

# **SWE 577-578 Term Paper Due: 18 May 2015, 09:00 (-25% / each late day)**

## **Term Paper Outline**

This term paper consists of a brief survey on a selected topic from the list given in the following pages. **You may pick topics offered by the SWE instructors or define your own software engineering related topic if you can take the approval of your selected advisor.**

Find technical publications on the selected topic. You are required to prepare a term paper in your own words by understanding and identifying that selected topic. **Do not cut and paste either from a paper or any other source on the Internet.** Please be aware that our department is using software for checking direct copying. Your paper **should not** exceed 4-6 pages (max 3000 words) of your writing with 12 pt fonts, single line spacing, 2.5 cm margins, non-exaggerated paragraph separations. Please, spell-check and proofread your term paper before you submit.

**A)** In order to select candidate advisors and topics, please send an e-mail to **pinar.yolum@boun.edu.tr** containing at least **three candidate topics** offered by **three different advisors** (some of the topics can be described by you as long as an advisor agrees with it) before **15 February 2015, Sunday at 23:59**. Watch out every instructor has a limited quota of 4-7 students and the requests will be handled in a First Come First Served (FCFS) order. **If you do not send your preferences until the due time, you will be assigned a term paper topic and an advisor automatically.**

**B) Your minimum 2 page long progress report should be e-mailed to your paper advisor (due 6 April 2015)**

**Name Your File:** SWE577-Paper-Progress-2015S-Lastname-Name.doc (or docx) for swe577  
SWE578-Paper-Progress-2015S-Lastname-Name.doc (or docx) for swe578

30% of your grade will be based on the progress report, 70% will be based on the final term paper.

**C) Your final term paper should be e-mailed to your paper advisor (due 18 May 2015) & should include:**

- A brief introduction, motivation
- Description of the selected system/software/application/phenomena
- Benefits, challenges, solutions related to the selected topic.
- Conclusions: Briefly: Maturity of the technology, open problems as future work if there are any.
- Cited References: List full bibliographic, information about papers and any additional references you may have resorted to. Be sure that all these references are cited in your paper. Do not include secondary references (references in other references) you have not seen.

**Name Your File:** SWE577-Paper-2015S-Lastname-Name.doc (or docx) for swe577  
SWE578-Paper-2015S-Lastname-Name.doc (or docx) for swe578

- You may check electronic resources in our library such as: <http://ieeexplore.ieee.org> and <http://info.scopus.com/>
- You may also want to see the link for technical writing at <http://www.cs.columbia.edu/~hgs/etc/writing-style.html>

**Note:** If you have questions related to your research project, please contact with the advisor of your project.

## Candidate Term Paper Titles:

### By Suzan Üsküdarlı:

1. **Tracking and Analyzing Social Media:** Social media content is of interest to marketers, politicians, news organizations, researchers, and many more. The sheer volume and differing characteristics of social media content present interesting opportunities and plenty of challenges. You will investigate the contexts, issues, and approaches that are being utilized and/or studied with regards to tracking and analyzing social media.
2. **Ontologies in Healthcare:** Ontologies are used for modelling domain knowledge. The short definition of an ontology is a specification of a conceptualization. They are used for sharing and reusing knowledge. One of the domains where the use of ontologies is increasing is in healthcare for the purpose of using common terminology with a shared understanding. They are applied in mapping vocabulary, computer based reasoning, etc. Standardization in health related information has many interesting applications and is of significant interest to this domain. You will investigate the development and use of ontologies as it relates to healthcare. You will provide examples of such use cases. Finally, you will describe a selected ontology and how it is used.
3. **Linked Data:** Linked data keeps track of interrelated documents. It aims to expose, share, and connect data and information on Semantic Web in terms of URI (Uniform Resource Identifier) and RDF (Resource Definition Framework). It encompasses many domains including music, news, geography, people, encyclopedic information, dictionaries, and much much more. RDF documents are machine processable. You will examine the content on Linked Data and the best practices for exposing and relating this information. You will explore efforts related to publishing and utilizing such content.
4. **Argumentation for the Social Semantic Web:** Argumentation is the study of using reasoning to come to a conclusion of multiple parties. The social web and semantic web (social semantic web) is a rich context where argumentation has many interesting applications. Social Media models must focus on informal argumentation models. Web 2.0 and especially Web 3.0 content render argumentation of such content very useful. You will explore the models for informal reasoning on social semantic web. You will explore tools and applications that make use of argumentation. You can consider this a World Wide Argument Web.
5. **Federated Social Web:** Federated social web refers to enabling connections among differing social networking communities via clean interfaces. The world wide web is a federated set of documents, where documents are stored on different computers, but accessed via a clean interface through browsers. Federated systems ease participation as participants have more control and flexibility. Social Networking systems mostly act as if no other social web systems exist, making it difficult to interact and share content and presence across social web environments. You will examine the current state of social web, motivation, issues and challenges (technical and social) in creating a federated social web. (See the W3C workgroup at <http://www.w3.org/community/fedsocweb/>)
6. **Web annotation:** W3C Web Annotation Group (<http://www.w3.org/annotation/>) has been developing an annotation model for annotating web content. Their aim to enable web content producers and consumers to include annotations of the content and thereby providing a mechanism to enrich and further utilize the content. This is a kind of tagging application, which supports a more comprehensive framework for managing the annotations. Applications vary between medical annotations to annotations on comments of social networking applications. You will examine and report on the state of the art approaches and applications of web annotation.

