## Anforderungen

[I.1.1 Änderungsgeschichte 1](#_Toc327434545)

[I.1.2 Tools 2](#_Toc327434546)

[I.1.3 Funktionale Anforderungen 2](#_Toc327434547)

[I.1.4 Nicht-funktionale Anforderungen 9](#_Toc327434548)

[I.1.4.1 Funktionalität 9](#_Toc327434549)

[I.1.4.1.1 Angemessenheit 9](#_Toc327434550)

[I.1.4.2 Zuverlässigkeit 9](#_Toc327434551)

[I.1.4.2.1 Reife 9](#_Toc327434552)

[I.1.4.3 Benutzbarkeit 9](#_Toc327434553)

[I.1.4.3.1 Verständlichkeit & Erlernbarkeit 9](#_Toc327434554)

[I.1.4.3.2 Bedienbarkeit 9](#_Toc327434555)

[I.1.4.3.3 Attraktivität 9](#_Toc327434556)

[I.1.4.4 Effizienz 10](#_Toc327434557)

[I.1.4.4.1 Zeitverhalten 10](#_Toc327434558)

[I.1.4.5 Änderbarkeit & Wartbarkeit 10](#_Toc327434559)

[I.1.4.6 Übertragbarkeit 10](#_Toc327434560)

[I.1.4.6.1 Austauschbarkeit 10](#_Toc327434561)

[I.1.4.6.2 Installierbarkeit 10](#_Toc327434562)

[I.1.5 Design Constraints 10](#_Toc327434563)

[I.1.6 Zugänglichkeit (Accessibility) 10](#_Toc327434564)

### Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Änderung | Autor |
| 18.05.2012 | 1.0 | Erste Version des Dokuments | CH |
| 19.05.2012 | 1.1 | Review Accessibility | DT |
| 29.05.2012 | 1.2 | Funktionale & Nicht-funktionale Anforderungen | CH |
| 29.05.2012 | 1.3 | Review Funktionale Anforderungen | DT |
| 04.06.2012 | 1.4 | Design Constraints | CH |
| 04.06.2012 | 1.5 | Nicht-funktionale Anforderungen | DT |
| 04.06.2012 | 1.6 | Tools | DT |
| 05.06.2012 | 1.7 | Tools | CH |
| 09.06.2012 | 1.8 | Review und Korrekturen | LE |
| 11.06.2012 | 1.9 | Korrekturen | CH |
| 12.06.2012 | 1.10 | Review | DT |

### Tools

Zur Durchführung der Arbeit und Entwicklung der Videowall wurden die nachfolgend aufgelisteten

Werkzeuge verwendet.

|  |  |  |
| --- | --- | --- |
| Tool | Version | Link |
| Windows 7 | SP 1 | <http://windows.microsoft.com/de-CH/windows/home> |
| Tortoise SVN | 1.7.4 | <http://tortoisesvn.net/> |
| Adobe Reader | X | <http://get.adobe.com/de/reader/> |
| .NET | 4.0.30319 | <http://www.microsoft.com/net> |
| Kinect SDK | 1.5 | <http://www.microsoft.com/en-us/kinectforwindows/> |
| Visual Studio 2010  Ultimate mit Power-Tools | 10.0.40219.1 SP1Rel | <http://www.microsoft.com/visualstudio/11/de-de> |
| ReSharper | 6.1 | <http://www.jetbrains.com> |
| dotCover | 1.2 | <http://www.jetbrains.com> |
| GhostDoc | 3.0 | <http://submain.com/download/ghostdoc/> |
| Expression Blend | 4.0.20525.0 | <http://www.microsoft.com/expression/products/Blend_Overview.aspx> |
| WPF Inspector | 0.9.9 | <http://www.wpftutorial.net/Inspector.html> |
| NDepend Trial | v4.0 | <http://www.ndepend.com/NDependDownload.aspx> |
| Adobe Photoshop | CS4 Extended | <http://www.adobe.com/de/products/photoshop.html> |
| Adobe InDesign | CS4 Extended | <http://www.adobe.com/de/products/indesign.html> |
| Adobe Illustrater | CS4 Extended | <http://www.adobe.com/de/products/illustrator.html> |
| Microsoft Office 2010 | 14.0.6112.5000 | <http://office.microsoft.com/de-ch/> |
| Redmine | 2.0.1 | <http://redmine.org> |
| Matrox PowerDesk | 1.14.183.508 | <http://www.matrox.com/graphics/de/products/multi_display_software/powerdesk/> |

