

Chapter 2 : What is RAD model- advantages, disadvantages and when to use it?

Rapid Development builds beautiful custom homes for residents scattered throughout Fort Loramie, Ohio and the surrounding area. At Rapid Development, we build true custom homes which are tailor made to meet the needs and wants of each customer through our in house designer.

In view of these innovations in continuous evolution, the words found in the Decree of the Second Vatican Ecumenical Council, *Inter Mirifica*, promulgated by my venerable predecessor, the servant of God Paul VI, December 4, , appear even more pertinent: It tells us that the use of the techniques and the technologies of contemporary communications is an integral part of its mission in the third millennium. Moved by this awareness, the Christian community has taken significant steps in the use of the means of communication for religious information, for evangelization and catechesis, for the formation of pastoral workers in this area, and for the education to a mature responsibility of the users and the recipients of the various communications media. Many challenges face the new evangelization in a world rich with communicative potential like our own. The communications media have acquired such importance as to be the principal means of guidance and inspiration for many people in their personal, familial, and social behavior. We are dealing with a complex problem, because the culture itself, prescinding from its content, arises from the very existence of new ways to communicate with hitherto unknown techniques and vocabulary. Ours is an age of global communication in which countless moments of human existence are either spent with, or at least confronted by, the different processes of the mass media. I limit myself to mentioning the formation of personality and conscience, the interpretation and structuring of affective relationships, the coming together of the educative and formative phases, the elaboration and diffusion of cultural phenomena, and the development of social, political and economic life. The mass media can and must promote justice and solidarity according to an organic and correct vision of human development, by reporting events accurately and truthfully, analyzing situations and problems completely, and providing a forum for different opinions. An authentically ethical approach to using the powerful communication media must be situated within the context of a mature exercise of freedom and responsibility, founded upon the supreme criteria of truth and justice. Gospel Reflection and Missionary Commitment 4. Salvation History recounts and documents the communication of God with man, a communication which uses all forms and ways of communicating. The human being is created in the image and likeness of God in order to embrace divine revelation and to enter into loving dialogue with Him. Because of sin, this capacity for dialogue at both the personal and social level has been altered, and humanity has had to suffer, and will continue to suffer, the bitter experience of incomprehension and separation. God, however, did not abandon the human race, but sent his own Son Cf. In the Word made flesh communication itself takes on its most profound saving meaning: The communication between God and humanity has thus reached its perfection in the Word made flesh. The act of love by which God reveals himself, united to the response of faith by humanity, generates a fruitful dialogue. In light of so decisive and definitive a communication, the media provide a providential opportunity to reach people everywhere, overcoming barriers of time, of space and of language; presenting the content of faith in the most varied ways imaginable; and offering to all who search the possibility of entering into dialogue with the mystery of God, revealed fully in Christ Jesus. The Incarnate Word has left us an example of how to communicate with the Father and with humanity, whether in moments of silence and recollection, or in preaching in every place and in every way. He explains the Scriptures, expresses himself in parables, dialogues within the intimacy of the home, speaks in the squares, along the streets, on the shores of the lake and on the mountaintops. The personal encounter with him does not leave one indifferent, but stimulates imitation: There is, however, a culminating moment in which communication becomes full communion: Thanks to the Redemption, the communicative capacity of believers is healed and renewed. The encounter with Christ makes them new creatures, and permits them to become part of that people which he, dying on the Cross, has won through his blood, and introduces them into the intimate life of the Trinity, which is continuous and circular communication of perfect and infinite love among the Father, the Son and the Holy Spirit. Communication permeates the essential dimensions of the

