advanced solutions and new paradigms for decreasing the discovery and development times for new drugs, and potentially reducing the development costs.

Implantable materials: nanotechnology brings a variety of new high surface area biocompatible nanomaterials and coatings to increase the adhesion, durability and lifespan of implants. Nanostructures are being researched for the preparation and improvement of tissue regeneration scaffolds.

Implantable devices: micro and nanosized sensors can make use of a wide range of technologies that most effectively detect a targeted chemical or physical property. Implantable sensors can also work with a series of medical devices that administer treatment automatically if required.

Diagnostic tools are based on two areas: genetic testing and imaging. As example we can consider such devices as nanoparticle probes. Nanoparticles with a magnetic core are attached to a cancer antibody that attracts cancer cells. The nanoparticles are also linked with a dye which is highly visible on an MRI. When these nanoprobes latch onto cancer cells they can be detected on the MRI. The cancer cells can then be destroyed by laser or low dosage killing agents that attack only the diseased cells.

A numerous novel nanomedicine-related application are under development or nearing commercialization. New nanotechnologies may offer the only hope for systematic, affordable, and long term improvements to the health status of our population. This is because nano therapies could, in the long run, be much more economical, effective and safe and could greatly reduce the cost of or substantially eliminate current medical procedures. So, nanomedicine is future medicine.

List of references:
1. Robert A. Freitas Jr. Nanomedicine- Georgetown, 1999;
2. http://www.foresight.org/Nanomedicine/

AGILE SOFTWARE DEVELOPMENT MANAGEMENT

Belarusian State University of Informatics and Radioelectronics
Minsk, The Republic of Belarus

Turach D. V.

Lazarenko A. M. – senior lecturer

Computer science is a young science. Computer programmers my age were trained by engineers. That training dictated how we approached software development for an entire generation. But now after decades of building software to be expensive, unwanted, and unreliable we have come to realize software is different. Building software is more like creating a work of art, it requires creativity in design and ample craftsmanship to complete. Software remains malleable, often illogical, and incomplete forever. Agile software development is based on fundamental changes to what we considered essential to software development ten years ago.

Agile software development is a group of software development methods based on iterative and incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional teams. It promotes adaptive planning, evolutionary development and delivery, a time-boxed iterative approach, and encourages rapid and flexible response to change. It is a conceptual framework that promotes foreseen interactions throughout the development cycle.

Everyone realize the way a team works together is far more important than any process. While a new process can easily improve team productivity by a fraction, enabling team to work effectively as a cohesive unit can improve productivity by several times. Of course to be eligible for such a big improvement you must be working at a fraction of your potential now. Unfortunately, it isn't that uncommon.

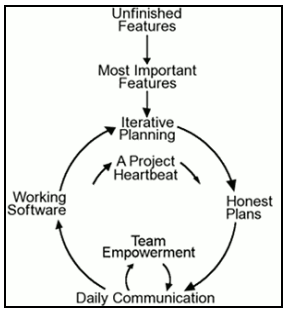
The most brilliant programmers alive working competitively in an ego-rich environment can’t get as much done as ordinary programmers working cooperatively as a self disciplined and self-organizing team. Therefore, everyone need a process where team empowerment and collaboration thrive to reach your full potential.

The first change is making the customer, the one who funds the software development, a valuable and essential team member. When the dead line gets close a traditional approach to reducing scope is to let the developers decide what will work properly and what won't. Instead let the customer make scope decisions a little at a time throughout the project.

When customer, or domain expert works directly with the development team everyone learns something new about the problem. True domain expertise and experience is essential to finding a simple, elegant, correct solution. A document can have plenty of information, but real knowledge is hard to put on paper. Left alone programmers must assume they know everything they need. When asking questions is difficult or slow the knowledge gap grows. The system will get built, but it won't solve the problem like one guided by an expert on a daily basis.

Perhaps the biggest problem with software development is changing requirements. Agile processes accept the reality of change versus the hunt for complete, rigid specifications. There are domains where requirements can't change, but most projects have changing requirements. For most projects readily accepting changes can actually cost less than ensuring requirements will never change.

Agile can produce working software starting with the first week of development so why not show it to the customer? Agile can learn so much more about the project requirements in the context of a working system. The changes team get this way are usually the most important to implement.



Pic. 1 – Agile software development process

Agile also means a fundamental change in how team manage its projects. If working software is what team will deliver then measure teams progress by how much team, have right now. Team will change its management style to be based on getting working software done a little at a time. The documents team used to create as project milestones may still be useful, just not as a measure of progress.

Instead of managing teams activities and waiting till the project ends for software, team will manage its requirements and demonstrate each new version to the customer. It is a hard change to make but it opens up new ways to develop software.

List of used sources:

1. Abrahamsson, P., Salo, O., Ronkainen, J., (2002). Agile Software Development Methods: Review and Analysis. VTT Publications

OBSTACLES TO RENEWABLE ENERGY

Belarusian State University of Informatics and Radioelectronics
Minsk, The Republic of Belarus

A.S. Heister

A.M. Lazarenko – senior lecturer

There is a great deal of information and enthusiasm today about the development and increased production of human's global energy needs from alternative energy sources. Solar energy, wind energy, hydropower, biomass energy and geothermal energy are all traditional sources of alternative energy that are making progress. The enthusiasm everyone shares for these developments has in many ways created a sense of complacency that our future energy demands will easily be met.

One major advantage with the use of renewable energy is that as it is renewable it is therefore sustainable and so will never run out. Renewable energy facilities generally require less maintenance than traditional generators. Their fuel being derived from natural and available resources reduces the costs of operation. Even more importantly, renewable energy produces little or no waste products such as carbon dioxide or other chemical pollutants, so has minimal impact on the environment. Renewable energy projects can also bring economic benefits to many regional areas, as most projects are located away from large urban centers and suburbs of the capital cities. These economic benefits may be from the increased use of local services as well as tourism.

It is easy to recognize the environmental advantages of utilizing the alternative and renewable forms of energy but we must also be aware of the disadvantages.

One disadvantage with renewable energy is that it is difficult to generate the quantities of electricity that are as large as those produced by traditional fossil fuel generators. This may mean that we need to reduce the amount of energy we use or simply build more energy facilities. It also indicates that the best solution to our energy problems may be to have a balance of many different power sources.

Another disadvantage of renewable energy sources is the reliability of supply. Renewable energy often relies on the weather for its source of power due to imperfect technology of renewable energy storage. Hydro generators need rain to fill dams to supply flowing water. Wind turbines need wind to turn the blades, and solar collectors need clear skies and sunshine to collect heat and make electricity. When these resources are unavailable so is the capacity to make energy from them. This can be unpredictable and inconsistent. The current cost of renewable energy technology is also far in excess of traditional fossil fuel generation. This is because it is a new technology and as such has extremely large capital cost.

Despite the promise of alternative energy sources, more appropriately called renewable energy, collectively they provide only about 8 percent of the world's energy needs [1]. This means that fossil fuels, along with nuclear energy — a controversial, non-renewable energy source — are supplying 92 percent of the world's energy resources.