Tabelle 1 - Tools

### Funktionale Anforderungen

Um die funktionalen Anforderungen möglichst effizient und trotzdem exakt zu definieren, wurden User Stories als Teil von Scrum verwendet. Nachfolgend sind die User Stories nach Sprint gruppiert. Die User Stories sind mit dem jeweiligen Sprint, in welchem sie umgesetzt wurden, gekennzeichnet. Nicht umgesetzte User Stories sind mit „U“ markiert. Weitere Details sind in der Tabelle User Stories im Anhang (TODO Verlinkung Anhang) zu entnehmen.

Nachfolgend eine Übersicht über die User Stories: Legende: U -> Uplanned

| Titel | User Story | Definition of Done | Sprint |
| --- | --- | --- | --- |
| Poster werden angezeigt | Als HSR-Besucher möchte ich die Poster über die Bachelorarbeiten lesen können, um Eindrücke über die Schule und die Möglichkeiten, die man an dieser Schule als Student hat, zu erhalten. | Ein Poster wird angezeigt und kann gelesen werden. | Bis SP7 |
| Poster browsen | Als Videowall Benutzer möchte ich durch alle Poster browsen können, damit ich einen Eindruck von den Arbeiten erhalte. | Durch Drücken einer Vor- und Rückwärtstaste kann durch die Liste der Poster navigiert werden. | Bis SP7 |
| Handcursor wird dargestellt | Als Videowall Benutzer möchte ich sehen, wo sich meine Hand auf der Wall befindet, damit ich fähig bin, die Wall zu bedienen. | Ein Cursor wird dargestellt und bewegt sich analog zur Hand des Benutzers. | Bis SP7 |
| Eigenes Skelett wird dargestellt | Als Videowall Benutzer möchte ich in Form eines Skeletts sehen, dass die Videowall mich erkannt hat, damit ich nachverfolgen kann, wie das System auf meine Bewegungen reagiert. | In der Applikation wird das Skelett des Benutzers dargestellt, das sich zu den Bewegungen des Benutzers dynamisch anpasst. | Bis SP7 |
| Sofortiges Erfolgserlebnis für Einstieg sichergestellt | Als neuer Videowall Benutzer möchte ich sofort ein Erfolgserlebnis erleben, weil ich einen positiven ersten Eindruck haben möchte. | Das sofortige Erfolgserlebnis (Skelett wird dargestellt) ist dokumentiert. | Bis SP7 |
| Handcursor schön dargestellt | Als Videowall Benutzer möchte ich als Cursor eine Hand sehen, damit mir sofort klar ist, dass ich die Applikation mit der Hand steuern kann. | Der Cursor wird als kleine Hand dargestellt (analog XBox-Spiele). | SP8 |
| Skelett schön dargestellt | Als Videowall Benutzer möchte ich ein Skelett mit Verbindungslinien der einzelnen Punkte (Hand, Ellbogen, Hüfte etc.) sehen können, damit ich meine Bewegungen besser nachvollziehen kann. | Vom Benutzer wird ein Skelett mit Verbindungslinien wird dargestellt. | SP8 |
| Handcursor ruckelt weniger 1 | Als Videowall Benutzer möchte ich, dass der Cursor weniger ruckelt, damit ich besser auf die Buttons drücken kann. | Da der Input des Skeletal Trackings ungenau ist und über einen etwas hohen Jitter verfügt, wird für das Hand Tracking jeweils der Mittelwert der letzten paar Aufzeichnungen verwendet. Somit ruckelt die Hand dann weniger. | SP8 |
| Video wird dargestellt | Als Entwickler möchte ich testen, dass Videos auf der Videowall laufen, damit ich weiss, dass dies problemlos funktioniert und später bei Bedarf Videos eingebunden werden können. | Ein Video kann mittels einer WPF-Applikation abgespielt werden und auf der Test Hardware/Mitsubishi Wall getestet werden. | SP9 |
| Applikation ist mit linker Hand bedienbar | Als Videowall Benutzer möchte ich die Applikation auch mit der linken Hand bedienen können, damit mir die Bedienung leichter fällt. | Die Applikation soll automatisch erkennen, wenn die linke Hand höher als die rechte Hand gehalten wird, und dann die Kontrolle an die linke Hand übergeben. Wird die rechte Hand dann wieder merklich höher als die linke Hand gehalten, so soll die Applikation wieder mit der rechten Hand bedient werden können. | SP9 |
