БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра информатики

Факультет ИНО

Специальность ИиТП

Контрольная работа № 1

по дисциплине «Управление разработкой программного обеспечения»

Выполнил студент: Дегтярев А.А.

группа 393551

Зачетная книжка № 902021-26

Минск 2018

#### Практическая часть: разработка Концепции программного продукта

**Концепция программного продукта**

**Microed –** microlearning**.**

**Microed** - сервис, доступный из web и с мобильных устройств, предоставляет маленькие курсы длинной около 5 минут, которые можно быстро пройти, изучив небольшую тему. Курс представлен удобным интерфейсом в виде карточек (Tinder, Reigns), карточки могут содержать как текст, так и какой-то интерактивный контент(тест, опросник, голосование и т.д). Сервис имеет большой каталог курсов, охватывающих разные сферы: инженерию, социологию, медицину, биологию, математику, менеджмент и т.д, вплоть до пошаговых рецептов каких-то блюд;

**Microed** также представляет web-based инструмент для создания коротких курсов/карточек, который может быть доступен сторонним авторам; Позволяет создавать отдельные брендированные «каналы» с курсами. Например, курсы по AdWords/AdMob/Analytics/Маркетингу от Google, рекламирующий, естественно, в первую очередь, продукты компании.

**Теоретическая часть (вопросы)**

*8. Понятие мотивации и вознаграждения. Иерархия потребностей по А.Маслоу. Мотивация труда программистов и их менеджеров.*

*Основные теоретические положения:* понятие потребности; признаки классификации потребностей; пирамида потребностей по А. Маслоу; закон возвышения потребностей.

**Потребность** - по-разному трактуемое понятие:

|  |  |  |
| --- | --- | --- |
| Нужда | Состояние индивида при нужде в условиях жизни, предметах и объектах, без которых невозможно его существование и развитие. | С. Л. Рубинштейн |
| отношение | Потребность – это система отношений между субъектом и средой обитания | Д. А. Леонтьев |
| отклонение от уровня адаптации | Потребность есть результат отклонения внешней или внутренней реальности от сложившихся ожиданий субъекта по поводу этой реальности | Д. К. Мак-Клелланд |
| [состояние](https://ru.wikipedia.org/wiki/%D0%9F%D1%81%D0%B8%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D0%B5) | Под потребностью понимается динамическое состояние повышенного напряжения, которое «толкает» человека к определённым действиям. Это напряжение «разряжается» при удовлетворении потребности. Таким образом, в процессе возникновения и удовлетворения потребностей человек проходит через ряд динамических состояний, отличающихся уровнем своей напряжённости. | Курт Левин |
| программа поведения | Потребности - это основные программы поведения, посредством которых реализуется функционирование (жизнедеятельность) субъекта. | Б. И. Додонов |
| психопатия | Потребность есть вынужденное субъективное страдание психики, являющееся основной причиной всех неврозов. | В. В. Монастырский |

Потребности проявляются в виде эмоционально окрашенных желаний и стремлений, а их удовлетворение – в виде оценочных эмоций. По мере удовлетворения одних – у человека возникают новые потребности, что позволяет экономистам утверждать, что в общем потребности безграничны. Они определяют избирательность восприятия мира человеком, фиксируя его внимание преимущественно на объектах, которые могут удовлетворить его потребности;

Средствами удовлетворения человеческих потребностей называются *блага*, а степень удовлетворенности человека – *благосостояние.* Совокупность действий, направленных на оптимальное удовлетворение потребностей человека составляет жизнеобеспечение. Те, что относятся к потребностям в пище, одежде, жилье, здоровье – называется *быт*. Сокращение или лишение возможности удовлетворения основных потребностей называется *депривация*.

Потребности человека представляют собой самую неоднозначную категорию и обуславливаются, помимо общей с животными первой сигнальной системы, наличием сложной психической организации, представленной второй сигнальной системой (мышлением и речью)

Так как процесс удовлетворения потребностей выступает как целенаправленная деятельность – потребности являются источником активности личности.

Существует множество классификаций потребностей:

* по сферам деятельности:
  + биологические
  + потребности труда
  + познания
  + общения
  + отдыха
* по объекту потребностей:
  + материальные (жилище, одежда, орудия производства)
  + биологические (пища, воздух, вода, климатические условия)
  + социальные (общение, общественное признание)
  + духовные (знания, творческая деятельность)
  + этические
  + эстетические
  + другие
* по временной устойчивости:
  + устойчивые (жесткие, неэластичные)
  + ситуативные (эластичные)
* по функциональной роли:
  + доминирующие/второстепенные
  + центральные/периферические
  + устойчивые/ситуативные
* по субъекту потребностей:
  + групповые
  + индивидуальные
  + коллективные
  + общественные
* по масштабу и структуре:
  + Абсолютные(максимальный объем благ, который мог быть потреблен если бы был в наличии)
  + Перспективные (которые могут быть удовлетворены в будущем)
  + Действительные (настоящие, возможные при данном уровне развития производства)
  + Подлежащие удовлетворению
  + Фактически удовлетворяемые
* По участию в производственном процессе
  + Производственные
  + Непроизводственные
* По происхождению
  + Экономические / Неэкономические
  + Естественные / Культурные
  + Врожденные / Простые приобретенные / Сложные приобретенные
* От роли в воспроизводстве рабочей силы
  + Физические
  + Социальные
  + Духовные

Наибольшего распространения получила классификация потребностей по А. Маслоу, согласно которой, все потребности можно представить в виде пирамиды (от низжих к высшим, по последовательности их удовлетворения).



