**Вопросы по лекции №14.**

**Низкоуровневые средства обнаружения ошибочного функционирования ПО и низкоуровневые методы защиты при конструировании безопасного ПО.**

Бригада СанУ11: Ковалев К.А., Терентьев Е.А., Щекочихин Д.А., Оганисян Т.С., Шабакаев А.Р.

Выполнил: Оганисян Т.С.

Группа 6411

**1.Назовите треугольник ограничений при разработке ПО.**

Управление разработкой программных проектов – это деятельность, направленная на организацию, планирование и контроль работы участников разработки ПО, на мотивацию их деятельности с целью выполнения треугольника ограничений, сопровождающих каждую инженерную разработку: обеспечения требуемых качества ПО, сроков и бюджета разработки ПО.

**2. Из каких действий состоит управление программными проектами**

**Планирование** преобразует цели и требования проекта в конкретные задачи и задаёт порядок действий для их достижения. Сроки планирования «танцуют» от трудоемкости разработки.

Для выполнения поставленных задач и планов по их достижению нужно создать **организацию работ**, которая задаёт распределение труда, ответственности и ролей разработчиков ПО, структуру и организационные формы их взаимодействия и взаимодействия с заказчиком при разработке ПО.

**Контроль выполнения планов** создаёт обратную связь, определяющую фактическое состояние разработки и сравнивающую фактическое и планируемое на данный момент состояние для выработки управляющих воздействий.

**Мотивация** - воодушевляет и стимулирует людей на работу.

**3. Назовите основные функции планирования**

1.Преобразование потребностей пользователя (заказчика) и целей программного проекта в конкретные задачи, выполнением которых можно управлять и контролировать.

2.Определение необходимых ресурсов для выполнения проекта

3.Координация командной работы над проектом. Выполнение проекта разбивается на отдельные работы, которые можно выполнять последовательно или параллельно.

4.Оценка потенциальных рисков проекта. Знание о существовании этих рисков позволит своевременно приготовиться к их адресации.

5.Сигнализация о возникновении проблем.

**4. Что такое критический путь сетевого графика разработки ПО? Что он определяет?**

Сетевой график всегда начинается с единственного начального события, содержит цепочку последовательно выполняемых событий и заканчивается событием завершения работ.

Наиболее длинный из этих путей называется критическим путём графика. Критический путь сетевого графика определяет срок выполнения проекта в целом.

**5.Почему важна правильная топология сетевого графика разработки ПО? От какой ситуации она защищает?**

Правильная и подробная топология плана разработки ПО гарантирует защиту от ситуаций типа «всё в основном есть», но работу завершить нельзя – упустили разработку одной программы и теперь все будут ждать пока она подтянется и встанет на сборку ПО. Правильный и подробный сетевой график позволяет также «расшить» узкие места путем изменения топологии.

**6.Приведите сравнительные характеристики планирования с использованием сетевых графиков и диаграмм Ганта.**

В отличие от сетевого графика диаграммы Ганта – это матрица. На диаграмме Ганта по вертикали записывается перечень последовательных работ, а по горизонтали – календарная сетка. В этих координатах для каждой работы указывается ее длительность. Это – достаточно древний способ представления плана, по которому удобно контролировать планируемое и реальное состояние работы.

Однако для сложных работ сетевой график, как цепочка последовательных и параллельных событий, связанных между собой, показывает в графическом виде «что от чего зависит», что не видно на диаграммах Ганта. Поэтому пересчитывать критические пути и искать выходы из критических ситуаций планирования удобнее на сетевом графике, а контролировать ход работ удобнее на диаграммах Ганта.

**7.Как определяются плановые сроки разработки ПО?**

Сроки разработки объективно не очень поддаются точным предсказаниям. Поэтому сроки топологически правильного графика работ не всегда выполняются. Здесь руководителям программных проектов важно своевременно предпринимать меры по усилению тех или иных сорванных направлений работ с коррекцией сроков графика.

В программостроении уже стало обычным явлением то, что разработчики без достаточного основания называют слишком оптимистичные сроки, не учитывая множество препятствий, носящих случайный характер. Среди опытных руководителей и заказчиков даже распространено неписаное правило: умножать на 2 оценку трудоемкости, которую сделал разработчик программы.

