Limited-cost team is formed by applying the *Skunk Works*[1] to develop a software prototype with new promising technology to get confidence in it. Core members of this team choose specialists based on their track record and broader interests by applying the *Self Selecting Team*[2]. Compatible members into this team are also selected by choosing co-workers with similar temperaments employing *Diverse Groups*[3]. Team has thanks to the application of *Unity of Purpose*[4] the clear vision about what software product they are going to build. High-level manager in the role of *Patron Role*[5] mentors and helps this team with special issues. Team grows naturally and transforms into the small software project team created by applying *Size The Organization*[6]. Core members hired from the start of the team know most about the software under development and are domain experts. But team needs more specialists, that are hired by applying *Phasing It In*[7]. New hires are turned into the experts by assigning them mentors through *Apprenticeship*[8] program. Program utilities used by the team are developed and maintained by the *Solo Virtuoso*[9]. Software modules are tested by pairs programmer-tester applying *Developing In* Pairs[10]. Team is structured into groups of people working on part of the system or given functional requirements applying *Holistic Diversity*[11]. This way, expertise in domain and functional areas is created. This is a consequence of applying *Domain Expertise In Roles*[12]. Domain expertise is formed in specialized groups responsible for product's sub-modules according to *Subsystem By Skill*[13]. Knowledge of the most experienced staff is transfered to novices by applying *Moderate Truck Number*[14]. Software components are tested early on at time when functional requirements for the sub-module are clear, according to *Software Design Is Bounded By Test Design*[15]. According to *Group Validation*[16], team strives to test product internally before handing it to the Quality Assurance sub-organization in the company. Separate Quality Assurance organization is devised in the company according to *Engage Quality Assurance*[17] *that* tests outputs from the development process by also communicating with potential users of the product. Expectations regarding business functions of the new software product are using *Scenarios Define Problem*[18] documented in use cases. Voice of the potential users is important and therefore provided as further insights and feedback for the development team by applying *Engage Customers*[19]. But according to *Fire Walls*[20] customers do not have a direct contact with developers fearing it would jeopardize successful finish of the software project by having chance to update schedule and its tasks. Information about the software project leaks into the rest of the organization on a need-to-know basis by applying *Gate Keeper*[21]. Software project team is based on the *Team Pride*[22] proud about the innovative software product they develop. Initial members of the skunk works team are *Legend Role*[23] regarded as people with most knowledge about the potential of the product and its usage. Members of the team need to socialize and *Matron Role*[24] is responsible for well-being of its members. New *Public Character*[25] role serves team to help with other social processes. There is a *Wise Fool*[26] role representing member of the team who sometimes points to uncomfortable truth during status update meetings. Project team is rewarded applying *Compensate Success*[27] according to success of the project itself. In the retrospective, it is found and acknowledged that it is good high-level managers in the company trusted the idea and did not cancel development of this product early on as in the *Failed Project Wake*[28]. Information about the time the product is deployed is conveyed to general public through the *Surrogate Customer*[29].