Project Management Plan (PMP)

1. Inleiding

1.1 projectomschrijving

Het project betreft een applicatie die de mogelijkheid biedt om meerdere huizen te beveiligen d.m.v. de Arduino Uno. Op elementair niveau bestaat het systeem uit drie componenten: De arduino plus laptop, de server, en de gebruiker. De gebruiker zal zijn huis kunnen beheren via zijn pc en wordt gemeld via SMS als er een inbraak is. Mogelijke uitbreidingen voor het systeem zijn bijvoorbeeld het nabootsen van een realitische situatie met x aantal huizen, een hoofdbeheerder, en een videocamera en/of microfoon die de inbreker kan identificieren.

2. Projectuitvoering

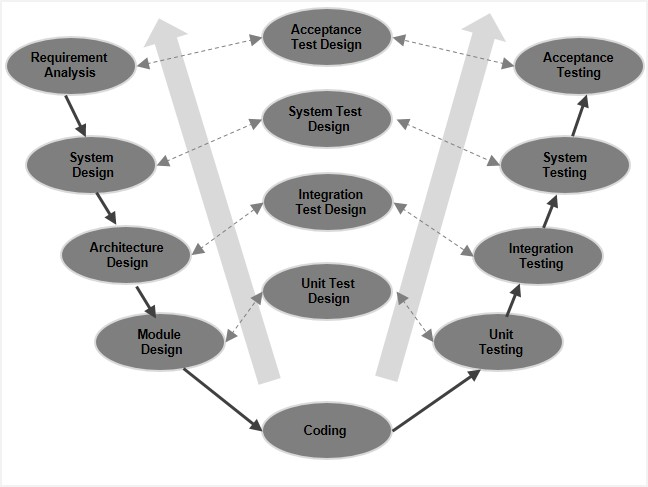
2.1 Software Development Cycle Lifespan Model

Het gekozen ontwikkelmodel is het iteratief model. Dit model is gekozen voor een aantal redenen, sommige redenen met een hogere prioriteit dan de ander. Een van de belangrijkste punten is [1] het feit dat er onzekerheid is in de benodigde tijd en dus risiconiveau van de te bouwen features. Door het gebruiken van het iteratief model kunnen de factoren tijd en risico beter beheert worden. [1.1] Een geanticipeerd probleem is dat er te veel tijd in een specifieke feature zou gestopt worden, wij willen dus de mogelijkheid van tijdsnood tegen gaan. Door uit te breiden op vorige versies kunnen we het ons permiteren om uitdagerende requirements op te stellen voor het project. [2] Wijziging van requirements en het toevoegen van features om het project natuurlijk te laten groeien en uitbreiden. [4] De vooruitgang kan makkelijk gemeten worden, elke versie is dus een milestone die bereikt moet worden.

2.1.1 Wijzigingen betreft het gekozen SDCL model

Na overleg op maandag 9 September met Michael Franssen, onze begeleider van dit project, zijn er een aantal wijzigingen gemaakt in het originele SDCL plan. [1] In der plaats van volledig het iteratief model te gebruiken wordt nu het V-model gebruikt als het primaire SDCL model, i.e., voor de documentatie en testing. [2] De begeleider gaf als argument om het V-model te gebruiken om ons bekwaam te maken met een formeel ontwikkelingsmethodiek. Dit alhoewel onze beredenering solide en onderbouwd in onze keuze was. [3] Het iteratief model wordt gebruikt in de implementatiefase van het V-model. De redenen en onderbouwing voor het gebruik van het iteratief model zijn eerder benoemd. Zie dus ook deze eerder benoemde punten in paragraaf 2.1 hier voor.

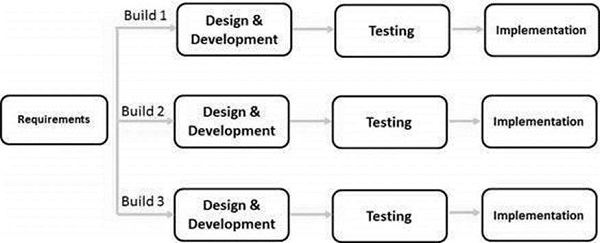
Op de volgende pagina staat het diagram van het V-model. Voor een planning en hoeveel tijd wij alloceren per fase zie de projectsplanning.



2.2 Aanpak betreft implementatiefase

Zie het hieronder weergeven diagram om de aanpak van de implementatiefase te zien. Let op dat dit een generiek model is en niet een accurate weergave is van hoe wij exact te werk gaan,. Het presenteerd wat het iteratief model is en hoe men in het algemeen dit model toepast. De wijzigingen die wij op dit model maken zijn [1] de documentatie en testing fase beschouwen wij als excessief en zullen niet uitgevoerd worden. De sub-componenten van Design & Development en Testing worden dus uitgevoerd volgens het V-model.Wat blijft is het bouwen van meerdere builds en het maken van de requirements voor dat de implementatie begint. Zie V-model en project planning wanneer de requirements worden gemaakt. Hieronder de verschillende fases die wij momenteel voor ogen zien.

[1] De eerste fase betreft het ontwikkelen van een minimum viable product, oftewel MVP. In het MVP zullen wij de basis en verticale funtionaliteit bouwen. [2] De tweede fase betreft het verbeteren en het streamlinen van het huidige product. Denk hierbij aan het bouwen van een dashboard om het systeem te monitoren en te beheren. [3] De derde fase zal een van de uitdagerende functies toegevoegd worden zoals bijvoorbeeld het sturen v.d. sms berichten naar de gebruiker. De derde fase kan herhaald worden als er genoeg tijd is om meer complexe features toe te voegen e.g., geluids- en video-opname bij inbraak. [4] Tot slot zal in de laatste fase het systeem geadapteerd worden om in de echte wereld gebruikt kunnen worden. Hiermee zullen we proberen om een echte situatie na te bootsen. Denk hierbij aan meerde huizen die onder bewaking staan van het systeem. Een hoofdbewaker die het complete systeem monitoort, en gebruikers die via een dashboard hun instellingen kunnen aanpassen. Het doel is dus om een narrative en business use te kunnen demonstreren i.p.v. een geïsoleerd product.



3. Projectbeheersing

3.1 Versiebeheer

Git is geselecteerd als onze “version control system”, hiermee zullen dus de verschillende versies van de applicatie en elke kleine incrementele verandering gedocumenteerd worden. De primaire reden dat Git is gekozen is door de gemakkelijke toegangbaarheid van de remote repository en het maken van verschillende feature branches.