Overview:
Each student is required to form a two-member team to work on a research project: conducting review on an advanced topic in software engineering. At the end of the semester, each group will give a 45-minute professional presentation to the entire class. The presentation serves two purposes: 1) to allow other students in the class to learn about the software engineering topic being reviewed; 2) to allow the instructor to evaluate how well a team does in the research project.

There are two types of advanced topic review projects.
• The first type is research-oriented, like the traditional literature review. The goal is to cover a topic and review the “state of the art” for that topic. For this type, “state of the art” and breadth are crucial.
  o Example: The State of the Art in End-User Software Engineering.
  o Example: Software Engineering in Industrial Automation: State-of-the-Art Review
• The second type is learning-oriented. It covers a useful, interesting, and relatively new topic and introduces it using well-designed examples to help others to have in-depth understanding of the subject. For this type, depth and understandability are crucial.
  o Example: Mobile Apps development
  o Example: Game software development

After the teams and topics are decided, we will have a preview of all the literature reviews before the final presentations start. In the preview, each team needs to give an overview description of its topic (in about 2 minutes) and show the list of references to the class. A final report of your literature review is required. The length of your final report will be 6-8 pages. The format of your report should follow the IEEE style (Times New Roman, 10 font size, double columns).

More requirements for the research-oriented literature review (the first type):
The literature review should cover both general background and two or three specific methods/technologies that represent the current research on the selected topic. Each team should read at least 5 journal or conference papers related to the selected topic and write a final report based on the literature review.
• It is important for your review to cover the state of the art of the selected topic. To ensure your review is about “current” research, it is required the main papers that you review are published in the recent 3 years (that is, year 2014, 2015, and 2016). If we use the fuzzy words new and old (e.g., new means something in the recent 3 years), below is a rough guidance for a selected topic and the kind of technologies that are reviewed for the topic:
  New technology applied to an old topic --- OK
  New technology applied to a new topic --- OK
  Old technology applied to a new topic --- OK
  Old technology applied to an old topic --- Not good
• This is an advanced class and needs research “depth” in your review. Do not spend all your review on the general introduction of the selected topic; your review should have a balanced coverage between general introduction and specific methods/technologies. The selected papers to
be reviewed should be technical oriented, e.g., from IEEE or ACM journals or conferences (see resources below).

- It is encouraged to review theories, methodologies, and practices of software engineering related to emerging applications in recent years (the emphasis should be on the software engineering aspect, not the application itself).
- Your review of the selected papers should be coherently integrated. It works against your scores if your review simply covers several disjoined aspects of a topic.

More requirements for the learning-oriented literature review (the second type):
For learning-oriented review, at least two references should be conference/journal papers (e.g., covering the general topic or the specific tools/techniques); the others can be white papers, tutorials, web links, etc. The review cannot simply be a demonstration of a tool or a walkthrough of an example. The review should also cover information such as motivation, underlying models/theory, limitations, and applicability of the topic that is being reviewed. If the topic being reviewed is “old” (e.g., software testing), strong justification is needed in order to convince the instructor and classmates that the content being covered is “useful, interesting, and representing state-of-the-art technology”.

Software engineering topics
As a reference, a list of software engineering topics can be found from ICSE 2016: http://2016.icse.cs.txstate.edu/researchTrack

Resources (list not complete)
Note: most papers can be accessed through IEEE digital library and/or ACM digital library
- IEEE Transactions on Software Engineering
- ACM Transactions on Software Engineering Methodology
- Springer Software and Systems Modeling
- IEEE Software
- ICSE, the International Conference on Software Engineering (http://www.icse-conferences.org/)
- ASE, IEEE/ACM International Conference on Automated Software Engineering (http://www.ase-conferences.org/)
- FSE/ESEC, the ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE) (http://www.sigsoft.org/fse20/)
- OOPSLA, Object-Oriented Programming, Systems, Languages & Applications, (http://www.oopsla.org/oopsla-history/)