### By Alper Şen:

7. **Survey of Software Testing:** Software testing consume a large portion of the software development process. Testing techniques differ based on the type of software. The student will investigate the literature on software testing techniques. Potential topics include GUI testing, API testing, Web testing, Cloud testing, Model based testing, Search based testing,
8. **Survey of Android Malware Detection:** Android has become the most popular mobile platform. The number of malwares in the Android platform is growing exponentially leading to security and privacy concerns. The student will investigate the literature on Android malware detection.

9. **Survey of Software Bug Prediction:** It is crucial to determine whether a given software contains bugs or not. There are incomplete methods that use machine learning techniques to determine the existence of potential bugs.
10. **Survey of Data Race Detection:** Data races are common problems in multithreaded programs. There are static and dynamic data race detection techniques in the literature.
11. **Survey of Software Coverage Metrics:** Coverage plays a big role in determining the quality of the software. Various metrics are developed to assess whether the testing process can be converged or not. The student will investigate the literature on software coverage techniques and describe the state of the art used for various types of software.
12. **GPGPUs for Parallel Programming:** This will be a survey about the usage of GPGPUs for parallel computation.
13. **Survey of Regression Testing:** Regression testing is commonly used in many software projects when new versions are generated. Various approaches exist to reduce the cost of testing for regression testing.
14. **Survey of Software Evolution:** Software changes over time. It requires maintenance that involves addition of new features and debugging potential problems. A potential topic includes mining software repositories.
15. **Survey of Software Parallel Patterns:** OO design patterns have become very popular. Similarly, for parallel applications parallel design patterns are gaining popularity. The survey includes the applications of parallel design patterns.

**By Haluk Bingöl:**

16. **Interoperability standards and adoption practice standards for distributed learning.**
17. **Learning Tools Interoperability**
18. **How to grade an exam/homework in a programming language course automatically.** Methods, standards, best practices.
19. **Content Packaging Specification**
20. **Best practices on how to teach object oriented programming.**

**By Fatih Alagöz:**

21. **Green Software Engineering:** Myth or Fact: We need energy-efficient systems in order to protect our environment, cope with global warming, and facilitate sustainable development. It was estimated that 3 percent of worldwide energy consumption was caused by the ICT infrastructure that generated about 2 percent of the worldwide CO2 emissions. When I thought more about green subject, I believe that the software engineering should also put some effort for greening itself. That is, software products should provide the greenest solution. What do you think? The student will investigate the idea of green software engineering.
22. **Challenges in Agile Global Software Development:** Dissatisfaction with the overheads involved in software design methods of the 1980s and 1990s led to the creation of agile methods. These methods: Focus on the code rather than the design, are based on an iterative approach to software development, and are intended to deliver working software quickly and evolve this quickly to meet changing requirements [Sommerville]. However, there are a number of challenges. The student will prepare a survey on challenges in Agile Global Software Development.
23. **A Survey on COCOMO:** Cost estimation techniques for software may be experience-based, where managers judge the effort required, or algorithmic, where the effort required is computed from other estimated project parameters. The COCOMO II costing model is an algorithmic cost model that uses project, product, hardware and

personnel attributes as well as product size and complexity attributes to derive a cost estimate [Sommerville]. The student will prepare a survey on COCOMO.

24. **A Survey on Software Defined Radio Networks:** A Software-Defined Radio (SDR) system is a radio communication system where components that have typically been implemented in hardware (i.e. mixers, filters, amplifiers, modulators/demodulators, detectors. etc.) are instead implemented using software on a personal computer or other embedded computing devices. While the concept of SDR is not new, the rapidly evolving capabilities of digital electronics are making practical many processes that were once only theoretically possible [Wikipedia]. The student will prepare a survey of Software Defined Radio Networks.
25. **Biometric techniques for personal identification:** Personal identification technology is applied to a wide range of systems including area-access control, PC login, and e-commerce. Biometric techniques for personal identification have been attracting attention recently because traditional methods such as keys, passwords, and Personal Identification Numbers (PIN) have problems in terms of theft, loss, and reliance on the user's memory. Over the past few years, various biometric systems such as face, fingerprint and iris images have been developed for personal verification purposes to overcome these disadvantages. The student will prepare a survey on this topic.
26. **Authentication for biometric personal identification:** Personal identification technology is applied to a wide range of systems including area-access control, PC login, and e-commerce. Biometric techniques for personal identification have been attracting attention recently because traditional methods such as keys, passwords, and Personal Identification Numbers (PIN) have problems in terms of theft, loss, and reliance on the user's memory. Over the past few years, various biometric systems such as face, fingerprint and iris images have been developed for personal verification purposes to overcome these disadvantages. However, authentication to biometrically supported systems is also a research topic that should be carefully addressed. The student will prepare a survey on this topic.
27. **A survey on Digital Watermarking for digital voice copyright:** Digital watermarking is the process of embedding information into a digital signal. The signal may be audio, pictures or video, for example. If the signal is copied, then the information is also carried in the copy. In invisible watermarking, information is added as digital data to audio, picture or video, but it cannot be perceived as such. An important application of invisible watermarking is to copyright protection systems, which are intended to prevent or deter unauthorized copying of digital media. Steganography is an application of digital watermarking, where two parties communicate a secret message embedded in the digital signal. Annotation of digital photographs with descriptive information is another application of invisible watermarking. While some file formats for digital media can contain additional information called metadata, digital watermarking is distinct in that the data is carried in the signal itself. [Wikipedia] The student will prepare a survey of Digital Watermarking for digital voice copyright.
28. **Survey on Domain-Driven Design:** Domain-driven design (DDD) is an approach to developing software for complex needs by deeply connecting the implementation to an evolving model of the core business concepts [Wikipedia]. DDD combines design and development practice by modeling the core logic of an application. It introduces common design principles to reflect the domain and the domain logic of the business problem. DDD consists of a set of patterns for building enterprise applications. The student will prepare a survey on Domain-Driven Design.
29. **A survey on Hadoop:** Hadoop is a framework for running applications on large cluster built of commodity hardware. The Hadoop framework transparently provides applications both reliability and data motion. Hadoop implements a computational paradigm named Map/Reduce where the application is divided into many small fragments of work, each of which may be executed or re-executed on any node in the cluster. In addition, it provides a distributed file system (HDFS) that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. Both Mapreduce and the Hadoop Distributed File System are designed so that node failures are automatically handled by the framework. The student will prepare a survey on Hadoop.
30. **Are we ready for cashless systems?** : Swedes are among the technologically savvy people on the planet, based on their high rates of use of mobile phones and internet banking. In fact, only 3% and 7% of all financial transactions in Sweden and USA, respectively, are made using cash these days. E.g. public transport tickets are pre-paid using