| Ideen gesammelt wie Personen werden von Videowall angezogen werden | Als Passant möchte ich beim Vorbeigehen an der Videowall etwas Interessantes sehen, damit mein Interesse geweckt wird und ich auf die Videowall zugehe und mit ihr interagiere. | Es werden mindestens zwei verschiedene Ideen zusammengetragen, diskutiert und ausgewertet. Am Schluss wird die beste Idee ausgewählt und dokumentiert. | SP10 |
| Plug-in Möglichkeit | Als Student möchte ein App, das mit den vorgegebenen Schnittstellen funktioniert, für die Videowall schreiben, damit ich es später auf die Videowall hochladen kann. | In der Videowall-Applikation ist es möglich, dynamisch ein Plug-in zu laden und anzuzeigen. Für jedes geladene Plug-in wird ein Menubutton angezeigt. Dynamisch laden heisst, dass ein Plug-in (.dll) in den Ordner, in dem die Applikation gerade läuft, hineinkopiert werden kann und das Plug-in dann von der Applikation automatisch geladen wird. | SP10 |
| Mittagsmenu App in Plug-in umgewandelt | Als Plug-in Entwickler möchte ich ein Beispiel für ein Plug-in ansehen können, damit ich weiss, wie ich vorgehen muss, wenn ich ein Plug-in entwickeln möchte. | Das Mittagsmenu Plug-in ist in einem eigenen Projekt und die Videowall-Applikation hat keine Referenz zu diesem Projekt. Das Plug-in (Mittagsmenu) wird von der Videowall App aber automatisch hineingeladen, siehe "Plug-in Möglichkeit". | SP11 |
| Poster App in Plug-in umgewandelt | Als Plug-in Entwickler möchte ich ein Beispiel für ein Plug-in ansehen können, damit ich weiss, wie ich vorgehen muss, wenn ich ein Plug-in entwickeln möchte. | Das Poster Plug-in ist in einem eigenen Projekt und die Videowall-Applikation hat keine Referenz zu diesem Projekt. Das Plug-in (Poster) wird von der Videowall App aber automatisch hineingeladen, siehe "Plug-in Möglichkeit". | SP11 |
| Demomodus (Verfolgung von Passanten) Kraftfeld besprochen und dokumentiert | Als Videowall Benutzer möchte ich einen Demomodus sehen, der meine Aufmerksamkeit auf die Wall zieht und ich so beginne, mit ihr zu interagieren. | Die ausgesuchte Idee mit dem Krafteld wurde im Team besprochen und dokumentiert. | SP11 |
| Demomodus: Vom Demomodus wird in den Interaktionsmodus gewechselt | Als Videowall Benutzer möchte ich, dass die Applikation vom Demomodus in den Interaktionsmodus wechselt, nachdem ich eine Zeit lang (z.B. 5 Sekunden) gewartet habe, damit ich mit der Applikation interagieren kann. | Die Applikation wechselt, nachdem für 5 Sekunden ein Skelett erkannt wurde, in den Interaktionsmodus. Es darf kein "Flackern" auftreten. | SP11 |
| Demomodus: Vom Interaktionsmodus wird in den Demomodus gewechselt | Als Videowall Benutzer möchte ich, dass die Applikation vom Interaktionsmodus in den Demomodus wechselt, nachdem eine Zeit lang (z.B. 10 Sekunden) kein Skelett erkannt wurde, damit ich durch den Demomodus von der Videowall angezogen werde. | Die Applikation wechselt, nachdem für 10 Sekunden kein Skelett erkannt wurde, automatisch in den Demomodus. Es darf kein "Flackern" auftreten. | SP11 |
| Demomodus: externes Design erstellt | Als Videowall Benutzer möchte ich einen ansprechenden Demomodus, damit mich die Videowall optisch anspricht und ich mit dieser interagieren will. | Das in WPF umgesetzte externe Design wurde vom Auftraggeber nach einer Demonstration abgenommen. | SP11 |