Church which is called to announce to all the joyful message of salvation. For this reason, the Church takes advantage of the opportunities offered by the communications media as pathways providentially given by God to intensify communion and to render more penetrating the proclamation of His word. We give thanks to God for the presence of these powerful media which, if used by believers with the genius of faith and in docility to the light of the Holy Spirit, can facilitate the communication of the Gospel and render the bonds of communion among ecclesial communities more effective. A Change of Mentality and Pastoral Renewal 7. In the communications media the Church finds a precious aid for spreading the Gospel and religious values, for promoting dialogue, ecumenical and inter-religious cooperation, and also for defending those solid principles which are indispensable for building a society which respects the dignity of the human person and is attentive to the common good. The Church willingly employs these media to furnish information about itself and to expand the boundaries of evangelization, of catechesis and of formation, considering their use as a response to the command of the Lord: This is certainly not an easy mission in an age such as ours, in which there exists the conviction that the time of certainties is irretrievably past. Many people, in fact, believe that humanity must learn to live in a climate governed by an absence of meaning, by the provisional and by the fleeting. Those individuals in the Church community particularly gifted with talent to work in the media should be encouraged with pastoral prudence and wisdom, so that they may become professionals capable of dialoguing with the vast world of the mass media. The appreciation of the media is not reserved only to those already adept in the field, but to the entire Church Community. If, as has already been noted, the communications media take into account different aspects of the expression of faith, Christians must take into account the media culture in which they live: The current phenomenon of communications impels the Church towards a sort of pastoral and cultural revision, so as to deal adequately with the times in which we live. Pastors, above all, must assume this responsibility. Everything possible must be done so that the Gospel might permeate society, stimulating people to listen to and embrace its message. Such is the importance of the mass media that fifteen years ago I considered it inopportune to leave their use completely up to the initiatives of individuals or small groups, and suggested that they be decisively inserted into pastoral programs. One clear example today is how the Internet not only provides resources for more information, but habituates persons to interactive communication. However, alongside the Internet, other new means of communication, as well as traditional ones, should be used. Daily and weekly newspapers, publications of all types, and Catholic television and radio still remain highly useful means within a complete panorama of Church communications. While the content being communicated must obviously be adapted to the needs of different groups, the goal must always be to make people aware of the ethical and moral dimension of the information. The Church, which in light of the message of salvation entrusted to it by the Lord is also a teacher of humanity, recognizes the duty to offer its own contribution for a better understanding of outlooks and responsibilities connected with current developments in communications. Especially because these influence the consciences of individuals, form their mentality and determine their view of things, it is important to stress in a forceful and clear way that the mass media constitute a patrimony to safeguard and promote. The communications media must enter into the framework of organically structured rights and duties, be it from the point of view of formation and ethical responsibility, or from reference to laws and institutional codes. The positive development of the media at the service of the common good is a responsibility of each and every one. We are faced with three fundamental options: In the first place, a vast work of formation is needed to assure that the mass media be known and used intelligently and appropriately. The new vocabulary they introduce into society modifies both learning processes and the quality of human relations, so that, without proper formation, these media run the risk of manipulating and heavily conditioning, rather than serving people. This is especially true for young people, who show a natural propensity towards technological innovations, and as such are in even greater need of education in the responsible and critical use of the media. In the second place, I would like to recall our attention to the subject of media access, and of co-responsible participation in their administration. If the communications media are a good destined for all humanity, then ever-new means must be found — including recourse to opportune legislative measures — to make possible a true participation in their management by all. The culture of co-responsibility must be nurtured. Finally, there cannot be forgotten the

great possibilities of mass media in promoting dialogue, becoming vehicles for reciprocal knowledge, of solidarity and of peace. In a meeting with the editors of Catholic publications, my venerable predecessor, Pius XII, stated that something would be missing from the life of the Church were it not for public opinion. Communication both within the Church community, and between the Church and the world at large, requires openness and a new approach towards facing questions regarding the world of media. This communication must tend towards a constructive dialogue, so as to promote a correctly-informed and discerning public opinion within the Christian community. The Church, like other institutions and groups, has the need and the right to make its activities known. However, when circumstances require, it must be able to guarantee an adequate confidentiality, without thereby prejudicing a timely and sufficient communication about Church events. The latter, on the other hand, aided by the experience of the laity, can more clearly and more incisively come to decisions regarding both spiritual and temporal matters. To Communicate with the Power of the Holy Spirit