mobile phones, and pay for everything from grocery to buying your car. The student will prepare a survey on different payment techniques for cashless society.

31. **Code Cloning Problems in Software Maintenance:** Code cloning can be dangerous for two main reasons: maintenance manpower may be increased due to duplicated codes, and inconsistent changes to cloned code may result with incorrect program behavior. What is the reason for code cloning? The student will prepare a survey on code clone detection techniques.
32. **Cloud Computing:** Cloud computing gives opportunity to the users with flexible services. The security, privacy, and reliability of the cloud based services are still under quest. The student will prepare a survey on cloud computing challenges and solutions.

**By Taflan Gündem:**

33. **Column Store Database Systems:** They store data tables as several columns of data rather than as rows of data. They have benefits in some applications such as data warehouses.
34. **Personalized Data Management:** There are various aspects of personalization in databases. For example, in preference queries the same query may receive differing results based on personal preferences of different persons. In context aware data management, the query processor considers the context in processing the query. The context may express conditions on situations external to the database such as time, location, etc.
35. **Recommender Systems:** They provide content that is likely to interest users, based on current and/or past user behavior.
36. **Data quality with Fusion and Cleaning:** Multiple (Web) data sources may provide overlapping, complementary, and sometimes contradictory information about the same concept or real world entity. Techniques such as resolution, mapping, fusion, and data cleaning are used in obtaining a clean and unified data.
37. **Self-Managing Databases:** What are the results of research in auto-tuning and self-managing database systems? How successful are they?
38. **Data Provenance:** Data provenance refers to the process of tracing and recording the origins of data and its movement between databases.
39. **Data Management using Modern Storage Hardware:** What are the important recent storage media? How do database management systems utilize modern storage media such as flash memories?
40. **Crowd Databases:** Some types of computations are easier for human beings. For example the person or city associated with a photograph. The queries associated with such computations are listed and the persons who answer such a question is paid some money. Example Amazon's Mechanical Turk.
41. **Privacy preservation in databases:** Some type of data such as illness of a patient is considered private information. Thus it should not be revealed to the public. However some other types of information should be revealed to the public for various reasons such as research activity. For example the number of persons having a certain kind of illness in a certain city should be revealed without revealing the names of the patients. What are the techniques used in privacy preservation?
42. **Spatial and Spatial-temporal Databases:** They store space and time related information. What are the storage and query processing algorithms for such databases?
43. **Result Diversity:** If a search is not specific, the results returned should be different from each other to give the user a change to choose. What is done for result diversity in databases?
44. **Software requirements for health information systems:** Health information system is a wide area of db applications. It includes patient records among others. There are various studies on the peculiarities of health information requirements specification. A survey of important recent papers in this area would be interesting.

45. **Privacy preservation in requirements:** How do privacy requirements affect the rest of the requirements? How do they affect the object and operational models for the functional requirements?
46. **Adaptive requirements:** Requirements may change according to conditions during the execution of the system. Adaptive requirements research is about handling changing conditions in requirements specification. Summarize recent papers in this area.
47. **Internal memory databases:** We have large internal memories currently. Some database systems reside only in the internal memory. What are some of the issues and solutions in internal memory databases.
48. **Stream databases:** In stream databases data is not persistent in contrast to regular database management systems. Data comes in a stream and is not stored totally in secondary storage. However the queries are relatively stable. They do not change very often.
49. **Stream Data warehouse:** Examine data warehouses in the stream database context. How are stream data warehouses created? What are the operations and how are they processed?
50. **Distributed Cloud DBMS:** Examine cloud dbms in distributed dbms context. What are the issues? What are the solutions proposed?
51. **Map Reduce:** Given an algorithm it is divided into parts and send to different processors for parallel processing. Then the intermediate results are combined to obtain the final result.
52. **Skyline Queries:** Special type of multi criteria optimization. Return objects that are not dominated by any other object in all attributes (or criteria) .
53. **Adaptive indexing:** Examine different types of adaptive indexes in the literature.
54. Any database or software requirements related survey.

