет за първия месец

| **Разходи** | **Сума (USD)** |
| --- | --- |
| **Разработка** | **6,750$** |
| **Поддръжка** | **500$** |
| **Онлайн маркетинг** | **1,000$** |
| **Офлайн маркетинг** | **500$** |
| **Заплати** | **7,200$** |
| **Сървъри и хостинг** | **5000$** |
| **Офис оборудване** | **6,000$** |
| **Юридически и административни разходи** | **200$** |
| **Резервен фонд** | **2,000$** |
| **Обща сума за първия месец** | **35,150$** |

A pie chart with different colored circles

Description automatically generated

### Общ месечен бюджет (след първия месец)

| **Разходи** | **Сума (USD)** |
| --- | --- |
| **Поддръжка** | **500$** |
| **Онлайн маркетинг** | **1,000$** |
| **Офлайн маркетинг** | **500$** |
| **Заплати** | **7,200$** |
| **Сървъри и хостинг** | **350$** |
| **Юридически и административни разходи** | **200$** |
| **Резервен фонд** | **2,000$** |
| **Обща месечна сума** | **11,750** |

A pie chart with numbers and a diagram

Description automatically generated

## Предмет на техническото задание

Нашата идея е да разработим иновативна система за онлайн продажба на недвижими имоти, която да предостави потребителите удобен и ефективен начин за търсене, разглеждане и закупуване на желана собственост. Системата ще съчетае в себе си функционалности за навигация, филтриране и визуализация на наличните имоти.

#### Уникални Предимства и Приноси:

1. **Разнообразие и Избор:** Потребителите ще имат възможност да избират от разнообразие от налични недвижими имоти, което да им осигури най-доброто съответствие на техните нужди и предпочитания.
2. **Визуализация и Виртуален Тур:** Възможността за визуализация на имотите и виртуален тур позволява на потребителите да изпитат уникалното усещане за имота преди да вземат решение за покупка.

## Изисквания към програмния продукт

### Общи изисквания

* Сигурност и защита на данните на потребителите, включително защита на личната информация и данни за плащане.
* Оптимизация за бърза и надеждна работа дори при големи натоварвания.
* Съвместимост с различни устройства и браузъри за максимално удобство на потребителите.
* Лесна поддръжка и актуализации на софтуера за осигуряване на непрекъсната работа и подобряване на функционалностите.

# Обосновка на избора на система за контрол на версиите

Основната причина за избора на GitHub за система за контрол на версиите е свързана със следните предимства:

1. Индивидуални Repository-та: GitHub позволява на всеки участник от екипа да притежава собствено repository, което дава възможност за паралелна работа върху един и същ проект. Това е отличен начин за съвместна работа, без конфликти.
2. Гъвкавост и Управление на Проекта: GitHub улеснява презаписването и актуализирането на файлове, както и връщането към предходни състояния на проекта, което е особено полезно, ако се откажем от дадена идея. Създаването на клонове позволява разработването на различни идеи, без да се възпрепятства главният проект.Трудности при използването;

#### Нашите Впечатления:

След работа с GitHub, ние бихме искали да споделим нашите впечатления и опит:

##### Положителни Аспекти:

1. Удобството от работата с индивидуални repository-та ни позволи да развиваме проекта ефективно и без конфликти.
2. Възможността за връщане към предходни версии на проекта се оказа полезна, особено при необходимостта за отмяна на определени промени.

##### Трудности:

1. Преодоляването на конфликтите и управлението на branch-овете изискваха време и упоритост.
2. Сливането на комити след големи промени представляваше предизвикателство, като понякога отнемаше значително време за идентифициране и отстраняване на възникналите проблеми.

#### Заключение:

Използването на GitHub се оказа ценен опит за нас. Въпреки предизвикателствата, които срещнахме по пътя, успяхме да се адаптираме и да извлечем ползи от функционалностите му. Нашата работа с GitHub ни научи на ценни умения за сътрудничество и управление на проекти, които ще ни бъдат от полза в бъдещите ни усилия.

# Обосновка на избора за система за управление на проекти:

### Основни Причини за Избора:

При нас Trello беше предпочитаната система за управление на проекти поради следните предимства:

1. Интуитивен и Визуален Подход: Trello предлага интуитивен интерфейс, който е лесен за разбиране и използване дори за потребители без технически познания. Визуалната представа на задачите в вид на карти позволява бързо и лесно проследяване на напредъка на проекта.
2. Гъвкавост и Адаптируемост: С Trello можем да създаваме различни дъски за различни проекти или за различни етапи от един проект. Това ни дава гъвкавостта да адаптираме системата към специфичните нужди и работни процеси на екипа.