The great challenge of our time for believers and for all people of good will is that of maintaining truthful and free communication which will help consolidate integral progress in the world. Everyone should know how to foster an attentive discernment and constant vigilance, developing a healthy critical capacity regarding the persuasive force of the communications media. Also in this field, believers in Christ know that they can count upon the help of the Holy Spirit. Such help is all the more necessary when one considers how greatly the obstacles intrinsic to communication can be increased by ideologies, by the desire for profit or for power, and by rivalries and conflicts between individuals and groups, and also because of human weakness and social troubles. The modern technologies increase to a remarkable extent the speed, quantity and accessibility of communication, but they above all do not favor that delicate exchange which takes place between mind and mind, between heart and heart, and which should characterize any communication at the service of solidarity and love. The eternal Word made flesh, in communicating Himself, always shows respect for those who listen, teaches understanding of their situation and needs, is moved to compassion for their suffering and to a resolute determination to say to them only what they need to hear without imposition or compromise, deceit or manipulation. I tell you, on the Day of Judgment people will render an account for every careless word they speak. By your words you will be acquitted, and by your words you will be condemned. Do not be afraid of being opposed by the world! May the Blessed Virgin help us to communicate by every means the beauty and joy of life in Christ our Savior. To all I give my Apostolic Blessing! AAS 68 , AAS 81 , ; cf. AAS 84 , AAS 91 , AAS 88 , AAS 83 , Vatican City, , p. The Catechism of the Catholic Church, num. AAS 63 , According to the knowledge, competence, and prestige which they possess, they have the right and even at times the duty to manifest to the sacred pastors their opinion on matters which pertain to the good of the Church and to make their opinion known to the rest of the Christian faithful, without prejudice to the integrity of faith and morals, with reverence toward their pastors, and attentive to common advantage and the dignity of persons. Unofficial translation of the Pontifical Council.

Chapter 3 : Rapid Application Development: Definition, Steps, Advantages and Case Study

Rapid Development: Rapid Devment _p1 and millions of other books are available for Amazon Kindle. Learn more Enter your mobile number or email address below and we'll send you a link to download the free Kindle App.

Application Insights Storage Accounts At a high level, if we look around the architectural perspective here is how it looks like. App Services URL “https: As a basic template, the bot behaves like an Echo chatbot, so whatever you type it will reply the same. Navigate to the Channel option under the Bot Management Section and configure the channels your bot uses to communicate with users. Configuring Bot Channels Use this section for adding new channels, removing channels or updating the configuration for existing channels. Editing Code for Bot You can make quick changes to your bot code online, build the code in the web editor, and test your changes instantly. It enables us to manage end to end bot development and deployment. In this section, you have learned how to create, test and connect a bot using Azure Bot services. You have also learned about internal of services created through this Bot services service and how they are interlinked. Now, as a next step, we will take up the code from Azure service and set up the Git Code Repository and connect with our local development editor. Setting up Code Repository The App Service Code Editor also allows us to connect with Git Repository and publish the code, however, we will get the code downloaded and will use Visual Studio to take it further. In this section, we will explore how to get the code from Azure Bot services and download it and then develop locally using Visual Studio. Download the source code This process will create a Zip file package and would let you download the complete source code. Save the source code, unzip it and open it in Visual Studio. Even though Zip file has the same name of the Bot you have provided, the solution comes within has a different name which comes as part of the template. I hope this will be changed in future. For now, I have updated the projects and solution files to match the name that we provided for our Bot. Initial Solution Structure Now we have the source code in our favorite editor, time to push the code in source code repository. While creating the team project, make sure you choose Git as version control for the project. Publish the code in remote repository Once the code is published, go back to your team project and verify your code is published and available in the remote git repository. So, go ahead test it out quickly. Test your Code Repository It is a straightforward process just to ensure everything works fine. Do some changes in the solution files and push the code in the GIT. Testing Code Repository Changes You can modify your bot, changes business logics, Bot behaviors and push your code now. Once the code is published to Git Repo and find the reflected changes, you are good to go. App Services is still running with the initial code created during the Bot service creation process. In this section we will explore how to enable the Continuous Integration CI , so that code pushed to git is merged with master branch and then automatically deployed to App Service using Continuous Delivery CD. Some of them are: Right-click in the solution and select Configure Continuous Delivery option from the context menu. Select the branch as master as this will be the final branch for this solution and choose the respective subscription. Visual Studio will configure the resources for continuous build delivery on Team Project smarttaskbot to App Service smarttaskbot. Click on those links to review the build and release definition created through the process. Visual Studio Team Services now contains following two definitions: Drop folders take the build drop packages created by the smarttaskbot-CD and then deployments happen. Release Definition View Click on the Drop icon under the artifacts options to get some more insights. The build drop folders are the location where artifacts from the smarttaskbot-CI are getting published for the deployment. Then as a part of next step, the deployment process picks up the drop and publish it. Build Drop for CI Build To know more build and release management process, select the individual definition for both smarttaskbot-CI and smarttaskbot-CD and review the internal task and flows. This indicates, the CI build will trigger as soon as there are any changes in the master branch. At this point in time, if you push any changes from your local dev environment to Git, it will automatically trigger the CI Build and followed by a deployment to the App services using the release pipeline. You can review the similar process as shown in below screenshots post your code is pushed to Git. This streamline your end to end development and release management for your Bot. Because as an overall process the build will still validate the same set of