Еще один распространенный источник занижения сроков — необоснованные ожидания на применение новых технологий и средств разработки. Средняя производительность в программном производстве растет всего лишь на 3–5% в год. Если выполнить график не удаётся, то его надо корректировать, в противном случае работы проводятся вне графика т.е. не координируются и не управляются.

Выполнил: Терентьев Е.А.

Группа 6411

**1. Назовите треугольник ограничений при разработке ПО.**

Обеспечения требуемых качества ПО, сроков и бюджета разработки ПО.

**2. Из каких действий состоит управление программными проектами**

- Планирование. Обозначает цели, задачи и риски, и задает порядок действий для их достижения.

- Организация. Создает структуру исполнения запланированных действий, задаёт распределение труда, ответственности и ролей разработчиков ПО, структуру и организационные формы их взаимодействия и взаимодействия с заказчиком при разработке ПО.

- Контроль. Позволяет определить какие отклонения от запланированных целей и задач имеют место, сравнивает фактическое и планируемое состояние для выработки управляющих воздействий.

- Мотивация. Воодушевляет и стимулирует людей на работу.

**3. Назовите основные функции планирования**

Основными функциями планирования являются:

1. Преобразование потребностей заказчика и целей программного проекта в конкретные задачи, выполнением которых можно управлять и контролировать.

2. Определение необходимых ресурсов для выполнения проекта.

3. Координация командной работы над проектом. Планы делают возможной координацию коллективных действий путем определения того, кто, что и когда делает.

4. Оценка потенциальных рисков проекта.

5. Сигнализация о возникновении проблем. Отклонение проекта от плана – сигнал о возникновении проблемы, которую надо решать.

**4. Что такое критический путь сетевого графика разработки ПО? Что он определяет?**

Сетевой график ПО определяет связи между различными работами, представленными в виде событий графика, имеющих определённый номер, и связанных между собой стрелками. Сетевой график всегда начинается с единственного начального события, содержит цепочку последовательно выполняемых событий и заканчивается событием завершения работ. Наиболее длинный из путей от начального события к событию завершения работ называется критическим путём графика. Критический путь сетевого графика определяет срок выполнения проекта в целом. Критический путь графика всегда подвергаются анализу на предмет возможностей его сокращения. Так происходит оптимизация сетевого графика проведения работ.

**5. Почему важна правильная топология сетевого графика разработки ПО? От какой ситуации она защищает?**

Топология сетевого графика – графическое отображение связей между событиями плана, определяющее возможность последовательного или параллельного проведения работ. Сетевой график очень удобная форма представления плана, на которой видны объективные связи и зависимости событий плана друг от друга. Правильная и подробная топология плана разработки ПО гарантирует отсутствие ситуации, когда, часть работ уже завершена, но для продолжения необходимо выполнение определенных работ, которые в данный момент задерживаются. В данной ситуации разработку придется приостановить, до того момента пока все работы не будут выполнены.

**6. Приведите сравнительные характеристики планирования с использованием сетевых графиков и диаграмм Ганта.**

Для сложных работ сетевой график, как цепочка последовательных и параллельных событий, связанных между собой, показывает в графическом виде «что от чего зависит», что не видно на диаграммах Ганта. Поэтому пересчитывать критические пути и искать выходы из критических ситуаций планирования удобнее на сетевом графике, а контролировать ход и сроки выполнения работ удобнее на диаграммах Ганта.

**7. Как определяются плановые сроки разработки ПО?**

Плановые сроки разработки ПО зависят от множества факторов. Плановые сроки выполнений событий плана носят более субъективный характер.

Трудоемкость разработки. В малых программных проектах разработка непосредственно кода является наиболее трудоемкой частью проекта. В больших программных проектах выбор архитектуры, структуризация, затем интеграция и комплексная отладка взаимодействия занимают все большее время

Желание руководителей. «Агрессивный» график программного проекта с весьма сжатыми и нереалистичными сроками появляется из-за того, что руководство проектом и/или заказчик боятся «отпустить проект», полагая, что, работы по проекту займут все отведенное для него время. Если разработчикам будет выделено слишком много времени, то в начале они будут работать спустя рукава, а затем аврально с ошибками наверстывать упущенное время. Если график излишне жесткий, то с целью сэкономить время, недостаточно времени и внимания уделяется начальным этапам - анализу требований и проектированию. Исправление ошибок, допущенных на этих этапах, приведет к существенным дополнительным затратам впоследствии. Здесь руководителям программных проектов важно держать руку «на пульсе» и своевременно корректировать сроки выполнения проекта.