**By Oğuz Tosun**

55. **Pair Programming vs other SW Quality Assurance Techniques:** Pair programming is a SW development technique in which two programmers (driver & observer) work together at a single computer frequently switching roles. Pair Programming practice is assumed to reduce the defect content (15% to 50% depending on task complexity and programmer experience) on the cost of doubling the total programmer-hours. The aim of the project is to survey the literature and compile empirical evidences comparing pair programming with other quality assurance techniques.
56. **Security and Reliability Issues in Cloud Computing:** Cloud computing is the delivery of computing as a service rather than a product, whereby shared resources (hardware and software) are provided to user as a utility over internet. As this new delivery model provides dynamically scalable and often virtualised resources, “security” and “reliability” are major concerns in cloud computing. The purpose of the project is to survey literature on these aspects of cloud computing including techniques like “risk-based security testing” & “Dynamic scheduling for cloud reliability” and compile a state of art report.
57. **Software Aging /Evolution :** Today’s society increasingly relies on software at all levels. Nevertheless, software quality generally continues to fall short of expectations, and software systems continue to suffer from symptoms of aging as they are adapted to changing requirements and environments. The only way to overcome or avoid the negative effects of software aging is by placing change and evolution in the center of the software development process. Research is being conducted on anti-aging , gracefully aging, run time sw adaptation etc to extend life time of SW. The purpose of the project is to survey literature on dealing with challenges of SW aging and compile a state of art report.

58. **Software Developer Network and Applications :** Social network metrics are extracted from collaboration graphs that model the development effort of a specific software product . The nodes of the graph usually consist of developers and an edge is drawn between nodes if a certain type of collaboration exists between them. Collaboration might be in the form of editing the same file during the same release; or checking out the same file from the repository;. Common social network metrics, such as degree, closeness and betweenness, are then calculated from the constructed collaboration graphs. The aim of the project is to do literature survey and comment on how well developer network represents development effort and what conclusions such as “predicting the defect content” may be drawn from such networks.
59. **Exascale computing : Reliability Challenge:** Over the past few years reliability has become a major issue for large Petascale systems and future Exascale ones. These systems will typically gather from half a million to several millions of CPU cores running up to a billion of threads. From the current knowledge and observations of existing large systems, it is anticipated that Exascale systems will experience various kind of faults many times per day. It is also anticipated that the current approach for tolerating such faults, which relies on automatic or application level checkpoint-restart, will not work because the time for checkpointing and restarting will exceed the mean time to failure of a full system. The challenge is finding new approaches, to run applications until their normal termination despite the essentially unstable nature of Exascale systems. The problem is widely researched since community has only five to six years to solve the problem. The purpose of the project is to survey literature on these aspects of “Exascale resilience” and compile a state of art report.
60. **Object Oriented (OO) Software Metrics :** A set of special metrics are defined to measure the complexity of OO Software. Goal of the project is to survey the literature on those metrics proposed and write a report on how to monitor and measure or calculate those OO metrics and provide a discussion if any of those metrics are correlated. Extracting those metrics on a sample OO program will be quite usefull to be integrated into teaching material.
61. **Software fault injection :** Software fault injection may have different objectives like “test assesment” , “reliability estimation” etc. The aim of the project is to survey the literature to gather information about different applications of the concept and the associated fault injection tools developed like “Ferrari”, “Doctor”, “orchestra”, “Grid-FIT” etc. And write a report guiding user on the selection of right tool.
62. **Code Cloning and its effect on Software Maintenance:** Code clones are the code fragments that are used with or without modification after being copy pasted from one part of the software project to another part. This “multiple copies” phenomenon creates problems in software maintenance. The purpose of the project is to survey literature on the matter and write a report on pros & cons of code cloning and the associated tools like clone detection” etc.

**By Fikret Gürgen:**

63. **Survey of Emotion Recognition:** Emotion recognition is a popular area. Human emotions are categorized by classification techniques using various inputs such as speech, video, etc. The student will investigate the literature on emotion recognition techniques.
64. **Survey of Human Computer Interaction (HCI) Techniques:** HCI has become an important issue with variety of techniques. Various HCI techniques are used in different applications such as speech, face recognition, etc. The student will investigate the literature on HCI techniques.
65. **Survey of Computer based Medical Diagnosis Applications:** Computer based Medical Diagnosis Applications are commonly used for general purpose medical diagnosis. It is important to summarize the applications with used classification techniques. The student will investigate the literature on Computer based Medical Diagnosis.
66. **Survey of Human Biometric Techniques:** Human Biometric techniques are commonly used in security systems. These techniques are speech, face, fingerprint, handprint, etc. Various techniques are developed to verify human identity for different applications. The student will investigate the literature on verification using human biometric techniques.

67. **Survey of Software Engineering Reliability Issues:** Software engineering algorithms are commonly used in various applications. Reliability is a key issue for these applications. Giving a brief summary of the standards and reliability in these systems is essential to the presentation. The student will investigate the literature on reliability of software engineering systems.

68. **Survey of Telehealth Systems in Software Engineering:** Telehealth systems are commonly used in today's medical applications. These techniques may be remote monitoring of health, telesurgery, etc. Various software implementations are developed to be used in these applications. The student will investigate the literature on telehealth applications of software engineering and give brief presentations of issues.