### Впечатления:

След работа с Trello, ние забелязахме следните положителни и отрицателни страни:

#### Положителни Аспекти:

* + Лесно Управление на Задачите: Trello ни помогна да организираме и проследим задачите по проекта по ясен и структуриран начин. Възможността за създаване на списъци и карти ни позволи да разпределим задачите по категории и приоритети.
  + Добра Комуникация и Сътрудничество: Системата предлага възможност за коментиране и прикачване на файлове към всяка задача, което подобри комуникацията и сътрудничеството между членовете на екипа.

#### Изисквания и Трудности:

* + Ограничени Възможности за Сложни Проекти**:** За по-сложни проекти, в които се налага множество вътрешни зависимости и изисквания, Trello може да се окаже ограничен във функционалността си.

### Заключение:

Въпреки някои ограничения, Trello се оказа ценен инструмент за управление на нашите проекти. Неговата лесна за употреба и гъвкава система ни позволи да управляваме проектите си ефективно и да подобрим комуникацията в екипа. С опит и практика, ние успяхме да се адаптираме към неговите възможности и да извлечем максимална полза от него за нашите проекти.

# Обосновка и разяснения към избраната методология за разработка на софтуера

### Причина:

Избрахме Waterfall методологията, защото ни предоставя възможността да доставим издръжлив и завършен проект в кратък период от време. Ето някои от основните причини за избора:

1. Стабилни Изисквания: Waterfall методът ни позволява да започнем работа върху проекта с ясно дефинирани и стабилни изисквания. Това ни осигурява ясност и определеност относно целите на проекта още от началото.
2. Бърз Прогрес: Строгата последователност на Waterfall подхода ни позволява да постигнем бърз напредък. Всяка фаза на проекта се изпълнява последователно и веднъж завършена, което ускорява процеса на разработка.
3. Простота на Управлението: Waterfall методът предоставя ясна и предсказуема структура на проекта, което прави управлението му по-лесно. Това е особено важно при краткосрочни проекти, където е необходимо бързо и ефективно ръководство.
4. Издръжлив и Завършен Продукт: Строгата последователност на Waterfall методологията осигурява, че всеки етап на проекта е завършен и добре тестван преди преминаване към следващия. Това ни дава увереност, че ще доставим качествен и завършен продукт в кратки срокове.

В крайна сметка, избрахме Waterfall методологията, защото ни дава възможност да работим ефективно и да доставим качествен и завършен продукт в кратък период от време, което е от съществено значение за успеха на нашия проект.

|  |  |
| --- | --- |
| **Водопаден модел** | **Спираловиден модел** |
| Моделът е предпочитан от **клиентите** | Моделът е предпочитан от **разработчиците** |
| Грешки и рискове се идентифицират и отстраняват след приключване на съответният етап | Грешките и рисковете са идентифицирани и отстранени по-рано |
| Добре приложим при малки проекти | Добре приложим за големи проекти |
| Необходимо планиране в ранен етап | Не е необходимо планиране в ранен етап |
| Много малка гъвкавост | Гъвкавостта за промяна не е трудна |
| Сравнително евтин модел | Много скъп модел |
| Необходимо е задълбочено разбиране на изискванията от самото начало | Изискванията могат да се добавят в новите итерации, ако е необходимо |
| Произвежда единичен краен продукт в края на процеса | Генерира груб работещ продукт при всяка итерация |

# Описание на стъпките по реализацията:

## Екип:

### Определяне на полите:

##### Програмисти (3 човека):

## Създаване на програмен код за софтуерния продукт.

## Разработване на софтуерната архитектура.

## Тестване на софтуерния код и отстраняване на бъгове.

##### Маркетинг специалисти (1 човек):

## Планиране на маркетингови кампании за продукта.

## Изграждане на маркетингова стратегия.

## Промотиране на продукта чрез рекламни канали.

##### IT персонал (1 човек):

## Инсталиране и конфигуриране на необходимите софтуерни и хардуерни компоненти.

## Поддръжка и управление на ИТ инфраструктурата.

## Решаване на технически проблеми и възстановяване на системата при необходимост.

##### Администратори (1 човек):

## Управление на системите и сигурността на данните.

## Конфигуриране на сървъри и мрежови устройства.

## Мониторинг на работата на системата и реагиране при необходимост.

## Определяне на срокове:

1. **Проектиране:**
   * Продължителност: 2 дни (12 работни часа)
2. **Кодиране:**
   * Продължителност: 14 дни (84 работни часа)
3. **Документиране:**
   * Продължителност: 3 дни (18 работни часа)
4. **Интегриране:**
   * Продължителност: 3 дни (18 работни часа)
5. **Тестване:**
   * Продължителност: 3 дни (18 работни часа)