Биологические потребности обусловлены необходимостью поддержки жизнедеятельности.

Физиолого-психологическая потребность в безопасности и уверенности в будущем, позволяет сохранить гомеостаз; Для продолжения рода необходим секс. Так же к этому классу потребностей можно отнести потребность в информации, так как в отсутствие нервных сигналов – ткань деградирует, а психика в условиях сенсорной депривации расстраивается.

Потребность в любви и общении является психосоциальной потребностью, реализация которой позволяет людям действовать в группах

Потребность в признании и самоутверждении является чисто социальной потребностью, реализация которой позволяет определить место в обществе

Потребность в самовыражении является творческой, созидательной потребностью.

*15 Законы Брукса* **Программа и программный продукт.** Программный продукт отличается от программы:

* максимально обобщённым диапазоном и видом входных данных
* тщательным тестированием, что является неожиданно сложным этапом
* наличием подробной документации

Программный продукт требует в 3 раза больших затрат времени, чем программа (глава 1).

**Мифический человеко-месяц.** Время выполнения проекта не обратно пропорционально числу программистов, по крайней мере по 2 причинам.

1. В программировании, в отличие от, например, сбора хлопка, работа не может быть произвольно разделена на несколько независимых частей. Части проекта зависят друг от друга, и некоторые задачи можно начинать выполнять только после того, как будут закончены другие.
2. Программисты должны тратить часть времени на взаимодействие друг с другом.

Если есть *N* программистов, то количество пар программистов равно *N*(*N*—1)/2, то есть с ростом числа программистов затраты времени на взаимодействие растут квадратично. Поэтому начиная с какого-то N, рост числа программистов *замедляет* выполнение проекта.

Если сроки сорваны, наём новых программистов замедляет выполнение проекта и по другой причине: новичкам требуется время на обучение. В книге сформулирован «закон Брукса»: **«Если проект не укладывается в сроки, то добавление рабочей силы задержит его ещё больше».**

При очень большом числе программистов проект может быть вообще никогда не закончен: из-за общей неразберихи, попытки исправить существующие ошибки в программном обеспечении порождают новые ошибки, так что система не улучшается (глава 2).

**Хирургические группы.** Разумно, если в группе разработчиков есть один «хороший» программист, реализующий самые критические части системы, и несколько других, помогающих ему или реализующих менее критические части. Так делаются хирургические операции. Кроме того, по мнению Брукса, лучшие программисты работают в 5-10 раз быстрее остальных (глава 3).

**Концептуальная целостность.** Для обеспечения концептуальной целостности системы необходимо отделить архитектуру от реализации. Один главный архитектор (или небольшая группа), действуя в интересах пользователя, решают, что должно входить в систему, а что не должно. «Очень крутая» идея может быть отвергнута, если предлагаемая возможность не вписывается в общий дизайн системы. Простота очень важна; может быть полезным реализовать только часть возможностей, на которые способна система, потому что если система слишком сложна, часть её возможностей будет оставаться неиспользованной.

Главный архитектор должен сформулировать свои решения в виде руководства для пользователя (глава 4).

**Эффект второй системы.** Программист, разрабатывающий свою *вторую* систему, склонен добавлять все те возможности, которые он не смог добавить в свою первую систему (из-за нехватки времени). Поэтому вторая система часто получается перегруженной возможностями (глава 5).

**Формальные документы.** Каждый менеджер проекта должен составить небольшой набор формальных документов, описывающих цели проекта, как, кем и когда они будут реализованы, и сколько они будут стоить. Эти документы могут вскрыть несоответствия, которые иначе было бы трудно заметить.

Каждая группа разработчиков получает набор требований к своей части системы, включая точное описание её функциональности и предельные требования к процессорному времени, занимаемой памяти, месту на диске и т. д.

**Взаимодействие.** Чтобы предотвратить катастрофу, группы разработчиков должны взаимодействовать друг с другом всеми возможными способами. Вместо того чтобы строить предположения по поводу реализуемой им функции, разработчик должен задавать архитектору уточняющие вопросы, поскольку предположения могут оказаться совершенно неверными. «Предположение — мать провала».

**Пилотная система.** Перед тем, как разрабатывать окончательную систему, необходимо разработать пилотную систему. Пилотная система выявит ошибки в проектировании, после чего она должна быть полностью переделана (глава 11). Эту идею Брукс отвергает через 20 лет в главе 19, так как за 20 лет изменился подход к созданию программ — на место принятой в 60-х—70-хкаскадной модели разработки пришла итеративная.

**Версии и замораживание системы.** По мере создания системы, требования пользователя к ней могут меняться под влиянием его опыта работы с незаконченной системой. Эти пожелания пользователя следует учитывать, но только до какого-то момента, иначе работа над системой никогда не будет закончена. После этого спецификации замораживаются, и все последующие требования изменений откладываются до начала работы над следующей версией (глава 11).

**Специализированные утилиты.** Вместо того, чтобы каждый программист писал собственные утилиты, в каждой группе разработчиков должен быть один программист, ответственный за написание утилит для своей группы (например, генератор кода, создающий код в соответствии с какими-то спецификациями). Должна быть также группа, создающая утилиты для всех работающих над данной системой (глава 12).

**Снижение стоимости разработки.** Брукс приводит 2 способа снизить стоимость разработки программного обеспечения:

* Нанять программистов только после того, как построена архитектура системы. Иначе при длительности этой стадии, например, в несколько месяцев программистам будет нечего делать.
* Купить часть программного обеспечения у других разработчиков.