**By A. Taylan Cemgil:**

69. **Machine Learning pipelines and big data appliances:** Investigate and experiment with novel data processing environments for machine learning on big data: Hadoop, Storm and/or Spark. The project will include the implementation of a basic ML algorithm on the platform and measuring its performance.

70. **Analysis and visualization of Social Media Data:** In recent years, social media platforms such as Twitter and Facebook have become very popular. The generated data is huge and dynamic. Exploitation of this rich but highly unstructured data became a focus in current research. In this project, you will investigate data analysis in publicly available twitter data, develop and implement methods for analysis and visualization.

71. **Sentiment analysis in social Media Data:** Sentiment analysis is important for deriving meaningful statistics from bulk social media data. For example, merely reporting the number of tweets containing a particular hashtag is not sufficient for measuring the public opinion; a breakdown in terms of positive, negative and neutral opinions is much more informative. With this vision, you will investigate approaches such as topic models that have potential to boost performance when combined with modern supervised classification methods.

72. **Recommendation Systems and topic models:** Topic models have become one of the key tools in recommendation systems for collaborative filtering. In this project you will investigate and implement basic methods such as nonnegative matrix/tensor factorizations or latent Dirichlet allocation and test the results on benchmark datasets.

73. **Analysis of Network flows:** The analysis of network usage by individuals is of big interest for internet service providers and mobile operators. However, privacy concerns reduce the possibility for deep packet inspection. This project will investigate statistical methods for detection and segmentation of user characteristics from simple statistics such as packet sizes and counts and time dependent

74. **Statistical methods for detection and prevention of Distributed Denial of Service (DDoS) attacks:** DDoS attacks are costing companies and enterprises literally millions in terms of lost profit and user satisfaction. In this project, you will investigate methods for detecting and preventing such attacks using statistical techniques such as anomaly detection and outlier analysis. The project will involve a survey of DDoS attacks and their evolution over the times.

75. **Large Scale Machine Learning for Music Copyright Management:** The amount of raw data from TV and radio broadcast in audio-visual archives is ever-increasing. Monitoring this data stream online is critical in diverse applications ranging from media copyright management, speech transcription and recognition to hearing aids. Indexing and extracting relevant information in such data streams while carefully managing computational resources is a promising but yet largely unexplored area. We have a large dataset available from a music copyright management organisation. The data set consists of audio tracks (about 250 GB) and one day of broadcast data from 40 TV channels (audio-video). The goal is developing fast methods for indexing and retrieving hybrid data.

76. **Machine learning algorithms on a GPU:** CUDA is NVIDIA's parallel computing architecture. It enables dramatic increases in computing performance by exploiting the architecture of the GPU. In this project you will learn the CUDA architecture and GPU programming and Investigate parallel computing for common tasks in machine learning and data mining such as solving matrix factorisation, clustering and classification problems.

77. **Machine learning algorithms on Distributed Systems:** Distributed computation is key for scaling up algorithms to massive datasets. Modern computer systems such as clusters are typically hybrid systems where a cluster of multiple machines consists of shared memory multicore systems. In this project, you will learn and practice how to tailor specific algorithms for such architectures using MPI and OpenMP programming.
78. **Implementation of a real-time pitch tracker and/or a real-time tempo tracker as a VST plugin:** Steinberg's Virtual Studio Technology (VST) is an interface for integrating software audio synthesizer and effect plugins with audio editors and hard-disk recording systems. VST and similar technologies use Digital Signal Processing to simulate traditional recording studio hardware with software. Thousands of plugins exist, both commercial and freeware, and VST is supported by a large number of audio applications. The goals of this project are - learning the C VST SDK - MIDI and Audio Programming - implementation of algorithms for real time pitch/tempo tracking and MIDI conversion.

By M. Ufuk Çağlayan

79. **Comparative Evaluation of IOS and Android API's:** Existing resources Apple IOS and Google Android API's. for A survey of major functions of Apple IOS and Google Android API's. Criteria to evaluate these two important API's. The student should provide a detailed comparative evaluation of Apple IOS and Google Android API's.
80. **Survey of Service Level Agreements (SLA's):** What is Service Level Agreement? What are the important aspects of Service Level Agreement? How is the compliance with a Service Level Agreement assessed? What are the contractual and technical issues? The student should provide a detailed review of Service Level Agreement issues in relation to computer networks services and specifically cloud services.
81. **Comparative Evaluation of xDSL Technologies:** Existing xDSL Technologies, past and present. Where are the historical roots of xDSL Technologies. User side and service provider side equipment. Criteria to evaluate xDSL Technologies. Future trends. The student should provide a detailed comparative evaluation of existing xDSL Technologies.
82. **Survey of SDL and Z.100 as Specification Notation:** Overview of SDL and Z.100. What are the important properties of SDL and Z.100? A survey of existing SDL tools and their comparative evaluation.
83. **Survey of Cloud Models and Services:** What is the cloud model? Overview of Cloud Models and Services. What are the important properties of cloud services? Cloud layers, deployment models, architectures and overall issues, especially the security issues. A survey of existing cloud services and their comparative evaluation.
84. **Survey of Issues in Cyber Physical Systems:** What are the cyber physical systems? Overview of existing research on cyber physical systems. Low and high level design issues. A survey of existing cyber physical systems and their comparative evaluation.
85. **Survey of Common Criteria Standards and Compliance:** What is Common Criteria? What are the historical roots of Common Criteria? An overview of Common Criteria levels. How is the software security compliance with Common Criteria assessed? The student should provide a detailed review of Common Criteria issues in relation to secure software development.
86. **Systems Analysis for Secure Software:** What are the issues in Systems Analysis for Secure Software? Who does what in this area? The student should provide a detailed review of systems analysis approaches for secure software development.
87. **Secure Software Design:** What are the issues in Secure Software Design? Who does what in the area of Secure Software Design? The student should provide a detailed review of Secure Software Design approaches.
88. **IPv6 Support and IPv4 to IPv6 Transition Approaches:** The student should provide a detailed review of IPv6 current status and IPv4 to IPv6 transition approaches. IPv4 to IPv6 transition approaches such as ISATAP, 6to4, Teredo, etc for LINUX and Window environments need be covered and comparatively evaluated.
89. **Overview of RFID Technology from Software Engineering Point of View:** The student should provide a detailed review of RFID technologies from software engineering point of view. Development environments need be covered and comparatively evaluated.
90. **Overview of Satellite TV Smartcards:** The student should provide a detailed review of satellite TV smartcards and comparatively evaluate existing satellite TV smartcard technologies and related software.
91. **Evaluation of Java Card Platform:** Java Card is an industry-standard technology platform that was developed by Sun Microsystems to enable Java-based applications - applets - to run on smart cards that support this standard. Java Card helps developers build, test and deploy smart card-based applications quickly and efficiently with an object oriented programming model and off-the-shelf development tools. For smart card issuers, it delivers a