Свою лепту в установку сроков графика разработки ПО вносят и сами разработчики ПО. Часто разработчики без достаточного основания называют слишком оптимистичные сроки, не учитывая множество препятствий, носящих случайный характер.

Еще один источник сроков — необоснованные ожидания на применение новых технологий и средств разработки. Эти ожидания, как правило, не оправдываются.

Выполнил: Шабакаев А.Р.

Группа 6411

**1.Назовите треугольник ограничений при разработке ПО.**

Треугольник состоит из:

- обеспечения требуемого качества ПО;

- сроков;

- бюджета разработки ПО.

**2. Из каких действий состоит управление программными проектами**

- организация;

- планирование;

- контроль работы участников разработки ПО;

- мотивацию их деятельности;

Цель: выполнение треугольника ограничений.

**3.Назовите основные функции планирования**

- Преобразование потребностей пользователя (заказчика) и целей программного проекта в конкретные задачи, выполнением которых можно управлять и контролировать;

- Определение необходимых ресурсов для выполнения проекта;

- Координация командной работы над проектом;

- Оценка потенциальных рисков проекта;

- Сигнализация о возникновении проблем.

**4. Что такое** **критический путь сетевого графика разработки ПО? Что он определяет?**

Критический путь сетевого графика разработки ПО – полный путь от начального до конечного события, наиболее длинный из всех возможных путей сетевого графика разработки ПО.

Критический путь сетевого графика определяет срок выполнения проекта в целом.

**5.Почему важна правильная топология сетевого графика разработки ПО? От какой ситуации она защищает?**

Правильная топология сетевого графика учитывает все необходимые работы по разработке ПО. В ней видны объективные связи и зависимости событий плана друг от друга.

Правильная и подробная топология плана разработки ПО гарантирует защиту от ситуаций типа «всё в основном есть», но работу завершить нельзя. К примеру, упустили разработку одной программы и теперь все будут ждать пока она подтянется и встанет на сборку ПО. Правильный и подробный сетевой график позволяет также «расшить» узкие места путем изменения топологии.

**6.Приведите сравнительные характеристики планирования с использованием сетевых графиков и диаграмм Ганта.**

**Сетевой график**: представлен в виде графа, включающего цепочки последовательных и параллельных событий, на котором видно «что от чего зависит», в отличие от диаграмм Ганта. Следовательно, по сетевому графику удобнее пересчитывать критические пути и искать выходы из критических ситуаций планирования.

Диаграмма Ганта: представлен в виде матрицы, где по вертикали приведена перечень последовательных работ, по горизонтали календарная сетка. В отличие от сетевого графика, по ней удобнее контролировать планируемое и реальное состояние работы, ход работ.

**7.Как определяются плановые сроки разработки ПО?**

Плановые сроки определяются посредством анализа требований к ПО, проектированию ПО, составления топологически правильного графика работ с учетом множества препятствий случайного характера.

Недостаточно обоснованное уменьшение срока разработки может возникнуть в силу разных причин:

- со стороны менеджеров: задание срока разработки, объективно не поддающийся точным предсказаниям. Делают они это потому, что хотят себя «отделить» от конкурентов, а иногда под давлением руководства проектом или под давлением обстоятельств.

- со стороны разработчиков ПО: слишком оптимистичная оценка срока разработки ПО, не имеющая достаточного основания. Иногда в этом сказывается желание понравиться заказчику и отделить себя от более осторожных конкурентов. Более опытные руководители и заказчики, как правило, оценку трудоемкости, которую дал разработчик, умножают на 2.

- необоснованные ожидания на применение новых технологий и средств разработки. Согласно статистике, средняя производительность в программном производстве растет всего лишь на 3–5% в год.

- «агрессивный» график работ. Появляется такой график из-за опасений руководства проекта или заказчика «отпустить проект» - полагая, что согласно закону Паркинсона, работы по проекту займут все отведенное для него время. Из-за этого, может недостаточно времени уделяться начальным этапам разработки ПО, что приведет к еще большим затратам ресурсов на проект (в том числе временных затрат).