the task from the code, the only thing we need to disable the new build from Continuous Integration. Once the build definition is created we must link the build definition with the branch policies to trigger with. Setup Branch Policies In the Branch policies settings, you can see the types of rules you can specify, including blocking a pull request from completion if the build is failed. You can also set mandatory linking work items or have a code-reviewer approver with this. With all these, you can ensure the level of quality for your code is getting committed. This should not be the ideal case when a team is involved, it must be reviewed by peers and will add the Build Policies.

Chapter 4 : Rapid Application Development (RAD), a software development method | ToolsHero

The Rapid City Economic Development Partnership is a great starting point for any businesses considering a start-up, expansion, or relocation. Tools & Resources The Rapid City Economic Development Partnership is committed to helping businesses grow and expand.

Rapid Application Development model relies on prototyping and rapid cycles of iterative development to speed up development and elicit early feedback from business users. After each iteration, developers can refine and validate the features with stakeholders. RAD model is also characterized by reiterative user testing and the re-use of software components. Hence, RAD has been instrumental in reducing the friction points in delivering successful enterprise applications. Learn more about the benefits of Rapid Application Development Steps involved in implementing Rapid Application Development model Rapid development typically involves the following basic steps that complete the process Planning: The first step where developers, designers, engineers and technology people come together to discuss the project needs, limitations and specifications. The prototyping needs are identified and agreed upon. Once the design requirements are gathered, initial prototyping and modelling is done. Feedback from user experiences, help in designing the prototype and the overall architecture of the application. There can be multiple iterations. With the design and prototype in place, the real work begins with development. The basic coding, architecture, testing , deployment and integration to backend services happens in this phase. With rapid application development model, there can be changes and enhancements made as per the need of the project. The final transition stage is where the release of the developed application happens.. The application is deployed and the user can now use it to increase business efficiency WaveMaker RAD model WaveMaker makes use of the RAD model to provide a Rapid Application Development platform to create web and mobile applications. Open standards, easy customization and rapid prototyping are central to the platform. The biggest advantage of rad development is its inherent synergy with the requirements of the medium itself: Whereas other forms of creation, such as a towering skyscraper or a sleek new automobile, require meticulous planning and logical development, the very essence of software is both malleable and in constant evolution. Since code itself “ and by extension, the software which is powered by said code ” can be morphed quickly and easily during development, software is inherently adaptable, lending itself well to iteration and experimentation. Rapid Application development has many advantages over any traditional application model. As the name suggests, the model makes application development fast and easy with iterative prototyping. Simple drag and drop features and the use of minimum coding efforts make it usable even for business people who have limited coding and technology knowledge. By utilizing a rapid application development method, designers and developers can utilize knowledge and discoveries conceived during the development process itself to shape the design and or alter the software direction entirely. Ideal condition for development using a RAD Model is: When there is need to complete a project with great speed and agility, RAD is the best solution. With frequent possible iterations and prototyping possibility, RAD helps in building applications fast and deliver results. Rapid Application Model is appropriate for building applications that involve lower risks and need to built in a very short span of time. Keep me informed about your products, services, events and content in the future.