secure and interoperable platform that can be used to store and update multiple applications on a single end-user device. The student should provide an overview of Java Card Platform and evaluate its pros and cons with respect to similar smart card development platforms.

92. **SIM Application Toolkit:** The SIM Application Toolkit is a set of commands which define how the card should interact with the outside world and extends the communication protocol between the card and the handset. With SIM Application Toolkit, the SIM/USIM card has a proactive role with the handset (this means that the SIM initiates commands independently of the handset and the network). In 2G networks, SIM Application Toolkit (SAT) was defined in GSM 11.14 standard. From release 4 onwards, GSM 11.14 is replaced by 3GPP 31.111 which also includes specifications of USIM Application Toolkit (USAT) for 3G networks. The student should provide a detailed review of SIM Application Toolkit and the available software tools in this area.
93. **Overview of Smart Card Standards Including ISO 7816:** The student should provide a detailed review of ISO 7816 Smart Card Standard and other related smart card standards.
94. **OpenSC Project:** OpenSC provides a set of libraries and utilities to work with smart cards. Its main focus is on cards that support cryptographic operations, and facilitate their use in security applications such as authentication, mail encryption and digital signatures. The student should provide a detailed review of the OpenSC Project <http://www.opensc-project.org/opensc>.
95. **Overview of Transportation Smartcards:** The student should provide a detailed review of transportation smartcards and comparatively evaluate existing transportation smartcard technologies such as OYSTER and especially AKBIL, and their security aspects.
96. **Smart Card Based Medical and Health Systems:** The student should provide a detailed review of smartcard use in medical and health systems and comparatively evaluate existing medical and health systems using smartcard technologies
97. **Overview of Global Positioning System (GPS) and Related Hardware/Software Tools:** The student should provide a detailed review of existing and future GPS infrastructures and present a comparative evaluation of existing hardware and software tools.
98. **GPS Based Intelligent Guided Vehicles With Collision Prevention:** The student should provide a detailed review of techniques used in GPS based intelligent guided vehicles with collision prevention.
99. **Hardware/Software for Heart Beat Rate/EKG Monitoring and Recording:** The student should provide a detailed review of existing hardware and software systems to monitor and record heart beat rate and EKG's in humans.
100. **Detection Of Cardiac Disorders by Hardware/Software:** The student should provide a detailed review of analysis techniques already implemented in software to diagnose cardiac disorders.
101. **Detection Of Epileptic Disorders by Hardware/Software:** The student should provide a detailed review of analysis techniques already implemented in software to diagnose epileptic disorders.
102. **Body Area Networks Hardware/Software:** The student should provide a detailed review of current and future body area network technologies.
103. **Survey of Wiki Software Tools in Public Domain:** Wiki software tools in public domain are very important for collaborative work. There are various wiki software tools for different hardware and OS platforms. The student will prepare a survey of wiki software tools in public domain.
104. **Secure Software Development APIs:** The student should provide a detailed review of Secure Software Development APIs in public domain. APIs for LINUX and Window environments need be covered and comparatively evaluated.
105. **Formal Representation of Software Design Products:** What are the issues in formal representation of software design products? Is UML sufficiently formal to represent software design products? What other formal representation techniques exist? The student should provide a detailed and comparative review of software design formal representation approaches.
106. **A Survey of Software Change Management Tools:** What are the issues in Software Change Management? Which Change Management software tools exist commercially and in public domain? The student should provide a detailed and comparative review of existing Change Management software tools.
107. **A Survey of Software License Enforcement and Anti-Theft Mechanisms:** What are the issues in software license enforcement and anti-theft mechanisms? Which software tools exist commercially and in public domain? The student should provide a detailed and comparative review of existing software license enforcement and anti-theft mechanisms.