Выполнил: Щекочихин Д.А  
группа 6411

**1.Назовите треугольник ограничений при разработке ПО.**

Обеспечения требуемых качества ПО, сроков и бюджета разработки ПО.

**2. Из каких действий состоит управление программными проектами.**

1. Планирование. Обозначает цели, задачи и риски, и задает порядок действий для их достижения.
2. Организация. Создает структуру исполнения запланированных действий, задаёт распределение труда, ответственности и ролей разработчиков ПО, структуру и организационные формы их взаимодействия и взаимодействия с заказчиком при разработке ПО.
3. Контроль. Позволяет определить какие отклонения от запланированных целей и задач имеют место, сравнивает фактическое и планируемое состояние для выработки управляющих воздействий.
4. Мотивация. Воодушевляет и стимулирует людей на работу.

**3.Назовите основные функции планирования.**

1. Преобразование потребностей заказчика и целей программного проекта в конкретные. задачи, выполнением которых можно управлять и контролировать.
2. Определение необходимых ресурсов для выполнения проекта.
3. Координация командной работы над проектом.
4. Оценка потенциальных рисков проекта.
5. Сигнализация о возникновении проблем.

**4. Что такое критический путь сетевого графика разработки ПО? Что он определяет?**

Сетевой график ПО определяет связи между различными работами, представленными в виде событий графика, имеющих определённый номер, и связанных между собой стрелками. Сетевой график всегда начинается с единственного начального события, содержит цепочку последовательно выполняемых событий и заканчивается событием завершения работ. Наиболее длинный из путей от начального события к событию завершения работ называется критическим путём графика. Критический путь сетевого графика определяет срок выполнения проекта в целом. Критический путь графика всегда подвергаются анализу на предмет возможностей его сокращения.

**5.Почему важна правильная топология сетевого графика разработки ПО? От какой ситуации она защищает?**

Топология сетевого графика – графическое отображение связей между событиями плана, определяющее возможность последовательного или параллельного проведения работ. Сетевой график очень удобная форма представления плана, на которой видны объективные связи и зависимости событий плана друг от друга. Правильная и подробная топология плана разработки ПО гарантирует отсутствие ситуации, когда, часть работ уже завершена, но для продолжения необходимо выполнение определенных работ, которые в данный момент задерживаются. В данной ситуации разработку придется приостановить, до того момента пока все работы не будут выполнены.

**6.Приведите сравнительные характеристики планирования с использованием сетевых графиков и диаграмм Ганта.**

Для сложных работ сетевой график, как цепочка последовательных и параллельных событий, связанных между собой, показывает в графическом виде «что от чего зависит», что не видно на диаграммах Ганта. Поэтому пересчитывать критические пути и искать выходы из критических ситуаций планирования удобнее на сетевом графике, а контролировать ход и сроки выполнения работ удобнее на диаграммах Ганта.

**7.Как определяются плановые сроки разработки ПО?**

Плановые сроки разработки ПО зависят от множества факторов:

1. Трудоемкость разработки. В малых программных проектах разработка непосредственно кода является наиболее трудоемкой частью проекта. В больших программных проектах выбор архитектуры, структуризация, затем интеграция и комплексная отладка взаимодействия занимают все большее время
2. Желание руководителей. «Агрессивный» график программного проекта с весьма сжатыми и нереалистичными сроками появляется из-за того, что руководство проектом и/или заказчик боятся «отпустить проект», полагая, что, работы по проекту займут все отведенное для него время. Если разработчикам будет выделено слишком много времени, то в начале они будут работать спустя рукава, а затем аврально с ошибками наверстывать упущенное время. Если график излишне жесткий, то с целью сэкономить время, недостаточно времени и внимания уделяется начальным этапам - анализу требований и проектированию. Исправление ошибок, допущенных на этих этапах, приведет к существенным дополнительным затратам впоследствии. Здесь руководителям программных проектов важно держать руку «на пульсе» и своевременно корректировать сроки выполнения проекта.
3. Свою лепту в установку сроков графика разработки ПО вносят и сами разработчики ПО. Часто разработчики без достаточного основания называют слишком оптимистичные сроки, не учитывая множество препятствий, носящих случайный характер.
4. Еще один источник сроков — необоснованные ожидания на применение новых технологий и средств разработки. Эти ожидания, как правило, не оправдываются.