Chapter 5 : Rapid Application Development Model – WaveMaker

Rapid application development (RAD) describes a method of software development which heavily emphasizes rapid prototyping and iterative delivery. The RAD model is, therefore, a sharp alternative to the typical waterfall development model, which often focuses largely on planning and sequential design practices.

Finalize Software Define the Requirements At the very beginning, rapid application development sets itself apart from traditional software development models. The broad nature of the requirements helps you give specific requirements at different points of the development cycle. Prototype This is where the actual development takes place. Instead of following a strict set of requirements, developers create prototypes with different features and functions as fast as they can. More often than not, these prototypes are quickly made to work, just to show off certain features, without proper polish. This is normal, and the final product is only created during the finalization stage where the client and developer can both agree on the final product. With this feedback in mind, prototyping continues. Finalize Software Here, features, functions, aesthetics, and interface of the software are finalized with the client. Stability, usability, and maintainability are of paramount importance before delivering to the client. RAD software is great for small teams and quick projects. Here are a few advantages and disadvantages to using rapid application development. Rapid Application Development vs Other Development Models When compared to other software development models, rapid application development varies by a considerable amount. Obviously, the major difference is how rapid application development focuses on speed, when compared to other models which usually focus on bringing a working product to the customer. Another thing to note here is that rapid application development prefers having a single team without too many members. This allows for fast communication with quick meetings for quick information transfer. Other development models such as the waterfall model prefer having larger teams divided into different specializations. Since RAD framework is focused on speed, the development time here is less than that of other models. But the difference is usually small, since rapid application development prefers to churn out a lot of prototypes before the finalized product. Rapid application development is also heavily focused on keeping the end user involved throughout the entire stage of the development process. Other models usually only have user input at the beginning and the end of the development cycle. Prototypes built through the rapid application development model depend on the feedback from previous iterations, so reliable feedback from dependable sources can be immensely helpful. Rapid application development takes an on-the-fly approach, which makes sense for quick development which can change direction on a dime. Case Study of Rapid Application Development- Centric Consulting Rapid application development is particularly useful for small businesses that need software done quickly, whilst having a lot of input during the development process. Centric Consulting, a developer familiar with rapid application development and agile development methodologies, had a client come to them with a requirement for a software that would interface with their customers, one of which had over 35, employees, for procurement, invoicing, and payment. Centric Consulting was able to use agile and rapid application development methodologies to quickly understand what the client needed, speed up the development process using Ruby on Rails, and keep costs low using open source infrastructure. Throughout the development process, the customer was able to provide input as to what functionalities were required. All those functionalities were rapidly added as and when they were demanded, and ultimately, the product was delivered to the client. In the end, Centric Consulting not only met the demands of their client, but was also able to meet their needs and grow their business. While it still remains the champion, a lot has changed in 20 years. It is a no-code platform that lets anyone develop their own automated process in a matter of minutes instead of days or weeks. This is rapid application development taken to a new level—making applications as quickly as possible that are ready-to-use instantly by the entire company.

Chapter 6 : Rapid Application Development | Accenture Liquid Studios

The author emphasizes efficient development concepts with an examination of rapid development strategies and a study of classic Emphasizes possible, realistic and "best practice" approaches for managers, technical leads and self-managed teams.