## Програмни средства за реализацията

1. **Интегрирани разработвателски среди (IDE):**
   * Visual Studio Code - мощен и лек редактор, подходящ за разработка на уебсайтове и уеб приложения.
   * IntelliJ IDEA - изключително интуитивна и пълнофункционална IDE, подходяща за разработка на Java и Kotlin приложения.
   * PyCharm - специализирана IDE за Python разработка, която предоставя богати възможности за управление на проекти и код.
2. **Системи за управление на версиите (VCS):**
   * Git с хостинг на GitHub - позволява управление на изходния код на проекта, проследяване на промените и сътрудничество между членовете на екипа.
3. **Системи за управление на проекти (Project Management Tools):**
   * Trello - удобен за организацията на задачите и проследяване на напредъка им чрез дъски и карти.
4. **Средства за комуникация и сътрудничество:**
   * Slack - платформа за комуникация, която предоставя възможности за обмен на съобщения, файлове и интеграция с други инструменти за разработка.
   * Zoom - платформа за видеоконференции, подходяща за онлайн срещи и обучения на разстояние.
5. **Инструменти за автоматизация на процесите:**
   * Jenkins - инструмент за автоматизиране на непрекъснатата интеграция на софтуера.
   * CircleCI - уеб базирана платформа за автоматизиране на тестовете и доставката на софтуера.

## Стъпки, зависещи от конкретния подход за разработка

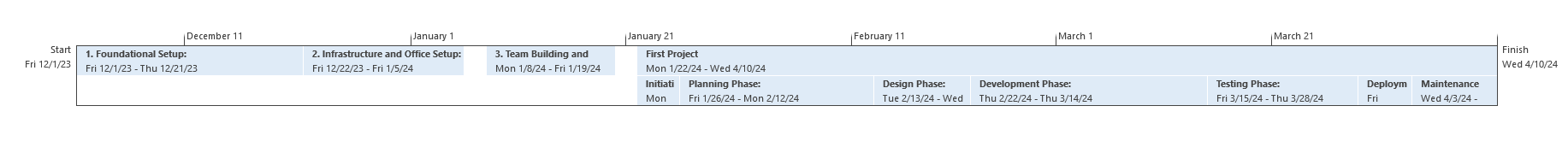
1. **Анализ на изискванията:**
   * Идентифициране на основните функционалности и изисквания на системата за продажба на недвижими имоти.
   * Провеждане на срещи с клиента за обсъждане на неговите нужди и изисквания.
   * Съставяне на документ с изисквания (SRS - Software Requirements Specification), който описва детайлно функционалностите на системата.
2. **Проектиране:**
   * Дизайн на архитектурата на системата, включително базата данни, потребителския интерфейс и връзките с външни системи.
   * Изготвяне на диаграми на потоците на данни, диаграми на класовете и други диаграми за ясно определяне на структурата и взаимодействието на компонентите на системата.
3. **Разработка:**
   * Програмиране на основните функционалности на системата, включително модулите за търсене на имоти, управление на потребителите, обработка на плащания и други.
   * Използване на подходящи програмни езици и технологии за реализация на функционалностите.
4. **Тестване:**
   * Изпълнение на функционални и нефункционални тестове за проверка на работата на системата.
   * Провеждане на тестове за интеграция и системно тестване за проверка на взаимодействието между различните компоненти на системата.
5. **Внедряване и поддръжка:**
   * Инсталиране и конфигуриране на системата на живо.
   * Обучение на персонала за използване на системата.
   * Поддръжка и подобрения на системата след внедряването й.

## Жалонни точки

A screenshot of a computer

Description automatically generated

# Microsoft Project



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

## Календар

A screenshot of a computer

Description automatically generated

## Диаграма на Гант

A screenshot of a computer

Description automatically generated

A computer screen shot of a diagram

Description automatically generated

## Ресурсен лист

A graph and chart with text

Description automatically generated with medium confidence

A screenshot of a spreadsheet

Description automatically generated

## Диаграма на свършената работа

A diagram of a project

Description automatically generated