Выполнил: Ковалев К.А  
группа 6411

**1.Назовите треугольник ограничений при разработке ПО.**

Обеспечения требуемых качества ПО, сроков и бюджета разработки ПО.

**2. Из каких действий состоит управление программными проектами**

* Планирование. Обозначает цели, задачи и риски, и задает порядок действий для их достижения.
* Организация. Создает структуру исполнения запланированных действий, задаёт распределение труда, ответственности и ролей разработчиков ПО, структуру и организационные формы их взаимодействия и взаимодействия с заказчиком при разработке ПО.
* Контроль. Позволяет определить какие отклонения от запланированных целей и задач имеют место, сравнивает фактическое и планируемое состояние для выработки управляющих воздействий.
* Мотивация. Воодушевляет и стимулирует людей на работу.

**3.Назовите основные функции планирования.**

* Преобразование потребностей заказчика и целей программного проекта в конкретные задачи, выполнением которых можно управлять и контролировать;
* Определение необходимых ресурсов для выполнения проекта;
* Координация командной работы над проектом;
* Оценка потенциальных рисков проекта;
* Сигнализация о возникновении проблем.

**4. Что такое критический путь сетевого графика разработки ПО? Что он определяет?**

Сетевой график всегда начинается с единственного начального события, содержит цепочку последовательно выполняемых событий и заканчивается событием завершения работ. Наиболее длинный из этих путей называется критическим путём графика. Критический путь сетевого графика определяет срок выполнения проекта в целом.

**5.Почему важна правильная топология сетевого графика разработки ПО? От какой ситуации она защищает?**

Топология сетевого графика – графическое отображение связей между событиями плана, определяющее возможность последовательного или параллельного проведения работ. Сетевой график очень удобная форма представления плана, на которой видны объективные связи и зависимости событий плана друг от друга. Правильная и подробная топология плана разработки ПО гарантирует отсутствие ситуации, когда, часть работ уже завершена, но для продолжения необходимо выполнение определенных работ, которые в данный момент задерживаются. В данной ситуации разработку придется приостановить, до того момента пока все работы не будут выполнены.

**6.Приведите сравнительные характеристики планирования с использованием сетевых графиков и диаграмм Ганта.**

Для сложных работ сетевой график, как цепочка последовательных и параллельных событий, связанных между собой, показывает в графическом виде «что от чего зависит», что не видно на диаграммах Ганта. Поэтому пересчитывать критические пути и искать выходы из критических ситуаций планирования удобнее на сетевом графике, а контролировать ход и сроки выполнения работ удобнее на диаграммах Ганта.

**7.Как определяются плановые сроки разработки ПО?**

* Трудоемкость разработки. В малых программных проектах разработка непосредственно кода является наиболее трудоемкой частью проекта. В больших программных проектах выбор архитектуры, структуризация, затем интеграция и комплексная отладка взаимодействия занимают все большее время;
* Желание руководителей. «Агрессивный» график программного проекта с весьма сжатыми и нереалистичными сроками появляется из-за того, что руководство проектом и/или заказчик боятся «отпустить проект», полагая, что, работы по проекту займут все отведенное для него время. Если разработчикам будет выделено слишком много времени, то в начале они будут работать спустя рукава, а затем аврально с ошибками наверстывать упущенное время. Если график излишне жесткий, то с целью сэкономить время, недостаточно времени и внимания уделяется начальным этапам - анализу требований и проектированию. Исправление ошибок, допущенных на этих этапах, приведет к существенным дополнительным затратам впоследствии. Здесь руководителям программных проектов важно держать руку «на пульсе» и своевременно корректировать сроки выполнения проекта;
* Свою лепту в установку сроков графика разработки ПО вносят и сами разработчики ПО. Часто разработчики без достаточного основания называют слишком оптимистичные сроки, не учитывая множество препятствий, носящих случайный характер;
* Еще один источник сроков — необоснованные ожидания на применение новых технологий и средств разработки. Эти ожидания, как правило, не оправдываются.