While agile dictates the ideal working environment just shy of how many rubber ducks to keep on your desk, RAD focuses on how to build software products for your clients and end-users. Rapid Application Development Methodology Though exact practices and tools vary between specific RAD methodologies, their underlying phases remain the same: Define Requirements Rather than requiring that you spend months developing specifications with users, RAD begins by defining a loose set of requirements. We say loose because among the key principles of rapid application development is the permission to change requirements at any point in the cycle. This can be a prototype that satisfies all or only a portion of requirements as in early stage prototyping. Most RAD approaches have a finalization stage at which developers pay down technical debts accrued by early prototypes. Absorb Feedback With a recent prototype prepared, RAD developers present their work to the client or end-users. They collect feedback on everything from interface to functionality—it is here where product requirements may come under scrutiny. Clients may change their minds or discover that something that seemed right on paper makes no sense in practice. Clients are only human, after all. With feedback in hand, developers return to some form of step 2: If feedback is strictly positive and the client is satisfied with the prototype, developers can move to step 4. However, RAD has its drawbacks as well. Advantage Description Speed In the traditional waterfall approach, developers were unlikely to go on vacation after delivering the product. Clients would invariably request changes ranging from interface to functionality after first delivery. Cost In rapid application development, developers build the exact systems the client requires, and nothing more. In waterfall, IT risks building and fleshing out complex feature sets that the client may choose to gut from the final product. The time spent building zombie features can never be recovered, and that means the budget spent on them is lost. RAD reduces this risk and therefore reduces the cost. Developer Satisfaction In the traditional waterfall approach, developers work in silos devoid of feedback and positive affirmation for a product well-made. And when they finally get the opportunity to present their work to the client, the client may not roll out the red carpet for them. In RAD, the client is there every step of the way and the developer has the opportunity to present their work frequently. This gives them the confidence that when the final product is delivered, their work receives appreciation. Disadvantage Description Scale A close-knit team of developers, designers, and product managers can easily incorporate RAD practices because they have direct access to one another. When a project expands beyond a single team or requires inter-team communication, the development cycle invariably slows and muddles the direction of the project. Commitment In waterfall, the client spent most of their time apart from the development team after completing specifications. This allowed clients to focus on their primary tasks and developers to focus on building. In RAD, the frequent cycle of prototypes requires developers and clients to commit to frequent meetings that, on the outset, may appear to consume unnecessary cycles. Interface-Focus RAD methodology motivates developers to find the perfect solution for the client. The client judges the quality of the solution by what they can interact with—and often, all they interact with is a facade. As a consequence, some developers forego best practices on the back-end to accelerate development of the front-end-focused prototype. With the pros and cons of rapid application development laid out, we can determine which types of projects benefit most from RAD, and which do not. If you need to build an internal business tool or even a customer-facing portal, like an app or website, RAD techniques will help your team deliver a better experience to your end-user. However, if you are tasked with building mission-critical software flight controls, implant firmware, etc. A pilot with a failing control module or a heart attack survivor with a malfunctioning pacemaker cannot provide prototype feedback from beyond the grave. Tools for Rapid Application Development As you may now understand, rapid application development is more of a software development methodology rather than a specific language, tool, or interface. However, tools can help facilitate rapid design, development, and

feedback solicitation. Design and Prototyping Tools The products in this category help teams craft interactive designs at impressive speeds. And some tools on this list, like Webflow, allow designers to export the completed design as a functional cross-browser prototype.

Chapter 7 : What Is Rapid Application Development?

Therefore, Rapid Development takes every precaution to ensure your complete satisfaction and peace of mind. We work hard to guide and educate you, making the experience exciting and rewarding. We are also committed to staying on-time and within budget throughout the process.