| Bild der Hand ist auf die rechte bzw. linke Hand abgestimmt | Als Videowall Benutzer möchte ich für meine Hand das passende Handsymbol sehen, damit das Symbol die Hand zeigt, mit der die Bedienung gerade stattfindet. | Ist die rechte Hand aktiv, so wird das Symbol der rechten Hand angezeigt. Sobald die Bedienung zur anderen Hand wechselt, so wechselt auch das Symbol. | SP11 |
| Deployment Entwickler PC | Als Entwickler möchte ich die entwickelte Applikation auf dem lokalen PC öffnen können, damit die Applikation weiterentwickelt werden kann. | Der Entwickler kann die Solution öffnen und die Applikation ausführen, ohne dass diese abstürzt. | SP12 |
| Plug-in Schnittstelle definiert | Als Entwickler möchte ich eine Plug-in Schnittstelle definieren, damit Interessierte Apps für meine Applikation erstellen können. | Das Interface oder die Interfaces für das Plug-in System sind implementiert. | SP12 |
| Demomodus: Demotext zu aktiver App wird angezeigt | Als Videowall Benutzer möchte ich beim Demomodus einen ansprechenden und interessanten Text sehen, damit dieser mein Interesse weckt und ich mit der Videowall interagieren will. | Für jede App existiert ein Demotext (durch das Interface IApp), der jeweils im Demomodus durch ein Binding auf die aktuelle App angezeigt wird. | SP12 |
| Demomodus: Apps werden automatisch gewechselt | Als Videowall Benutzer möchte ich Texte von verschiedenen Apps sehen, damit ich die App wählen kann, die mich am meisten anspricht. | Während dem Demomodus wird alle 20 Sekunden die aktive Applikation gewechselt, wodurch auch automatisch der Demotext gewechselt wird. | SP12 |
| Navigation mit schönen "Tabs" dargestellt | Als Entwickler möchte ich dem Benutzer die Navigation zwischen den Ansichten mit Tabs ermöglichen, damit auf einen Blick ersichtlich ist, welche Ansichten/Informationen die Wall zur Verfügung stellt und wie von der einen in die andere gewechselt werden kann. | Externes Design ist implementiert. | SP12 |
| Das Mittagsmenu wird angezeigt | Als Mensabesucher möchte ich gerne das aktuelle Mittagsmenu sehen, damit ich dies nicht erst in der Mensa am kleinen Tisch tun muss. | Das Mittagsmenu wird in einem gut lesbaren Format angezeigt. | SP12 |
| Mittagsmenu App automatisch aktualisiert | Das Mittagsmenu soll automatisch aktualisiert werden. | Das aktuelle Menu wird von http://hochschule-rapperswil.sv-group.ch/de/menuplan.html geladen, sobald die Videowall-Applikation gestartet wird. | SP12 |
| Administration der Videowall Inhalte definiert | Als Betreiber des Systems möchte ich, dass ein möglicher Ablauf für die Administration (Erstellung, Löschung, Mutationen) definiert ist, damit ich weiss, was für Systeme dazu noch entwickelt werden müssen. | Es liegt eine Dokumentation vor, wie die Daten der Videowall bearbeitet werden können. | SP13 |
| Externes Design festgelegt und validiert | Als Entwickler möchte ich für die Design User Stories eine "Definition of Done" festlegen können, damit der Abschluss der User Stories validiert werden kann. | Externes Design ist festgelegt und validiert. | SP14 |
| Deployment Videowall Server | Als Entwickler möchte ich die entwickelte Applikation auf den Videowall Server deployen können, damit die Applikation dann darauf läuft. |  | U |
| Deployment Web | Als Entwickler möchte ich die entwickelte Applikation auf den Web Server deployen können, damit die Applikation dann darauf läuft. |  | U |
| Sekretärin kann Poster verwalten | Als Sekretärin möchte ich die Poster verwalten können, damit die dargestellten Poster auf der Wall immer auf dem aktuellsten Stand sind. |  | U |
| About View | Als Bewunderer der Applikation möchte ich sehen, wer diese Applikation unter der Leitung von wem entwickelt hat, weil mich interessiert, bei welchem Dozent eine so coole Arbeit realisiert werden kann. |  | U |