108. **A Survey of Software Quality Measurement Metrics:** What are the issues in software quality measurement metrics? Which software tools exist commercially and in public domain? The student should provide a detailed and comparative review of existing software quality measurement metrics and related software tools.
109. **A Survey of Software Project Duration Estimation Methods:** What are the issues in software project duration estimation? Analysis, design and implementation/testing durations for large and complex software. Which software tools exist commercially and in public domain? The student should provide a detailed and comparative review of existing software project duration estimation methods and related software tools.
110. **A Survey of SDL, Z.100 and Related Software Tools:** What are the issues in SDL, Z.100 and related software tools? Which software tools exist commercially and in public domain? The student should provide a detailed review of SDL and Z.100 and also comparative review of existing SDL/Z.100 software tools.
111. **A Survey of Web Crawlers:** What are the issues in web crawlers? Which software tools exist commercially and in public domain? The student should provide a detailed review of web crawler issues, crawler prevention techniques and also comparative review of existing web crawlers.
112. **A Survey of Microsoft Windows Operating System Evolution:** The student should provide a detailed review of the evolution of Microsoft Windows Operating Systems, starting with Windows 3.1. Complexity and the properties of Windows API should especially be emphasized.
113. **A Survey of LINUX Operating System Kernel Evolution:** The student should provide a detailed review of the evolution of LINUX Operating Systems kernels up to our day. Complexity and the properties of LINUX API should especially be emphasized.
114. **A Survey of Smartphone Processors:** The student should provide a detailed review of smartphone processors, mainly ARM processors and their competitors. Increased complexity of smartphone processors should be examined comparatively.
115. **A Survey of Open Software Tools:** The student should provide a detailed survey of open software tools, categorizing them according to business and industry sectors to which the open software tool is targeted.
116. **Use of Model Checking Techniques for Verification of Distributed Software Specifications-SPIN Tool and Promela:** Model checking exists for almost 20 years, to verify the integrity of software specifications at any level, that is requirements, design and implementations. The most frequently used approach is by Holzmann, the Promela specification language and SPIN verification tool. There are also other approaches like SMV and UPPAAL. The student will prepare a survey of the Promela specification language and SPIN Tool.
117. **Use of Model Checking Techniques for Verification of Distributed Software Specifications-SMV:** Model checking exists for almost 20 years, to verify the integrity of software specifications at any level, that is requirements, design and implementations. One frequently used approach is SMV verification tool. The student will prepare a survey of SMV verification tool.
118. **Use of Model Checking Techniques for Verification of Distributed Software Specifications-UPPAAL:** Model checking exists for almost 20 years, to verify the integrity of software specifications at any level, that is requirements, design and implementations. One frequently used approach is UPPAAL verification tool. The student will prepare a survey of UPPAAL verification tool.
119. **Survey of Game Software Development Environments:** Game software is an important business area and there are various game software development environments for different hardware and OS platforms. The student will prepare a survey of game software development environments.
120. **Survey of Network Management Software Tools in Public Domain:** Network management software tools in public domain are very important to monitor and manage a large scale network with hundreds of servers, client computers and active network devices. There are various network management software tools for different hardware and OS platforms. The student will prepare a survey of network management software tools in public domain.
121. **A Survey of Microsoft Windows Application Development Tools:** The student should provide a detailed survey of Microsoft Windows application development tools, categorizing them according to specific programming languages in which an application will be developed. Both commercial and open systems should be included.
122. **A Comparative Evaluation of UML Tools:** The student should provide a detailed and comparative survey of UML tools, both commercial and open system.

123. **A Survey of Microsoft Azure Cloud Application Development Tools:** The student should provide a detailed survey of Microsoft Windows Azure cloud application development tools, both commercial and open system.
124. **Comparative Evaluation of Amazon and Microsoft Azure Clouds:** The student should provide a detailed and comparative survey of Amazon cloud services and Microsoft Windows Azure cloud.

**By Pinar Yolum**

125. **Collaborative Filtering:** Collaborative filtering is a method for recommendation; mostly used for e-commerce Web sites. It exploits the fact that similar people act similarly. Explain how the main algorithm works and discuss its variations. Investigate amazon.com in light of collaborative filtering. Discuss how the site is using collaborative filtering. Pick a Web site that does not use recommendation and explain in detail how collaborative filtering can be applied there.
126. **Content-Based Filtering:** Content-based filtering is a method for recommendation. It is based on building an accurate user model and making recommendations based on user's previous actions. Explain how the main algorithm works and discuss its variations. Investigate amazon.com in light of content-based filtering. Discuss how the site is using content-based filtering. Pick a Web site that does not use recommendation and explain in detail how content-based filtering can be applied there.
127. **Semantic Search:** Searching for information is important for any Web site. Traditional way of searching is keyword-based search where the user enters keywords and the systems searches for the exact words in the documents. Investigate how search can be done more intelligently. Explain what semantic search is. Discuss how semantics of words can be used to improve search accuracy. Describe traditional metrics to measure search accuracy. Suggest a new metric to measure the accuracy for semantic search. Justify your choice.
128. **Gamification:** Gamification is the process of taking an application and turning it into a game as to engage more users. Duolingo is a prime example that uses gamification. Investigate different types of gamification. What are the different techniques for gamification? Take an application that does not currently apply gamification and design it as a game. Describe in details the rules, incentives, and interaction styles.
129. **Privacy Checking:** For many years, computer systems have been concerned with developing techniques for preserving their users' privacy. Similarly, with social applications (such as Facebook) the need still exists. However, social applications enable users to upload more private content and share them with others in various ways, thereby leading to new types of privacy breaches. What are typical and atypical privacy violations in social systems? What are existing languages for specifying users' privacy requirements? What are the current mechanisms for managing users' privacy and checking if privacy violations take place? Discuss possible tools that can be developed to preserve users' privacy better.
130. **Agent-Oriented Software Engineering:** What is agent-based software engineering? How is it different than object-oriented software engineering? What are the existing methodologies? Pick one of the methodologies (such as Gaia or Tropos) and analyze it to explain its strengths and weaknesses.
131. **Ontology-Based Medical Applications:** Ontologies have been used for knowledge representation for many years in AI applications. Recently, they have become part of several practical applications, especially in medical systems to represent medical domain concepts and their relations. What are examples of medical systems that use ontologies? What kind of inferences are part of these systems? Which ontology languages are used? How does the use of ontologies increase the capabilities of these systems and how do they affect their performance? This project will survey and critically discuss ontology-based medical applications.
132. **Linked Data:** What is Linked Data? Why does it exist? What are different languages for Linked Data? Explain Resource Description Framework (RDF) in detail. Give examples of things you can express in RDF. Study <http://linkedmdb.org/> in terms of linked data. Discuss an example application you can build using linkedmdb. Discuss how Linked Data is useful for the application
133. **Big Data Storage:** Traditional relational databases are not effective for storing unstructured big data. Research on why traditional DBMSs are not enough. What are the technical challenges for storing big data in relational