After reading you will understand the basics of this powerful project management and software development approach and method. Unlike the so-called Waterfall method, RAD enables organizations to develop software faster. The Waterfall method is based on phases that follow one another and merge into one another in the software development process as if they were a waterfall. This sometimes resulted in unworkable applications. Integrated system Rapid Application Development is not about dependent phases but about simultaneous modular developments and introductions that allow a faster and better integration of the software system. The entire development process has already been integrated into the planning, and, where necessary, it can be supplemented, altered or refined. Rapid Application Development makes use of prototypes that serve as a working model that form an immediate part of the end product. Rapid Application Development does use a detailed pre-planning phase, which makes it easier to incorporate changes in the development process. A Rapid Application Development project consists of a small team of developers, experts and representatives from the customer to directly respond to the wishes and requirements of the customer that are used in developing the software. Reusable The main feature of Rapid Application Development is that it embraces the reuse of the prototypes that have been developed. Therefore, there is little waste of time and effort. Rapid Application Development focuses on the gathering of customer wishes and requirements through workshops, focus groups and prototypes the customer will test during the development process. Existing prototypes are reused as usable components. This creates a continuous integration of new components and enables a fast delivery to the customer. Phases of the model design The Rapid Application Development model divides the analysis, design, construction and the testing phases of the software development into a short series, which is developed in a cycle. It consists of the following phases: Business modelling The specific business objective and principles of the business model must be clear in advance. This makes clear what information needs to be distributed among the different business channels. A complete business analysis is performed to find vital information flows about the customer and how the information flow can be processed successfully. Data modelling All information that has been collected from the business model is assessed and analysed so that it will be of benefit to the customer. The characteristics of all the data are identified and defined. The relationship between these data is recorded and identified and described in detail so that it will become relevant to the business model. Process modelling The object- data sets defined in the data model are converted into the business information flow which is necessary to achieve specific business objectives that are in accordance with the business model. In this business model changes and improvements are implemented immediately. Also new process descriptions are added, unusable descriptions are removed or altered. Application The current software system has now been fully built and can be applied by means of coding and automation tools with which process and data models can be transformed into actual prototypes. Testing The total testing time is significantly reduced with Rapid Application Development, because prototypes have been previously tested independently of each other. This is done in a continuous process. The information flow and the overlaps between the components must be tested thoroughly to ensure complete coverage. As most of the programming components have been tested before, it reduces the risk of potential problems at the end of the process. Implementation This is the phase where the new system is being deployed and introduced. This happens in earlier phases. The developed prototypes are used and there where needed adjusted, extended or further developed. The prototype is the base for the end product. There are exceptional situations where the prototypes are totally redesigned. These situations hardly occur because the end-users are directly involved in the design of the prototype. Benefits Rapid Application Development is a concept that enables organizations to develop high-quality software systems more quickly. The important advantages to working with Rapid Application Development are: Working with usable prototypes enables early user testing. There is almost no

wastage because software components can easily be reused. New developments and improvements can be implemented quickly in a next series. There is little to no waste of time. Do you recognize the practical explanation or do you have more additions? What are your success factors regarding the Rapid Application Development approach? Share your experience and knowledge in the comments box below. If you liked this article, then please subscribe to our Free Newsletter for the latest posts on Management models and methods. More information Mackay, H. Social studies of science , 30 5 , Taming Wild Software Schedules. The multimod application framework: Computer methods and programs in biomedicine, 85 2 , How to cite this article: Retrieved [insert date] from ToolsHero: Your rating is more than welcome or share this article via Social media!

Chapter 8 : About Us - Rapid City Economic Development

Rapid application development (RAD) is a suite of software development methodology techniques used to expedite software application development. RAD uses predefined prototyping techniques and tools to produce software applications.