| Kinect Hand hervorgehoben | Als Videowall Benutzer möchte ich, dass ich meine aktive Hand sehe, damit ich weiss, dass ich die Applikation mit dieser Hand bedienen kann. |  | U |
| Handcursor dreht nicht bei aktivem Menu | Als Videowall Benutzer möchte ich, dass keine Handcursor Animation bei dem zurzeit aktiven Menu abgespielt wird, damit ich nicht meine, dass noch etwas passiert. |  | U |
| Am Handcursor soll erkennbar sein, ob man auf einem interaktiven Objekt ist | Als Videowall Benutzer möchte ich, dass der Handcursor, je nach dem, ob er sich über einem interaktiven Objekt (z.B. Button) befindet oder nicht, anders gekennzeichnet wird, damit ich weiss, wann ein 'Klicken' möglich ist. |  | U |
| Poster filtern nach Studiengang | Als Videowall Benutzer möchte ich die Poster nach Studiengang filtern, damit ich nur die Poster meiner Abteilung browsen kann. |  | U |
| News anzeigen | Als Videowall Benutzer möchte ich News/Headlines über das Thema meines Studiengangs sehen, da mich dies interessiert. |  | U |
| Sekretärin kann News Feeds der Wall verwalten | Als Sekretärin möchte ich die News Feeds der Wall verwalten können, damit die News Feeds auf dem aktuellsten Stand sind. |  | U |
| News mit QR-Code | Als Videowall Benutzer möchte ich neben den News für meinen Studiengang einen QR-Code sehen, den ich abfotografieren kann, damit ich später auf dem Computer nochmals die gleichen Headlines sehen und mich mittels Internetrecherche in die Themen vertiefen kann. |  | U |
| Studenten können App hochladen | Als Student möchte ich meine selbst entwickelte App hochladen können, damit sie dann auf der Videowall verfügbar ist. |  | U |
| Plug-in Dokumentation geschrieben | Als App Entwickler möchte ich eine Dokumentation zur Verfügung haben, damit ich nachschauen kann, wie ich beim Programmieren einer App vorgehen muss. |  | U |
| Informationen zu aktuellen Events werden auf der Videowall angezeigt | Als Eventbesucher möchte ich auf der Videowall aktuelle Informationen zum Event angezeigt haben, damit ich mich dort informieren kann. |  | U |
| Sekretärin kann Informationen zu Events verwalten | Als Sekretärin möchte ich Informationen zu aktuellen Events hochladen, damit sich die Eventbesucher auf der Wall informieren können. |  | U |
| Browsing Modus schön dargestellt | Als Videowall Benutzer möchte ich eine ansprechende Darstellung des Browsing Modus, damit mir die Applikation besser gefällt und mir die Bedienung mehr Spass macht. |  | U |
| Demomodus schön dargestellt | Als Videowall Benutzer möchte ich eine ansprechende Darstellung des Demomodus, damit mich dieser eher anspricht und ich die Applikation verwenden möchte. |  | U |
| Animationen bei Browsing Modus | Als Videowall Benutzer möchte ich eine Animation sehen können, wenn ich zum nächsten Poster navigiere, damit es für mich besser ersichtlich ist, dass das Poster gewechselt hat. |  | U |
| Animation bei Tab/Ansichten-Wechsel | Als Videowall Benutzer möchte ich eine Animation sehen können, wenn ich ein anderes Tab wähle, damit es für mich besser ersichtlich ist, dass die Ansicht gewechselt hat. |  | U |
| Avatar wird dargestellt | Als Videowall Benutzer möchte ich anstelle eines Skeletts einen Avatar sehen, damit ich sofort begreife, dass ich als Mensch erkannt bin. |  | U |
| Algorithmus zur Poster Sortierung | Als Videowall Benutzer möchte ich, dass aktuelle Poster öfter vorkommen, da mich diese mehr interessieren. |  | U |
| QR-Code wird pro Poster dargestellt | Als Videowall Benutzer möchte ich einen QR Code fotografieren, damit ich die Informationen von der Videowall auf meinem Mobiltelefon mitnehmen kann. |  | U |