DBMSs? Investigate various systems that are available for storing big data, such as Hadoop and Spark. Explain their design principles. Discuss their advantages over relational DBMSs.

**By Cem Ersoy:**

134. **State of the Art E-health Software: Are we there yet?** E-health is a quickly developing area of ICT and related software in which the medical state of the individuals are monitored and the necessary alarms are reported. In this study, the state of the art ICT software and systems will be surveyed. The capabilities of the current and near future e-health applications will be reported.
135. **Participatory Collaborative Smartphone Applications:** Availability and popularity of very powerful smart phones made participatory collaborative smartphone applications possible. For example, a municipality can more efficiently collect the sound map of a metropolitan city by collecting geo-tagged decibel values from mobile phone users without actually sending special teams to many different locations in the city. In this work, related ICT technologies and already implemented software applications will be surveyed and their capabilities, current and future applications will be reported.
136. **How smart is the smart grid? Challenges in smart grid software solutions:** The energy supply of the future will be characterized by decentralized and fluctuating generation of renewable energy. This creates a challenge for balancing energy demand and energy generation. The increasing injection of renewable energy in medium and even low voltage networks demands new solutions from the grid operators. In this study, the emerging demands and resulting challenges in the smart grid management software will be surveyed. The survey should include a classification of the control system functions required for smart grid management operations.
137. **Context Awareness for Smart Phone Users:** Smartphone users are growing quickly. Some smartphones have eight sensors. It is possible to identify the context of the user such as the mode of transportation (e.g., in a bus, train, on a bicycle). Moreover, activities of smart phone users can be identified such as walking, running, standing. In this survey, capabilities of smartphones and related software and future directions are identified for context awareness.
138. **Intelligent transportation software solutions: From traffic safety to information retrieval:** Intelligent Transport Systems (ITS) applications mainly aim to provide innovative services for transport and traffic management and enable users to be better informed and make safer, more coordinated, and 'smarter' use of transport networks. In this survey, the software aiming for different modes of transport and traffic management regarding traffic safety and coordination will be surveyed together with the network-to-user and user-to-user facilities provided for user safety and comfort.
139. **Automotive software solutions: Intelligence in the cars:** Although the cars had been considered as luxury devices in the past, recently the transportation mainly relies on personal vehicles to access to our work, school and leisure time activities. Being so involved in our lives, the automotive industry have involved the intelligence to the cars to improve driver experience and comfort by either involving telematics in the operation of car, or providing a more sophisticated software for user comfort. In this survey, the in-car software solutions will be studied regarding the basic functionalities provided to the users, user interaction and experience.
140. **Mobile Social Network applications: Ubiquitous communication in mobile world:** The capabilities of mobile devices and ubiquitous Internet access enable an uninterrupted access to the social networks. Such connectivity encourages the social network software developers to integrate the mobile device properties with the services provided for mobile devices. In this study, the mobile social networking software will be surveyed, regarding the services provided for mobile device capabilities.
141. **Wearables for Health Applications:** Wearables communicating with smartphones are getting smarter with new types of sensors, better processing capacities and extended batteries. This survey will be covering applications for everyday healthy lifestyle as well as applications targeting specific diseases such as diabetes or Parkinson's Disease.

**By Cem Say:**

142. **Other Kinds of Software:** So you have learned programming PCs, and can build software systems. Is this all? What about alternative abstract computational devices, such as cellular automata, where we have not one, but massive numbers of computers that communicate with their neighbors in the population, but each individual processor is extremely weak and simple? Do our notions of programming even work for such "machines"? The student will investigate the relevant theoretical literature and come up with an answer.

**By Can Özturan:**

- 143. Tools and environments for parallel software development**
- 144. Parallel programming patterns**
- 145. Transactional Memory**
- 146. Review of Cloud Computing Infrastructures**
- 147. Software engineering for multicore systems**
- 148. Applications of the block-chain technology**
- 149. Review of graph based databases**
- 150. Applications of graph partitioning algorithms**