One of the problems with these methods is that they were based on a traditional engineering model used to design and build things like bridges and buildings. Software is an inherently different kind of artifact. Software can radically change the entire process used to solve a problem. As a result, knowledge gained from the development process itself can feed back to the requirements and design of the solution. RAD approaches, on the other hand, recognize that software development is a knowledge intensive process and provide flexible processes that help take advantage of knowledge gained during the project to improve or adapt the solution. Boehm and other subsequent RAD approaches emphasized developing prototypes as well as or instead of rigorous design specifications. Prototypes had several advantages over traditional specifications: A prototype could test some of the most difficult potential parts of the system early on in the life-cycle. This can provide valuable information as to the feasibility of a design and can prevent the team from pursuing solutions that turn out to be too complex or time consuming to implement. This benefit of finding problems earlier in the life-cycle rather than later was a key benefit of the RAD approach. The earlier a problem can be found the cheaper it is to address. Users are better at using and reacting than at creating specifications. In the waterfall model it was common for a user to sign off on a set of requirements but then when presented with an implemented system to suddenly realize that a given design lacked some critical features or was too complex. In general most users give much more useful feedback when they can experience a prototype of the running system rather than abstractly define what that system should be. Prototypes can be usable and can evolve into the completed product. One approach used in some RAD methods was to build the system as a series of prototypes that evolve from minimal functionality to moderately useful to the final completed system. The advantage of this besides the two advantages above was that the users could get useful business functionality much earlier in the process. It is important to distinguish between RAD as a general alternative to the waterfall model and RAD as the specific method created by Martin. The Martin method was tailored toward knowledge intensive and UI intensive business systems. These practitioners, and those like them, helped RAD gain popularity as an alternative to traditional systems project life cycle approaches. The RAD approach also matured during the period of peak interest in business re-engineering. The idea of business process re-engineering was to radically rethink core business processes such as sales and customer support with the new capabilities of Information Technology in mind. RAD was often an essential part of larger business re engineering programs. The rapid prototyping approach of RAD was a key tool to help users and analysts "think out of the box" about innovative ways that technology might radically reinvent a core business process. Users, managers, and IT staff members discuss and agree on business needs, project scope, constraints, and system requirements. It ends when the team agrees on the key issues and obtains management authorization to continue. User design phase " during this phase, users interact with systems analysts and develop models and prototypes that represent all system processes, inputs, and outputs. User Design is a continuous interactive process that allows users to understand, modify, and eventually approve a working model of the system that meets their needs. Construction phase " focuses on program and application development task similar to the SDLC. In RAD, however, users continue to participate and can still suggest changes or improvements as actual screens or reports are developed. Its tasks are programming and application development, coding, unit-integration and system testing. Cutover phase " resembles the final tasks in the SDLC implementation phase, including data conversion, testing, changeover to the new system, and user training. Compared with traditional methods, the entire process is compressed. As a result, the new system is built, delivered, and placed in operation much sooner. The advantages of RAD include: By having users interact with evolving prototypes the business functionality from a RAD project can often be much higher than that achieved via a waterfall model. The software can be more usable and has a better chance to focus on business problems that

are critical to end users rather than technical problems of interest to developers. Although much of the literature on RAD focuses on speed and user involvement a critical feature of RAD done correctly is risk mitigation. A RAD approach can focus in early on the key risk factors and adjust to them based on empirical evidence collected in the early part of the process. More projects completed on time and within budget. By focusing on the development of incremental units the chances for catastrophic failures that have dogged large waterfall projects is reduced. In the Waterfall model it was common to come to a realization after six months or more of analysis and development that required a radical rethinking of the entire system. With RAD this kind of information can be discovered and acted upon earlier in the process. The risk of a new approach. For most IT shops RAD was a new approach that required experienced professionals to rethink the way they worked. Humans are virtually always averse to change and any project undertaken with new tools or methods will be more likely to fail the first time simply due to the requirement for the team to learn. Requires time of scarce resources. One thing virtually all approaches to RAD have in common is that there is much more interaction throughout the entire life-cycle between users and developers. In the waterfall model, users would define requirements and then mostly go away as developers created the system. In RAD users are involved from the beginning and through virtually the entire project. This requires that the business is willing to invest the time of application domain experts. The paradox is that the better the expert, the more they are familiar with their domain, the more they are required to actually run the business and it may be difficult to convince their supervisors to invest their time. Without such commitments RAD projects will not succeed. One of the advantages of RAD is that it provides a flexible adaptable process. The ideal is to be able to adapt quickly to both problems and opportunities. There is an inevitable trade-off between flexibility and control, more of one means less of the other. If a project e. The focus on prototypes can be taken too far in some cases resulting in a "hack and test" methodology where developers are constantly making minor changes to individual components and ignoring system architecture issues that could result in a better overall design. RAD typically focuses on small to medium-sized project teams. The other issues cited above less design and control present special challenges when using a RAD approach for very large scale systems.

Chapter 9 : Zoho Creator: A platform for rapid application development.

RAD model is Rapid Application Development model. It is a type of incremental blog.quintoapp.com RAD model the components or functions are developed in parallel as if they were mini projects.