| L Poster Problematik | Als Videowall Benutzer möchte ich die L-Poster lesen können, damit ich mich auch über die Arbeiten dieser Studienabteilung informieren kann. |  | U |
| Easteregg | Als Entwickler möchte ich beim Präsentieren der Wall ein Easteregg ausführen, damit meine Besucher und ich Spass haben. |  | U |
| Poster und Informationen sind auf einer Website verfügbar | Als Videowall Benutzer möchte ich Zusatzinformationen zu den Postern auf einer Website ansehen können, damit ich mithilfe eines fotografierten QR Codes (User Story: QR-Code wird pro Poster dargestellt) weitere Informationen zum Poster erhalte. |  | U |
| Sekretärin kann Zeitschlitze für Inhalte definieren | Als Sekretärin möchte ich zu verschiedenen Zeiten verschiedene Informationen auf der Videowall anzeigen. |  | U |
| Handcursor ruckelt weniger 2 | Als Videowall Benutzer möchte ich, dass der Cursor noch weniger ruckelt, damit ich besser auf die Buttons drücken kann. |  | U |
| Die Seetemperatur wird angezeigt | Als Student sehe ich die Seetemperatur, weil ich wissen möchte, wie angenehm der Sprung in den See sein wird. |  | U |
| Das Wetter wird angezeigt | Als Besucher sehe ich das aktuelle Wetter und das der nächsten zwei Tage, weil mich das derzeitige und zukünftige Wetter interessiert. |  | U |
| Sonneneinstrahlung wird angezeigt | Als gesundheitsbewusste Person möchte ich die Sonneneinstrahlung sehen, damit ich weiss, ob ich einen Hut anziehen muss. |  | U |
| Gesten können erkannt werden | Als Benutzer der Videowall möchte ich die Applikation mit Gesten bedienen können, damit so eine alternative Steuerung zum Handcursor existiert. |  | U |
| Zur Entspannung kann ein Spiel gespielt werden | Als HSR-Besucher möchte ich auch mal keine Informationen erhalten, sondern ein Spiel spielen, damit ich abschalten kann. |  | U |
| Kinect Winkel wird automatisch ausgerichtet | Als Betreiber des Systems möchte ich, dass die Kinect automatisch auf einen festen Winkel eingestellt wird, den ich in einer Konfigurationsdatei verändern kann, damit die Personen gut von der Kinect erkannt werden können. |  | U |
| Steuerung mittels Spracherkennung ist möglich | Als fauler Video-Wall-Nutzer möchte ich mittels Sprachbefehlen navigieren, weil ich zu träge bin, um mich zu bewegen. |  | U |
| Lesemodus Poster | Als Videowall Benutzer möchte ich nach der Auswahl eines Postertitels in den Lesemodus wechseln, damit ich das Poster besser lesen kann. |  | U |
| Rating | Als Videowall Benutzer möchte ich die gelesenen Poster raten, damit zukünftige Leser sehen, welche Poster ich interessant fand. |  | U |
| Kommunikation mit anderen Videowall Interagierenden | Als Videowall Benutzer möchte ich direkt mit anderen Videowall Benutzern (geografisch getrennt) kommunizieren können. |  | U |
| Zweiter Mitspieler hat Pointer | Als zweiter Mitspieler möchte ich, dass meine Hand als Pointer dargestellt wird, damit ich meine Mitspieler auf etwas hinweisen kann. |  | U |
| Interaktive Hilfe wird angezeigt | Als Videowall Benutzer möchte ich eine interaktive Hilfe sehen, die mir zeigen kann, was für Gesten ich in bestimmten Teilen der Applikation nutzen kann, damit ich nicht frustriert von der Wall davonlaufen muss. |  | U |
| Skelett ruckelt nicht | Als Videowall Benutzer möchte ich, dass das Skelett nicht ruckelt, damit ich vom Ruckeln nicht irritiert/abgelenkt werde. |  | U |
| Navigationshilfe | Als Videowall Benutzer möchte ich, nachdem ich mich für längere Zeit unschlüssig mit dem Cursor am selben Ort befinde, die Objekte, mit welchen interagiert werden kann, aufleuchten sehen. |  | U |

Tabelle 2 - User Stories

### Nicht-funktionale Anforderungen

Die nichtfunktionalen Anforderungen lassen sich zum Teil aus den User Stories ableiten. Einige Anforderungen, wie z.B. die Wartbarkeit, können jedoch nicht daraus abgeleitet werden. Deshalb ist es notwendig, diese festzuhalten.

#### Funktionalität

##### Angemessenheit

Die Videowall soll für alle Passanten einfach bedienbar sein. Die Inhalte sollen interessant sein und für jede Benutzergruppe etwas bieten. Die Wall soll die Aufmerksamkeit der Passanten wecken und die Nutzer durch attraktive und aktuelle Inhalte zur erneuten Nutzung der Videowall animieren.

Mit Usability Tests soll geprüft werden, ob die Applikation die Aufmerksamkeit der Passanten erlangen kann.

#### Zuverlässigkeit

##### Reife

Obwohl es sich die Videowall-Applikation um einen Prototyp handelt, werden gewisse Stabilitätsanforderungen an die Wall gestellt. Die Videowall soll 24 Stunden am Stück in Betrieb sein können, ohne dass die Applikation terminiert oder gravierende Memory Leaks enstehen, die zu einem Absturz führen könnten.

Dies ist durch einen Stabilitätstest zu belegen.

#### Benutzbarkeit

##### Verständlichkeit & Erlernbarkeit

Die Applikation muss beim ersten Kontakt sogleich verständlich sein. Ansonsten verliert der Nutzer schnell das Interesse an der Videowall. Er wird diese verlassen und sie auch zu einem späteren Zeitpunkt nicht erneut nutzen wollen. Die Bedienung muss intuitiv sein, damit der Nutzer nicht zuerst ein Handbuch lesen muss.

Die schnelle Verständlichkeit soll mit Usability Tests validiert werden.

##### Bedienbarkeit

Die Bedienung soll einfach und intuitiv sein und über die Bewegung der Hand geschehen. Die einzelnen Komponenten sollen über eine genügend grosse Fläche verfügen, sodass sie mit dem Handcursor treffsicher ausgewählt werden können.

Die Bedienung mit der Hand soll durch einen Usability Test verifiziert werden.

##### Attraktivität

Der Entwicklungsprozess ist so gestaltet, dass neben den Risiken auch die Benutzerbedürfnisse im Fokus stehen. Diese Bedürfnisse sollen mit Hilfe von Befragungen und Usability Tests erkannt werden.

Durch einen Demomodus soll der Nutzer auf die Applikation aufmerksam gemacht und angelockt werden. Die Darstellung des eigenen Skeletts soll auf den Nutzer ansprechend wirken und ihn dazu animieren, herauszufinden, wie er die Applikation mit seinem eigenen Körper steuern kann.

Das Skelett soll das sofortige Erfolgserlebnis für den Nutzer sicherstellen. Dies sowie die Attraktivität des Demomodus werden durch Usability Tests geprüft.

#### Effizienz

##### Zeitverhalten

Die Applikation soll innerhalb von fünf Minuten aufgestartet sein. Das Handtracking soll innerhalb einer Sekunde auf Benutzereingaben reagieren.  
Wurde ein Nutzer erkannt und bewegt dieser sich vor der Videowall, so soll die Reaktion des dargestellten Skeletts und des Handcursors auf diese Bewegungen so schnell erfolgen, dass der Nutzer das Gefühl der direkten Manipulation erhält.

Dies ist mit Usability Tests zu verifizieren.

#### Änderbarkeit & Wartbarkeit

Die Software wird zukünftig vom Institut für Software (IFS) weiterentwickelt. Damit dies einfach geschehen kann, soll auf die Codequalität geachtet werden, wobei ReSharper genutzt wird, um die Qualität zu prüfen.

Auch die Code-Metriken sollen beachtet werden. Ziel ist es, einen „Maintainability Index“ von mindestens 50% zu erreichen, dies auf Ebene Projekt.

Es soll eine einfache Möglichkeit geben, die Videowall dynamisch mit Inhalten zu erweitern. Um dies realisieren zu können, soll ein Plug-in System entwickelt werden.

#### Übertragbarkeit

##### Austauschbarkeit

Indem mit dem vorgegebenen Interface gearbeitet wird, können Applikationen für die Videowall unabhängig entwickelt werden. Das Plug-in-System ermöglicht das dynamische Hinzufügen von Inhalten.

##### Installierbarkeit

Um das Projekt weiterentwickeln zu können, soll es möglich sein, die Applikation auf dem lokalen Computer zu öffnen und auszuführen, ohne dass diese abstürzt.

### Design Constraints

Kinect ist durch die Aufgabenstellung als Inputgerät festgelegt. Es ist auch festgelegt, dass eine Monitorwand zu verwenden ist, und kein Beamer. Das Corporate Design der HSR gibt Richtlinien für das externe Design vor.

### Zugänglichkeit (Accessibility)

Der eigene Körper dient für die Videowall als Steuerelement. Für Personen mit einer körperlichen Behinderung ist die Applikation daher bedingt geeignet. Durch das Wizard of Oz - Experiment (siehe TODO Verlinkung Domain Analyse, Durchführen & Fazit) wurde das Konzept „Meine Hand ist die Maus“ bestätigt. Sofern der Nutzer einen Arm hat, ist die Bedienung daher gewährleistet. Es kann hierbei jedoch passieren, dass gewisse Punkte des Skeletts, welche für die Komplettierung des Skeletts fehlen, fehlinterpretiert werden. Hält man sich beispielsweise einen Arm hinter den Rücken, so können die Punkte des Ellbogens und der Hand nicht mehr erkannt werden und werden hier im Beispiel entlang der Hüfte angezeigt (siehe gelbe Punkte in Abbildung 1 - Handerkennung bei Arm hinter dem Rücken).



Abbildung 1 - Handerkennung bei Arm hinter dem Rücken

Die Applikation ist nicht auf Personen mit einer Sehbehinderung ausgelegt. Um diesen Personen eine optimale Bedienung der Videowall bieten zu können, müssten die Interaktionen mit akustischen Signalen beantwortet werden. Zu diesem Stand der Entwicklung sind Musik oder andere akustische Signale ausgeschlossen. Diese würden für Mitarbeiter, welche im Gebäude 4 arbeiten, störend wirken.   
Weiter kann die Applikation mit beliebig gestalteten, von interessierten Programmierern erstellten Applikationen erweitert werden. Es ist nicht gewährleistet, dass diese Erweiterungen für Personen mit partiell-funktionalen Sehbehinderungen optimiert sind.